

# Master's thesis

**NTNU**  
Norwegian University of Science and Technology  
Faculty of Engineering  
Department of Civil and Environmental Engineering

Aron Vogelsang

## REEF2FAST

A Coupling Tool for Enhanced Floating Offshore Wind Turbine Analysis Using REEF3D::NHFLOW and OpenFAST

Master's thesis in Marine Civil Engineering

Supervisor: Hans Bihs

Co-supervisor: Tommi Mikkola

June 2025



 NTNU

Norwegian University of  
Science and Technology

**A!** Aalto-yliopisto  
Aalto-universitetet  
Aalto University



Aron Vogelsang

## **REEF2FAST**

A Coupling Tool for Enhanced Floating Offshore Wind  
Turbine Analysis Using REEF3D::NHFLOW and  
OpenFAST

Master's thesis in Marine Civil Engineering  
Supervisor: Hans Bihs  
Co-supervisor: Tommi Mikkola  
June 2025

Norwegian University of Science and Technology  
Faculty of Engineering  
Department of Civil and Environmental Engineering



Norwegian University of  
Science and Technology



# Abstract

The advancement of floating offshore wind turbines (FOWTs) requires simulation tools that can capture the interaction between atmospheric loading and realistic sea states. While established system-level tools like OpenFAST offer aero-hydro-servo-elastic modeling capabilities, their implemented wave modules may rely on simplified wave theory, which cannot fully capture complex sea conditions. To close this gap, this thesis introduces REEF2FAST, a non-invasive coupling pre-processor that integrates wave kinematics from the shock-capturing non-hydrostatic Navier-Stokes solver REEF3D::NHFLOW into OpenFAST.

REEF2FAST employs a memory-efficient, time-step-streaming approach and a parallelized spatial interpolation scheme based on a k-d tree nearest neighbor search and inverse distance weighting (IDW). It enables the projection of high-fidelity NHFLOW wavefields onto OpenFAST's Cartesian grid, including wave crest regions above the still water level (SWL) using the Wheeler stretching method. Written in C++ and released as an open-source command-line application, REEF2FAST is designed for simple integration into coupled simulation pipelines.

Verification tests demonstrate the accurate transmission of wave kinematics across various wave conditions. Validation against experimental platform motions from the OC5 DeepCWind Campaign Phase II confirms the tool's ability to reproduce realistic six-degree-of-freedom (6-DoF) dynamics under both second-order Stokes waves and the irregular JONSWAP spectrum.

By enabling the use of externally computed wavefields, REEF2FAST expands OpenFAST's applicability to more complex hydrodynamic environments, such as shallow waters and higher-order wave theories. The modular design of REEF2FAST lays the groundwork for future developments, including two-way fluid-structure interaction and the implementation of complex bathymetries.

# Acknowledgements

I would like to thank Hans Bihs for the opportunity to work on this thesis and for enabling me to deepen my understanding of CFD and engineering programming. My sincere thanks go to Widar W. Wang, Elyas Lakermani, and Ahmet Soydan for their continuous support, guidance, and numerous insightful discussions throughout the project. I also thank Tommi Mikkola for evaluating the progress of the work and providing valuable feedback on the challenges encountered during the coupling process. Furthermore, I gratefully acknowledge the National Renewable Energy Laboratory, especially Lucas Carmo and Amy Robertson, for providing reference data and helpful clarifications regarding OpenFAST.

# Contents

<b>Abstract</b>	<b>I</b>
<b>Acknowledgements</b>	<b>II</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Conceptual Framework</b>	<b>4</b>
2.1 Coupling . . . . .	4
2.2 REEF3D . . . . .	5
2.2.1 Overview of Numerical Modules . . . . .	5
2.2.2 REEF3D::NHFLOW . . . . .	6
2.3 OpenFAST . . . . .	8
2.3.1 Mooring System Modeling . . . . .	9
2.3.2 Wave Kinematics . . . . .	10
2.3.3 Hydrodynamic Loads and Platform Motions . . . . .	11
2.3.4 Internal Coupling Architecture of OpenFAST . . . . .	12
<b>3 Methodology</b>	<b>14</b>
3.1 REEF2FAST . . . . .	14
3.1.1 Software Architecture . . . . .	14
3.1.2 Interpolation . . . . .	16
3.1.3 Wheeler Projection . . . . .	18
3.1.4 Surface Elevation . . . . .	19
3.1.5 Acceleration . . . . .	19
3.2 Verification . . . . .	20
3.2.1 Regular and Second-Order Stokes Wave Cases . . . . .	21
3.2.2 Case 3: Irregular JONSWAP Spectrum . . . . .	23
3.3 Validation . . . . .	25
3.3.1 Experimental Test Setup . . . . .	25
3.3.2 Turbine Modeling . . . . .	27
<b>4 Results</b>	<b>29</b>
4.1 Verification Test Results . . . . .	29
4.1.1 Case 1 - Wave Kinematics . . . . .	29
4.1.2 Case 2 - Wave Kinematics . . . . .	30
4.1.3 Case 3 - Wave Kinematics . . . . .	32
4.2 Validation Test Results . . . . .	33
4.2.1 Case 2 - Platform Response . . . . .	34
4.2.2 Case 3 - Platform Response . . . . .	35

<b>5 Analysis</b>	<b>37</b>
5.1 Verification Analysis . . . . .	37
5.1.1 Correctness and Deviations . . . . .	37
5.1.2 Influence of the Wheeler Stretching . . . . .	38
5.2 Validation Analysis . . . . .	38
5.2.1 Correctness and Deviations . . . . .	38
5.2.2 Modeling Uncertainties . . . . .	38
<b>6 Outlook</b>	<b>40</b>
6.1 Complex Bathymetries . . . . .	40
6.2 Two-Way Coupling . . . . .	41
6.3 Performance Enhancements . . . . .	42
<b>7 Conclusion</b>	<b>43</b>
<b>Bibliography</b>	<b>45</b>

# Chapter 1

## Introduction

Offshore wind energy plays a central role in the global transition towards renewable energy systems. As the demand for renewable energy generation accelerates, wind turbines have grown not only in size but also in technological complexity. Projects such as Hornsea 2 [1] in England and Hywind Tampen [2] in Norway demonstrate the industry's shift towards large-scale deployment or floating solutions in deep waters. However, deployment at scale and the expansion into deeper waters using floating substructures remain constrained by high costs. Accurate prediction of structural loads and system performance is a key lever for cost reduction, particularly during design and optimization phase.

To address these challenges, the offshore wind industry has developed a wide range of numerical models. Tools such as WAMIT [3] and OrcaFlex [4] have traditionally been used for the computation of hydrodynamic loads and structural motions. Additionally, computational fluid dynamics (CFD) and wave model projects like REEF3D [5] have gained attention for resolving complex free-surface phenomena including wave breaking, run-up, and nonlinear diffraction modeling. On the experimental side, testing campaigns such as FOCAL [6] and OC5 [7] provide validation data, but are costly, limited in scale, and often restricted to model-scale induced simplified conditions.

Despite progress in numerical and experimental techniques, standalone methods frequently involve trade-offs between fidelity and computational efficiency. High-resolution CFD solvers offer detailed resolution of flow phenomena but are computationally demanding. In contrast, low-cost structural solvers provide rapid evaluations but tend to oversimplify nonlinear interactions, particularly in wave-structure interactions and mooring dynamics. To overcome these limitations, hybrid and coupled approaches have emerged, aiming to combine the strengths of different tools. For example, Zhang et al. [8] overcame the limitations of standalone solvers by combining Higher-Order Spectral (HOS) wave theory with CFD in a coupled framework. While HOS efficiently generates broadband irregular wavefields, it lacks the ability to model nonlinear wave deformation and wave-structure interaction in detail. By feeding the HOS-generated waves into a CFD domain, the authors were able to capture complex hydrodynamic responses of a semi-submersible floating offshore wind turbine (FOWT) over a 3-hour simulation, including wave run-up and mooring loads, with high physical fidelity. Moreover, Tagliafierro et al. [9] employed a fully Lagrangian Smoothed Particle Hydrodynamics (SPH) approach to resolve nonlinear interactions between wave groups and a semi-submersible FOWT. By coupling DualSPHysics for the fluid domain with Project Chrono for structural and mooring dynamics, the authors were able to simulate wave breaking, vorticity generation, and platform response with high resolution. This method allowed the accurate reproduction of experimental results, particularly in regimes dominated by nonlinear and dispersive wave effects, demonstrating the potential of SPH-MBS (Multi-Body System) coupling for challenging offshore scenarios.

A prominent example of a solver that inherently follows a coupling approach within a modular solver architecture is OpenFAST [10], developed by the National Renewable Energy Laboratory (NREL). OpenFAST is one of the most widely used open-source tools for aero-hydro-servo-elastic coupled simulations of wind turbines. Its architecture enables flexible configuration of turbine, platform,

and control systems, making it particularly attractive for the coupled analysis of FOWTs. However, its hydrodynamic modeling relies primarily on linear wave theory and potential or strip theory assumptions. As a result, it struggles to capture a range of complex hydrodynamic scenarios, including wave breaking, shallow-water effects, and most notably nonlinear higher-order wave propagation. Consequently, although OpenFAST adopts a modular coupled architecture, it still involves simplifications of various physical phenomena. Nevertheless, this modular architecture leaves room for extensions and coupling strategies to overcome these challenges. Especially in the field of hydrodynamics, coupling approaches involving OpenFAST have become increasingly popular in offshore wind research.

One early development was FASTLink, which connects OpenFAST with the commercial software OrcaFlex to enhance the dynamic modeling of mooring systems and floating platforms [11]. More recently, FRYFAST extended this idea by integrating OpenFAST with the multiphysics solver FRyDoM, allowing for detailed multibody interactions, added mass effects, and time-step synchronization [12].

A particularly active branch of this field involves the use of CFD solvers, particularly OpenFOAM. OF<sup>2</sup>, for example, is a coupling framework that combines OpenFAST and OpenFOAM to simulate the full aero-hydro-servo-elastic behavior of FOWTs under challenging conditions such as yaw drift, mooring failure, and turbulent inflow [13]. Kim et al. [14] extended the OpenFOAM-OpenFAST coupling approach and validated it against the SOFTWIND benchmark, achieving high accuracy in replicating experimental platform responses and mooring tensions under irregular wave scenarios. Beyond fully integrated frameworks, some studies have explored semi-coupled configurations to address specific limitations. For instance, Jeon et al. [15] invented a coupling between OpenFOAM and the MoorDyn library to investigate the moored motion of a semi-submersible platform under wave loading. While their setup omits aerodynamic and structural modules typically included in OpenFAST-based simulations, it presents a modular, targeted approach to wave-mooring interaction analysis. Such partial couplings are valuable when the hydrodynamic fidelity of CFD is needed, but full aero-hydro-servo-elastic coupling is not required.

To simplify and generalize coupling processes, Willeke and Uekermann [16] introduced a modular adapter that can connect OpenFAST to arbitrary solvers through the preCICE library [17]. This black-box coupling approach lowers integration barriers and enables flexible co-simulation architectures.

He et al. [18] generally categorizes coupling methods in the context of offshore wind into uncoupled, partially coupled, and fully coupled models, each offering trade-offs in terms of computational cost and physical accuracy. A comparative study by Yu et al. [19] highlights the diversity and trade-offs among existing modeling strategies. By evaluating the hydrodynamic response of an FOWT across various solvers, including OpenFAST, and multiple experimental campaigns, the study concludes that no single approach can generally capture all relevant physical effects. This highlights the value of coupling approaches that combine the strengths of different tools.

While coupling strategies primarily focus on improving the external hydrodynamic input, internal advancements in OpenFAST have also expanded its capabilities. Carmo et al. [20] recently validated OpenFAST's ability to compute local structural loads on floating substructures by extending OpenFAST's HydroDyn module to support flexible pontoons. Their study, based on the FOCAL experimental campaign, confirmed that OpenFAST can accurately capture local loads under regular and irregular waves when supplied with high-resolution wave input, even using simplified strip theory models.

These developments underscore the increasing demand for high-fidelity wave kinematics in floating wind simulations. However, existing coupling strategies often require extensive code modification or rely on tightly integrated co-simulation frameworks. In contrast, the approach taken in this thesis adopts a more lightweight and modular philosophy.

The central contribution of this work is the development of a dedicated pre-processing toolchain, named REEF2FAST, which transforms and exports wavefield data from the open-source hydrodynamics framework REEF3D into OpenFAST-compatible formats. Within the REEF3D framework, the shock-capturing non-hydrostatic Navier-Stokes solver NHFLOW was chosen as the hydrodynamic

input source. NHFLOW offers a favorable balance between computational efficiency and physical accuracy, sufficiently resolving the vertical distribution of flow structures, unlike shallow-water equation or Boussinesq-type models [21]. Moreover, it provides an intrinsic pressure field, in contrast to potential flow solvers, while avoiding the high computational costs of full CFD models. Consequently, the coupling between NHFLOW and OpenFAST allows the use of high-fidelity wavefields in OpenFAST while preserving OpenFAST's efficiency for lightweight simulations. REEF2FAST is designed to support both regular and irregular wave conditions, and operates on either two- or three-dimensional domains. By automating the data transfer between REEF3D and OpenFAST without modifying either, REEF2FAST enables physically consistent, high-resolution hydrodynamic input for FOWT simulations, offering a practical and extensible solution for offshore wind research.

First, the theoretical background of numerical coupling is presented, along with an overview of the two computational frameworks involved; REEF3D and OpenFAST. The following chapter introduces the coupling tool REEF2FAST and explains its underlying concepts and implementation. Subsequently, the tool is verified to ensure the correct transfer of wave kinematics from REEF3D to OpenFAST. A validation study based on six-degree-of-freedom (6-DoF) platform motions is then carried out to assess the physical accuracy of the coupled system. Finally, the analysis outlines limitations of the current framework and the outlook chapter proposes directions for future development.

# Chapter 2

## Conceptual Framework

### 2.1 Coupling

In the field of computational engineering, the term *coupling* refers to the facilitation of interaction between physical subdomains, typically fluid, structure, or environment, that influence each other during a simulation. These so-called multiphysics problems require the simultaneous solution of separate sets of governing equations, such that the physical feedback between subsystems is captured consistently. Hence, the demand of coupling concepts occurs either within an integrated multiphysics solver or in synchronizing two independent simulation tools. Depending on the direction and strength of the coupling, different mathematical strategies and solver architectures are applied.

A fundamental distinction is made between *one-way* and *two-way* coupling. In a one-way coupling strategy, information flows unidirectionally. For example, wave kinematics derived from a CFD model might be passed to a structural solver as an external forcing, without accounting for the back-reaction of the structure on the fluid domain. While computationally inexpensive, this approach neglects dynamic feedback effects and is only valid in cases where such influence is negligible [22]. In contrast, two-way coupling involves bidirectional exchange of information between domains. Here, fluid forces may deform or displace the structure, and the updated geometry, in turn, modifies the flow field [23]. To illustrate this distinction, Figure 2.1 visualizes the data flow in one-way and two-way coupling strategies, respectively. The diagrams highlight the unidirectional force transfer in one-way coupling, and the iterative convergence loop required for strong two-way coupling between fluid and structural solvers [22].

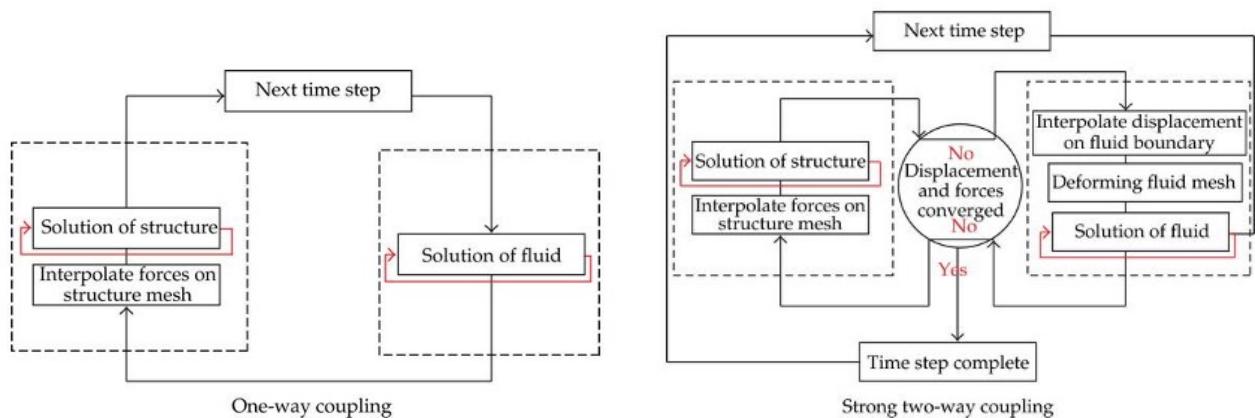


Figure 2.1: Coupling strategies in fluid-structure interaction simulations. Adapted from Benra et al. [22].

In addition to directional coupling, solver strategies can be classified into *monolithic* and *partitioned* approaches. Monolithic strategies treat the coupled system as a single, unified set of equations solved simultaneously. This often yields high stability and convergence properties, particularly in stiff interaction regimes, but comes at the cost of limited modularity and solver flexibility [24]. In contrast, partitioned approaches solve the subproblems using independent, often domain-specific solvers, which exchange data in time or post-simulation. This allows reuse of mature CFD and structural dynamics codes and simplifies implementation, at the expense of requiring designed data exchange schemes [25, 26].

Finally, the temporal aspect of coupling must be considered. *Loose (explicit)* coupling solves each domain sequentially within a time step, without sub-iterations. This can lead to instabilities in the presence of strong interdependence. In CFD, this could occur when fluid and structure have indistinguishable quantities, an issue known as the *added-mass effect* in the context of similar densities of structure and fluid [27]. To mitigate this, *tight (implicit)* coupling strategies introduce iterative sub-cycles within each time step to enforce convergence at the fluid-structure interface.

## 2.2 REEF3D

REEF3D is an open-source computational framework for hydrodynamic simulations, specifically developed for applications in coastal, marine, and hydraulic engineering [5]. The software is written in modern C++ and follows a modular, object-oriented architecture that enables the implementation and coupling of various physics models governed by different sets of equations. These include modules based on fully nonlinear potential flow model, shallow water equations, and three-dimensional incompressible Navier-Stokes equations. Each of these modules is optimized for a specific range of flow regimes, allowing users to select the most efficient solver for a given application. The framework includes several dedicated algorithms for multiphysics problems such as free surface flow, sediment transport, wave-structure interaction, and floating body dynamics. Complex geometries and natural topographies can be incorporated through immersed boundary methods (IBM) and triangulated surface meshes in stereolithographic (STL) format. In order to fully exploit modern high-performance computing capabilities, REEF3D is parallelized using the domain decomposition strategy and the Message Passing Interface (MPI). This ensures excellent scalability, making it suitable for large-scale simulations involving billions of computational cells. The modular design and open-source nature of REEF3D make it an accessible and flexible tool for both academic research and engineering practice.

The following section provides a brief overview of the four core solvers of REEF3D, each specifically designed for different hydrodynamic regimes and modeling requirements. As this work primarily employs NHFLOW, the shock-capturing non-hydrostatic Navier-Stokes solver of REEF3D, special emphasis is placed on the description of this particular module.

### 2.2.1 Overview of Numerical Modules

- **REEF3D::CFD:** Solves the incompressible Navier-Stokes equations for two-phase flow using Reynolds-Averaged (RANS) or Large Eddy Simulation (LES) numerical methods. The free surface is modeled with a level-set method, and solid structures are modeled using a ghost-cell immersed boundary method. Several turbulence models are available, including the  $k-\omega$  and  $k-\varepsilon$  closures.
- **REEF3D::SFLOW:** Solves the depth-averaged, incompressible Euler equations for shallow water flows, extended with a non-hydrostatic pressure correction to capture weakly dispersive wave phenomena. A quadratic vertical pressure profile assumption improves the accuracy of the dynamic pressure distribution, allowing for the simulation of wave propagation, wave

breaking, and bathymetric interactions. The theoretical foundations, numerical implementation, and validation are described in Wang et al. [28].

- **REEF3D::FNPF:** Solves the fully nonlinear potential flow equations for inviscid, irrotational free surface flows by employing a terrain-following  $\sigma$ -coordinate transformation. The model captures nonlinear wave propagation and wave transformations in both shallow and deep water. Wave breaking is handled via energy dissipation algorithms, and wetting-drying processes are incorporated through a level-set method. A comprehensive derivation and validation of the model can be found in the work of Wang et al. [29].
- **REEF3D::NHFLOW:** The governing equations, numerical methodology, and validation procedures are described in Section 2.2.2 and in the work of Bihs et al. [30, 31].

## 2.2.2 REEF3D::NHFLOW

### Governing Equations

REEF3D::NHFLOW solves the incompressible Euler equations with non-hydrostatic pressure correction for modeling nonlinear, dispersive shallow water waves. The governing equations consist of the continuity equation and the momentum equations without explicit diffusion terms:

$$\frac{\partial u_i}{\partial x_i} = 0, \quad (2.1)$$

$$\frac{\partial(hu_i)}{\partial t} + \frac{\partial(hu_i u_j)}{\partial x_j} = -\frac{h}{\rho} \frac{\partial p}{\partial x_i} - hg_i, \quad (2.2)$$

where  $u_i$  are the velocity components, and  $h(x, y, t) = \eta(x, y, t) + d(x, y)$  denotes the total water column height, with  $\eta(x, y, t)$  as the instantaneous free surface elevation and  $d(x, y)$  as the local still water depth. The non-hydrostatic pressure is denoted by  $p$ .  $\rho$  is the fluid density, and  $g_i$  represents the gravitational acceleration components. The free surface is represented by integrating the horizontal velocity fluxes over the vertical water column:

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial x_i} \int_{-d}^{\eta} hu_i dz = 0. \quad (2.3)$$

To efficiently handle moving free surfaces and variable bathymetry, NHFLOW employs a terrain-following  $\sigma$ -coordinate system. The transformation from the physical vertical coordinate  $z$  to the computational coordinate  $\sigma$  is defined as [32]:

$$\sigma = \frac{z + h(x, y, t)}{d(x, y)}, \quad (2.4)$$

This approach enables the mesh to dynamically adapt to free surface movements and seabed variations, facilitating wave propagation over complex topographies. The governing equations can be reformulated in vector form as:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x} + \frac{\partial \mathbf{G}(\mathbf{U})}{\partial y} + \frac{\partial \mathbf{H}(\mathbf{U})}{\partial \sigma} = \mathbf{S}, \quad (2.5)$$

considering conservative shallow water equations, the flux vectors are given by [33, 34]:

$$\mathbf{F} = \begin{bmatrix} hu^2 + \frac{1}{2}g\eta^2 + g\eta d \\ huv \\ huw \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} huv \\ hv^2 + \frac{1}{2}g\eta^2 + g\eta d \\ hvw \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} u\omega \\ v\omega \\ w\omega \end{bmatrix}, \quad (2.6)$$

Here,  $\omega$  denotes the relative vertical velocity in the  $\sigma$ -coordinate frame, defined as:

$$\omega = h \left( \frac{\partial \sigma}{\partial t} + u \frac{\partial \sigma}{\partial x} + v \frac{\partial \sigma}{\partial y} + w \frac{\partial \sigma}{\partial z} \right). \quad (2.7)$$

The source term vector  $\mathbf{S}$  incorporates the hydrostatic pressure gradient splitting, the non-hydrostatic pressure gradient, and direct forcing contributions [35, 36]:

$$\mathbf{S} = \begin{bmatrix} g\eta \frac{\partial d}{\partial x} - \frac{h}{\rho} \left( \frac{\partial p}{\partial x} + \frac{\partial p}{\partial \sigma} \frac{\partial \sigma}{\partial x} \right) + f_{x,\text{df}} \\ g\eta \frac{\partial d}{\partial y} - \frac{h}{\rho} \left( \frac{\partial p}{\partial y} + \frac{\partial p}{\partial \sigma} \frac{\partial \sigma}{\partial y} \right) + f_{y,\text{df}} \\ -\frac{h}{\rho} \frac{\partial p}{\partial \sigma} \frac{\partial \sigma}{\partial z} + f_{z,\text{df}} \end{bmatrix}. \quad (2.8)$$

## Numerical Schemes

The spatial discretization of the governing equations utilizes a Godunov-type finite volume method. At each cell face, the fluxes are reconstructed using a fifth-order Weighted Essentially Non-Oscillatory (WENO) scheme to achieve high resolution in smooth regions and avoid oscillations near discontinuities [37]. The approximate Riemann problem at each face is solved using the Harten-Lax-van Leer (HLL) Riemann solver [38].

Temporal integration is performed with a second-order Total Variation Diminishing (TVD) Runge-Kutta scheme, ensuring stability and preventing the generation of new extrema in the solution [39]. Due to the depth-integrated shallow water assumption, relatively large time steps can be employed compared to two-phase flow models. Adaptive time stepping is applied based on a Courant-Friedrichs-Lowy (CFL) criterion based on the shallow water wave speed and the maximal velocities:

$$\Delta t = \text{CFL} \cdot \min \left( \frac{\Delta x_i}{|u_i| + \sqrt{gh}} \right). \quad (2.9)$$

The non-hydrostatic pressure correction is handled via Chorin's projection method [40]. First, a provisional velocity field is computed, neglecting the non-hydrostatic pressure gradient. Subsequently, the non-hydrostatic pressure correction is obtained by solving a Poisson equation. The Poisson equation is discretized using second-order central finite differences and solved using the parallelized BiCGStab solver of the HYPRE library with a geometric multigrid PFMG preconditioner to enhance convergence rates [41].

To support this approach, REEF3D employs a staggered grid [42]. The pressure is stored at cell centers, while the three velocity components are defined on the respective cell faces they pass through. This arrangement improves numerical stability, prevents pressure oscillations, and enables conservative flux computations in the projection step. It also aligns with the structure of the projection method itself, and facilitates accurate reconstruction of the pressure-velocity coupling, especially in the presence of a dynamic free surface [42].

Parallelization across multiple processors is achieved using the Message Passing Interface (MPI) with a domain decomposition strategy. Three layers of ghost cells are introduced at each subdomain boundary to accommodate the fifth-order WENO stencil width.

## Immersed Boundary Method

Solid structures are embedded into the computational domain using a direct forcing immersed boundary method [36]. A signed-distance function derived from a triangulated surface mesh represents the geometry. Due to the dynamic nature of the  $\sigma$ -grid, the signed-distance function is updated at

each time step using an optimized ray-casting algorithm based on the point-in-polygon test [43]. Hydrodynamic forces are calculated by integrating the total pressure  $p_{\text{tot}}$  over the wetted surface:

$$F = \int_A p_{\text{tot}} dA, \quad (2.10)$$

where the total pressure combines hydrostatic and non-hydrostatic components:

$$p_{\text{tot}} = p_{\text{nh}} + p_{\text{hs}}, \quad (2.11)$$

$$p_{\text{hs}} = (d + \eta - z)\rho g. \quad (2.12)$$

## Validation

The REEF3D::NHFLOW module has been validated against several experimental benchmarks, demonstrating its capability to predict hydrodynamic forces under both non-breaking and breaking wave conditions:

- Dam-break flow interaction with an obstacle, based on the experiments conducted by Tso-Ren Wu [44],
- Non-breaking wave forces on vertical cylinders, based on the experimental study by Zang et al. [45],
- Breaking wave impacts measured in large-scale experiments at the Large Wave Flume (GWK) in Hannover by Irschik et al. [46].

The combination of accuracy and efficiency, induced by solving the shock-capturing Navier-Stokes equations, makes NHFLOW particularly suitable for coupling strategies in the system-level analysis of FOWTs. Compared to other available solvers in the REEF3D framework, NHFLOW exhibits advantageous properties for the application as input generator in hydrodynamic coupling tools involving OpenFAST. While REEF3D::SFLOW is computationally efficient, its vertically averaged formulation cannot resolve vertical flow variations. This is relevant for capturing kinematic and dynamic wave properties correctly across the submerged structure of an FOWT. Potential flow solvers like REEF3D::FNPF offer computationally inexpensive simulations and can handle nonlinear wave effects, but lack intrinsic pressure field algorithms. However, OpenFAST requires pressure data, in order to estimate dynamic loads (see Section 2.3.3). NHFLOW, in contrast, offers access to non-hydrostatic pressure and velocity fields throughout the water column, at lower cost than full turbulence-resolving CFD solvers like REEF3D::CFD. These characteristics make NHFLOW the most balanced and applicable choice for hydrodynamic coupling with OpenFAST within the REEF3D framework as they allow integration of complex, nonlinear wave dynamics into OpenFAST simulations while preserving its inherent computational inexpensiveness.

## 2.3 OpenFAST

OpenFAST is an open-source, multiphysics simulation tool developed for the analysis of land-based and offshore wind turbines [10]. OpenFAST was developed by NREL and is designed in a modular architecture, primarily written in Fortran. Originally evolving from the earlier FAST (Fatigue, Aerodynamics, Structures, and Turbulence) tool, OpenFAST extends its predecessor by adopting a flexible, object-oriented structure that enables coupling of various physical domains through dedicated modules. The main objective is to provide a high-fidelity yet computationally efficient framework that facilitates both academic research and industrial design studies. At its core, OpenFAST solves the fully coupled aero-hydro-servo-elastic system equations of motion, supporting a wide array of

modeling options for aerodynamic loads, hydrodynamic forces, structural dynamics, control systems, and mooring dynamics. It is capable of handling complex offshore configurations such as floating wind turbines, semi-submersibles, and spar platforms, in addition to traditional onshore turbines. The modular structure of OpenFAST ensures that each physical process is encapsulated within a dedicated module, governed by its own set of equations and assumptions. Modules interact via a centralized coupling algorithm, enabling the exchange of forces, moments, and kinematic states at each time step. This architecture promotes flexibility, allowing users to replace, disable, or extend modules depending on the intended application. A typical OpenFAST simulation involves the implementation of several key modules:

- **AeroDyn:** Calculates aerodynamic loads on the rotor blades and tower based on blade element momentum (BEM) theory or the generalized dynamic wake (GDW) model. It includes corrections for tip/hub losses, dynamic inflow, and skewed wakes. Additionally, it implements the Beddoes-Leishman dynamic stall model and accounts for tower shadow effects [47, 48].
- **HydroDyn:** Computes hydrodynamic forces acting on offshore structures. It supports potential theory through boundary element methods or Morison-type load formulations for slender structures [49, 50].
- **ElastoDyn:** Solves the structural dynamics of wind turbine components, including flexible blades, flexible towers, and rigid body motions of the platform. It uses a nonlinear multibody dynamics formulation with nodal representations for flexible members [51].
- **InflowWind and SeaState:** Define the environmental conditions acting on the structure. InflowWind generates wind fields for aerodynamic loading, while SeaState generates wavefields and current profiles for hydrodynamic loading [52, 49, 53].
- **ServoDyn:** Models turbine control systems including blade pitch, generator torque, and yaw control. ServoDyn supports both built-in open- and closed-loop control schemes as well as user-supplied controllers through interface routines [54].
- **MoorDyn or SubDyn:** MoorDyn models the dynamics of mooring lines using a lumped-mass approach, including tension, drag, and inertia effects, while SubDyn models the structural response of fixed-bottom substructures based on linear modal dynamics [55, 56, 57].

By combining these specialized modules, OpenFAST provides a multiphysics simulation environment capable of capturing the highly nonlinear and coupled behavior of wind turbines under realistic operating conditions.

In the following sections, the key physical modeling capabilities of OpenFAST are described. The description is organized according to the main physical processes required for the scope of this work, namely, mooring system dynamics, wave kinematics, hydrodynamic load computation, and platform motions. For a comprehensive user guide tailored to the use of OpenFAST in the context of this work, the reader is referred to the documentation available within the REEF2FAST GitHub repository.

### 2.3.1 Mooring System Modeling

Mooring within OpenFAST is modeled with a nonlinear stress-strain relationship that accounts for the elastic and damping behavior of synthetic mooring lines. Each mooring line is represented as a series of lumped point masses (nodes) connected by spring-damper segments. The total force  $\vec{f}_i$  acting at a node includes contributions from weight and buoyancy, axial tension and damping, hydrodynamic

drag, and seabed contact forces. Tension in a line segment is evaluated at midpoints between nodes, based on the strain  $\epsilon_{i+\frac{1}{2}}$  and strain rate  $\dot{\epsilon}_{i+\frac{1}{2}}$  using [55]:

$$T_{i+\frac{1}{2}} = \frac{\pi d^2}{4} \left( E(\epsilon) \epsilon_{i+\frac{1}{2}} + B(\dot{\epsilon}) \dot{\epsilon}_{i+\frac{1}{2}} \right) \quad (2.13)$$

where  $E(\epsilon)$  is the elasticity modulus and  $B(\dot{\epsilon})$  is a structural damping coefficient. These properties can be provided via user-defined nonlinear look-up tables, allowing flexible modeling of complex material behavior. Additionally, 6-DoF elements called *Rods* and *Bodies* are included, enabling the modeling of submerged structures such as floats or weights [57]. These are governed by the full rigid-body equations of motion:

$$\mathbf{M}_{6 \times 6} \ddot{\vec{r}} = \vec{f}_{\text{net}} \quad (2.14)$$

The mass matrix  $\mathbf{M}_{6 \times 6}$  consists of three distinct  $3 \times 3$  tensors:

$$\mathbf{M}_{6 \times 6} = \begin{bmatrix} \mathbf{M} & \mathbf{J} \\ \mathbf{J}^\top & \mathbf{I} \end{bmatrix} \quad (2.15)$$

where  $\mathbf{M}$  is the translational mass matrix,  $\mathbf{I}$  the rotational inertia tensor, and  $\mathbf{J}$  accounts for the coupling between translation and rotation. Coordinate transformations of these tensors are applied via tensor transformation rules when the reference frame is offset from the center of mass [58]. The overall mooring system is solved in a top-down hierarchy, where motion is propagated from the floating structure to the attached lines, and reaction forces are accumulated in reverse. Mooring line failures can also be modeled dynamically by introducing massless connection points that decouple the line upon exceeding a tension threshold [55].

Mooring dynamics are implemented within the MoorDyn module, which provides a robust and extensible framework for simulating advanced mooring systems in floating wind turbine simulations [57].

### 2.3.2 Wave Kinematics

Since OpenFAST v4, wave kinematics and currents are computed by the newly introduced, independent SeaState module. Previously, environmental hydrodynamic conditions were handled within the HydroDyn module. SeaState provides time-varying velocity, acceleration, non-hydrostatic pressure, and elevation fields at discretized user-defined locations [53]. These kinematic quantities are subsequently used by HydroDyn to evaluate hydrodynamic forces based on potential theory or Morison-type formulations [49]. Internally, OpenFAST supports several wave theories for the generation of wave kinematics:

- Regular linear waves based on Airy theory
- JONSWAP or Pierson-Moskowitz spectra, which represent empirical wave energy spectra of irregular sea states
- White-noise spectra for irregular waves with a flat energy distribution across frequencies

While nonlinear high-order waves beyond first-order wave theory cannot inherently be modeled, second-order wave effects can optionally be implemented through the superposition of sum- and difference-frequency Quadratic Transfer Functions (QTFs), which capture nonlinear wave-wave interactions. Difference-frequency QTFs represent low-frequency load contributions resulting from interactions between closely spaced wave frequencies. These effects are particularly relevant for modeling slow-drift motions in surge, sway, and yaw. In contrast, sum-frequency QTFs capture high-frequency load components that can lead to structural springing and ringing responses. To prevent

the inclusion of unphysical contributions, frequency cutoffs can be defined for both difference- and sum-frequency QTFs [49].

The SeaState module precomputes the full three-dimensional wavefield on a Cartesian grid defined by user-specified domain dimensions and discretization parameters. The vertical direction employs a cosine-stretched node distribution to cluster points near the still water level (SWL), where gradients in wave kinematics are most pronounced. The vertical node positions  $z_i$  are computed as:

$$z_i = (\cos(i \cdot \Delta\theta) - 1) h, \quad \text{with} \quad \Delta\theta = \frac{\pi}{2(N_Z - 1)}, \quad (2.16)$$

where  $N_Z$  is the number of nodes in the vertical direction. Notably, the vertical grid reaches from the flat seabed ( $z = -z_{\max}$ ) only up until the SWL ( $z = 0$ ). During the simulation, the module interpolates the precomputed wave kinematics to the positions of the hydrodynamic nodes, enabling efficient and consistent evaluation of wave-induced loads [50]. In the context of this work, it is important to note that SeaState supports only rectangular spatial domains with a constant seabed bathymetry. Additionally, the internal wave modeling framework assumes idealized wave shapes and lacks feedback from the platform motion on the wavefield.

### 2.3.3 Hydrodynamic Loads and Platform Motions

In OpenFAST, the hydrodynamic response of floating structures is computed within the HydroDyn module, which solves for wave-induced forces and moments acting on the structure. These loads are then coupled with the structural dynamics solver ElastoDyn to predict the platform's 6-DoF motion. Two alternative formulations are available for hydrodynamic load calculation: Morison-type strip theory and potential theory based on precomputed frequency domain data [50].

#### Strip theory

In the strip theory formulation, the submerged structure is divided into horizontal strip-elements. For each strip, the wave-induced hydrodynamic force per unit length  $\vec{f}_{\text{strip}}$  is calculated using the Morison equation:

$$\vec{f}_{\text{strip}} = \rho A C_M \frac{\partial \vec{u}_{\text{rel}}}{\partial t} + \frac{1}{2} \rho C_D D |\vec{u}_{\text{rel}}| \vec{u}_{\text{rel}}, \quad (2.17)$$

where  $\vec{u}_{\text{rel}} = \vec{u}_{\text{wave}} - \vec{u}_{\text{body}}$  is the relative fluid velocity,  $A$  the cross-sectional area of the strip,  $D$  the diameter, and  $C_M$ ,  $C_D$  the added mass and drag coefficients, respectively. To obtain the total hydrodynamic load on the body,  $\vec{f}_{\text{strip}}$  is integrated over the submerged length:

$$\vec{F}_{\text{hydro}} = \sum_{i=1}^{N_{\text{strip}}} \int_{z_i}^{z_{i+1}} \vec{f}_{\text{strip}}(z) dz. \quad (2.18)$$

The validity of the strip theory approach has been assessed through experimental validation. Carmo et al. [6] compared simulation results from OpenFAST against measurement data obtained during the FOCAL Campaign at the University of Maine. Their work focused on the prediction of local structural loads at the pontoon roots of an FOWT under regular and irregular wave conditions. By employing a Morison-based strip theory model, they demonstrated that OpenFAST can reproduce key load characteristics with good agreement, especially for axial forces and bending moments [20].

#### Potential theory

In the alternative potential theory formulation, OpenFAST utilizes hydrodynamic coefficients pre-computed by external diffraction/radiation solvers such as WAMIT [59, 3]. These include frequency-dependent added mass  $\mathbf{A}(\omega)$ , and radiation damping  $\mathbf{B}(\omega)$  matrices and excitation force transfer

functions  $\vec{F}_{\text{exc}}(\omega)$ . The hydrodynamic force is calculated in the time domain via the Cummins equation [50]:

$$\vec{F}_{\text{hydro}}(t) = \vec{F}_{\text{exc}}(t) - \int_0^t \mathbf{K}(t-\tau) \dot{\vec{\xi}}(\tau) d\tau - \mathbf{A}_\infty \ddot{\vec{\xi}}(t), \quad (2.19)$$

where  $\mathbf{K}(t)$  is the impulse response function (IRF) of the radiation damping and  $\mathbf{A}_\infty$  the added mass at infinite frequency. In OpenFAST, the excitation forces are interpolated over displaced platform positions to account for phase shifts induced by large displacements. This enhances accuracy in nonlinear wave regimes [49].

### Rigid-body dynamics

The 6-DoF platform motion is governed by Newton-Euler equations:

$$\mathbf{M}_{\text{RB}} \ddot{\vec{\xi}} + \mathbf{C} \dot{\vec{\xi}} + \mathbf{K} \vec{\xi} = \vec{F}_{\text{aero}} + \vec{F}_{\text{hydro}} + \vec{F}_{\text{moor}} + \vec{F}_{\text{other}}, \quad (2.20)$$

where  $\vec{\xi} = [x, y, z, \phi, \theta, \psi]^\top$  is the 6-DoF platform displacement vector (surge, sway, heave, roll, pitch, yaw),  $\mathbf{M}_{\text{RB}}$  the rigid-body mass and inertia matrix, and  $\mathbf{C}, \mathbf{K}$  optional damping and restoring terms. The coupling between translational and rotational motions is reflected in the full  $6 \times 6$  mass and damping matrices. These terms can be non-diagonal due to off-axis loads or hydrodynamic cross-coupling. The calculated motions  $\vec{\xi}(t)$  are fed back into the hydrodynamic load calculations through updated (relative) velocities and positions, enabling consistent and fully coupled time-domain simulations (see Section 2.3.4).

Several additional modeling features are available within HydroDyn to improve the robustness and physical fidelity of the hydrodynamic force computation. One such feature is load smoothing at the wet-dry interface, which mitigates unphysical force spikes caused by sudden transitions between submerged and non-submerged states of structural elements. Another enhancement is the MacCamy-Fuchs correction, which modifies the added mass term in the Morison equation to account for diffraction effects in short waves and for large-diameter members. The corrected inertia coefficient is given by [49]:

$$C_M^{\text{corr}} = 1 + \frac{2J_1(kD/2)}{kD/2}, \quad (2.21)$$

where  $J_1$  is the Bessel function of the first kind and  $k$  is the wave number. This correction improves the accuracy of inertia force estimation when the element diameter becomes comparable to the wavelength.

Within the scope of this project, only the strip theory is employed to compute the hydrodynamic force vector  $\vec{F}_{\text{hydro}}$ , due to limitations imposed by OpenFAST's ability to read-in externally generated wavefields.

#### 2.3.4 Internal Coupling Architecture of OpenFAST

As described, OpenFAST is designed as a modular, multiphysics simulation framework in which individual subdomains are represented by dedicated solver modules. These modules are coupled through a centralized interface known as the *Glue Code*, which governs the temporal coordination, data exchange, and solution synchronization across the system [60]. Each module in OpenFAST operates as an independent component implementing a standardized set of interface routines. These routines are invoked in a prescribed order at every time step by the Glue Code, ensuring consistent data flow and proper initialization and shutdown behavior. The simulation proceeds in discrete time steps, each of which includes the following core sequence per iteration [10]:

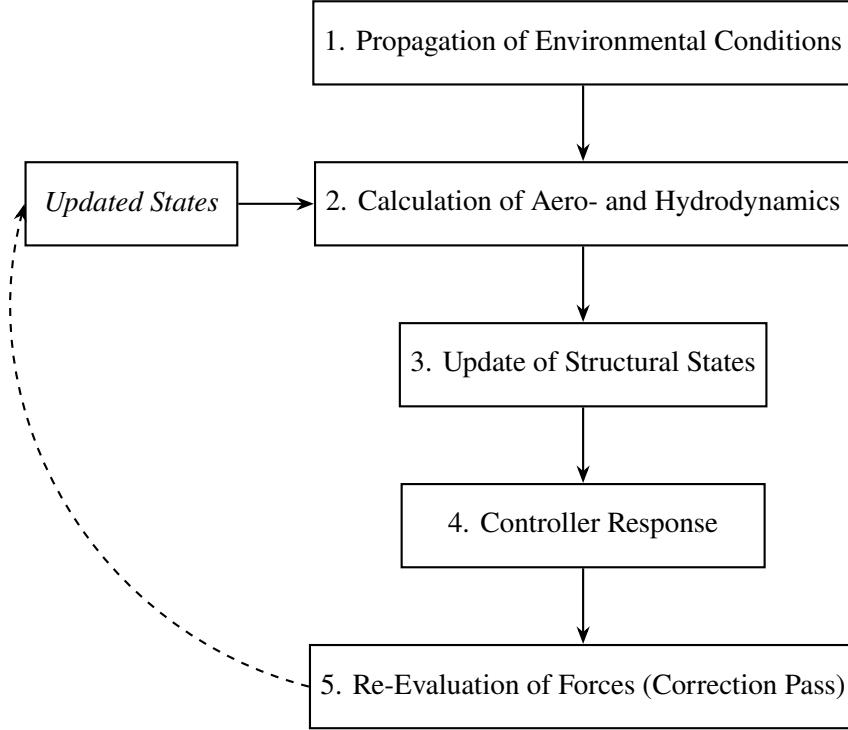


Figure 2.2: Internal module sequence within a single OpenFAST time step. The correction pass uses updated structural and control states to refine the computed forces.

At the heart of this architecture is a loosely coupled, partitioned approach in which each module solves its own governing equations independently. Communication between modules occurs once per time step through shared data structures such as the *FAST\_InputType* and *FAST\_OutputType*, which are populated and exchanged by the Glue Code. Although the solver sequence is sequential, OpenFAST performs a second *CalcOutput* evaluation after all module states have been updated. This correction step ensures that interdependencies, for example, aerodynamic forces depending on structural motion, are consistently applied within the same time level [10]. The modules mostly employ explicit time integration, using fixed-step time advancement schemes. However, due to the correction pass and consistent reuse of shared state variables, the internal coupling approximates a weakly implicit behavior, improving stability without incurring the computational cost of full sub-iterations [60]. Notably, the internal data exchange does not involve a monolithic solution of the full system. Instead, the design emphasizes modularity and extensibility. For example, the structural solver ElastoDyn computes rigid 6-DoF motion of the turbine based on hydrodynamic and aerodynamic loads, while passing the updated motion back to the force modules in the next iteration. Similarly, MoorDyn and SubDyn return constraint forces to ElastoDyn, which are treated as generalized loads in the platform equations of motion. The time step across all modules is synchronized, and OpenFAST assumes a constant global time step that governs the pace of the simulation. All modules are evaluated at the same physical time level, with no sub-cycling. However, some modules (e.g., AeroDyn with dynamic stall models) implement their own internal time-history buffers to account for time-delay effects.

Overall, OpenFAST adopts a hybrid coupling approach: while fundamentally partitioned and loose, it includes correction mechanisms that capture important interdependencies. This allows OpenFAST to retain the flexibility of modular solvers while maintaining sufficient physical consistency across tightly coupled subsystems.

# Chapter 3

## Methodology

### 3.1 REEF2FAST

The REEF2FAST pre-processor is a standalone, file-based coupling tool developed to integrate high-fidelity wave kinematics computed by REEF3D::NHFLOW into the OpenFAST simulation framework. Its primary purpose is to enable hydrodynamically consistent, high-resolution structural simulations of FOWTs by allowing OpenFAST to utilize externally generated wavefields. The method follows a one-way, partitioned coupling strategy as outlined in Section 2.1, allowing independent solver execution while maintaining physical consistency through strict time synchronization and robust data transformation. The need for REEF2FAST arises from the limitations of OpenFAST, which natively supports only simplified wave theories, as shown in Section 2.3.2. By incorporating wavefields computed by NHFLOW, capable of capturing nonlinear transformations, complex bathymetry, and wave-structure interactions (see Section 2.2), REEF2FAST enables OpenFAST to perform simulations based on more realistic hydrodynamic inputs. Moreover, the system is designed to be lightweight and scalable: its memory usage is independent of simulation duration, making it feasible to post-process long CFD sequences on hardware with limited resources. The internal processing follows a streaming concept, where time steps are handled sequentially using a sliding memory buffer.

The decision to design REEF2FAST as a file-based, one-way coupling pre-processor comes from the modular philosophy of OpenFAST itself. Since OpenFAST internally operates mostly with loosely coupled subsystems that exchange information via interface modules and apply correction mechanisms rather than relying on monolithic solvers (see Section 2.3.4), the NHFLOW-OpenFAST coupling with REEF2FAST adopts the same philosophy. It effectively replaces the wave kinematics provided by SeaState with externally computed hydrodynamic inputs from NHFLOW, while maintaining the existing coupling strength between the hydrodynamic conditions, the glue code, and the rest of the OpenFAST modules. This avoids the added complexity and computational cost that would result from stronger coupling approaches such as fully implicit or monolithic schemes. Consequently, the low computational expenses of REEF2FAST’s coupling strategy aligns with the general approach of OpenFAST’s simulation philosophy.

The following section outlines the general architecture of REEF2FAST and provides a description of the theoretical foundations underlying its core functionalities. For practical guidance on software usage and configuration, the reader is referred to the user documentation available in the official REEF2FAST GitHub repository.

#### 3.1.1 Software Architecture

REEF2FAST is implemented in modern C++17 and structured as a modular command-line application. Its architecture centers around a streaming pipeline that operates on three consecutive time steps: the

current step and its immediate neighbors (i.e.,  $t-1$ ,  $t$ , and  $t+1$ ). This structure enables temporal differentiation and interpolation while keeping memory usage bounded by the spatial resolution rather than the simulation duration. The pipeline consists of several modules, each responsible for a specific computational task:

- **File Reader:** Streamingly loads REEF3D::NHFLOW output data from CSV files, supporting both two- and three-dimensional wavefields.
- **Wheeler Projection:** Applies vertical stretching to capture NHFLOW wavefield quantities located outside the vertical SeaState domain.
- **Interpolator:** Applies a k-d tree-based nearest-neighbor scheme to project scattered  $\sigma$ -grid point-cloud data from NHFLOW onto the structured Cartesian SeaState grid used by OpenFAST.
- **Surface Elevation Mapper:** Derives OpenFAST compatible surface elevation arrays, using either NHFLOW’s vertical coordinate  $z$  or its surface elevation  $\eta$ .
- **Acceleration Estimator:** Computes particle accelerations using finite difference schemes.
- **Exporter:** Writes SeaState-compatible output files, including three-dimensional velocities and accelerations, elevation, and non-hydrostatic pressure arrays.

All computational kernels are thread-safe. The interpolation stage is parallelized using OpenMP [61]. Internal data structures are memory-aligned to reduce fragmentation and improve cache efficiency. Input/output (I/O) operations are buffered and dynamic memory allocations are minimized to enhance performance. REEF2FAST reads its configuration from REEF3D control files and expects wavefield data (at least velocity and non-hydrostatic pressure at grid points) as a time-resolved series, combined in one CSV file. The CSV file can be exported from REEF3D using the *vtk* python package or *Paraview*.

The wavefield file is processed stream-wise, and the output is written in a format directly compatible with OpenFAST, requiring no further user modification. For large-scale simulations, a two-dimensional mode is available, which exploits geometric symmetry along the  $y$ -axis. In this mode, the interpolation is performed on a single  $x$ - $z$  slice and duplicated across the transverse direction, significantly reducing both runtime and memory usage. This is only applicable, when lateral variation is negligible. The main algorithmic sequence for each time step proceeds as follows:

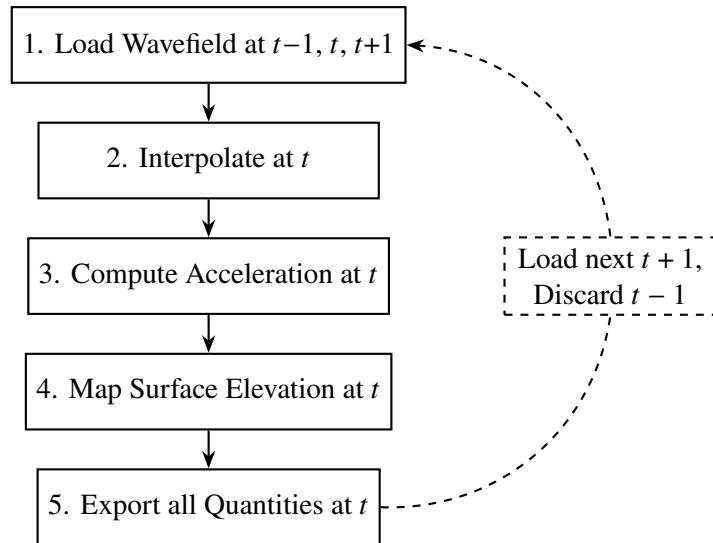


Figure 3.1: Processing loop within REEF2FAST for each simulation time step. All modules operate sequentially within a streaming buffer before advancing to the next time step.

This sequence is repeated for each time step until the entire wavefield has been processed. Notably, the current step  $t$  becomes  $t-1$  in the next iteration,  $t+1$  becomes  $t$ , and the previous  $t-1$  is discarded from memory, as it is no longer needed.

### 3.1.2 Interpolation

The interpolation module in REEF2FAST maps scattered wavefield data exported by REEF3D onto the structured Cartesian grid required by OpenFAST's SeaState module. While NHFLOW outputs are defined on a dynamic, non-orthogonal  $\sigma$ -grid ( $\sigma = \sigma(\vec{x}, t)$ ), that follows the instantaneous free surface (see Equation 2.4), SeaState operates on a static grid that extends from the seabed up to the SWL, featuring uniform horizontal spacing and a vertically stretched distribution following a cosine function, as described in Section 2.3.2. This difference in spatial discretization is illustrated in Figure 3.2, which shows a cross-section in the  $x$ - $z$  plane. Throughout the remainder of this thesis, terminology from multidimensional interpolation theory is used as follows: the term *source grid* refers to REEF3D::NHFLOW's  $\sigma$ -grid, while *target grid* denotes the rigid Cartesian SeaState grid used in OpenFAST. Interpolation is performed by projecting (scalar) fields defined on the source grid onto the locations specified by the target grid.

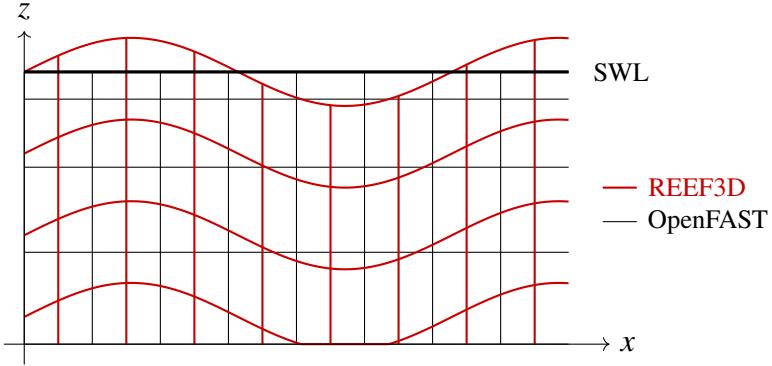


Figure 3.2: Illustrative  $x$ - $z$  cross-section of the two grids. *Red:* REEF3D::NHFLOW source grid: dynamic  $\sigma$ -grid that follows the free surface. *Black:* OpenFAST target grid: Cartesian grid with vertical stretching, defined up to the SWL.

#### Nearest Neighbor Search

To interpolate scattered wavefield data from the source grid onto the target grid, REEF2FAST requires an efficient method for identifying spatially close data points for each target location. This task is commonly referred to as a *nearest neighbor search* in computational geometry. Given a set of  $N$  points in a  $k$ -dimensional space and a query point  $\mathbf{x} \in \mathbb{R}^k$ , the goal is to identify the  $n$  source points closest to  $\mathbf{x}$  in terms of Euclidean distance. In REEF2FAST, this operation is performed in three-dimensional space ( $x, y, z$ ), with  $n = 8$  neighbors. The choice of  $n = 8$  is motivated by the number of surrounding nodes used in classical trilinear interpolation, which also considers the eight nearest vertices of a cube in structured grids. This choice provides a physically meaningful basis for the weighting scheme (see Eq. 3.1). For two-dimensional simulations, the number of neighbors is reduced to  $n = 4$ , reflecting the lower spatial complexity.

Standard interpolation techniques, such as trilinear or spline-based interpolation, assume a regular or at least time-independent grid. However, REEF3D::NHFLOW outputs its wavefields on a dynamic, non-orthogonal  $\sigma$ -grid that deforms with the instantaneous free surface (see Figure 3.2). Thus, the source grid lacks fixed topology and may vary in shape at each time step. As a result, the interpolation algorithm in REEF2FAST does not attempt to infer the underlying source grid logic. Instead, it treats the NHFLOW output data as a time-varying, unstructured point cloud. Consequently, a new nearest

neighbor search must be performed independently for each time step, and a new spatial index must be constructed for the evolving point distribution.

To address this, REEF2FAST leverages a spatial data structure known as a *k-d tree* (short for *k*-dimensional tree), which organizes the source points in a hierarchical, axis-aligned binary partitioning of space. By recursively splitting the point set along alternating coordinate axes, the k-d tree enables logarithmic query times on average. While a brute-force nearest neighbor search scales with  $\mathcal{O}(N)$ , the k-d tree reduces the complexity to  $\mathcal{O}(\log N)$  per query [62]. A visual example of a recursive two-dimensional partitioning is shown in Figure 3.3, where alternating vertical and horizontal splits divide the point cloud into increasingly localized search regions. REEF2FAST utilizes the open-source *nanoflann* header-only library to build and query a static k-d tree over the source point cloud at each time step. The tree construction is based on the spatial coordinates of the data, while the actual values (e.g., velocity components) are stored separately. For each target grid point in the domain, the tree is queried to retrieve the eight nearest points from the source grid. These neighbors then form the local support for the interpolation step. To ensure that the true k-nearest neighbors are found, the search algorithm employs backtracking. Once the query reaches a leaf node, the algorithm may traverse back up the tree and explore sibling branches if there is a possibility that closer points exist there. This mechanism guarantees that all candidate regions are considered, even if the optimal point lies outside the initially followed subtree. The entire search-algorithm is parallelized across all target points to ensure scalability with increasing grid resolution.

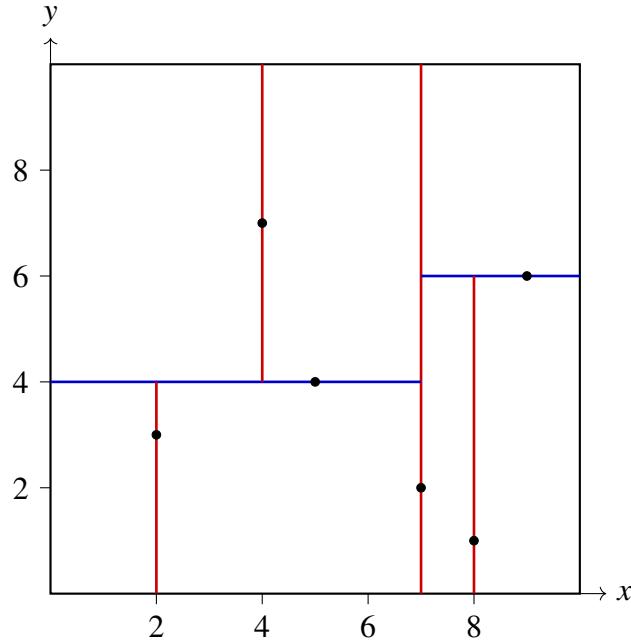


Figure 3.3: Recursive partitioning of a two-dimensional point cloud using a k-d tree. *Red*: vertical (x-axis) splits. *Blue*: horizontal (y-axis) splits.

### Inverse Distance Weighting

Once the  $n$  nearest neighbors of a target point have been identified, REEF2FAST performs spatial interpolation using *Inverse Distance Weighting (IDW)* [63]. This approach is widely used in geostatistics and numerical modeling to interpolate scalar fields from discrete, irregularly spaced data.

The core idea of IDW is that the influence of each source point decreases with distance: nearby points contribute more to the interpolated value than distant ones. Mathematically, the interpolated value  $\tilde{f}(\vec{x})$  at a target location  $\vec{x} \in \mathbb{R}^k$  is computed as a weighted average of the values  $f_i$  of the  $n$  nearest neighbors located at  $\vec{x}_i$ :

$$\tilde{f}(\vec{x}) = \frac{\sum_{i=1}^n w_i f_i}{\sum_{i=1}^n w_i}, \quad \text{with } w_i = \frac{1}{\|\vec{x} - \vec{x}_i\| + \varepsilon} \quad (3.1)$$

Here,  $\|\vec{x} - \vec{x}_i\|$  denotes the Euclidean distance (in  $\mathbb{R}^k$  with  $k = 2, 3$ ) between the target  $\vec{x}$  and the  $i$ -th neighbor, and  $\varepsilon$  is a small positive constant ( $\varepsilon = 10^{-8}$ ) added for numerical stability. The IDW scheme yields a convex combination of the input values. The weights are defined as the inverse of the Euclidean distance ( $w_i \geq 0$ , per definition), and the interpolated value is computed by normalizing the weighted sum with the total weight as shown in Equation 3.1. This ensures that the interpolated value lies between the minimum and maximum of the neighbor values:

$$\min(f_i) \leq \tilde{f}(\vec{x}) \leq \max(f_i), \quad \forall i = 1, \dots, n.$$

However, this does not imply that the interpolation point  $\vec{x}$  itself lies within the convex hull of the source points  $\vec{x}_i$ . As a result, IDW may still behave like an extrapolator in edge regions or strongly anisotropic data distributions, potentially introducing artifacts if gradients are steep.

In summary, IDW combined with the k-d tree nearest neighbor search, avoids the need for complex mesh generation or element connectivity, while still producing smooth and physically reasonable interpolation surfaces in highly dynamic and unstructured point clouds.

**Note.** Only a spatial interpolation is necessary. No temporal interpolation is applied, as REEF3D exports wavefields at equidistant time steps in the VTU files. This constant time step is read by REEF2FAST and adopted as the temporal resolution in the subsequent OpenFAST simulation. Moreover, only the non-hydrostatic pressure and the three-dimensional velocity fields are interpolated explicitly using the presented algorithm. The mapping of the surface elevation follows the procedure described in Section 3.1.4.

### 3.1.3 Wheeler Projection

A key challenge in coupling NHFLOW wavefields to OpenFAST arises from the vertical mismatch between the two spatial grids. As shown in Figure 3.2, the target grid defines its vertical domain only up to the SWL, whereas NHFLOW resolves kinematic quantities throughout the full free-surface region, including wave crests that rise above the SWL. Consequently, NHFLOW source points located at  $z > \text{SWL}$  lie outside the SeaState domain and are potentially excluded from the interpolation, since they may not be the nearest neighbor of any point on the OpenFAST target grid or may contribute less because they have a low weight  $w$ . This leads to an underrepresentation of areas with strong velocity and pressure gradients that are physically important for capturing crest dynamics.

To resolve this issue, REEF2FAST applies a vertical projection technique, called *Wheeler stretching* [64]. Rather than extrapolating the field values, this approach solely shifts the spatial location of source points above the SWL vertically downward into the SeaState domain. For a point located at  $z = h + \eta(\vec{x}, t)$ , where  $\eta > 0$ , the projected coordinate  $z'$  is computed as:

$$z' = h \frac{h + z}{h + \eta} - h \quad (3.2)$$

This equation ensures that the vertical structure of the wave crest is proportionally compressed into the sub-SWL region, making it accessible for interpolation. By doing so, the algorithm preserves high-resolution crest information and avoids undersampling in dynamically critical regions. Since REEF2FAST employs nearest-neighbor- and IDW-based interpolation without relying on a specific

source grid topology, points projected below the SWL simply act as valid neighbors for target grid nodes during the interpolation process. The projection is visualized in Figure 3.4.

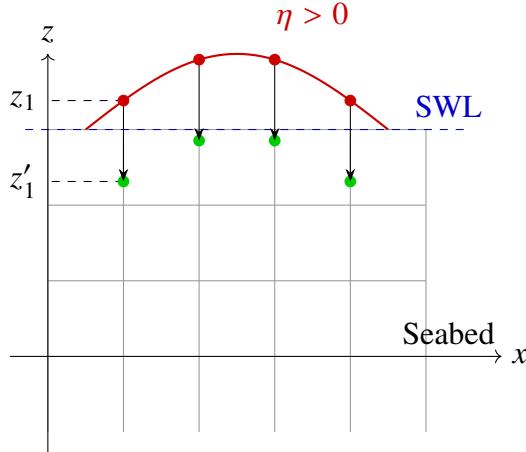


Figure 3.4: Wheeler stretching: REEF3D::NHFLOW points above SWL are projected below SWL to retain crest kinematics during interpolation.

### 3.1.4 Surface Elevation

In REEF3D and OpenFAST, the concept of surface elevation  $\eta$  differs. In REEF3D::NHFLOW's VTU data, each point in the computational volume is assigned a surface elevation field value  $\text{eta} = \eta(\vec{x}, t)$ , even if the point lies below the free surface. Like the vertical coordinate  $z$ ,  $\eta$  evolves over time as the  $\sigma$ -grid deforms with the instantaneous wave shape. As a result, two different quantities ( $\eta$  or  $z$ ) are available at every point in the domain and can both be interpreted as vertical indicators of the state of the instantaneous free-surface. In contrast, OpenFAST defines surface elevation only at the free surface, as a two-dimensional scalar field over the horizontal plane,  $\eta(x, y, t)$ . Accordingly, REEF2FAST must condense the volumetric REEF3D surface elevation data into a surface-aligned format that matches OpenFAST's expectation. To achieve this, REEF2FAST identifies the topmost value of  $\eta$  (or alternatively  $z$ ) for each unique  $x$ - $y$  pair in the REEF3D wavefield at every time step. This corresponds to finding the free-surface value in a vertical column aligned with the horizontal location  $(x, y)$ :

$$\eta_{\max}(x, y, t) = \begin{cases} \max \{\eta(x, y, z, t) \mid z \in [z_{\min}, z_{\max}]\}, & \eta\text{-based} \\ \max \{z \mid [x, y, z] \in \Omega_{\text{reef}}\}, & z\text{-based} \end{cases} \quad (3.3)$$

with  $\Omega_{\text{reef}}$ , the source grid domain. Once extracted, the scattered  $\eta_{\max}(x, y, t)$  data is interpolated onto the SeaState grid using a two-dimensional k-d tree and IDW, analogous to the method described in Section 3.1.2, but limited to the horizontal plane. Directly interpolating all volumetric  $\eta$ -values onto the target grid would be computationally inefficient and unnecessary.

Both methods have proven robust in simulations and are functionally equivalent for smoothly resolved surfaces in OpenFAST. For consistency and fast computation, the user selects one of the two approaches per run.

### 3.1.5 Acceleration

For a full description of the wavefield, OpenFAST requires not only wave velocities but also the corresponding particle accelerations at each grid point. Since NHFLOW can only directly generate velocity output data, REEF2FAST computes accelerations via numerical differentiation of the interpolated velocity field. This is done for each point  $\vec{x}$  in the target grid and at each time step  $t$ . To estimate the

time derivative  $\dot{u}_i(\vec{x}, t) = \frac{du_i(\vec{x}, t)}{dt}$ , REEF2FAST applies finite difference approximations depending on the position within the time series:

$$\dot{u}_i(\vec{x}, t) \approx \begin{cases} \frac{u_i^1 - u_i^0}{\Delta t}, & t = 0 \quad (\text{forward difference}) \\ \frac{u_i^{t+1} - u_i^{t-1}}{2\Delta t}, & 1 \leq t \leq N-1 \quad (\text{central difference}) \\ \frac{u_i^N - u_i^{N-1}}{\Delta t}, & t = N \quad (\text{backward difference}) \end{cases} \quad (3.4)$$

with  $i \in \{x, y, z\}$ . Consequently, whenever applicable, a central difference scheme is used. This choice of scheme ensures second-order accuracy in the interior of the time domain while maintaining stability and simplicity at the boundaries.

## 3.2 Verification

In order to evaluate whether wave kinematic data can be accurately transferred from REEF3D to OpenFAST via the REEF2FAST coupling interface, a series of three verification cases is constructed. Each case is initially modeled in NHFLOW to generate hydrodynamic input fields. These fields are then processed by REEF2FAST and subsequently used in OpenFAST simulations. To ensure a meaningful comparison, each case is evaluated at two stages along the simulation pipeline: The wavefields generated by NHFLOW are directly extracted using its native output routines. These are compared to the corresponding wavefields obtained at the end of the REEF3D-OpenFAST coupling pipeline. The workflow is visualized in Figure 3.5. The key quantities examined for verification are the velocity components, and the surface elevation. All verification cases are modeled in the two-dimensional  $x$ - $z$  plane, as the wavefields in the verification cases, are invariant in the transverse  $y$ -direction. The wavefield is later expanded along the  $y$ -axis using REEF2FAST's duplication functionality. In addition to verifying the coupling mechanism itself, each case is first checked for physical consistency by comparing the NHFLOW wavefield to theoretical reference solutions. This step serves as an internal validation of the simulation setup and ensures that only physically correct wave data is passed to the OpenFAST coupling stage. This is important for the following validating 6-DoF study (see Section 3.3).

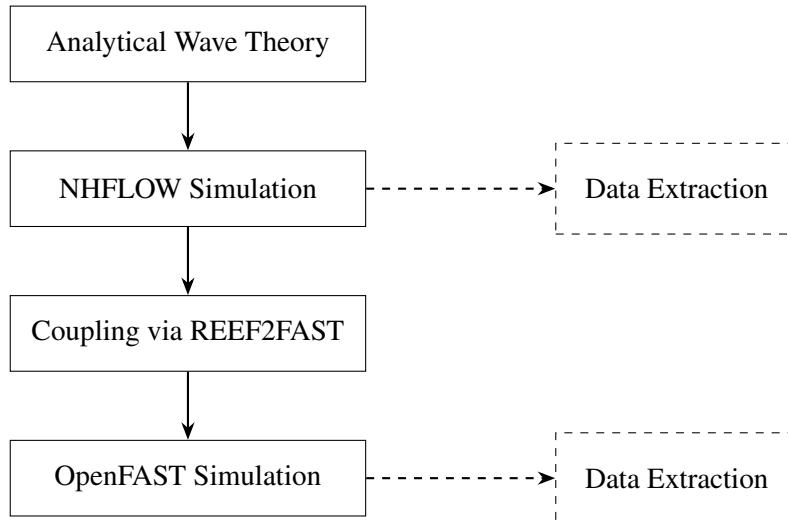


Figure 3.5: Verification workflow for the NHFLOW-OpenFAST coupling. Wavefields are compared both between theory and REEF3D and at the end of the coupling workflow.

### 3.2.1 Regular and Second-Order Stokes Wave Cases

This section presents two verification cases based on linear and weakly nonlinear wave theory: a regular Airy wave and a second-order Stokes wave. Both cases share the same computational domain and mesh setup. The fluid domain spans a total length of  $L = 1400$  m with a constant water depth of  $h = 100$  m. It is discretized uniformly in  $x$ -direction using  $N_x = 560$  cells, resulting in  $\Delta x = 2.5$  m. In  $z$ -direction the resolution is  $N_z = 10$ . A sinusoidal stretching function is applied vertically to increase resolution near the free surface while conserving computational cost at depth. A wave generation zone of 400 m is applied at the inlet boundary ( $x \in [-700, -300]$ ), and a 600 m long numerical wave absorbing beach is implemented at the outlet side ( $x \in [100, 700]$ ). The FOWT is placed, in accordance with the OpenFAST conventions, at the domain center,  $x = 0$  m. The wave kinematic data is extracted at  $x = -150$  m. Following REEF2FAST's one-way coupling approach, in this setup the turbine is not modeled in REEF3D, but later added in OpenFAST. Both cases are run for 20 minutes using dynamic time stepping, with a Courant-Friedrichs-Lowy number constrained to  $CFL < 0.5$ . Figure 3.6 provides a visual overview of the domain configuration.

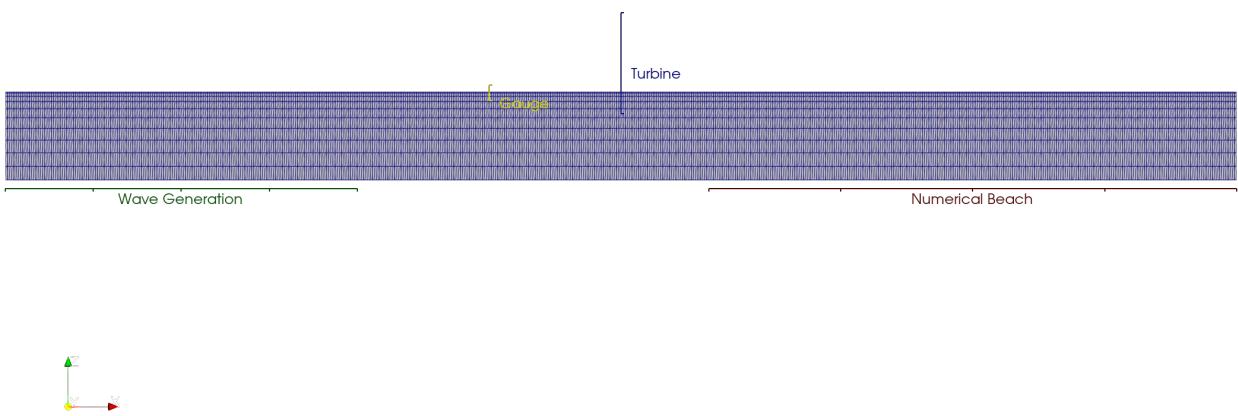


Figure 3.6: Computational domain and mesh layout used for Verification Cases 1 & 2.

#### Case 1: Regular Wave

A monochromatic wave is modeled using linear Airy wave theory with a period of  $T = 8$  s and a wave height of  $H = 1$  m. This case serves as a baseline to verify that the NHFLOW-OpenFAST pipeline preserves wave kinematics under idealized linear conditions. A snippet of the time series of the surface elevation from the REEF3D simulation, together with the corresponding analytical solution, is visualized in Figure 3.7.

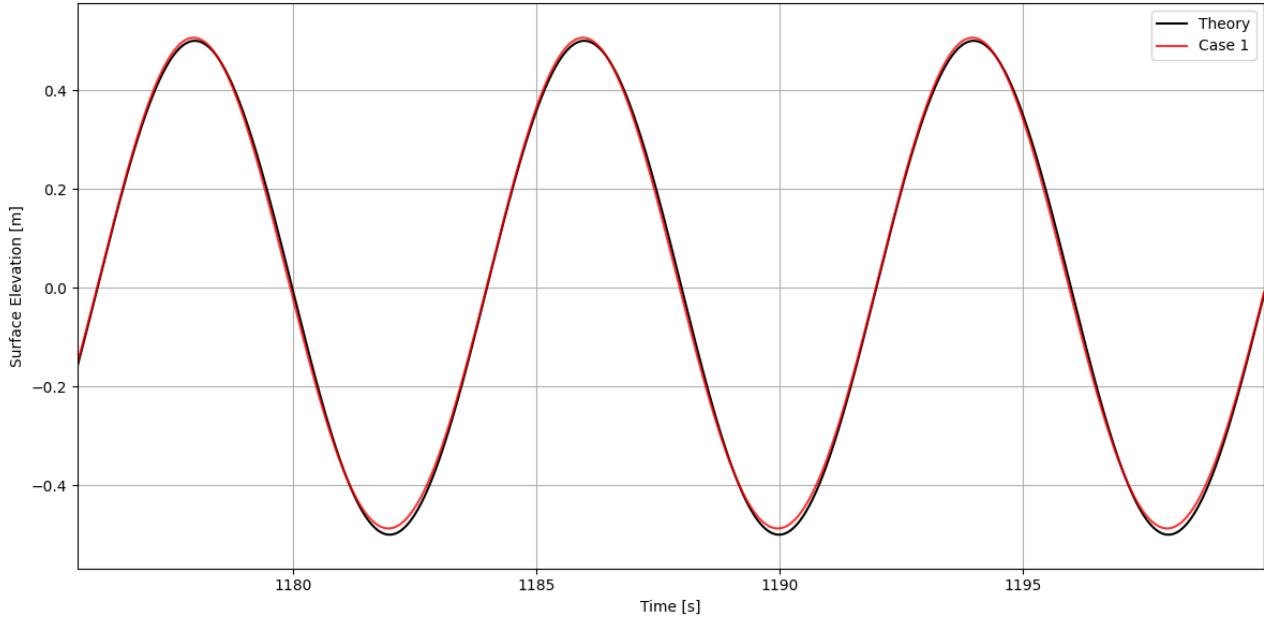


Figure 3.7: Surface elevation at the wave gauge for Verification Case 1: comparison between REEF3D simulation and linear Airy wave theory.

### Case 2: Second-Order Stokes Wave

The second test case adopts the same wave parameters as Load Case 3.1 (LC 3.1) from NREL's OC5 Phase II test campaign involving the DeepCWind semi-submersible platform (see Section 3.3.1) [65]. It features a regular wave train with moderate nonlinearity, characterized by a wave period of  $T = 12.07$  s and a wave height of  $H = 7.37$  m. These parameters induce noticeable asymmetries in the wave profile, such as sharper crests and flatter troughs, which are not captured by linear theory but are well represented by second-order Stokes theory. The aim is to verify whether the nonlinear characteristics of the wavefield can be faithfully preserved through the entire pipeline. A time series of the simulated surface elevation and its second-order Stokes solution is visualized in Figure 3.8.

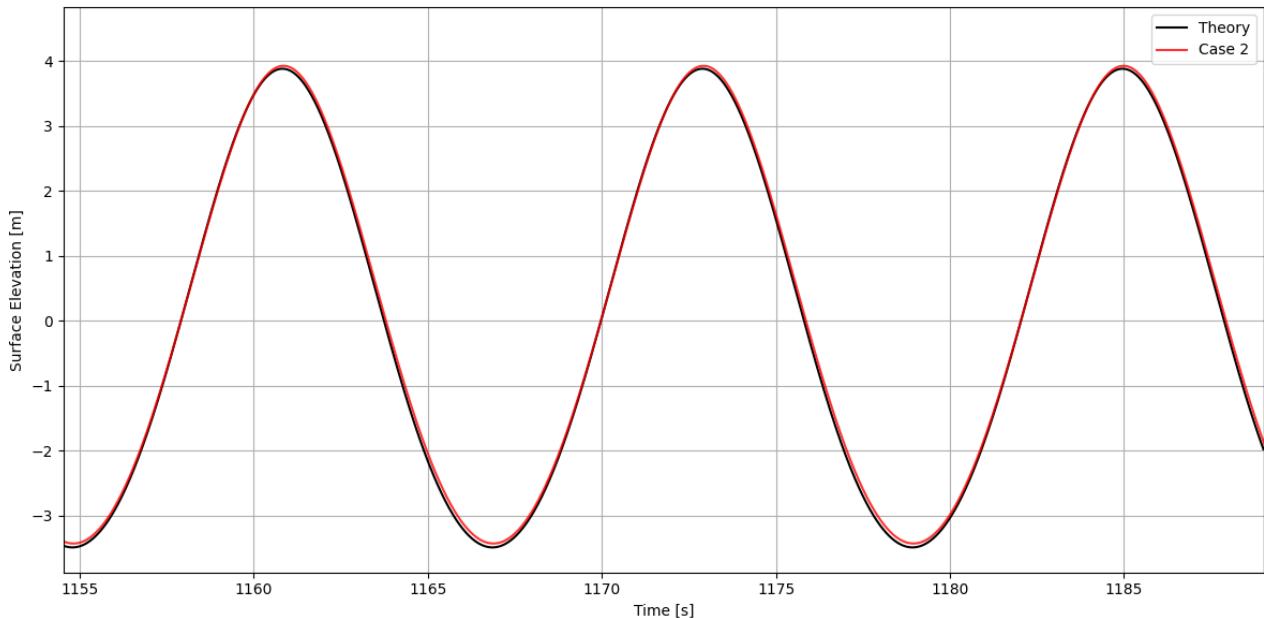


Figure 3.8: Surface elevation at the wave gauge for Verification Case 2: comparison between REEF3D simulation and Second-Order Stokes wave theory.

Figure 3.9 places both test cases within the context of wave theory applicability [66]. The diagram classifies various wave theories based on the non-dimensional wave steepness  $\frac{H}{gT^2}$  and relative depth  $\frac{h}{gT^2}$ . Case 1 (Airy wave with  $H = 1$  m,  $T = 8$  s) lies well within the regime of linear deep water waves. In contrast, Case 2 ( $H = 7.37$  m,  $T = 12.07$  s) exhibits increased nonlinearity and falls into the transition region between linear theory and nonlinear models, such as second-order Stokes waves. Both cases, however, are situated in the deep water regime, where dispersive effects are dominant and the influence of finite depth remains negligible.

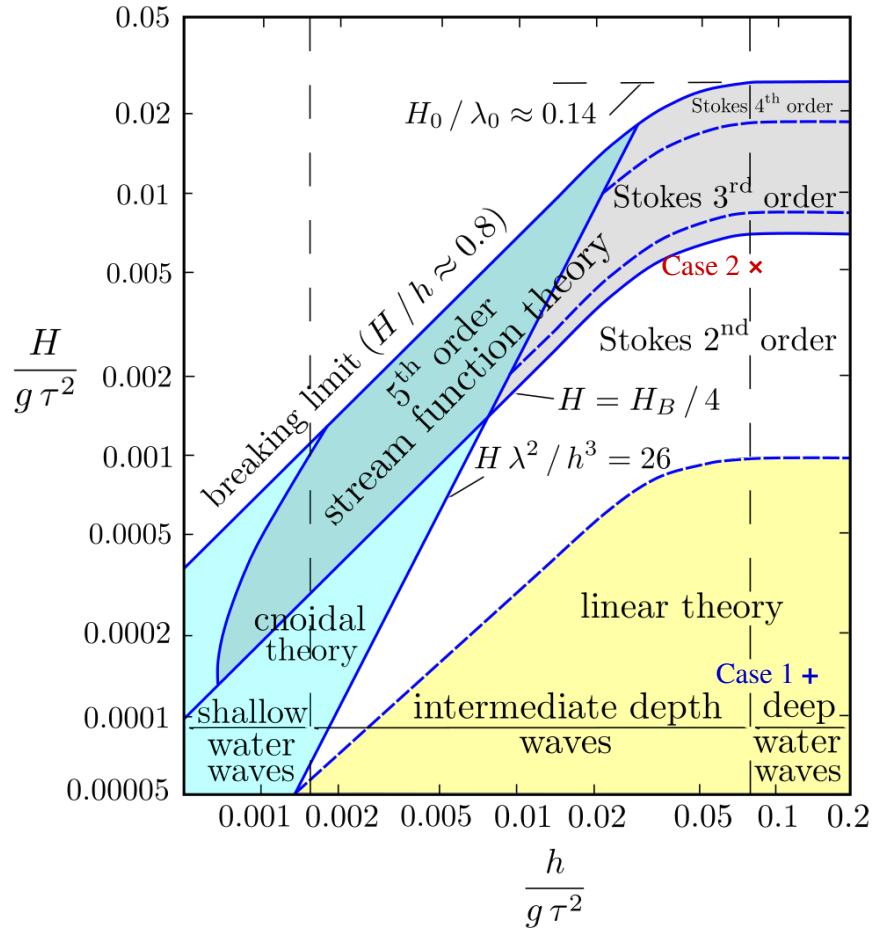


Figure 3.9: Applicability of wave theories depending on non-dimensional wave height and depth. *Blue Plus:* Case 1. *Red Cross:* Case 2. Adapted from Méhauté [66].

### 3.2.2 Case 3: Irregular JONSWAP Spectrum

The third verification case is based on LC 3.4 from the OC5 Phase II test campaign conducted by NREL (see Section 3.3.1). It aims to test the ability of the NHFLOW-OpenFAST pipeline to transfer irregular wave kinematics derived from a realistic sea state without distortion. A JONSWAP spectrum is used to model the irregular sea state, defined by a peak period  $T_p = 14.3$  s, a significant wave height  $H_s = 10.5$  m, and a peak enhancement factor  $\gamma = 3.0$ . This spectrum represents a long-period, moderately nonlinear sea state typical of deep-water conditions. The computational domain spans  $x \in [-860, 860]$  m with a constant water depth of  $h = 200$  m. The mesh is discretized with  $688 \times 20$  cells in the  $x$  and  $z$  directions. The same sinusoidal grid stretching in the vertical direction as in Cases 1 and 2 is applied, so that the overall resolution matches the previous cases. The 600 m wave generation

zone is positioned at the inlet, and the 800 m numerical beach at the outlet. The simulation is run for a total duration of 12,800 s, with dynamic time stepping constrained by a Courant-Friedrichs-Lowy number of  $\text{CFL} < 0.75$ . The last 6,000 s of the REEF3D simulation are extracted and passed to OpenFAST. The wave elevation and velocity are, as in Cases 1 and 2, extracted at a probe point located at  $x = -150$  m, upstream of the turbine location. In order to capture the broad-band characteristics of the irregular wave train, a spectral resolution of 2048 bins is used. The cut-off frequency range is selected based on the energy content of the simulated spectrum ( $\omega_{\min} = 0.3$  rad/s,  $\omega_{\max} = 1.32$  rad/s). These values are derived from a Welch analysis of the experimental data and applied consistently in the REEF3D setup. A visual representation of the computational mesh, including the turbine position, gauge location, and boundary zones, is provided in Figure 3.10.

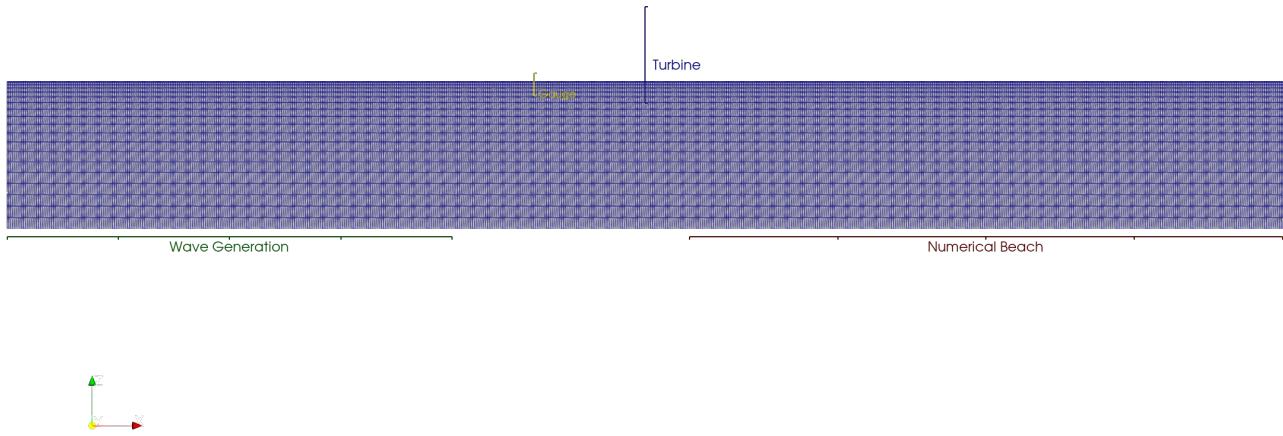


Figure 3.10: Computational domain and mesh layout used for Verification Case 3

In contrast to the previous regular wave cases, a direct comparison in the time domain is not meaningful due to the stochastic nature of irregular waves. Therefore, a frequency-domain analysis is conducted by applying a Fast Fourier Transform (FFT) using `scipy.fftpack` to compute the variance spectral density. The left-hand side of Figure 3.11 shows the resulting energy spectra of the simulated surface elevation compared to the target JONSWAP spectrum. As mentioned, due to computational limitations, only the final 6000 s of the simulation are used for coupling in REEF2FAST. This results in deviations from the full-spectrum response, as illustrated in the right-hand side of Figure 3.11, most notably a reduced spectral peak and a spectral bending around  $f = 0.08$  Hz. The right-hand side of Figure 3.11 also compares the REEF3D simulation sets to the experimental data used in the following validation (see Section 3.3), showing an even greater spectral peak deviation between simulation and experiment or theory. In order to facilitate convergence of the fairlead tensions and ensure a successful initialization of the mooring lines in a catenary configuration, MoorDyn requires calm hydrodynamic conditions at the beginning of the simulation. If the simulation is initialized with fully developed wave conditions, which is the case after 6000 s of the REEF3D run, the resulting abrupt start can lead to convergence issues. To address this, a linear ramp-up function was applied over the first 600 s of the OpenFAST simulation, linearly increasing wave kinematics from zero to their full magnitude.

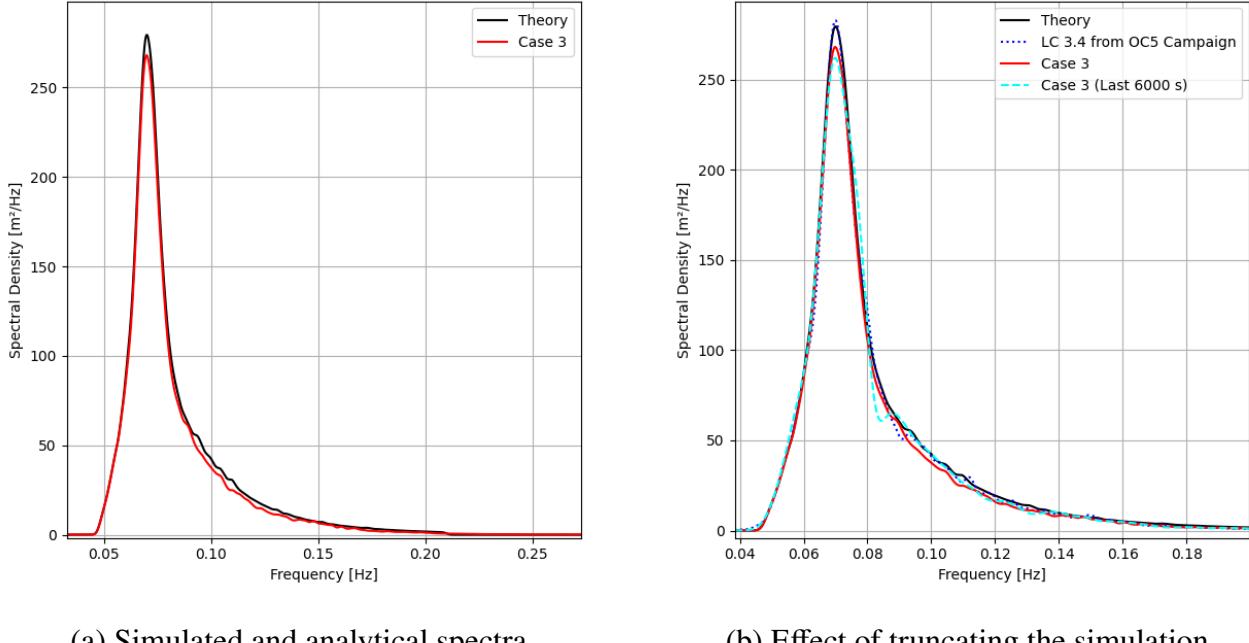


Figure 3.11: Spectral comparison of simulated surface elevation in Verification Case 3 using FFT-based variance spectral density. Data obtained at the wave gauge

A summary of the simulation cases, including wave conditions and simulation settings, is provided in Table 3.1. Input files to run the described cases in REEF3D and more plots of the cases are provided in the REEF2FAST GitHub repository.

Table 3.1: Overview of verification cases including wave characteristics and simulation parameters.

Case	Description	Test Ref. [65]	Wave Type	$H$ or $H_s$ [m]	$T$ or $T_p$ [s]	Duration [min]
1	Regular Wave (Synthetic)	–	Linear	1.0	8.0	20
2	Design Wave (LC 3.1)	902001	2nd-Order Stokes	7.37	12.07	20
3	Irregular Wave (LC 3.4)	907001	JONSWAP ( $\gamma = 3.0$ )	10.5	14.3	180 (100)

## 3.3 Validation

While the verification focuses on assessing the numerical consistency of the REEF2FAST interface, ensuring that wave kinematics are preserved from REEF3D to OpenFAST, the goal of this validation is to evaluate the physical realism of the coupled framework. In particular, the validation examines whether the hydrodynamic input generated by REEF3D::NHFLOW leads to realistic 6-DoF motion responses of FOWTs when used in OpenFAST. For this purpose, the experimental test LC 3.1 and LC 3.4 from NREL’s OC5 Phase II campaign are used as benchmark references.

### 3.3.1 Experimental Test Setup

NREL’s phase II experimental campaign focused on conducting tests of a 1:50-scale model of an FOWT at MARIN’s offshore basin. The model represents the DeepCWind semi-submersible platform supporting a scaled version of the NREL 5 MW reference turbine. All physical parameters were scaled according to Froude scaling. Although the experiment was conducted in model-scale, all measurements were converted to full-scale prior to publication. Thus, the quantities presented in this

thesis correspond all to the full-scale values. MARIN's deep-water offshore basin is equipped with programmable wave generators. The wavemaker can produce monochromatic waves and irregular broadband spectra, like JONSWAP spectrum, with precise control over wave height and period. In the wave-only cases, which have been used for this validation, all wind generation was turned off and the blades were parked ( $\text{pitch} = 90^\circ$ ) to eliminate aerodynamic forces. The platform was held in position by its mooring system, consisting of three catenary mooring lines, spaced  $120^\circ$  apart. The mooring lines were attached to the platform at fairleads 14 m below SWL and anchored to the basin floor. The constant water depth was  $h = 200$  m. The overall mooring setup together with the wave direction is visualized in Figure 3.12 [7].

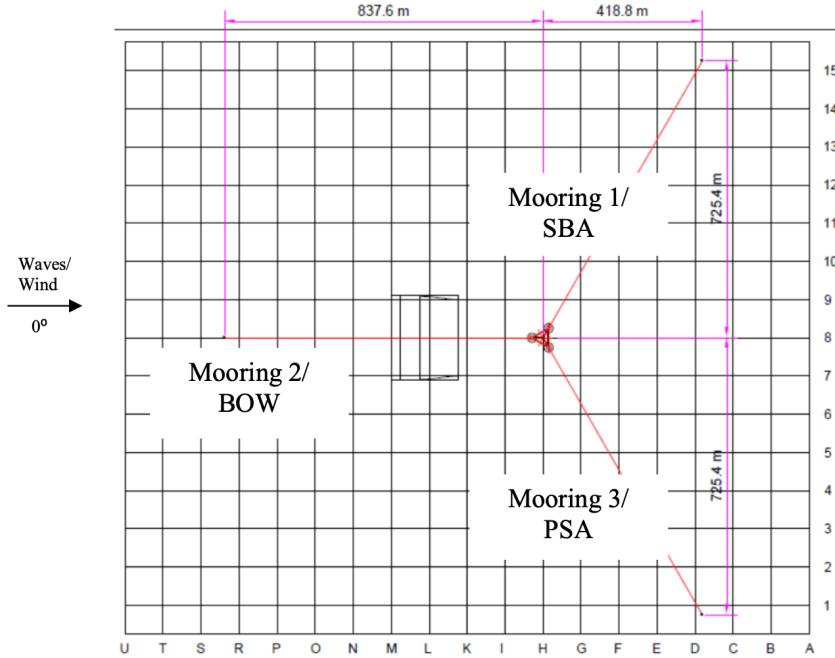
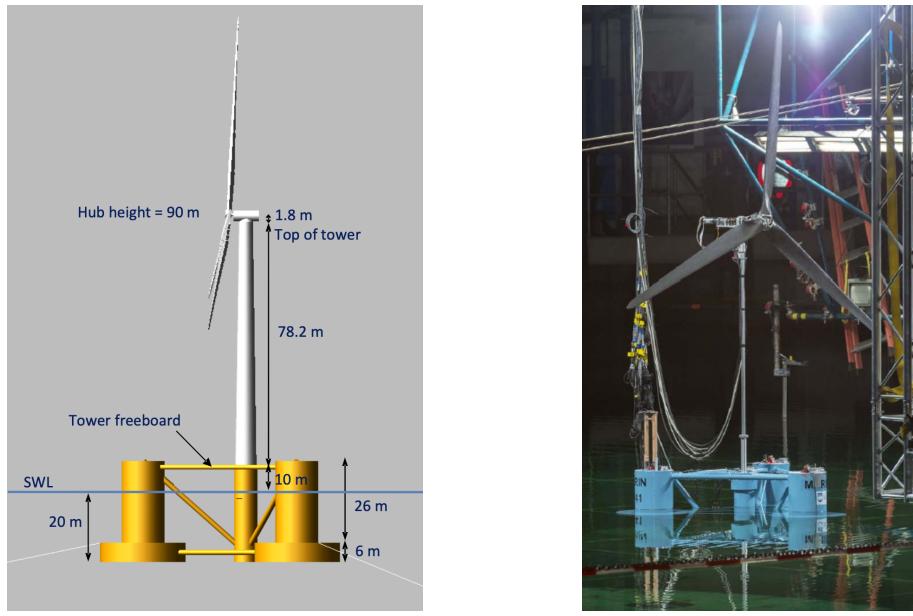


Figure 3.12: Mooring line arrangement and wave direction. Adapted from Robertson et al. [7]

The 1:50-scale model itself reproduced the geometry and mass distribution of the full-scale system. The DeepCwind platform is a three-column semi-submersible structure (one central column supporting the tower and three outer buoyant columns connected by bracing pontoons). The platform has a draft of 20 m and a diameter of roughly 50 m, with the tower bringing the hub height to 90 m above SWL. The NREL 5-MW turbine was reproduced with a performance-scaled rotor (using the MARIN Stock Wind Turbine) and a flexible model-tower. Key dynamic characteristics (e.g. center of mass position, moments of inertia, and hydrostatic restoring in pitch/roll) were verified against the full-scale design. Recorded measurements included the 6-DoF motions of the platform tracked at high frequency, as well as mooring line tensions via submersible load cells. Wave characteristics were recorded with wave probes from the undisturbed wavefield. The full-scale system together with the model-scale test setup is visualized in 3.13 [7].



(a) Full-scale OC5 system [7].

(b) Model-scale setup [67].

Figure 3.13: Representation of full-scale OC5 system (a) and corresponding model-scale setup at MARIN (b). Adapted from Robertson et al. and Helder et al. [7, 67].

A wide matrix of test conditions was executed in the OC5 Phase II campaign, including various wind- or wave-only, and combined wind-wave cases. For the purpose of this project, two wave-only load cases from the campaign (LC 3.1 and LC 3.4) are selected for validation comparisons. For an overview of the wave conditions applied in these two scenarios, see Table 3.1. Both tests were conducted long enough to capture steady-state behavior, after allowing initial transients to decay.

All test data is publicly available at the International Wind Data Hub [65]. A more detailed description of the conduction of the campaign can be found in the report *Definition of the OC5 DeepCwind Semisubmersible Floating System* from Robertson et al. [7].

### 3.3.2 Turbine Modeling

The FOWT modeled in OpenFAST for the validation replicates the OC5-DeepCwind system, based on the OC4-DeepCwind turbine, for which OpenFAST input files are already available [10]. While the overall configuration remains similar between OC4 and OC5, key differences exist, most notably in the tower properties, mooring system, and the platform's mass and inertia characteristics [7]. To recreate the OC5 configuration in OpenFAST, legacy FAST v8 input files provided by Amy Robertson (NREL) were used as a reference to modify the existing OC4 files. The following components were updated:

- **Mooring system:** Line diameters and line mass densities updated to 0.1387 m and 125.6 kg/m. Fairlead and anchor coordinates adapted according to OC5 geometry. Line stiffness adjusted to  $7.49E + 8$  N.
- **Platform mass and inertia:** Floated members (ballast groups) were removed. Platform mass was increased to 13,170,702.55 kg. The roll, pitch, and yaw inertia were updated accordingly to  $7.55E + 9$  kg m<sup>2</sup>,  $8.22E + 9$  kg m<sup>2</sup>, and  $1.3612E + 10$  kg m<sup>2</sup>, respectively.
- **Tower properties:** The tower height was increased to 88.2 m. Mass distribution kept.

- **Overall mass distribution:** Mass of Hub and Generator removed. Nacelle weight increased to  $4.779E + 5$  kg.

These modifications resulted in a slightly lower center of mass and a draught of approximately 21 m. Input files for the OC5-DeepCwind semi-submersible turbine are available in the REEF2FAST GitHub repository. A visualization of the modeled turbine, rendered in ParaView, is shown in Figure 3.14.



Figure 3.14: Representation of NREL’s OC5 semi-submersible FOWT in ParaView via OpenFAST

# Chapter 4

## Results

### 4.1 Verification Test Results

This section presents the wave kinematics computed at the gauge points for verification Cases 1-3. The numerical results obtained from REEF3D are compared against the interpolated values used in OpenFAST via REEF2FAST. The agreement between the fields of the reference simulation  $f_{\text{reef}}$  and the resulting coupled solution  $f_{\text{coupled}}$  is quantified using the normalized mean absolute error (NMAE) over the entire signal duration:

$$\text{NMAE} = \frac{1}{N} \sum_{i=1}^N \frac{|f_{\text{reef},i} - f_{\text{coupled},i}|}{\max|f_{\text{reef}}|}. \quad (4.1)$$

#### 4.1.1 Case 1 - Wave Kinematics

Figures 4.1-4.3 show the visual alignment of all wave kinematic quantities at the evaluation point for Case 1. Table 4.1 lists the NMAE values between REEF3D and OpenFAST, comparing both the Wheeler and No Stretching configurations within the REEF2FAST pipeline.

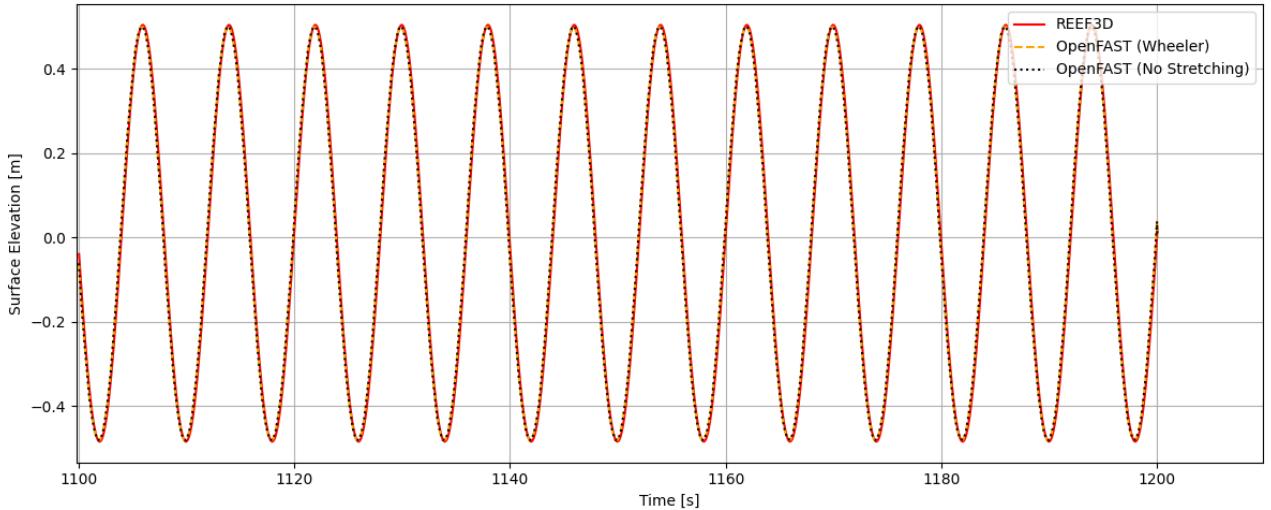


Figure 4.1: Surface elevation  $\eta$  comparison for Case 1 over the final 100 seconds of the simulation.

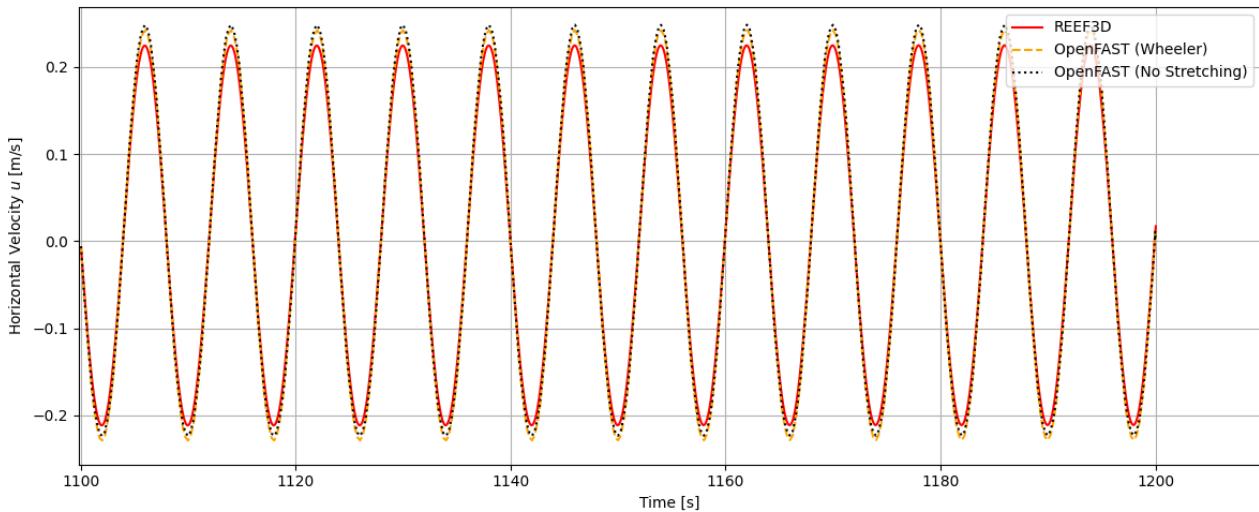


Figure 4.2: Horizontal velocity  $u$  comparison for Case 1 over the final 100 seconds of the simulation.

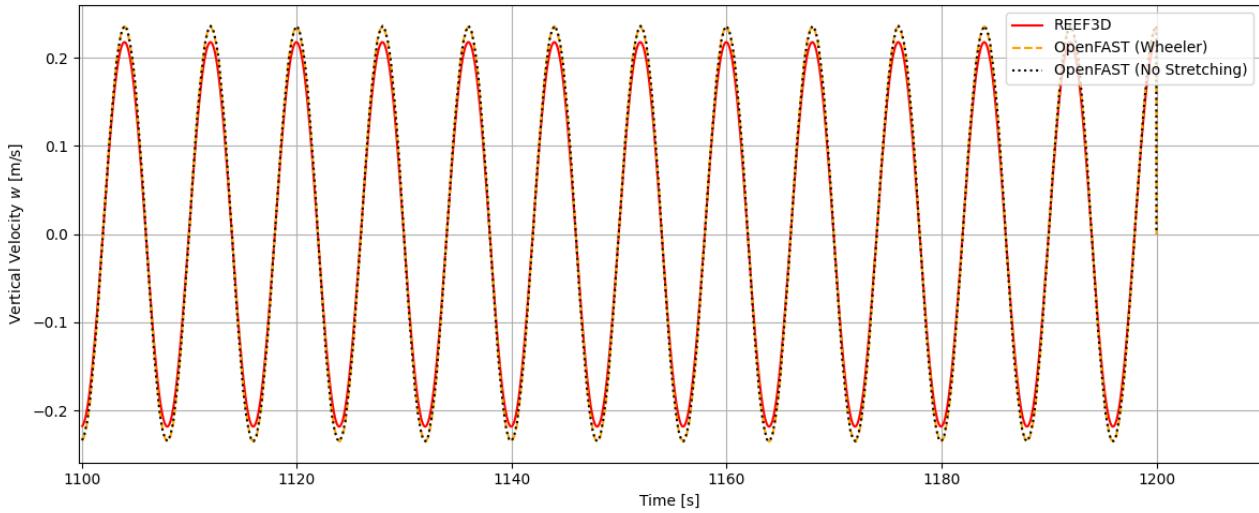


Figure 4.3: Vertical velocity  $w$  comparison for Case 1 over the final 100 seconds of the simulation.

Table 4.1: NMAE for wave kinematics in Case 1.

Quantity	Wheeler Stretching	No Stretching
Elevation $\eta$	3.52 %	3.52 %
Velocity $u$	4.57 %	4.63 %
Velocity $w$	4.63 %	4.63 %

### 4.1.2 Case 2 - Wave Kinematics

Figures 4.4-4.6 show the visual alignment of all wave kinematic quantities at the evaluation point for Case 2. Table 4.2 lists the NMAE values between REEF3D and OpenFAST, comparing both the Wheeler and No Stretching configurations within the REEF2FAST pipeline.

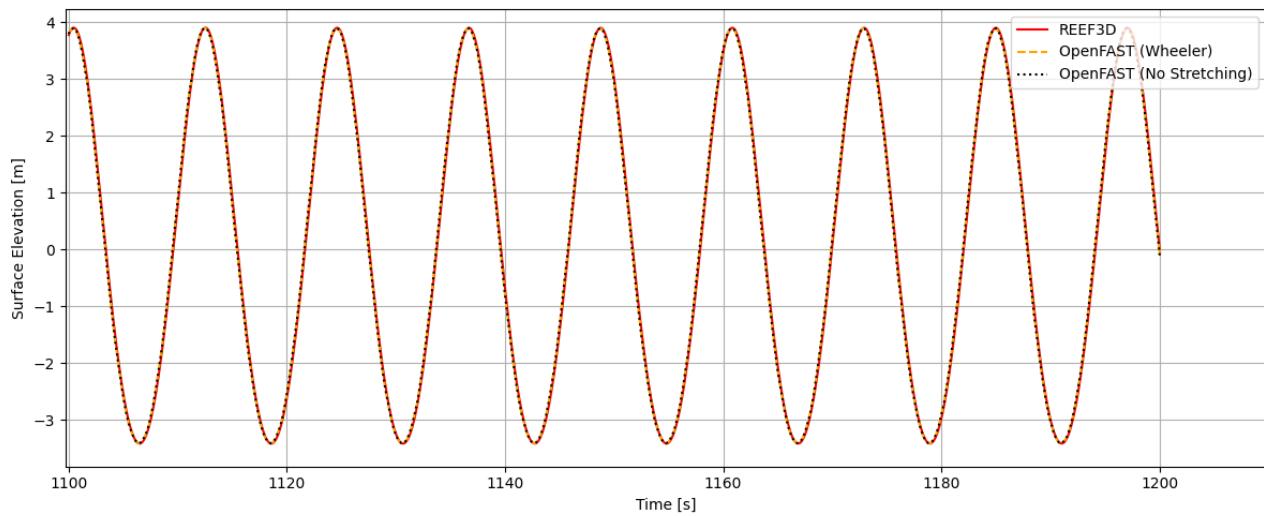


Figure 4.4: Surface elevation  $\eta$  comparison for Case 2 over the final 100 seconds of the simulation.

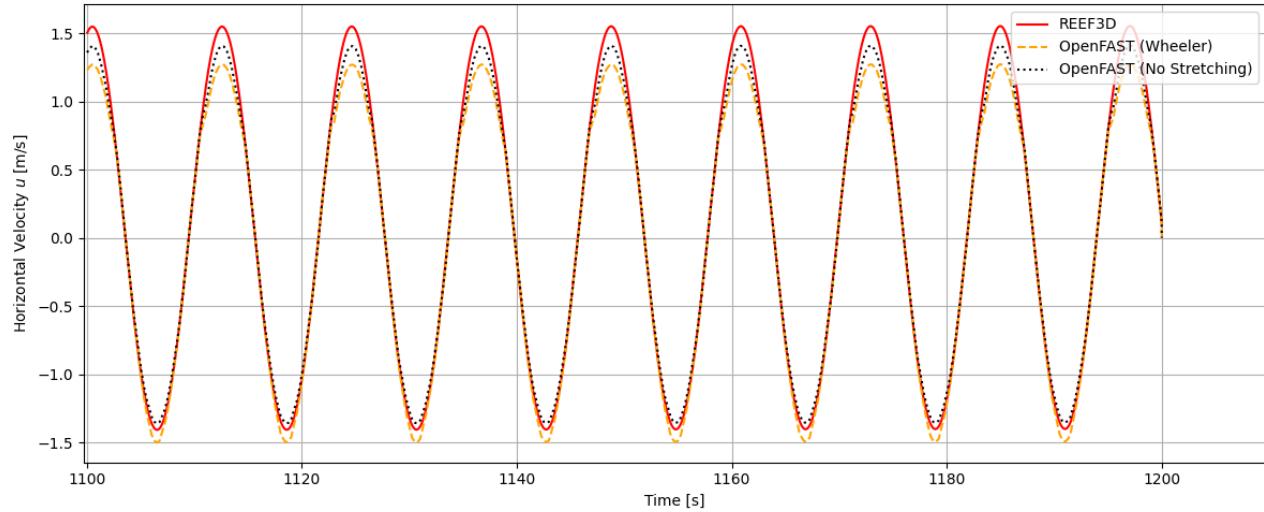


Figure 4.5: Horizontal velocity  $u$  comparison for Case 2 over the final 100 seconds of the simulation.

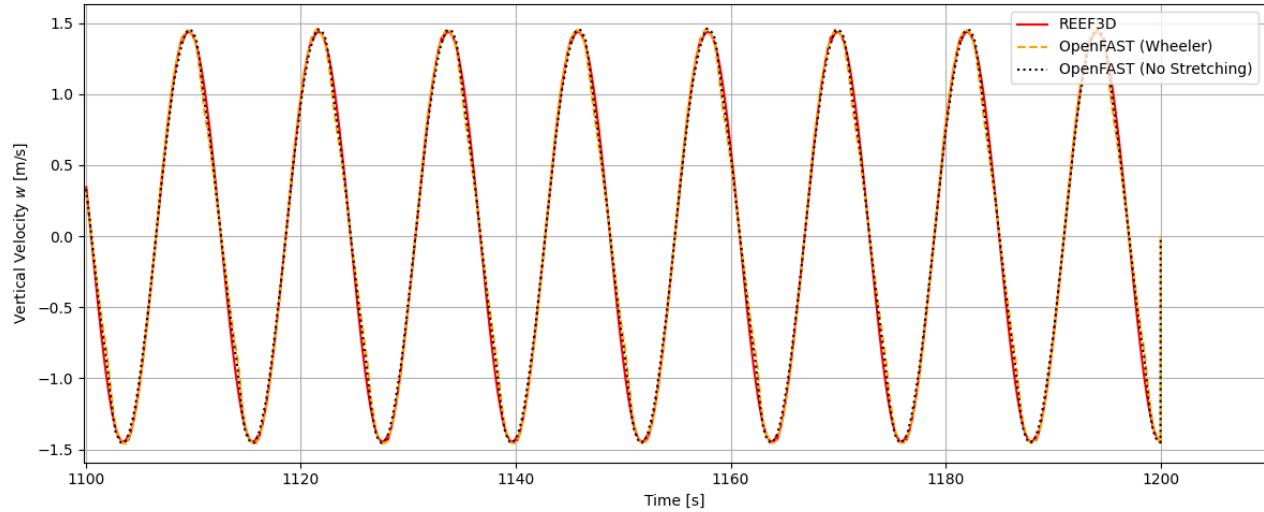


Figure 4.6: Vertical velocity  $w$  comparison for Case 2 over the final 100 seconds of the simulation.

Table 4.2: NMAE for wave kinematics in Case 2.

Quantity	Wheeler Stretching	No Stretching
Elevation $\eta$	1.53 %	1.53 %
Velocity $u$	5.46 %	3.44 %
Velocity $w$	2.90 %	2.20 %

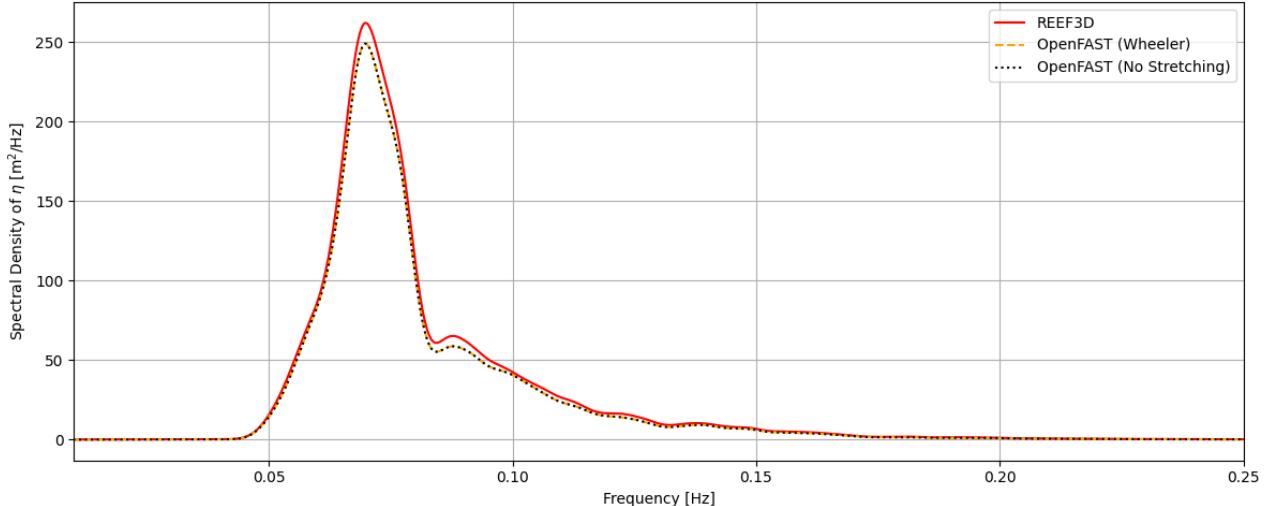
### 4.1.3 Case 3 - Wave Kinematics

In Case 3, the NMAE is no longer suitable due to the irregular nature of the JONSWAP spectrum. To quantify the deviation, the Relative Energy Error (REE) is introduced:

$$\text{REE} = \frac{\int_{\Omega} |S_{\text{coupled}} - S_{\text{reef}}| df}{\int_{\Omega} S_{\text{reef}} df}. \quad (4.2)$$

Here,  $S(f)$  denotes the spectral density in  $\text{m}^2/\text{Hz}$ , and  $\Omega$  is the full frequency domain. The integrals over the spectral densities result in total spectral energies in  $\text{m}^2$ , making the REE a dimensionless quantity that characterizes the relative deviation between both energy contents.

Figures 4.7-4.9 show the spectral comparison of all wave kinematic quantities at the evaluation point for Case 3. Table 4.3 lists the REE values for each quantity, comparing both the Wheeler and No Stretching configurations within the REEF2FAST pipeline.

Figure 4.7: Spectral density of surface elevation  $\eta$  for Case 3.

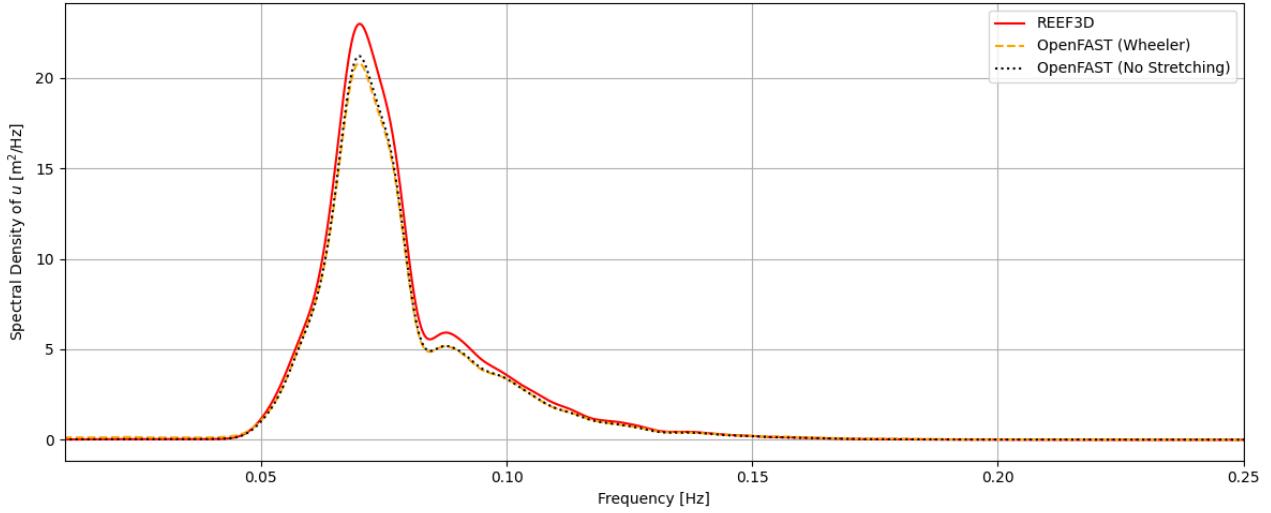


Figure 4.8: Spectral density of horizontal velocity  $u$  for Case 3.

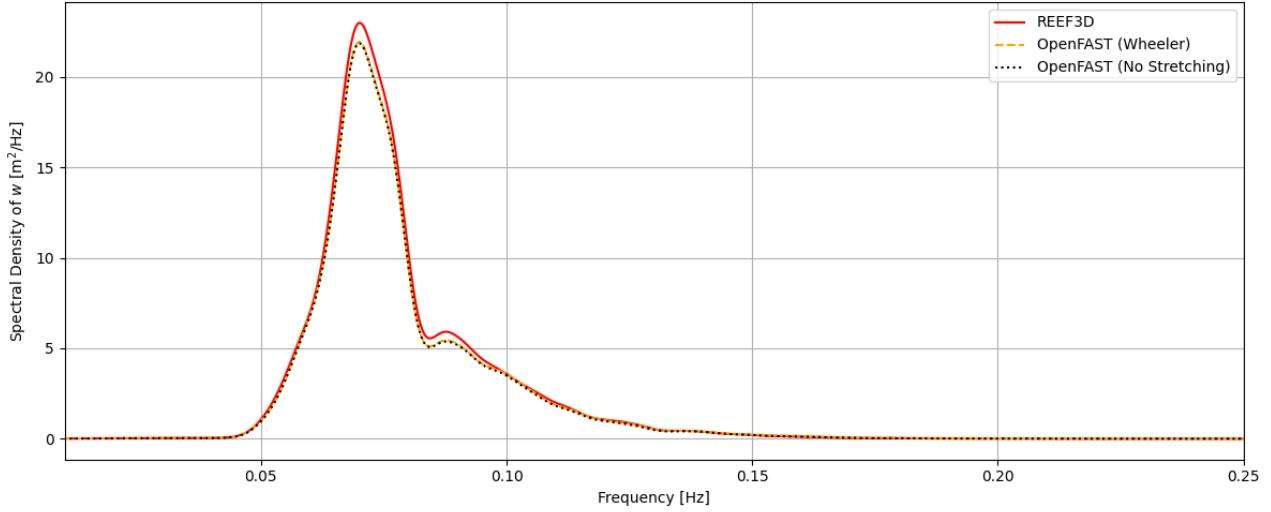


Figure 4.9: Spectral density of vertical velocity  $w$  for Case 3.

Table 4.3: Relative Energy Error (REE) for wave kinematics in Case 3.

Quantity	Wheeler Stretching	No Stretching
Elevation $\eta$	7.11%	7.11%
Velocity $u$	11.87%	9.26%
Velocity $w$	5.77%	6.17%

## 4.2 Validation Test Results

This section presents 6-DoF platform motions surge, heave, and pitch, focusing on two-dimensional motion in the  $x$ - $z$  plane. The motions computed by the coupled REEF3D::NHFLOW-OpenFAST framework are compared against the experimental measurements from NREL's OC5 Phase II test campaign, Load Cases 3.1 and 3.4.

Due to the transient nature of the simulation startup and the absence of defined initial conditions, a steady-state response is only achieved after a short ramp-up period. To enable meaningful comparison, a uniform temporal shift was applied to the simulation data to align with the experimental phase.

### 4.2.1 Case 2 - Platform Response

Figures 4.10-4.12 show the comparison of the simulated motions with the experimental results for Case 2. A phase shift of the OpenFAST results of  $t = +4$  s is applied. The experimental surge signal was initially offset positively ( $\approx 2\text{m}$ ) and is centered around 0. Figure 4.13 shows the transient phase of the simulation, the phase shift, and the original data of the surge motion.

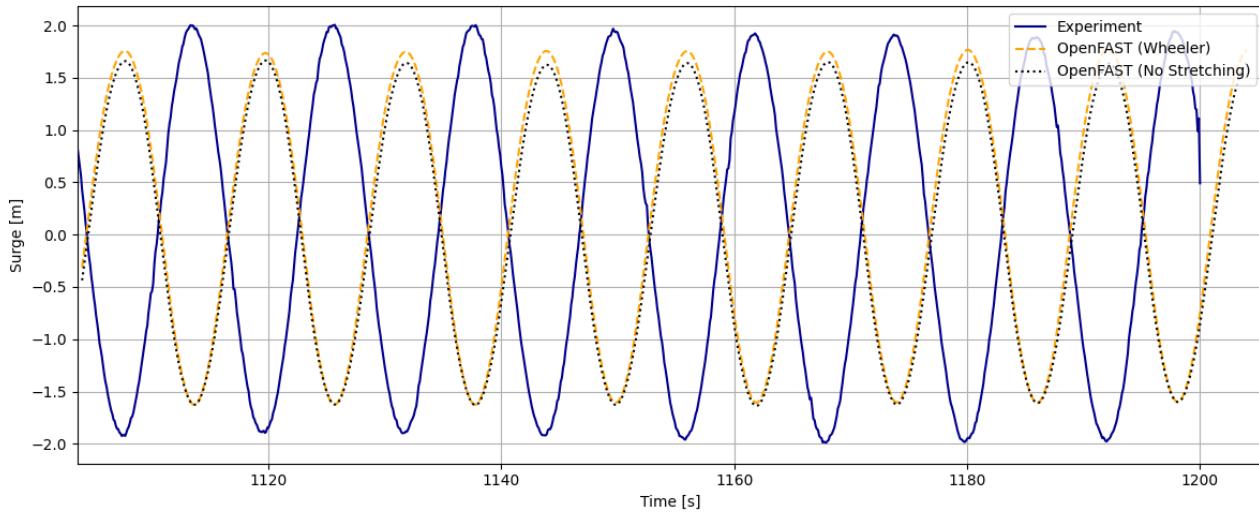


Figure 4.10: Surge motion comparison for Case 2 over the final 100 seconds of the simulation.

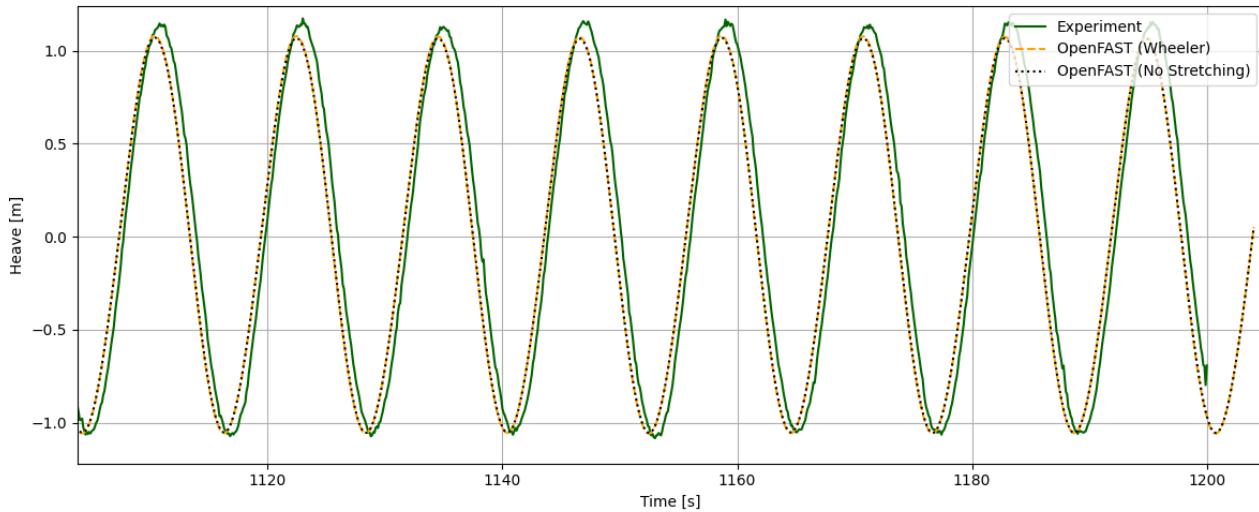


Figure 4.11: Heave motion comparison for Case 2 over the final 100 seconds of the simulation.

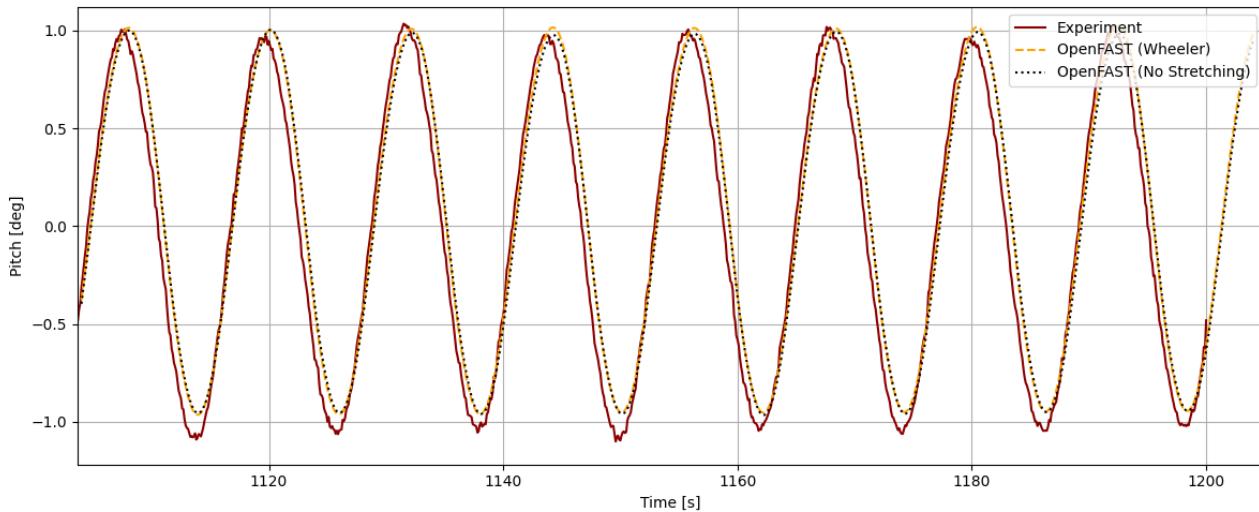


Figure 4.12: Pitch motion comparison for Case 2 over the final 100 seconds of the simulation.

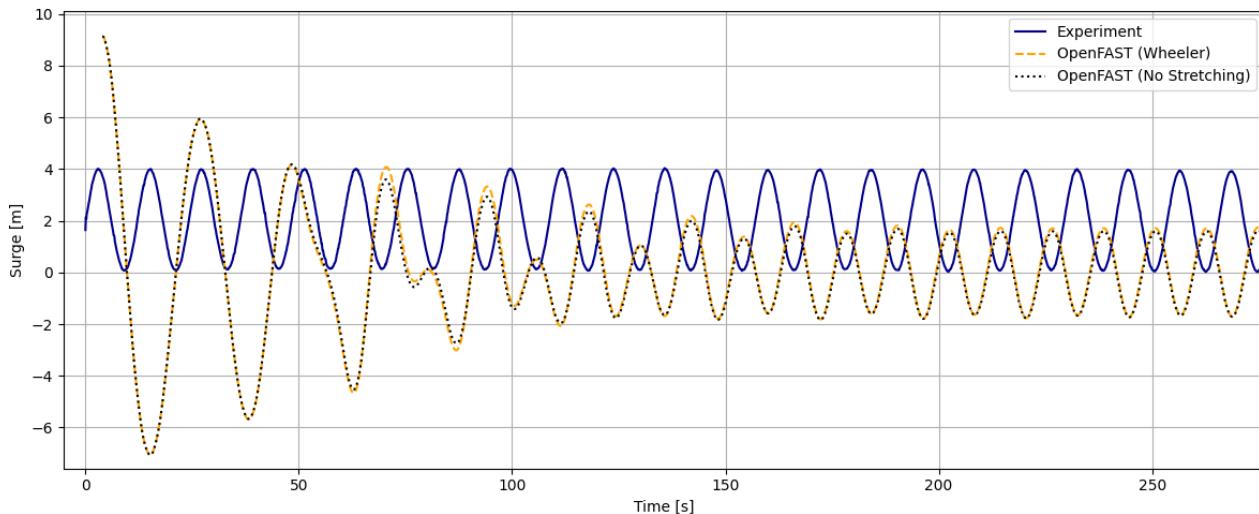


Figure 4.13: Phase shift, transient phase, and initial experimental offset of surge motion in Case 2.

#### 4.2.2 Case 3 - Platform Response

Figures 4.14-4.16 show the comparison between simulated and experimental platform motions for Case 3 in the frequency domain. The experimental time series was filtered to exclude the initial 5% (9 min) to remove transient effects. As outlined in Section 3.2.2, the simulation data includes a ramp-up phase that was necessary due to computational limitations (only the final 6000 s of the simulation could be utilized) and stability considerations. This ramp-up period is included in the plotted results.

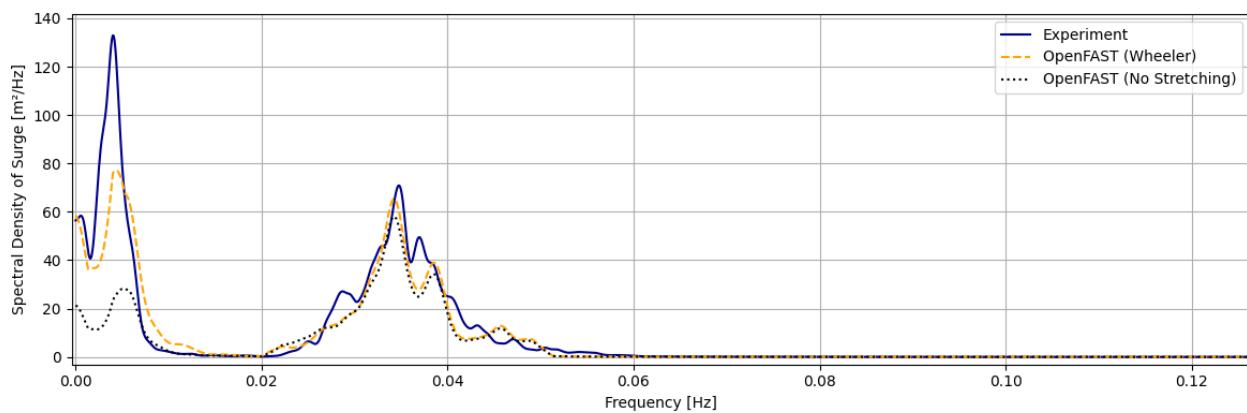


Figure 4.14: Surge motion comparison for Case 3 in the frequency domain.

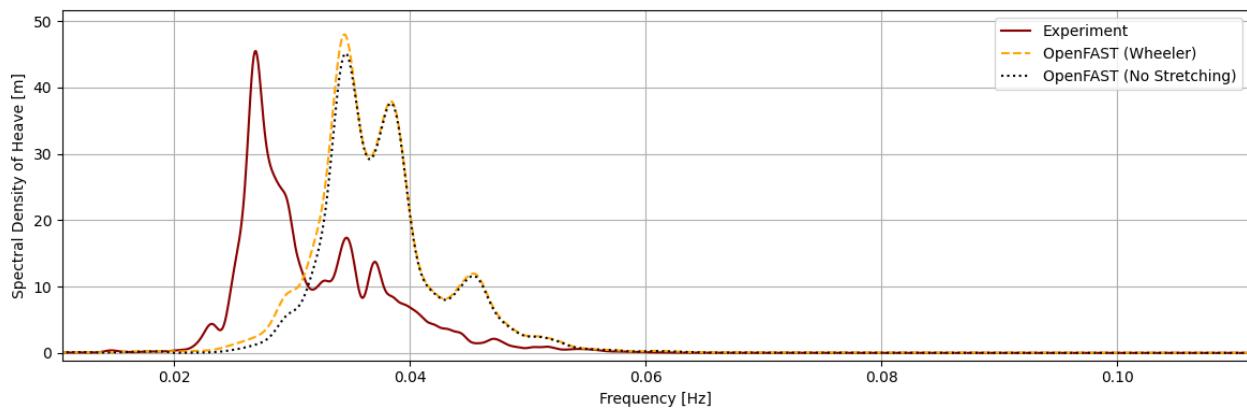


Figure 4.15: Heave motion comparison for Case 3 in the frequency domain.

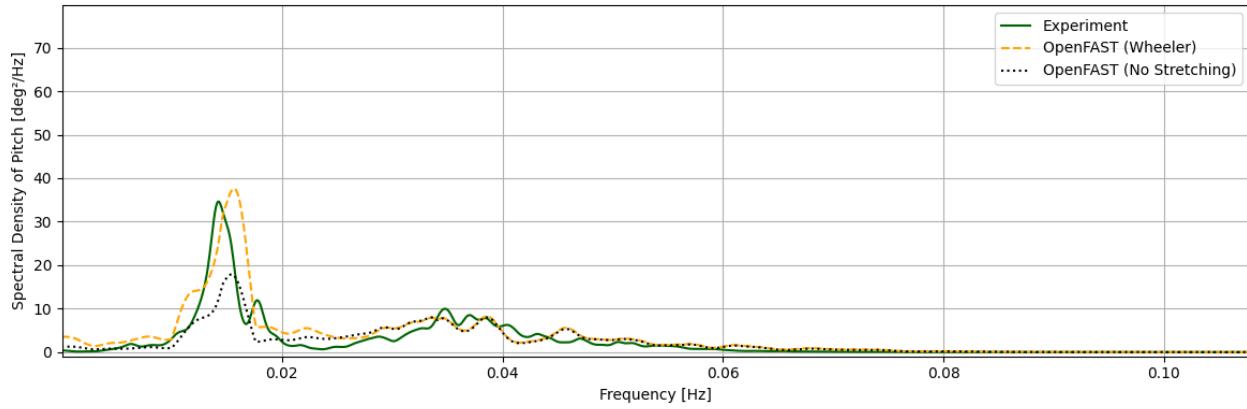


Figure 4.16: Pitch motion comparison for Case 3 in the frequency domain.

# Chapter 5

## Analysis

### 5.1 Verification Analysis

#### 5.1.1 Correctness and Deviations

An overall agreement between the original REEF3D data and the REEF2FAST-converted OpenFAST wave kinematics is visible visually and quantitatively, as confirmed by the corresponding figures and error metrics. Notably, the deviations in surface elevation are the smallest across all three simulation scenarios. In Figures 4.1 and 4.4, aside from a minor phase shift observed in Case 1, no significant differences between OpenFAST and REEF3D can be detected, resulting in NMAE's of 3.5% and 1.5%, respectively. It should be noted that the NMAE metric is more sensitive to phase shifts than the human eye, as it accumulates even small pointwise deviations across the entire signal. In contrast, visual inspection is more responsive to amplitude differences such as under- or overshooting. This distinction explains why an apparently well-aligned signal may still yield a relatively high error, like the surface elevation in Case 1.

A more pronounced deviation between the initial and the interpolated data is observed in the velocity components. REEF2FAST interpolates over a two-dimensional plane, rather than a one-dimensional line as in the surface elevation. The interpolation acts directly on the raw volumetric values of the velocity components, whereas the surface elevation is first aggregated as the column-wise maximum before being interpolated in a single horizontal direction. This inherently stabilizes the elevation data and reduces susceptibility to overshooting artifacts that can occur in two-dimensional IDW schemes. Moreover, As shown in Section 3.1.2, IDW guarantees that interpolated values lie within the range of the neighboring data. It does not guarantee that the interpolation point lies within the convex hull of the contributing neighbors. If a point on the target grid lies outside the convex hull of its neighboring source points, which typically occurs in sparsely populated regions of the domain, it may lead to extrapolation behavior of the interpolation algorithm and coherent errors, as observed in the time series comparisons for Cases 1 and 2 (see Figures 4.2, 4.3, and 4.5). So, the higher deviation comes perhaps from the increased complexity and steep local gradients in the velocity fields, which are more sensitive to spatial irregularities.

All three wave kinematic components show larger visual and quantitative deviations in Case 3 compared to Cases 1 and 2. Several factors contribute to this behavior. The JONSWAP spectrum represents a broadband irregular sea states characterized by multiple local maxima and steeper gradients that can occur simultaneously. Due to the smoothing effects of spatial interpolation, this spectral complexity leads to an overall reduction in energy content across the frequency range, as observed in Figures 4.7-4.9. Also, the superposition of numerous wave components in an irregular spectrum causes constructive and destructive interference at each time step. As a result, even minor interpolation-induced distortions are amplified, leading to larger deviations in spectral energy compared to regular wave cases.

Nevertheless, the interpolation-induced errors discussed above can be mitigated by increasing the spatial and temporal resolution of the source data. However, a perfect match with the original REEF3D data is not to be expected, due to the inherent limitations and smoothing of interpolation algorithms.

### 5.1.2 Influence of the Wheeler Stretching

The Wheeler stretching method projects wave kinematic quantities located above the SWL downward into the fluid domain. This projection alters the vertical distribution of the wave signal and, in verification cases, can degrade the agreement between the interpolated and source wavefields even further. As seen in Figure 4.5 and the corresponding error metrics in Table 4.2, the application of Wheeler stretching results in larger deviations in the velocity components. This behavior can be attributed to the fact that the evaluation point lies 10 m below the SWL. Due to this projection, velocity information originating near the wave crest is artificially mapped to this depth. Since these contributions are not physically present in the original data, the resulting signal is less accurate. This leads to a reduced amplitude and a poorer representation of the original wave kinematics in the interpolated field. In Verification Case 2, the influence of the projection is more pronounced compared to Case 1, because of the larger wave height. Higher crests and deeper troughs result in more extreme vertical excitement of the wave surface, causing wave kinematic points to be projected further into the domain below SWL. As a consequence, the artificially introduced velocities at  $h = -10$  m exhibit greater deviations from the original REEF3D::NHFLOW data.

## 5.2 Validation Analysis

### 5.2.1 Correctness and Deviations

In the validation, the 6-DoF platform motion of the OpenFAST model of the OC5 semi-submersible turbine, excited by wave kinematics generated via the REEF2FAST coupling framework, is validated against the experimental motion of the model-scale FOWT tested in the OC5 campaign.

In Case 2, the most notable deviation is observed in the surge motion (see Figure 4.10). While the period of all examined motion components is consistent, the simulated surge response shows a phase shift of approximately half a wave period relative to the other simulated motions. Moreover, both surge and heave motions show reduced amplitudes compared to the experiment. The application of the Wheeler projection method leads to an amplified surge response, improving the alignment in amplitude magnitude. Heave and pitch motions demonstrate good visual agreement with the experimental results, mostly independent of Wheeler stretching (see Figures 4.11 and 4.12). Overall, Case 2 showcases higher deviations to the experiment in the translational motions, while the rotational pitch is more accurately reproduced.

Case 3 shows overall visually higher deviations compared to Case 2. Most notably, the low-frequency components of surge and pitch motions are not captured accurately by the simulation setup, as shown in Figures 4.14 and 4.16. However, the application of Wheeler stretching helps to mitigate the discrepancy, improving the representation of low-frequency motions. Higher frequency components around  $f \approx 0.03$  Hz in both surge and pitch were adequately reproduced. Additionally, in the heave motion the platform responds in a higher frequency, peaking at  $f \approx 0.035$  Hz, whereas the experimental data shows a peak at  $f \approx 0.03$  Hz, as visible in 4.15. Nevertheless, the overall energy content of the heave response remains comparable.

### 5.2.2 Modeling Uncertainties

In addition to the interpolation errors discussed in Section 5.1.1, the discrepancies of the motions observed in Cases 2 and 3 are most likely mainly amplified by uncertainties in the structural and

hydrodynamic modeling of the OC5 DeepCWind system in OpenFAST. Due to the absence of publicly available OpenFAST input files for OC5, the model had to be approximated based on OC4 data, which required several assumptions regarding the mass distribution, mooring system, and inertia parameters. For example, although the overall mass was updated to match the OC5 specifications, the tower's mass distribution per unit length remained unchanged, potentially affecting the center of mass and inertia matrix of the system. Moreover, the tower modeled exhibits lower flexibility than the original OC5 tower, which may also result in different motion patterns. These uncertainties can have a pronounced impact on the coupling between surge, heave, and pitch motions.

The observed half period phase shift of the surge in Figure 4.10 (Case 2) and the underrepresentation of low-frequency response in the surge motion of Case 3, as indicated in Figure 4.14, may be attributed to overly stiff or insufficiently damped mooring lines or inaccurate representation of hydrostatic restoring forces. Consequently, without access to detailed turbine specifications, a perfect agreement with the experimental motion results cannot be expected. Additionally, part of the observed deviations of the platform motions may be rooted in the inherent limitations of the hydrodynamic model used in OpenFAST itself. The 6-DoF motion computation rely on strip theory, as presented in Section 2.3.3. Especially for large-volume structures such as semi-submersible platforms, this approach introduces systematic errors in the prediction of added mass and wave excitation forces. This is particularly critical at low frequencies, where long wavelengths induce diffraction and radiation effects that are not captured by the local, cross-sectional assumptions of strip theory, potentially resulting in systematic underprediction of hydrodynamic forces and platform response.

Despite these modeling challenges, the accurate reproduction of the wave kinematics across all verification cases and the qualitative alignment of the platform motions in the validation cases demonstrate that the REEF2FAST coupling itself performs robustly and reliably under the tested conditions. Consequently, REEF2FAST builds a promising foundation for further refinements and extensions of REEF3D-OpenFAST coupling tool-chains.

# Chapter 6

## Outlook

This chapter outlines potential avenues for strengthening and expanding the REEF2FAST coupling framework. Key aspects include the extension of the application scope, improvements in modeling accuracy, and enhancements in computational performance.

### 6.1 Complex Bathymetries

The next step in extending the use-cases of the REEF2FAST coupling framework is to involve scenarios with complex bathymetries. Given that REEF3D::NHFLOW is inherently capable of resolving arbitrary seabed geometries, the coupling pipeline, in principle, can accommodate these features as well. The key requirement for compatibility with OpenFAST's SeaState module is, that the input water domain remains rectangular, defined on a Cartesian grid. Complex bottom topographies, such as slopes, islands, or banks, can be embedded into this grid by assigning nodes, that lay below the local seabed elevation, wave kinematic field values with a non-integer variable like NaN. REEF2FAST would then need to recognize these placeholders and exclude them from the interpolation. So, these variables effectively mask out solid regions during interpolation and signal to REEF2FAST that the data at those nodes is undefined or physically irrelevant. A conceptual overview of this approach, sketched in the  $x$ - $z$  plane, is visualized in Figure 6.1. In OpenFAST, every non-integer variable in the wave kinematic field arrays is treated as 0. Hence, no further transformation of the field values contained in the masked-out nodes is needed.

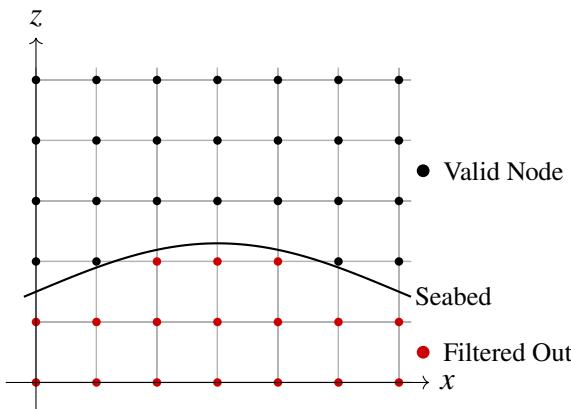


Figure 6.1: Concept of bathymetry masking in REEF2FAST. Points below the seabed are excluded from interpolation by assigning non-integer values. This enables a rectangular mesh structure compatible with OpenFAST.

If this approach is not realizable, for instance, due to grid restrictions from REEF3D's  $\sigma$ -grid, another option would be to signal REEF2FAST, that a case with a non-rectangular bathymetry is to be

processed by user-flagging via command-line options. REEF2FAST can then analyze the wavefield data and determine the lowest vertical coordinate  $z_{\min}$  in each  $x$ - $y$  column, following an approach similar to the method used in the surface elevation algorithm (see Section 3.1.4). Based on this information, REEF2FAST can autonomously expand the domain downwards to form a rectangular bounding box and assign non-integer placeholders to all nodes lying below the inferred seabed. This transformative approach is visualized in Figure 6.2. Both concepts would lead to the same outcome in OpenFAST and enable REEF2FAST to handle complex bathymetries in a robust manner.

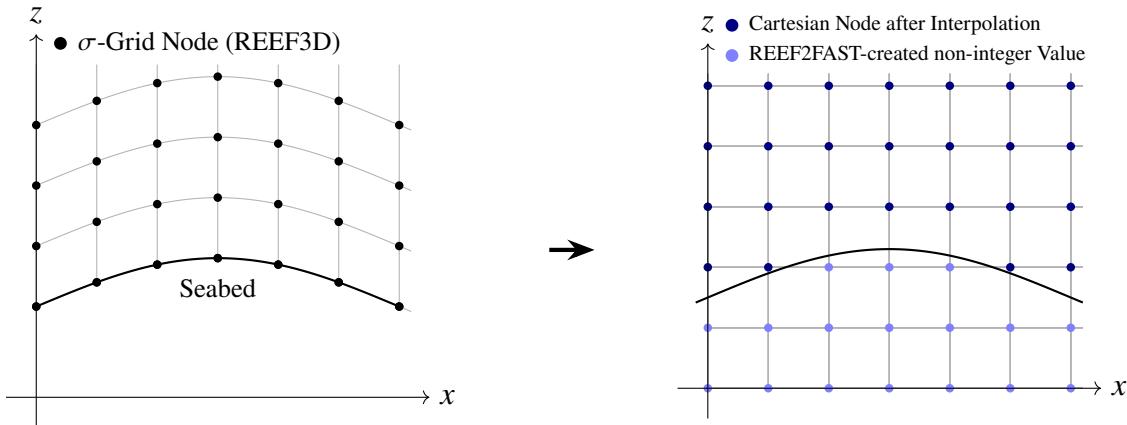


Figure 6.2: Transformation of REEF3D’s  $\sigma$ -grid to an OpenFAST-compatible Cartesian grid with integrated complex bathymetry concept. REEF2FAST identifies the lowest valid depth in each  $x$ - $y$  column and assigns non-integer values below the seabed to exclude them from interpolation.

## 6.2 Two-Way Coupling

While two-way coupling offers the potential to resolve feedback mechanisms between the structure and the surrounding wavefield, its implementation must be considered in terms of the OpenFAST’s underlying internal coupling strategy. The current one-way, partitioned approach realized by REEF2FAST aligns with OpenFAST’s internal coupling, which emphasizes modular subsystems with interface-driven data exchange and sub-iterations (see Section 2.3.4). Consequently, REEF2FAST maintains architectural consistency while keeping computational complexity at a minimum. Nevertheless, a two-way coupling strategy would enhance the existing framework by enabling dynamic interaction between the floating structure and the wavefield, allowing for a more accurate representation of phenomena such as wave deformation or breaking induced by the presence of the structure. In the current REEF2FAST framework, conventional two-way coupling as outlined in Section 2.1 is not directly realizable due to the modular, partitioned, and file-based architecture of the coupling process. Achieving two-way interaction would require modifications to the codes of either REEF3D or OpenFAST. However, an alternative and promising approach is to implement the structure directly within the REEF3D simulation. By doing so, structural effects on the wavefield, such as diffraction, reflection, or induced wave breaking, can be inherently resolved inside the REEF3D simulation. The concept of complex bathymetry handling introduced in Section 6.1 could then be extended to include floating bodies. These geometries could be passed through REEF2FAST and be integrated into the OpenFAST input as part of the hydrodynamic environment. This strategy would preserve the advantages of OpenFAST’s aero-servo-elastic modeling while simultaneously enabling REEF3D to resolve complex fluid-structure interactions with full fidelity. However, such an approach must be validated by comparing 6-DoF platform motions with conventional OpenFAST simulations. Additionally, the increased computational cost must be taken into account, as simulating floating structures in REEF3D is considerably more expensive than in OpenFAST.

## 6.3 Performance Enhancements

In its current implementation, REEF2FAST utilizes VTU files exported from REEF3D as the source of wave kinematics. VTU files store data at the cell edges. In REEF3D, due to the staggered grid, the non-hydrostatic pressure quantities are defined at the cell centers, while velocity fluxes are located at the cell faces they pass through. Consequently, the VTU export already involves an internal interpolation of the original data, introducing a first source of numerical approximation. To improve accuracy, a more direct data transfer is recommended by using REEF3D state files that preserve the original variable locations (cell centers and faces) without prior interpolation. This would allow REEF2FAST to perform its own interpolation, using the raw numerical data. A further step towards optimization would be to integrate REEF2FAST directly into the REEF3D source code. This would enable REEF2FAST to carry out the interpolation within the REEF3D time loop, eliminating the need to write and read external input files entirely. In addition to skipping the VTU-based interpolation step, this integration would significantly reduce computational overhead by avoiding file I/O operations, thereby accelerating the overall coupling workflow.

# Chapter 7

## Conclusion

As floating wind energy matures, the ability to simulate realistic sea states becomes increasingly essential for a robust and efficient design of FOWT. Coupling tools that combine high-fidelity hydrodynamics with established wind turbine multiphysics solvers are able to close the gap between detailed fluid dynamics and engineering-scale system simulations. Bringing these two domains together requires a toolchain that can efficiently handle data transmission, spatial mapping, and temporal synchronization between hydrodynamic fluid solvers and system-level simulation tools.

One of the most widely used system-level simulation tools in the field of FOWT analysis is OpenFAST. However, its native hydrodynamic modeling capabilities are limited, which creates the need for extension through external wave-resolving solvers. Hence, a coupling strategy between the open-source hydrodynamics framework REEF3D and OpenFAST is introduced. REEF3D offers with its shock-capturing non-hydrostatic Navier-Stokes solver REEF3D::NHFLOW a tool that can extend OpenFAST hydrodynamic capabilities with various nonlinear wave models, while still maintaining computational inexpensiveness. NHFLOW provides fully three-dimensional, vertically resolved velocity and non-hydrostatic pressure fields, suitable for straightforward integration into OpenFAST, at significantly lower cost than full turbulence-resolving CFD models.

Based on this approach, the REEF2FAST coupling pre-processor is developed. It transfers wave kinematics generated by NHFLOW into OpenFAST, thereby enabling high-fidelity wavefields to be utilized in nonlinear aero-hydro-servo-elastic simulations. Through its time-step-streaming architecture, REEF2FAST offers a lightweight and memory-efficient solution to the coupling task, making it suitable for long-duration simulations and large-scale CFD input. Its parallelized interpolation pipeline combines a k-d tree-based nearest neighbor search with IDW, enabling the flexible projection of REEF3D wavefields, without explicit reliance on the underlying  $\sigma$ -grid logic, onto OpenFAST's Cartesian target grid. The integrated Wheeler stretching method ensures that wave crests can be incorporated into the OpenFAST target grid, as the vertical domain in OpenFAST is only defined up to the SWL. This approach improves the accuracy of crest-resolving simulations without requiring grid extensions beyond the instantaneous free surface. REEF2FAST is written in modern C++ and released as an open-source command-line application. Its modular architecture and simple interface make it well-suited for integration into automated pipelines or coupled simulation frameworks.

Verification results across three representative wave cases confirmed that REEF2FAST accurately transmits wave kinematics from REEF3D to OpenFAST. Surface elevation signals matched the original REEF3D data with NMAEs of  $\leq 3.5\%$ . Minor over- and undershoots of the recreated velocity fields were observed, primarily due to interpolation errors, for instance, convex hull limitations of IDW. Nevertheless, the overall alignment remained within acceptable margins.

Validation against load cases from the NREL OC5 DeepCWind Campaign Phase II further demonstrated that the coupling tool is capable of qualitatively reproducing the experimental 6-DoF platform motions. In the second-order Stokes wave test case, heave and pitch responses showed strong alignment with the measured data, while surge motion exhibited slightly larger deviations, including a

half-period phase shift and reduced amplitude. These, along with the more pronounced discrepancies observed in the irregular JONSWAP spectrum test case, are likely attributed to structural modeling assumptions and hydrodynamic simplifications in OpenFAST rather than the coupling toolchain itself. Despite these limitations, REEF2FAST enabled physically realistic wave-structure interaction under complex sea conditions.

By enabling externally computed wavefields to drive OpenFAST simulations, REEF2FAST facilitates the inclusion of complex and realistic wave dynamics in FOWT simulations. This enhancement expands OpenFAST's ability to model shallow water conditions, strong wave nonlinearities, and higher-order wave effects. Furthermore, REEF2FAST lays the foundation for future extensions of OpenFAST's hydrodynamic modeling capabilities, such as the implementation of complex bathymetry.

Due to its file-based, non-invasive structure, the current version of REEF2FAST supports only one-way coupling from wave to structure. While a two-way coupling concept is not implemented in this study, its design is conceptually outlined and remains a target for future development. Prospective research may also focus on the integration of complex bathymetries by leveraging the topographic capabilities of REEF3D, and consequently enabling more physically accurate system-level simulations of structures in nearshore environments in OpenFAST.

# Bibliography

- [1] Thomas Kvarts et al. “Hornsea Projects 1 and 2 – Design and Optimisation of the Cables for the World’s Largest Offshore Wind Farms”. In: *JICABLE’19 – 10th International Conference on Insulated Power Cables*. 6 pages, presented at Jicable HVDC’13, publication reference: JICABLE’19, REE N°5/2019. Jicable, 2019. doi: [10.23723/1301:2019-5/94786](https://doi.org/10.23723/1301:2019-5/94786). URL: <https://doi.org/10.23723/1301:2019-5/94786>.
- [2] Samuel J. Welch et al. *Hywind Tampen Floating Offshore Wind Farm: Sound Source Characterisation of Operational Floating Turbines*. Technical Report Document 03642, Version 1.0. Contract 4504323516, P001802-001. Submitted to Equinor Energy AS: JASCO Applied Sciences (UK) Ltd, 2025.
- [3] WAMIT, Inc. *WAMIT: Wave Analysis MIT*. <https://www.wamit.com/>. Accessed: 2025-06-04. 2025.
- [4] Orcina Ltd. *OrcaFlex: Offshore Dynamics Software*. <https://www.orcina.com/orcafex/>. Accessed: 2025-06-04. 2025.
- [5] Hans Bihs. *REEF3D: An open-source CFD program for wave-structure interaction*. <https://github.com/REEF3D/REEF3D>. Accessed: 2025-06-04. 2025.
- [6] R Kimball et al. “Results from the FOCAL experiment campaign 1: turbine control co-design”. In: *Journal of Physics: Conference Series* 2265.2 (2022), p. 022082. doi: [10.1088/1742-6596/2265/2/022082](https://doi.org/10.1088/1742-6596/2265/2/022082). URL: <https://dx.doi.org/10.1088/1742-6596/2265/2/022082>.
- [7] A. Robertson et al. *Definition of the OC5 DeepCwind Semisubmersible Floating System*. Tech. rep. NREL/TP-5000-63097. OC5 Phase II – Offshore Code Comparison Collaboration Continuation. Golden, CO, USA: National Renewable Energy Laboratory (NREL), 2016.
- [8] Yali Zhang et al. “Hydrodynamic analysis and validation of the floating DeepCwind semi-submersible under 3-h irregular wave with the HOS and CFD coupling method”. In: *Ocean Engineering* 287 (2023), p. 115701. ISSN: 0029-8018. doi: <https://doi.org/10.1016/j.oceaneng.2023.115701>. URL: <https://www.sciencedirect.com/science/article/pii/S0029801823020851>.
- [9] Bonaventura Tagliafierro et al. “Numerical validations and investigation of a semi-submersible floating offshore wind turbine platform interacting with ocean waves using an SPH framework”. In: *Applied Ocean Research* 141 (2023), p. 103757. ISSN: 0141-1187. doi: <https://doi.org/10.1016/j.apor.2023.103757>. URL: <https://www.sciencedirect.com/science/article/pii/S0141118723002985>.
- [10] National Renewable Energy Laboratory (NREL). *OpenFAST: A Modular Platform for Wind Turbine Simulation*. <https://github.com/OpenFAST/openfast>. Version 4.0.2, Accessed: 2025-05-02. 2025.

- [11] Marco Masciola et al. *Investigation of a FAST-OrcaFlex Coupling Module for Integrating Turbine and Mooring Dynamics of Offshore Floating Wind Turbines: Preprint*. Tech. rep. NREL/CP-5000-53197. Presented at the 30th International Conference on Ocean, Offshore and Arctic Engineering (OMAE), Rotterdam, Netherlands. Golden, CO, USA: National Renewable Energy Laboratory (NREL), 2011. URL: <https://www.nrel.gov/docs/fy12osti/53197.pdf>.
- [12] Camille Chauvigné et al. “FRyFAST: A Coupling Between FRyDoM and OpenFAST for the Simulation of Floating Offshore Wind Turbines With High Complexity Platforms”. In: *Proceedings of the ASME 2022 41st International Conference on Ocean, Offshore and Arctic Engineering (OMAE)*. Volume 8: Ocean Renewable Energy. Hamburg, Germany: ASME, 2022, V008T09A029. doi: 10.1115/OMAE2022-79155. URL: <https://doi.org/10.1115/OMAE2022-79155>.
- [13] Guillén Campaña-Alonso et al. “OF2: Coupling OpenFAST and OpenFOAM for High-Fidelity Aero-Hydro-Servo-Elastic FOWT Simulations”. In: *Wind Energy Science* 8 (2023), pp. 1597–1611. doi: 10.5194/wes-8-1597-2023. URL: <https://wes.copernicus.org/articles/8/1597/2023/>.
- [14] Young Jun Kim et al. “High Fidelity Simulations of a Floating Offshore Wind Turbine in Irregular Waves by Coupling OpenFOAM and OpenFAST”. In: *Renewable Energy* 215 (2025), p. 122486. doi: 10.1016/j.renene.2025.122486. URL: <https://doi.org/10.1016/j.renene.2025.122486>.
- [15] Wooyoung Jeon, Sunho Park, and Seokkyu Cho. “Moored motion prediction of a semi-submersible offshore platform in waves using an OpenFOAM and MoorDyn coupled solver”. In: *International Journal of Naval Architecture and Ocean Engineering* 15 (2023), p. 100544. ISSN: 2092-6782. doi: <https://doi.org/10.1016/j.ijnaoe.2023.100544>. URL: <https://www.sciencedirect.com/science/article/pii/S209267822300033X>.
- [16] Leonard Willeke and Benjamin Uekermann. *A preCICE-OpenFAST Adapter to Couple OpenFAST to CFD Simulation Tools*. 2022. doi: 10.5281/zenodo.14809780. URL: <https://zenodo.org/records/14809780>.
- [17] preCICE Consortium. *preCICE - The Coupling Library*. <https://precice.org>. Accessed: 2025-06-03. 2024.
- [18] Jinlong He et al. “Coupled Aero–Hydrodynamic Analysis in Floating Offshore Wind Turbines: A Review of Numerical and Experimental Methodologies”. In: *Journal of Marine Science and Engineering* 12.12 (2024). ISSN: 2077-1312. doi: 10.3390/jmse12122205. URL: <https://www.mdpi.com/2077-1312/12/12/2205>.
- [19] Shimin Yu et al. “Modelling the hydrodynamic response of a floating offshore wind turbine – a comparative study”. In: *Applied Ocean Research* 155 (2025), p. 104441. ISSN: 0141-1187. doi: <https://doi.org/10.1016/j.apor.2025.104441>.
- [20] Lucas Carmo et al. *Validation of Local Structural Loads Computed by OpenFAST Against Measurements From the FOCAL Experimental Campaign*. Conference Paper, NREL/CP-5000-88671. Golden, CO, USA, 2024. URL: <https://www.nrel.gov/docs/fy24osti/88671.pdf>.
- [21] Per A. Madsen and Ole R. Sørensen. “A new form of the Boussinesq equations with improved linear dispersion characteristics. Part 2. A slowly-varying bathymetry”. In: *Coastal Engineering* 18.3 (1992), pp. 183–204. ISSN: 0378-3839. doi: [https://doi.org/10.1016/0378-3839\(92\)90019-Q](https://doi.org/10.1016/0378-3839(92)90019-Q). URL: <https://www.sciencedirect.com/science/article/pii/037838399290019Q>.

- [22] Friedrich-Karl Benra et al. “A Comparison of One-Way and Two-Way Coupling Methods for Numerical Analysis of Fluid-Structure Interactions”. In: *Journal of Applied Mathematics* 2011.1 (2011), p. 853560. doi: <https://doi.org/10.1155/2011/853560>. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1155/2011/853560>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1155/2011/853560>.
- [23] Yuri Bazilevs, Kenji Takizawa, and Tayfun E. Tezduyar. *Computational Fluid–Structure Interaction: Methods and Applications*. John Wiley & Sons, 2013. ISBN: 978-0-470-97409-3. URL: <https://onlinelibrary.wiley.com/doi/book/10.1002/9781118483565>.
- [24] C. Michler et al. “A monolithic approach to fluid–structure interaction”. In: *Computers & Fluids* 33.5 (2004). Applied Mathematics for Industrial Flow Problems, pp. 839–848. ISSN: 0045-7930. doi: <https://doi.org/10.1016/j.compfluid.2003.06.006>. URL: <https://www.sciencedirect.com/science/article/pii/S0045793003000951>.
- [25] Carlos A. Felippa, K.C. Park, and Charbel Farhat. “Partitioned analysis of coupled mechanical systems”. In: *Computer Methods in Applied Mechanics and Engineering* 190.24 (2001). Advances in Computational Methods for Fluid-Structure Interaction, pp. 3247–3270. ISSN: 0045-7825. doi: [https://doi.org/10.1016/S0045-7825\(00\)00391-1](https://doi.org/10.1016/S0045-7825(00)00391-1). URL: <https://www.sciencedirect.com/science/article/pii/S0045782500003911>.
- [26] Jan Vierendeels et al. “Implicit coupling of partitioned fluid–structure interaction problems with reduced order models”. In: *Computers & Structures* 85.11 (2007). Fourth MIT Conference on Computational Fluid and Solid Mechanics, pp. 970–976. ISSN: 0045-7949. doi: <https://doi.org/10.1016/j.compstruc.2006.11.006>. URL: <https://www.sciencedirect.com/science/article/pii/S0045794906003865>.
- [27] P. Causin, J.F. Gerbeau, and F. Nobile. “Added-mass effect in the design of partitioned algorithms for fluid–structure problems”. In: *Computer Methods in Applied Mechanics and Engineering* 194.42 (2005), pp. 4506–4527. ISSN: 0045-7825. doi: <https://doi.org/10.1016/j.cma.2004.12.005>. URL: <https://www.sciencedirect.com/science/article/pii/S0045782504005328>.
- [28] Weizhi Wang et al. “An Improved Depth-Averaged Nonhydrostatic Shallow Water Model with Quadratic Pressure Approximation”. In: *International Journal for Numerical Methods in Fluids* 92.8 (2020), pp. 803–824. doi: <10.1002/fld.4807>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.4807>.
- [29] Weizhi Wang et al. “A Flexible Fully Nonlinear Potential Flow Model for Wave Propagation over the Complex Topography of the Norwegian Coast”. In: *Applied Ocean Research* 122 (2022), p. 103103. doi: <10.1016/j.apor.2022.103103>. URL: <https://www.sciencedirect.com/science/article/pii/S0141118722000566>.
- [30] Hans Bihs and Widar Weizhi Wang. *REEF3D::NHFLOW – A High-Performance Non-Hydrostatic Solver for Coastal Wave Propagation*. Available at SSRN: <https://ssrn.com/abstract=5145387> or <http://dx.doi.org/10.2139/ssrn.5145387>. 2024.
- [31] Hans Bihs, Ronja Ehlers, and Widar Weizhi Wang. *A Shock-Absorbing Non-Hydrostatic Navier–Stokes Solver on  $\sigma$ -Grids for Wave Modeling Over Irregular Topography*. June 2024. doi: <10.1115/OMAE2024-127429>. eprint: <https://asmedigitalcollection.asme.org/OMAE/proceedings-pdf/OMAE2024/87844/V006T08A016/7361621/v006t08a016-omae2024-127429.pdf>. URL: <https://doi.org/10.1115/OMAE2024-127429>.
- [32] Bin Li and Christopher A. Fleming. “A Three Dimensional Multigrid Model for Fully Nonlinear Water Waves”. In: *Coastal Engineering* 30.3 (1997), pp. 235–258. doi: [10.1016/S0378-3839\(96\)00046-4](10.1016/S0378-3839(96)00046-4). URL: <https://www.sciencedirect.com/science/article/pii/S0378383996000464>.

- [33] Volker Roeber, Kwok Fai Cheung, and Marcelo H. Kobayashi. “Shock-Capturing Boussinesq-Type Model for Nearshore Wave Processes”. In: *Coastal Engineering* 57.4 (2010), pp. 407–423. doi: [10.1016/j.coastaleng.2009.11.007](https://doi.org/10.1016/j.coastaleng.2009.11.007). URL: <https://www.sciencedirect.com/science/article/pii/S0378383909001860>.
- [34] Gangfeng Ma, Fengyan Shi, and James T. Kirby. “Shock-Capturing Non-Hydrostatic Model for Fully Dispersive Surface Wave Processes”. In: *Ocean Modelling* 43–44 (2012), pp. 22–35. doi: [10.1016/j.ocemod.2011.12.002](https://doi.org/10.1016/j.ocemod.2011.12.002). URL: <https://www.sciencedirect.com/science/article/pii/S1463500311001892>.
- [35] Qiuhua Liang and Fabien Marche. “Numerical Resolution of Well-Balanced Shallow Water Equations With Complex Source Terms”. In: *Advances in Water Resources* 32.6 (2009), pp. 873–884. doi: [10.1016/j.advwatres.2009.02.010](https://doi.org/10.1016/j.advwatres.2009.02.010). URL: <https://www.sciencedirect.com/science/article/pii/S0309170809000396>.
- [36] Ahmet Soydan, Widar W. Wang, and Hans Bihs. “An improved direct forcing immersed boundary method for floating body simulations in waves”. In: *Applied Ocean Research* 158 (2025), p. 104523. ISSN: 0141-1187. doi: <https://doi.org/10.1016/j.apor.2025.104523>.
- [37] Guang-Shan Jiang and Chi-Wang Shu. “Efficient Implementation of Weighted ENO Schemes”. In: *Journal of Computational Physics* 126.1 (1996), pp. 202–228. ISSN: 0021-9991. doi: [10.1006/jcph.1996.0130](https://doi.org/10.1006/jcph.1996.0130). URL: <https://www.sciencedirect.com/science/article/pii/S0021999196901308>.
- [38] Amiram Harten, Peter D. Lax, and Bram van Leer. “On Upstream Differencing and Godunov-Type Schemes for Hyperbolic Conservation Laws”. In: *SIAM Review* 25.1 (1983), pp. 35–61. doi: [10.1137/1025002](https://doi.org/10.1137/1025002). URL: <https://doi.org/10.1137/1025002>.
- [39] Sigal Gottlieb and Chi-Wang Shu. “Total Variation Diminishing Runge-Kutta Schemes”. In: *Mathematics of Computation* 67.221 (Jan. 1998), pp. 73–85. ISSN: 0025-5718. doi: [10.1090/S0025-5718-98-00913-2](https://doi.org/10.1090/S0025-5718-98-00913-2). URL: <https://doi.org/10.1090/S0025-5718-98-00913-2>.
- [40] Alexandre Chorin. “Numerical Solution of the Navier–Stokes Equations”. In: *Mathematics of Computation* 22 (Oct. 1968), pp. 745–762. doi: [10.2307/2004575](https://doi.org/10.2307/2004575). URL: <https://doi.org/10.2307/2004575>.
- [41] Steven F. Ashby and Robert D. Falgout. “A Parallel Multigrid Preconditioned Conjugate Gradient Algorithm for Groundwater Flow Simulations”. In: *Nuclear Science and Engineering* 124.1 (1996), pp. 145–159. doi: [10.13182/NSE96-A24230](https://doi.org/10.13182/NSE96-A24230). URL: <https://doi.org/10.13182/NSE96-A24230>.
- [42] Hans Bihs et al. “A New Level Set Numerical Wave Tank with Improved Density Interpolation for Complex Wave Hydrodynamics”. In: *Computers & Fluids* 140 (2016), pp. 191–208. ISSN: 0045-7930. doi: [10.1016/j.compfluid.2016.09.012](https://doi.org/10.1016/j.compfluid.2016.09.012). URL: <https://www.sciencedirect.com/science/article/pii/S0045793016302729>.
- [43] Joseph O’Rourke. *Computational Geometry in C*. 2nd ed. Cambridge University Press, 1998.
- [44] Tso-Ren Wu. “A Numerical Study of Three-Dimensional Breaking Waves and Turbulence Effects”. PhD thesis. Cornell University, 2004.
- [45] Jun Zang et al. “Experimental Study of Non-Linear Wave Impact on Offshore Wind Turbine Foundations”. In: *Proceedings of Coastlab2010 – 3rd International Conference on the Application of Physical Modelling to Port and Coastal Protection*. Barcelona, Spain, Oct. 2010.
- [46] Kai Irschik, Uwe Sparboom, and Hocine Oumeraci. “Breaking Wave Characteristics for the Loading of a Slender Pile”. In: *Coastal Engineering* 2002. 2002, pp. 1341–1352. doi: [10.1142/9789812791306\\_0113](https://doi.org/10.1142/9789812791306_0113). eprint: [https://worldscientific.com/doi/pdf/10.1142/9789812791306\\_0113](https://worldscientific.com/doi/pdf/10.1142/9789812791306_0113). URL: [https://worldscientific.com/doi/abs/10.1142/9789812791306\\_0113](https://worldscientific.com/doi/abs/10.1142/9789812791306_0113).

- [47] Patrick J. Moriarty and Andrew C. Hansen. *AeroDyn Theory Manual*. Tech. rep. NREL/TP-500-36881. Golden, CO, USA: National Renewable Energy Laboratory (NREL), 2005. doi: [10.2172/15014831](https://doi.org/10.2172/15014831). URL: <https://www.osti.gov/biblio/15014831>.
- [48] National Renewable Energy Laboratory (NREL). *AeroDyn User's Guide and Theory Manual*. 2025. URL: <https://openfast.readthedocs.io/en/main/source/user/aerodyn/index.html>.
- [49] Lu Wang et al. “Recent Hydrodynamic Modeling Enhancements in OpenFAST”. In: *Proceedings of the ASME 2022 4th International Offshore Wind Technical Conference (IOWTC)*. Vol. V001T01A004. International Conference on Offshore Mechanics and Arctic Engineering. Dec. 2022. doi: [10.1115/IOWTC2022-98094](https://doi.org/10.1115/IOWTC2022-98094). eprint: <https://asmedigitalcollection.asme.org/OMAE/proceedings-pdf/IOWTC2022/86618/V001T01A004/6977388/v001t01a004-iowtc2022-98094.pdf>. URL: <https://doi.org/10.1115/IOWTC2022-98094>.
- [50] National Renewable Energy Laboratory (NREL). *HydroDyn User's Guide and Theory Manual*. 2025. URL: <https://openfast.readthedocs.io/en/main/source/user/hydrodyn/index.html#>.
- [51] National Renewable Energy Laboratory (NREL). *ElastoDyn User's Guide and Theory Manual*. 2025. URL: <https://openfast.readthedocs.io/en/main/source/user/elastodyn/>.
- [52] Andy Platt, Bonnie Jonkman, and Jason Jonkman. *InflowWind User's Guide*. Tech. rep. National Renewable Energy Laboratory (NREL), 2016. URL: <https://openfast.readthedocs.io/en/main/source/user/inflowwind/index.html>.
- [53] National Renewable Energy Laboratory (NREL). *SeaState User's Guide*. 2025. URL: <https://openfast.readthedocs.io/en/main/source/user/seastate/index.html>.
- [54] National Renewable Energy Laboratory (NREL). *ServoDyn User's Guide*. Tech. rep. National Renewable Energy Laboratory (NREL), 2024. URL: <https://openfast.readthedocs.io/en/main/source/user/servodyn/>.
- [55] Matthew Hall. “MoorDyn V2: New Capabilities in Mooring System Components and Load Cases”. In: *Proceedings of the ASME 2020 39th International Conference on Ocean, Offshore and Arctic Engineering (OMAE)*. Volume 9: Ocean Renewable Energy. Virtual, Online: ASME, Aug. 2020, V009T09A078. doi: [10.1115/OMAE2020-19341](https://doi.org/10.1115/OMAE2020-19341). eprint: <https://asmedigitalcollection.asme.org/OMAE/proceedings-pdf/OMAE2020/84416/V009T09A078/6607556/v009t09a078-omae2020-19341.pdf>. URL: <https://doi.org/10.1115/OMAE2020-19341>.
- [56] Rick Damiani and Jason Jonkman. *SubDyn User's Guide and Theory Manual*. Tech. rep. NREL/TP-5000-63065. Available via U.S. Department of Energy OSTI. Golden, CO, USA: National Renewable Energy Laboratory (NREL), 2015. doi: [10.2172/1225918](https://doi.org/10.2172/1225918). URL: <https://www.osti.gov/biblio/1225918>.
- [57] National Renewable Energy Laboratory (NREL). *MoorDyn User's Guide and Theory Manual*. 2025. URL: <https://openfast.readthedocs.io/en/main/source/user/moordyn/index.html>.
- [58] Keyvan Sadeghi and Atilla Incecik. “Tensor Properties of Added-Mass and Damping Coefficients”. In: *Journal of Engineering Mathematics* 52.4 (Aug. 2005), pp. 379–387. doi: [10.1007/s10665-004-6090-6](https://doi.org/10.1007/s10665-004-6090-6). URL: <https://doi.org/10.1007/s10665-004-6090-6>.
- [59] C Lee. “WAMIT Theory manual”. In: *Massachusetts Institute of Tech: Cambridge, MA, USA* (1995).

- [60] Jason Jonkman et al. *Implementation of Substructure Flexibility and Member-Level Load Capabilities for Floating Offshore Wind Turbines in OpenFAST*. American English. Technical Report. 2020. doi: [10.2172/1665796](https://doi.org/10.2172/1665796).
- [61] L. Dagum and R. Menon. “OpenMP: an industry standard API for shared-memory programming”. In: *IEEE Computational Science and Engineering* 5.1 (1998), pp. 46–55. doi: [10.1109/99.660313](https://doi.org/10.1109/99.660313).
- [62] Jon Louis Bentley. “Multidimensional binary search trees used for associative searching”. In: *Commun. ACM* 18.9 (1975), pp. 509–517. issn: 0001-0782. doi: [10.1145/361002.361007](https://doi.org/10.1145/361002.361007). url: <https://doi.org/10.1145/361002.361007>.
- [63] Donald Shepard. “A two-dimensional interpolation function for irregularly-spaced data”. In: *Proceedings of the 1968 23rd ACM National Conference*. ACM ’68. New York, NY, USA: Association for Computing Machinery, 1968, pp. 517–524. isbn: 9781450374866. doi: [10.1145/800186.810616](https://doi.org/10.1145/800186.810616). url: <https://doi.org/10.1145/800186.810616>.
- [64] J.D. Wheeler. “Method for Calculating Forces Produced by Irregular Waves”. In: *Journal of Petroleum Technology* 22.03 (Mar. 1970), pp. 359–367. issn: 0149-2136. doi: [10.2118/2712-PA](https://doi.org/10.2118/2712-PA). eprint: <https://onepetro.org/JPT/article-pdf/22/03/359/2228185/spe-2712-pa.pdf>. url: <https://doi.org/10.2118/2712-PA>.
- [65] Wind Data Hub. *OC5 Phase II Dataset*. Maintained by Wind Data Hub for U.S. Department of Energy, Office of Energy Efficiency and Renewable Energy. 2025. doi: [10.21947/1574499](https://doi.org/10.21947/1574499). url: <https://doi.org/10.21947/1574499> (Accessed: 05/13/2025).
- [66] Bernard Méhauté. *An Introduction to Hydrodynamics and Water Waves*. 1st ed. Springer Study Edition. 89 b/w illustrations. Springer Berlin, Heidelberg, 1976. VIII + 323. isbn: 978-3-642-85567-2. doi: [10.1007/978-3-642-85567-2](https://doi.org/10.1007/978-3-642-85567-2). url: <https://doi.org/10.1007/978-3-642-85567-2>.
- [67] J. A. Helder and M. Pietersma. *UMAINE – DEEPCWIND/OC4 SEMI FLOATING WIND TURBINE REPEAT TESTS*. Tech. rep. University of Maine, Aug. 2017. doi: [10.2172/1375023](https://doi.org/10.2172/1375023). url: <https://www.osti.gov/biblio/1375023>.

