Direct Finite Element Simulation of the Turbulent Flow Past a Vertical Axis Wind Turbine

Van-Dang Nguyen[a], Johan Jansson[a], Anders Goude[b], Johan Hoffman[a]

[a]*Department of Computational Science and Technology, KTH Royal Institute of Technology, Stockholm, Sweden*
[b]*Ångström Laboratory, Uppsala University, Uppsala, Sweden*

Abstract

There is today a significant interest in harvesting renewable energy, specifically wind energy, in offshore and urban environments. Vertical axis wind turbines get increasing attention since they are able to capture the wind from any direction. They are relatively easy to install and to transport, cheaper to build and maintain, and quite safe for humans and birds. Detailed computer simulations of the fluid dynamics of wind turbines provide an enhanced understanding of the technology and may guide design improvements. In this paper, we simulate the turbulent flow past a vertical axis wind turbine for a range of rotation angles in parked and rotating conditions. We propose the method of Direct Finite Element Simulation in a rotating ALE framework, abbreviated as DFS-ALE. The simulation results are validated against experimental data in the form of force measurements. It is found that the simulation results are stable with respect to mesh refinement and that the general shape of the variation of force measurements over the rotation angles is captured with good agreement.

Highlights

- An efficient DFS-ALE method to simulate the turbulent flow of a vertical axis wind turbine in parked and rotating conditions.
- A high-performance computing software implemented in FEniCS-HPC.

Email addresses: `vdnguyen@kth.se` (Van-Dang Nguyen), `jjan@kth.se` (Johan Jansson), `anders.goude@angstrom.uu.se` (Anders Goude), `jhoffman@kth.se` (Johan Hoffman)

- Good validations against experimental data in parked and rotating conditions.

Keywords: Turbulent simulation, VAWT, DFS-ALE, FEniCS-HPC.

1. Introduction

1.1. Background

The interest in harvesting renewable energy, especially wind energy, in offshore as well as in the urban environment has increased significantly in recent years [1, 2, 3, 4]. Vertical axis wind turbines (VAWT) get more and more attention since they are able to capture the wind from any direction. They are easy to install, easy to transport, cheaper to build and maintain, and quite safe to humans and birds. They are especially suitable for urban areas or areas with extreme weather. However, they are less efficient than horizontal axis turbines because only one blade is active at a time, and VAWTs also generate a relatively high degree of vibration and noise pollution. These are some of the issues that researchers address to make VAWT more popular, see e.g. [5].

Detailed computational fluid dynamics (CFD) simulations of the flow around VAWTs may give new insights, and provide guidance for design improvements. Fluid dynamics is governed by the Navier-Stokes equations, and the balance of viscous and inertial effects is determined by the Reynolds number (Re),

$$Re = \frac{\rho U \mathcal{L}}{\mu} = \frac{U \mathcal{L}}{\nu}, \tag{1}$$

where ρ is density, μ dynamic viscosity, U a characteristic velocity scale, $\mathcal{L}$ a characteristic length scale, and ν kinematic viscosity $\nu = \mu/\rho$. For high Re the flow is turbulent, which corresponds to chaotic particle trajectories and vortices on a range of scales. The main challenge of CFD is to model turbulent flow, which typically is always present in the flow around a VAWT at operational conditions.

Simulation of turbulence is difficult due to the complex mixing of spatial and temporal scales. The industry standard has long been RANS, where a statistical average of the flow is simulated, using turbulence models to model the effect of the fluctuating components of the flow. Large Eddy Simulation (LES) was developed based on the idea of approximating a filtered solution of the Navier-Stokes equations, with the effect of unresolved scales modeled in a subgrid model. LES is very expensive but is still a viable option to RANS in some applications. The main problem with both LES and RANS is that the subgrid and turbulence models may be problem dependent, so that model parameters must be tuned to the particular problem at hand. Direct numerical simulation (DNS) is based on the full resolution of the Navier-Stokes equations, without any turbulence model or subgrid model, but with a computational cost so high that a VAWT simulation would exceed even the most powerful computers of today. We refer to the following VAWT studies for examples of LES [6, 7, 8], RANS [9, 10], and a coupled LES-RANS approach [11].

1.2. Problem formulation

In this paper, we will investigate an alternative to RANS, LES, and DNS for simulation of VAWT fluid dynamics. With a focus on aerodynamics, we have developed an approach to turbulence simulation without any turbulence or subgrid model, as in LES or RANS, but avoiding the computational cost of DNS. In short, it is a Galerkin finite element method, with least-squares stabilization of the residual of the Navier-Stokes equations. Further, adaptive mesh refinement is used to minimize the computational cost of the method, which we refer to as Direct Finite Element Simulation (DFS) [12, 13, 14]. The stabilization works like an automatic turbulence model which is based on the equations themselves, through the residual, without any explicit model terms, and thus do not require tuning to specific problems. Residual-based turbulence simulation is becoming a popular methodology, also for turbine simulations, e.g. in the form of Variational Multiscale (VMS) methods [15].

We extend the method to an Arbitrary Lagrangian-Eulerian (ALE) discretization to model a rotating turbine, and refer to as the method DFS-ALE.

The numerical method is implemented in the FEniCS-HPC framework [16, 17], which shows near optimal weak and strong scaling up to thousands of cores on supercomputers. To model turbulent boundary layers, we use a simple wall shear stress model where we assume the skin friction to be negligible for high Reynolds numbers, which corresponds to a slip velocity boundary condition.

To validate the DFS-ALE method for VAWT simulations, we consider a VAWT problem for which we also have experimental data in terms of force measurements in parked conditions [18] and rotating conditions [19]. The case of a parked turbine is an important design case with respect to the survival of the turbine over time and two approaches are used. First, simulations are performed for a set of fixed sampled angles using adaptive mesh refinement, and then a simulation is performed where the turbine is slowly rotating to cover all the rotation angles continuously using a local mesh refinement strategy. The first approach is more reliable in terms of mesh refinement since a dual-based error control [20] is used. It is, however, time-consuming to wait for a fully developed state to start the measurement for each fixed angle. The second approach for the fixed turbine is also used to validate the rotating case.

2. Method

2.1. Vertical axis wind turbine

We consider a 3-bladed H-rotor turbine used in [21, 22, 18], with a radius of $r = 3.24\,\mathrm{m}$ and a blade length of $5\,\mathrm{m}$ (Fig. 1b). The blades are pitched 2 degrees outwards with a chord length of $0.25\,\mathrm{m}$ at the middle of the blade. Table 1 gives further details of the turbine.

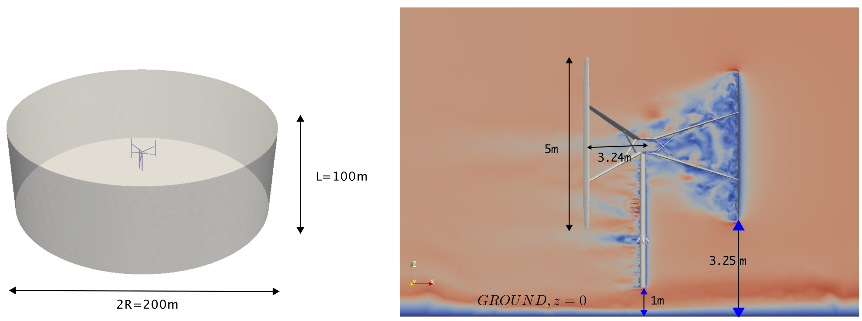
For simplification, we assume that the turbine axis is coincident with the $z-$axis and that the turbine $\Omega^{\mathcal{T}}$ is placed in a cylinder $\Omega^{\mathcal{C}}$ (Fig. 1a)

$$\Omega^{\mathcal{C}} = \left\{ (x, y, z) \in \boldsymbol{R}^3 \middle| x^2 + y^2 \leq R^2, z \in [0; L] \right\}. \tag{2}$$

We set $R = 100\mathrm{m}$ and $L = 100\mathrm{m}$, which is large compared to the turbine size to avoid artificial blockage effects, following the recommendations in [10]:

Table 1: The turbine characteristics

Hub height	6 m
Swept area	32 m^2
Blade airfoil	NACA0021
Tapering, linear	1 m (from tip)
Tip chord length	0.15 m
Mass of blade and support arms	35.79 kg



(a) (b)

Figure 1: A vertical axis turbine reproduced from [22] placed in a cylinder. It is 1m high from the cylinder bottom to model the ground effect.

- The minimum ratio between the distance from the turbine center to the domain inlet and outlet and the turbine diameter is 10. In our setting, the ratio is about 15.4.
- The minimum ratio between the domain width and the turbine diameter is 20. Since the ground is attached to the setting, it is still reasonable to reduce the ratio by half. In the current setting, it is about 15.4.
- The blockage ratio in this setting is about 0.16% which is smaller than 5% as recommended.

More detailed studies of the blockage effect can be found in [23, 10].

The turbine axis is placed in the center-line of the cylinder domain, and it is

1m high from the bottom of the cylinder to model the ground effect (Fig. 1b).

2.2. Wind profiles for simulations

We consider two wind profiles: a constant profile $U_x = -1\,\text{m/s}$, and a logarithmic profile (Fig. 2b) defined as follows,

$$U_x = \begin{cases} -\frac{1}{M}\log\Big((z - z_{min} + z_0)/z_0\Big) & \text{if } z - z_{min} < d \\ -1 & \text{otherwise} \end{cases} \tag{3}$$

with $M = \log\Big((d + z_0)/z_0\Big)$, $z_0 = 0.025\,\text{m}, z_{min} = 0$, and $d = 16.5\,\text{m}$ (double the turbine height).

Fig. 2a shows the wind directions. The blade to be measured is located at -41^{o} from North.

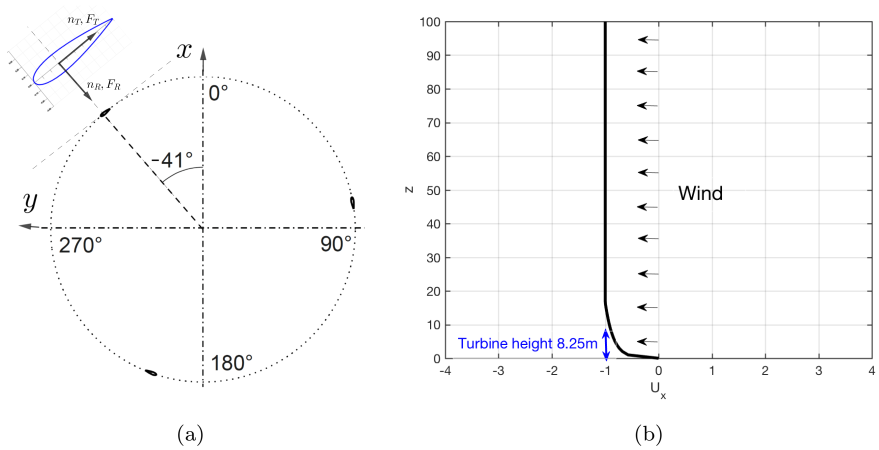


Figure 2: Wind directions in the relation to the blade to be measured located at -41^{o} from North in the image with the drawing of the turbine from above (a), and a logarithmic wind profile with $z_0 = 0.025\,\text{m}, z_{min} = 0, d = 16.5\,\text{m}$ (double the turbine height), $L = 100\,\text{m}$ (b).

2.3. Measurements

A total force F_{tot} acting on a blade can be decomposed into the radial force F_R and the tangential force F_T (see Fig. 2a), i.e.

$$F_{tot} = \sqrt{F_R^2 + F_T^2}.$$

One can compute the corresponding force coefficient C as

$$C = \frac{2\,F}{\rho\, U_\infty^2\, A} \tag{4}$$

where $\rho = 1.25\,\mathrm{kg/m^3}$ is the air density, $A = 1.15\,\mathrm{m^2}$ is the reference area, and U_∞ is the average of the wind speed acting on the blades

$$U_\infty = \frac{1}{5}\int_{3.25}^{8.25} U_x\, dz.$$

From now on, C_R, C_T, and C_{tot} are used to denote the force coefficients corresponding to F_R, F_T, and F_{tot}.

For the logarithmic profile defined by Eq. (3) we have $U_\infty = -0.83\,\mathrm{m/s}$ which is close to the inflow velocity at the middle of the blades, i.e $U_x(z = 5.75\mathrm{m}) \approx -0.84\mathrm{m/s}$.

The forces are computed as follows

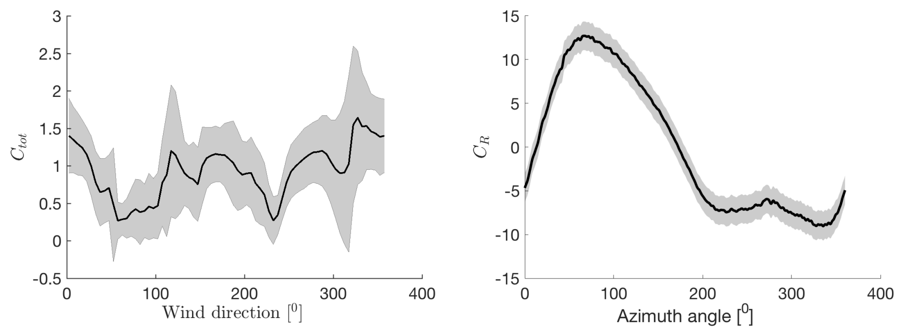
$$F_R = \int_{\partial\Omega^\mathcal{T}} (\sigma n \cdot n_R)\, n_R\, ds, \quad F_T = \int_{\partial\Omega^\mathcal{T}} (\sigma n \cdot n_T)\, n_T\, ds$$

where $\sigma = 2\nu\epsilon(u) - p\mathcal{I}$ is the stress tensor, $\epsilon(u) = \frac{1}{2}(\nabla u + \nabla u^T)$ the strain rate tensor, n is the normal vector pointing outward on the turbine surface, n_R is the normal vector pointing inward the turbine center, and n_T is the tangential vector which is perpendicular to n_R (see Fig. 2a).

The force coefficients have been experimentally measured for one blade with its two supporting arms for the parked condition [18] and rotating condition [19] (see Fig. 3). The black curves represent the mean value and the shaded regions represent the measurement error which is the standard deviation for the parked condition and maximum measurement error for the rotating condition.

The difference between the simulation force coefficient C^s and the experiment force coefficient C^e is measured by the root mean square error (RMSE) which is defined as

$$\mathrm{RMSE}(C^e, C^s) = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(C_k^e - C_k^s)^2} \tag{5}$$



(a) (b)

Figure 3: Experimental measurements on one blade with its two supporting arms in the parked condition (a) and in the rotating condition (b). The black curves represent the mean value and the shaded regions represent measurement errors which is the standard deviation for the parked condition [18] and maximum measurement error for the rotating condition [19].

where N is the number of discretized angles of the azimuth angle interval $[0°, 360°]$ of the experimental force measurement which is 72 for the parked case and 1830 for the rotating case. The curves representing the simulation forces are interpolated between these discretized angles using the 1D data interpolation `interp1` in MATLAB.

2.4. Navier-Stokes equations

The airflow around the VAWT is modeled by the Navier-Stokes equations. For incompressible flow, the equations read

$$\begin{cases} \dot{u} + (u \cdot \nabla)u - \nu \Delta u + \nabla p = f, & \text{in } \Omega \times I, \\ \nabla \cdot u = 0, & \text{in } \Omega \times I, \\ u(\cdot, 0) = u_0, & \text{in } \Omega, \end{cases} \tag{6}$$

where u is the velocity, p pressure and f a given body force. $\Omega \subset \mathbb{R}^3$ is a spatial domain with boundary Γ, and $I = [0, T]$ a time interval.

For a moving or deforming domain, we may use an Arbitrary Lagrangian-Eulerian (ALE) method [24], which is based on the introduction of a separate set of reference coordinates. Often we let these reference coordinates trace the deformation of the finite element mesh, described by the mesh velocity β. In an ALE method, the convection term is modified to take the mesh velocity into account, which gives the modified Navier-Stokes equations on ALE form,

$$\begin{cases} \dot{u} + \Big((u-\beta)\cdot\nabla\Big)u - \nu\Delta u + \nabla p = f, & \text{in } \Omega\times I, \\ \nabla\cdot u = 0, & \text{in } \Omega\times I, \\ u(\cdot,0) = u_0, & \text{in } \Omega. \end{cases} \tag{7}$$

2.5. Finite element discretization

A Galerkin least-squares space-time finite element method (GLS) [20] is used to discretize the flow around the VAWT. Let $0 = t_0 < t_1 < \cdots < t_N = T$ be a time partition associated with the time intervals $I_n = (t^{n-1}, t^n]$ of length $k_n = t^n - t^{n-1}$. We denote the finite element space of continuous piecewise linear functions by Q_h, with the derived spaces $Q_{h,0} = \{q \in Q_h : q(x) = 0,\ x \in \Gamma\}$ and $V_h = [Q_{h,0}]^3$. The DFS-ALE method with least-squares stabilization is stated as: For all time intervals I_n, find $(U_h^n, P_h^n) \in V_h \times Q_h$ such that

$$\begin{aligned} \Big(\frac{U_h^n - U_h^{n-1}}{k_n} &+ ((\bar{U}_h^n - \beta_h)\cdot\nabla)\bar{U}_h^n, v_h\Big) \\ &+ \Big(\nu\nabla\bar{U}_h^n, \nabla v_h\Big) - \Big(P_h^n, \nabla\cdot v_h\Big) + \Big(\nabla\cdot\bar{U}_h^n, q_h\Big) \\ &\qquad + SD_\delta\Big(\bar{U}_h^n, P_h^n; v_h, q_h\Big) = \big(f, v_h\big) \end{aligned}$$

for all test functions $(v_h, q_h) \in V_h \times Q_h$, where $\bar{U}_h^n = \frac{U_h^n + U_h^{n-1}}{2}$, and (U_h^n, P_h^n) is a numerical approximation of (u,p) at $t = t_n$, and with stabilization term

$$\begin{aligned} SD_\delta^n(\bar{U}_h^n, P_h^n; v, q) := &\left(\delta_1 \Big(\Big(\bar{U}_h^n - \beta_h\Big)\cdot\nabla\bar{U}_h^n + \nabla P_h^n - f_n\Big),\ \Big(\bar{U}_h^n - \beta_h\Big)\cdot\nabla v_h + \nabla q_h\right) \\ &+ \big(\delta_2\,\nabla\cdot\bar{U}_h^n,\ \nabla\cdot v_h\big). \end{aligned}$$

Here δ_1 and δ_2 are given stabilization parameters:

$$\delta_1 = C_1\Big(k_n^{-2} + |U_h^{n-1} - \beta_h|^2 h_n^{-2}\Big)^{-1/2}, \quad \delta_2 = C_2\,|U_h^{n-1}|\,h_n.$$

We note that under a CFL condition, i.e $k_n = \frac{h_n}{|U_h^{n-1} - \beta_h|}$, δ_1 is simplified to $\delta_1 = C_1\frac{h_n}{|U_h^{n-1} - \beta_h|}$.

2.6. Operational mesh velocity

We model a rotating VAWT by rotating the complete computational domain $\Omega^{\mathcal{C}}$ with an operational mesh velocity $\beta_h = (\dot{x}, \dot{y}, \dot{z})$, which can be explicitly computed for a given angular velocity ω, and time-step size k:

$$\begin{cases} \dot{x} &= \dfrac{1}{k}\Big(x\,(\cos(\omega) - 1) - y\,\sin(\omega) \Big), \\ \dot{y} &= \dfrac{1}{k}\Big(y\,(\cos(\omega) - 1) + x\,\sin(\omega) \Big), \\ \dot{z} &= 0. \end{cases} \tag{8}$$

2.7. Boundary layer model

Since it is not feasible to resolve a turbulent boundary layer with current computational resources, appropriate boundary conditions need to be chosen to model the effect of turbulent boundary layers without full resolution. Here we use a slip with friction and penetration with resistance boundary condition [25]:

$$\begin{aligned} u \cdot n + \alpha\, n^T \sigma n &= 0, \\ u \cdot \tau_k + \zeta^{-1}\, n^T \sigma\, \tau_k &= 0, \end{aligned} \tag{9}$$

where n and τ_k are normal and tangential vectors ($k = 1, 2$) respectively, α is a penetration parameter, ζ is a skin friction parameter, and σ is the stress tensor. The no-slip condition $u = 0$ corresponds to $(\alpha, \zeta) \to (0, \infty)$, and the free slip condition $u \cdot n = 0$ corresponds to $(\alpha, \zeta) \to (0, 0)$.

In this paper we will use the approximation of a free slip boundary condition, which is shown in [26] to be a good approximation at high Reynolds numbers.

2.8. Adaptive mesh refinement

Adaptive mesh refinement algorithms provide a method for efficient use of computational resources, based on a posteriori error estimates that approximate the local contribution to the global error of all the cells in the mesh. The localization of the global error is based on the solution of an adjoint problem to the Navier-Stokes equations. The adjoint problem takes the form of a system of convection-diffusion-reaction equations that runs backward in time, linearized at the exact velocity u and its numerical approximation U_h [20]:

$$\begin{cases} -\dot{\varphi} + \Big((u-\beta)\cdot\nabla\Big)\varphi + \nabla U_h^T \cdot \varphi + \nabla\theta = \Psi, & \text{in } \Omega \times I, \\ \nabla \cdot \varphi = 0, & \text{in } \Omega \times I, \\ \varphi(\cdot, T) = 0, & \text{in } \Omega, \end{cases} \tag{10}$$

where $(\nabla U_h^T \cdot \varphi)_j = (U_h), j \cdot \varphi$, and Ψ is a given weight function used to define a quantity of interest,

$$M(\hat{u}) = (\hat{u}, \Psi),$$

with of $\hat{u} = (u, p)$.

Let $\hat{U} = (U_h, P_h)$, from a standard analysis [13], it follows that

$$|M(\hat{u}) - M(\hat{U})| = |(R(\hat{U}), \hat{\varphi})| = |\sum_{K \in \mathcal{T}_n} (R(\hat{U}), \hat{\varphi})_K|, \tag{11}$$

where $R(\hat{U})$ is the residual of the Navier-Stokes equations (Eq. 6):

$$\begin{aligned} R(\hat{U}) &= \Big(R_1(\hat{U}), R_2(\hat{U})\Big), \\ R_1(\hat{U}) &= \dot{U_h} + \Big((U_h - \beta)\cdot\nabla\Big)U_h - \nu\Delta U_h + \nabla P_h - f, \\ R_2(\hat{U}) &= \nabla \cdot U_h, \end{aligned} \tag{12}$$

$\hat{\varphi} = (\varphi, \theta)$ is the solution to the adjoint problem, and $(\cdot,\cdot)_K$ is the local $L_2(K)$ inner product,

$$(v, w)_K = \int_K v \cdot w \; dx. \tag{13}$$

The local error indicator is then defined for each element K, as

$$e^K = (R(\hat{U}), \hat{\varphi}_h)_K, \tag{14}$$

where $\hat{\varphi}_h$ is the numerical solution of the adjoint problem.

2.9. Boundary conditions

We choose the boundary conditions as follows:

- Inflow: $u = (U_x, 0, 0)$ on $\Gamma_{in} = \Big\{(x, y, z)\Big| x > 0 \wedge x^2 + y^2 = R^2\Big\}$

- Outflow: $p = 0$ on $\Gamma_{out} = \left\{(x, y, z)\middle| x < 0 \wedge x^2 + y^2 = R^2\right\}$
- No-slip boundary condition $u = 0$ to model the ground at the bottom of the computational domain $z = z_{min}$.
- The free slip boundary condition $u \cdot n = 0$ is prescribed for the turbine surface and on top of the computatinal domain $z = z_{min} + L$.

2.10. Implementation

The numerical method has been implemented in the Unicorn solver [12, 27, 28] in the FEniCS-HPC platform [16, 17] which is a high performance computing branch of FEniCS [29, 30]. This branch is optimized for massively parallel architectures, and implements duality-based adaptive error control, implicit parameter-free turbulence modeling by the use of stabilized FEM and shows strong linear scaling up to thousands of cores [12, 31, 32, 27, 33].

3. Results

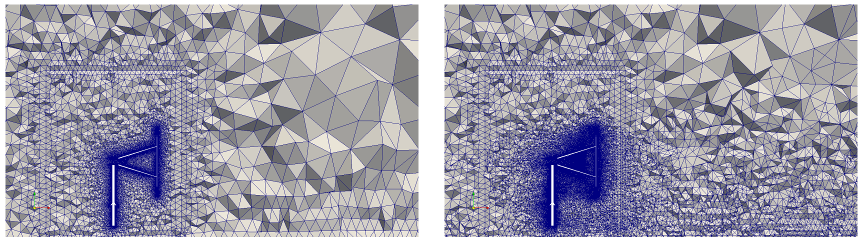
3.1. Parked case

We first consider an angle-wise validation with the automated mesh adaptivity. Simulations are performed for a set of 19 angles between 0 and 360 degrees (see Fig. 2a). For each angle, the automated mesh adaptivity is used with an initial mesh of 524,686 vertices (2,887,293 tetrahedrons). The error indicator (Eq. 14) is computed element-wise and the mesh is refined locally by chosing 10% of the elements for refinement based on the absolute values of the error indicator.

The effect of the automated mesh adaptivity is visualized in Fig. 4 showing the difference between the initial mesh, and the mesh after three iterations with 1,751,236 vertices (9,520,104 tetrahedrons).

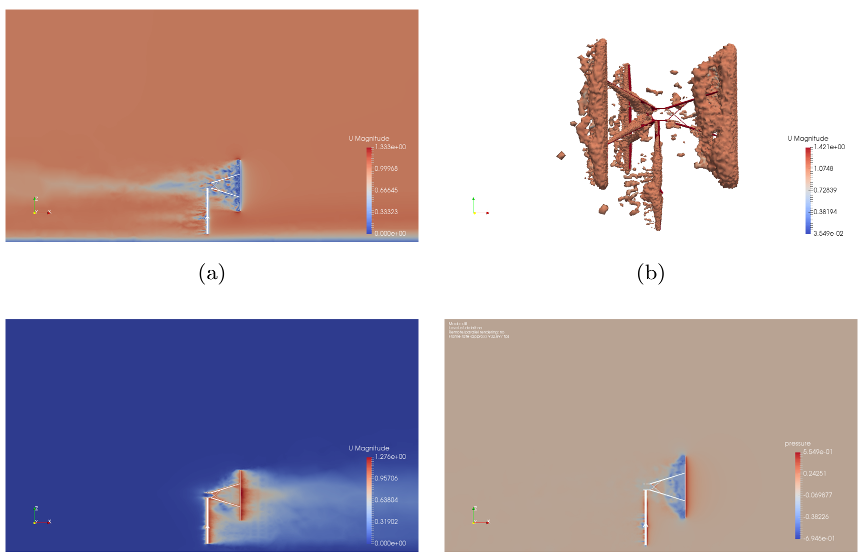
Fig. 5 shows numerical approximations of the primal problem by solving Eq. (6) and the dual problem by solving Eq. (10).

Fig. 6 shows the wake formed downstream the turbine.



(a) (b)

Figure 4: Automated mesh adaptivity: (a) initial mesh with 524,686 vertices (2,887,293 tetrahedrons), and (b) after three iterations with 1,751,236 vertices (9,520,104 tetrahedrons).



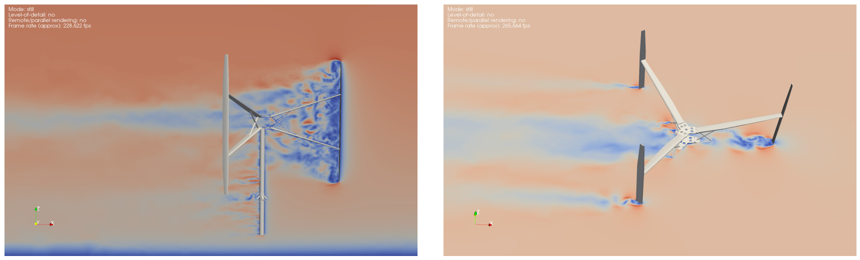
(a) (b)

(c) (d)

Figure 5: Numerical solutions: primal velocity (a, b), dual velocity (c) and pressure (d).

To validate against the experimental results, the simulations were performed for the time interval $I = [0, 200]$ s and the dynamic force coefficient $C_{tot}^{dyn}(t)$ was measured in the time interval $I_m = [100, 200]$ s when the flow is fully developed, i.e.

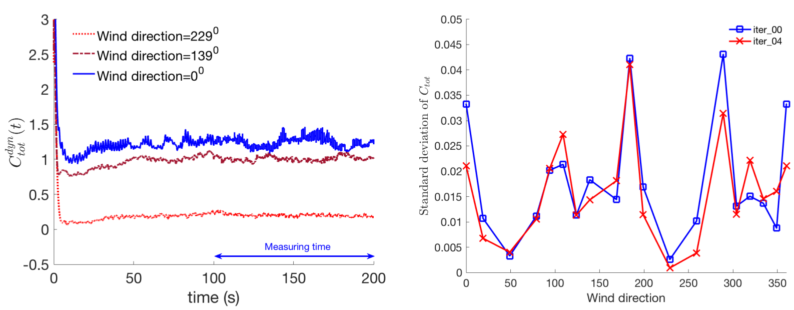
$$C_{tot} = \frac{\sum_{t=100}^{200} C_{tot}^{dyn}(t)}{100}$$



(a) (b)

Figure 6: Vortices on different scales formed in the wake downstream the turbine.

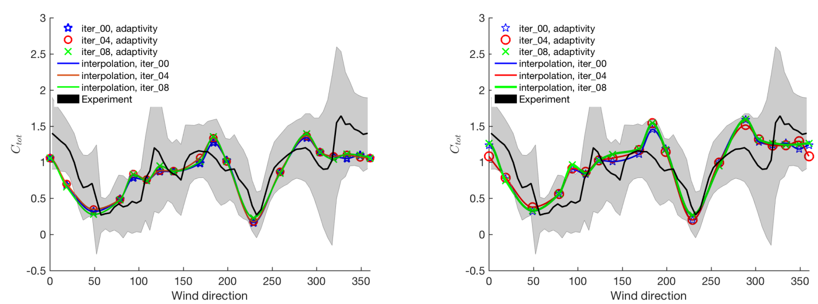
Fig. 7a shows $C_{tot}^{dyn}(t)$ versus the simulation time on the initial mesh for three wind directions $0°, 139°$, and $229°$. During the time $C_{tot}^{dyn}(t)$ is measured, the flow is fully developed. Fig. 7b shows the standard deviations of $C_{tot}^{dyn}(t)$ over the measurement time interval I_m for 19 discretized angles and two levels of mesh refinement. Since $C_{tot}^{dyn}(t)$ is stable, the standard deviations are quite small.



(a) (b)

Figure 7: Measuring force coefficients C_{tot} in the simulations: (a) $C_{tot}^{dyn}(t)$ versus the simulation time interval $I = [0, 200]$ s for three wind directions, and (b) the standard deviations of the measurement in the measurement time interval $I_m = [100, 200]$ s for 19 discretized angles and two levels of mesh refinement. Since $C_{tot}^{dyn}(t)$ is stable, the standard deviations are quite small.

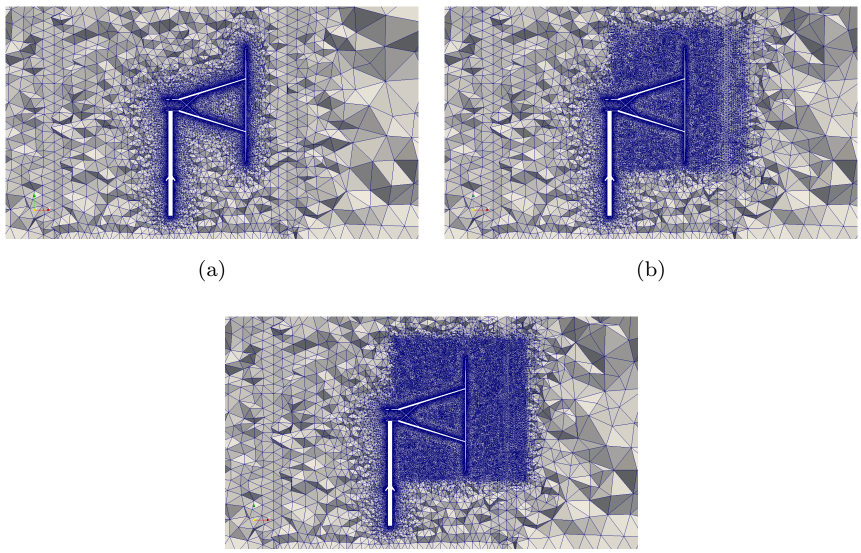
The comparison against experimental data is shown in Fig. 8, for the constant and logarithmic wind profiles. The markers represent the simulated force coefficients for different levels of mesh refinement, the colored curves represent the interpolations of simulated force coefficients, and the black curve represents the experimental data with the standard deviation represented by the shaded region. The two different wind profiles generate slightly different force coefficients. For example, near 0°, the log profile appears to approximate the experiment somewhat better than the constant profile does, but for angles near 180° and 270°, the situation is the opposite. For both inflow profiles, the simulations capture the general shape of the experimental force coefficient curve, although the oscillations of the experimental force between 120° and 200° are not seen clearly which could be an effect of the interpolation. The RMSE is about 0.26 for the both wind profiles and the difference is still within the standard deviation of the experimental measurement (shaded region). It is also found that the results are stable under different levels of mesh refinement.



(a) (b)

Figure 8: Angle-wise force coefficients and their interpolations for three iterations of the adaptive mesh refinement algorithm against experimental data: (a) constant wind profile with RMSE of 0.26, (b) logarithmic wind profile with RMSE of 0.27.

Next, C_{tot} is continuously measured by slowly rotating the computational domain around the $z-$axis with rotational speed of $\omega = -\pi/180\,\mathrm{rad/s}$. The mesh is locally refined in a region containing the blade with its two supporting arms (Fig. 9). We also start with the same initial mesh as above. After 4 iterations the mesh has 885,842 vertices (4,863,350 tetrahedrons). After 8 iterations of refinement, the mesh has 1,136,739 vertices (6,254,266 tetrahedrons). The slow rotation takes place after the flow is fully developed, i.e $t > 100\,\mathrm{s}$.



(a) (b)

(c)

Figure 9: Local mesh refinement: (a) initial mesh with 524,686 vertices (2,887,293 tetrahedrons), (b) after 4 iterations with 885,842 vertices (4,863,350 tetrahedrons), and (c) after 8 iterations with 1,136,739 vertices (6,254,266 tetrahedrons).

The local refinement works well for this purpose and the shape of the experimental force curves is well captured (Fig. 10). Here we compare radial coefficient C_R, tangential coefficient C_T, and the total coefficient C_{tot}. Different colored curves here represent different levels of mesh refinement, and the curves are almost on top of each other which shows that the results are stable under mesh refinement. This method gives a better approximation to the experiment than the angle-wise validation does. Especially, the oscillations between 120° and 200° are stronger and the peak between 300° and 360° is more clearly seen which could be since now force coefficients are computed for all angles, without any interpolation. In general, the two inflow profiles give similar RMSE which is about 0.30 and 0.24 respectively, although the RMSE for C_T is 0.16 and 0.21 corresponding to the constant profile and the logarithmic profile.

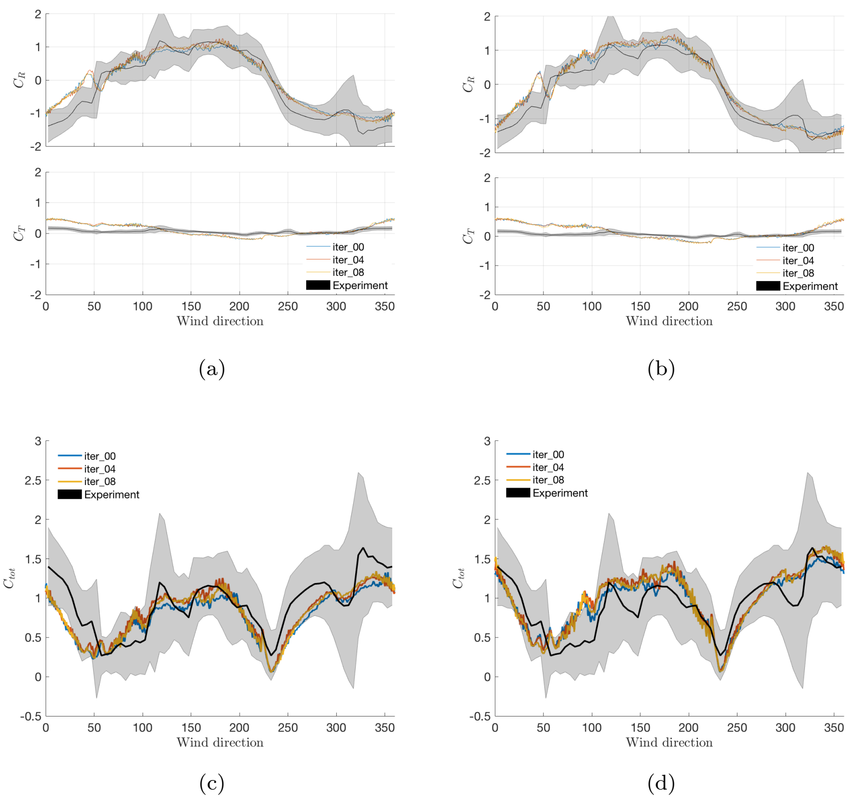


Figure 10: Force coefficient curves versus wind directions for the constant wind profile (a, c) and the logarithmic wind profile (b, d). The RMSE of C_R and C_{tot} is quite similar for both wind profiles which is about 0.30 and 0.24 respectively, although the RMSE for C_T is 0.16 and 0.21 corresponding to the constant profile and the logarithmic profile.

3.2. Rotating case

In this section, we perform simulations to validate the radial force coefficient C_R against the experimental measurement published in [19] for the tip speed ratio of $\lambda = 3.44$. Here we recall that

$$\lambda = \frac{\omega\, r}{U_\infty}$$

where $\omega = 64.81\,\mathrm{rpm} = 6.79\,\mathrm{rad/s}$ is the rotor rational speed, $r = 3.24\,\mathrm{m}$ is the rotor radius, and $U_\infty = 6.39\,\mathrm{m/s}$. The local mesh refinement is used with three levels of refinement as in the previous section. For each refinement level, we compute the radial forces for several revolutions until the wake has stabilized. It

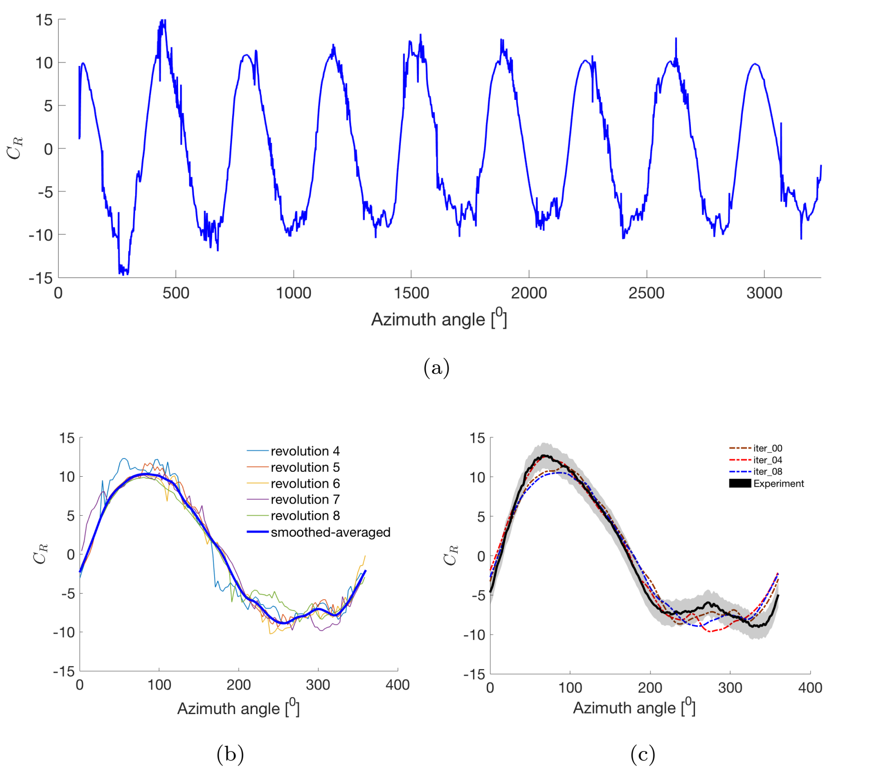


Figure 11: Radial force coefficients C_R computed by DFS-ALE with the logarithmic profile on the initial mesh with 8 revolutions (a) and averaged over the 5 last revolutions (b). The smoothed-averaged curves for 3 levels of mesh refinement are compared to the experimental result (c). The experimental curve is well captured with the RMSE of 1.5, 1.4 and 1.3 respectively for three refinement levels. Here the RMSE was calculated with $N = 1830$ data points from the experimental data.

can be seen from the simulations that the first revolution is not reliable and one can start taking the average after the second evolution when the flow becomes stabilized. Fig. 11a shows C_R on the initial mesh with 8 revolutions. Since the wake has not stabilized yet, the force for the first revolution is higher than the others. Fig. 11b shows the smoothed-averaged C_R over the 4 last revolutions on the initial mesh. Fig. 11c shows the smoothed-averaged C_R on three levels of mesh refinement compared to the experiment data. The experimental curve is well captured with the RMSE of 1.5, 1.4 and 1.3 respectively for the three refinement levels. More importantly, the simulation curves lie within the shaded region representing the measurement error, except they are slightly out at two peaks.

4. Conclusions and future work

The paper presents a DFS-ALE method for simulation of the turbulent flow past a VAWT, which is validated by experimental measurements. The method consists of a Galerkin least-squares finite element method, coupled with an arbitrary Lagrangian-Eulerian method in order to allow us to compute high Reynolds number turbulent flow on moving meshes. The results are stable with respect to the mesh refinement and the general shape of the force curve is well captured. More importantly, the simulation curves lie within the shaded region representing the measurement error, except they are slightly out at two peaks.

The slow-rotation strategy is more efficient than the angle-wise strategy in terms of computational time, since the force coefficients are valid only when the flow is fully developed. The waiting time to reach developed flow in the second case needs to be represented for each angle, whereas in the first case it is done only once. The manual local refinement is, however, less reliable than the automated mesh adaptation used in the angle-wise strategy. We also note that in the angle-wise strategy some important local extrema can be missed since we only sample a subset of angles and interpolate in between.

There is also a good agreement in radial forces between the simulation and the experiment for the rotating conditions. Although the local mesh refinement works well, an efficient automated mesh adaptation is worth considering.

In the future we will extend our study to passively rotating turbines which rotate due to the incoming wind. The bending and vibration in the blades are important phenomena, and may be predicted with a fluid-structure interaction model of the VAWT. A full fluid-structure interaction model with slip boundary conditions at the internal interfaces is under development.

Acknowledgement

This research has been supported by the European Research Council, the Swedish Energy Agency, Standup for Energy, the Basque Excellence Research Center (BERC 2014-2017) program by the Basque Government, the Spanish Ministry of Economy and Competitiveness MINECO: BCAM Severo Ochoa accreditation SEV-2013-0323, the ICERMAR ELKARTEK project of the Basque Government, the Projects of the Spanish Ministry of Economy and Competitiveness with reference MTM2013-40824-P and MTM2016-76016-R. We acknowledge the Swedish National Infrastructure for Computing (SNIC) at PDC – Center for High-Performance Computing for awarding us access to the supercomputer resource Beskow. The initial volume mesh was generated with ANSA from Beta-CAE Systems S. A., who generously provided an academic license for this project.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] M. Islam, S. Mekhilef, R. Saidur, Progress and recent trends of wind energy technology, Renewable and Sustainable Energy Reviews 21 (2013) 456 – 468. doi:http://dx.doi.org/10.1016/j.rser.2013.01.007.

URL http://www.sciencedirect.com/science/article/pii/S1364032113000312

[2] A. Tummala, R. K. Velamati, D. K. Sinha, V. Indraja, V. H. Krishna, A review on small scale wind turbines, Renewable and Sustainable Energy Reviews 56 (2016) 1351 – 1371. doi:http://dx.doi.org/10.1016/j.rser.2015.12.027.
URL http://www.sciencedirect.com/science/article/pii/S1364032115014100

[3] J. Pitteloud, S. Gsänger, Small wind world report summary (2016).

[4] B. Blocken, 50 years of computational wind engineering: Past, present and future, Journal of Wind Engineering and Industrial Aerodynamics 129 (2014) 69 – 102. doi:http://dx.doi.org/10.1016/j.jweia.2014.03.008.
URL http://www.sciencedirect.com/science/article/pii/S016761051400052X

[5] M. Casini, Small vertical axis wind turbines for energy efficiency of buildings, Journal of Clean Energy Technologies 4 (1).

[6] C. Li, S. Zhu, Y. lin Xu, Y. Xiao, 2.5d large eddy simulation of vertical axis wind turbine in consideration of high angle of attack flow, Renewable Energy 51 (2013) 317 – 330. doi:http://dx.doi.org/10.1016/j.renene.2012.09.011.
URL http://www.sciencedirect.com/science/article/pii/S0960148112005770

[7] S. Shamsoddin, F. Port-Agel, Large eddy simulation of vertical axis wind turbine wakes, Energies 7 (2) (2014) 890–912. doi:10.3390/en7020890.
URL http://www.mdpi.com/1996-1073/7/2/890

[8] S. Shamsoddin, F. Port-Agel, A large-eddy simulation study of vertical axis wind turbine wakes in the atmospheric boundary layer, Energies 9 (5).

doi:10.3390/en9050366.
URL http://www.mdpi.com/1996-1073/9/5/366

[9] M. Nini, V. Motta, G. Bindolino, A. Guardone, Three-dimensional simulation of a complete vertical axis wind turbine using overlapping grids, Journal of Computational and Applied Mathematics 270 (2014) 78 – 87, fourth International Conference on Finite Element Methods in Engineering and Sciences (FEMTEC 2013). doi:http://dx.doi.org/10.1016/j.cam.2014.02.020.
URL http://www.sciencedirect.com/science/article/pii/S037704271400106X

[10] A. Rezaeiha, I. Kalkman, B. Blocken, Cfd simulation of a vertical axis wind turbine operating at a moderate tip speed ratio: Guidelines for minimum domain size and azimuthal increment, Renewable Energy 107 (2017) 373 – 385. doi:http://dx.doi.org/10.1016/j.renene.2017.02.006.
URL http://www.sciencedirect.com/science/article/pii/S0960148117300848

[11] S. H. Hezaveh, E. Bou-Zeid, M. W. Lohry, L. Martinelli, Simulation and wake analysis of a single vertical axis wind turbine, Wind Energy 20 (4) (2017) 713–730, wE-15-0121.R3. doi:10.1002/we.2056.
URL http://dx.doi.org/10.1002/we.2056

[12] J. Hoffman, J. Jansson, R. V. de Abreu, N. C. Degirmenci, N. Jansson, K. Muller, M. Nazarov, J. H. Spuhler, Unicorn: Parallel adaptive finite element simulation of turbulent flow and fluidstructure interaction for deforming domains and complex geometry, Computers & Fluids 80 (2013) 310 – 319, selected contributions of the 23rd International Conference on Parallel Fluid Dynamics ParCFD2011. doi:http://dx.doi.org/10.1016/j.compfluid.2012.02.003.
URL http://www.sciencedirect.com/science/article/pii/S0045793012000473

[13] J. Hoffman, J. Jansson, N. Jansson, R. V. De Abreu, C. Johnson, Computability and Adaptivity in CFD, John Wiley & Sons, Ltd, 2017. doi:10.1002/9781119176817.ecm2057.
URL http://dx.doi.org/10.1002/9781119176817.ecm2057

[14] J. Hoffman, J. Jansson, N. Jansson, FEniCS-HPC: Automated Predictive High-Performance Finite Element Computing with Applications in Aerodynamics, Springer International Publishing, Cham, 2016, pp. 356–365. doi:10.1007/978-3-319-32149-3_34.
URL http://dx.doi.org/10.1007/978-3-319-32149-3_34

[15] A. Korobenko, M.-C. Hsu, I. Akkerman, Y. Bazilevs, Aerodynamic simulation of vertical-axis wind turbines, Journal of Applied Mechanics 81 (2) (2013) 021011–021011–6. doi:10.1115/1.4024415.
URL http://dx.doi.org/10.1115/1.4024415

[16] FEniCS-HPC, Fenics-hpc, http://www.fenics-hpc.org.

[17] N. Jansson, J. Jansson, J. Hoffman, Framework for massively parallel adaptive finite element computational fluid dynamics on tetrahedral meshes, SIAM Journal on Scientific Computing 34 (1) (2012) C24–C41.

[18] A. Goude, M. Rossander, Force measurements on a vawt blade in parked conditions, Energies 10 (12) (2017) 1954. doi:10.3390/en10121954.
URL http://dx.doi.org/10.3390/en10121954

[19] E. Dyachuk, A. Goude, Numerical validation of a vortex model against experimentaldata on a straight-bladed vertical axis wind turbine, Energies 8 (10) (2015) 1180011820. doi:10.3390/en81011800.
URL http://dx.doi.org/10.3390/en81011800

[20] J. Hoffman, C. Johnson, Stability of the dual Navier-Stokes equations and efficient computation of mean output in turbulent flow using adaptive DNS/LES, Computer Methods in Applied Mechanics and Engineering 195 (2006) 1709–1721. doi:10.1016/j.cma.2005.05.034.

[21] M. Rossander, E. Dyachuk, S. Apelfrjd, K. Trolin, A. Goude, H. Bernhoff, S. Eriksson, Evaluation of a blade force measurement system for a vertical axis wind turbine using load cells, Energies 8 (12) (2015) 59735996. doi:10.3390/en8065973.
URL http://dx.doi.org/10.3390/en8065973

[22] E. Dyachuk, A. Goude, Numerical validation of a vortex model against experimentaldata on a straight-bladed vertical axis wind turbine, Energies 8 (10) (2015) 11800–11820. doi:10.3390/en81011800.
URL http://www.mdpi.com/1996-1073/8/10/11800

[23] A. Goude, O. gren, Simulations of a vertical axis turbine in a channel, Renewable Energy 63 (2014) 477 – 485. doi:https://doi.org/10.1016/j.renene.2013.09.038.
URL http://www.sciencedirect.com/science/article/pii/S096014811300517X

[24] J. Donea, A. Huerta, J.-P. Ponthot, et al., Arbitrary lagrangian eulerian methods, Encyclopedia of Computational Mechanics.

[25] J. Hoffman, C. Johnson, Computational turbulent incompressible flow, Applied mathematics body and soul, Springer, Berlin, 2007.
URL http://cds.cern.ch/record/1644577

[26] J. Hoffman, J. Jansson, N. Jansson, R. V. D. Abreu, Towards a parameter-free method for high reynolds number turbulent flow simulation based on adaptive finite element approximation, Computer Methods in Applied Mechanics and Engineering 288 (Supplement C) (2015) 60 – 74, error Estimation and Adaptivity for Nonlinear and Time-Dependent Problems. doi:https://doi.org/10.1016/j.cma.2014.12.004.
URL http://www.sciencedirect.com/science/article/pii/S0045782514004836

[27] J. Hoffman, J. Jansson, N. Jansson, M. Nazarov, Unicorn: A unified continuum mechanics solver, in: Automated Solutions of Differential Equations

by the Finite Element Method, Springer, 2011.
URL http://www.fenicsproject.org/pub/documents/book/

[28] V. D. Nguyen, J. Jansson, M. Leoni, B. Janssen, A. Goude, J. Hoffman, Modelling of rotating vertical axis turbines using a multiphase finite element method, in: MARINE 2017 : Computational Methods in Marine Engineering VII15 - 17 May 2017, Nantes, France, 2017, pp. 950–960, qC 20170629.
URL http://congress.cimne.com/marine2017/frontal/Doc/Ebookmarine.pdf

[29] A. Logg, K.-A. Mardal, G. N. Wells, Automated solution of differential equations by the finite element method : the FEniCS book, Springer Verlag, 2012, xIII, 723 s. : ill.

[30] FEniCS, Fenics project, http://www.fenicsproject.org.

[31] N. Jansson, J. Hoffman, J. Jansson, Framework for Massively Parallel Adaptive Finite Element Computational Fluid Dynamics on Tetrahedral Meshes, SIAM J. Sci. Comput. 34 (1) (2012) C24–C41.

[32] R. C. Kirby, FIAT: Numerical Construction of Finite Element Basis Functions,, Springer, 2012, Ch. 13.

[33] J. Hoffman, J. Jansson, N. Jansson, C. Johnson, R. V. de Abreu, Turbulent flow and fluid-structure interaction, in: Automated Solutions of Differential Equations by the Finite Element Method, Springer, 2011.
URL http://www.fenicsproject.org/pub/documents/book/