# Direct finite element simulation of turbulent flow for marine based renewable energy

Johan Jansson[b,a], Van Dang Nguyen[a], Margarida Moragues Ginard[b], Daniel Castanon Quiroz[b], Laura Saavedra[c], Ezhilmathi Krishnasamy[b], Anders Goude[d], Johan Hoffman[a,b,*]

[a] *KTH Royal Institute of Technology, Stockholm, Sweden*
[b] *Basque Center for Applied Mathematics, Bilbao, Spain*
[c] *Universidad Politecnica de Madrid, Madrid, Spain*
[d] *Uppsala University, Uppsala, Sweden*

**Abstract**

In this article we present a computational framework for simulation of turbulent flow in marine based renewable energy applications. In particular, we focus on floating structures and rotating turbines. This work is an extension to multiphase turbulent flow, of our existing framework of residual based turbulence modeling for single phase turbulent incompressible flow. We illustrate the framework in four examples: a regular wave test where we compare against an exact solution, the standard MARIN wave impact benchmark with experimental validation data, a vertical axis turbine with complex geometry from an existing turbine, and finally a prototype simulation of decay test in a coupled moving boundary rigid-body and two-phase fluid simulation.

*Keywords:* direct finite element simulation, marine based renewable energy, multiphase flow
*2010 MSC:* 65M60, 76F65

## 1. Introduction

Simulation of structures for marine based renewable energy presents particular challenges; turbulent flow in water and air, breaking waves, rotating turbines, fluid-structure interaction, aero- and hydroacoustics. For example, a floating wind turbine experiences forces from gravity and boyancy, ocean currents and wind loads, and impact forces from ocean waves. The Reynolds number is high, and thus the flow is turbulent, and a rotating turbine exhibits rapidly changing flow separation patterns, including dynamic stall phenomena.

*Corresponding author
*Email address:* jhoffman@kth.se (Johan Hoffman)
*URL:* www.csc.kth.se/~jhoffman (Johan Hoffman)

In this article we focus on the computational fluid dynamics (CFD) aspects of the problem, and leave the acoustics to future work. Full resolution of all turbulent scales in a direct numerical simulation (DNS) is not feasible due to the high computational cost. Instead various methods based on averaging have been proposed, e.g. a statistical ensemble average (RANS) or a spatial filter (LES). In these methods the finest turbulent scales of the problem are modeled in a turbulence model (RANS) or a subgrid model (LES), see e.g. [1] for an overview of different approaches.

Recently a new type of turbulence model have been developed based on the residual of the equations in weak form [2, 3], equivalent to stabilized finite element methods for the Navier-Stokes equations. Residual-based numerical stabilization is consistent, so that an exact solution of the Navier-Stokes equations satisfies also the stabilized finite element formulation of the equations. Similarly, with a residual-based subgrid model it is possible to directly connect the finite element approximation to weak solutions of the Navier-Stokes equations, which provides a mathematical framework for computational approximation of turbulent flow, in the form of functionals of weak solutions, without introducing LES filtering or RANS statistical averaging. Combined with duality based a posteriori estimation of the error in such functionals of weak solutions, adaptive methods for turbulent flow have been developed under the notions of *Adaptive DNS/LES*, *General Galerkin (G2)*, and *Direct Finite Element Simulation (DFS)*, see e.g. [2, 4, 5]. The smallest scale in the method is the mesh scale, which is not determined a priori, but is computed as part of the method.

For high Reynolds number flow, turbulent dissipation from the residual based turbulence model will dominate the viscous dissipation of the Navier-Stokes equations. Assuming also the skin friction to be small at solid boundaries we may use a slip boundary condition to model turbulent boundary layers, which leads a method which is essentially parameter-free [6].

In the context of marine based renewable energy, we want to treat the wind and ocean in the same model, including breaking waves and fluid-structure interaction. To this end we formulate a model based on the variable density Navier-Stokes equations, where the ocean surface is implicitly represented as a level curve of the density field. This implicit representation of the ocean surface allows e.g. for direct simulation of breaking waves. To avoid excessive smearing of the density field we include a phase separation term in the formulation of the method.

There is a vast literature on numerical methods for multiphase and variable density flow, including also finite element methods. Our approach is similar to a level set approach but without a distance function, as in e.g. [7], but directly using the density field, similar to [8]. The volume-of-fluid (VOF) method is also related, where an interface marker is explicitly represented and convected. In [9] a finite element variational multi-scale (VMS) VOF method is presented, with a limiting approach for the marker, and a normal reconstruction as part of the technique for mass-conservation. The novelty of our method is the context of high Reynolds number turbulent flow with breaking waves, which is straight forward to extend with adaptive algorithms and fluid-structure interaction within

this framework [6, 10]. All methods are implemented in the open source software FEniCS, targeting the whole range of platforms from standard PCs to massively parallel supercomputers [11].

As a first step toward full fluid-structure interaction formulation, we here inclide moving boundaries in the methodology, both with prescribed motion and rigid-body objects with translational motion, to be able to support "heaving" motion, standard in marine applications. The moving boundaries are handled by an Arbitrary Lagrangian-Eulerian (ALE) formulation of the method We will return to the full fluid-structure interaction problem of flexible structures in future work, by connecting these methods to our previous work on unified continuum fluid-structure interaction [12, 10].

Simulation of a rotating turbine is a major computational challenge, still to be considered as an open problem in all its complexity. But impressive results have been achieved using stabilized finite element methods, see e.g. [13, 14]. In this article, the rotating turbine blades are modeled by prescribed moving boundaries with the ALE methodology, where the whole computational domain in the form of a cylinder is rotated;

The contribution of this article is thus a general framework of computational methods and open source software implementations, to address a set of simulation problems of key interest in the marine based renewable energy sector. In future work we will extend this results to fluid-structure interaction and aero-/hydroacoustics.

In Section 2 we present the underlying mathematical model, and the finite element method used to solve the equations. Four computational examples are presented Section 3: a regular wave test where we compare against an exact solution, the standard MARIN wave impact benchmark with experimental validation data, a vertical axis turbine with complex geometry from an existing turbine, and finally a prototype simulation of decay test in a coupled rigid-body and two-phase fluid simulation.

This article serves as a companion paper, presenting a detailed description of the methodology, for our IEA-OES Task 10 paper at EWTEC 2017 [15] where simulation results for a wave energy becnhmark from several groups are compiled.

## 2. Mathematical model

In this section we give the underlying mathematical model we use to simulate the ocean-wind system, in the form of the variable density Navier-Stokes equations, with the two phases represented by two different densities $\rho_1$ and $\rho_2$ with a diffusive interface in between. We discretise the equations by a stabilized finite element method, where we use a phase separation term to maintain a thin diffusive interface over time, which is constructed to conserve the total mass in time. No explicit physics based turbulence model is used, instead the residual based numerical stabilization is interpreted as a turbulence model based on the primitive Navier-Stokes equations, as a direct finite element simulation (DFS).

There is no explicit representation of the two phases or the interface, aside from the phase separation term, which acts as an optimization of the method, reducing diffusion of the interface. For single phase turbulent flow this methodology is validated for a number of benchmark problems, see e.g. [16, 17, 18, 19]

### *2.1. The variable density incompressible Navier-Stokes equations*

We write the variable density incompressible Navier-Stokes system of equations in a space-time domain $\Omega \times I$, in dimensionless form:

$$\mathbf{R}(\hat{\mathbf{u}}) = \mathbf{0} \begin{cases} \rho(\partial_t \mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u}) - \frac{1}{Re}\nabla\cdot(\mu(\nabla\mathbf{u} + \nabla\mathbf{u}^t)) + \nabla p - \frac{1}{Fr^2}\rho\mathbf{e}_z = 0, \\ \partial_t \rho + (\mathbf{u}\cdot\nabla)\rho = 0, \\ \nabla\cdot\mathbf{u} = 0 \end{cases} \tag{1}$$

where the variables $\rho, \mathbf{u}$, $p$ and $\mu$, all functions of $(x,t) \in \Omega \times I$, are dimensionless variables scaled through a selection of appropriate characteristic scales $L_0, U_0, \rho_0, \mu_0$ for the length, velocity, density and viscosity, respectively. We may recall that the Froude number is defined as $Fr = \frac{U_0}{\sqrt{gL_0}}$, and the Reynolds number as $Re = \frac{\rho_0 L_0 U_0}{\mu 0}$. . To complete the system we also need to add initial and boundary conditions to (1).

In some of our benchmark problems we can consider very high Reynolds numbers, we drop all viscous terms in the Navier-Stokes equations, corresponding to the inviscid Euler equations. We denote by $\hat{\mathbf{u}} = (\mathbf{u}, \rho, p)^T$ the vector of unknowns, and we denote by $\mathbf{R}_m$, $R_d$, and $R_c$ the strong residuals of the momentum, density, and continuity equations, respectively. The total residual can then be expressed as $\mathbf{R} = (\mathbf{R}_m, R_d, R_c)$.

### *2.2. Finite element method*

We now formulate the DFS finite element method of (1) with slip boundary conditions for the velocity ($\mathbf{u}\cdot\mathbf{n} = 0$) and homogeneous Neumann boundary conditions for the density and pressure. We subdivide the domain $\Omega$ into a conforming tetrahedral mesh $\Omega_h$, and the time interval $I$ into a collection of subintervals $I_n = (t_{n-1}, t_n)$. From a space-time finite element formulation with continuous piecewise linear approximation in time, and corresponding piecewise constant test functions in time, we formulate the method over a time interval $I_n$ as: find $\hat{\mathbf{u}}(t_n) \equiv \hat{\mathbf{u}}^{\mathbf{n}} \in (W_0)^3 \times W \times W$ such that

$$\mathbf{r}(\hat{\mathbf{u}}, \hat{\mathbf{v}}) = 0\,, \quad \forall\, \hat{\mathbf{v}} \in (W_0)^3 \times W \times W\,, \tag{2}$$

where $\hat{\mathbf{v}} = (\mathbf{v}, \eta, q)^T$ is the vector of test functions, $W$ a standard finite element space of piecewise linear Lagrange basis function, and $(W_0)^3$ the corresponding finite element space of vector functions satisfying a slip boundary condition.

The weak residual we define as

$$\begin{aligned}\mathbf{r}(\hat{\mathbf{u}}, \hat{\mathbf{v}}) = \bar{\rho}\left((\dot{\mathbf{u}},\, \mathbf{v}) + ((\bar{\mathbf{u}}\cdot\nabla)\bar{\mathbf{u}},\, \mathbf{v})\right) + \mu\ (\nabla^s\bar{\mathbf{u}},\, \nabla\mathbf{v}) \\ + (\nabla p,\, \mathbf{v}) - (\bar{\rho}g,\, \mathbf{v}) + \mathrm{LS}_m + \mathrm{SC}_m \\ + (\dot{\rho},\, \eta) + ((\bar{\mathbf{u}}\cdot\nabla)\bar{\rho},\, \eta) - c_{sep}\,(\rho_1 - \bar{\rho})(\rho_2 - \bar{\rho})\ (\mathbf{1},\, \nabla\eta) + \mathrm{LS}_d + \mathrm{SC}_d \\ + 2\,\Delta t\ \left(\nabla(p^n - p^{n-1}),\, \nabla q\right) + (\nabla\cdot\mathbf{u}^n,\, q) + \mathrm{LS}_c\,, \end{aligned} \tag{3}$$

where the bar and the point over the unknowns mean $\bar{(\cdot)} = \frac{(\cdot)^n + (\cdot)^{n-1}}{2}$ and $\dot{(\cdot)} = \frac{(\cdot)^n - (\cdot)^{n-1}}{\Delta t}$, respectively, $(\cdot,\cdot)$ is the $L^2$ scalar product, $\nabla^s(\cdot) = \nabla(\cdot) + \nabla(\cdot)^{\mathrm{T}}$, and $\Delta t = t^n - t^{n-1}$ is the time step length. The term $-c_{sep}\,(\rho_1 - \bar{\rho})(\rho_2 - \bar{\rho})\ (\mathbf{1},\, \nabla\eta)$ in Eq. (3) is here called separation term because it reduces the mixing of the two fluids. In this term, $\mathbf{1} = (1,1,1)$ is the vector of ones, $\rho_1$ and $\rho_2$ are the densities of the two phase fluids, satisfying $\rho_2 > \rho_1$, $c_{sep} = \frac{0.2}{0.1\|\nabla\bar{\rho}\|_1 + \bar{\rho}_0}$, and $\|\cdot\|_1$ is the $L^1$ norm. The term $2\,\Delta t\ \left(\nabla(p^n - p^{n-1}),\, \nabla q\right)$ in the last component of the residual in (3) corresponds to a Schur complement preconditioner [20], where the $n$ superscript denotes the iteration index in the fixed-point iteration, which means that the Schur term vanishes as the iteration converges.

The Galerkin least squares (GLS) stabilization terms in (3) are defined by

$$\mathrm{LS}_m = \left(d\ \tilde{\mathbf{R}},\, \mathbf{R}_{\mathbf{v}}\right)\,, \quad \mathrm{LS}_d = \left(d\ \tilde{\mathbf{R}},\, \mathbf{R}_{\eta}\right)\,, \quad \mathrm{LS}_c = \left(d\ \tilde{\mathbf{R}},\, \mathbf{R}_{q}\right)\,, \tag{4}$$

with the stabilization parameter $d = Ch^{3/2}$, $C > 0$ a constant of unit size, and

$$\tilde{\mathbf{R}} = \begin{pmatrix} \bar{\rho}(\bar{\mathbf{u}}\cdot\nabla)\bar{\mathbf{u}} + \nabla p - \bar{\rho}g \\ \bar{\mathbf{u}}\cdot\nabla\bar{\rho} - \mathbf{1}\cdot\nabla\left(c_{sep}(\rho_1 - \bar{\rho})\bar{\rho}\right) \\ \nabla\cdot\bar{\mathbf{u}} \end{pmatrix}\,, \quad \mathbf{R}_{\mathbf{v}} = \begin{pmatrix} \frac{1}{2}\bar{\rho}(\bar{\mathbf{u}}\cdot\nabla)\bar{\mathbf{v}} \\ \frac{1}{2}\bar{\mathbf{v}}\cdot\nabla\bar{\rho} \\ \nabla\cdot\bar{\mathbf{v}} \end{pmatrix}\,,$$

$$\mathbf{R}_{\eta} = \begin{pmatrix} \frac{1}{2}\eta(\bar{\mathbf{u}}\cdot\nabla)\bar{\mathbf{u}} \\ \frac{1}{2}\bar{\mathbf{u}}\cdot\nabla\eta - \mathbf{1}\cdot\nabla\left(c_{sep}(\rho_1 - \bar{\rho})\eta\right) \\ 0 \end{pmatrix}\,, \quad \mathbf{R}_{q} = \begin{pmatrix} \nabla q \\ 0 \\ 0 \end{pmatrix}\,.$$

The shock-capturing terms in (3) are defined as

$$\mathrm{SC}_m = \|\mathbf{u}^n\|_1 \left(c_1 h^2 R_{SC} + c_2 h^{3/2}\right)(\nabla\bar{\mathbf{u}},\, \nabla\mathbf{v})\,, \tag{5}$$

$$\mathrm{SC}_d = \|\mathbf{u}^n\|_1 \left(c_1 h^2 R_{SC} + c_2 h^{3/2}\right)(\nabla\bar{\rho},\, \nabla\eta)\,, \tag{6}$$

where $R_{SC} = R_d(\bar{\mathbf{u}}, \bar{\rho}) + \|\mathbf{R}_m(\bar{\mathbf{u}}, \bar{\rho}, p)\|_1$, with $c_1 > 0$ and $c_2 > 0$ constants of unit size.

We note that we recover the constant density case by removing the density equation from (1), together with the corresponding modifications in the finite element method. The DFS methodology for the constant density Navier-Stokes equations is described in detail e.g. in [6].

*2.3. ALE finite element method*

The problems we target in the marine based renewable energy area often includes a moving interface, between water and air, or between a fluid and a structure, or both at the same time. There are two ways to address this problem in the computational model. Either the interface is implicitly represented (interface capturing), e.g. as a level set of a scalar function, or the interface is explicitly represented (interface tracking), e.g. through deformation of the computational mesh. The approach described above, to model the two phase flow of air and water by a variable density is an example of interface capturing.

To track the interface explicitly with the mesh, the mesh deformation needs to be taken into account also in the finite element formulation. This can be done e.g. by a local ALE (Arbitrary Lagrangian-Eulerian) approach described in [12] for fluid-structure interaction, by subtracting the mesh velocity $\beta$ from the convective velocity over each time interval $I_n$. That is, all terms of the form

$$\bar{\rho}(\bar{\mathbf{u}} \cdot \nabla)\bar{\mathbf{u}}$$

$$(\bar{\mathbf{u}} \cdot \nabla)\bar{\rho}$$

are modified into

$$\bar{\rho}((\bar{\mathbf{u}} - \beta) \cdot \nabla)\bar{\mathbf{u}}$$

$$((\bar{\mathbf{u}} - \beta) \cdot \nabla)\bar{\rho}$$

in equation (3). In the next section we illustrate the framework both for a variable density model, a constant density model, and an ALE constant density model.

*2.4. Adaptive finite element method*

As described in [21], the DFS framework is naturally equipped with adaptive algorithms. An adaptive algorithm is based on a local error indicator for each element in the mesh, and a stop criterion for the algorithm. Both the local error indicators and the stop criterion are derived from a posteriori estimates of the error in chosen output functions. The basic component of the a posteriori error estimation is the solution of an adjoint problem which is derived from the continuous Navier-Stokes equations. In this article we illustrate an adaptive mesh refinement algorithm for a constant density model, whereas we leave adaptive algorithms for the variable density model to future work.

The adjoint problem is a system of convection-diffusion-reaction equations solved backwards in time, with the convection reversed so that transport is directed upstream. To estimate the error in drag, data for the adjoint problem is chosen as boundary data on the turbine, and the error indicator takes the form of the residual weighted by the solution of the adjoint problem. For a detailed description of a posteriori error estimation and the adjoint problem, see [21].

An adaptive mesh refinement algorithm takes the following form, where we for simplicity use the same space mesh and the same time step length for all time steps: Given an initial coarse computational space mesh $\Omega_h^0$, start at $k = 0$, then do:

1. Compute approximation of the primal problem using $\Omega_h^k$.
2. Compute approximation of the adjoint problem using $\Omega_h^k$.
3. If the stop criterion is satisifed then STOP, else:
4. On the basis of the size of the local error indicator, mark a fixed fraction of the elements in $\Omega_h^k$ for refinement. Obtain a new refined mesh $\Omega_h^{k+1}$, using a standard algorithm for local mesh refinement.
5. Set $k = k + 1$, then goto (1).

## 3. Results

We now consider four examples to highlight the potential of the DFS framework for high Reynolds number flow simulation in marine based renewable energy applications. The first two problems are free surface problems which we simulate using the variable density model, one regular wave where we compare against an exact solution, and the MARIN benchmark for which we have detailed validation data.

Next we consider the problem of a vertical axis turbine, to be used e.g. for ocean stream energy conversion. Validation data is not available at this point, but will be available for future studies. Instead we use the example to illustrate the potential of the ALE version of the methodology, where the rotating turbine is modeled by rotating the whole computational domain. The model geometry is fairly detailed, but several simplifications were used in this initial study, such as not including the ground and assuming a constant inflow velocity.

Finally, we present a prototype decay test for a floating cube. This test illustrates the coupling of the presented two-phase method together with the ALE metodology.

### *3.1. The regular wave benchmark*

In this section we will simulate a regular wave derived from the Stokes linear wave theory, commonly used to validate two phase flows solvers. The surface elevation of the wave $\eta$ and the velocity of the water $\mathbf{u}_w$ are the exact solution of the equations for an irrotational and incompressible two dimensional flow with dynamic free surface boundary condition, see [22] for instance.

The velocity of the wave is given by

$$\mathbf{u}_w = \frac{\pi H}{T \sinh(kd)} \left(\cosh[k(y+d)] \cos\theta, \sinh[k(y+d)] \sin\theta\right), \tag{7}$$

and the surface elevation is

$$\eta = \frac{H}{2} \cos\theta, \tag{8}$$

where $H$ is the wave height, $d$ is the mean water depth, $T$ the wave period, $\theta = kx - \omega t$, with $k$ is the wave number and $\omega = 2\pi/T$ the angular wave frequency. Since we are considering deep water we have to obtain the wave number from the equation $k = \omega^2/g \tanh(kd)$.

For our simulations, we consider $H = 2m$, $T = 4s$ and $d = 40m$. We solve the non dimensional form of the Navier Stokes equations 1 with the following characteristic scales

$$\rho_0 = 1000kg/m^3, \mu_0 = 2 \times 10^{-3}kg/sm, L_0 = 1m, U_0 = 1m/s, \tag{9}$$

therefore $\rho_1 = \frac{\rho_{air}}{\rho_0} = 0.001$ and $\rho_2 = \frac{\rho_{water}}{\rho_0} = 1$. With these values we obtain $Re = 5 \times 10^5$. To obtain the exact solution from the linear wave theory the viscous effects are not taken into account, but in our simulations we include the viscous term in our equations, in addiction to the stabilization terms.Thus we define the viscosity as

$$\mu = \mu_1 + \frac{\rho - \rho_1}{\rho_2 - \rho_1}(\mu_2 - \mu_1) \tag{10}$$

where $\mu_1 = \frac{\mu_{air}}{\mu_0} = 10^{-2}$ and $\mu_2 = \frac{\mu_{water}}{\mu_0} = 1$.

The computational domain considered is $\Omega = [0, 100] \times [-4, 6]$ being $y = 0$ the sea level, and we impose Dirichlet conditions for the density and velocity on the bottom, left and right of the boundary and outflow on the top. For the initial conditions we consider the exact solution at $t = 0$. Our triangular mesh has 41013 vertices and $h_{min} = 0.049$.

To goal of this test is to compare the behaviour of our method with and without the separation term to ensure that it effectively maintains better the separation surface between the two fluids.

If Figure 1 the density obtained with our simulations and the exact solution are compared at instant $t = 4s$. As can be observed both solutions are similar and show good agreement with the exact solution but our solutions show more diffusivity as can be expected. However, the separation term makes the solution less diffusive.

To have a more precise idea of the error we compute $\|\rho_h - \Pi_h \rho_{wave}\|_{L^2(\Omega)}$ in every time step, with $\rho_{wave}$ defined as

$$\rho_{wave}(x) = \begin{cases} \rho_1 \text{ if } x > \eta(x) \\ \rho_2 \text{ if } x \leq \eta(x) \end{cases} \tag{11}$$

and $\Pi_h$ the classical Lagrange interpolation operator. In Figure2 we compare the error obtained with and without using the separation term in our simulations.

### *3.2. The MARIN benchmark*

A standard benchmark in marine engineering for a wave impact or dam break is the MARIN benchmark [23], consisting of a door opening, releasing a volume of water and creating a wave which impacts a box, representing for example a container on a ship, see Figure 3. Pressure sensors are mounted on the box, providing validation data.

We apply the DFS model for variable density (2) to simulate the MARIN benchmark, using a tetrahedral mesh with approximately uniform mesh size and ca. 2 million vertices. In Figure 4 we visualize the density in the simulation,

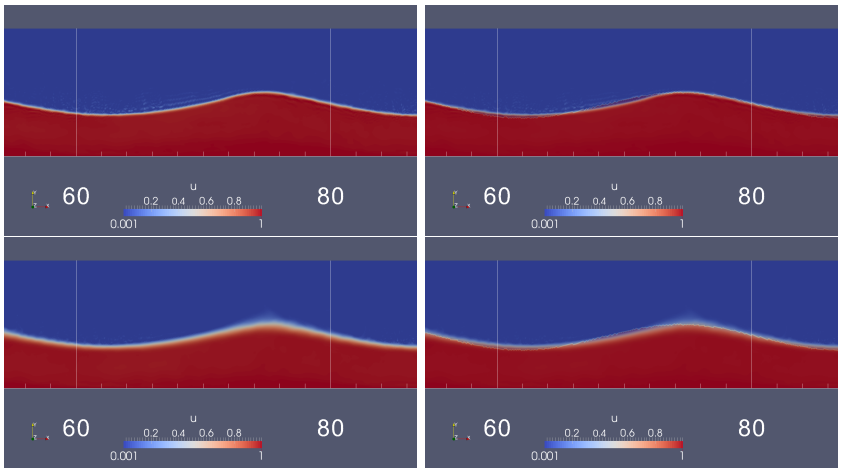



Figure 1: Density field at $t = 4s$ simulated (left) and compared with the exact solution (right), where the exact and simulated solutions are blended together, to allow inspection of the diffusion of the interface. Above the solution with the separation term and below without it. The exact solution has the sharp interface with clearly visible aliasing.

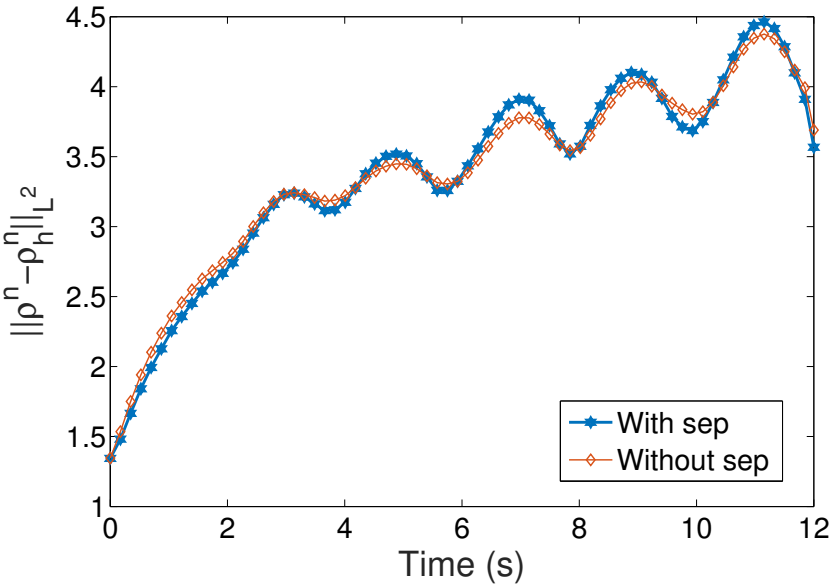



Figure 2: Error $\|\rho_h - \Pi_h \rho_{wave}\|_{L^2(\Omega)}$ in every time step computed with and without separation term.

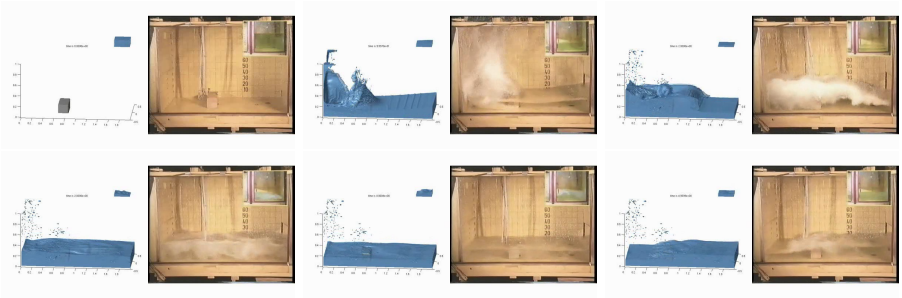


Figure 3: Snapshots from a video of the experiment (right) and reference VOF simulation (left) for the MARIN benchmark [23] at $t = 0, 1, 2, 3, 4, 5s$.

showing the evolution of the water surface e.g. identified as the level surface for $\rho = 0.5$, and in Figure 5 we compare to the case without the phase separation term in (2). The pressure signal for pressure sensors P1 and P7 [23] are compared between the simulations and experiment in Figure 6.

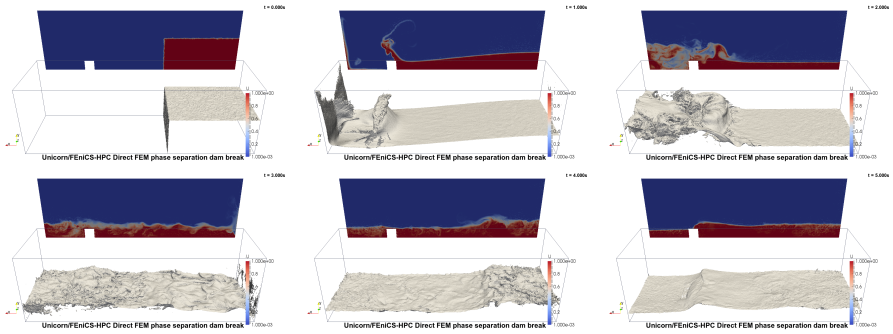



Figure 4: DFS simulation with phase separation for the MARIN benchmark [23]. Density slice and isosurface for $\rho = 0.5$ at $t = 0, 1, 2, 3, 4, 5s$.

We note that the evolution of the water surface is qualitatively reproduced qualitatively in the simulation, but without the phase separation term we find that the transition region in the density between water and air is widening as the simulation progresses. Quantitative comparison of the pressure sensor data reveals that the simulation with the separation term closely follows the experimental measurements, at least up to time $t \approx 5.5$, whereas the simulation without separation term largely fail to capture the pressure signal. Similar behaviour can be seen in Figure 7 where the water height on two sensors H2 and H4 of the experiment is compared against the simulation results with and without separation term.

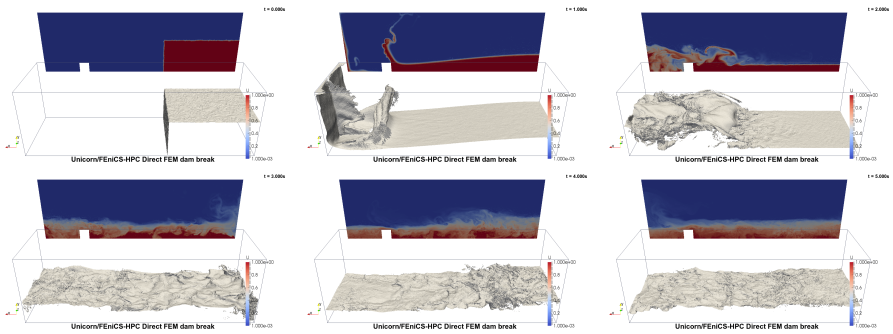


Figure 5: DFS simulation without phase separation for the MARIN benchmark [23]. Density slice and isosurface for $\rho = 0.5$ at $t = 0, 1, 2, 3, 4, 5s$.

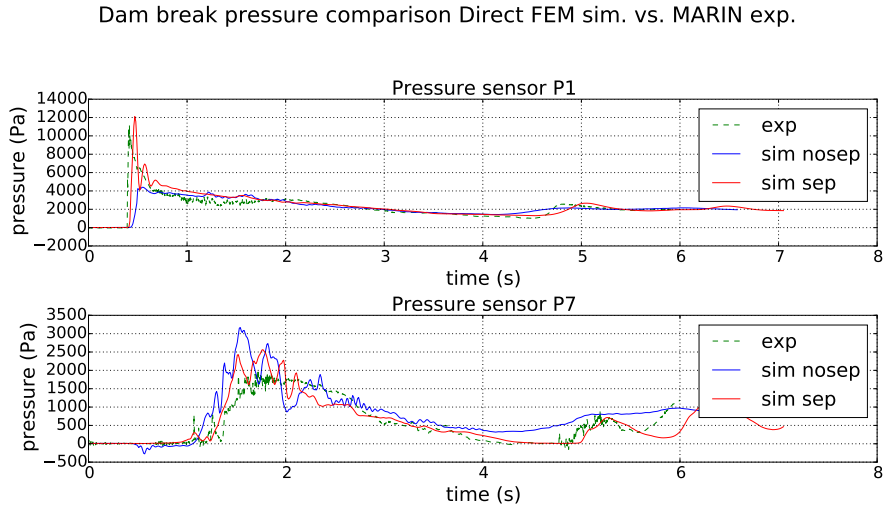



Figure 6: Pressure over time for experiment and simulation with and without the phase separation term for the sensors P1 and P7 in the MARIN benchmark [23].

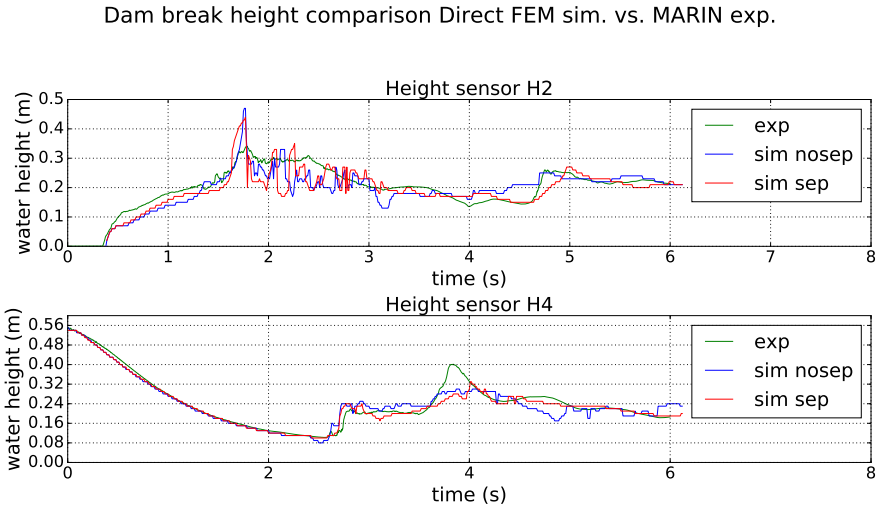



Figure 7: Water height over time for experiment and simulation with and without the phase separation term for the sensors H2 and H4 in the MARIN benchmark [23].

### *3.3. Vertical axis turbine*

A geometric model of a vertical axis turbine was created, similar to an existing turbine which is deployed to the field. In future studies validation data will thus be available, but here we include initial simulations to illustrate the complexity of the geometrical model, and the added challenges of a rotating turbine. The turbine diameter is 6.48 m and the blade height is 5 m. The blades are 0.25 m in chord in the 3 m center section and are linearly tapered the last meter on each side to a chord of 0.15 m at the blade tip. The profile is a NACA0021 on the blades.

In this reduced model we leave out the ground effects completely, so that the turbine is placed inside a volume of air with a constant inflow wind speed of $1m/s$, see Figure 8. In the case of a vertical axis turbine it is possible to simulate the rotating turbine by rotation of the full computational domain, with a suitable choice of domain. Here we choose a cylindrical domain of diameter $60m$ and length $20m$, with the turbine axis placed in the centerline of the cylinder domain, see Figure 9.

In Figures 10-11 we illustrate an adaptive mesh refinement algorithm with respect to the full drag force on the turbine, were we plot the mesh, the velocity and the adjoint velocity. Figure 15 shows the mesh sensitivity in the total drag of the turbine, with a coarsest mesh of 484 489 vertices and the finest mesh of 1 754 970 vertices. In Figure 12 we plot the drag force over time for a stationary turbine, and two rotation speeds, with two inflow speeds. We see a clear oscillatory behavior of the drag, and also a higher drag for the low inflow speed-high rotation speed case.

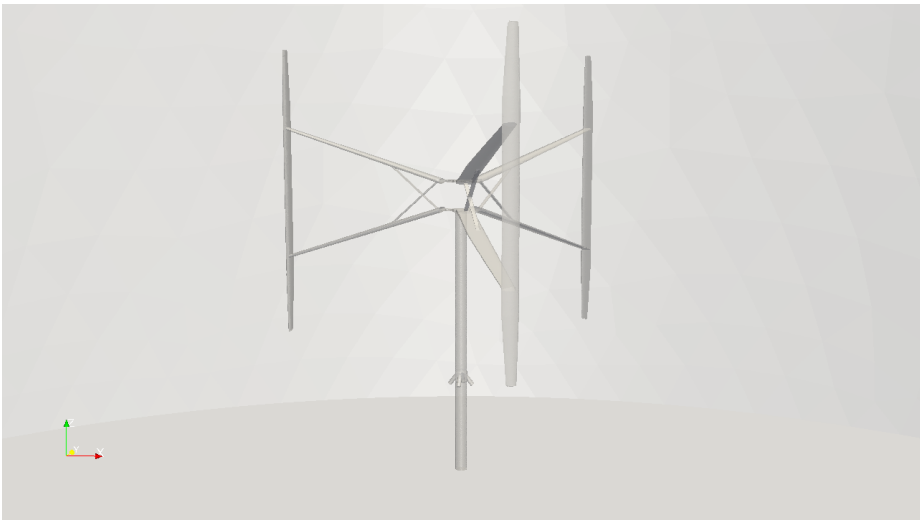


Figure 8: Turbine CAD model.

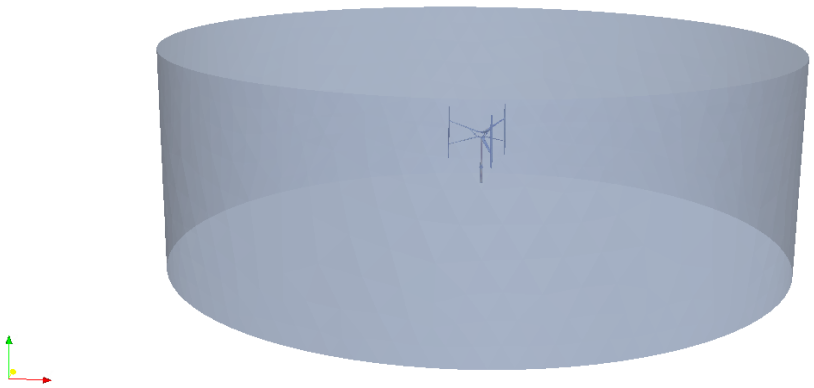


Figure 9: Cylindrical computational domain that contains the turbine model.

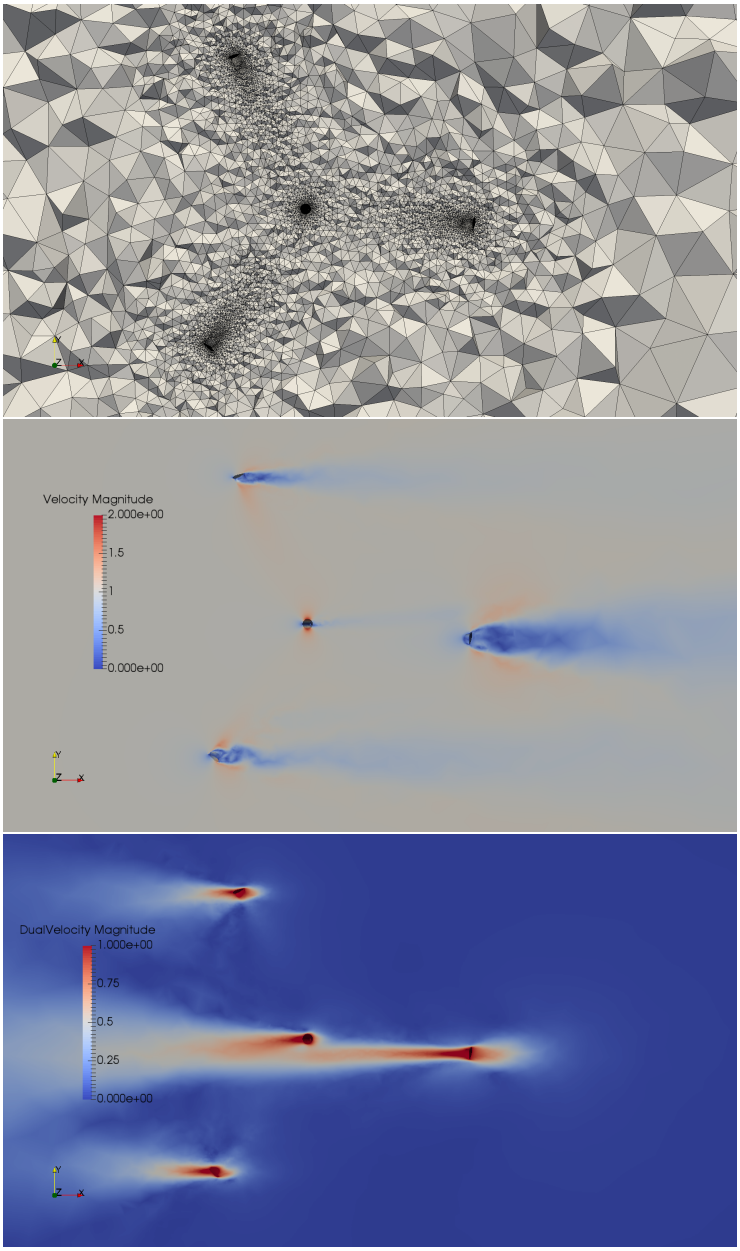



Figure 10: Vertical axis turbine vertical cut: starting mesh (upper), velocity magnitude (middle), and magnitude of the adjoint velocity (lower).

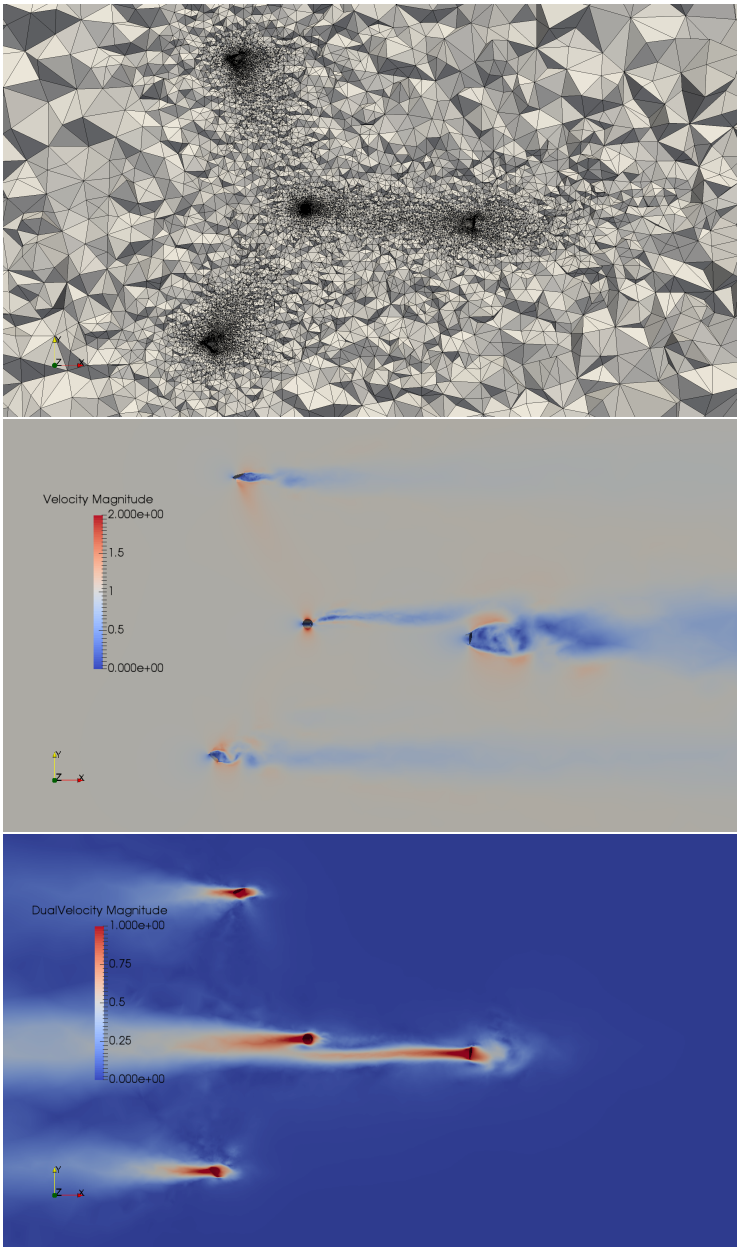



Figure 11: Vertical axis turbine vertical cut: mesh after 4 adaptive iterations (upper), velocity magnitude (middle), and magnitude of the adjoint velocity (lower).

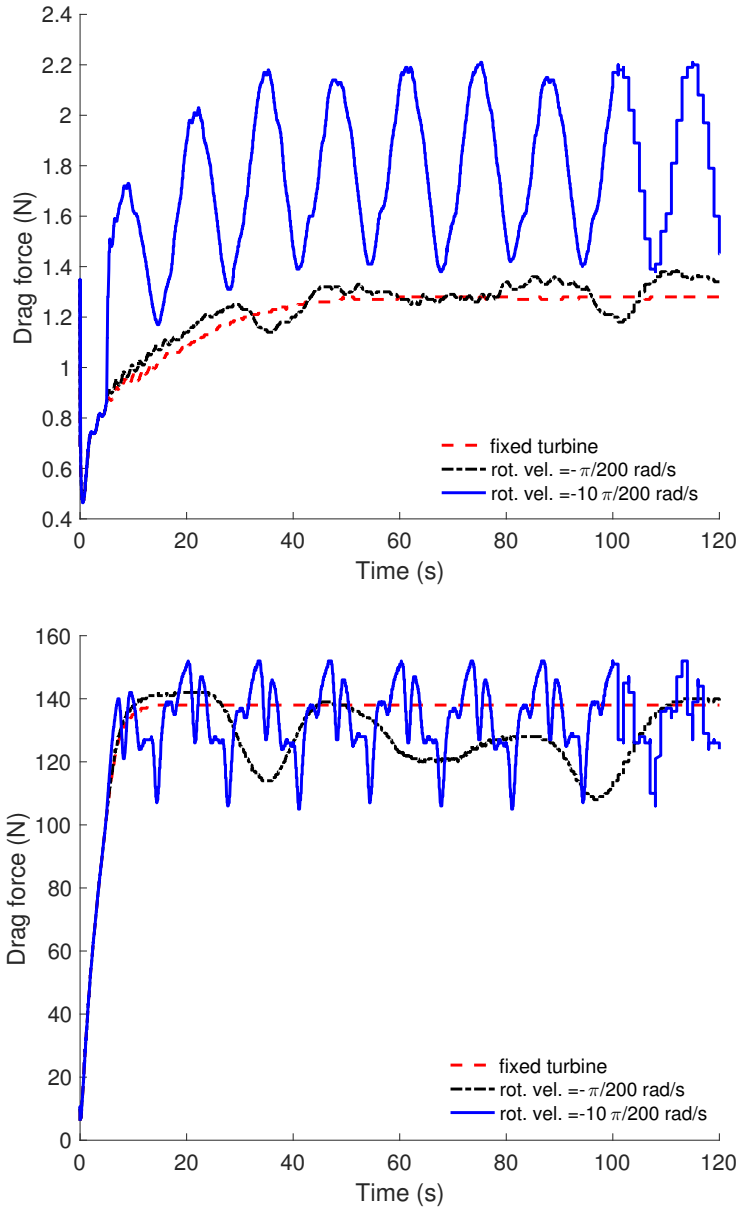



Figure 12: Drag force for two inflow velocities (1,0,0) (top) and (10,0,0) (bottom). Three different rotational velocities are considered for each case: fixed turbine, $-\pi/200$ rad/s and $-10\pi/200$ rad/s

*3.4. Rotating turbine*

We next turn to the case of a rotating turbine. Due to the particular construction of the vertical axis turbine, it is possible to simulate the rotation of the turbine by rotation of the full computational domain using an ALE technique. We illustrate this approach in two examples, one case with a rotational speed $-\pi/200rad/s$ and one case with rotational speed $-10\pi/200rad/s$, visualized in Figure 13 and Figure 14, respectively. The mesh corresponds to the coarsest mesh in the fixed turbine case.

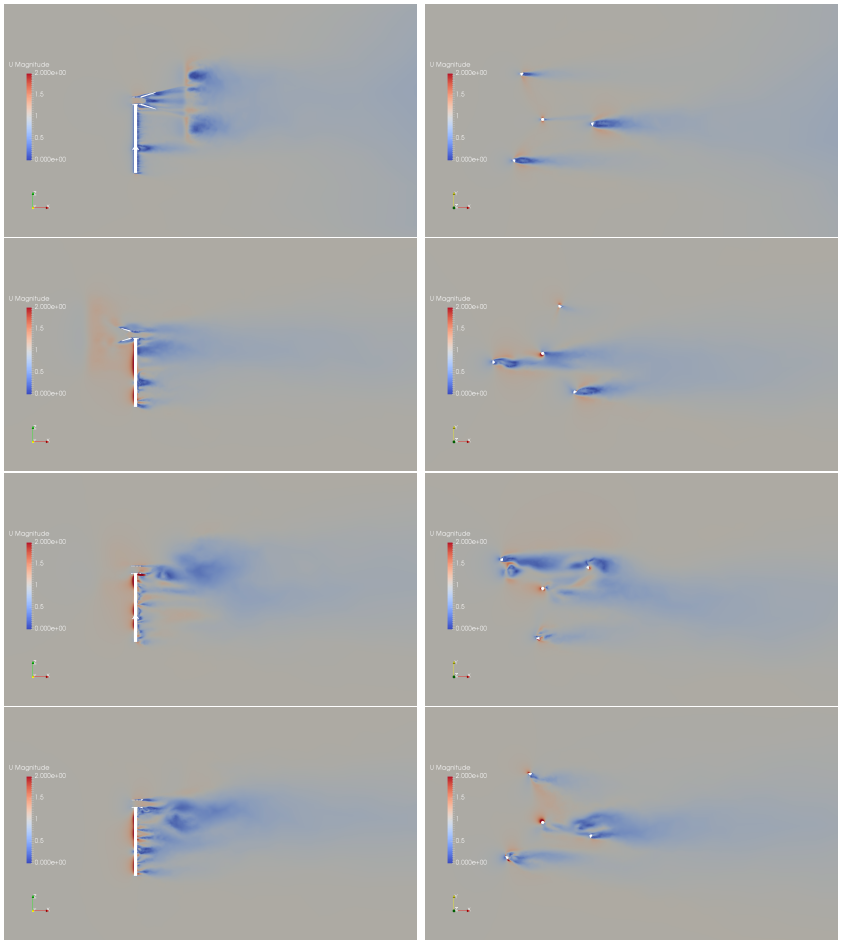


Figure 13: Rotating turbine at rotational speed $-\pi/200rad/s$ , for 4 snapshots in time: horisontal (left) and vertical cuts (right).

What can be observed is that the wake structures are different between the two cases. The slow rotation results in larger wakes, whereas with the faster rotation the wakes are smaller. This is consistent with observations of a flow

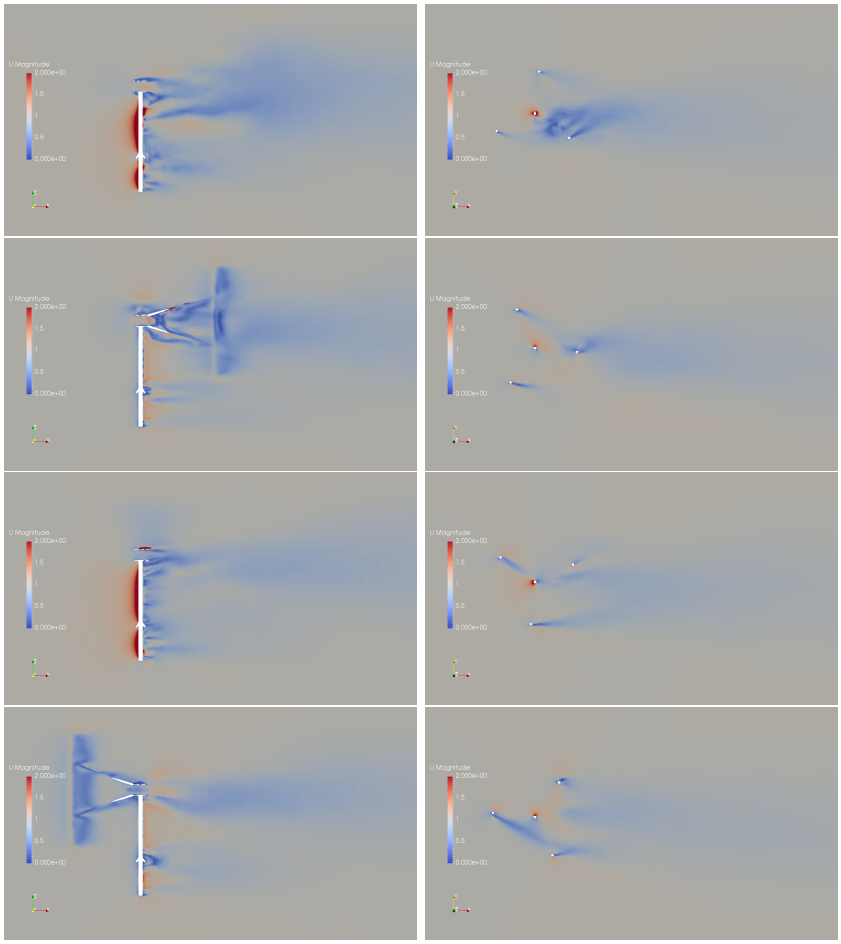


Figure 14: Rotating turbine at rotational speed $-10\pi/200 rad/s$ , for 4 snapshots in time: horisontal (left) and vertical cuts (right).

separation dependence on the rotation speed, e.g. dynamic stall effects, see e.g. [24]. In future studies we will investigate such effects in more detail, with particular focus on flow separation over the turbine blades.

We note that in this example we have not used adaptive mesh refinement. One reason for that is that we cannot directly employ the same strategy as for the fixed turbine, since even if the computational domain is symmetric in the vertical plane, the locally refined mesh is not. Another adaptive strategy is needed for the ALE case of a rotating domain, which we leave for future work.

### *3.5. Floating body decay test*

We simulate three-dimensional two-phase flow around a floating cube. The decay test consists of giving a downwards initial velocity to the cube and study the oscillation that occurs. We call $x_1$, $x_2$, and $x_3$ the three space coordinates. This is a preliminary test case were the force of the fluid on the cube is only taken into account in the vertical direction $x_2$, then the cube only moves in this direction. To this purpose, a simplification of the rigid body motion [25] is used here and briefly described in what follows. We compute the vertical force applied by the fluid on the cube boundary by $\mathbf{F} = \rho_c V_c\, g + \int_{\Gamma_c} p\, \mathbf{e}_2\, ds$, where $\rho_c$, $V_c$, and $\Gamma_c$ are the density, the volume, and the boundary of the cube, respectively, and $\mathbf{e}_2$ is the unit vector in the $x_2$ direction. By the Newton's second law $\mathbf{F} = \rho_c V_c\, \partial_t \mathbf{w}$, we compute $\mathbf{w}$ which is the vertical velocity of the cube, and move the mesh consequently. As explained in subsection 2.3, the boundary of the cube is tracked with the mesh by an ALE approach using the cube velocity $\mathbf{w}$.

The dimensions of the computational domain are $-10 \leq x_1 \leq 10$, $-4 \leq x_2 \leq 4$, and $-4 \leq x_3 \leq 4$. A mesh of 353740 grid points is used for this simulation. The minimum edge lenght of the mesh being 0.0308. The floating cube is initially defined by $-0.5 \leq x_1 \leq 0.5$, $-0.5 \leq x_2 \leq 0.5$, and $-0.5 \leq x_3 \leq 0.5$, being the center of mass of the cube placed at the origin (0, 0, 0) when the simulation starts. The flow is initially at rest and the cube has an initial vertical velocity of (0, -1, 0). The non dimensional form of the Navier-Stokes equations (1) are used in this paper. The non dimensional densities of the two flows are $\rho_1 = 1e-3$ and $\rho_2 = 1$, and the non dimensional density of the cube is set to $\rho_c = 0.55$. As shown in Fig. 17a, the interphase between the two fluids is the plane $x_2 = 0.0$, then the initial density is $\rho_2$ when $x_2 \leq 0.0$ and $\rho_1$ otherwise. In this preliminary test case we prescribe a uniform non dimensional viscosity of $\mu = 1.0e-5$, in the future we should use different viscosities for the two fluids. The characteristic scales used in this test case for the nondimensionalization of the equations are: $\rho_0 = 1\, kg/m^3$, $\mu_0 = 1\, kg/sm$, $L_0 = 1\, m$, $U_0 = 1\, m/s$. Slip boundary conditions are set everywhere.

In Fig. 17 we plot the filled contours of the density on a $x_1x_2$-plane passing through the origin of the domain, for different times. The density of the initial time as well as times corresponding to maximum and minimum displacements are ploted. In Fig. 16 we plot the vertical displacement of the center of mass of the cube over time. We observe the expected decay of the vertical movement of the cube when time advances.

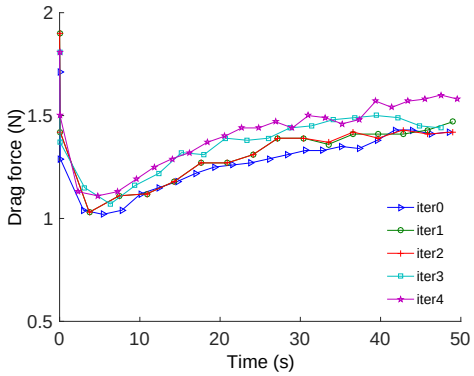



Figure 15: Drag force of the fixed vertical axis wind turbine, over 4 adaptive iterations.

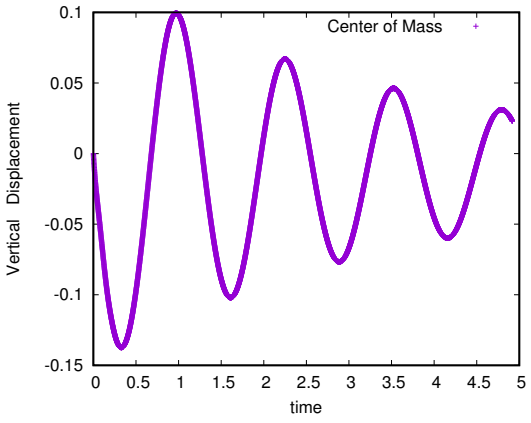
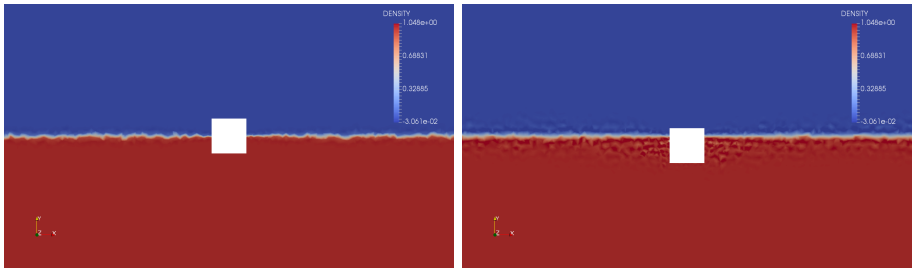
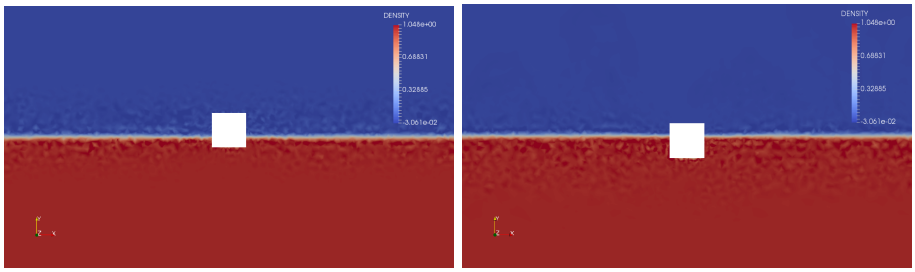



Figure 16: Vertical displacement of the center of mass of the floating cube over time.

(a) $t = 0.0\,s$ (b) $t = 0.3\,s$




(c) $t = 1.0\,s$ (d) $t = 3.0\,s$




(e) $t = 3.6\,s$


Figure 17: Filled contours of the density on a slice of the domain on the $x_1x_2$-plane, for different times, from $t = 0.0\,s$ to $t = 3.6\,s$.

## 4. Summary and Discussion

In this article we have presented a framework for simulation of turbulent flow in applications of marine based renewable energy, with the characteristics of high Reynolds numbers, free surface flows and breaking waves, and fluid-structure interaction. The framework extends the DFS (Direct finite element simulation) framework for constant density flow [21] to variable density flow, which is used to model ocean waves in a multiphase model of water and air, and ALE moving domains. The DFS methodology has a strong mathematical foundation [21], and extensive validation of the methods [16, 17, 18, 19]. All methods are implemented in the open source software FEniCS, targeting the whole range of platforms from standard PCs to massively parallel supercomputers [11]. The framework can be extended with adaptive algorithms.

We illustrate the capabilities of the methods in several examples: a regular wave example where we see a close match to the exact solution, the MARIN benchmark where we show that the simulation qualitatively and quantitatively reproduce the wave and local pressure of the available validation data; a vertical axis turbine which is investigated for the case of a fixed turbine, as well as two different rotation speeds using an ALE method; and finally a prototype simulation of a decay test of a floating body, with both multi-phase and moving boundary, where we observe the qualitative decay behavior..

We conclude that a DFS method for inviscid variable density incompressible flow, without any explicit treatment of a level set distance function, apart from the addition of a phase separation term, appears to perform well as a two phase model, demonstrated for both the regular wave and the MARIN benchmark. In [7], a similar performance is observed for the MARIN benchmark at similar mesh resolution, but where an explicit level-set method is used which requires the solution of an eikonal equation.

The DFS method is straightforward to extend to ALE to treat rotating turbines as demonstrated, and also in the multi-phase setting as demonstrated in the floating body decay test prototype simulation, and one future goal is to combine a rotating turbine mounted on a floating platform with waves hitting the platform. In future work we will further develop the adaptive algorithms for variable density flow and ALE methods, and we will also study the vertical axis turbine model in more detail.

## Acknowledgements

This research has been supported by the European Research Council, the Swedish Energy Agency, 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. The regular wave results were developed in collaboration with Tecnalia in the setting of the ICERMAR project.

## References

[1] P. Sagaut, Large eddy simulation for incompressible flows, 3rd edition, Springer, 2005.

[2] J. Hoffman, C. Johnson, A new approach to computational turbulence modeling, Computer Methods in Applied Mechanics and Engineering 195 (23) (2006) 2865–2880.

[3] Y. Bazilevs, V. Calo, J. Cottrell, T. Hughes, A. Reali, G. Scovazzi, Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows, Computer Methods in Applied Mechanics and Engineering 197 (1) (2007) 173–201.

[4] J. Hoffman, C. Johnson, Computational Turbulent Incompressible Flow: Applied Mathematics: Body and Soul 4, Vol. 4, Springer, 2007.

[5] M. Nazarov, J. Hoffman, Residual-based artificial viscosity for simulation of turbulent compressible flow using adaptive finite element methods, International Journal for Numerical Methods in Fluids 71 (3) (2013) 339–357.

[6] J. Hoffman, J. Jansson, N. Jansson, R. V. De 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 (2015) 60–74.

[7] C. E. Kees, I. Akkerman, M. W. Farthing, Y. Bazilevs, A conservative level set method suitable for variable-order approximations and unstructured meshes, Journal of Computational Physics 230 (12) (2011) 4536–4558.

[8] E. Olsson, G. Kreiss, S. Zahedi, A conservative level set method for two phase flow ii, Journal of Computational Physics 225 (1) (2007) 785–807.

[9] E. F. Lins, R. N. Elias, F. A. Rochinha, A. L. Coutinho, Residual-based variational multiscale simulation of free surface flows, Computational Mechanics 46 (4) (2010) 545–557.

[10] J. Hoffman, J. Jansson, R. Vilela de Abreu, N. C. Degirmenci, N. Jansson, K. Müller, M. Nazarov, J. H. Spühler, Unicorn: Parallel adaptive finite element simulation of turbulent flow and fluid–structure interaction for deforming domains and complex geometry, Computers & Fluids 80 (2013) 310–319.

[11] J. Hoffman, J. Jansson, N. Jansson, Fenics-hpc: Automated predictive high-performance finite element computing with applications in aerodynamics, Proceedings of the 11th International Conference on Parallel Processing and Applied Mathematics, PPAM 2015. Lecture Notes in Computer Science 9573.

[12] J. Hoffman, J. Jansson, M. Stöckli, Unified continuum modeling of fluid-structure interaction, Mathematical Models and Methods in Applied Sciences 21 (03) (2011) 491–513.

[13] Y. Bazilevs, M.-C. Hsu, I. Akkerman, S. Wright, K. Takizawa, B. Henicke, T. Spielman, T. Tezduyar, 3d simulation of wind turbine rotors at full scale. part i: Geometry modeling and aerodynamics, International Journal for Numerical Methods in Fluids 65 (1-3) (2011) 207–235.

[14] Y. Bazilevs, M.-C. Hsu, J. Kiendl, R. Wüchner, K.-U. Bletzinger, 3d simulation of wind turbine rotors at full scale. part ii: Fluid–structure interaction modeling with composite blades, International Journal for Numerical Methods in Fluids 65 (1-3) (2011) 236–253.

[15] F. F. Wendt, Y.-H. Yu, K. Nielsen, K. Ruehl, T. Bunnik, I. Touzon, B. W. Nam, J. S. Kim, K.-H. Kim, C. E. Janson, et al., International energy agency ocean energy systems task 10 wave energy converter modeling verification and validation, in: 12th European Wave and Tidal Energy ConferencEuropean Wave and Tidal Energy Conference, 2017.

[16] J. Hoffman, Adaptive simulation of the subcritical flow past a sphere, Journal of Fluid Mechanics 568 (2006) 77–88.

[17] J. Hoffman, Efficient computation of mean drag for the subcritical flow past a circular cylinder using general galerkin g2, International journal for numerical methods in fluids 59 (11) (2009) 1241–1258.

[18] R. V. de Abreu, N. Jansson, J. Hoffman, Computation of aeroacoustic sources for a complex nose landing gear geometry using adaptivity, in: Proceedings of the Second Workshop on Benchmark problems for Airframe Noise Computations (BANC-II), Colorado Springs, 2012.

[19] J. Hoffman, J. Jansson, N. Jansson, R. V. de Abreu, Time-resolved adaptive fem simulation of the dlr-f11 aircraft model at high reynolds number, in: AIAA 2014-0917, Proc. 52nd Aerospace Sciences Meeting, AIAA SciTech, 2014.

[20] G. Houzeaux, M. Vázquez, R. Aubry, J. Cela, A massively parallel fractional step solver for incompressible flows, Journal of Computational Physics 228 (17) (2009) 6316–6332.

[21] J. Hoffman, J. Jansson, N. Jansson, R. Vilela De Abreu, C. Johnson, Computability and adaptivity in cfd. encyclopedia of computational mechanics, stein, e., de horz, r. and hughes, tjr eds (2016).

[22] R. G. Dean, R. A. Dalrymple, Coastal processes with engineering applications, Cambridge University Press, 2004.

[23] K. Kleefsman, G. Fekken, A. Veldman, B. Iwanowski, B. Buchner, A volume-of-fluid based simulation method for wave impact problems, Journal of computational physics 206 (1) (2005) 363–393.

[24] E. Dyachuk, A. Goude, Simulating dynamic stall effects for vertical axis wind turbines applying a double multiple streamtube model, Energies 8 (2) (2015) 1353–1372.

[25] R. Panahi, E. Jahanbakhsh, M. S. Seif, Development of a vof-fractional step solver for floating body motion simulation, Applied Ocean Research 28 (3) (2006) 171 – 181. doi:http://dx.doi.org/10.1016/j.apor.2006.08.004.
URL //www.sciencedirect.com/science/article/pii/S0141118706000812