UCLouvain

epl

École polytechnique de Louvain

Modelling the aeroelastic effects of wind turbine blades in the Actuator Disk method

Author: **Lucas Jeanson**
Supervisors: **Philippe Chatelain, Grégoire Winckelmans**
Readers: **João Almeida, Maud Moens, Hadrien Rattez**
Academic year 2023–2024
Master [120] in Mechanical Engineering

Remerciements

Tout d'abord, je tiens à remercier mes deux promoteurs, les professeurs Philippe Chatelain et Grégoire Winckelmans, qui m'ont permis de mener à bien ce travail et de prendre les bonnes décisions quant à la direction des sujets étudiés. Leurs précieux conseils m'ont aiguillé dans la bonne direction, et m'ont donné l'opportunité d'accomplir ce projet le plus qualitativement possible.

Je souhaite également adresser mes remerciements aux deux autres personnes ayant suivi ma progression tout au long de mon travail, le docteur Maud Moens, et tout particulièrement François Trigaux qui a été présent quotidiennement et a toujours repondu à mes questions dans les délais les plus brefs, tout en trouvant une solution dès qu'un problème se présentait. Je le remercie pour sa gentillesse et son soutien qui fut important, notamment durant la dernière partie du travail durant laquelle mes doutes étaient au plus haut.

Enfin, je voudrais exprimer ma gratitude envers ma famille, et surtout mes parents, qui ont toujours été présents pour m'encourager et me soutenir, même lorsque je me trouvais face à des problèmes qui me semblaient insurmontables.Leur soutien et leurs encouragements ont été inestimables, et je leur suis infiniment reconnaissant pour cela.

Technical acknowledgements Computational resources have been provided by the supercomputing facilities of the Université catholique de Louvain (CISM/UCL) and the Consortium des Équipements de Calcul Intensif en Fédération Wallonie Bruxelles (CÉCI) funded by the Fond de la Recherche Scientifique de Belgique (F.R.S.-FNRS) under convention 2.5020.11 and by the Walloon Region.

Contents

Chapter 1

Introduction

1.1 Wind energy and motivation

In an era where sustainable energy sources are pivotal to mitigate environmental degradation and meet escalating global energy demands, wind energy emerges as a compelling solution and stands as a cornerstone of renewable power generation.
Over the past two decades, the wind energy sector has exhibited remarkable growth, an upward trajectory set to continue. Globally, 2022 saw the integration of 77.6 GW of new wind power capacity into grids, resulting in a cumulative installed capacity of 906 GW, a 9 % rise compared to 2021, as seen in Figure 1.1. Furthermore, GWEC (Global Wind Energy Council) Market Intelligence forecasts 680 GW of new capacity in the next five years (2023-27), corresponding to a total growth of 75 % [1].

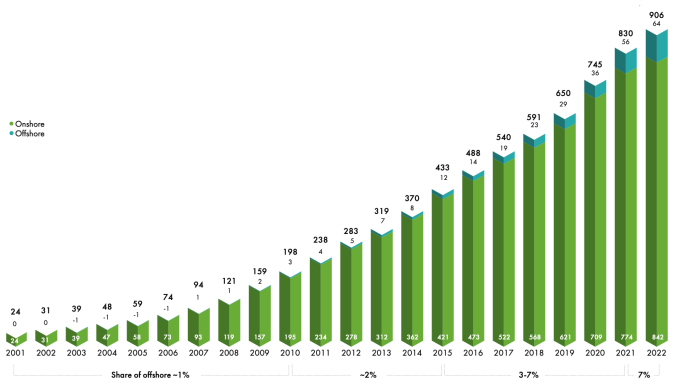


Figure 1.1 – *Historic development of global wind power generation [GW] [1].*

However, despite this robust growth, fossil fuels continue to dominate global power systems, constituting over 60 % of the 2022 global electricity generation, while wind-based energy accounts for 7.6 % [7]. On the other hand, according to projections by Ember, showed in Figure 1.2, fossil fuels could begin a new era of decline from now on.

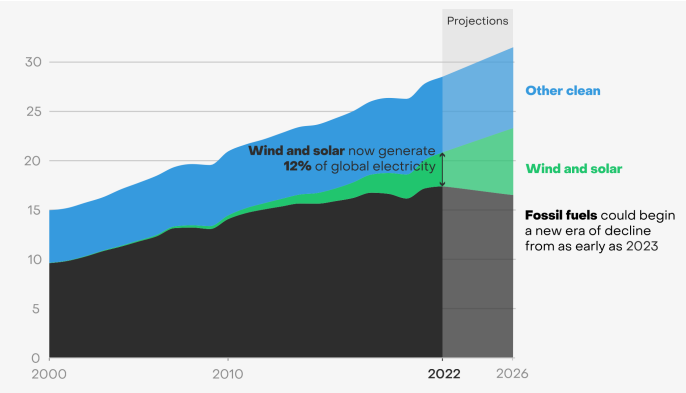


Figure 1.2 – *Evolution and projection of global energy generation, and distribution between sources [TWh] [7].*

An evolution worth noting is the rise of offshore wind turbines, demonstrating a notable increase from 3.5 % to 7 % among the total number of turbines installation between 2017 and 2022, corresponding to a staggering 237% rise in offshore installations. Distinguished from their onshore counterparts by their installation in shallow open waters, offshore turbines' development is essential as they offer several advantages:

- Offshore turbines generate more energy. There are two reasons for that, first, the size of the rotor can be bigger, meaning that more energy can be captured, but mainly because offshore wind speeds are more important than land speeds.
- They enjoy a less troublesome inconsistency of wind power and availability, providing in turn a more reliable and efficient power generation.
- They are less intrusive and have fewer environmental impacts than their onshore equivalents as they don't affect how towns use their land, or create noise pollution.
- You can build bigger offshore wind farms because there are fewer obstacles.

Despite all these advantages, the large majority of wind turbines are still built onshore because the costs of construction and maintenance for onshore wind farms are significantly lower than those involved in offshore farms.

In order to reduce the costs in offshore configurations, the turbines are often grouped together in large wind farms which brings some constraints. When wind turbines are close to each other, they often happen to sit in the wake of upstream turbines. The wake is the region aft of a wind turbine where the wind speed is decreased and the turbulence is important. When a wind turbine is located in this zone, it induces a drastic reduction of the power production (turbines affected by a wake generate approximately 40 % less power than their potential output [25]) as the air has already been extracted from its kinetic energy by the upstream turbines. Moreover, wakes from different turbines may interact and create particularly unstable and turbulent air, which can cause damage to the downstream wind turbines encountering the wake. In order to mitigate the negative impacts of wakes on downstream turbines, innovative control strategies are in development, such as wake steering that redirects wakes away from downstream turbines by adjusting the yaw or tilt angles of turbines [22, 26].

1.2 Objectives and outline

The purpose of this work is to develop a method that models the aeroelastic effects of the blades, producing accurate results while keeping reasonable computational cost in the perspective of being used for wind farm simulations. The method is then assessed through different grid resolutions and wind speeds, and compared to a reference model.

In order to accurately reproduce the behaviour of the flow, blades and wake, we use numerical tools that can produce different level of preciseness with the compromise that the higher the fidelity, the more important the computational cost. The interactions between wakes and wind turbines lead to complex behaviours that are hard to capture numerically and can become very costly. We therefore need to develop methods that tackle this trade-off efficiently to predict accurately the power production as well as the behaviour of the wind turbines in a farm when encountering adverse wakes. We will then be able to know how to control each turbine of the farm through their different physical parameters, such as yaw, tilt or pitch, to maximize the power output of the whole farm.

The tools we propose to use and that will be assessed in this study are based on Large Eddy Simulation (LES) [30] coupled to a numerical model for the wind turbine. LES is a method that resolves explicitly the large-scale turbulent structures while modelling the effects of smaller scales. This method strikes a balance between accuracy and computational cost, as it provides more detailed information than traditional Reynolds-averaged Navier-Stokes (RANS) models while being less computationally demanding than Direct Numerical Simulation (DNS) [29], which resolves all the scales of turbulence present in a flow.
The flow solver is an in-house developed fourth-order finite differences code using a subgrid scale

model to simulate the smaller scales in the flow. On the other hand, the rotor is modelled through an actuator model. The two most common approaches are used in this study and are the Actuator Disk (AD) model which can be used in coarse grid resolution, and the Actuator Line (AL) model requiring more refined resolution and imposing constraints on the time step.

As the focus is set towards wind farm simulations, requiring a lot of computations, the most appropriate method is the AD, allowing good precision from coarser resolution, and reasonable CPU need. Currently, the AD does not account for the aeroelasticity, i.e. the flexibility, of the blades which is a critical parameter to estimate the fatigue life and for the modellization of the wake. This work therefore aims to integrate the elasticity of the blades as a post-processing operation for the computation of different output of a simulation performed using the AD method. The accuracy of the results will then be assessed by comparing them to the reference AL model which is already validated and accounts for the aeroelastic effects of the blades.

Chapter 2 This chapter introduces the numerical tools used throughout this work. First, the flow solver and the way loads are computed using Euler-Bernoulli beam theory are presented. Then, the modelling of the turbine using the Actuator Line and Actuator Disk methods is introduced. Three different methods are developed using the Actuator Disk approach, a direct integration method and two mode superposition methods. Eventually, the whole convergence process used to compute the solution is explained thoroughly.

Chapter 3 The results obtained from the different developed methods are gathered in this chapter and compared to a reference simulation to assess their accuracy. The numerical setup used for the different simulations is developed and the methods are evaluated through their the tip deflection over time, loads and fatigue analysis and the power and thrust histories. Finally, the computational cost of the different methods is analyzed in detail.

Chapter 4 In this chapter, we draw the general conclusion and bring some perspective aspects that could be developed to go further in advancing the flexible AD method.

Chapter 2

Methodology

2.1 Flow solver

We consider turbulent flows solved through Large Eddy Simulation (LES), employing an in-house developed fourth-order finite differences code known as BigFlow. The code operates within a cartesian coordinate system and solves the incompressible Navier-Stokes equations, supplemented by a subgrid scale (sgs) model, as described in Eqs. (2.1) and (2.2).

$$\frac{\partial u_i}{\partial x_i} = 0 \tag{2.1}$$

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} + \frac{\partial \tau_{ij}^{sgs}}{\partial x_j} + \frac{f_i}{\rho} \tag{2.2}$$

where $i, j = 1, 2, 3$ corresponding to the three directions in space, u_i is the velocity associated with direction i, t is time, p is the pressure, τ_{ij} is the viscous stress tensor, τ_{ij}^{sgs} is the viscous stress tensor associated with sgs effects, f_i is the volumetric force, and ρ is the fluid density.

Considering a Newtonian fluid, the viscous stress tensor can be written as $\tau_{ij} = 2\nu S_{ij}$ with the strain-rate tensor defined as:

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \tag{2.3}$$

The subgrid scales can be resolved with a classical Smagorinsky model [23] or a Regularized Variational Multiscale (RVM) [5, 2]. It has been shown that the interaction between the RVM model and wall models affects the results negatively [3]. Therefore, if a wall needs to be modelled, the Smagorinsky model will be used, otherwise we use RVM.
The simulations in this study have been performed in turbulent flows that does not involve an atmospheric boundary layer, therefore no wall modelling is required and the RVM model is employed.

The RVM model separates the large scales from the LES-resolved small scales to specifically calculate the strain-rate tensor of the latter. The simulations performed are resolved using a uniform grid, the sgs viscous stress tensor thus reads:

$$\tau_{ij}^{sgs} = 2C_s\Delta x^2(2S_{mn}^s S_{mn}^s)^{0.5} S_{ij}^s \tag{2.4}$$

where S_{ij}^s is the strain-rate tensor of the small scales, Δx is the uniform grid spacing and C_s is the Smagorinsky coefficient.

The version of the flow solver that will be used has been implemented by Duponcheel [3]. The spatial discretization of the equations is done with the fourth-order finite differences scheme developed by Vailyev [28] and implemented to conserve the discrete kinetic energy up the same order.
The Navier-Stokes equations are then solved temporally using a fractional-step method proposed by Lee *et al.* [11] projecting the velocity solution onto a solenoidal field, i.e. divergence-free. To do so, a Poisson equation is solved using a multigrid solver giving the pressure increment corresponding to the correct projection. See Duponcheel [3] and George [6] for further information regarding the process of multigrid solving.

2.2 Generation of the inflow

The inflow entering the domain contains turbulent fluctuations, which are generated in this work using Mann algorithm. It is a method used to generate synthetic turbulence in CFD simulations that was introduced by Mann [14] in 1998. By employing statistical properties derived from measurements or theoretical models of atmospheric turbulence, the Mann algorithm generates turbulent velocity fields that simulate realistic turbulent flows.

The method consists in creating a "box" situated just before the domain of simulation and inside of which is imposed a turbulent field over a mean inflow velocity. Periodic boundary conditions are applied to ensure that the output observed at the end of the box is periodic with a period corresponding to its entire length.

The box used in this study has the following characteristics:

- size: 48Dx6Dx6D (D = 126 m, the diameter of the turbine rotor) ;
- mean wind speed: 9 m/s ;
- turbulence intensity (TI): 6 % ;
- uniform grid of resolution (in number of points): 768x96x96 (16 points per diameter).

The data given as AxBxC correspond to the directions streamwise (A), vertical (B) and spanwise (C). Typically expressed as a percentage, turbulence intensity is a measure that indicates the relative strength of fluctuations or disturbances within a flow compared to the average velocity and is defined as:

$$TI = \frac{\sqrt{\overline{u'^2}}}{\overline{U}} \tag{2.5}$$

where $\overline{u'^2}$ is the average of the square of the velocity fluctuations, and $\overline{U}$ denotes the mean velocity.

2.3 Wind turbine modelling

In order to model the interaction between the wind turbine rotor and its surrounding flow field, a common approach consists of using actuator models. These models are used to simulate the presence of a physical rotor in a flow and account for its effects through equivalent body forces added to the Navier-Stokes equations. These models use control points placed on the rotor, and at which the forces applied by the flow to the blade are computed using blade element theory, which is presented in the next section. The forces are then included in the flow solver as a reaction force applied by the blade to the wind.
In this study, the two most common approaches are considered, the Actuator Line (AL) method and the Actuator Disk (AD) method, each having their own distribution of control points on the rotor.

2.3.1 Blade element theory

First, we need to understand how the forces are computed from the local flow velocity using the blade element theory [15]. This method breaks down a blade into smaller radial segments of length Δr, allowing the application of two-dimensional aerodynamics within these segments. Those are assessed individually to compute the aerodynamic forces acting on each one of them by accounting for local airflow velocity, blade's rotational speed, effective angle of attack, airfoil geometry and corresponding lift and drag coefficients.

The effective angle of attack is derived by subtracting the twist and pitch angles at the airfoil position, β and γ respectively, from the flow angle ϕ which is induced by the downwash angle and wake dynamics:

$$\alpha = \phi - (\beta + \gamma) \tag{2.6}$$

The flow angle ϕ is determined from the local axial and tangential velocities of the flow, u_n and

u_θ, measured at the center of the radial element and is computed as:

$$\phi = atan\left(\frac{u_n}{\Omega r - u_\theta}\right) \tag{2.7}$$

In order to derive the lift and drag aerodynamic forces acting on a blade element of length Δr, the theory employs experimental lift and drag coefficients, C_l and C_d, specific to the airfoil, and corresponding to the effective angle of attack α. The coefficients are then multiplied by the dynamic pressure $0.5\rho V_{rel}^2$ and the local chord length c:

$$L = 0.5\rho V_{rel}^2 \; c\Delta r \; C_l \tag{2.8}$$

$$D = 0.5\rho V_{rel}^2 \; c\Delta r \; C_d \tag{2.9}$$

where V_{rel} is the relative flow velocity:

$$V_{rel}^2 = u_n^2 + (u_\theta - \Omega r)^2 \tag{2.10}$$

with Ω the rotation speed of the rotor in rad/s.

The lift and drag forces are expressed in the frame aligned with the relative velocity and are transferred into the rotor frame as normal and tangential components, F_n and F_θ using the flow angle ϕ:

$$\begin{aligned} F_n &= L \; cos(\phi) + D \; sin(\phi) \\ F_\theta &= L \; sin(\phi) - D \; cos(\phi) \end{aligned} \tag{2.11}$$

2.3.2 Actuator Line

In the AL method, presented by Sørensen and Chen [24], the blades are represented as rotating lines with distributed control points along their span, where the forces are computed, as shown in Figure 2.1. It provides a detailed representation of the blades and their aerodynamic effects on the flow, including on the resulting wake as it considers the unsteady wake interactions between the blades. The AL therefore allows the concurrent computations of the structure and the flow dynamics, passing on information between both in real time, enabling true fluid-structure interaction. This allows for an accurate representation of the wake dynamics and its influence on the aerodynamic performance.
The main drawbacks of this method is that it requires a rather fine mesh in order to obtain correct results, and it constrains the time step as the blades cannot travel more than one mesh cell over a time step. The AL approach is thus more appropriate for the simulation of one or a few wind

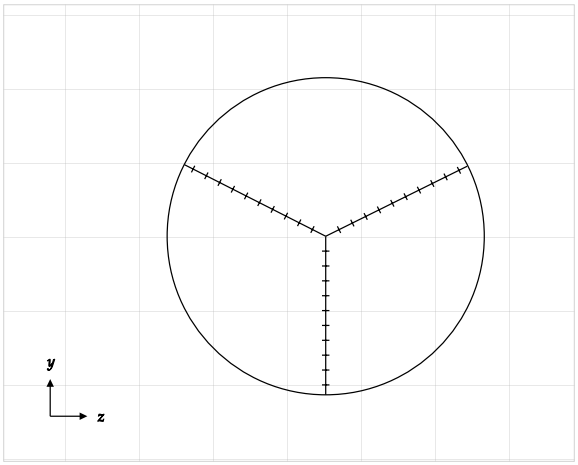


Figure 2.1 *– Actuator Line configuration for control points distribution. The blades are divided radially by the control points (small black lines).*

turbines but far less for wind farm simulations.

2.3.3 Actuator Disk

In the AD approach, the blades are represented as equivalent forces evenly distributed on a disk corresponding to the surface swept by the rotor. This means that the effect of the N_b blades is averaged over the disk surface through polar elements discretizing the disk. It is thus decomposed into radial and azimuthal elements, each bearing a control point at their center as shown in Figure 2.2.

Similarly to the AL, the equivalent forces are computed at every control point using the local velocities of the flow. The rotor mesh being polar, the velocities need to be interpolated from the cartesian LES mesh. This is being carried out with a third-order $M'4$ interpolation scheme [17]. The forces are then derived using the blade element theory, but as the effect of the aerodynamic forces of the N_b blades must be felt in each small element composing the disk, we need to add a weight to the equivalent forces.

Considering the polar elements situated at a certain radius r, they must all account for the effect of the N_b blades. So, for each azimuthal element, F_n and F_θ are multiplied by the number of blades N_b and weighted by the ratio between the element surface $\Delta S(r)$ and the surface of the total annular section of width Δr:

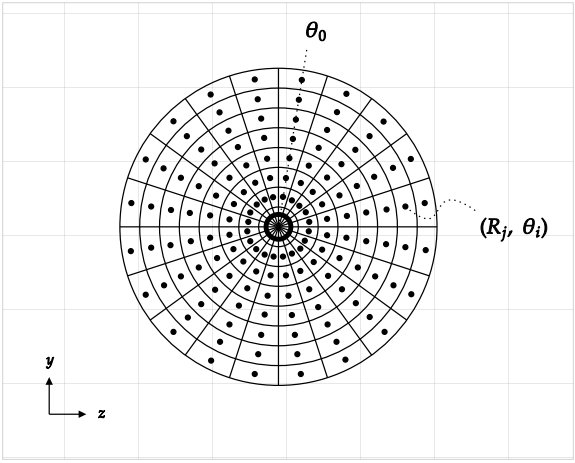


Figure 2.2 *– Actuator Disk configuration for control points distribution. The disk is divided azimuthally and radially, creating polar elements with control points (black dots) in their center denoted* (R_j, θ_i)*.*

$$F_{n,AD} = F_n \; N_b \frac{\Delta S}{2\pi r \Delta r} \tag{2.12}$$

$$F_{\theta,AD} = F_\theta \; N_b \frac{\Delta S}{2\pi r \Delta r} \tag{2.13}$$

However, due to its simplifications and limitations, the AD method sacrifices detailed blade representations and complex aerodynamic interactions for computational efficiency. It simplifies the aerodynamic effects of the blades by applying averaged forces across the disk, enabling faster simulations but at the cost of detailed accuracy in representing the turbine's behaviour at the blade level.

2.4 Structural solver

In the previous section, the forces are computed only from the flow velocity and does not take the blade deformation into account. However, the coupling between the actuator model and the structural solver is crucial to account for the interaction between structure and flow as both are contingent on one another and influence the forces computation. This coupling can be achieved in real time or through post-processing.

In the AL method, the coupling happens in real time during the simulation. This means that

the actuator model and the structural solver communicate and exchange information continuously, integrating the effects of blade deformation into the aerodynamic calculations to get the most realistic response of the blade in the flow.
On the other hand, in the AD method, the coupling occurs after the simulation is complete. The flow velocity obtained on the disk assuming a rigid turbine, i.e. without blade deformation, is then used to model the interaction between the blades and the flow in post-processing.

In both cases, a Euler-Bernoulli classical beam theory (EBBT) integrated into finite element analysis is used.
In the case of the AL method, the structure is solved in parallel to the flow computations, both coupled in real time and working in a staggered scheme [27] briefly explained. The flow is solved at time t and the resulting aerodynamic loads are used in the structural solver as input for the dynamic analysis. The updated displacement and velocities computed with the EBBT at time $t + \frac{dt}{2}$ are then sent to the AL model to compute the equivalent body forces to be used in the flow solver.
The AD method does not include aeroelastic considerations in the computations of the equivalent body forces in the flow solving and considers the rotor as rigid, i.e. undergoing no deformation. Theses forces are computed using the blade element theory only and the aeroelastic effects are included afterwards as explained in the next section.

2.4.1 Coupling with structural solver

In this section, the coupling between the AD model and the EBBT structural solver is presented.

First, in order to apply EBBT, the blade is represented as a beam using a finite element model. The beam is thus divided into n_e small elements, each of size Δr, delimited by $n_e + 1$ equidistant nodes going from the root of the blade to its tip. At each node, all six degrees of freedom are evaluated, i.e. three translational and three rotational.

In the disk environment, the beam can only take positions where control points are defined. Therefore, considering a disk discretization with n_R the number of radial elements and n_θ the number of azimuthal elements, there are n_θ positions around the disk where the beam is determined by an azimuthal value θ_i and the n_R radial corresponding elements. Referring to Figure 2.2, the beam can take n_θ positions. Each one corresponds to a θ_i such that the beam occupies the control points (R_j, θ_i) with $j = 1, .., n_R$.

The purpose of the structural solver is to compute the updated states (position, velocity and acceleration) of every node composing the beam, knowing their previous states, and the loads applied at the nodes positions. At each node, the states are evaluated in all six degrees of freedom, i.e. three translational and three rotational. In the calculations, five state variables are used: the displacement and the rotation in the flapwise direction, x_1^{struct} and θ_1^{struct}, and in the edgewise

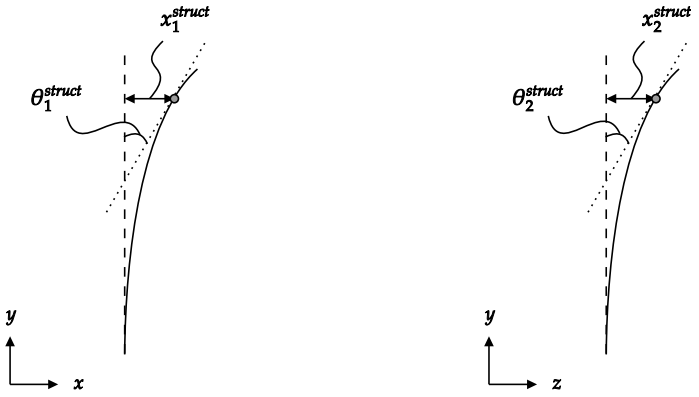


Figure 2.3 *– Definition of beam state variables for a particular node (small grey disk). The undeformed, rigid blade is represented in dashed lines, and the position of the deformed blade is drawn with solid lines. Flapwise deformation is depicted on the left, and edgewise deformation on the right.*

direction, x_2^{struct} and θ_2^{struct}, which are defined in Figure 2.3, and lastly the torsion ϕ^{struct}.

Those five state variables are then used for the computation of the loads and updated after the application of EBBT.
The loads are computed using the same method as in the blade element theory but adding to the equations the contributions of the structural deformations. We therefore need the normal and tangential flow velocities at the positions of the beam nodes, u_n and u_θ respectively. Those are retrieved from the simulation results at the corresponding control points of the Actuator Disk. For these velocities to be correctly interpreted by the beam, the number of nodes on the beam is chosen to be the same as the number of radial control points constituting the AD, i.e. $n_e + 1 = n_R$. In this manner, the nodes of the beam coincide with the control points of the AD.

Referring to the blade element theory described in Section 2.3.1, Eqs. (2.7) and (2.10) become:

$$\phi = atan\left(\frac{u_n^*}{\Omega r - u_\theta^*}\right) \tag{2.14}$$

and

$$V_{rel}^2 = (u_n^*)^2 + (u_\theta^* - \Omega r)^2 \tag{2.15}$$

where

$$\begin{aligned} u_n^* &= u_n \cdot cos(\theta_1^{struct}) - u_1^{struct} \\ u_\theta^* &= u_\theta \cdot cos(\theta_2^{struct}) - u_2^{struct} \end{aligned} \tag{2.16}$$

with u_1^{struct} and u_2^{struct} the flapwise and edgewise structural velocities respectively.

The torsion ϕ^{struct} also has influence on the angle of attack seen by the airfoil such that:

$$\alpha = \phi - (\beta + \gamma) - \phi^{struct} \tag{2.17}$$

Following the procedure, Eqs. (2.11) give the aerodynamic loads. In addition to those, the centripetal force and the force of gravity are also accounted for.
The structural solver then uses the loads as the external force term $F(t)$ in the dynamic equation of motion (Eq. (2.18)). Using the current states of the structure and the known structural properties of the blade, i.e. mass, damping and stiffness matrices, this equation is solved using an implicit Newmark integration scheme with finite elements methods, giving the updated states of the structure at each node.

$$M\ \ddot{x}(t) + C\ \dot{x}(t) + K\ x(t) = F(t) \tag{2.18}$$

where M, C, K are the mass, damping and stiffness matrices respectively, and x is the state variable, with $\dot{x}$ and $\ddot{x}$ its temporal derivatives.

2.5 Transient analysis

The transient analysis consists in studying the behaviour of a system over time when it is subjected to time-varying or conditions. To understand how a system behaves dynamically, we can solve differential equations that represent the system's dynamics, i.e. the equation of motion (Eq. (2.18)). This equation represents a system of linear differential equations of second order and will be solved using two different methods in this study, the direct integration method and the mode superposition one.

2.5.1 Direct integration method

In direct integration methods, no modification is applied to Eq. (2.18) before its numerical integration.

Unlike the AL for which the control points move circularly to represent the real temporal position of the blades, the AD control points are fixed and spread everywhere on the rotor area. Therefore, there is no way of knowing the actual position of the blades during the simulation, but instead the effect of the blades is averaged over the whole disk area.
In order to simulate blades rotating and to accurately capture their dynamic behaviour in the flow, we must therefore take into account the information stored in the whole disk. This is true for each time of the simulation, thus for every nt, the wind velocities known at all the control points, i.e. at every (n_R, n_θ) positions, must be considered even to simulate a single blade evolving in this velocity field. As we consider the situation at a certain fixed point in time, the wind velocities are constant for a given control point, the velocity field is thus said to be "frozen". It is defined at a time t by

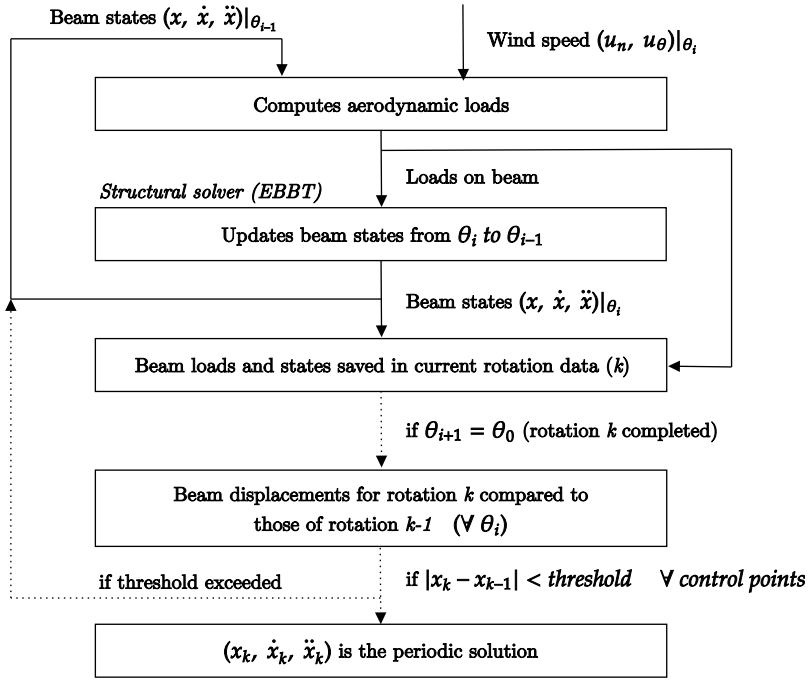


Figure 2.4 *– Process used for finding the periodic solution for blade position in a frozen velocity field. Focusing on a single azimuthal step between θ_{i-1} and θ_i on the upper half, and more generally between two entire rotations $k-1$ and k on the lower half.*

the set of wind velocities $(u_n, u_\theta)|_t$ evaluated at every control point, giving a matrix of size (n_R, n_θ). It therefore exists nt different matrices corresponding to one frozen velocity field of size (n_R, n_θ).

Frozen velocity field

For each frozen velocity field, we simulate the rotation of a single flexible blade represented by a beam and solved dynamically with EBBT presented above. The beam will thus occupy each azimuthal element one after the other and the states are updated at each step. The purpose is to find the periodic solution for the position, i.e. deformation, of the beam when evolving in a constant velocity field. In other words, we want the position of the beam $x(\theta)$ for all θ, such that $x(\theta_i) = x(\theta_i + 2\pi)$. The process undergone to obtain this solution is explained below and represented in Figure 2.4.

The rotor is divided azimuthally by n_θ elements, splitting the disk into $\frac{2\pi}{n_\theta}$ small angles. In order to comply with the physical rotational speed of the blade, the time step used for time integration dt_{EBBT} is taken as the time needed to travel one angle element, thus depending on the rotation

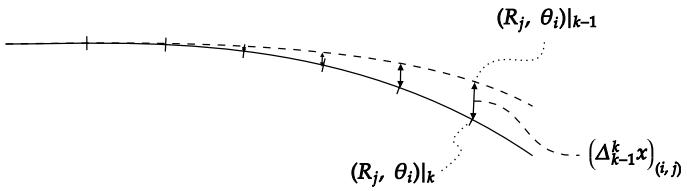


Figure 2.5 *– Comparison of blade deformation at azimuthal position θ_i for two consecutive rotations $k-1$ and k.*

speed of the rotor ω: $dt_{EBBT} = \frac{2\pi}{n_\theta}\frac{1}{\omega}$ [s].

During the first time step, the beam starts undeformed at θ_0 and the structure is solved at θ_1 using the wind velocities at this azimuthal position with EBBT. The updated states of the beam nodes are known and will be used for the next application of EBBT at position θ_2.
The process repeats itself for every azimuthal position until the blade comes back to its initial position at θ_0. After a complete rotation, the deformation of the beam, i.e. the position of its nodes, is known for every azimuthal element of the disk. The method used to determine if the solution converges is by comparing, after each rotation and at every azimuthal element, the deformed shape of the beam of the last rotation with its deformed shape during the second-to-last rotation. The beam thus needs to make at least two rotations in the frozen velocity field to be able to compare two solutions to one another. If two successive rotations give blade positions close enough to one another at every azimuthal position, this means the observed deformation of the blade does not evolve anymore and the solution is periodic. If another rotation is performed, the same solution, i.e. blade position, will be found. The last solution is thus the converged periodic solution and is the one used in further calculations.

When comparing two blade deformations (from two different rotations) at a certain azimuthal position, the difference between the positions of the beam nodes is assessed both in the flapwise direction and in the edgewise direction. This means that for every node (R_j, θ_i), their positions at rotation k will be compared to those of rotation $k-1$, giving a set of (n_R, n_θ) values $\Delta^k_{k-1}x|_{(i,j)}$ representing the difference in length between two successive rotations. The solution is taken as converged if all of these differences do not exceed a certain threshold. The thresholds are chosen as one percent of the maximum deflection of the blade in each direction. The maximum deflection flapwise is taken as ten percent of the blade length, while we take one percent of the blade length in the edgewise direction.

Solution interpolation

The whole process described in the previous section is repeated for every time of the simulation, i.e. for nt steps. However the beam only starts from its undeformed position when the first frozen velocity field is considered, i.e. at time $t = 0$. When the beam starts rotating in the frozen velocity field at a time $t = i\ dt$, $i > 0$, it uses, as input in the first application of EBBT, the last deformed shape of the beam computed from the previous converged solution, at $t = (i-1)dt$.

Therefore, for every time t, we have a set of (n_R, n_θ) solutions in terms of beam states, corresponding to the periodic solutions found in the different frozen velocity fields. These solutions are used to model the real motion of blade in the flow. To do so, only one blade is considered and starts at θ_0 at $t = 0$.
The first solution is thus the one taken at angle θ_0 from the periodic solution found at t_0. During one time step dt, the blade travels an angle $\omega\ dt$ and the set of solutions used is the one found for $t = dt$. In this set, we only have solutions at angles multiple of $\frac{2\pi}{n_\theta}$ as this is the positions of the control points where the solutions have been computed. In the real flow, the blade sits at an angle $\omega\ dt$ which can be located between control points. The right solution is thus interpolated to match the correct angle of the blade.
This process is repeated for the nt time steps such that the real position of the blade is $(\omega\ i\ dt)$ and is interpolated from the solutions found previously for the frozen velocity field of time $t = i\ dt$. After interpolating the solutions for the nt time steps, the movement of a single deformed blade is known throughout the entire simulation, representing its dynamic behaviour.

2.5.2 Mode superposition method

Direct integration methods can lead to substantial computational costs when the system matrices are of high order or if a lot of time steps must be solved. Therefore, another method described in [21] is assessed in this section, with the purpose of reducing the analytical computations and making the process faster.

Due to the way Newmark integration scheme works, which is an implicit method, computational costs are directly proportional to the order and the bandwidth of the stiffness matrix and to the number of time steps. Consequently, implicit direct integration proves effective for shorter duration of response. For extended time step requirements, transforming the equilibrium equations (2.19) into a less resource-intensive form might be beneficial.

$$M\ddot{x}(t) + C\dot{x}(t) + Kx(t) = F(t) \tag{2.19}$$

The equilibrium equations will thus be solved using the modal analysis technique with the

following transformation:

$$x(t) = \Phi\ u(t) \tag{2.20}$$

where $x(t)$ is the vector of order n of the nodal displacements, Φ is a $n \times n$ matrix and $u(t)$ is a vector of order n containing the generalized displacements.
Combining Eq. (2.19) and (2.20), we get the new form of the equilibrium equations:

$$\hat{M}\ddot{u}(t) + \hat{C}\dot{u}(t) + \hat{K}u(t) = \hat{F}(t) \tag{2.21}$$

The new matrices are defined as follows, with the mass, damping and stiffness matrices, $\hat{M}, \hat{C}, \hat{K}$, having a smaller bandwidth than the original matrices:

$$\hat{M} = \Phi^T M \Phi\ ;\ \hat{C} = \Phi^T C \Phi\ ;\ \hat{K} = \Phi^T K \Phi\ ;\ \hat{F} = \Phi^T F. \tag{2.22}$$

In order to ensure the latter statement, the transformation matrix Φ must be selected accordingly. An effective choice is by using the displacement solutions of the free-vibration equilibrium equations with damping neglected, defined by Eq. (2.23). Assuming the form of the solution as $x = \Phi sin(\omega t)$, this leads to the generalized eigenvalue problem (2.24).

$$M\ddot{x} + Kx = 0 \tag{2.23}$$

$$(K - \lambda M)\Phi = 0 \tag{2.24}$$

Assuming the dimension of the dynamic matrix is n, this eigenvalue problems yields n scalar values λ, the eigenvalues, and n corresponding vectors of order n, the eigenvectors Φ. The eigenvalues are related to the natural frequencies of the system such that $\lambda = \omega^2$.

The transformation matrix Φ used in Eq. (2.20) is therefore the matrix of the eigenvectors from problem (2.24). We now have a new equilibrium system whose solution is expressed as generalized modal displacements $u(t)$, and which can be solved the same way as for the direct integration method but with less costly computations. The nodal displacements $x(t)$, which physically represent the deformation of the blade, can be computed using Eq. (2.20). The actual gain will be discussed in Section 3.3.1.

The new system (Eq. (2.21)) is thus solved using the eigenvectors corresponding to the natural frequencies of the system. If matrix Φ includes all of the eigenvectors, the effect of every dynamic mode will be accounted for and the dimension of the system will be the same as for the direct integration method.
However, only certain frequencies λ_i, and hence certain vibration modes Φ_i, significantly influence

the dynamic behaviour of a system, and by considering the most important modes only, the system's behaviour can still be accurately represented.

Reduction of modes number

Through modal analysis, it is discovered that only a few modes contribute significantly to the system's response. These dominant modes often correspond to the lowest frequencies and contain the majority of the system's vibrational energy. Therefore, they have the most substantial effect on the system's overall dynamic response.
By focusing on these dominant modes, it is possible to create a reduced-order model of the system that captures its essential dynamic behaviour. The dimension of the system indeed reduces to the number of modes considered in the analysis.

In this case, the three degrees of freedom of the blade having the strongest influence on its dynamic behaviour are chosen, and for each degree of freedom, its first mode is taken as a dominant mode. We thus have three dominant modes to represent the whole system.
The transformation matrix now contains only three eigenvectors, $\Phi = (\Phi_1, \Phi_2, \Phi_3)$ and because of relation (2.22), matrices $\hat{M}$, $\hat{C}$, $\hat{K}$ are reduced to size 3×3. The system (2.21) is now of dimension three and therefore much faster to solve. Computational gains will be discussed in Section 3.3.1.

In order to choose the three modes having the most influence, we need to analyze which degrees of freedom have the greater impact.
The influence of flapwise and edgewise structural velocities, u_1^{struct} and u_2^{struct}, can be observed in Eqs. (2.14) and (2.15) which are crucial in the computations of the loads applied to the blade which in turn dictate its dynamic response in the structural solver. These two degrees of freedom should therefore be accurately represented. The angles in the flapwise and edgewise directions, θ_1^{struct} and θ_2^{struct}, also appear in those equations but their influence is less marked. Moreover, because of the geometry of the blade, namely its twist, the four degrees of freedom mentioned, i.e. flapwise and edgewise displacements and angles, are coupled. Therefore, a variation in one of them will have an impact on the other three. This can be observed through modal analysis and allow those degrees of freedom to still be represented even without their principal modes considered in the calculations.
However, when the modes associated with the torsion, referred to as ϕ^{struct} in the previous equations, are studied, it is observed that no coupling with other degrees of freedom exist. Since the torsion plays an important role in the computation of loads, as it helps decrease the angle of attack (Eq. (2.17)), we must consider it and choose its first mode as one of the dominant ones.

These assumptions and choices are then verified through modal analysis. We choose to compare the first ten modes corresponding to the ten lowest natural frequencies of the blade and weight them in terms of their influence on the exact solution. Their individual effects are represented in

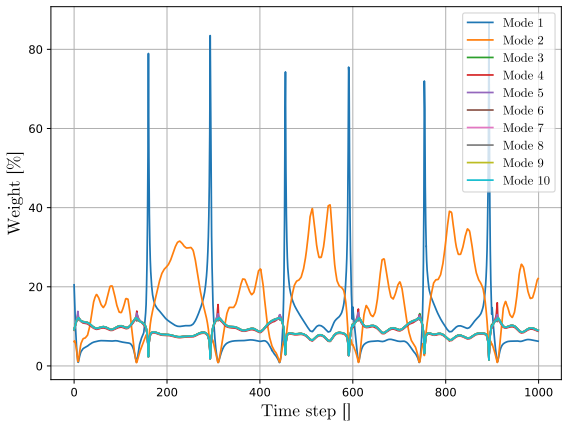


Figure 2.6 – *Comparison of the first ten modes of a blade for a resolution using the mode superposition method with one mode at a time. The solutions found are weighted with respect to the reference solution obtained using the direct integration method.*

Figure 2.6. The figure represents the weights in percentages of the first ten modes over 1000 time steps by comparing the exact solution of a particular simulation with the solutions obtained using only one mode at a time. This means that, for example, if a mode reaches 20 percent on the y-axis, it accounts for 20 percent of the exact solution among the ten first modes only, i.e. the sum of the percentages of these ten modes will be 100 percent. In the figure, mode 1 corresponds to x_1^{struct}, mode 2 to x_2^{struct}, and mode 6 to ϕ^{struct}. The result for only one simulation is displayed here but similar results are observed for the other simulations.

While excluding less influential modes simplifies analysis, the representation of the system's dynamic behaviour remains accurate due to the strong influence of the dominant modes. These modes effectively summarize the system's behaviour without the need to consider every mode.

Chapter 3

Results

3.1 Numerical setup

The wind turbine used in this study is the NREL-5MW [10], which is a reference wind turbine for offshore system development. It has a rotor diameter $D = 126\ m$, hence a rotor radius $R_{tip} = 63\ m$, with blades of length $L = 61.5\ m$. The simulations are performed in a domain of size 8D in the streamwise direction and 4D in the transverse directions (8D x 4D x 4D), with the center of the wind turbine located at position 3D x 2D x 2D.

The mesh is a uniform cartesian grid (no refinement near the rotor) defined by a certain number of grid points per rotor diameter D. The inlet flow is turbulent, and periodic boundary conditions are imposed on the vertical and lateral directions. In this study, three grid resolutions are considered, 16 and 32 points per rotor diameter for the Actuator Disk and 64 points per diameter for the Actuator Line. For each resolution, three inflow velocities U_{inf} are used, 5 m/s, 9 m/s and 14 m/s. The disk discretization parameters are given for each resolution in Table 3.2.

The reference simulation chosen is the AL of resolution 64 points per D and will be used to verify the solutions obtained for the different resolutions of the AD. This reference case resolution can accurately capture the near wake and resolve the interactions of tip vortices with the larger-scale eddies of the ambient flow [9].

3.1.1 Evaluated configurations

The flow conditions are taken to represent three typical rotor behaviours [16]: when the rotor starts ($U_{\text{inf}} = 5\ m/s$), during the optimum operating condition, i.e. with maximized power, ($U_{\text{inf}} = 9\ m/s$), and during the rated operation when the rotor speed is kept constant to avoid excessive loading on the generator ($U_{\text{inf}} = 14\ m/s$). These different inflow velocities correspond to certain

operating parameters defined in the data of the wind turbine and displayed in Table 3.1. Each upstream velocity U_{inf} is associated with a certain rotor speed Ω expressed in rotations per minute, a tip pitch angle γ in degrees, and a tip-speed ratio λ (or TSR) defined as $\Omega R_{tip}/U_{\text{inf}}$. The disk controller is deactivated, therefore the rotor speed and the blade pitch angle stay constant throughout the entire simulation.

U_{inf} [m/s]	Ω [rpm]	γ [°]	λ [ ]	Δt [s]	nt
5	7.51	0	9.90	0.02665	15130
9	10.30	0	7.54	0.01943	11529
14	12.10	8.67	5.70	0.01653	8712

Table 3.1 *– Operating parameters.*

3.1.2 Grid spacing

The size of the macro domain grid is determined by the simulation resolution as it imposes a certain number of grid points per diameter. The diameter is 126 m, therefore dividing it by the number of points given by the resolution choice gives the the uniform grid spacing Δx. The values for each resolution is shown in Table 3.2.
To ensure stability, it is also advised to have the length of the blade elements smaller than the mesh spacing [9]. This condition is verified if the number of control points dividing the blade is chosen to match the flow resolution, i.e. for a resolution N_{reso} pt/D, the blade of length $L = 61.5$ m is split by $\frac{N_{reso}}{2}$ elements.

Resolution	Δx [m]	n_θ	n_R
16 pt/D	7.88	62	8
32 pt/D	3.94	112	16
64 pt/D	1.97	3*	32

Table 3.2 *– Parameters for different resolutions. Δx is the grid spacing, n_θ is the azimuthal discretization of the disk, and n_R is its radial discretization. *The resolution 64 pt/D is only used with the AL which always divides the disk azimuthally in three elements, regardless of the resolution.*

In terms of computational cost, the coarser the grid the cheapest the computations. As the grid is three dimensional and uniform in all directions, the computational cost varies as a power of three in terms of resolution variation. This means that for a resolution 64 pt/D, 64 times more computations will be required compared to a simulation using 16 pt/D as $(\frac{64}{16})^3 = 64$.

3.1.3 Time step

The AL imposes certain numerical constraints to ensure the stability of the numerical schemes. In order to keep the solving stable, the blade tip, which is the fastest part of the blade, cannot travel more than one grid cell in a time step [19]. This condition involves the tip speed (ΩR_{tip}) and the resolution speed on the grid $\frac{\Delta x}{\Delta t}$ as shown in Eq. (3.1), where Δx is the grid spacing in the plane of the rotor and Δt is the time step:

$$\frac{\Omega R_{tip} \Delta t}{\Delta x} < 1 \tag{3.1}$$

This condition actually relates the classical CFL number used in numerical simulations to ensure the stability of time integration schemes and the tip-speed ratio λ such that $\lambda\ CFL = \frac{\Omega R_{tip} \Delta t}{\Delta x}$.

As the grid spacing is known from the resolution choice, the maximum time step allowed is inferred from condition (3.1) for each studied inflow velocity. Even though the AD does not have to verify the same condition imposed to the AL concerning the tip speed, in order to minimize the number of varying parameters and for better comparison, the time steps are chosen constant for a given velocity regardless of the grid resolution or the method used (AD or AL). It is thus chosen to verify the strictest condition (3.1) on Δt, which occurs when the resolution is 64 points per D. The size of the time steps are chosen such that there are 300 time steps per rotation, which is sufficient to verify the condition, and are shown for each velocity in Table 3.1.

3.1.4 Number of time steps

The length of the simulation is chosen for each velocity such that the entire domain is covered twice and the averaged results based on the second half of the simulation are representative of the normal operation of the turbine, without accounting for the starting phase when it begins to move from rest.

The number of time steps nt is thus computed based on the size of the domain ($8D$) and the size of the time step. The time needed for an air element to travel twice the whole domain is about $\Delta t_{tot} = 2\frac{8D}{U_{\text{inf}}}\ [s]$, and since the time step is $\Delta t\ [s]$, it would require $nt = \frac{\Delta t_{tot}}{\Delta t}$ time steps. The corresponding number of time steps for each studied velocity is given in Table 3.1.

3.2 Direct integration method

In this section, the direct integration method (DIM) described in Section 2.5.1 will be used to obtain the periodic solutions of the transients for each time step. The solutions will then be interpolated to model the physical behaviour of the blade. In order to validate this method, the results are compared with the reference model chosen which is the AL resolved with a kernel topology of 64 points per diameter.

We will then compare this complete method with simplified methods that aim to be much faster but staying as precise as possible. The three different methods assessed work in the same fashion concerning the interpolation of the physical solution from the periodic solutions found from the transients. However they differ in the way they converge to those periodic solutions.

The DIM is the most costly and precise method because it solves the transient considering every control point of the Actuator Disk until the blade reaches its periodic displacement in the frozen velocity field.
On the other hand, the mode superposition method relies on the natural frequencies and mode shapes of the blade. The second assessed method considers all the modes representing the blade and should give the same results as the DIM as it accounts for the whole dynamic behaviour of the blade. The third method only accounts for three modes of the blade, namely the first mode in flapwise deflection, the first mode in edgewise deflection and the first mode in torsion.

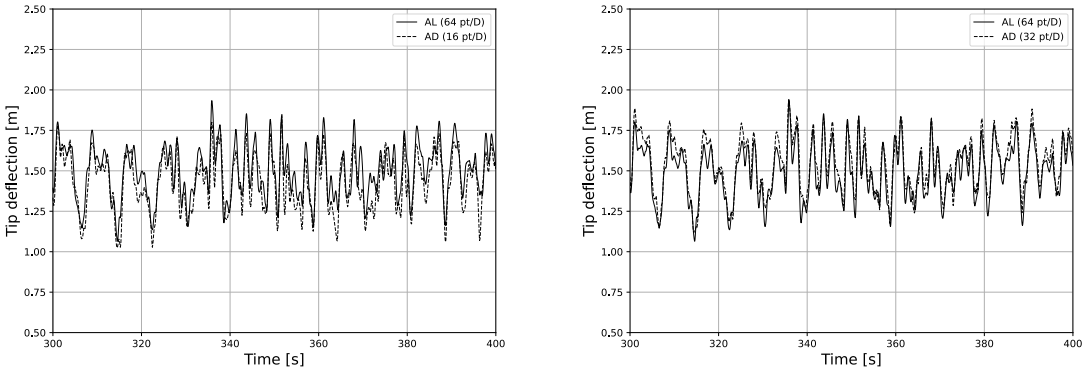
The AD and AL methods inherently differ in terms of how they model the turbine, and therefore the type of results available are different for one or the other.
The AL method outputs the results for all the blades of the wind turbine separately as it considers the N_b physical blades in the simulation through their real movement and interactions with the flow. Whereas the AD simulation outputs sets of wind velocities at every control point and for each time step which reflect the presence of the blades through an averaged effect on the whole disk surface. In this case, only one blade will be assumed to rotate in the flow, taking $1/N_b$ of the total effect, and is taken to represent the behaviour of every other blade.
In order to compare both methods, we thus select the results for one blade of the AL and put it in perspective with the results found from the single blade of the AD method.

The desired outcome is to get results from the AD methods matching as closely as possible those of the AL because both simulations have been done in the same conditions and the blade behaviour should not diverge from one method to the other.

The methods will be assessed for a single wind turbine but the validation is done in the perspective of using the methods in wind farm simulations. The validation will be assessed through the blade tip displacement over time, the time-averaged loads on the blade, the power and the thrust. Different parameters will be considered such as the influence of the kernel topology and the inflow velocity.
We will first compare the results obtained from the different methods, and then assess their computational cost and how fast is the periodic solution is found.

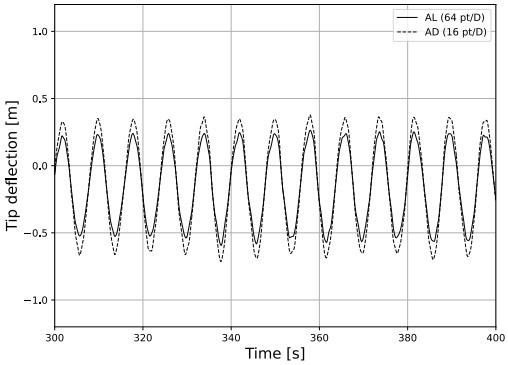
3.2.1 Blade deflection

The first parameter used to assess the accuracy of the DIM is the blade deflection. In this case, we will only compare the tip deflection between the AL and the two different kernel resolutions of the AD, namely 16 and 32 pt/D. Figures 3.1, 3.2 and 3.3 show the results for different tip-speed ratios (TSR). The flapwise deflection is measured in the direction of the flow as the distance of the tip from the rotor plane, and for the edgewise deflection, values below the sine mean correspond to a deformation in the same direction as the rotation. For the sake of clarity, all the graphs are shown for the last 100 seconds of the simulation to capture best their representative behaviour.

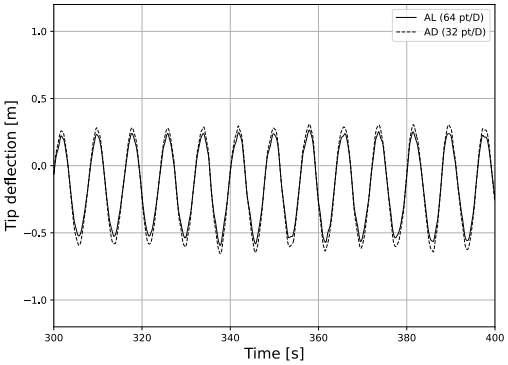


(a) *Flapwise, 16 pt/D*

(b) *Flapwise, 32 pt/D*



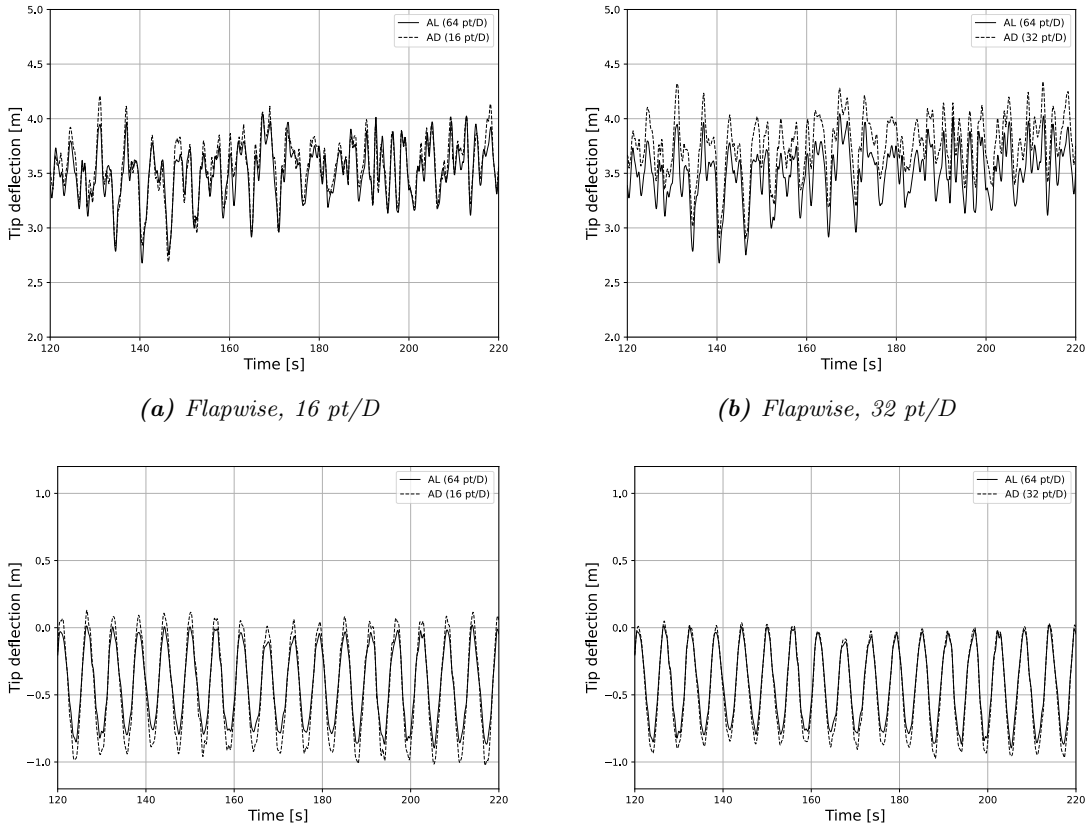
(c) *Edgewise, 16 pt/D*



(d) *Edgewise, 32 pt/D*

Figure 3.1 *– Tip displacement comparison between AL (64 pt/D) and different resolutions of AD solved with DIM for* $\lambda = 9.90$.

Overall, we observe a good agreement between all results. Some discrepancies are observed between AD and AL curves but both methods inherently differ in the way they work, and AL may



(a) *Flapwise, 16 pt/D*

(b) *Flapwise, 32 pt/D*

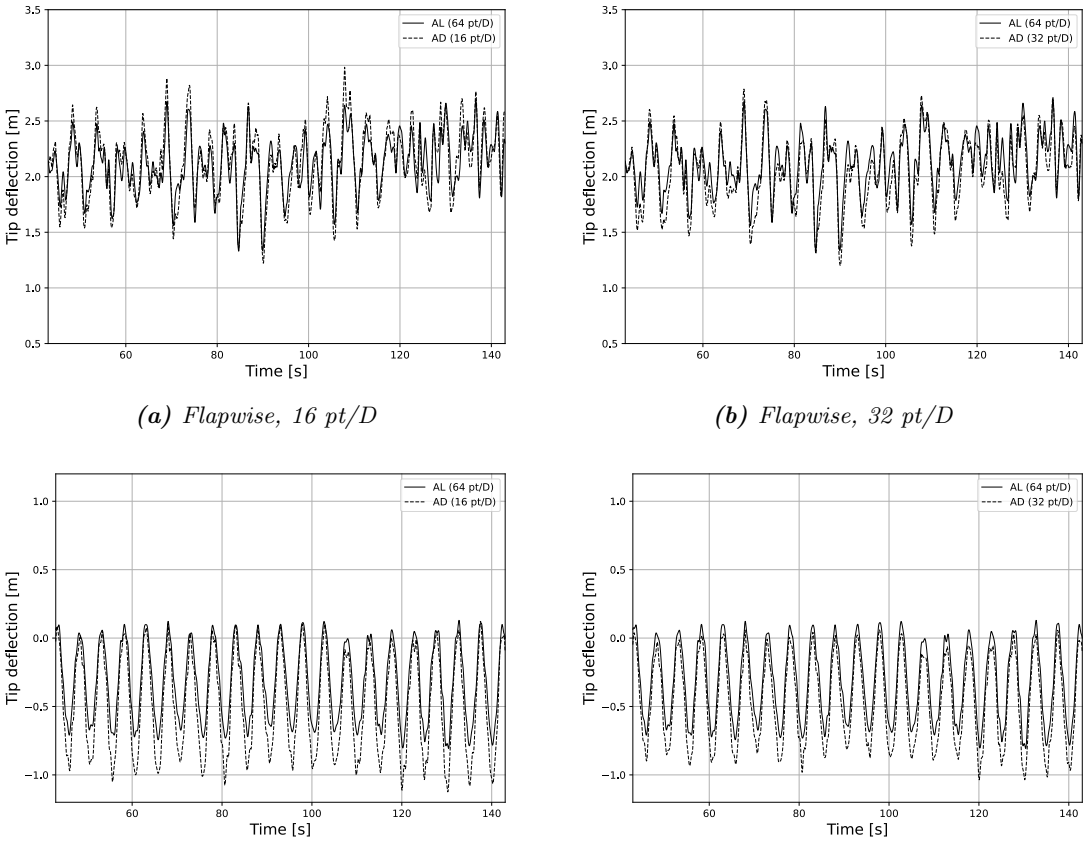
(c) *Edgewise, 16 pt/D*

(d) *Edgewise, 32 pt/D*

Figure 3.2 *– Tip displacement comparison between AL (64 pt/D) and different resolutions of AD solved with DIM for* $\lambda = 7.54$.

account for parameters AD currently does not. Besides, we must recall that all computations for the AD are done from wind velocities not accounting for blade deformation as the calculations are done in post-processing.

Regarding the flapwise deflection, the curves from 32 pt/D are very close to those of the AL for $\lambda = 9.90$ and $\lambda = 5.70$, while the 16 pt/D curves only differ slightly, mainly in amplitude, keeping a correct mean deflection. The overestimation of the tip deflection for 16 pt/D is related to the force on the tip of the blade which is too important because the tip is considered as the last node on the blade, and when increasing the resolution, this last node gets closer to the actual tip of the blade. However, for TSR = 7.54, the results obtained using a resolution 16 pt/D are paradoxically closer to what is expected than those from 32 pt/D. The shape made by the deflection over time is still correct but is shifted upwards by about 0.25 m.



(a) Flapwise, 16 pt/D

(b) Flapwise, 32 pt/D

(c) Edgewise, 16 pt/D

(d) Edgewise, 32 pt/D

Figure 3.3 *– Tip displacement comparison between AL (64 pt/D) and different resolutions of AD solved with DIM for* $\lambda = 5.70$.

This behaviour may be due to the blade pitch as for this TSR, it is set to zero which causes the flapwise forces to be more important than for $\lambda = 5.70$ which has a non-null blade pitch and for which such a behaviour is not observed.

For every case, we can notice a tendency of the edgewise tip deflection to overestimate the amplitude of the movement, but getting closer to the AL curve when the resolution increases to 32 pt/D. This can be explained by the fact that the amplitude error is due to the gravity loading affecting the blade, giving this sinusoidal shape. The period of this sine curve correspond to the period of one rotation of the blade as the oscillation of the edgewise deformation is mainly dictated by gravity, while the mean value is determined by the aerodynamic forces.
Indeed, if the blade starts from a vertical position with its tip on top, during the first half rotation,

the gravity will contribute to deform the blade negatively, while the aerodynamic forces act on the opposite direction. When the blade travels the second half rotation, the gravity and the aerodynamic forces both act on the same direction to deform the blade negatively. That is why the mean deformation is below zero and the more important the aerodynamic forces, the lower the mean. Overall, the gravity loads cancel each other, thus the amplitude of the sine wave is the same for every simulation with the same resolution.
However, the amplitude is more accurate when 32 pt/D resolution is considered because gravity load is associated to each blade element with its own density. The blade density can vary a lot along its length, especially near its root, and is averaged on each element to be applied at its center. Therefore, the more blade elements, the more accurate the average density on the elements and the physical behaviour of the blade with respect to gravity.

Finally, we can observe that when the TSR increases, so does the error on the sine mean, especially for $\lambda = 5.70$. As explained above, this is due to an overestimation of the aerodynamic forces tending to deform the blade on the opposite direction of its movement.

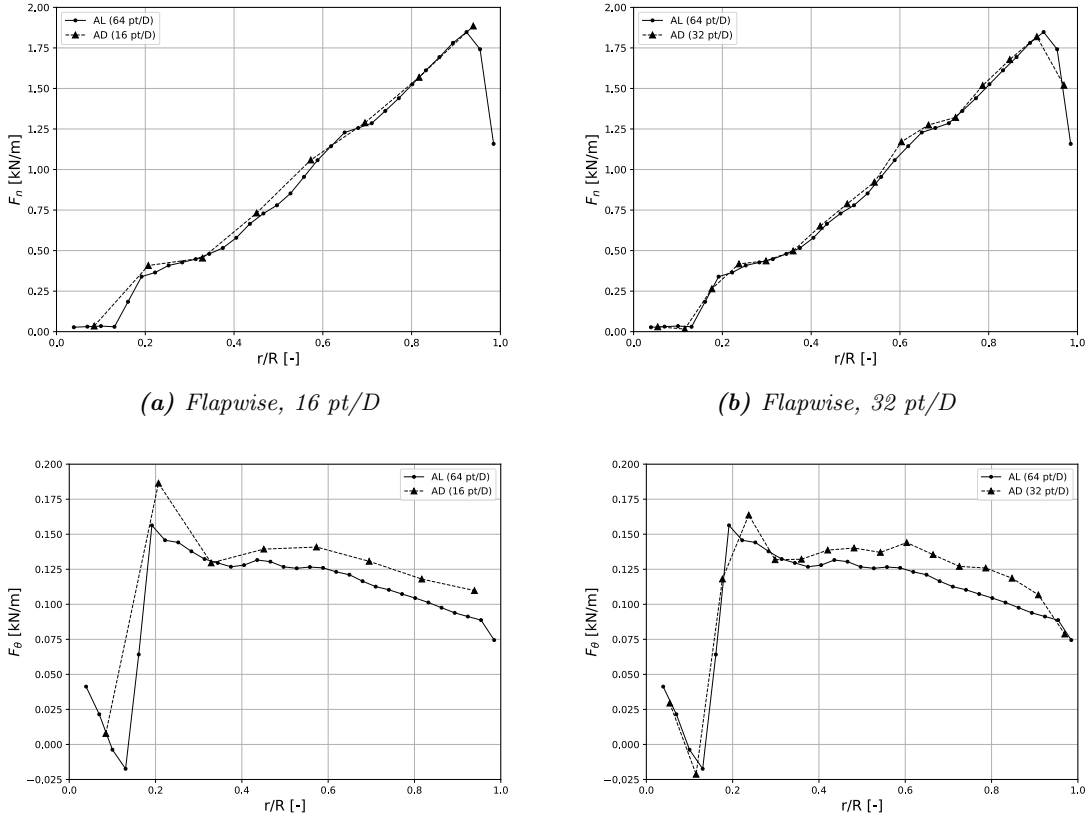
3.2.2 Loads and fatigue analysis

Another important parameter is the aerodynamic loads undergone by the blades. Figures 3.4, 3.5 and 3.6 show the flapwise and edgewise loads expressed in [kN/m] for the different resolutions and TSRs. The values are averaged on the second half of the simulation.

Regarding the flapwise results obtained with resolution 32 pt/D, they follow almost exactly those of the AL, except for $\lambda = 5.70$. The simulations using 16 pt/D still yields correct values but the discrepancy near the tip is much larger due to difference in blade radial decomposition with nodes. For the 16 pt/D simulations, the last node on the blade is situated at 57.7 m ($r/R = 0.94$) whereas for 32 pt/D, it is located at 59.6 m ($r/R = 0.97$). The 64 pt/D AL has its last node at 60.5 m ($r/R = 0.98$). This difference in tip position makes it impossible for coarser resolutions to know what happens further away from its last node, decreasing tip performance.
In the edgewise direction, the results do not comply as well, and in almost every case, the AD overestimates the force even is the shape is correct. We thus observe an offset between both curves with both resolutions performing approximately equally, except at the tip.

Root bending moment

In this section, we examine the unsteady loads exerted by the flow and leading to blade root bending moments, inducing fatigue loads through blade deformation. We consider two types of root bending moments, the aerodynamic moment and the structural moment. The former is computed from the flow perspective and determines how the wind energy is captured and converted into rotational energy by the turbine. It influences the efficiency and performance of the wind turbine by affecting



(a) *Flapwise, 16 pt/D*

(b) *Flapwise, 32 pt/D*

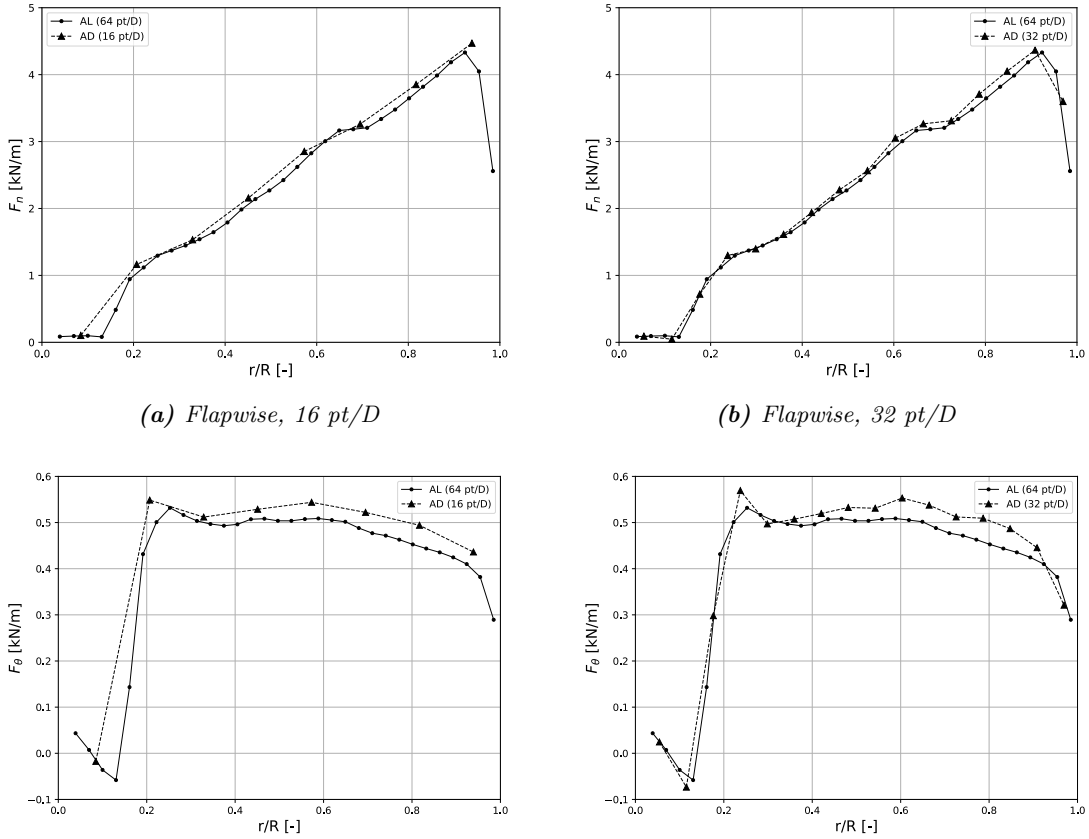
(c) *Edgewise, 16 pt/D*

(d) *Edgewise, 32 pt/D*

Figure 3.4 *– Time-averaged aerodynamic forces comparison between AL (64 pt/D) and different resolutions of AD solved with DIM for* $\lambda = 9.90$*.*

the rotational speed, torque, and power generation.
On the other hand, structural moment refers to the internal forces and moments within the blade structure resulting from external loads, such as aerodynamic forces, gravity, and centrifugal forces.

Aerodynamic root bending moment First, the aerodynamic moment refers to the moment generated by the aerodynamic forces acting on the blade as it interacts with the flow. These forces create a moment around the blade's center of rotation. This kind of moment excludes gravity loading and centripetal force and describes the aerodynamic moment in the flapwise and edgewise directions. Yielding the flapwise moment M_f, caused by the forces perpendicular to the rotor axis F_n, pushing the blade out of the rotor plane, and the edgewise moment M_e, derived from the forces acting in the rotor axis and perpendicular to the blade, F_θ. They are defined by Eqs. (3.2) and (3.3) where F_n and F_θ are the aerodynamic loads expressed in [N/m]:



(a) *Flapwise, 16 pt/D*

(b) *Flapwise, 32 pt/D*

(c) *Edgewise, 16 pt/D*

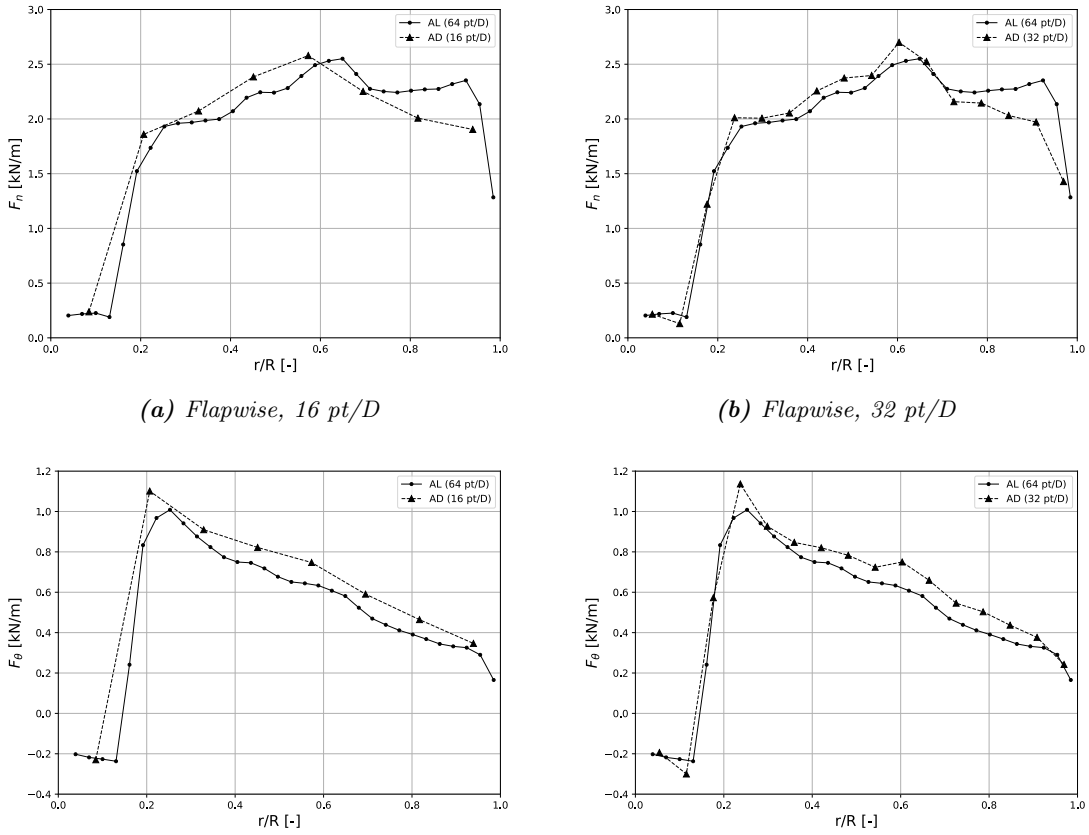
(d) *Edgewise, 32 pt/D*

Figure 3.5 – *Time-averaged aerodynamic forces comparison between AL (64 pt/D) and different resolutions of AD solved with DIM for* $\lambda = 7.54$.

$$M_f = \int_{R_{hub}}^{R_{tip}} (r - R_{hub})\ \mathbf{F_n}.\mathbf{e_n} dr \tag{3.2}$$

$$M_e = \int_{R_{hub}}^{R_{tip}} r\ \mathbf{F}_\theta.\mathbf{e}_\theta dr \tag{3.3}$$

The AL simulation outputs the quantities linked to each blade separately, it is therefore straightforward to retrieve the values for the bending moments. Whereas for the AD, we use the reconstructed trajectory and the blade-attached quantities from previous computations to be able to get the bending moments. As the AD uses a single blade as representative for the others, in order to compare both methods we retrieve the data for the AL corresponding to the blade following the same trajectory as the one in the AD method.



(a) *Flapwise, 16 pt/D*

(b) *Flapwise, 32 pt/D*

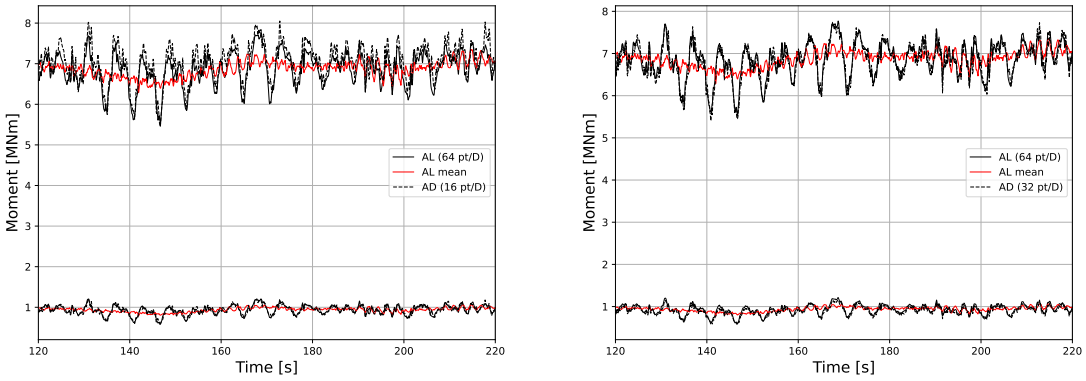
(c) *Edgewise, 16 pt/D*

(d) *Edgewise, 32 pt/D*

Figure 3.6 *– Time-averaged aerodynamic forces comparison between AL (64 pt/D) and different resolutions of AD solved with DIM for* $\lambda = 5.70$.

Figure 3.7 shows the aerodynamic moments flapwise and edgewise for the AD solved with 16 and 32 pt/D with $\lambda = 7.54$. Following results will all use this TSR as representative since similar behaviours are observed for the other TSR values and $\lambda = 7.54$ correspond to the optimum operating condition of the turbine.

Regarding the single blade behaviour, we can observe that the AD gives results very close to those of the AL, with a small offset for 16 pt/D in the flapwise direction. When using 32 pt/D, the blade-attached bending moment follows the AL curve more closely for every TSR, but as seen in Tables 3.3 and 3.4, in most cases the mean moments are predicted best for the 16 pt/D resolution. Those tables represent the values for the mean aerodynamic root bending moments. For the AL, it is computed as the real mean from the three blades (red line in Figure 3.7) and for the AD, the



(a) *16 pt/D* ***(b)*** *32 pt/D*

Figure 3.7 *– Comparison of the aerodynamic moments for $\lambda = 7.54$ between AD (dashed), a single blade of AL (solid black) and the mean of the three blades of AL (solid red). Higher values are flapwise and lower ones are edgewise.*

TSR [-]	AL	AD (16 pt/D)	AD (32 pt/D)
9.90	2.55 MW	+1.2 %	-2.0 %
7.54	6.85 MW	+2.8 %	-1.5 %
5.70	5.02 MW	-5.5 %	-8.4 %

Table 3.3 *– Mean flapwise aerodynamic root bending moment for different AD resolutions solved with DIM. Three-blade mean for AL and one-blade mean for AD.*

mean is computed from the representative blade. This difference in mean computation can explain why the results are better for 16 pt/D. To fix that, instead of using one blade as representative, we could reproduce the trajectory for the three blades of the AD and then compute the mean from these three blade-attached quantities.

Structural root bending moment Aerodynamic moments only account for flow-induced loads and exclude gravity and centrifugal effects. Those are included in the structural moments and are crucial for ensuring the structural integrity, durability, and safety of the wind turbine blades as

TSR [-]	AL	AD (16 pt/D)	AD (32 pt/D)
9.90	0.22 MW	-0.5 %	-5.3 %
7.54	0.93 MW	+0.1 %	-3.9 %
5.70	1.26 MW	-3.3 %	-7.7 %

Table 3.4 *– Mean edgewise aerodynamic root bending moment for different AD resolutions solved with DIM. Three-blade mean for AL and one-blade mean for AD.*

they determine the stress distribution, bending, and deformation of the blade materials. This is the moment related to the deformation of the blade, and hence its fatigue resistance.

The structural bending moment relates to the internal stresses and moments within the blade structure. It is the effect of the blade's own weight, rotational forces, and external loads on the blade's structural integrity. It is computed following the finite element method for beams [20] detailed in Appendix A. It relies on Euler–Bernoulli beam theory which states that the bending moment is given by the flexural rigidity EI and the second derivative of the deflection in the direction of the deformation $\frac{\partial^2 w}{\partial x^2}$, described by Eq. (3.4):

$$M_{structural} = -EI\frac{\partial^2 w}{\partial x^2} \tag{3.4}$$

Figure 3.8 show the structural root bending moments for AL and AD with $\lambda = 7.54$. As the structural bending moment is related to the blade displacement, similar behaviours are observed as in the blade deflection analysis. Namely, the better prediction of the amplitude for edgewise moment with 32 pt/D, and the overall shift to lower values for the mean edgewise moment as the TSR increases. However, the flapwise behaviour is significantly better for 32 pt/D compared to what was observed for the tip deflection. Unlike the tip displacement which varies quickly, the structural moment is computed from the node displacement near the root where the amplitude of the movement is smaller.

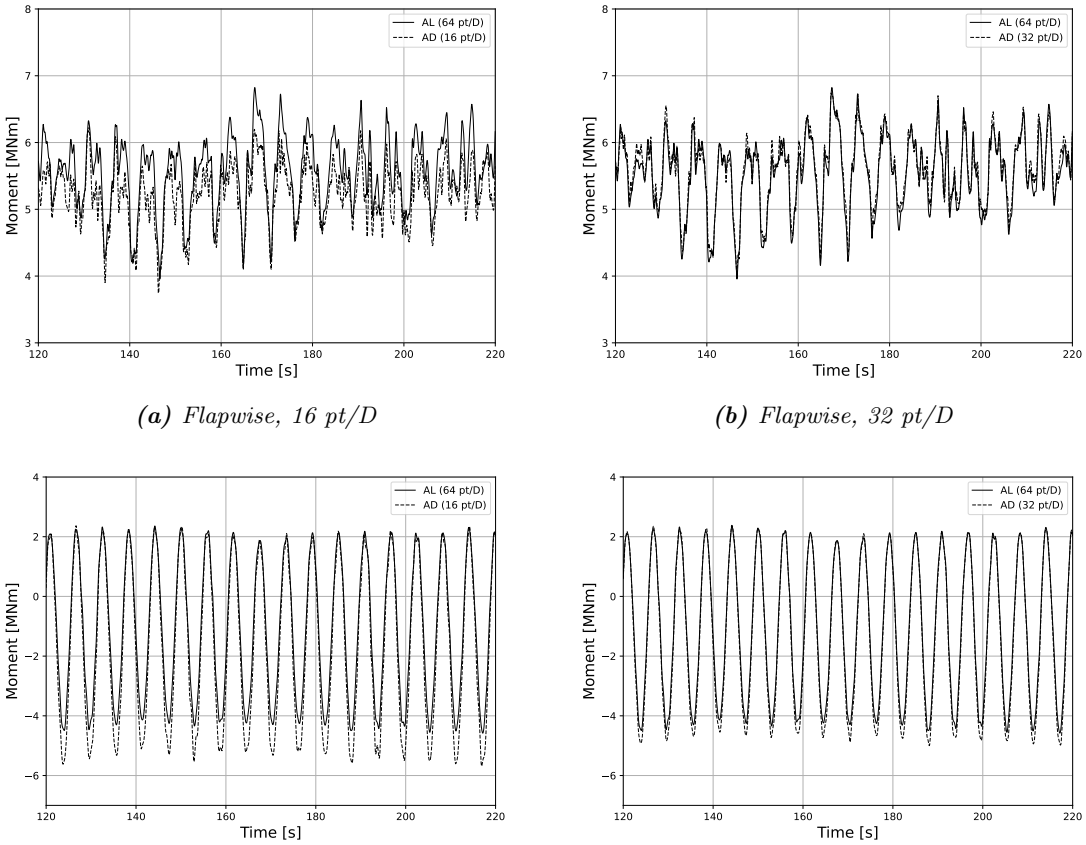
3.2.3 Power and thrust

Finally, we want to focus on the power and thrust which are two crucial parameters in wind turbines performance analysis. Thrust refers to the force exerted on the blades in the direction of the wind. While thrust is crucial for turbine efficient and stable operation, without overstressing the components, power output is not the key parameter of a wind turbine. It is the energy generated by the turbine as a result of harnessing the kinetic energy of the wind. This mechanical power is then converted into electrical power through a generator.
Power and thrust outputs are directly related to the aerodynamic behaviour, as seen in Eqs. 3.5 and 3.6, respectively defining power and thrust for one blade:

$$P = N_b M_e \Omega \tag{3.5}$$

$$T = N_b \Delta r \sum_{i=1}^{n_R} F_{n,i} \tag{3.6}$$

where N_b is the number of blades, Δr is the length of a blade element, n_R is the number of nodes on the blade, $F_{n,i}$ is the normal aerodynamic load acting on the ith node of the blade, M_e is



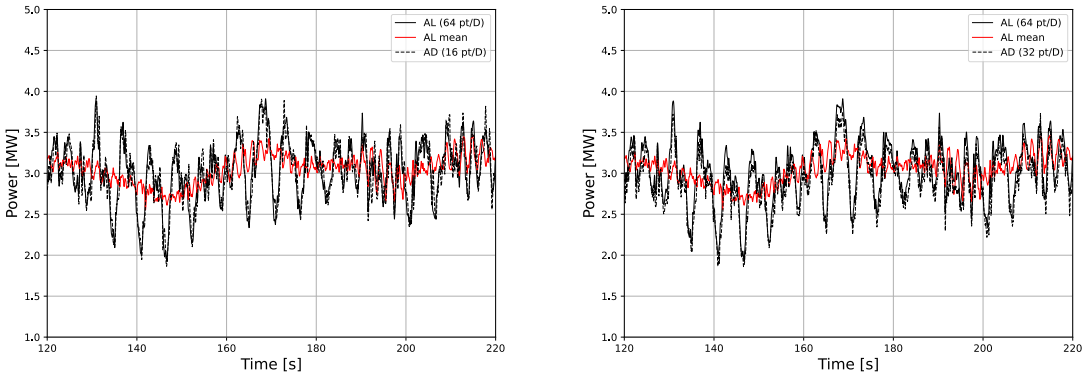
(a) *Flapwise, 16 pt/D*

(b) *Flapwise, 32 pt/D*

(c) *Edgewise, 16 pt/D*

(d) *Edgewise, 32 pt/D*

Figure 3.8 – *Structural root bending moments comparison between AL with 64 pt/D (solid) and different resolutions of AD (dashed) solved with DIM for* $\lambda = 7.54$.



(a) *16 pt/D* ***(b)*** *32 pt/D*

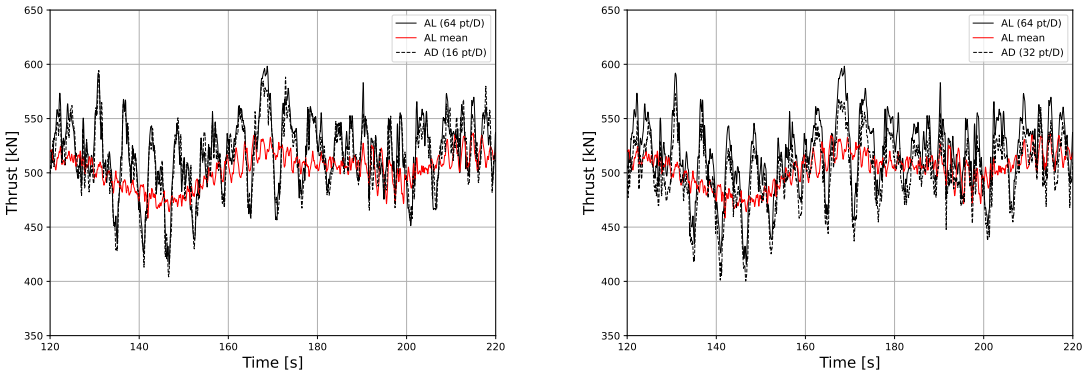
Figure 3.9 – *Comparison of power history for $\lambda = 7.54$ between AD (dashed), a single blade of AL (solid black) and the mean of the three blades of AL (solid red).*

the edgewise aerodynamic moment for one blade and Ω is the rotor speed in [rad/s].

Figures 3.9 and 3.10 represent the temporal evolution of power and thrust for $\lambda = 9.90$. In the same fashion as for the aerodynamic bending moments, power and thrust are displayed for AD representative blade-attached quantities and compared to the blade following the same trajectory in the AL method. The AL three-blade average is also represented in red.

As power is directly derived from the edgewise aerodynamic moment, identical results are observed regarding a single blade operation. For both resolutions, the results are extremely close to each other and to the AL's, and a difference between 16 and 32 pt/D can hardly be noticed. Although, when considering three blades for the AL, the AD with 16 pt/D predicts the power output better in all cases, as seen in Table 3.5.

In terms of thrust, the discrepancies between the AD and AL are slightly larger than for power but the AD curve keeps a correct aspect. It is consistent with the aerodynamic moments results as power is derived from the edgewise moment and thrust is computed using the same forces as the flapwise moment. We can also notice that for both one-blade behaviour and compared to the three-blade AL mean in Table 3.6, the 16 pt/D AD performs better in most cases.
Regarding the time-averaged values from Tables 3.5 and 3.6, power and thrust both perform quite well with a maximum difference of 7.6 % for power and 6.3 % for thrust, both for AD (32 pt/D) with $\lambda = 5.70$.



(a) *16 pt/D* ***(b)*** *32 pt/D*

Figure 3.10 – *Comparison of thrust history for $\lambda = 7.54$ between AD (dashed), a single blade of AL (solid black) and the mean of the three blades of AL (solid red).*

TSR [-]	AL	AD (16 pt/D)	AD (32 pt/D)
9.90	0.52 MW	-0.6 %	-5.5 %
7.54	3.03 MW	-0.7 %	-4.5 %
5.70	4.81 MW	-3.7 %	-7.8 %

Table 3.5 – *Time-averaged power comparison for different AD resolutions. Three-blade mean for AL and one-blade mean for AD.*

TSR [-]	AL	AD (16 pt/D)	AD (32 pt/D)
9.90	181 kN	+1.1 %	-1.8 %
7.54	502 kN	+2.4 %	-0.5 %
5.70	413 kN	-2.1 %	-5.4 %

Table 3.6 – *Time-averaged thrust comparison for different AD resolutions. Three-blade mean for AL and one-blade mean for AD.*

3.3 Mode superposition

The results obtained with the mode superposition method will be compared to those from the direct integration method. The motivation to use a modal resolution is the increased solving speed as the computations can be greatly reduced. The quantification of this gain is detailed in Section 3.3.1. The DIM will therefore be the reference for the other two methods, namely mode superposition accounting for all the modes of the blade, referenced as "complete modal", and mode superposition using only three modes. The AL curve is also displayed as a general reference. We shall expect the exact same results between the complete modal and the DIM as every mode is resolved and very close results for the three-mode methods as the three main modes of the blade are accounted for.

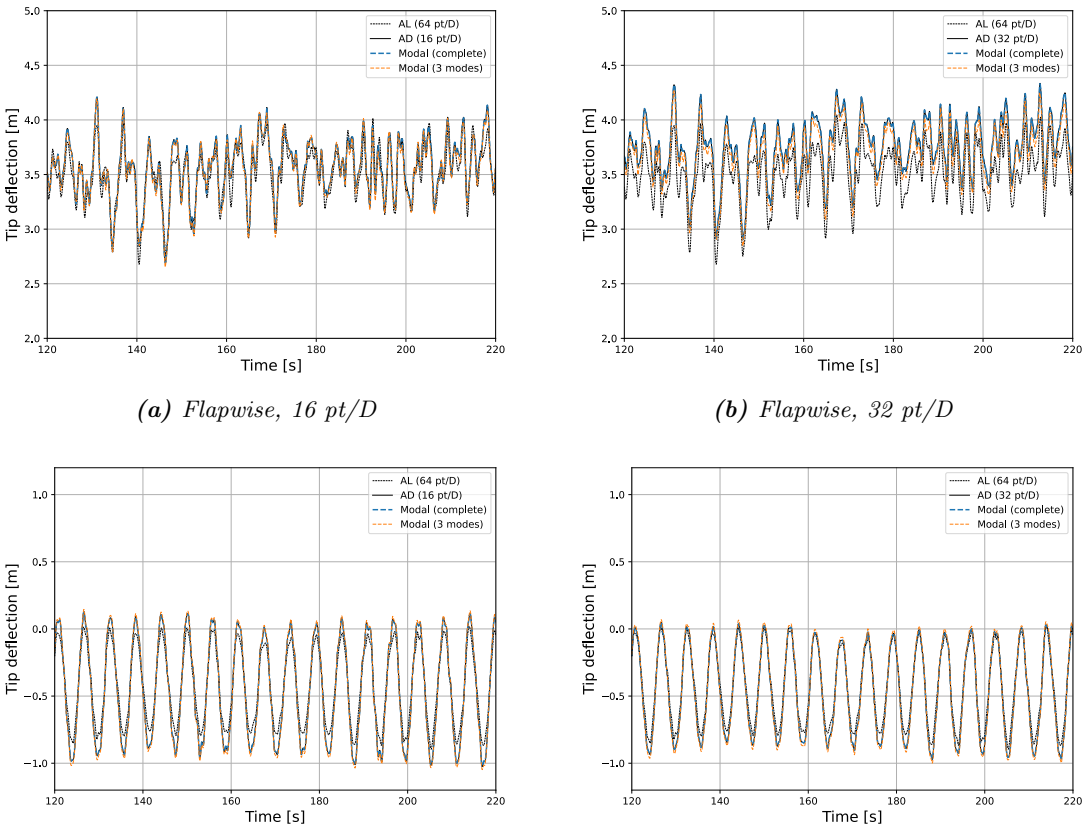
Tip deflection Results are shown for $\lambda = 7.54$ and $\lambda = 5.70$ in Figures 3.11 and 3.12 respectively. Overall, we observe that the three methods give very similar results with the complete modal yielding exactly the same ones as the DIM's, as expected. The curves from the three-mode method have the correct shape in every case with some small discrepancies in the flapwise direction, mainly for higher TSR due to a general overestimation of the displacement.
We can also notice that similarly to what was observed for the DIM with $\lambda = 7.54$, the flapwise deflection is still overestimated when using 32 pt/D, the modal methods thus don't correct this issue.

Loads Regarding the aerodynamic loads, the three methods concur exactly for $\lambda = 7.54$ but, when increasing the TSR, some slight differences appear for the three-mode method, mainly near the tip. For both modal methods, the first node starts from zero because, in the calculations, the blade is seen as a cantilever beam with its first node locked in place, hence with all degrees of freedom cancelled. It is thus skipped in the force computation process, which is problematic for investigating close-to-root behaviour but for an overall analysis this difference is trivial.

Aerodynamic moments All following results are displayed for $\lambda = 7.54$ only, as it is representative of the results of the modal methods for other TSRs. Regarding the aerodynamic moments in Figure 3.15, the three methods give identical results.

Power and torque Power and thrust histories are shown in Figures 3.16 and 3.17 respectively. As they both derive from the same forces as the aerodynamic moments, they exhibit the same behaviour and the three methods follow the exact same curve.

Structural moments The structural moment is found in Figure 3.18 and, in this case, some differences can be observed. The complete modal method still agrees exactly with the DIM, except for some negligible amplitude difference in the edgewise 16 pt/D case. The three-mode method



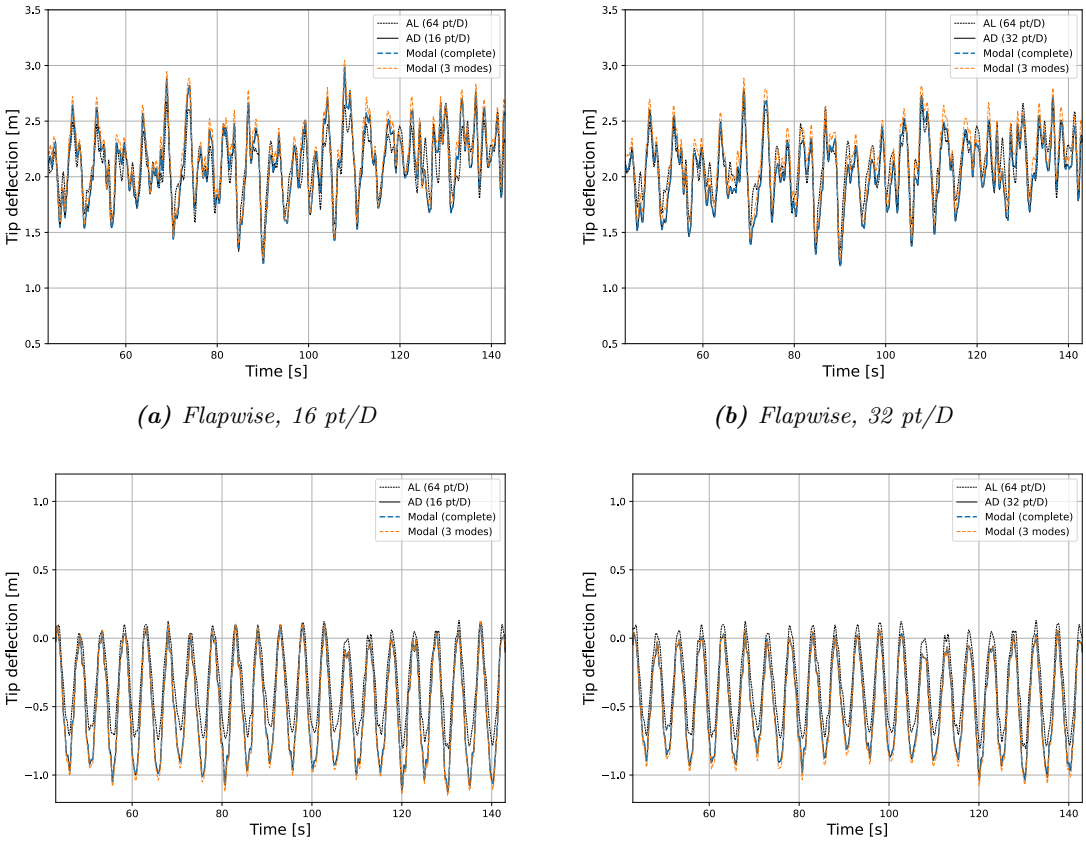
(a) *Flapwise, 16 pt/D*

(b) *Flapwise, 32 pt/D*

(c) *Edgewise, 16 pt/D*

(d) *Edgewise, 32 pt/D*

Figure 3.11 – *Tip deflection comparison between different AD methods for* $\lambda = 7.54$.



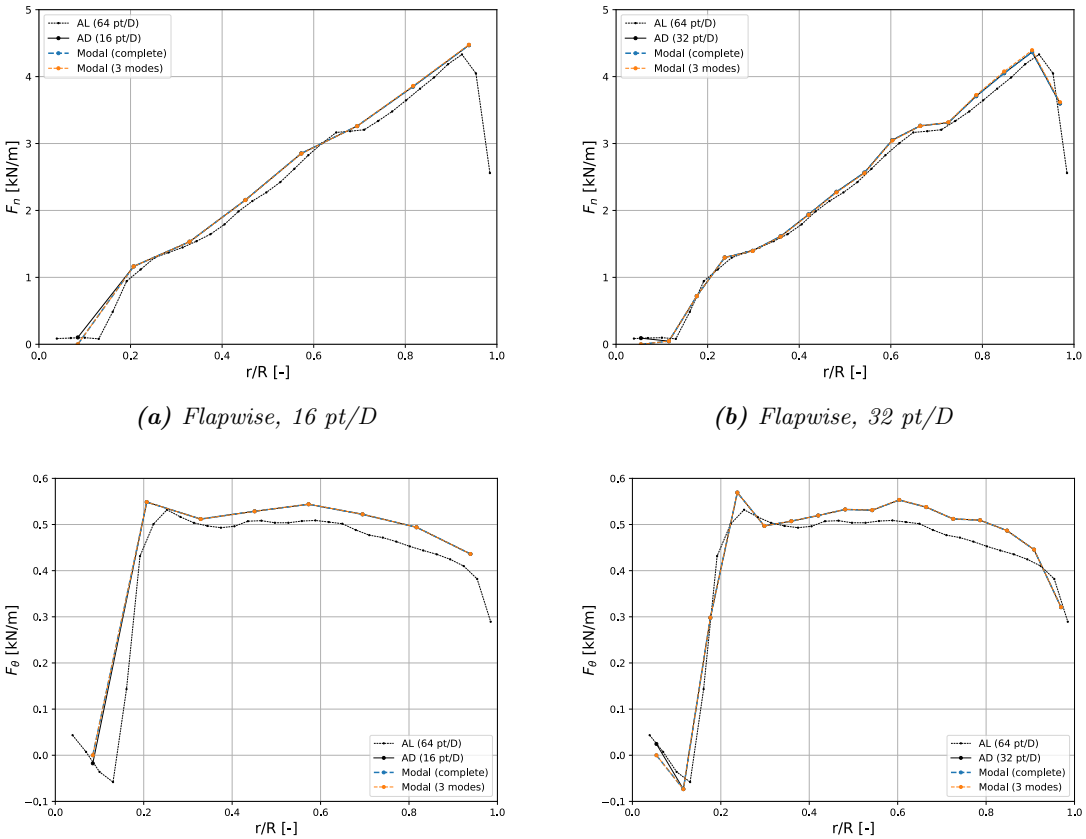
(a) *Flapwise, 16 pt/D*

(b) *Flapwise, 32 pt/D*

(c) *Edgewise, 16 pt/D*

(d) *Edgewise, 32 pt/D*

Figure 3.12 – *Tip deflection comparison between different AD methods for* $\lambda = 5.70$.



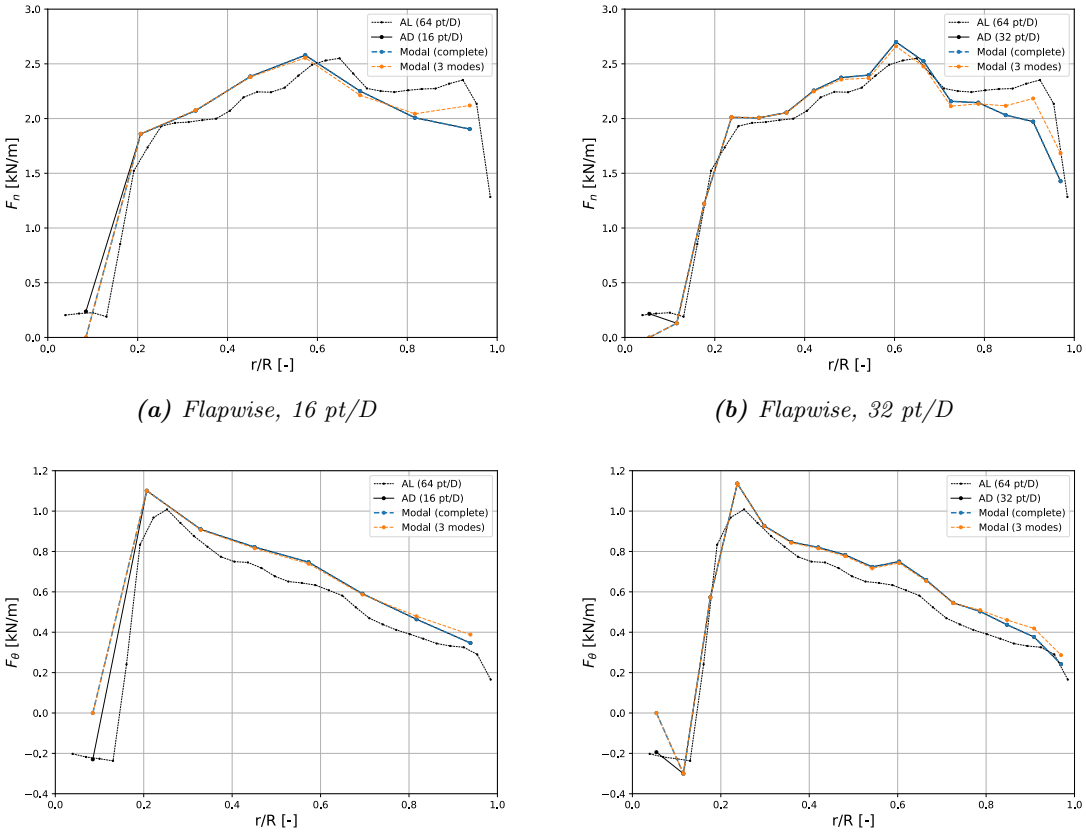
(a) *Flapwise, 16 pt/D*

(b) *Flapwise, 32 pt/D*

(c) *Edgewise, 16 pt/D*

(d) *Edgewise, 32 pt/D*

Figure 3.13 – *Time-averaged aerodynamic forces comparison between different AD methods for* $\lambda = 7.54$.



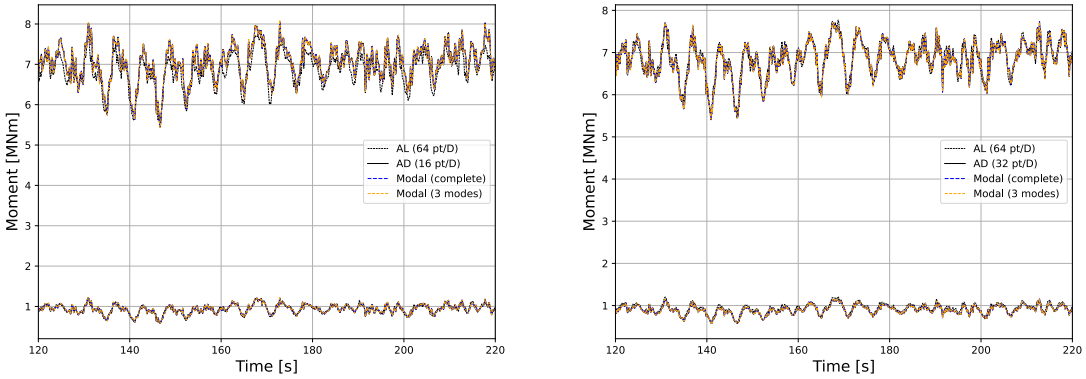
(a) *Flapwise, 16 pt/D*

(b) *Flapwise, 32 pt/D*

(c) *Edgewise, 16 pt/D*

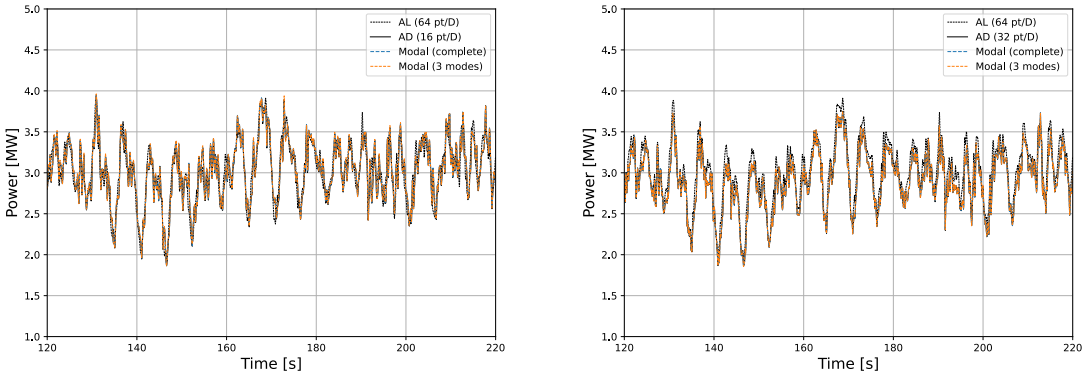
(d) *Edgewise, 32 pt/D*

Figure 3.14 – *Time-averaged aerodynamic forces comparison between different AD methods for* $\lambda = 5.70$.



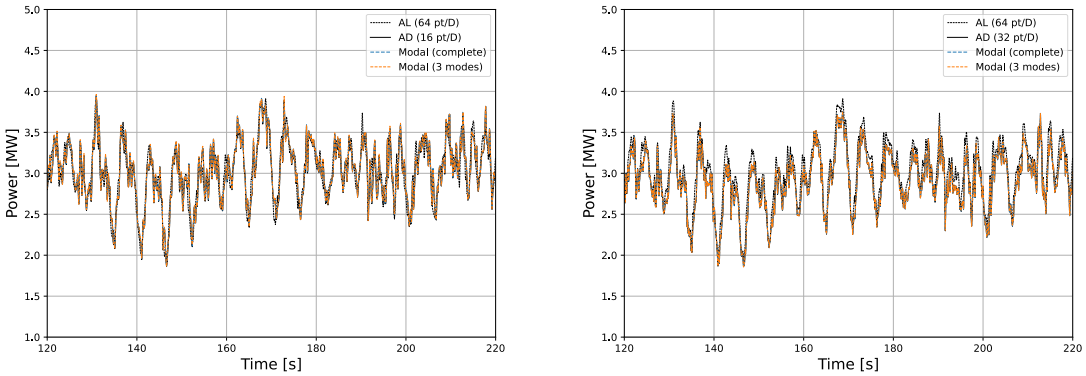
(a) *Flapwise, 16 pt/D* ***(b)*** *Flapwise, 32 pt/D*

Figure 3.15 – *Comparison of the aerodynamic moments between different AD methods for* $\lambda = 7.54$.



(a) *Flapwise, 16 pt/D* ***(b)*** *Flapwise, 32 pt/D*

Figure 3.16 – *Comparison of the power history between different AD methods for* $\lambda = 7.54$.



(a) Flapwise, 16 pt/D *(b) Flapwise, 32 pt/D*

Figure 3.17 – *Comparison of the thrust history between different AD methods for* $\lambda = 7.54$.

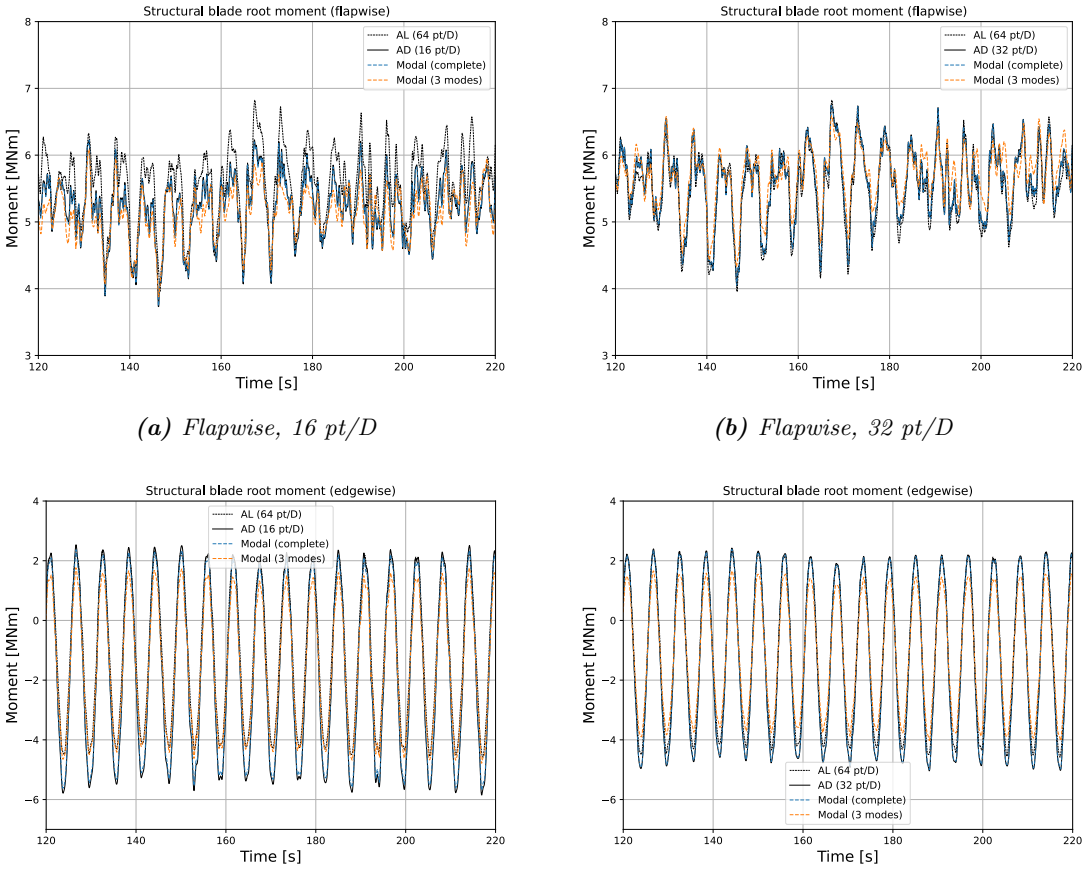
keeps a correct shape but displays some discrepancies in terms of amplitude.

In the case of 16 pt/D for the flapwise moment, the lower end of the curve, between 4 and 5 MNm, is estimated better with the higher ones being underestimated, whereas for 32 pt/D, the higher values, near 6 MNm, are approximated better and the lower ones are overestimated.
When it comes to the edgewise moment, the three-mode method estimates wrongly the amplitude which is too small in both cases, but in the other hand, the other two methods overestimate it. Considering an important amount of cycles, this difference can lead to erroneous results for the fatigue life of the blade.

To quantify those differences, Table 3.7 holds the values for the means of the structural moments for the three methods. The DIM and the complete modal give the exact same results and are therefore combined.
For the flapwise moment, the variation stays reasonably small with a better resolution for the DIM and complete modal, and an improvement for all methods when using 32 pt/D.
Regarding the edgewise moment, for 16 pt/D the deviation is extensively large for all cases as they don't predict the amplitude correctly. As we can see in Figure 3.18, the three-mode method is superimposed with the AL curve for the lower half values of edgewise 16 pt/D, whereas the other two go along with the AL at their upper end. Both methods are flawed but it results in a better approximation from the three-mode method with only 3.5 % difference with 32 pt/D. We also need to keep in mind that the values are given as percentages, and in terms of absolute values, 48 % of 1.14 edgewise is equivalent to only 10 % of 5.53 flapwise.

	AL (64 pt/D)		DIM & complete	Three modes
M_n	5.53 MNm	16 pt/D	-5.6 %	-7.2 %
		32 pt/D	+ 1.3 %	+2.9 %
M_θ	-1.14 MNm	16 pt/D	-48.2 %	-34.2 %
		32 pt/D	-17.5 %	-3.5 %

Table 3.7 *– Mean structural moments, flapwise (M_n) and edgewise (M_θ), for different AD methods compared to the reference AL for $\lambda = 7.54$. "complete" is the mode superposition method employing all the modes and "Three modes" is the mode superposition with only three modes used.*



(a) *Flapwise, 16 pt/D*

(b) *Flapwise, 32 pt/D*

(c) *Edgewise, 16 pt/D*

(d) *Edgewise, 32 pt/D*

Figure 3.18 *– Comparison of the structural moments between different AD methods for $\lambda = 7.54$.*

TSR [-]	9.90	7.54	5.70
Direct integration [s]	7397	5563	4210
Modal (complete) [s]	2014 (**27.2 %**)	1330 (**23.9 %**)	1133 (**26.9 %**)
Modal (3 modes) [s]	882 (**11.9 %**)	684 (**12.3 %**)	520 (**12.4 %**)

Table 3.8 *– Runtime comparison for post-processing operations between different methods and TSRs for 16 pt/D. Runtimes for modal methods are compared to the direct integration method with a percentage of its total runtime.*

TSR [-]	9.90	7.54	5.70
Direct integration [s]	36072	26079	19857
Modal (complete) [s]	7446 (**20.6 %**)	13061 (**50.1 %**)	4912 (**24.7 %**)
Modal (3 modes) [s]	3079 (**8.5 %**)	2375 (**9.1 %**)	1813 (**9.1 %**)

Table 3.9 *– Runtime comparison for post-processing operations between different methods and TSRs for 32 pt/D. Runtimes for modal methods are compared to the direct integration method with a percentage of its total runtime.*

3.3.1 Computational cost

In order to analyze the computational cost for the different methods, we need to recall how the convergence procedure works.
The converging process described thoroughly in Section 2.5.1 can be decomposed as follows. Each time step of the global simulation is defined by a frozen velocity field considered on the rotor area. Using EBBT, a single blade represented as a flexible beam is set to rotate in the velocity field and solved dynamically for each azimuthal element of the disk. When the beam is back to its original position, its displacement throughout the entire rotation is compared to that of the previous rotation, and if both are close enough, the last rotation is the solution chosen as final and periodic.
To assess the effectiveness of this process, two parameters need to be analyzed. First, the cost of the dynamic solving between each azimuthal element of a rotation, and also the number of rotations needed to find the periodic solution.

Dynamic solving During the process of dynamic solving, two computational operations take place: the calculation of the aerodynamic loads from the local wind velocity and the resolution of the system of ODEs. Both operations are longer as the resolution gets finer, because the number of nodes on the blade increases.

The number of computations of the first part is constant for a given resolution as the forces are computed at each node once, and it is common to all three methods as it serves as the building block for all subsequent computations.
However, the second process can grow significantly from coarser to finer resolution, but it can also

	Total time [s]	Forces computation	System resolution
DIM	0.235	9.9 %	89.8 %
Modal (complete)	0.060	60.8 %	36.2 %
Modal (3 modes)	0.028	74.1 %	22.5 %

Table 3.10 *– Comparison of different methods for the runtime to solve one rotation of the blade in a frozen velocity field and execution time distribution (%) among the two main computational processes.*

be greatly reduced from one method to another. The set of ordinary differential equations that is solved to get the dynamic behaviour of the beam is defined by (2.18) used in its tensor form. The cost of resolution of this system is determined by its dimension, i.e. the size of the matrices M, C and K, which represent the internal properties of the blade. Those matrices aim at representing the blade through its nodes and for its different degrees of freedom. The ODEs are solved using Newmark integration scheme, which most computationally costly part involves the calculation of the acceleration through the inversion of the mass matrix.

For the direct integration method, considering all six degrees of freedom are taken into account and n_R is the number of nodes splitting the blade, the matrices will have a size ($6\ n_R \times 6\ n_R$).
For the mode superposition method, when considering all the modes composing the blade, the matrices will have the same size as the n_R nodes all have six associated modes (one for each degree of freedom). Therefore, for a resolution of 16 points per D, $nR = 8$ and the system is of dimension 48, and for a resolution of 32 points per D, $nR = 16$ and the system dimension is 96. The DIM and complete modal method may have the same system dimension but due to simplifications detailed in Section 2.5.2, the solving is much faster for the modal method as seen in Tables 3.8 and 3.9.

The details of the cost distribution between the computation processes is presented in Table 3.10, highlighting the fact that the resolution of the ODEs becomes the lower cost process while the computation of forces takes most of the time when using the complete modal method. The case 16 pt/D is shown but a similar distribution is observed for 32 pt/D.
Regarding the third method, using mode superposition with three modes, the effect just stated extends even more as the matrices of the system are of size (3×3) (see Section 2.5.2) and the forces computation stays constant.

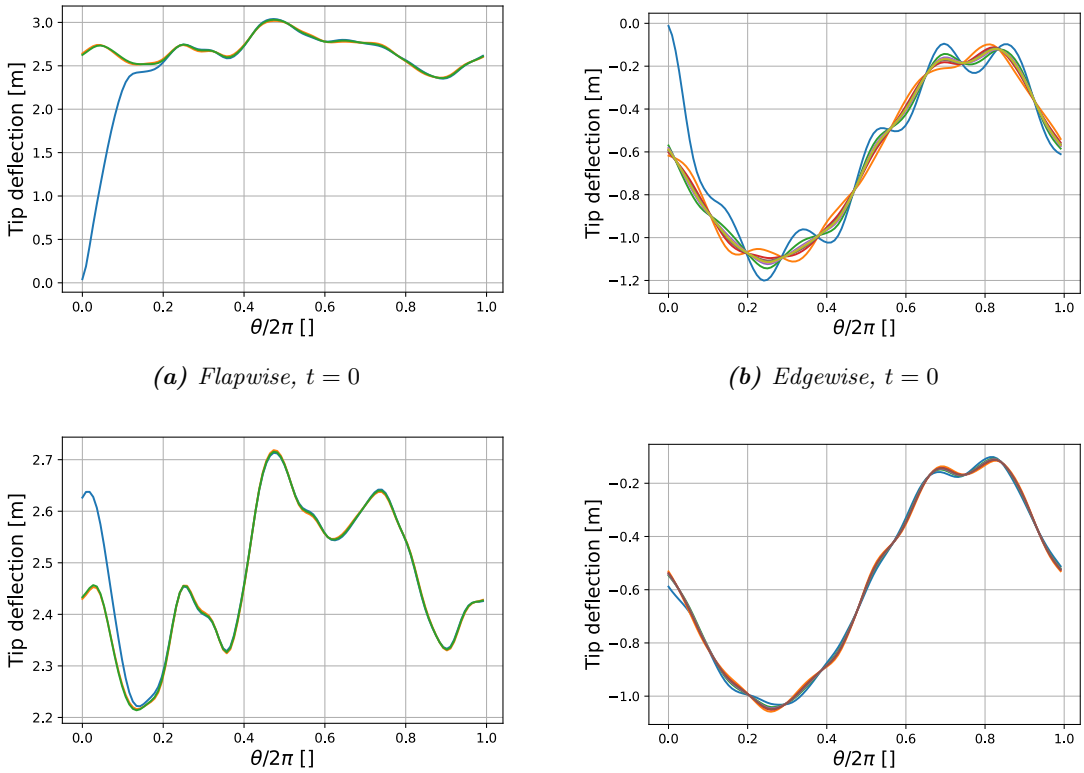
Number of rotations The other key parameter in terms of convergence performance is the number of rotations needed to find the periodic solution. It is however only important at the start of the simulation, when the blade starts from rest and needs to stabilize. This phase occurs for the first time steps, i.e. when the first sets of frozen velocity fields are considered, but particularly for the very first one as the structural solver has no previous solution to start solving the transient. The blade initial position at rest, when $t = 0$, is on the rotor plane.

Figure 3.19 represents this process of flapwise and edgewise stabilization by displaying the tip deflection during a rotation of the beam in a frozen velocity field. The data are taken from the AD simulation with resolution 32 pt/D and $\lambda = 5.70$ as representative for all the other simulations. In this example, the tip deflection is shown but its behaviour reflects what happens at every node of the blade.

Figures 3.19a and 3.19b describe the tip deflection when the first time is considered ($t = 0$), i.e. the first frozen velocity field is used. In this frozen field, the beam has to make a certain amount of rotations until the periodic solution is found. Each colour on the graphs is associated with one rotation of the beam, with the blue one corresponding to the first rotation made as the tip deflection starts from zero, which corresponds to the undeformed blade, when it sits on the rotor plane. From there, every new rotation takes the last blade position as input in the dynamic computations, making the solution converge until it reaches its periodic orbit. We can clearly see that the edgewise tip deflection needs more rotations to converge to its periodic solution as specified in Table 3.11 for 16 pt/D. The difference between flapwise and edgewise convergence is mainly observable for $\lambda = 5.70$, but for the three TSRs it only requests two time steps to reach the minimum number of rotations (2, 2) (see Section 2.5.1). The only noticeable difference for the case 32 pt/D is for $\lambda = 5.70$ which reduces to (3, 9) for $t = 0$ and (3, 6) for $t = dt$.

The edgewise convergence is harder to reach due to the aerodynamic damping which has a large impact on the displacement stability. This kind of damping is due to the aerodynamic forces such that if the blade moves away from the rotor plane, it will induce a reduction of the force in this direction tending to keep the blade closer to the rotor. Conversely, if the blade tends to get closer to the rotor plane, the force opposing the movement will increase. This effect has a considerable influence flapwise and almost none edgewise. Aerodynamic damping phenomenon is detailed in Appendix B.

Figures 3.19c and 3.19d represent the same process of convergence towards a periodic solution but for the frozen velocity field after one time step, when $t = dt$. In this case, and for all the following frozen fields, i.e. for $t = i\ dt$ with $i = 1, ..., nt$, the structural solver uses the periodic solution found at the previous time step as the initial value for the computations. In other words, the tip starts from the position found in the periodic solution at $t = (i - 1)\ dt$, allowing to reach much faster the periodic solution, as less rotations are needed.



(a) *Flapwise, $t = 0$*

(b) *Edgewise, $t = 0$*

(c) *Flapwise, $t = \Delta t$*

(d) *Edgewise, $t = \Delta t$*

Figure 3.19 – *Tip deflection convergence process for the first two frozen velocity fields ($t = 0$ and $t = \Delta t$). Each colour represent a distinct rotation, with blue indicating the initial rotation in all graphs.*

TSR [-]	$t = 0$	$t = dt$	$t = i\ dt$*
9.90	(3, 4)	(3, 3)	(2, 2)
7.54	(3, 5)	(3, 3)	(2, 2)
5.70	(4, 22)	(3, 13)	(2, 2)

Table 3.11 – *Number of rotations needed to reach periodic solution for different frozen velocity fields and TSRs, with a resolution of 16 pt/D. Each pair (n_1, n_2) denotes the rotations required for flapwise and edgewise convergence, respectively. *$i = 2, ..., nt$.*

3.4 Assessment of aeroelastic effects

In this section, the effect of the aeroelasticity, i.e. the flexibility of the blades, is highlighted in terms of loads, power and thrust, and wake dynamics.

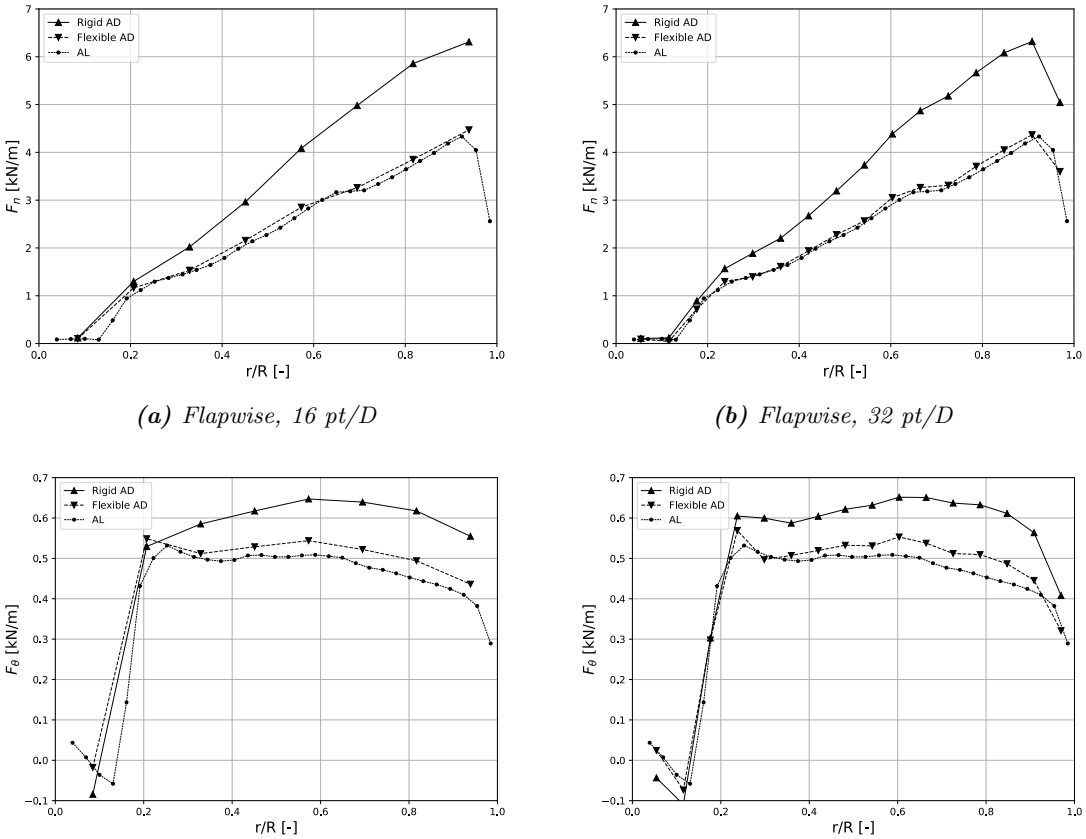
Loads analysis First, we recall that currently the simulation does not account for the flexibility of the blades and does not simulate real blades moving, instead it considers an average effect of the n_B blades on the whole disk surface. The method developed in this work added the blade flexibility in the computation of the loads and interpolated the real movement of a single blade from which the results are derived.

During the simulation, the forces are computed using the classical blade element theory described in Section 2.3.1 for every control point of the disk and for every time t, therefore not accounting for structural deformation. We thus have nt sets of values of size (n_r, n_θ) for flapwise and edgewise forces. In order to extract the averaged loads on the blades, we first average the values azimuthally over 2π, leaving nt sets of values of size (n_R), and then average it over time for the second half of the simulation, giving the n_R averaged values. This result is the time-averaged effect of the n_B blades on the whole disk reduced to the span of one blade. So, to approximate the effect of one blade, we consider the total average effect and divide it by n_B.

Figure 3.20 shows the loads obtained from the averaged effect of the rigid blades, called "Rigid AD", compared with those computed using the direct integration method and named "Flexible AD", and with the reference AL solved with 64 pt/D.

Power and thrust Power and thrust are also derived from the effect of the n_B blades averaged over the whole disk as they are computed from the loads described in the previous section. They are compared with the Actuator Line and the flexible AD graphically in Figures 3.21 and 3.22, and in terms of time-averaged values in Table 3.12.
Similarly to the observations on the loads, the rigid AD overestimates the values while exhibiting a general correct shape.

Wake behaviour Finally, wake statistics are investigated in terms of instantaneous velocity and vorticity for different resolutions of the rigid AD and compared to the flexible AL using 64 pt/D. The three cases shown in Figure 3.23 represent the same simulation performed in a turbulent flow with mean velocity 9 m/s ($\lambda = 7.54$), and at the same time of the simulation, when the wake is fully developed.



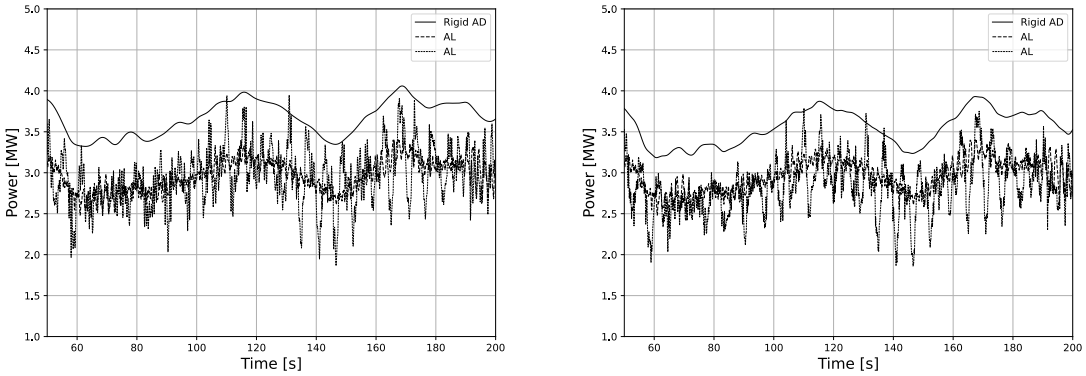
(a) *Flapwise, 16 pt/D* ***(b)*** *Flapwise, 32 pt/D*

(c) *Edgewise, 16 pt/D* ***(d)*** *Edgewise, 32 pt/D*

Figure 3.20 *– Time-averaged aerodynamic forces comparison between rigid AD, flexible AD and reference AL for* $\lambda = 7.54$*.*

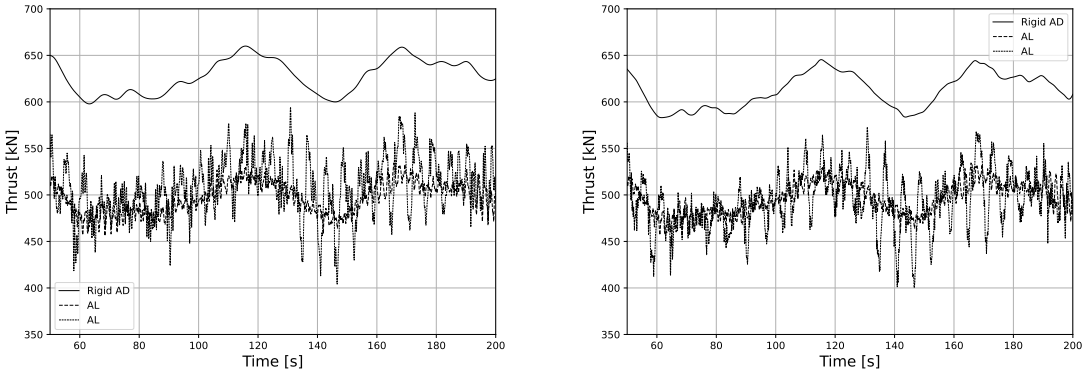
	AL		Flexible AD	Rigid AD
P	3.03 MW	16 pt/D	-0.7 %	+23.8 %
		32 pt/D	-4.5 %	+19.8 %
T	5.02 kN	16 pt/D	+2.4 %	+26.5 %
		32 pt/D	-0.5 %	+23.3 %

Table 3.12 *– Time-averaged power and thrust comparison between flexible and rigid AD, and the reference AL (64 pt/D).*



(a) *Flapwise, 16 pt/D* ***(b)*** *Flapwise, 32 pt/D*

Figure 3.21 – *Comparison of the power history between rigid AD and AL for* $\lambda = 7.54$.



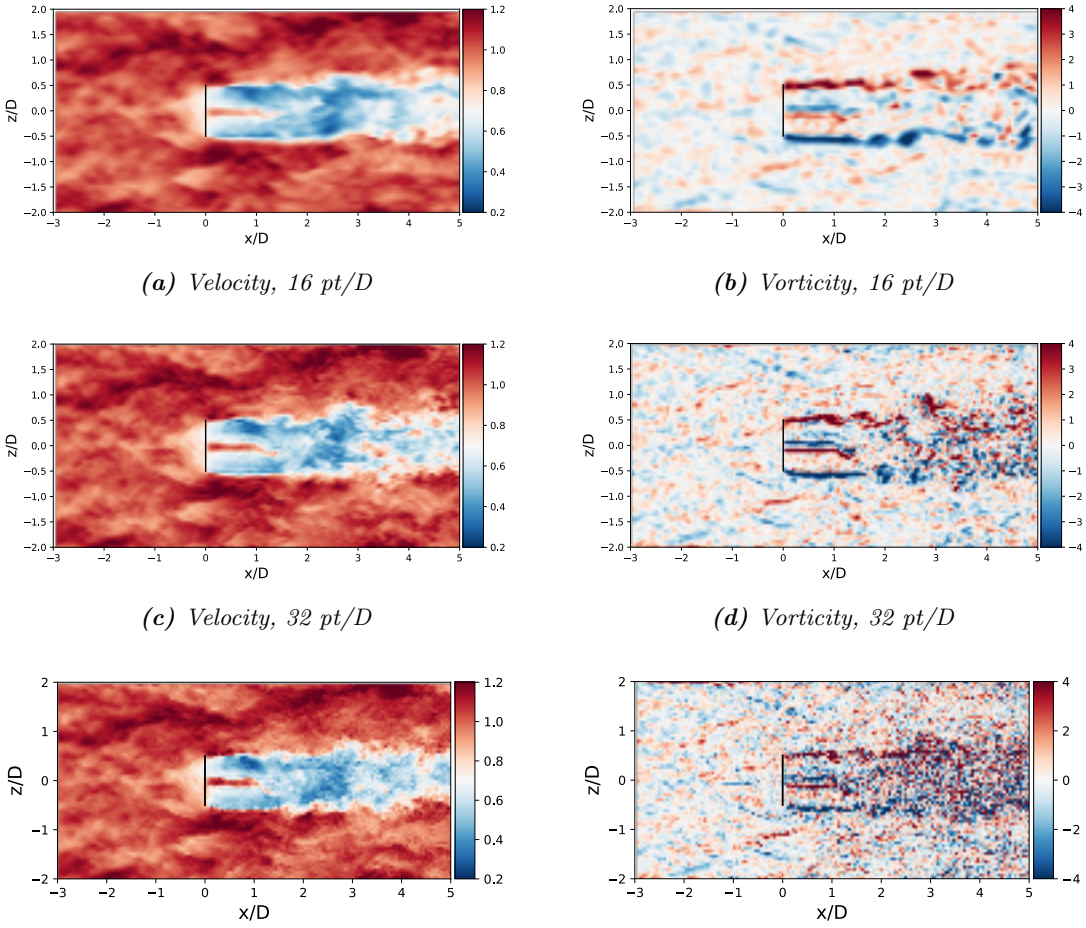
(a) *Flapwise, 16 pt/D* ***(b)*** *Flapwise, 32 pt/D*

Figure 3.22 – *Comparison of the thrust history between rigid AD and AL for* $\lambda = 7.54$.

Both methods have different approaches regarding the way they capture the physics of the wake as the AD represent the rotor as a fixed disk, and the AL approach consider discrete blades rotating in the flow. In an real rotor, the blades generate separate vortex sheets that coalesce to create concentrated vortices at the tip and hub. These vortices interact and play a role in disrupting the stability of the wake. Inherently to the AD method which continuously considers an average effect across the whole disk, the wake behind it is a continuous cylindrical vortex sheet, making it unable to capture the near-wake dynamics. However, the model can be modified and corrected to account for complex effects such as wind turbine wake interactions in a wind farm as done in Machefaux *et al.* [13] and Gomes *et al.* [8].
In addition to that, the AD is typically used at coarser resolutions, disregarding the small scales in the flow, and impairing its ability to represent accurately the wake physics.

The effect of coarse resolution is strongly noticed in Figure 3.23 with the wake structures becoming gradually simpler when the resolution decreases, and being mostly observable for the vorticity. However, in terms of velocity the overall behaviour is correctly defined at coarser resolution and describes well enough the wake velocity for the rigid AD. For both velocity and vorticity, large scale perturbations are relatively well conserved with smoother representation as the resolution decreases.

We need to recall that the wakes currently available for the AD only exist for the rigid version. In order to compare it with the wake produced by the flexible AD, the structural resolution needs to be concurrent to the flow resolution to allow fluid-structure interaction, as discussed in Section 4.2.3.



(a) *Velocity, 16 pt/D*

(b) *Vorticity, 16 pt/D*

(c) *Velocity, 32 pt/D*

(d) *Vorticity, 32 pt/D*

(e) *Velocity, 64 pt/D*

(f) *Vorticity, 64 pt/D*

Figure 3.23 – *Instantaneous streamwise velocity and vorticity for rigid AD (16 and 32 pt/D) and for flexible AL (64 pt/D) for* $\lambda = 7.54$. *The rotor hub is represented by a vertical black line.*

Chapter 4

Conclusion and perspectives

4.1 Conclusion

This work had for objective to develop a method that effectively models the aeroelastic effects of wind turbine blades, aiming for accurate results while managing computational expenses, with an eye towards its applicability in wind farm simulations.
Large Eddy Simulation coupled to an actuator model was used to simulate the flow and turbine interaction. The first actuator model considered was the Actuator Line which was chosen as reference throughout this study as this method is already developed to account for the deformation of the blades. However, when considering wind farm simulations, the computational cost become substantial and the AL approach becomes inapplicable. In order to address this issue, the Actuator Disk method which is more efficient computationally is considered. The AD approach however does not account for blade flexibility, and this work was dedicated to the development of different methods to include aeroelasticity as a post-processing operation.

The AL simulation inherently considers separate blades rotating in the flow, accurately representing their behaviour and allowing fluid-structure interaction. On the other hand, the AD represents the rotor as a rigid disk, averaging the effect of the blades over its whole surface. In order to include the flexibility effect in the computations, three methods have been developed. The methods are assessed for different grid resolutions and flow velocities, and are their results are then compared to those obtained from the AL.

First, a direct integration method that solves the transient dynamics until a periodic solution is found for every time of the simulation has been considered. This method is heavy computationally but its purpose is to be as complete and precise as possible. The results show good agreement between the Actuator Disk methods and the reference Actuator Line model in terms of blade de-

flection, loads, power, and thrust.
Then, a mode superposition method using the vibration modes of the blade has been developed to allow much faster resolution due to simplifications in the solving of the system describing the dynamic behaviour of the blade. This method yields results matching exactly those of the direct integration method while alleviating the computational cost.
Finally, the modal method has been enhanced computationally to only account for the three main dominant modes to solve the structural part. This method decreases a lot the time of the calculations while producing very close results compared to the other two methods.

The last part highlights the aeroelastic effects in the AD approach by comparing the raw results from the simulation with the enhancements developed in this work. Important differences can be noticed for loads applied on the blade and for power and thrust histories as the AD disregarding the aeroelastic effect overestimates those values. It is therefore important to use and improve the flexible AD methods for accurate representation of the physics of the blades.

In conclusion, the developed Actuator Disk methods show promising results and with further improvements could provide an efficient and accurate tool for wind farm simulations.

4.2 Perspectives

4.2.1 Multiple shooting algorithm

In addition to the methods presented in this work, another method had been considered and initial implementation efforts were undertaken. However, due to time constraints, this approach was not completed. The method and its integration into the currently used process will be presented here.

The method is based on a multiple shooting algorithm to compute the periodic orbits of non-linear ordinary differential equations systems through the use of Floquet multipliers [12]. The associated Python toolbox [18], named multiflap, has been implemented by Ducci *et al.* for the analysis of stability and sensitivity of bird flapping flight [4]. The use of Floquet multipliers allow, in addition to solving the system, to study its stability.

Description of the method

The multiple shooting algorithm is an iterative numerical technique used to solve boundary-value problems associated with ordinary differential equations. This method works by breaking down the problem of finding periodic solutions into smaller time intervals. It starts by guessing initial values for the periodic orbit, divides the time domain into segments, and solves the differential equations

for each segment separately. The algorithm then adjusts the guessed initial values iteratively until the solution of each segment matches continuously at the segment boundaries, ensuring that the solution obtained is periodic. By iteratively refining the initial guesses and solving the corresponding subproblems, the multiple shooting algorithm converges towards an accurate solution representing the limit cycle of the dynamical system.
In a dynamical system, such as those described by differential equations, a limit cycle represents a stable periodic solution. In other words, it corresponds to the periodic orbit formed by the trajectory of the system's state over time. Considering a general ODE system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t)$, the limit cycle would write $\mathbf{x}^*(t) = \mathbf{x}^*(t + T)$ where T is the period of the orbit.

It applies well in the context of this work as it solves the problem described in Section 2.5, which was to find, for each time step of a given Actuator Disk simulation, the periodic trajectory made by the blade in a frozen velocity field. The difference is that the process used in this study is based on solving the transient for every time step. This means that the blade is considered in the velocity field and its position is updated at every control points until the physical position of the blade reaches a periodic state. If the velocity field is very turbulent and the precision required to determine when convergence is reached is very high, this can lead to a large number of rotations. The advantage of the multiflap method is that it allows to avoid the solving of the whole transient phase to detect only the limit cycle by using initial and boundary conditions to find a solution that satisfies the periodicity constraints.

Application

Integration The way to integrate multiflap into the currently used method would be to replace the whole convergence process (Section 2.5.1) by the application of the multiple shooting algorithm. Multiflap would thus be used at every time step, i.e. once for each different frozen velocity field, and the periodic orbits found would then be used for the interpolation giving the real blade position over time.
The building block is the same as for the other methods, which is the computation of the loads on the blade from the local wind velocity. This process must be included in the ODE system as the external forces term $F(t)$.

Input The method requires four main input data related to the problem itself: an initial guess, a period guess, a system of ODEs, and a stability matrix (optional).
Then, the multiple shooting scheme is determined by four numerical parameters: the integrator order, the iteration error, the number of time steps over a period, and the amount of points composing the trajectory, in between which the integration is performed.

First, some observations and advice are given for the numerical parameters. In terms of performance, the fourth order Runge-Kutta scheme seems to be the most appropriate method in this case. Then, the number of points in the trajectory and the integration time step are parameters that depend on one another and should be tested to know exactly what configuration works best for the situation. Finally, the iteration error should be kept low, around 10^{-5}, for the scheme to give relevant results.

Regarding, the input for the problem, the initial guess x_0 is a key parameter of the method as it can be quite sensitive relative to the choice of x_0. In our case of the trajectory of a blade in a frozen velocity field, it can be hard to find an initial condition, but a possible solution is to use already known data for other simulations in the same conditions. Taking the average displacement of the blade over a simulation as initial guess.
On the other hand, if the rotor speed is constant, the period is easily computed as $T = \frac{2\pi}{\omega}$ with ω the rotor speed in rad/s.
Regarding the ODE system, we use the dynamic equation of motion (Eq. (4.1)) which governs the physical behaviour of the blade when external load is applied, and was already used previously to solve the transient.

$$M\ \ddot{x}(t) + C\ \dot{x}(t) + K\ x(t) = F(t) \tag{4.1}$$

The method requires first order differential equations. Eq. (4.1) is thus divided into two first order ODEs by keeping the position and the velocity as unknowns and expressing the acceleration as the derivative of the velocity. The position becomes variable x_1 and the velocity is x_2, which gives

$$\begin{aligned} v_1 &= \frac{\partial x_1}{\partial t} = x_2 \\ v_2 &= \frac{\partial x_2}{\partial t} = M^{-1}(F - K \cdot x_1 - C \cdot x_2) \end{aligned} \tag{4.2}$$

where v_1, v_2, x_1, x_2 are vectors, and M^{-1}, C, K, F are matrices with M^{-1} the inverse matrix of M.

In order to compute the Floquet multipliers determining the stability of the system, as well as the Jacobian matrix used in the multiple shooting scheme, the stability matrix defined by Eq. (4.3) is required.

$$\mathbb{A}_{i,j} = \frac{\partial v_i}{\partial x_j} \tag{4.3}$$

The stability matrix can be given as input or computed numerically during the process. The analytical choice is however much faster and can be easily computed.
Calling m the number of states of the dynamical system and applying Eq. (4.3) on Eqs. (4.2), the

stability matrix can be divided into four parts, each of size $[\frac{m}{2} \times \frac{m}{2}]$, such that

$$\mathbb{A}_{i,j} = \begin{cases} 0 & \text{for } i,\ j = [1,\ \frac{m}{2}] \\ I_{\frac{m}{2}} & \text{for } i = [1,\ \frac{m}{2}],\ j = [\frac{m}{2}+1,\ m] \\ -M^{-1} \cdot K & \text{for } i = [\frac{m}{2}+1,\ m],\ j = [1,\ \frac{m}{2}] \\ -M^{-1} \cdot C & \text{for } i,\ j = [\frac{m}{2}+1,\ m] \end{cases} \tag{4.4}$$

where $I_{\frac{m}{2}}$ is the identity matrix of dimension $\frac{m}{2}$ and the operator "$\cdot$" corresponds to matrix multiplication.

Modal superposition

Similarly to the direct integration method, a transformation to modal analysis can be performed in the use of multiflap. Following the procedure detailed in Section 2.5.2, the ODE system (Eq. (4.1)) takes a modal form given by Eq. (2.22) through the use of the transformation matrix ϕ. This modal transformation allows a much faster resolution of the system of size $[6n_R \times 6n_R]$, with n_R the number of nodes dividing the blade.
Finally, as described in Section 2.5.2, we can use only three modes and reduce the system to a size $[3 \times 3]$ to decrease even more the computational cost. This method using three modes give very convincing and accurate results when used with the other methods, using it in multiflap should be quite beneficial.

4.2.2 Wind farm simulation

The motivation to use the Actuator Disk method is its reduced computational cost compared to other methods, such as the Actuator Line method, especially to be used for wind farm simulations, which require very heavy computations.

In this study, the developed methods have been assessed only with isolated wind turbines, which is the first step to validate the method. However, the ultimate aim is to be able to simulate the behaviour of turbines as they influence one another in a wind farm situation, and estimate the impact on the resulting power and the wake dynamics. The next step to validate the methods is therefore to analyze the results from a wind farm simulation.

4.2.3 Integration in main simulation

Currently, the method has been developed for post-processing usage, which means that the wind velocity used for the loads computation is determined before the method is applied and does not depend on the flexibility of the blades. The fluid-structure interaction is therefore not taken into account and the wake dynamics as well as the real behaviour of the blade cannot be accurately

represented.

However, in wind farm configurations, turbine-wake interactions is a critical parameter that needs to be precise as it can quickly become very complex. Hence the need to develop a method producing accurate results and with the purpose to make real time computations. Knowing various data such as power output or loads undergone by the blades continuously would allow to control the different parameters of the turbines to optimize power production and load alleviation.
This can be achieved by updating the blade dynamics concurrently to solving the flow to allow interaction between both and accurate dynamics representation.

Appendices

Appendix A

Structural root bending moment

This appendix explains how the structural root bending moment is computed from the beam displacement using finite element method for beams described in [20].
The beam is divided into short elements that have uniform cross-section and constant elastic properties, and the beam is considered planar, flapwise and edgewise moments are thus dealt with in two separate computations. The flapwise moment uses the values of displacement in the plane containing the beam that is perpendicular to the rotor, i.e. in the direction of the wind, and the edgewise moment applies on the rotor plane. In such a configuration, each element composing the beam has a length l_e and is delimited by two nodes, η_1 and η_2, leading to two degrees of freedom per node, displacement in the transverse direction (u_1 and u_2) and rotation around the axis normal to the beam plane (θ_1 and θ_2). Each beam finite element thus has a total of four degrees of freedom.

The beam elements presented in this chapter are based on the Euler-Bernoulli theory which

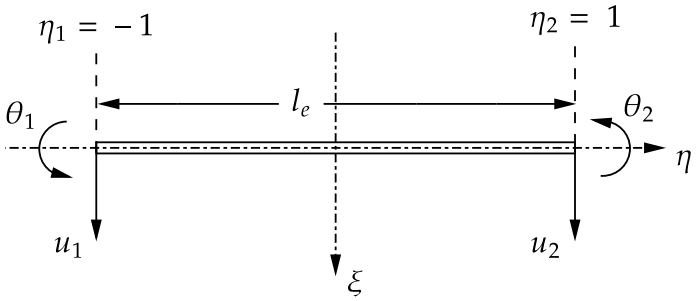


Figure A.1 *– Configuration and variables for a beam element using finite element method.*

computes the bending moment from the second derivative of node deflection as

$$M = -EI\frac{\partial^2 \tilde{w}}{\partial x^2} \tag{A.1}$$

Using the finite element method, the displacement along the element is written as

$$\tilde{w}(\eta) = \overline{\mathbf{N}}(\eta)\overline{\mathbf{u}} \tag{A.2}$$

where $\tilde{w}$ is the axial approximation within the finite element e, $\overline{u}$ is the vector of finite element displacements, defined as

$$\overline{\mathbf{u}} = \begin{bmatrix} u_1 \\ \theta_1 \\ u_2 \\ \theta_2 \end{bmatrix} \tag{A.3}$$

and N is the matrix of shape functions defined as

$$\overline{\mathbf{N}}(\eta) = [N_1(\eta)\ N_2(\eta)\ N_3(\eta)\ N_4(\eta)] \tag{A.4}$$

where the shape functions are written as

$$\begin{aligned} N_1(\eta) &= \frac{1}{4}(2 - 3\eta + \eta^3) \\ N_2(\eta) &= \frac{l_e}{8}(1 - \eta - \eta^2 + \eta^3) \\ N_3(\eta) &= \frac{1}{4}(2 + 3\eta - \eta^3) \\ N_4(\eta) &= \frac{l_e}{8}(-1 - \eta + \eta^2 + \eta^3) \end{aligned} \tag{A.5}$$

N_1 and N_3 are the two translational shapes functions corresponding to the displacement degree of freedom, while N_2 and N_4 are the rotational shapes functions related to the rotational degrees of freedom.

The purpose is to get the root bending moment, we therefore focus on the element of the blade that starts from the root, i.e. with its first node η_1 being the root of the blade. Considering a cantilever beam, the node situated at the root is by definition fixed and its degrees of freedom are

canceled, the vector of finite element displacement corresponding to this element is thus given by

$$\overline{\mathbf{u}} = \begin{bmatrix} 0 \\ 0 \\ u_2 \\ \theta_2 \end{bmatrix} \tag{A.6}$$

Eq. (A.1) requires the second derivative of the finite element displacement, which gives from Eq. (A.2)

$$\frac{\partial^2 \tilde{w}}{\partial x^2} = \left(\frac{\partial \eta}{\partial x}\right)^2 \frac{\partial^2 \overline{\mathbf{N}}}{\partial \eta^2}(\eta)\overline{\mathbf{u}} \tag{A.7}$$

with $\left(\frac{\partial \eta}{\partial x}\right)^2 = \frac{4}{l_e^2}$.

As only the third and fourth components of $\overline{\mathbf{u}}$ are non-null, we only need the third and fourth components of $\frac{\partial^2 \overline{\mathbf{N}}}{\partial \eta^2}(\eta)$, which are given by

$$\begin{aligned} \frac{\partial^2 N_3}{\partial \eta^2}(\eta) &= \frac{1}{4}(-6\eta) \\ \frac{\partial^2 N_4}{\partial \eta^2}(\eta) &= \frac{l_e}{8}(2+6\eta) \end{aligned} \tag{A.8}$$

Those functions are valid for the entire first beam element but we only need the values for the root, which is at position $\eta = -1$, yielding

$$\begin{aligned} \frac{\partial^2 N_3}{\partial \eta^2}(\eta) &= \frac{3}{2} \\ \frac{\partial^2 N_4}{\partial \eta^2}(\eta) &= \frac{-l_e}{2} \end{aligned} \tag{A.9}$$

Knowing the flexural rigidity EI which is a physical property of the blade and replacing those values in (A.1), we get the structural root bending moment from the axial and rotational displacements at node η_2 as

$$M_{root} = -EI\frac{4}{l_e^2}\left(\frac{3}{2}u_2 - \frac{l_e}{2}\theta_2\right) \tag{A.10}$$

Appendix B

Aerodynamic damping

Two types of damping act on the blade to attenuate its movement: aerodynamic damping and structural damping.
The structural damping, expressing itself through matrix C in the dynamic equation of motion (Eq. (B.1)), is constant and the same for AL and AD methods. This kind of damping refers to the inherent ability of a structure to reduce or dampen vibrations and oscillations due to an external force acting upon it. It is thus a property of the structure and its materials and does not depend on the flow conditions and variations.

$$M\ \ddot{x}(t) + C\ \dot{x}(t) + K\ x(t) = F(t) \tag{B.1}$$

The other kind of damping acting on the blade is aerodynamic damping. It is due to the aerodynamic forces and generated from the interaction between the structure and the surrounding air, which act in a manner that opposes or dampens the motion of the structure. It plays an important role in the dynamic behaviour of the blade and is a direct consequence of the motion of the structure and how it impacts the stability of the blade displacement.
Through aerodynamic damping, the amplitude and direction of the structure motion can be directly linked to the forces acting upon itself. If the blade moves away from the rotor plane, it will induce a reduction of the force in this direction tending to keep the blade closer to the rotor. Conversely, if the blade tends to get closer to the rotor plane, the force opposing the movement will increase. This damping mechanism is described here below.

First, one should recall that the lift and drag aerodynamic forces are computed based on the lift and drag coefficient which behaviour in function of the angle of attack is described in Figure B.1. The values of the aerodynamic coefficients in this figure are taken from the data of the airfoil DU21-A17 but the purpose of these graphs is to highlight the shape of the coefficients in terms of the angle of attack, and this stays approximately the same for every other airfoil used.

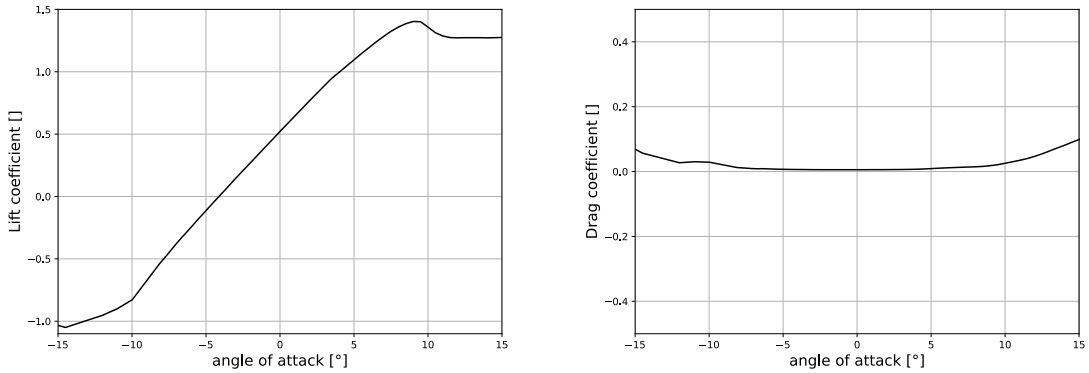


Figure B.1 – *Shape of the aerodynamic coefficients in function of the angle of attack*

For a wide range of angles, including the whole operating range of our simulations, the lift coefficient increases with the angle of attack, whereas the drag coefficient roughly stays constant for the same angle range. We can therefore infer that when the angle of attack increases, the lift coefficient increases, and because of Eq. (B.2), so does the lift force. The drag coefficient staying roughly constant, the angle of attack has little influence on the drag force, as seen in Eq. (B.3).

$$L = 0.5\rho V_{rel}^2 \; c\Delta r \; C_l \tag{B.2}$$

$$D = 0.5\rho V_{rel}^2 \; c\Delta r \; C_d \tag{B.3}$$

Next, as described in Eqs. (B.4) and (B.5) retrieved from the structural solver description, the flow angle ϕ depends on the structure motion, namely its flapwise and edgewise velocities, u_1^{struct} and u_2^{struct}. Regarding the flapwise direction, the structure velocity has a significant impact because its order of magnitude is near that of the wind speed in the same direction and both terms are important in the numerator of Eq. (B.4). However, the edgewise structure velocity and wind speed have little influence compared to the angular speed of the rotor and both terms can be neglected in this approximation.

$$\phi = atan\left(\frac{u_n^*}{\Omega r - u_\theta^*}\right) \tag{B.4}$$

where

$$\begin{aligned} u_n^* &= u_n \cdot cos(\theta_1^{struct}) - u_1^{struct} \\ u_\theta^* &= u_\theta \cdot cos(\theta_2^{struct}) - u_2^{struct} \end{aligned} \tag{B.5}$$

Then, because of the dependence between the angle of attack seen by the airfoil and the flow

angle, expressed in Eq. (B.6), the influence of the structure motion on the angle of attack can be approximated as described in Eq. (B.7).

$$\alpha = \phi - (\beta + \gamma) - \phi^{struct} \tag{B.6}$$

$$\alpha = atan\left(\frac{u_n \cdot cos(\theta_1^{struct}) - u_n^{struc}}{\Omega r}\right) + ... \tag{B.7}$$

Focusing on the effect of the structure velocity, it can be seen that for a structure deforming in the direction of the wind, i.e. $u_n^{struc} > 0$, the angle of attack decreases, and as explained hereabove, this is equivalent to a decrease in lift force, bringing the blade closer to the rotor plane. Conversely, when the blade moves in the direction opposite of that of the wind, the lift force increases, pushing the blade away from the plane of the rotor. A damping can thus be observed in the flapwise direction when looking at the aerodynamics.
However, regarding the edgewise direction, the structure motion influence on the angle of attack is negligible and so is the aerodynamic damping in this direction.

Bibliography

[1] Alex. *Global Wind Report 2023*. Global Wind Energy Council. Feb. 14, 2023. URL: https://gwec.net/globalwindreport2023/ (visited on 12/29/2023).

[2] Ramon Codina et al. "Variational Multiscale Methods in Computational Fluid Dynamics". In: *Encyclopedia of Computational Mechanics Second Edition*. Ed. by Erwin Stein, René Borst, and Thomas J R Hughes. 1st ed. Wiley, Dec. 15, 2017, pp. 1–28. ISBN: 978-1-119-00379-3 978-1-119-17681-7. DOI: 10.1002/9781119176817.ecm2117. URL: https://onlinelibrary.wiley.com/doi/10.1002/9781119176817.ecm2117 (visited on 01/05/2024).

[3] *Direct and Large-Eddy Simulation of Turbulent Wall-Bounded Flows : Further Development of a Parallel Solver, Improvement of Multiscale Subgrid Models and Investigation of Vortex Pairs in Ground Effect — DIAL.Pr - BOREAL*. URL: https://dial.uclouvain.be/pr/boreal/en/object/boreal%3A24942/datastreams (visited on 12/19/2023).

[4] Gianmarco Ducci et al. "Stability and Sensitivity Analysis of Bird Flapping Flight". In: *Journal of Nonlinear Science* 31 (Apr. 1, 2021), p. 47. DOI: 10.1007/s00332-021-09698-1.

[5] Fatma G. Eroglu, Songul Kaya, and Leo G. Rebholz. "A Modular Regularized Variational Multiscale Proper Orthogonal Decomposition for Incompressible Flows". In: *Computer Methods in Applied Mechanics and Engineering* 325 (Oct. 2017), pp. 350–368. ISSN: 00457825. DOI: 10.1016/j.cma.2017.07.017. arXiv: 1705.05339 [math]. URL: http://arxiv.org/abs/1705.05339 (visited on 01/05/2024).

[6] Laurent Georges. "Development and validation of a LES methodology for complex wall-bounded flows : application to high-order structured and industrial unstructured solvers /". UCL - Université Catholique de Louvain, 2007. URL: https://dial.uclouvain.be/pr/boreal/object/boreal:5209 (visited on 12/31/2023).

[7] *Global Electricity Review 2023*. Ember. URL: https://ember-climate.org/insights/research/global-electricity-review-2023/ (visited on 12/29/2023).

[8] V. M. M. G. Costa Gomes, J. M. L. M. Palma, and A. Silva Lopes. "Improving Actuator Disk Wake Model". In: *Journal of Physics: Conference Series* 524.1 (June 2014), p. 012170. ISSN: 1742-6596. DOI: 10.1088/1742-6596/524/1/012170. URL: https://dx.doi.org/10.1088/1742-6596/524/1/012170 (visited on 01/06/2024).

[9] Pankaj Jha et al. "Guidelines for Volume Force Distributions Within Actuator Line Modeling of Wind Turbines on Large-Eddy Simulation-Type Grids". In: *Journal of Solar Energy Engineering* 136 (Aug. 1, 2014), p. 031003. DOI: 10.1115/1.4026252.

[10] J. Jonkman et al. *Definition of a 5-MW Reference Wind Turbine for Offshore System Development*. NREL/TP-500-38060, 947422. Feb. 1, 2009, NREL/TP-500–38060, 947422. DOI: 10.2172/947422. URL: http://www.osti.gov/servlets/purl/947422-nhrlni/ (visited on 12/16/2023).

[11] Moon J. Lee, Byung Do Oh, and Young Bae Kim. "Canonical Fractional-Step Methods and Consistent Boundary Conditions for the Incompressible Navier–Stokes Equations". In: *Journal of Computational Physics* 168.1 (Mar. 20, 2001), pp. 73–100. ISSN: 0021-9991. DOI: 10.1006/jcph.2000.6682. URL: https://www.sciencedirect.com/science/article/pii/S0021999100966828 (visited on 12/19/2023).

[12] Kurt Lust. "Improved Numerical Floquet Multipliers". In: *International Journal of Bifurcation and Chaos* 11 (Nov. 26, 2000). DOI: 10.1142/S0218127401003486.

[13] E. Machefaux, G. C. Larsen, and J. P. Murcia Leon. "Engineering Models for Merging Wakes in Wind Farm Optimization Applications". In: *Journal of Physics: Conference Series* 625.1 (June 2015), p. 012037. ISSN: 1742-6596. DOI: 10.1088/1742-6596/625/1/012037. URL: https://dx.doi.org/10.1088/1742-6596/625/1/012037 (visited on 01/06/2024).

[14] Jakob Mann. "Wind Field Simulation". In: *Probabilistic Engineering Mechanics* 13.4 (Oct. 1, 1998), pp. 269–282. ISSN: 0266-8920. DOI: 10.1016/S0266-8920(97)00036-2. URL: https://www.sciencedirect.com/science/article/pii/S0266892097000362 (visited on 12/31/2023).

[15] Robert Flemming Mikkelsen. *Actuator Disc Methods Applied to Wind Turbines*. MEK-FM-PHD. Kgs. Lyngby, Denmark: Technical University of Denmark, Jan. 2004. ISBN: 978-87-7475-296-7.

[16] Maud Moens. "Large eddy simulation of wind farm flows : improved Actuator Disk model and investigations of wake phenomena". UCL - Université Catholique de Louvain, 2018. URL: https://dial.uclouvain.be/pr/boreal/object/boreal:207779 (visited on 12/16/2023).

[17] J. J Monaghan. "Extrapolating B Splines for Interpolation". In: *Journal of Computational Physics* 60.2 (Sept. 15, 1985), pp. 253–262. ISSN: 0021-9991. DOI: 10.1016/0021-9991(85)90006-3. URL: https://www.sciencedirect.com/science/article/pii/0021999185900063 (visited on 01/05/2024).

[18] *Multiflap*. Vortex Dynamics and Turbulence Laboratory, Dec. 20, 2023. URL: https://github.com/vortexlab-uclouvain/multiflap (visited on 01/03/2024).

[19] Jörn Nathan et al. "Analysis of the Sweeped Actuator Line Method". In: *E3S Web of Conferences* 5.NREL/JA-5000-68400 (Oct. 16, 2015). ISSN: 2267-1242. DOI: 10.1051/e3sconf/20150501001. URL: https://www.osti.gov/biblio/1352992 (visited on 01/06/2024).

[20] Maria Augusta Neto et al. "Finite Element Method for Beams". In: *Engineering Computation of Structures: The Finite Element Method*. Ed. by Maria Augusta Neto et al. Cham: Springer International Publishing, 2015, pp. 115–156. ISBN: 978-3-319-17710-6. DOI: 10.1007/978-3-319-17710-6_4. URL: https://doi.org/10.1007/978-3-319-17710-6_4 (visited on 12/15/2023).

[21] Maria Augusta Neto et al. "Introduction to Finite Element Method". In: *Engineering Computation of Structures: The Finite Element Method*. Ed. by Maria Augusta Neto et al. Cham: Springer International Publishing, 2015, pp. 43–79. ISBN: 978-3-319-17710-6. DOI: 10.1007/978-3-319-17710-6_2. URL: https://doi.org/10.1007/978-3-319-17710-6_2 (visited on 12/15/2023).

[22] Eric Simley et al. "Results from a Wake-Steering Experiment at a Commercial Wind Plant: Investigating the Wind Speed Dependence of Wake-Steering Performance". In: *Wind Energy Science* 6.6 (Nov. 12, 2021), pp. 1427–1453. ISSN: 2366-7443. DOI: 10.5194/wes-6-1427-2021. URL: https://wes.copernicus.org/articles/6/1427/2021/ (visited on 12/30/2023).

[23] J. Smagorinsky. "GENERAL CIRCULATION EXPERIMENTS WITH THE PRIMITIVE EQUATIONS: I. THE BASIC EXPERIMENT". In: *Monthly Weather Review* 91.3 (Mar. 1, 1963), pp. 99–164. ISSN: 1520-0493, 0027-0644. DOI: 10.1175/1520-0493(1963)091<0099:GCEWTP>2.3.CO;2. URL: https://journals.ametsoc.org/view/journals/mwre/91/3/1520-0493_1963_091_0099_gcewtp_2_3_co_2.xml (visited on 12/19/2023).

[24] Jens Norkær Sorensen and Wen Zhong Shen. "Numerical Modeling of Wind Turbine Wakes". In: *Journal of Fluids Engineering* 124.2 (May 28, 2002), pp. 393–399. ISSN: 0098-2202. DOI: 10.1115/1.1471361. URL: https://doi.org/10.1115/1.1471361 (visited on 12/15/2023).

[25] Richard Johannes Antonius Maria Stevens, Dennice F. Gayme, and Charles Meneveau. "Effect of Turbine Alignment on the Average Power Output of Wind-Farms". In: 2013 International Conference on Aerodynamics of Offshore Wind Energy Systems and Wakes, ICOWES 2013. June 17, 2013. URL: https://research.utwente.nl/en/publications/effect-of-turbine-alignment-on-the-average-power-output-of-wind-f (visited on 12/30/2023).

[26] François Trigaux. "Actuator Lines Modeling and Investigation of Tilted Wind Turbines for Improved Farm Efficiency". 2019.

[27] François Trigaux, Grégoire Winckelmans, and Philippe Chatelain. "A flexible actuator line method for aeroelastic simulations of wind turbines in atmospheric boundary layer". In: *Journal of Physics: Conference Series*. The Science of Making Torque From Wind (TORQUE 2022). Vol. 2265. 2022, p. 022050. URL: https://dial.uclouvain.be/pr/boreal/object/boreal:267263 (visited on 01/05/2024).

[28] Oleg Vasilyev. "High Order Finite Difference Schemes on Non-Uniform Meshes with Good Conversation Properties". In: *Journal of Computational Physics* (Jan. 20, 2000). URL: https://www.academia.edu/28856728/High_order_finite_difference_schemes_on_non_uniform_meshes_with_good_conversation_properties (visited on 12/19/2023).

[29] Wei Ye et al. "Chapter 3 - Design with Modeling Techniques". In: *Industrial Ventilation Design Guidebook (Second Edition)*. Ed. by Howard D. Goodfellow and Yi Wang. Academic Press, Jan. 1, 2021, pp. 109–183. ISBN: 978-0-12-816673-4. DOI: 10.1016/B978-0-12-816673-4.00008-0. URL: https://www.sciencedirect.com/science/article/pii/B9780128166734000080 (visited on 01/05/2024).

[30] Yang Zhiyin. "Large-Eddy Simulation: Past, Present and the Future". In: *Chinese Journal of Aeronautics* 28.1 (Feb. 1, 2015), pp. 11–24. ISSN: 1000-9361. DOI: 10.1016/j.cja.2014.12.007. URL: https://www.sciencedirect.com/science/article/pii/S1000936114002064 (visited on 12/19/2023).

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain
Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/epl**