Seventh International Conference on
Computational Fluid Dynamics (ICCFD7),
Big Island, Hawaii, July 9-13, 2012

ICCFD7-3503

Numerical simulation of horizontal-axis wind turbine (HAWT)

M. Bergmann[*], A. Iollo[*]
Corresponding author: michel.bergmann@inria.fr

[*] INRIA Bordeaux Sud Ouest, Team MC2.
Université de Bordeaux, 351 cours de libération, 33 405 Talence Cedex, France.

Abstract: The aim of this study is to estimate the wind power that can be extracted by an horizontal-axis wind turbine (HAWT) as a function of upstream wind. The incompressible Navier-Stokes equations are solved on a fixed cartesian mesh via a second-order accurate scheme in space and time. The turning blades and the mast are modeled by a penalization term in the governing equations within a collocated Chorin-Temam fractional time integration algorithm. This numerical procedure allows massive parallelization by using existing distributed linear-algebra libraries. The test case under consideration is the two blades (S809 airfoil) NREL ametest wind turbine for which wind tunnel data exists.

Keywords: Wind turbine, Cartesian mesh, Immersed Boundary, Turbulence Modeling.

1 Introduction

The aerodynamic optimization of an horizontal-axis wind turbine (HAWT) is a real challenge for renewable energy purposes. The challenge is to find efficient blade shapes for a large spectrum of tip speed ratio (ratio between the velocity at the blade tip to the wind velocity). Hence, it is important to estimate the extracted power as a function of incoming wind. Several simple models exist in the literature for this purpose. For example, the model introduced by Sorensen and Myken [1] is based on an actuator disk model. Despite its simplicity (steady, axisymmetric), this model gives reasonable results in terms of integral quantities. Of course, it does not take into account three-dimensional effects that can significantly affect the flow, especially near the wing tip and the hub. A full three-dimensional model can significantly improve the prediction of such phenomena. In this sense, we have developed a fully three-dimensional model based on the incompressible Navier-Stokes equations (§2). The numerical difficulty is to compute the solution past moving interfaces (the blades) and in this context, to conveniently model turbulence.

To our knowledge there exist very few numerical models able to overcome difficulties linked to both moving interfaces and turbulence. The existing methods are usually based on body fitted meshes. For those methods it is necessary to regularly remesh for moving blades. For meshes with several millions of nodes, this can be very expensive or even unfeasible. To avoid this step, a fixed mesh method is then used in this study. The three-dimensional incompressible Navier-Stokes equation are discretized on a cartesian mesh. The fluid/structure interface, represented by the zero level set functions, is computed thanks to a second order penalization [2] similar to immersed boundary methods [3]. The turbulence is modeled using a LES SubGrid Scale Smagorinsky-Lilly model.
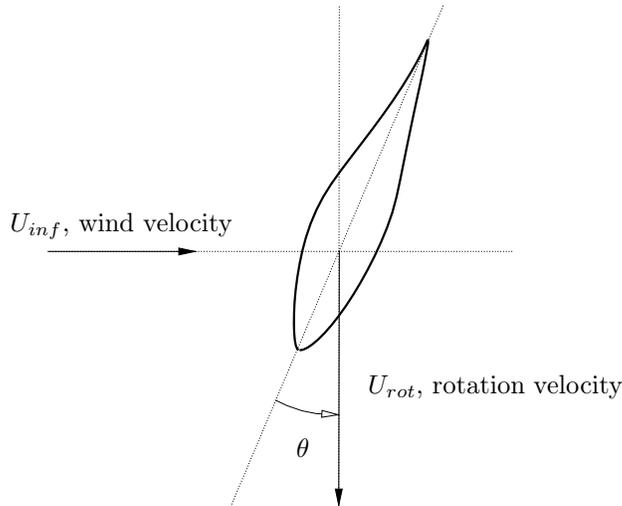
Figure 1: Blade profile and twist θ.

2 Modeling and numerical approach

2.1 Windturbine modeling

The windturbine under consideration is the NREL 10-m Wind Turbine, tested in NASA Ames 80'x120' Wind Tunnel (see `http://wind.nrel.gov/amestest/`). The blade profile (S809 airfoil) is presented in figure 1 where the twist angle θ varies between $30\,°$ near the hub to $-2.5\,°$ at the end. The whole blade is presented in figure 2. Note that the blade shape near the hub is circular. In this study we will consider the 2 blades windturbine. Figure 3 shows the whole windturbine with the 2 blade, the hub and the mast. The rotation of the 2 blades is fixed to be at $72\,rpm$. The position of the blades is determined via a level set, which is advected in a Lagrangian way on each mesh point (see figure 2 for mesh). The velocity of the a point $\boldsymbol{x}$ of the blade at time t is noted $\boldsymbol{u}_b(\boldsymbol{x},\,t)$. This velocity is the one used to model the flow around the windturbine (see next section).

2.2 Flow modeling

The domain under consideration is a 3D box $\Omega = \Omega_f \cup \Omega_b$, where Ω_f is the domain filled by fluid, and Ω_b is the domain defined by the windturbine (see figure 3). The box boundaries and the windturbine boundary are respectively denoted by $\partial\Omega_f$ and $\partial\Omega_b$. The flow around moving bodies is modeled using the incompressible Navier-Stokes equations:

$$\rho\left(\frac{\partial\boldsymbol{u}}{\partial t} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u}\right) = -\nabla p + \mu\Delta\boldsymbol{u} + \rho\boldsymbol{g} \ \ \text{in} \ \ \Omega_f, \tag{1a}$$

$$\boldsymbol{\nabla}\cdot\boldsymbol{u} = 0 \ \ \text{in} \ \ \Omega_f, \tag{1b}$$

with initial conditions $\boldsymbol{u}(\boldsymbol{x},\,0) = \boldsymbol{u}_0(\boldsymbol{x})$ in Ω_f, and boundaries conditions on $\partial\Omega_f$ and $\boldsymbol{u}(\boldsymbol{x},\,t) = \boldsymbol{u}_b(\boldsymbol{x},\,t)$ on $\partial\Omega_b$.

Since we aim at using a cartesian mesh we want to avoid imposing explicitly the boundary conditions $\boldsymbol{u}(\boldsymbol{x},\,t) = \boldsymbol{u}_s(\boldsymbol{x},\,t)$ on $\partial\Omega_b$ onto the body. The boundary condition can be imposed implicitly using an external force depending of the blade velocity $\boldsymbol{u}_b$, denoted $\boldsymbol{f}(\boldsymbol{u}_b)$, and so that we have to solve

$$\rho\left(\frac{\partial\boldsymbol{u}}{\partial t} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u}\right) = -\nabla p + \mu\Delta\boldsymbol{u} + \rho\boldsymbol{g} + \rho\boldsymbol{f}(\boldsymbol{u}_b) \ \ \text{in} \ \ \Omega, \tag{2a}$$

$$\boldsymbol{\nabla}\cdot\boldsymbol{u} = 0 \ \ \text{in} \ \ \Omega. \tag{2b}$$
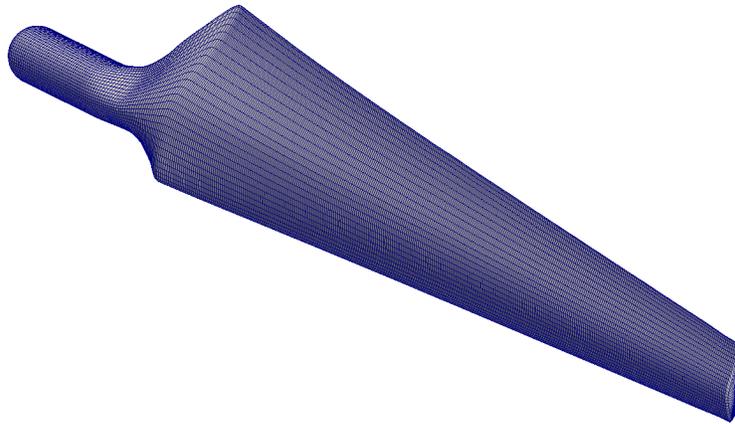
Figure 2: Blade geometry and mesh.

Figure 3: NREL turbine. Left, real turbine (http://www.nrel.gov/data/pix/Jpegs/17305.jpg). Right: Model of the windturbine.

In that study, the external force, $\boldsymbol{f}(\boldsymbol{u}_b)$, will be computed by a hybrid method, using both penalization [4, 2] and immersed boundary techniques [3, 5].

2.3 Numerical approach

Equations (2) are spatially discretized on a fixed cartesian mesh with space step $\Delta\boldsymbol{x} = (\Delta x,\,\Delta y,\,\Delta z)$. All the derivative are computed using a central finite difference scheme with second order accuracy except the convective terms that are computed using an third order upwind scheme. The temporal discretization is performed thanks to a second order projection method [6] originally based on fractional steps methods [7, 8]

The projection method used is the following:

1. The projection step;

$$\frac{\boldsymbol{u}^* - \boldsymbol{u}^n}{\Delta t} + \nabla q = -\left[(\boldsymbol{u}\cdot\nabla)\boldsymbol{u}\right]^{n+1/2} + \frac{\nu}{2}\left(\Delta\boldsymbol{u}^* + \Delta\boldsymbol{u}^n\right) + \lambda\chi(\boldsymbol{u}_b - \boldsymbol{u}^*) \tag{3a}$$

with boundary conditions. the term q is an approximation of the pressure fields $p^{n+1/2}$, $\lambda = 10^8$ is a penalty term, χ is the windturbine characteristic function, *i.e.* $\chi = 1$ in Ω_b and $\chi = 0$ elsewhere. The velocity $\boldsymbol{u}_b$ is the rigid velocity of the windturbine. The convective term at time $n + 1/2$ is obtained thanks to an Adams Bashforth scheme. The penalty term is introduced to limit the mass variation when body moves. The fresh cells that are created in the fluid have a velocity that is consistent with the neighbors in the fluid.

2. Poisson step;

$$\Delta\phi = \nabla\cdot\boldsymbol{u}^* \tag{3b}$$

3. Projection step (correction);

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^* - \nabla\phi \tag{3c}$$

$$\boldsymbol{p}^{n+1/2} = q + \phi - \frac{\nu\Delta t}{2}\Delta\phi \tag{3d}$$

4. Body motion and immersed boundary correction.

The windturbine is displaced thanks to the rotation velocity. Since the windturbine boundary $\partial\Omega_s$ does not fit the cartesian mesh, the boundary conditions $\boldsymbol{u}(\boldsymbol{x},\,t) = \boldsymbol{u}_b(\boldsymbol{x},\,t)$ on $\partial\Omega_b$ are imposed using immersed boundary method [3, 5]. The velocity of a mesh point $\boldsymbol{x} \in \Omega_b$ with at least one neighbor $\boldsymbol{x} \pm \Delta\boldsymbol{x}$ in Ω_f is imposed to obtain the desired velocity value $\boldsymbol{u}_b(\boldsymbol{x}^*,\,t)$ on the nearest fish boundary point $\boldsymbol{x}^*$. This velocity is denoted $\boldsymbol{u}_{ibm}$ This point $\boldsymbol{x}^*$ can easily be found after computing the level function $\phi(\boldsymbol{x},\,t)$ associated to the windturbine. The immersed boundary method allows to keep the second order accuracy in scpace near the windturbine boundaries.

The numerical scheme is globally second order accuracy in both space and time. A simple Smagorinsky-Lilly turbulence model is also used.

3 Results

Figure 4 presents a numerical simulation of the real NREL ametest HAWT (blades length $\ell = 5m$, $\nu = 10^{-6}$, incoming wind= $10m/s$ and rpm= 72). The vorticity obtained is physically consistent. Indeed, the tip vortices, the interactions with ground as well as vortices generated by the mast are clearly visible.

In what follows we will compare the thrust and the mechanical power generated by the windturbine obtained with our numerical solver to those obtained experimentally [9]. In order to increase the resolution, the computational domain will only consider the 2 blades with the hub, excluding the mast. The hub is centered onto the point $(x_G,\,y_G,\,z_G) = (0,\,0,\,0)$ (the center of mass of the 2 blades) and the domain is $-2\,m \leq x \leq 3\,m$, $-8\,m \leq y \leq 8\,m$ and $-8\,m \leq z \leq 8\,m$. The associated mesh is $600 \times 400 \times 400 \approx 100$ millions nodes. The leading grid steps are $\Delta y = \Delta z = 4\,cm$ and $\Delta x = 0.8\,cm$. Since the blade length,
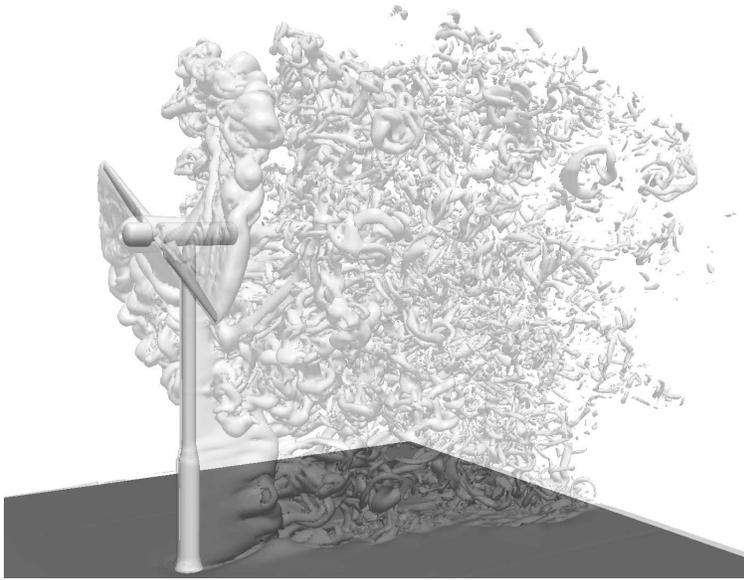
Figure 4: Vorticity representation of the flow generated by the NREL ametest HAWT with $72\,rpm$ and incoming wind with $10\,m.s^{-1}$

maximal width and maximal thickness are $5.03\,m$, $0.737\,m$ and $0.153\,cm$, there are 125 points on the blade span, 20 points on the blade chord and also 20 on the blade thickness. Previous simulations around cylinder and ellipses show that 20 points in each direction are enough to simulate with good accuracy laminar flows. The question of turbulence modeling and simulation remains open. As mentioned before we used a simple LES Smagorinsky Lilly model. Even if the domain is quite small in the wind direction, we did not observe any spurious pressure reflection at the outflow. Indeed, the prediction equation (3a) is solved using non reflecting outflow boundary conditions [10].

Introducing the dimensionless stress tensor $\mathbb{T}(\boldsymbol{u},\,p) = -p\boldsymbol{I} + \frac{1}{Re}(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^T)$ and $\boldsymbol{n}$ the unit outward vector to $\partial\Omega_b$, the forces and the torques exerted by the fluid onto the bodies are:

$$\boldsymbol{F}_b = -\int_{\partial\Omega_b} \mathbb{T}(\boldsymbol{u},\,p)\,\boldsymbol{n}\,\mathrm{d}\boldsymbol{x}, \tag{4a}$$

$$\boldsymbol{\mathcal{M}}_b = -\int_{\partial\Omega_b} \boldsymbol{r}_G \wedge (\mathbb{T}(\boldsymbol{u},\,p)\,\boldsymbol{n})\,\mathrm{d}\boldsymbol{x}, \tag{4b}$$

where $\boldsymbol{r}_G = (x - x_G,\, y - y_G,\, z - z_G)^T = (x,\, y,\, z)^T$. Since the boundary $\partial\Omega_b$ does not fit the fluid mesh, the forces and the torques are numerically integrated onto the blade mesh (see fig. 2) where $\nabla\boldsymbol{u}$ and p are obtained thanks to Lagrange interpolation on the fluid cells containing each blade mesh point.

The mechanical power can be obtained from $P = \boldsymbol{\mathcal{M}}_b \cdot \boldsymbol{\Omega}$, where $\boldsymbol{\Omega} = (1.2,\, 0,\, 0)^T$ is the angular velocity. The actual angular velocity is $72\,rpm$ that is 1.2 rotation per second.

Figure 5 shows the evolution of the thrust for several wind velocities experimentally obtained and with our numerical method. The two curves show good agreements. The windturbine acts as a (not efficient) propeller for wind velocities below $3\,m/s$. For wind velocities greater than that threshold the turbine creates drag. The mechanical power, however, does not show the same degree of accuracy as it can seen in figure 6. The mechanical power computed numerically is approximatively two times lower than the experimental one. It is however interesting to notice that the variations are similar, *i.e.* both values and gradients have the same sign. These differences can be explained by the domain confinement and by the fact that the boundary layers may not been computed accurately (turbulence modeling).
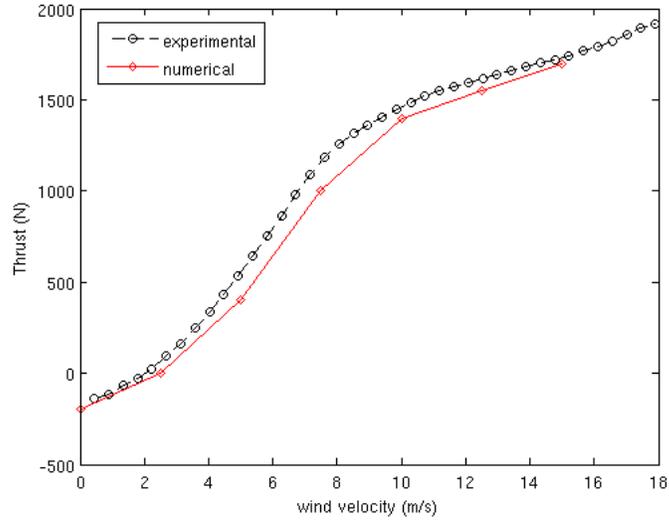
Figure 5: Evolution of the thrust versus the wind velocity. Comparison between experimental (NREL) and our numerical results.
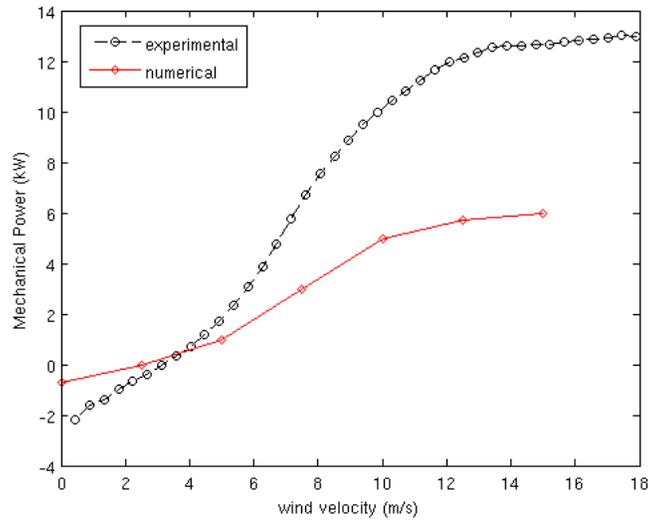
Figure 6: Evolution of the mechanical power versus the wind velocity. Comparison between experimental (NREL) and our numerical results.

4   Conclusion and Future Work

A method to model and simulate the flow over a windturbine is presented. This method is based on cartesian mesh where the bodies are modeled thanks to both penalization and immersed boundary. The windturbine under consideration is the 2 blades (S809 airfoil) NREL ametest that have been intensively studied experimentally in wind tunnel. Our numerical results show the same tendencies of experimental results, but the accuracy need to be increased. While the thrust generated by the blades is computed with high accuracy, the mechanical power extracted by the windturbine does not show the same accuracy. However, the numerical power show the same tendency than the experimental one. The main drawback of using cartesian mesh is that the boundary layers may not be computed the enough accuracy. One solution is to refine the mesh around the blades. Another solution is to compute numerical zoom closely around the windturbine using boundary conditions obtained from simulation on a larger domain with coarser grid. We will explore those paths in next future.

References

[1] JN. Sørensen and A. Myken. Unsteady actuator disc model for horizontal axis wind turbine. *J.s Wind Eng. Ind. Aerodyn.*, 39:139–149, 1992.

[2] M. Bergmann and A. Iollo. Modeling and simulation of fish-like swimming. *Journal of Computational Physics*, 230(2):329 – 348, 2011.

[3] R. Mittal and G. Iaccarino. Immersed boundary methods. *Annu. Rev. Fluid Mech.*, 37:239–261, 2005.

[4] P. Angot, C.H. Bruneau, and P. Fabrie. A penalization method to take into account obstacles in a incompressible flow. *Num. Math.*, 81(4):497–520, 1999.

[5] R. Mittal, H. Dong, M. Bozkurttas, F.M. Najjar, A. Vargas, and A. von Loebbecke. A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries. *Journal of Computational Physics*, 227(10):4825 – 4852, 2008.

[6] David L. Brown, Ricardo Cortez, and Michael L. Minion. Accurate projection methods for the incompressible navier-stokes equations. *Journal of Computational Physics*, 168(2):464 – 499, 2001.

[7] A.J. Chorin. Numerical solution of the Navier-Stokes equations. *Math. Comp.*, 22:745–762, 1968.

[8] R. Temam. Sur l'approximation de la solution des equations de navier-stokes par la méthode des pas fractionnaires ii. *Archiv. Rat. Mech. Anal.*, 32:377–385, 1969.

[9] P. Giguère and M.S. Selig. Design of a tapered and twisted blade for the nrel combined experiment rotor. Technical report, NREL/SR-500-26173, 1999.

[10] G. Jin and M. Braza. A Nonreflecting Outlet Boundary Condition for Incompressible Unsteady Navier-Stokes Calculations. *J. Comp. Phys.*, 107(2):239–253, 1993.