



HAL
open science

Towards vortex-based wind turbine design using GPUs and wake accommodation

Frédéric Blondel, Pierre-Antoine Joulin, Caroline Le Guern

► **To cite this version:**

Frédéric Blondel, Pierre-Antoine Joulin, Caroline Le Guern. Towards vortex-based wind turbine design using GPUs and wake accommodation. *Journal of Physics: Conference Series*, 2024, 2767 (5), pp.052016. 10.1088/1742-6596/2767/5/052016 . hal-04653131

HAL Id: hal-04653131

<https://ifp.hal.science/hal-04653131>

Submitted on 18 Jul 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

PAPER • OPEN ACCESS

Towards vortex-based wind turbine design using GPUs and wake accommodation

To cite this article: Frédéric Blondel *et al* 2024 *J. Phys.: Conf. Ser.* **2767** 052016

View the [article online](#) for updates and enhancements.

You may also like

- [Integral equations and boundary-element solution for static potential in a general piece-wise homogeneous volume conductor](#)
Matti Stenroos
- [Comparative performance of the finite element method and the boundary element fast multipole method for problems mimicking transcranial magnetic stimulation \(TMS\)](#)
Aung Thu Htet, Guilherme B Saturnino, Edward H Burnham et al.
- [Effect of leading-edge erosion on the performance of offshore horizontal axis wind turbine using BEM method](#)
H H Mian, M S Siddiqui, L Yang et al.



PRIMETM
PACIFIC RIM MEETING
ON ELECTROCHEMICAL
AND SOLID STATE SCIENCE

HONOLULU, HI
October 6-11, 2024

Joint International Meeting of
The Electrochemical Society of Japan (ECSJ)
The Korean Electrochemical Society (KECS)
The Electrochemical Society (ECS)

Early Registration Deadline:
September 3, 2024

**MAKE YOUR PLANS
NOW!**

Towards vortex-based wind turbine design using GPUs and wake accommodation

Frédéric Blondel, Pierre-Antoine Joulin, Caroline Le Guern

IFP Energies nouvelles, 1-4 Av. du Bois Préau, 92852 Rueil-Malmaison, France

E-mail: frederic.blondel@ifpen.fr

Abstract. Vortex methods are often cited as a promising alternative to the classical Blade Element Momentum (BEM) theory for wind turbine design, overcoming some of its limitations such as imposing steady-state, uniform, and aligned inflows. BEM assumptions are being questioned, especially for large rotors with very flexible blades or even floating wind turbines. However, vortex methods are not yet widespread due to their computational costs. The following work demonstrates how free vortex wake methods using filament discretizations can become an affordable and viable alternative to BEM. This is achieved by combining filament number reduction as well as graphical processing units (GPUs) computations. The efficiency and accuracy of the proposed approaches are demonstrated against simple test cases, including an elliptical wing and a rigid horizontal axis wind turbine (HAWT). Finally, the approaches are applied to the aero-hydro-servo-elastic simulation of floating horizontal axis wind turbines.

1. Introduction

The offshore wind industry's latest developments, with rotor sizes attaining more than 200m in diameters, present challenges to the current aerodynamic modelling methods. Amongst the latter, the Blade Element Momentum (BEM) method remains the most widespread, despite its limitations. Considering the basic BEM hypotheses, which require a uniform and steady inflow, or constrain the loads' evaluation in a rotor plane, the necessity for higher fidelity tools becomes increasingly apparent. In the context of large floating offshore wind turbines, the aforementioned BEM assumptions might be violated [17], as such wind turbines are highly flexible and their motion induces unsteady inflow. The present study focuses on a lifting-line-based free vortex filament flow solver which accounts for more phenomena including non-uniform inflow, unsteady blade-element behaviour and blade-wake interactions. The latter may not overcome all the issues: interactions between the turbulent ambient wind and the wake are not entirely accounted for, blades are simplified to vorticity-carrying lines, and the flow is assumed to be inviscid. The advantages brought by the vortex methods would make it a well-suited design tool, if it weren't for its prohibitive computational costs.

Approaches aiming at reducing computational costs have been introduced in the literature. Multilevel or fast-multipole algorithms can be used, as shown in [22], [6]. Our approach rather relies on reducing the number of wake elements and GPU acceleration to achieve better performances. [13] introduced in detail a so-called tip-vortex model, allowing a drastic reduction of the number of filaments by removing the shed filaments and merging all the trailing filaments in a pair of tip-vortex filaments. Later, [4] suggested to progressively merge the trailing filament rows and progressively neglecting the shed vortices. Additionally, some authors introduced



Graphical Processing Units (GPU)-accelerated vortex methods [6]. In the present work, we combine both approaches, demonstrating how combining adapted hardware and algorithms allows for a significant reduction in computational time. Gaining several orders of magnitude in computational costs compared to naïve CPU implementations further allow this method to be considered for horizontal axis wind turbine (HAWT) design, to simulate IEC design load cases with simulation costs close to real-time. For that matter, existing methods are improved and their implementation is clarified to benefit other interested developers. In particular, we propose a simple way to account for shed vorticity in a tip-vortex model and propose a trailing-filament merging procedure on top of the existing shed-filament merging procedure. First, the free-wake CASTOR solver and its so-called wake accommodation technique will be described. The accuracy of the solver will then be demonstrated through test cases including an elliptical wing and a HAWT. Finally, the solver’s applicability to wind turbine design on a realistic aero-hydro-servo-elastic wind turbine simulation is demonstrated.

2. CASTOR, a GPU-accelerated vortex solver

2.1. Code overview

CASTOR is a three-dimensional vortex code based on the generalized lifting-line theory which implies the use of airfoil polars to compute aerodynamic coefficients and that for any type of blade shape. The Euler equations (inviscid form of the Navier-Stokes equations) expressed in vorticity form are solved in a Lagrangian perspective [23], in which the vorticity field is discretized using a set of desingularized vortex filaments. The latter are inter-connected using wake nodes, as shown in Figure 1. Lifting-lines are discretized using straight bound filaments whereas the free wake is discretized using straight trailing and shed filaments which carry constant circulation. The filaments are desingularized using, in the present simulations, the Van-Garrel core model [21]. The implementation is flexible enough that several types of geometries can be considered, including horizontal or vertical axis wind turbines [3] or even multi-rotor concepts [1]. Details on the theory and methods behind lifting-line flow solvers can be found in [5, 8].

The implementation follows the work of [18]. The influence of wake filaments on blade and wake points is an n-body problem, that is solved on a single GPU. This excludes the bound-filament circulation convergence process, carried out on CPU as shown in algorithm 1). However, the solver is not entirely ported on GPU: to ease the implementation, only the abovementioned most critical parts have been written in CUDA. Additional validation elements can be found in [2] and [3].

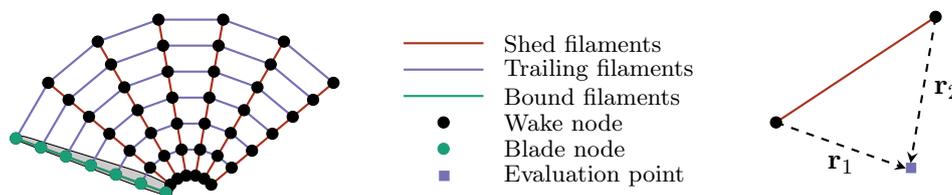


Figure 1. Schematic representation of the wake and notations for induction calculations with \mathbf{r}_1 , \mathbf{r}_2 the vectors pointing from the filament end nodes to the evaluation point.

The main algorithm and its different steps are described in Algorithm 1 and can be summarized as follows. First, a new set of shed and trailing filaments with zero initial circulation are emitted from the blade. Trailing filaments are attached on one side to blade nodes, and to previously advected wake markers on the other side. This step is followed by a copy of all filament’s information on GPU. In a third step, inductions from wake filaments on wake and blade nodes are computed: this is the most costly part of the algorithm and calls for a copy of

wake components and blade nodes as well as the application of Biot-Savart's law, all on GPU.

Algorithm 1: Vortex method algorithm overview

Data: Turbine geometry and kinematic: blades, tower, struts element data

Result: Turbine loads and wake geometry

- 1 Add a new row of shed and trail filaments in the wake;
 - 2 Copy filaments' information on GPU, clear GPU memory;
 - 3 Compute filament inductions on wake and blade nodes;
 - 4 Copy wake and blade nodes information on GPU;
 - 5 Apply Biot-Savart and perform the reduction at every wake node on GPU;
 - 6 Copy the induced velocities back to CPU;
 - 7 **while** $|\tilde{\Gamma}_b^{new} - \tilde{\Gamma}_b^{old}| > \epsilon$ **do**
 - 8 | Compute new induced velocities on the lifting line on CPU using updated $\tilde{\Gamma}_b^{new}$ and newly created Γ_s and Γ_t ;
 - 9 | Compute Γ_b and apply relaxation to obtain $\tilde{\Gamma}_b^{new}$;
 - 10 | Update newly created Γ_s and Γ_t ;
 - 11 **end**
 - 12 Advect all wake nodes and apply any stretching corrections on CPU;
 - 13 Apply wake accommodation techniques on CPU;
-

Once circulations and induced velocities are known, one can evaluate the effect of the corresponding filaments on bound filament centres using Kutta-Jukowski's theorem:

$$\Gamma_b = \frac{1}{2} \|\mathbf{u}_{rel}\| c C_l(\alpha, Re), \quad (1)$$

where Γ_b denotes the bound circulation, $\|\mathbf{u}_{rel}\|$ is the norm of the relative velocity, *i.e.*, the combination of the blade motion, infinite wind, and induced velocities at the blade node, c is the local chord, and $C_l(\alpha)$ is the lift coefficient interpolated from the user-provided airfoil polars, at the attack angle α and Reynolds number Re . The orientation matrices, translation velocities and positions of each bound filament centre are assumed to be known and provided by an external solver. The induced velocity $\mathbf{u}_{induced}$ is computed using the regularized Biot-Savart law [21], that writes, with F the number of considered filaments:

$$\mathbf{u}_{induced} = \sum_{f=1}^F \frac{\Gamma_f}{4\pi} \frac{(\|\mathbf{r}_1\| + \|\mathbf{r}_2\|) (\mathbf{r}_1 \times \mathbf{r}_2)}{\|\mathbf{r}_1\| \|\mathbf{r}_2\| (\|\mathbf{r}_1\| \|\mathbf{r}_2\| + \mathbf{r}_1 \cdot \mathbf{r}_2) + (\epsilon L_f)^2}, \quad (2)$$

where \times denotes the cross product operator and ϵ is the vortex core size, L_f is the f 'st filament length. After the bound's circulation update, the newly created shed and trailing filaments' circulations can in turn be adjusted allowing for an update of induced velocities at bound filament centres. At iteration j , their circulation is calculated as follows:

$$\Gamma_s(i, j) = \Gamma_b(i, j) - \Gamma_b(i, j - 1), \quad (3)$$

$$\Gamma_t(i, j) = \Gamma_b(i, j) - \Gamma_b(i - 1, j), \quad (4)$$

where $i \in [1, N]$ denotes the number of bound filaments and j the number of shed filament rows, respectively. The trailing filaments circulation beyond the blade is set to $\Gamma_t(1, j) = \Gamma_b(1, j)$ and $\Gamma_t(N + 1, j) = -\Gamma_b(N, j)$. This process is repeated until convergence. During this sub-iterative process, a relaxation coefficient is applied when updating the bound circulation to avoid the

divergence of the numerical algorithm. The new relaxed circulation is noted $\tilde{\Gamma}_b^{new}$, while $\tilde{\Gamma}_b^{old}$ is the one from the previous sub-iteration. Γ_b comes from Equation 1.

$$\tilde{\Gamma}_b^{new} = \tilde{\Gamma}_b^{old} + \delta \left(\Gamma_b - \tilde{\Gamma}_b^{old} \right), \quad (5)$$

with the relaxation factor taken as $\delta = 0.05$ in the present simulations. Once the convergence is reached (after a fixed number of iterations in our case), the wake markers are advected using a simple forward Euler scheme (the new positions are evaluated as follows: $\mathbf{x}(t + \Delta t) = \mathbf{x}(t) + \Delta t (\mathbf{u}_{wind} + \mathbf{u}_{induced})$, with \mathbf{x} the node position, Δt the time step and \mathbf{u}_{wind} the wind velocity at location \mathbf{x} and time t . The advection step is performed on CPU.

2.2. Wake accommodation techniques

Considering optimal time steps (5 to 10 degrees of blade rotation per time step [17]), the GPU-based vortex method is efficient enough to consider extensive parametric studies. However, when coupled with a servo-elastic solver, one has to deal with smaller time steps, often enforced by the needs of a controller's numerical model. This implies that the wake will accordingly consist of a larger amount of filaments, making the simulation too costly in practice (the algorithm complexity scales with the squared number of filaments). Various approaches can be used to avoid this issue. First, one may use different time steps for the vortex solver and the servo-elastic code. This approach is introduced in [10]. Another strategy consists in reducing the number of filaments in the wake by concatenating them. It is less efficient since the vortex solver is called frequently but it helps to drastically reduce the computational time. Two approaches to reduce the number of filaments, the tip-vortex model and a wake-coarsening strategy, are considered here, as shown in Figure 2. These have already been documented in the literature. The originality of the present work lies in the inclusion of shed vorticity in the tip-vortex model, together with a clarification of the circulation calculation and filament merging procedure in the wake coarsening approach. In the following, it is assumed that the merged trailing filaments are of similar sizes and orientations, as well as the merged shed filaments.

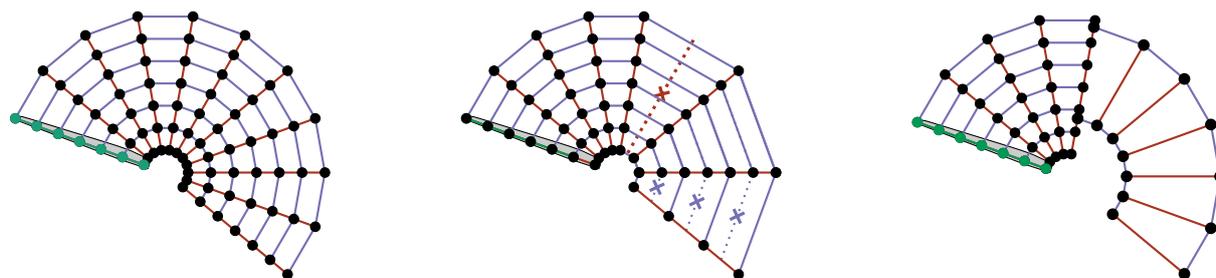


Figure 2. Schematic representation of the full wake model (left), the wake coarsening process with one shed- and one trail-merging steps (middle) and the tip-vortex model, with shed vorticity (right). Bound, shed and trailing filaments are green, red and blue, respectively.

2.2.1. Tip vortex model In this approach, the blade's trailing filaments of given age (defined as the number of iterations since their initial release) are merged, leading to a simplified system with only two remaining trailing filaments, whose radial locations are determined based on the local circulation at the user-defined filament age. In the present work, and in contrast with other approaches (see [17]), we also gather blades shed filaments of given age into a single one, reducing the discontinuities which appear at the creation of the tip vortex. Our implementation regarding the trailed vorticity closely follows the work of [13]. The trailing filaments are classified into two

subgroups, depending on the sign of their circulation. The circulations of each filament from the individual groups are summed and applied to the newly created tip vortices. The radial location of the attachment points of the tip vortices (see Figure 2) is computed as the ponderated average of the filaments radial locations and their circulations. The tip vortex circulation $\Gamma_{t,TV}$, and its radial position r_{TV} is then computed as follows:

$$\Gamma_{t,TV} = \sum_{f=1}^F \Gamma_t^f \quad \text{and} \quad r_{TV} = \frac{\sum_{f=1}^F r_f \Gamma_t^f}{\sum_{f=1}^F \Gamma_t^f}. \quad (6)$$

with f going through all the filaments of same circulation sign.

The circulation of the new shed is considered equal to the average of the variation of the bound circulations during a time step. However, the length of the newly created shed filament belonging to the tip-vortex system is different from the sum of the lengths of the previous filaments. Thus, we propose to estimate the shed vortex circulation as follows:

$$\Gamma_{s,TV} = \frac{1}{L_{s,TV}} \sum_{f=1}^F L_s^f \Gamma_s^f, \quad (7)$$

with L_s^f the length of the f 'st shed filament and $L_{s,TV}$ the distance between the two tip-vortex control points. As the length $L_{s,TV}$ of the new filament is shorter than the cumulative length of the previous ones ($\sum_{f=1}^F l_s^f$), we artificially increase its vorticity to compensate for the artificial contraction.

2.2.2. Wake coarsening The tip vortex model is a rather radical approach, where one loses a lot of accuracy in the flow field. Applying such a model directly in the near wake may lead to erroneous results. A more progressive wake coarsening approach reduces the filaments by merging two subsequent rows of shed (or trailing) filaments. In doing so, we also merge two successive rows of trailing (or shed) filaments. This procedure allows one to reduce the number of filaments while better controlling the accuracy. We propose to outline the procedure involving a shed merging, followed by a trailing merging, as illustrated in Figure 3.

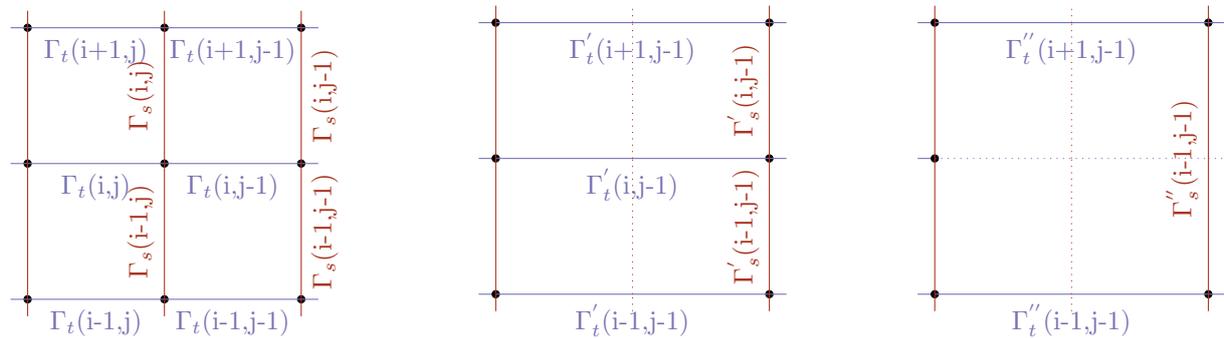


Figure 3. Schematic representation of the wake coarsening procedure, starting from a full mesh (left) and including a shed merging (middle) and a trail merging (right) process.

Let us consider that the filaments are emitted at a fixed rotation speed with a time step twice as small as the expected one. By definition, the circulation of a trailing filament is equal to the difference of bound circulations of the same age. Therefore, the circulation of the new trailing filament must be equal to the corresponding difference in bound circulation, *i.e.*, the circulation

of the youngest merged trailing filament. Concerning the new shed filament, its circulation is equal to the sum of the two distinct previous shed filaments to adhere to the temporal variation budget. The first-order moment of the shed vorticity is preserved, but higher order moments (*e.g.*, the variance of the vorticity field) are not. Then, the circulations of the newly created filaments, after the shed merging procedure, write:

$$\Gamma'_t(i, j - 1) = \Gamma_t(i, j), \quad (8)$$

$$\Gamma'_s(i, j - 1) = \Gamma_s(i, j - 1) + \Gamma_s(i, j). \quad (9)$$

For the trailing merging procedure, a locally linear behaviour of the bound circulation is assumed. Proceeding this way, the post-trailing-merging shed circulations Γ''_s , is linearly interpolated:

$$\Gamma''_s(i - 1, j - 1) = \frac{1}{2} \left(\Gamma'_s(i - 1, j - 1) + \Gamma'_s(i, j - 1) \right). \quad (10)$$

Assuming the trailing filament is positioned at the centre, its circulation, corresponding to the difference in bound circulation at the corresponding age, is uniformly distributed on both sides:

$$\Gamma''_t(i + 1, j - 1) = \Gamma'_t(i + 1, j - 1) + \frac{1}{2} \Gamma'_t(i, j - 1), \quad (11)$$

$$\Gamma''_t(i - 1, j - 1) = \Gamma'_t(i - 1, j - 1) + \frac{1}{2} \Gamma'_t(i, j - 1). \quad (12)$$

3. Assessing the accuracy of the method: rigid benchmarks

In this section, the accuracy of the vortex method and the proposed wake accommodation approaches are analysed. Three test cases are considered, including an elliptical wing, a HAWT aligned with the incoming wind and a yawed HAWT. For each case, comparisons are shown between the full wake approach (“Full wake”), the standard tip-vortex model (labelled “TV no shed wake”), the tip-vortex model including a shed vortex filament (labelled “TV wake”) and finally a coarsened wake using a shed merging (SM) step, followed by a trailing-filaments merging step (TM), another SM step and a final TM step (labelled “ $2 \times SM + 2 \times TM$ ” and schematized in Figure 8). The objective is to demonstrate the applicability of such approaches. Relative errors are given in Table 2.

3.1. First case: elliptical wing

The first test case is an elliptical wing in static and dynamic incidence. The wing has a span of $s = 5m$ and the centre chord is $c = 1m$, leading to an aspect ratio of approximately 6.37. In the static case, the incidence angle is $\alpha_{geom} = 6^\circ$, leading to a theoretical lift of 0.50 following [12], as stated in [18]. The infinite wind is $u_\infty = 1m/s$. In the oscillating case, the incidence follows a cosine law, with a mean angle $\alpha_{geom} = 5^\circ$, a period of $T = 1s$ and a peak-to-peak amplitude of $\Delta\alpha_{geom} = 10^\circ$. The same infinite wind value is used as for the static case. 20 seconds of physical time are simulated, using a time step of $\Delta t = 0.05s$. The tip vortex models are applied once filaments reach the age of 20 iterations. The shed and trailing filaments merging procedures are applied when the filaments reach the following ages (A): $A_{SM_1} = 20$, $A_{TM_1} = 40$, $A_{SM_2} = 60$ and $A_{TM_2} = 100$. Note that after each merging procedure, a global renumbering of the wake filaments is applied. As shown in Figure 4 (top/bottom left), the filaments’ merging approach leads to very similar results to those obtained using the full-wake approach as well as the theoretical value. The impact of the tip-vortex approaches is visible. At the blade centre, the lift coefficients are underestimated. At the blade tip, the lift distribution is less sharp. Looking at Figure 4 (top right), the traditional tip vortex approach fails at predicting

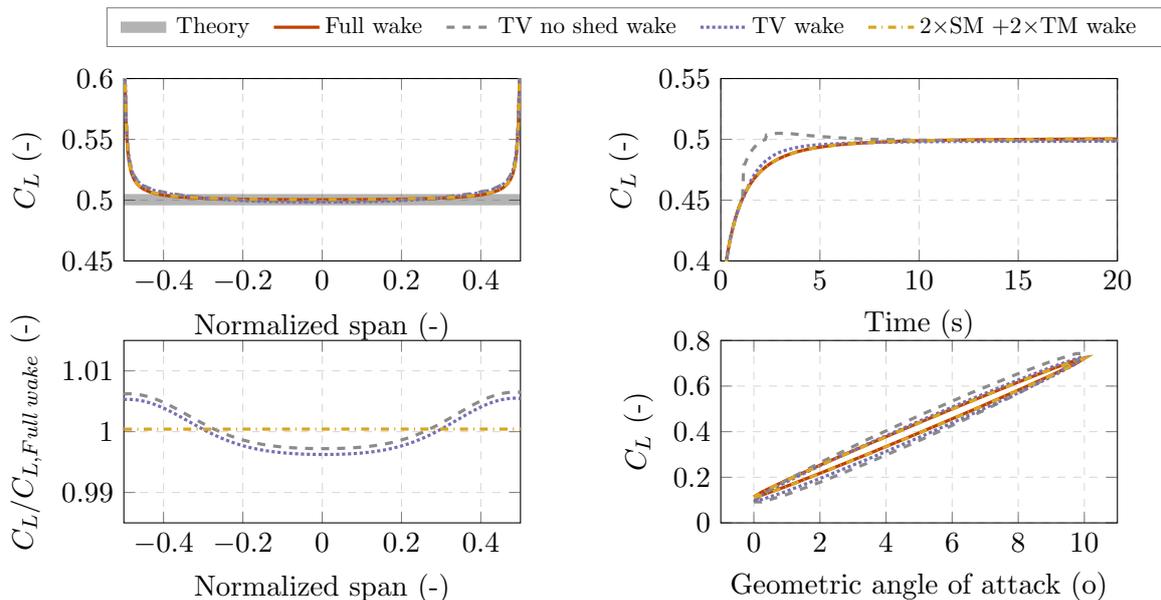


Figure 4. Static wing: lift coefficient (C_L) distribution (top left), time evolution of the mid-span lift (top right), ratios of lift to full-wake case lift distribution (bottom left). Oscillating wing: time evolution of the mid-span lift coefficient against the attack angle (bottom right).

the correct dynamic of the system. The induced velocities at the wing centre diminish with time, leading to a higher lift coefficient: this is due to the development of the starting vortex. While the steady-state solution seems accurate, the dynamic behavior deviates from the full-wake results, as the absence of shed vorticity in the vortex sheet violates Kelvin’s theorem. Introducing the corrected tip-vortex allows for a better fit to the full-wake solution. Some deviations remain and are attributed to the change in the blade lift introduced by the coarsened trailing filament distribution. Results based on the oscillating wing confirm these observations. Again, the traditional tip-vortex approach fails to follow the full-wake trends, here (Figure 4, right) highlighted by the evolution of the lift coefficient with the geometrical attack angle. The corrected tip-vortex model behaves, again, much better due to the partially compensated loss of shed vorticity in the system.

3.2. Second case: the rigid NewMexico rotor

The second considered case is the NewMexico rotor in axial and yawed conditions. The lower wind speed (i.e., strongest wake effect) of the test cases detailed in [19] are presented here. For the sake of conciseness, this widely described configuration will not be detailed here (see [19]). For the two considered cases, the rotor operates at a tip-speed ratio of approximately 10. Fifteen complete rotor rotations are considered. The tower presence is neglected, and no dynamic stall model nor stall-delay correction is applied. The blades are discretised using 35 cosine-distributed filaments, and the Van-Garrel regularisation is used. A time step corresponding to a blade rotation of 10° per step is chosen. For the yawed rotor case, a yaw angle of 30° is applied. The tip-vortex models are applied at iteration 36 in the wake. The shed and trailing merging procedures are applied when the filaments reach the following ages (A): $A_{SM_1} = 36$, $A_{TM_1} = 72$, $A_{SM_2} = 108$ and $A_{TM_2} = 144$. The spanwise distribution of the normal force per unit filament length) in the axial case is shown in Figure 5 (left). The tip-vortex models tend to underestimate the loads along the blade, except at the very root and tip. The wake coarsening

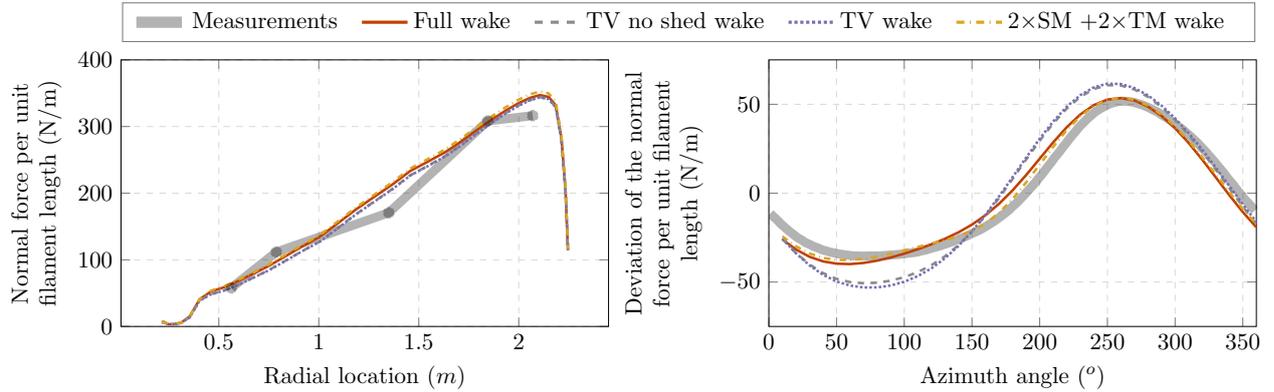


Figure 5. Normal force along the blade for the aligned turbine case (left) and azimuthal evolution of its deviation around the mean at normalized span 0.92 (right) for the yawed case.

approach predicts similar loading along the blade, except at the very tip, where the loads are slightly over-estimated. In 5 (right), are shown the deviations to the mean of the normal force near the tip of the blade (normalized span of 0.92) for the yawed rotor case. While the wake coarsening method results follow that of the full-wake approach and the experiments rather well, both tip-vortex approaches deviate from the full-wake results. The shed vorticity correction does not help in this unsteady test case, suggesting that the load evolution is governed by the trailing vorticity.

4. Application to aero-hydro-servo-elastic simulation of a floating wind turbine

The DeepLines WindTM [9] solver is used throughout this work. It allows for multi-physics modelling, gathering aero-hydro-servo-elastic fields. Here, the wind turbine's structure is operated by the so-called DISCON controller [7]. The structural aspect is modelled using a finite element method, where the tower and blades are modelled through 3D non-linear beam elements with large transformations for the latter. The numerical integration is performed using an implicit Newmark scheme. The aerodynamic solver is called before the first Newmark step and returns the aerodynamic loads at every lifting-line element centre. See [10] for more details. The present simulation corresponds to the load case 3.2 as described in [14]. In brief, it consists of the generic NREL 5MW wind turbine as described in [7] mounted on a semi-submersible floating system [15]. A turbulent wind with a mean velocity of $11.4m/s$ at hub height is used. Irregular wave conditions are imposed, with a significant wave height of $H_s = 6m$ and a peak-spectral wave period of $T_p = 10s$. The average rotational velocity of the wind turbine is close to $10rpm$ and a time step of $\Delta t = 0.05s$ is used. Three vortex cases are considered. In each, 15 wake rotations are kept in the simulation. For the full wake case, the maximal wake age is about 1800. For the tip-vortex model, the transition is set after one complete rotation of the wake. The shed and trailing merging procedures are applied at the following ages (A): $A_{SM_1} = 60$, $A_{SM_2} = 150$, $A_{TM_1} = 105$, $A_{TM_2} = 140$ corresponding to roughly 0.5, 1.5, 1 and 2 wake rotations, respectively. $3600sec$ of physical time are simulated, and only a subset is shown in Figure 6 (left). The thrust evolution predicted by the different methods is similar. The power spectral density of the tower-bottom fore-aft bending moment (Figure 6, right) shows at first that the system frequencies are well predicted by all methods. The energy content of the simulation based on the BEM model is higher than with the vortex method, especially at high frequencies. This may imply an over-prediction of the fatigue loads. The computational times and the ratio between simulation time and real-time are gathered in Table 1. While real-time

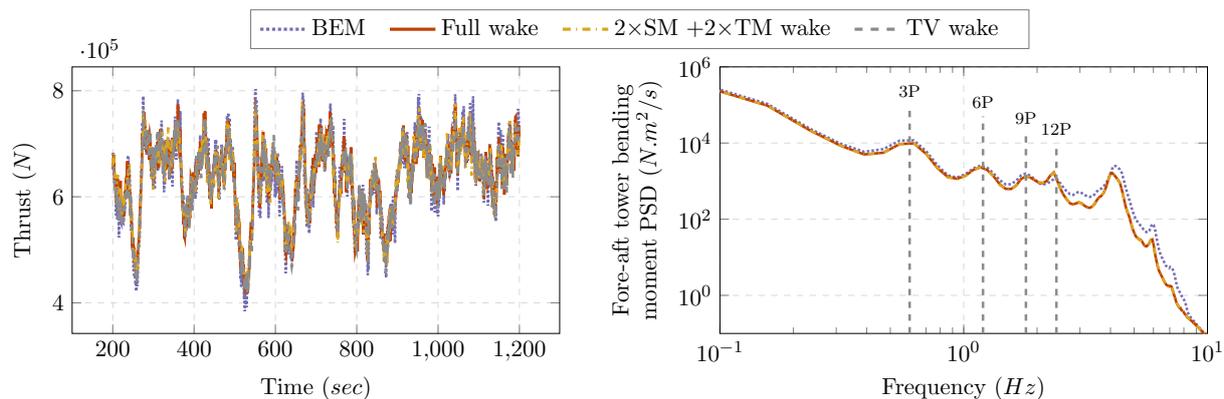


Figure 6. Evolution of the aerodynamic thrust (left) and power spectral density of the tower base fore-aft bending moment (right).

simulations are not yet attained using the vortex method, the gap diminishes when applying wake accommodation techniques. Interestingly, the tip vortex model appears to be less than three times slower than the BEM, thus showing good potential for industrial application. The full wake simulation remains out of reach for industrial purposes.

	BEM	Full wake	2SM+2TM wake	TV wake
Computational time (sec)	2029	73319	7551	5747
Ratio to real-time (-)	0.56	20.36	2.09	1.59

Table 1. Computational times and ratio to real-time for the different models

5. Code performance

The solver's performance evaluation is based on the NewMexico rigid rotor test case. The computational time per iteration is given as a function of the number of filaments. For this test case, more than 160 complete rotations of the wind turbine have been considered, leading to a maximum number of filaments of approximately 1.2 million. The simulation has been repeated twenty times, and the average computational time per iteration is shown. To focus on the vortex code, only the time spent in the aerodynamic routines is measured here: computations of node positions, velocities, accelerations, orientations, blade deformations or others are excluded. A single A100 GPU of the Topaze super-computer nodes is used. Results are shown in Figure 7. As already observed in [6] steps are present and the evolution of the computational time is not smooth. These steps appear regularly.

6. Conclusion

We introduced a GPU-based vortex solver using wake accommodation techniques. The presented approaches show that vortex-based aero-elastic wind turbine simulations can be efficient enough to run in a reasonable amount of time. Benchmarks allowed us to assess the accuracy of the model and compare simplified wake models to the baseline solution. Demonstration aero-hydro-servo-elastic simulations using the DeepLines-WindTM solver coupled with a BEM and the vortex model have been performed, showing that the wake-accommodation approaches combined with the GPU-based computations allow a drastic reduction of the simulations' computational cost. The fastest vortex simulation remains three times slower than a BEM-based approach. A model

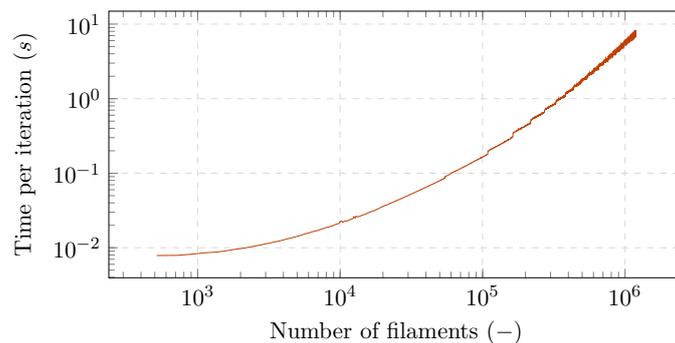


Figure 7. Computational time as a function of the number of filaments, twenty runs averaged.

to account for shed vorticity in the tip-vortex approach has been proposed. It showed an improved behaviour for the elliptical wing case. The wake coarsening approach allows a strong reduction in computational cost while maintaining high accuracy. It could be combined with the tip vortex approach. Guidelines regarding the optimal combination of shed/trailing-merging and tip-vortex concatenation procedures will be derived, allowing a reduction of the number of filaments while controlling the accuracy, complementing the existing work ([6], [17]).

Acknowledgments

The data used have been supplied by the consortium which carried out the EU FP5 project Mexico: 'Model rotor EXperiments In COntrolled conditions'. The consortium received additional support to perform the New Mexico measurements from the EU projects ESWIRP and INNWIND.EU.

References

- [1] Blondel F, Dyachuk E, Hummel N, Le Cunff C, Rahm M, 2017, *Wind Europe 2017*.
- [2] Blondel F, Ferrer G, Cathelain M, Teixeira D, 2017, *Congrès Français de Mécanique*.
- [3] Blondel F, Galinos C, Paulsen U, Bozonnet P, Cathelain M, Ferrer G, Madsen HA, Pirrung G Silvert F, 2018, *TORQUE 2018 conference*.
- [4] Boorsma K and Greco L and Bedon G, 2018, *TORQUE Conference*.
- [5] Branlard E, 2017, *Springer*.
- [6] Corniglion R, 2022, *Ph.D. thesis report, EDF R&D*.
- [7] Jonkman J, Butterfield S and Musial W and Scott G, 2009, *NREL Report NREL/TP-500-38060*.
- [8] Katz J, Plotkin A, 2001, *Cambridge University Press*.
- [9] Le Cunff C, Heurtier J.M. and Piriou L. and Berhault C. and Perdrizet T. and Teixeira D. and Ferrer G. and Gilloteaux J.C., 2013, *OMAE Conference*.
- [10] Le Guern C, Pfister J.L., Guy N, Bergmann M, 2024, *TORQUE Conference (submitted)*.
- [11] Dyachuk E, Rossander M, Goude A, Bernhoff H 2015 *Energies* **8**.
- [12] McCormick BW, 1979, John Wiley and Sons, Inc..
- [13] Montgomery B, 2004, *FOI Swedish Defense Research Agency*.
- [14] Robertson A et al., 2014, *OMAE Conference*.
- [15] Robertson A et al., 2014, *NREL Report NREL/TP-5000-60601*.
- [16] Rossander M, Dyachuk E, Apelfröjd S, Trolin K, Goude A, Bernhoff H, Eriksson S, 2015, *Energies* **8**.
- [17] Shaler K and Anderson B and Martínez-Tossas L.A and Branlard E and Johnson N, 2023, *Wind Energy. Sci.*
- [18] Sebastian T, Lackner MA, 2012, *Renewable Energy* **46**.
- [19] Scheppers G et al., 2018, *ECN*.
- [20] Türkal M and Novikov Y and Üşenmez S and Sezer-Uzol N and Uzol O, 2014, *TORQUE Conference*.
- [21] van Garrel A., 2003, *ECN report ECN-C-03-079*.
- [22] van Garrel A., Venner C.H., Hoeijmakers H.W., 2017, *35th Wind Energy Symposium*.
- [23] Soler A, 1990, *Vortex filament method*.

Appendix A: schematics of the wake geometries

The test case introduced in this work rely on different wake models, including tip-vortex approaches and so-called shed- and trailing-filaments merging approaches. Schematics of the different wakes are shown here.

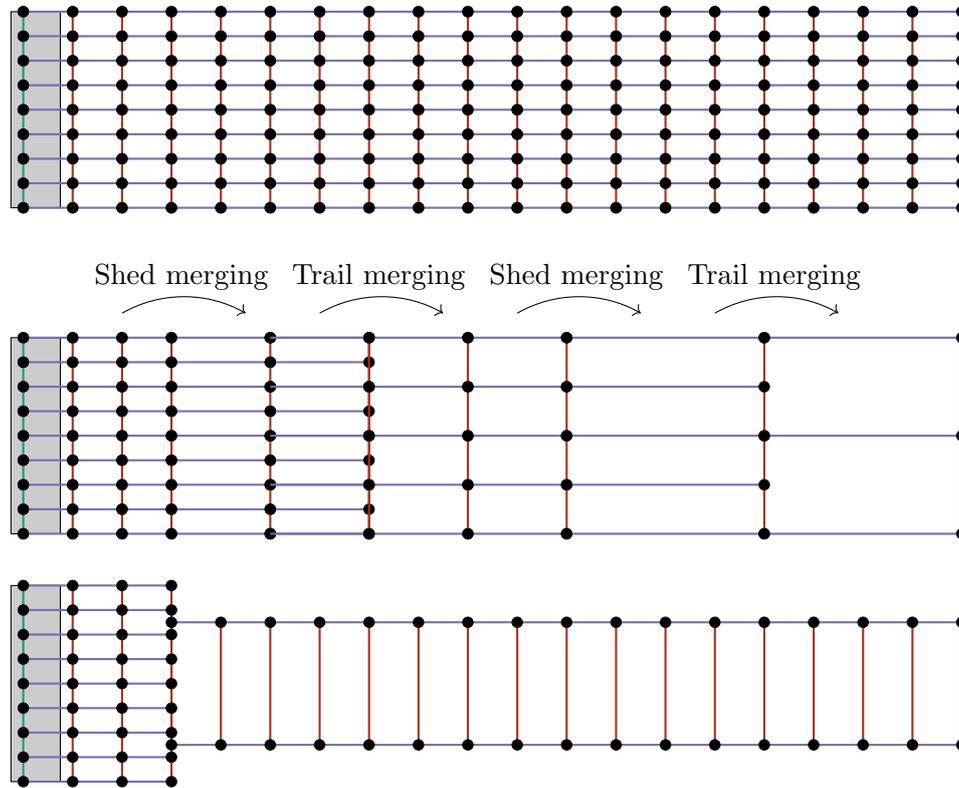


Figure 8. Schematic representation of the wake models used in the simulations, including the full wake (top), the coarsened wake (middle) and the tip-vortex wake (bottom).

Appendix B: test cases relative errors

To complement the analysis performed throughout this work, relative errors are provided here for the different test cases. The norm of the difference between the full-wake simulations and the coarsened wake simulations divided by the norm of the full-wake simulation outputs are given.

	TV wake no shed	TV wake with shed	2SM+2TM wake
Elliptical wing:	4.79×10^{-3}	4.21×10^{-3}	4.18×10^{-4}
Axial NewMexico:	2.23×10^{-2}	2.37×10^{-2}	1.60×10^{-2}
Yawed NewMexico:	2.42×10^{-1}	2.80×10^{-1}	6.95×10^{-2}
DeepLines Wind TM simulation:	N.A.	3.61×10^{-2}	3.51×10^{-2}

Table 2. Errors relative to the full wake case for the static lift distribution of the elliptical wing, spanwise and azimuthal distribution of the normal force per unit filament length of the axial and yawed New-Mexico test cases, respectively, and for the thrust evolution of the DeepLines-WindTM aero-hydro-servo-elastic test case.