

Technical University of Denmark



Validation of an Actuator Disc Model

Réthoré, Pierre-Elouan; Sørensen, Niels N.; Zahle, Frederik

Published in:
EWEC 2010 Proceedings online

Publication date:
2010

Document Version
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

Citation (APA):
Réthoré, P-E. M., Sørensen, N. N., & Zahle, F. (2010). Validation of an Actuator Disc Model. In EWEC 2010 Proceedings online European Wind Energy Association (EWEA).

DTU Library

Technical Information Center of Denmark

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Validation of an Actuator Disc Model

P.-E. Réthoré N. N. Sørensen F. Zahle
pire@risoe.dtu.dk nsqr@risoe.dtu.dk frza@risoe.dtu.dk
Wind Energy Division · Risø DTU · Denmark

Abstract

Wind turbine wake can be studied in CFD with the use of permeable body forces (e.g. actuator disc, line, surface). This paper presents a general flexible method to redistribute wind turbine blade forces as permeable body forces in a computational domain. The method can take any kind of shape discretization, determine the intersectional elements with the computational grid and use the size of these elements to redistribute proportionally the forces.

The special case of the actuator disc is successfully validated with an analytical solution for heavily loaded turbines and with a full rotor computation in CFD.

Keywords: Conway's Actuator Disc, Bessel Laplace Integrals, CFD.

1 Introduction

As computational power becomes more affordable, Computational Fluid Dynamics (CFD) methods become attractive solutions for modelling wind turbine and wind farm wakes. The most cost effective CFD wind turbine wake models are based on the concept of applying the wind turbine blades forces in the computational domain through permeable body forces. There are three main categories based on this concept, the actuator disc, line and surface (for a literature review, see [8, 15, 18, 22, 23]). All these concepts follow the same two-step approach. Firstly, the blade forces are estimated from the local flow information using different types of methods. Secondly, they are then redistributed in the computational domain. This article focusses on the second step.

As there is practically no physics involved in the second step, the vast majority of the papers describing the different actuator methods does not mention this issue. However, the speed and

accuracy of the solution can be deeply affected by the approach used to redistribute the forces. A poor approach can slow down significantly the code at each iteration, and can require a finer grid to obtain grid independence.

Moreover, because of the general lack of literature on the subject, there is also a need for validation methods for this step. Some papers validate this step indirectly by comparing the wind turbine wake with measurements (e.g. [1]). By doing so, both steps (force estimation, and force redistribution) are used together with a turbulence model to generate the flow field. When combining several complex methods together there is always the risk that multiple errors compensate each other. It is therefore considered safer to validate each of the models independently.

One way of doing this is to compare with other types of wind turbine wake models that can provide the blade loadings, such as analytical solutions or full rotor computations. Such comparison of actuator methods with analytical solution can be found in the literature (e.g. comparison with the heavily loaded actuator disc mode of Wu [13] and Conway [19]). It seems, however, that there is not any paper that discusses the comparison of full rotor computation with actuator methods.

This paper describes a general method (Sec.2) to redistribute body forces in a computational domain, which is implemented in EllipSys [12, 20]. The method has a low initialization time and need a relatively small amount of domain cells to redistribute correctly the forces. The model is based on three levels of discretization. The actual shape, defined by a grid, the computational domain

grid, and the intersectional grid between the shape and the computational domain grids. This independent description of the shape and the domain gives a wide flexibility of possible shapes and motions (e.g. disc, rotating lines, moving kites, trees). An indirect application of this method is to define immerse boundary conditions such a turbulence generator grids [9].

The special case of the actuator disc model is validated in Sec.3 with respect to the analytical solution for heavily loaded actuator discs of Conway [6], and a full rotor computation. Both comparison show excellent agreement between the different models presented in Sec.2. This shows that using actuator discs, with the proper loading, gives an accurate description of the wind turbine wake. Moreover, it is found that a coarse discretization (as low as 10 cells per rotor diameter) gives a relatively acceptable wake development in comparison with finer meshed full rotor computation.

2 Actuator Shape Model

2.1 General Idea

The actuator shape model is based on three different types of discretization.

1. The shape is defined by a grid, where each cells is somehow associated to a force vector.
2. The computational grid is where the shape forces are redistributed and solved with the Navier-Stokes equations.
3. The intersectional grid between the shape grid and the domain grid is used as a weight to redistribute the shape forces into the domain grid.

The independent shape grid and the domain grid gives a flexibility of the shape motion in the domain and a simple domain grid definition.

The shape position is defined by three vectors (one for the relative position with the domain, and two for the orientation of the shape). These vectors can be changed dynamically during the simulation.

The discretization of the shape can be done in 3 different ways: 1D cells (segments), 2D cells

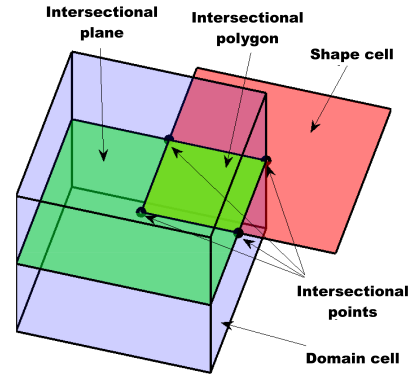


Figure 1: Illustration of the different elements considered in the intersectional polygon search algorithm (with 2D shape cells).

(quadrilaterals), 3D cells (hexahedrons). Each of the points are defined in 3D. The relative position vector mentioned previously is relating the domain grid origin to the shape grid origin.

2.2 Initialisation Phase

The initialisation phase, during which the intersectional grid is redefined, is carried out each time that the shape definition is changed. The intersectional grid is built on a two-step process.

Firstly, each of the shape grid points are associated to the domain cell they are located into. The search routine is based on a spreading recursive dichotomy methodology. A recursive dichotomy is first used to dive into the domain grid. When one shape point is found in a domain cell, the neighboring shape points are first tested in the same domain cell, then on the neighboring domain cells until all the points are found. This approach reduces dramatically the number of iteration needed to associate all the shape points with domain cells, and has the advantage to be reusable. When the shape definition has changed, the previous shape point - domain cell association is used as a first guess to the update.

Secondly, based on the “shape point - domain cell” association, the algorithm is looking for all the “shape cell segments - domain cell faces” and “shape cell faces - domain cell segments” intersectional points (see Fig.1). These points are defining the intersectional grid. The intersectional points are then rearranged into intersectional cells.

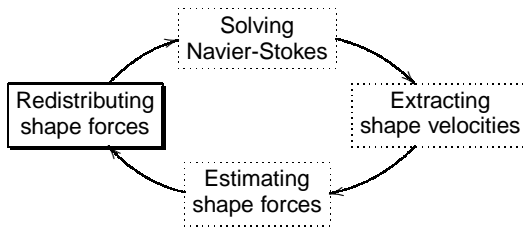


Figure 2: Active phase scheme.

2.3 Active Phase

The active phase is executed at every solving iteration. The iterative scheme, illustrated in Fig.2, is basically solving the Navier-Stokes equation, extracting the shape velocities, from which the shape forces are estimated and finally redistributed in the computational grid. The step of interest, in this paper, is the redistribution of the shape forces.

The redistribution of the forces is done by using the intersectional grid. According to the intersectional cell type, its relative size (i.e. length for 1D segments, area for 2D polygons, or volume for 3D polyhedrons) compared to the associated shape cell is used as a proportional weight to scale the shape cell force vector. All the intersectional cell force vectors associated to a domain cell are then added together to obtain the equivalent domain cell force vector (see Fig.3).

3 Validation

3.1 Methods

EllipSys Flow Solver

EllipSys is an in-house incompressible finite volume Reynolds averaged Navier-Stokes flow solver developed at Risø-DTU [20] and DTU-MEK [12]. The flow variables are collocated in the mesh to facilitate complex mesh geometries. The SIMPLE algorithm [14] is used to solve the Navier-Stokes equations. The convective terms are discretized using the QUICK scheme [10]. The pressure equation is solved using a modified Rhie-Chow algorithm to avoid odd/even pressure decoupling with body forces [15–17], and is accelerated by a multigrid technique [20]. The system is parallelized in a multiblock structure, where the blocks can be solved on a different processor. The communication of the block

ghost cells between each processor is done through the MPI libraries. The turbulence model used in this paper is Menter’s $k-\omega$ SST [11].

Actuator Disc Model of EllipSys

The actuator disc model is based on the actuator shape model described in the previous section. The shape grid is a coplanar polar disc of 11 radial and 62 angular elements. The force vector associated to each of the disc cells are found by the two other models describe in the following sections respectively.

Conway’s Actuator Disc

In a series of articles [4–7, 19], Conway has described an exact actuator disc model for a heavily loaded propeller or a wind turbine. The model is based on the idea of axially discretizing the slipstream azimuthal vorticity of a wind turbine into vortex disks. As there are ways to express the flow behaviour induced by vortex disks, if the vortex distribution of the slipstream is known, it is possible to derive an exact formulation of the induced flow features.

This method is relatively complex to implement as it requires solving Bessel-Laplace integrals using recursive rules in order to get an expression in terms of complete elliptic integrals and other associated functions. This section presents a concise explanation on how to setup this solution and gives some analytical expressions needed to solve the special case of a parabolic wake profile.

Conway has derived in [6] a solution for the special case of a parabolic wake profile including the slipstream expansion, where the vorticity distribution is taken as

$$\omega_\phi = ar, \quad (1)$$

where ω is the vorticity, ϕ is the tangential direction, a is a free parameter and r is the radial coordinate. In this special case, the velocity profile in the ultimate wake is known to be

$$U_z(r, \infty) = U_\infty + a(R_\infty^2 - r^2)/2, \quad (2)$$

where R_∞ is the ultimate width of the slipstream. In this case, the stream function Ψ and the axial and radial velocities V_z and V_r can be found by solving

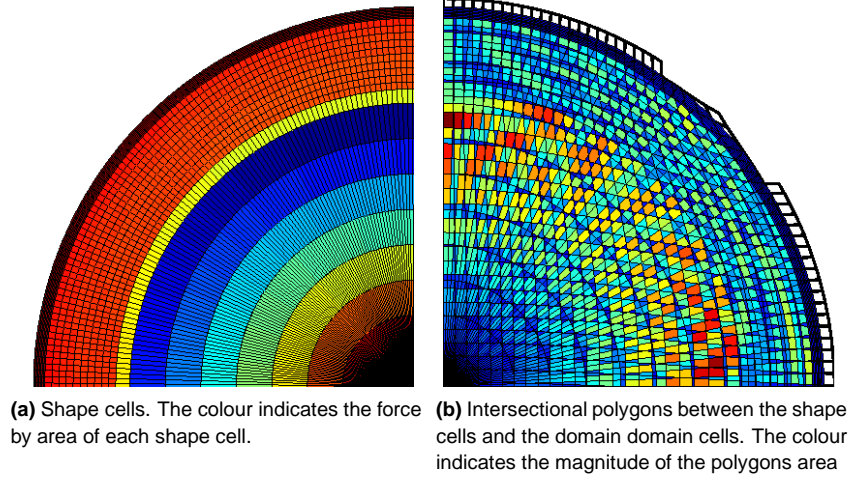


Figure 3: Force discretization algorithm.

$$\Psi(r, z) = \frac{U_\infty r^2}{2} + \frac{ar}{2} \int_0^\infty \zeta_{(-1,2,1)} dz', \quad (3)$$

$$U_z(r, z) = U_\infty + \frac{a}{2} \int_0^\infty \zeta_{(0,2,0)} dz', \quad (4)$$

$$U_r(r, z) = \frac{a}{2} \int_0^\infty \pm \zeta_{(0,2,1)} dz', \quad (5)$$

$$\zeta_{(\lambda,\mu,\nu)} = R_w^2(z') I_{(\lambda,\mu,\nu)}(R_w(z'), r, z - z'), \quad (6)$$

where $R_w(z)$ is the slipstream boundary radial position at an axial distance z from the disc (also defined as the wake width function), U_∞ is the freestream inflow velocity and $I_{(\lambda,\mu,\nu)}$ is the Bessel-Laplace integrals (BLI) introduced by Conway [5] and defined as Eq.7. The analytical expression of $I_{(-1,2,1)}$, $I_{(0,2,0)}$ and $I_{(0,2,1)}$, as well as the method used to derive them are described in the Appendix A.

$$I_{(\lambda,\mu,\nu)} = \int_0^\infty e^{-s|z|} s^\lambda J_\mu(sR) J_\nu(sr) ds, \quad (7)$$

where λ , μ and ν are integers and J_α are Bessel functions of the first kind.

In order to find the wake width function $R_w(z)$, Conway proposes in the appendix of [6] a recursive method based on the idea that the stream function Eq.3, is constant along the slipstream boundary, so that

$$\Psi(R_w(z), z) = \Psi(D/2, 0). \quad (8)$$

The loads on the disc are estimated by Conway's model from the stream function Eq.3, using Eq.9 [6].

$$L(r) = a\rho [\Psi(R_T, 0) - \Psi(r, 0)], \quad (9)$$

where L is the axial loading, a is the free parameter and ρ is the density.

To implement them into EllipSys, they are distributed over the disc grid and then inserted inside the computational mesh as described in Sec.2.

Full Rotor Computation

The complete geometry of the rotor and nacelle of a Nordtank 500 kW wind turbine is simulated up to a steady state solution [25] using EllipSys and the $k-\omega$ SST turbulence model [11]. The wind turbine is equipped with three LM 19.1 m blades and has a cylindrical nacelle (2 m diameter, 8.9 m length). In order to simplify the simulation, the geometry is lightly smoothed, the tower is not considered and the spinner and nacelle where rotated along with the rotor. Moreover, the simulation is carried out with a uniform inflow without the ground boundary.

The computational mesh, illustrated in Fig.4, is generated using Gridgen and HypGrid [21]. It is composed of 108 blocks of 32^3 cells (14.2 million cells). The y^+ at the solid walls is kept under 2. A no-slip boundary condition is imposed at the wall of the structure.

The Nordtank turbine rotates at 27.1 RPM. The inflow wind speed is set to 10 m/s. The whole steady-state simulation takes approximately 4 hours on 56 2.2Ghz nodes.

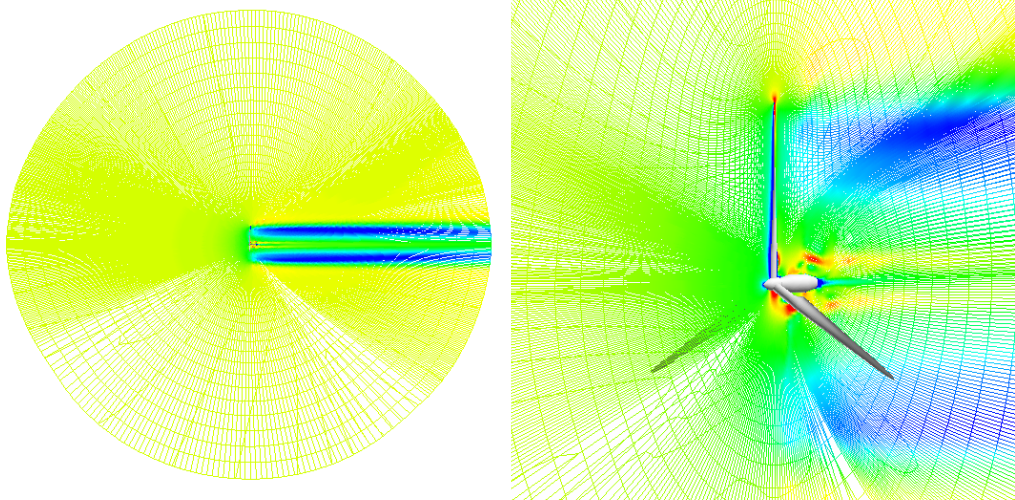


Figure 4: Illustration of the mesh surrounding the full-rotor computation. The colours illustrate the axial velocity magnitude.

The blades skin friction is integrated at 11 different positions and recombined into force vectors. These force vectors are then smeared over the corresponding annular section of the actuator disc grid. Finally, they are redistributed in the computational domain using the method presented in Sec.2 and in Fig.3.

3.2 Results

EllipSys vs Conway

Fig.5 presents the comparison of the CFD actuator disc model to Conway's heavily loaded actuator disc model. Conway's model uses a vorticity factor $a = -4U_\infty/D^2$, corresponding to $C_T = -0.4484$ applied in the CFD-AD. The model in EllipSys is used without any turbulence model in order to obtain a result close to the inviscid formulation of Conway.

The results show that the two models are in very close agreement, except for the radial velocity directly at the disc.

Actuator Disc vs Full Rotor Computation

The actuator disc model does not give a detailed description of the flow in the direct vicinity of the rotor blades. Instead, its application is to be used to model the far wake of a wind turbine. The region of comparison between the two kind of wind turbine flow model is chosen to

be $x \in [-3D, -1D]$ and $x \in [1D, 3D]$. The same inflow parameters for both the mean velocity and the turbulence are used in the actuator disc model and the full rotor computation.

On Fig.6, the axial U_x and tangential U_θ velocity components as well as the pressure are compared along the radial direction r , at different positions upstream and downstream of the rotor. The results are satisfactory in terms of mean velocity and pressure distribution.

However, the comparison in terms of turbulence parameters shows a large difference between the two models (not illustrated here). Indeed, the turbulence generated by the actuator disc lacks the detailed structures generated by the blade and nacelle geometry that still dominate in this region. The only turbulence generated by the actuator disc is produced through the mean velocity shear at the boundary of the wake. As the inflow turbulence is low, the production of turbulence is several orders of magnitude smaller than the turbulence generated by the blades and nacelle in the full rotor computation. Compared to atmospheric turbulence, the turbulence generated by the rotor is nonetheless an order of magnitude smaller and should not play a significant role in the far wake.

The difference between the two actuator disc computations (10 cells per rotor diameter (c/D))

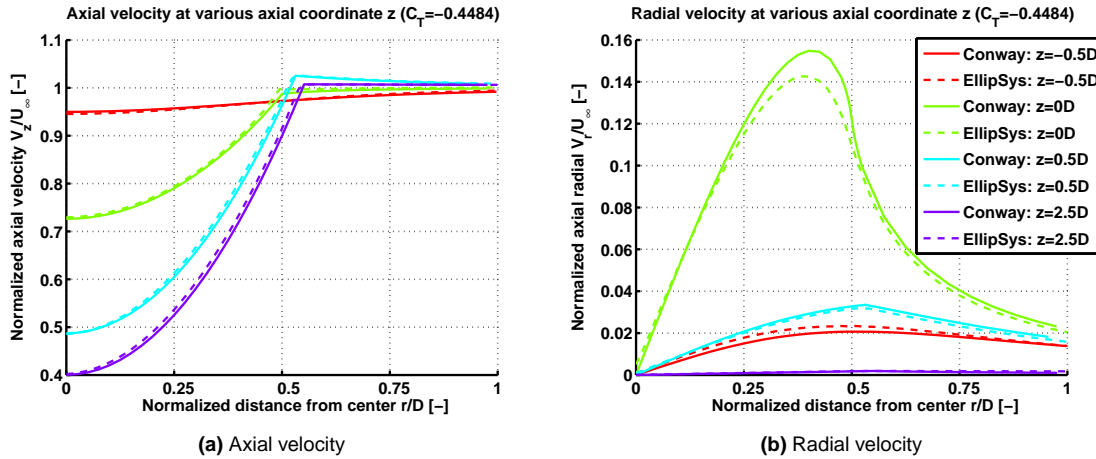


Figure 5: Actuator disc (parabolic wake case) compared to the analytical axisymmetric solution of Conway.

and 20 c/D) is visible, yet remains relatively small.

4 Discussion

The overall good agreements obtained in the results section demonstrate that actuator disc can be a cost effective way to model wind turbine wake. While the vortex structures present in the close wake region are not modelled correctly, the far wake region ($>3D$) seem to be a close match to a far more complex and expensive full rotor computation.

The small discrepancy in the radial direction at the rotor location probably comes from the fact that the CFD actuator disc model has a finite thickness, corresponding to the cell dimensions, while the model of Conway is infinitely thin. Moreover, the radial velocity evolves very rapidly at the disc position, which makes it difficult to precisely interpolate the velocities at the correct disc position. This was also commented by Schaffarczyk and Conway [19] who did a similar comparison of the analytical actuator disc with a CFD actuator disc.

This can present a problem rarely mentioned in the paper dealing with actuator discs. The position where the velocities are estimated, at the rotor, is a place where the velocity gradients are relatively high. As those velocities are necessary to estimate accurately the related

blade forces, the uncertainty associated with estimating those velocities is going to be propagated, through the forces estimation, in the wake of the wind turbine. When considering multiple wind turbines in cluster, the combined uncertainty may then become significant.

The comparison between the two actuator discs with 10 and 20 cells per rotor diameter indicates that using only 10 c/D is enough for obtaining a good resolution of the close wake flow features. In the far wake region, the gradients become smaller and the cell size becomes less critical. Using only 10 c/D open the possibility to numerically carry out large wind farm computations. For example, an hypothetical cluster of 10×10 wind turbines (WT), with a wind turbine spacing of 8D in each direction needs roughly $10 \text{ WT} \times 10D \times 10 \text{ c/D} = 1000$ cells in the two horizontal directions. With 128 cells in the vertical direction, this roughly needs 128M cells in the center of the wind farm, in order to obtain a good resolutions. These types of steady-state simulations can be solved in less than 10 hours on a large computer cluster.

However, please note that the studies presented here have been carried out without considering the large scale atmospheric turbulence. Computing the interaction between the large scale turbulence of the atmosphere and the smaller scale of turbulence of the wind turbine wake is a very complex task that only prohibitively expensive large eddy-simulation seems to be able to carry out [15].

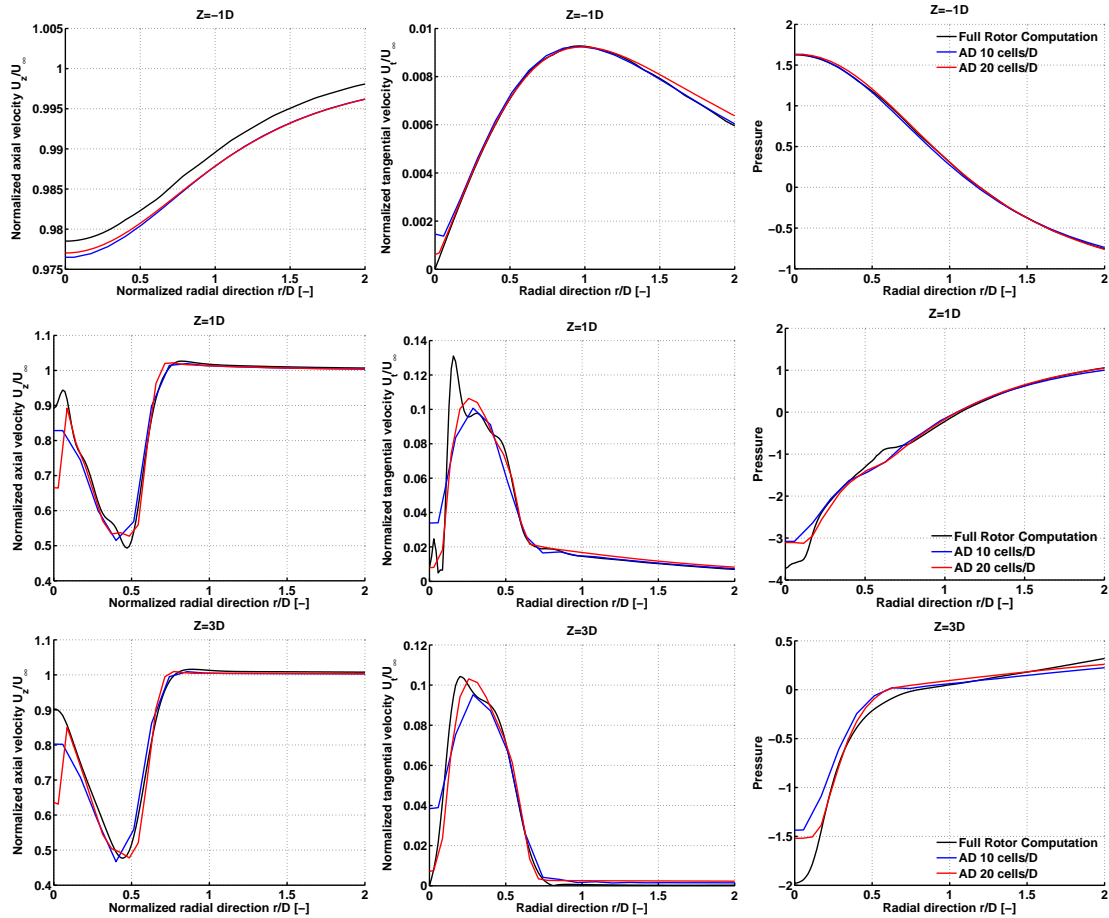


Figure 6: Comparison between the full rotor computation and the actuator disc model (with 10 and 20 c/D) of the normalized axial velocity in a cross section at hub height and at different positions upstream and downstream of the wind turbine.

5 Conclusion

This paper introduced the actuator shape model, a cost effective method to redistribute any kind of shape forces in a computer domain. The special case of the actuator disc is validated through two fundamentally different models, an analytical solution for an heavily loaded actuator disc, and full rotor computation. Both models give a close agreement with the actuator shape model introduced. The results show that even with a coarse resolution of the disc, the far wake region is well defined, which gives the possibility to carry out numerically full wind farm wake computation.

This study seem to show that the actuator disc is by itself a mature technology for the study of wind turbine wake. However, the lack of proper turbulence model that can account cost

effectively for both the large scale atmospheric scales and the smaller wake scales, still limits its use. More work is therefore needed on the topic of multiscale turbulence in order to address this problem.

Acknowledgement

This work has been funded by the Danish P.S.O. Research Project "Bottlenecks".

References

- [1] I. Ammara, C. Leclerc, and C. Masson. A viscous three-dimensional differential/actuator disc method for the analysis of wind farms. *ASME J. Sol. Energy Eng.*, 124(4):345–356, 2002.
- [2] J. T. Conway. Analytical solutions for the newtonian gravitational field induced by matter within axissym-

- metric boundaries. *Mon. Not. R. Astron. Soc.*, 316: 540–554, 2000.
- [3] J. T. Conway. Exact solutions for the magnetic fields of axisymmetric solenoids and current distributions. *IEEE Transactions on Magnetics*, 37:2977–2988, 2001.
- [4] J. T. Conway. Application of an exact nonlinear actuator disk theory to wind turbines. *ICNPAA 2002, Melbourne, Florida, USA*, 15 -17 May 2002.
- [5] J. T. Conway. Analytical solutions for the actuator disc with variable radial distribution of load. *J. Fluid Mech.*, 297:327–355, 1995.
- [6] J. T. Conway. Exact actuator disk solutions for non-uniform heavy loading and slipstream contraction. *J. Fluid Mech.*, 356:235–267, 1998.
- [7] J. T. Conway. Prediction of the performance of heavily loaded propellers with slipstream contraction. *CASI J.*, 44:169–174, 1998.
- [8] A. Crespo, J. Hernandez, and S. T. Frandsen. Survey of modelling methods for wind turbine wakes and wind farms. *Wind Energy*, (2):1–24, 1999.
- [9] L. Gilling, N. N. Sørensen, and P.-E. Réthoré. Imposing resolved turbulence by an actuator in a detached eddy simulation of an airfoil. *EWEC Marseille*, 2009. [PDF].
- [10] B. P. Leonard. A stable and accurate convective modelling procedure based on quadratic upstream interpolation. *Comp. Meth. in Appl. Mech. and Engrng.*, 19: 59–98, 1979.
- [11] F. R. Menter. Zonal two equation $k-\omega$ turbulence models for aerodynamic flows. *AIAA Journal*, (93-2906), 1993.
- [12] J. A. Michelsen. Basis3D - a platform for development of multiblock PDE solvers. Technical report AFM 92-05, Technical University of Denmark, Lyngby, 1992.
- [13] R. Mikkelsen. *Actuator Disc Methods Applied to Wind Turbines*. PhD thesis, Technical University of Denmark, Mek dept, 2003. [PDF].
- [14] S. V. Patankar and D. B. Spalding. A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *Int. J. Heat Mass Transfer*, 15:1787–1972, 1972.
- [15] P.-E. Réthoré. *Wind Turbine Wake in Atmospheric Turbulence*. PhD thesis, Aalborg University - Risø DTU, October 2009. [PDF].
- [16] P.-E. Réthoré and N. N. Sørensen. Actuator disc model using a modified Rhie-Chow/SIMPLE pressure correction algorithm. *EWEC Brussels*, 2008. [PDF].
- [17] C. M. Rhie and W. L. Chow. Numerical study of the turbulent flow past an airfoil with trailing edge separation. *AIAA Journal*, 21:1525–1532, 1983.
- [18] B. Sanderse. Aerodynamics of wind turbine wakes - literature review. Technical Report ECN-E-09-016, ECN, Netherlands, 2009. [PDF].
- [19] A. P. Schaffarczyk and J. T. Conway. Comparison of a nonlinear actuator disk theory with numerical integration including viscous effects. *CASI J.*, 46:209–215, 2000.
- [20] N. N. Sørensen. *General Purpose Flow Solver Applied to Flow over Hills*. PhD thesis, Technical University of Denmark, 1994.
- [21] N. N. Sørensen. HypGrid2D – a 2-D mesh generator. Technical Report Risø-R-1035(EN), Risø National Laboratory, 1998. [PDF].
- [22] N. Troldborg. *Actuator Line Modeling of Wind Turbine Wakes*. PhD thesis, DTU-MEK, Denmark, 2008. [PDF].
- [23] L. J. Vermeer, J. N. Sørensen, and A. Crespo. Wind turbine wake aerodynamics. *Progress in Aerospace Sciences*, 39:467–510, 2003.
- [24] J. W. Wrench. Review of: Heuman lambda function. *Mathematics of Computation*, 20:627–628, 1966.
- [25] F. Zahle and N. N. Sørensen. Characterisation of the unsteady flow in the nacelle region of a modern wind turbine. *EWEC Marseille*, 2009. [PDF].

Appendix

A Bessel-Laplace Integrals

Conway [2, 5] has developed a recursive method to express the Bessel-Laplace integrals (BLI) of type $I_{(\lambda,\mu,\nu)}$ Eq.7 using elliptic integral function to derive a numerical solution.

In order to solve the parabolic wake case, only three basic BLIs are needed ($I_{(-1,2,1)}$, $I_{(0,2,0)}$ and $I_{(0,2,1)}$). While the analytical expression of $I_{(-1,2,1)}$ is given in [3], the analytical expression of $I_{(0,2,0)}$ and $I_{(0,2,1)}$ could not be found in the literature. The recursive method is therefore used to derive them. The algorithm used to implement this method is briefly summarized in the following.

First of all, it is interesting to notice the symmetrical property of the Bessel-Laplace integrals [2],

$$I_{(\lambda,\mu,\nu)}(R, r, z) = I_{(\lambda,\nu,\mu)}(r, R, z). \quad (10)$$

Furthermore, the Bessel-Laplace integrals can be related to each other through recursive relations and some initial start formulae. The recursive relations are given as [2],

$$I_{(0,\nu,\nu)} = \frac{4(\nu-1)\omega}{2\nu-1} I_{(0,\nu-1,\nu-1)} - \frac{2\nu-3}{2\nu-1} I_{(0,\nu-2,\nu-2)}, \quad (11)$$

$$I_{(1,\nu,\nu)} = \frac{(2\nu-1)|z|k^4}{8Rr(1-k^2)} (I_{(0,\nu-1,\nu-1)} - \omega I_{(0,\nu,\nu)}), \quad (12)$$

$$I_{(0,\nu+1,\nu)} = \frac{r}{R} I_{(0,\nu,\nu-1)} + \frac{r}{2\nu} (I_{(1,\nu+1,\nu+1)} - I_{(1,\nu-1,\nu-1)}), \quad (13)$$

which is only valid for $\nu \neq 0$,

$$I_{(0,\mu,\nu)} = \frac{2(\nu+1)|z|}{r(\nu+1-\mu)} I_{(0,\mu,\nu+1)} - \frac{2(\nu+1)R}{r(\nu+1-\mu)} I_{(0,\mu-1,\nu+1)} + \frac{(\nu+1+\mu)}{(\nu+1-\mu)} I_{(0,\mu,\nu+2)}, \quad (14)$$

and

$$I_{(\lambda,\mu,\nu)} = \frac{R}{2\mu} (I_{(\lambda+1,\mu+1,\nu)} - I_{(\lambda+1,\mu-1,\nu)}), \quad (15)$$

which is only valid for $\lambda < 0$ and where $\omega = (R^2 + r^2 + z^2)/(2rR)$ is the parameter of an associated Legendre function and $k = \sqrt{4rR/[(R+r)^2 + z^2]}$ is the modulus of an elliptic integral.

The initial start formulae are $I_{(0,0,0)}$, $I_{(0,1,0)}$ and $I_{(0,1,1)}$ and can be found in [3],

$$I_{(0,0,0)} = \frac{k\mathbf{K}(k)}{\pi\sqrt{rR}}, \quad (16)$$

if $r < R$,

$$I_{(0,1,0)} = \frac{1}{R} \left[1 - \frac{|z|k\mathbf{K}(k)}{2\pi\sqrt{rR}} - \frac{\Lambda_0(|\beta|, k)}{2} \right] \quad (17)$$

if $r > R$,

$$I_{(0,1,0)} = \frac{1}{R} \left[-\frac{|z|k\mathbf{K}(k)}{2\pi\sqrt{rR}} + \frac{\Lambda_0(|\beta|, k)}{2} \right] \quad (18)$$

$$I_{(0,1,1)} = \frac{(2-k^2)\mathbf{K}(k) - 2\mathbf{E}(k)}{\pi k\sqrt{rR}} \quad (19)$$

where $\beta = \arcsin(|z|/\sqrt{(|z|-R)^2 + z^2})$ is the Jacobi amplitude, $\mathbf{K}(k)$ is the complete elliptic integral function of the first kind, $\mathbf{E}(k)$ is the complete elliptic integral function of the second kind and $\Lambda_0(\beta, k)$ is Heuman's Lambda function defined as [2, 24],

$$\Lambda_0(\beta, k) = \frac{2}{\pi} \left[\mathbf{E}(k)F(\beta, k') + \mathbf{K}(k) \left(E(\beta, k') - F(\beta, k') \right) \right], \quad (20)$$

where $k' = \sqrt{1-k^2}$ is the complementary modulus for an elliptic integral of modulus k .

Using the recursive laws and the start formulae, it is possible to create a recursive algorithm to express any Bessel-Laplace integral in terms of elliptic integral functions. The procedure suggested is to first use the relation $I_{(\lambda,\mu,\nu)}$ Eq.15 to find an expression where $\lambda = 0$. Then use relation $I_{(0,\mu,\nu)}$ Eq.14 to reduce the gap between μ and ν in order to use the relation $I_{(0,\nu+1,\nu)}$ Eq.13 and finally $I_{(0,\nu,\nu)}$ Eq.11. Whenever possible, a shortcut in the recursion can be taken, using the symmetrical relation Eq.10. This procedure is only valid for $\lambda \leq 0$, yet this encompasses all the cases described by Conway's actuator disc model.

In order to test the algorithm, one can compare to some of the Bessel-Laplace integral solutions given in [3].

Using this algorithm it is possible to express $I_{(-1,2,1)}$, $I_{(0,2,0)}$ and $I_{(0,2,1)}$ with respect to elliptic integral functions. $I_{(-1,2,1)}$ is also given in [3], if ($r < R$),

$$I_{(-1,2,1)} = \mathbf{E}(k)\Theta_{(-1,2,1)}^E + \mathbf{K}(k)\Theta_{(-1,2,1)}^K + \frac{r}{R^2} \left(\frac{z\Lambda_0(\beta, k)}{2} - |z| \right) \quad (21)$$

if ($r > R$),

$$I_{(-1,2,1)} = \mathbf{E}(k)\Theta_{(-1,2,1)}^E + \mathbf{K}(k)\Theta_{(-1,2,1)}^K + \frac{rz}{2R^2}\Lambda_0(\beta, k), \quad (22)$$

where the coefficients $\Theta_{(-1,2,1)}^E$ and $\Theta_{(-1,2,1)}^K$ are given as,

$$\Theta_{(-1,2,1)}^E = \frac{1}{k} \frac{1}{2\pi} \sqrt{\frac{r}{R}} \left[\frac{4r}{R} - \frac{8(2-k^2)}{3k^2} \right], \quad (23)$$

$$\Theta_{(-1,2,1)}^K = \frac{k}{2\pi} \sqrt{\frac{r}{R}} \left[\frac{(4-k^2)(4-3k^2)}{3k^4} - \frac{r^2}{R^2} \right]. \quad (24)$$

$I_{(0,2,0)}$ is found using the recursive rules, if $(r > R)$

$$I_{(0,2,0)} = \frac{\mathbf{E}(k)\Theta_{(0,2,0)}^E + \mathbf{K}(k)\Theta_{(0,2,0)}^K}{k^3\pi(rR)^{5/2}} - \frac{\Lambda_0(\beta, k)|z|}{R^2}, \quad (25)$$

if $(r < R)$

$$I_{(0,2,0)} = \frac{\mathbf{E}(k)\Theta_{(0,2,0)}^E + \mathbf{K}(k)\Theta_{(0,2,0)}^K}{k^3\pi(rR)^{5/2}} + \frac{[\Lambda_0(\beta, k) - 2]|z|}{R^2}, \quad (26)$$

where the coefficients $\Theta_{(0,2,0)}^E$ and $\Theta_{(0,2,0)}^K$ are found as Eq.27 and Eq.28.

$$\Theta_{(0,2,0)}^E = -4rR \left[R \left(2(-2 + k^2)r + k^2R \right) + k^2|z|^2 \right], \quad (27)$$

$$\Theta_{(0,2,0)}^K = rR^2 \left[(-16 + 16k^2 - 3k^4)r - 2k^2(-2 + k^2)R \right] + r \left[k^4(r - 2R) + 4k^2R \right] |z|^2. \quad (28)$$

$I_{(0,2,1)}$ is also found using the recursive rules, if $(r > R)$,

$$I_{(0,2,1)} = \mathbf{E}(k)\Theta_{(0,2,1)}^E + \mathbf{K}(k)\Theta_{(0,2,1)}^K + \frac{\Lambda_0(\beta, k)r}{2R^2}, \quad (29)$$

if $(r < R)$,

$$I_{(0,2,1)} = \mathbf{E}(k)\Theta_{(0,2,1)}^E + \mathbf{K}(k)\Theta_{(0,2,1)}^K + \frac{[2 - \Lambda_0(\beta, k)]r}{2R^2}, \quad (30)$$

where the coefficients $\Theta_{(0,2,1)}^E$ and $\Theta_{(0,2,1)}^K$ are given as,

$$\Theta_{(0,2,1)}^E = \frac{2|z|}{k\pi R\sqrt{rR}}, \quad (31)$$

$$\Theta_{(0,2,1)}^K = -\frac{|z|(k^2(r - 2R) - 4R)}{2k\pi R^2\sqrt{rR}}. \quad (32)$$