Offshore Wind Turbine Model Development from In-Situ Measurements: Model Calibration and Uncertainty Quantification

Rad Haghi , Babak Moaveni * , Abani Patra , Eric Hines

Posted Date: 29 April 2026

doi: 10.20944/preprints202604.2052.v1

Keywords: offshore wind turbine; model calibration; uncertainty quantification; OpenFAST; inverse design; structural health monitoring

 doi:10.20944/preprints202604.2052.v1

Article

Offshore Wind Turbine Model Development from In-Situ Measurements: Model Calibration and Uncertainty Quantification

Rad Haghi [1,2], Babak Moaveni [1,*], Abani Patra [2] and Eric Hines [1]

[1] Department of Civil and Environmental Engineering, Tufts University, Medford, MA, USA
[2] Tufts Institute for Artificial Intelligence, Tufts University, Medford, MA, USA
* Correspondence: babak.moaveni@tufts.edu

Abstract

This study presents a framework for developing, emulating, and validating offshore wind turbine models when proprietary blade designs are unavailable. The methodology addresses a critical industry challenge by demonstrating that accurate aero-servo-hydro-elastic models can be constructed using only publicly available reference designs and operational measurements. An inverse design approach based on differential evolution optimization reconstructs blade aerodynamic characteristics from field data, enabling the creation of models that replicate operational behavior without requiring access to proprietary geometries. The framework incorporates comprehensive uncertainty quantification through machine learning techniques to predict simulation errors based on environmental and operational conditions. Validation against extensive field measurements from an operational offshore wind turbine demonstrates the effectiveness of the methodology. This approach offers a practical pathway for model calibration and error prediction for offshore wind turbines, particularly when complete design documentation is unavailable.

Keywords: offshore wind turbine; model calibration; uncertainty quantification; OpenFAST; inverse design; structural health monitoring

1. Introduction

Mid-fidelity aero-servo-hydro-elastic models for an offshore wind turbine (OWT) have become essential tools for accurately simulating OWT structural dynamics, loads, and operational performance [1,2]. To build confidence in these numerical models, extensive calibration and validation efforts against operational measurements—primarily SCADA (Supervisory Control and Data Acquisition) systems and strain gauge data—have been widely pursued [3,4].

Early foundational studies focused on onshore turbines demonstrated comprehensive validation techniques employing statistical and time-series analyses. Guntur et al. validated the OpenFAST model using extensive field data from a 2.3 MW Siemens turbine at the National Laboratory of the Rockies (NLR). This study leveraged IEC 61400-13 guidelines to statistically compare simulated loads to high-resolution strain gauge measurements, confirming the accuracy improvements in OpenFAST. Brown et al. expanded upon these methods by reconstructing site-specific turbulent wind fields to enable direct one-to-one comparisons between measured and simulated performance metrics. While achieving high accuracy in power and thrust predictions, discrepancies remained in fatigue load estimations, highlighting limitations inherent in aerodynamic damping and controller parameter modeling.

Calibration and validation for offshore turbines introduce additional complexities, including hydrodynamic interactions and support structure dynamics. Jonkman and Robertson validate an offshore wind turbine model on a jacket foundation by comparing the results of different aero-servo-hydro-elastic codes, as accessing measurement data proved challenging. Driscoll et al. provided

 doi:10.20944/preprints202604.2052.v1

a landmark validation of the Hywind floating turbine, demonstrating close agreement in power output, structural loads, and platform motions. Similarly, validation efforts for fixed-bottom turbines at the Alpha Ventus wind farm have emphasized the importance of calibrating external influences such as wake-induced turbulence [4,6]. Kretschmer et al. systematically calibrated wake turbulence parameters and sensor biases using measured SCADA and strain gauge data, enabling accurate modeling of downstream turbine loads under waked conditions.

Further studies expanded model calibration through optimization methods. Sanderse et al. utilized Bayesian calibration frameworks to refine aerodynamic model parameters, significantly reducing discrepancies between simulations and experimental data. Genetic algorithms and particle swarm optimization methods have also been widely applied to systematically calibrate model parameters to match measured power, torque, and thrust [8–10]. Such optimization-based calibration approaches demonstrate improved model accuracy and reliability, critical for fatigue and structural health predictions. For instance, Stewart and Lackner [8] optimized blade aerodynamic parameters to closely match experimental torque and thrust measurements, while Xu et al. [10] applied particle swarm optimization to refine structural stiffness and mass parameters based on vibration measurements, resulting in improved modal accuracy.

Despite these advancements, a persistent gap remains in scenarios where proprietary design data (particularly blade geometry and airfoil characteristics) are inaccessible to researchers, as is commonly encountered with operational utility-scale wind turbines. This limitation hinders the accuracy and applicability of traditional calibration methods to build a full functional mid-fidelity aero-servo-hydro-elastic model. Veers et al. underlined that the availability of utility-scale wind turbine data is often restricted by proprietary rights held by private companies, hindering progress in turbine modeling. Similarly, Griffith et al. emphasize that commercial blade specifications are typically proprietary, leading researchers to depend on obsolete or less relevant reference models. Most existing validation studies have relied on special agreements with manufacturers to access detailed turbine design documentation. For example, Brown et al. and Guntur et al. provided complete blade geometry and airfoil data by the wind turbine manufacturer, or Driscoll et al. had access to design data provided by the utility and wind turbine manufacturing companies. These levels of manufacturer-researcher collaboration are rarely available to the broader research community.

To address this gap, this study proposes an inverse-design optimization methodology that utilizes publicly available reference turbines as a baseline. The primary contribution of this work is the development of a systematic approach to calibrate and tailor a publicly available wind turbine model to represent a specific utility-scale wind turbine using limited field measurements from strain gauges and SCADA data.

Utilizing one month of high-resolution SCADA data and measured bending moments along the tower and foundation time series from a 6 MW offshore wind turbine operating in the North Sea, we optimized the blade geometry and airfoils within WISDEM framework. The primary objective was to achieve identical aerodynamic torque and thrust characteristics across the operating envelope without requiring explicit knowledge of the proprietary blade geometry. The optimized rotor configuration was implemented in OpenFAST and validated comprehensively against measured SCADA and load data, confirming close alignment in aerodynamic performance metrics. This inverse-optimization framework thus provides a practical methodology to develop validated aeroelastic models for turbines lacking publicly available design details, significantly broadening the potential for structural dynamic analyses and predictive assessments for operational offshore wind turbines. In addition, we build a large dataset of simulations based on to the environmental and operational conditions (EOCs) and developed a data-driven model to predict the error of the OpenFAST model.

This manuscript is organized as follows. In Section 2 we describe the methodology utilized in this piece of work extensively without losing the general picture. In Section 3 we introduce the data set from the 6 MW OWT that we have access too. Afterwards, in Section 4 we show the results of the aero-servo-hydro-elastic time marching simulations output employing the developed OWT model. In

 doi:10.20944/preprints202604.2052.v1

the same section, we quantify the error between the simulations and measurement and show it can be predicted using measured environmental and operational conditions (EOCs). This manuscript ends in Section 5 where we present the conclusion and future works.

2. Methodology

The methodology used in this paper to build and calibrate an aero-hydro-servo-elastic model of an OWT from measurements is presented in Figure 1. This involves starting from an initial design, which is based on a publicly available wind turbine design, and moving towards the optimized blade while the constraints are based on the data collected from the turbine.

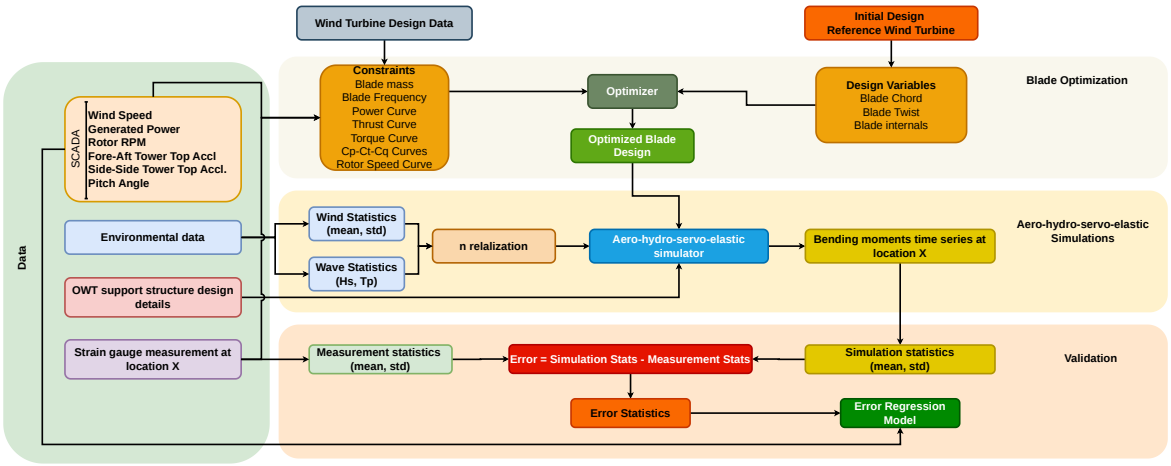


Figure 1. Comprehensive framework for OWT model calibration and validation using a data-driven approach. The framework consists of four main components: (1) Data collection including measurements, environmental conditions, and structural design details; (2) Blade optimization using differential evolution to match target aerodynamic characteristics; (3) Aero-hydro-servo-elastic simulations with multiple realizations; and (4) Validation through error quantification and regression modeling.

The configuration presented in Figure 1 has four building blocks, which are briefly explained below:

- **Data**: The vertical Data block is the backbone of the process. It includes all data collected from the measurement campaign of the OWT, environmental data from the OWT site, and the support structure (tower, foundation and transition piece) design details.
- **Optimization**: This shows the general process of blade parameters optimization. The aim is to develop a blade design that can generate the same torque and thrust as the OWT.
- **Aero-hydro-servo-elastic Simulations**: The main goal here is generating a large synthetic simulations dataset based on the optimized blade design. The input to the simulations is the wind and wave (if accessible) measurement, at the same time stamps that the strain gauge collected data. As the wind and wave input are based on statistics, it is essential to populate the input by having n realizations from those statistics. Also, the output of the model, which is bending moment time series, should be at the same location on the support structure that we have the strain gauge. These two considerations are crucial for comparing simulations against measurements and calculating the error.
- **Validation**: After optimizing the blade and generating the simulation dataset, it is important to quantify the error in the model. To do so, we calculate the difference between the measurements and the corresponding simulations. Here we calculate two sets of errors. The first is the average of error/difference over n simulations for a given EOC, which is defined as ErrorMean. The second is the standard deviation of the error/difference over these n simulations, defined as ErrorStd.
- **Error modeling**: In the last step, we build a data-driven regression model that maps the EOCs to the ErrorMean and ErrorStd.

 doi:10.20944/preprints202604.2052.v1

There are two other inputs to the optimization procedure. One is the available data from the wind turbine, which helps define the optimization constraints. The second is the initial design for optimizing the turbine blades. As utility-scale wind turbine designs are not publicly available, this needs to be a publicly available reference wind turbine blade design. In the continuation of this section, we explain the optimization, aero-hydro-servo-elastic simulation and validation in detail. The 6MW OWT measurement dataset is described extensively in Section 3.

2.1. Blade Parameters Optimization

A wind turbine energy extraction from wind is mainly dictated by its aerodynamic design. The core of a wind turbine aerodynamic comes from the airfoils which are used in the blade design. For the utility scale wind turbines, these airfoils are intellectual property of the wind turbine manufacturing company and are protected. These airfoils are designed and optimized through a complex multi-objective optimization procedure to produce maximum power while minimizing material cost and satisfying the required structural properties of the design [13]. As we do not have access to these airfoils, we are using publicly available airfoil designs in the initial design. However, we aim to discover effective blade geometries that mimic the target turbine by choosing the blade cost per generated energy unit in the turbine's lifetime (BCE) as the objective function. It means, the optimizer goal is generating more power, while it tries to minimize the material cost of the blade, when the constraints for blade stiffness and mass are respected. The BCE can be formulated as:

$$BCE = \frac{Blade\ Cost}{Life\ time\ energy\ production} \tag{1}$$

The effectiveness of the optimized design from this approach depends on the intended output of the model. If the objective is to analyze the loads on the blade roots or hub, this approach is not suitable, as we do not have precise information about the aerodynamics of a utility-scale wind turbine. However, if the goal is to assess the bending moments along the support structure and the power generated during operation, it is sufficient to utilize a blade design that produces the same power, thrust and torque as the utility-scale wind turbine. This latter scenario is the focus of the optimization output.

The core of any optimization procedure is the optimizer algorithm. The optimization algorithms have been developed extensively in the last decades [14,15]. The proper optimization algorithm for the problem at hand is crucial for the optimizer to converge. The blade optimization problem has a complex and non-convex design space. As there is lack of smoothness in response surface for gradinet-based methods, we chose an Evolutionary Algorithm (EA) as the optimization algorithm. EAs offer significant advantages over gradient-based optimization methods, especially in engineering design problems characterized by complex, non-convex, or discontinuous objective functions. Unlike gradient-based methods, which require derivative information and may struggle with local minima, EAs do not rely on gradient information, making them suitable for problems where gradients are difficult or impossible to compute. This characteristic enables EAs to navigate complex design spaces effectively, avoiding entrapment in local minima which is a common issue with gradient descent approaches [14]. Among EAs, we utilized Differential Evolution (DE) due to its ease of tuning and implementation. DE is a population-based evolutionary algorithm [16] for global optimization. DE iteratively improves a population of candidate solutions using mutation, crossover (also known as recombination), and selection operators. A key feature of DE is its differential mutation strategy, in which the difference between two randomly selected population members is scaled by a factor and added to a third member to form a mutant solution. This mutant is then recombined with the current solution to produce a trial vector, which replaces the original if it yields an improved objective value. As DE is only using a few control parameters, it is practical and easy to implement in a complex design space.

For this optimization procedure, another requirement is the constraints. Here, we defined the constraints primarily based on the information available to us from the measurement campaign and

 doi:10.20944/preprints202604.2052.v1

the information we had access to about the wind turbine. Table 1 shows the constraints that we considered for this work. The presented upper and lower bounds for the constraints were determined experimentally.

Table 1. Employed constraints for blade optimization with their source and upper and lower bounds

Constraint	Source	Bounds
Blade mass	Wind turbine design data	±1%
Blade tip tower distance	Wind turbine design data	±1%
Rated wind speed	SCADA and wind turbine design data	±10%
Rated rotor speed	SCADA and wind turbine design data	±5%
Rated Thrust	FA moments at location X	±5%
Rated Torque	SCADA	±5%
Power curve	SCADA	±5%
Cp-Ct-Cq curves	FA moments at location X and SCADA	±2%

The rotor thrust can be calculated directly from the moments measurement at any location along the support structure. There is an additional piece of information needed for this which is the Rotor Nacelle Assembly (RNA) overhang distance. For the majority of the utility size wind turbines the RNA center of mass is not directly above the tubular support structure centreline. This offset causes a bending moment in Fore-Aft (FA) direction which need to be taken care of when we calculate the thrust. The thrust and torque calculation can be formulated as:

$$Thrust = \frac{M_{FA} - r_{overhang} m_{RNA} \times g}{r_{SG}} \tag{2}$$

$$Torque = \frac{Power}{Rotor\ Speed} \tag{3}$$

where M_{FA} is the FA bending moment from the measurement, $r_{overhang}$ is the offset, m_{RNA} is the RNA mass, r_{SG} is the strain gauge distance from the hub height, and g is the gravity acceleration.

Having access to the generated power, thrust and torque one can calculate the aerodynamic coefficients of the rotor. The coefficients calculation can be formulated as the following [17]:

$$C_p = \frac{Power}{0.5\rho A V^3} \tag{4}$$

$$C_t = \frac{Thrust}{0.5\rho A V^2} \tag{5}$$

$$C_q = \frac{Torque}{0.5\rho A V^2 R} \tag{6}$$

where $\rho = 1.225\ kg/m^3$ is the chosen air density according to practice and IEC standard [18], A is the rotor area, R is the rotor radius, and V is the wind speed. We have access to the wind speed from the SCADA data.

The objective function is a function of the design variables. The design variables are the blade chord, twist, and internals (spar caps, shear webs, shell/skin layers, leading edge reinforcement and trailing edge reinforcement). The blade chord and twist dictate blade and rotor aerodynamic coefficients. They also affect the blade mass and stiffness (blade tip tower distance). The internal design variables are required to meet the mass and stiffness criteria. We have thus far formulated the optimization problem, identified the solver and objective function, and specified the governing constraints.

 doi:10.20944/preprints202604.2052.v1

2.2. Aero-Hydro-Servo-Elastic Simulations

Upon convergence, the optimizer yields an aerodynamically optimized blade that, when integrated into a rotor, produces the measured power, thrust, and torque. This model, which is the aerodynamic model of the wind turbine, needs to be combined with the support structure model, aeroelastic model, hydrodynamics, and controller to form the aero-hydro-servo-elastic model [19–21]. The controller is the most important submodel, that needs to be tuned accurately to have similar behaviour as the wind turbine that we have access to the measurement. The controller is usually modeled as Proportional-Integral-Derivative (PID) controller which regulates primary generated power and rotor speed by adjusting the pitch angle of the blades [22]. The full explanation of each of these submodels is out of this manuscript scope.

The validation of aero-hydro-servo-elastic simulations requires comparing model predictions with field measurements. However, while the bending moments are available as time series, the corresponding environmental data are provided only as statistics: mean wind speed ($\bar{U}$), wind speed standard deviation (σ_U), significant wave height (H_s), and peak wave period (T_p). The actual time series of wind and wave conditions that produced the measured response is not available. To address this limitation, we generate n synthetic realizations (seeds) of wind and wave time series that conform to the measured environmental statistics. We then perform aero-hydro-servo-elastic simulations for each realization, producing an ensemble of n response time series. From this ensemble, we compute statistics of the bending moment time series and compare these directly with the corresponding statistics derived from the measurements. This approach enables validation despite the absence of measured environmental time series, by ensuring that both simulated and measured responses correspond to the same statistical description of environmental conditions.

For this study, we selected $n = 36$ realizations per measured EOC. This choice represents six times the minimum requirement of six seeds specified in IEC 61400-1 for design load calculations [18]. While the IEC minimum may not guarantee accurate estimation of overall statistics [23], using 36 seeds provides improved statistical convergence while maintaining computational feasibility. Given our computational resources and the need to simulate 1,443 EOC cases observed in the measurement window, this number balances accuracy with the total computational time of approximately 72 hours per measurement case.

The wind and wave statistics can be utilized to generate wind time series and wave height time series. The process of generating these synthetic time series is discussed in [24] for synthetic wind generation and [25].

Having the synthetic wind and wave time series and the aero-hydro-servo-elastic model at hand, one can run the time-marching aero-hydro-servo-elastic simulations. We recommend keeping the length of these time-marching simulations the same as the length of the measurement time series for ease of later comparison. Also, it is important to have the bending moment time series output at the same location(s) along the support structure where the strain gauge are installed. As the final aim is validating the model and uncertainty quantification (UQ) of model prediction, it is important to minimize the manageable differences between the model simulations and measurements as much as possible.

2.3. Validation

So far, we have optimized the design of a wind turbine blade, which, when integrated into a rotor, can generate the same power, thrust, and torque as the wind turbine for which we have access to data. Additionally, we developed an aero-hydro-servo-elastic model of the OWT based on information regarding the support structure and controller. We then simulated the model and registered the bending moment responses for a large number of realizations (seeds) based on the EOC cases observed in the measurement windows. These steps mean we have a dataset with n realizations of each response per EOC unique to each measurement.

To perform the statistical validation, we took the following steps:

 doi:10.20944/preprints202604.2052.v1

- **Measurement statistics**: For every measurement time series window (10-minutes), we calculate the mean μ_{mes}.
- **Simulation statistics**: For every simulation time series corresponding to measurements, we calculate the mean $\boldsymbol{\mu}_{sim}$, which is a 1D array with n members, due to the fact that we have n simulations per each measurement.
- **Error calculation**: The error can be formulated as:

$$\boldsymbol{E} = \mu_{mes} - \boldsymbol{\mu}_{sim} \tag{7}$$

  As we have n realizations per measurement, there are n data points for $\boldsymbol{E}$.
- **Error statistics**: For every measurement, we have populations of $\boldsymbol{E}$ that follow a probability distribution. If the error follows a normal distribution [26], then it can be characterized as:

$$\boldsymbol{E} \sim \mathcal{N}(\mu_{mean}, \sigma_{mean}) \tag{8}$$

2.4. Uncertainty Quantification and Error Modeling

Our aim here is to describe the UQ methodology that we utilized to characterize and model the relationship between the simulation error statistics and the EOCs. UQ is used to systematically assess and quantify uncertainties inherent in computational models and simulations, particularly focusing on input uncertainties, model errors, and numerical inaccuracies. The importance of UQ lies in its ability to provide confidence intervals or probabilistic predictions, thus enhancing the reliability and decision-making capability of simulation-based analyses [27,28].

To quantify the error, we aim to build regression models predicting the distribution parameters in Eq. (8). This can be generalized succinctly as:

$$\mu_{\text{mean}} = f_\mu(x) + \epsilon_\mu, \quad \sigma_{\text{mean}} = f_\sigma(x) + \epsilon_\sigma \tag{9}$$

where $f_\mu(\cdot)$ and $f_\sigma(\cdot)$ are nonlinear functions, x is the EOCs and ϵ_μ and ϵ_μ are the regression errors.

The output of this process enables us to explain the relationship between the EOCs and the error we see in the simulation results. For the regression, we utilized an Artificial Neural Network (ANN) [29]. ANNs are flexible, general-purpose regression models that can approximate arbitrary continuous functions given sufficient network size, reflecting their theoretical status as universal approximators [30,31]. Their layered architecture with nonlinear activation functions enables ANNs to capture complex input-output relationships beyond the reach of linear models [32,33]. ANNs are commonly trained by iterative gradient descent algorithms, using backpropagation to compute gradients for weight updates in order to minimize a loss function [29,34]. These characteristics have enabled ANNs to achieve state-of-the-art performance across diverse domains, underscoring their broad applicability as general-purpose regression tools [35].

Given the complexity of wind turbine dynamics and the potential nonlinear relationships between environmental conditions and simulation errors, ANNs were selected over simpler regression methods. While ANNs outperform linear regression across different applications, particularly when dealing with complex, non-linear relationships and large datasets [36], the choice of method should be justified. For simpler problems with clear linear relationships, small datasets, or when interpretability is crucial, linear regression remains competitive and computationally efficient [37,38]. In our case, preliminary analysis with linear models showed inadequate performance, confirming that the relationship between EOCs and errors exhibits nonlinear behavior that requires the modeling flexibility of ANNs.

To understand which environmental and operational conditions most strongly influence the prediction errors, we employed permutation feature importance analysis [39]. This technique quantifies the contribution of each feature to model performance by measuring the degradation in predictive accuracy when the feature's association with the target variable is disturbed through random permutation. For each feature, the values are randomly shuffled while maintaining all other features and

 doi:10.20944/preprints202604.2052.v1

their inter-relationships intact, thus isolating the predictive contribution of the individual feature on the trained model. The standard deviation across repetitions provides confidence intervals for the importance estimates, with a smaller standard deviation indicating more reliable feature rankings. Features with higher permutation importance values have greater influence on model predictions, as their disruption leads to larger decreases in model performance, thereby identifying the most critical EOCs for the error regression models. This analysis provides valuable insights into which operational and environmental parameters should be monitored most carefully for accurate model predictions. The permutation feature importance is quantified using the Mean Squared Error (MSE) increase factor, defined as:

$$\beta_j = (\mathrm{MSE}_{\mathrm{perm},j}/\mathrm{MSE}_{\mathrm{baseline}}) - 1 \tag{10}$$

where $\mathrm{MSE}_{\mathrm{perm},j}$ represents the mean squared error after randomly permuting feature j across all test samples, and $\mathrm{MSE}_{\mathrm{baseline}}$ is the original model error with all features intact. This metric provides a dimensionless measure of feature importance, where $\beta_j = 0$ indicates no contribution to model performance, $\beta_j = 1$ indicates that permuting the feature doubles the prediction error, and larger values suggest increasingly critical dependence on that feature. For instance, $\beta_j = 10$ implies that corrupting feature j increases the model's MSE by a factor of 11, effectively degrading the model performance to near-random predictions.

The preceding sections have outlined a comprehensive framework for developing, simulating, and validating aero-hydro-servo-elastic models of utility-scale offshore wind turbines, with particular emphasis on uncertainty quantification and model-measurement discrepancies. This work advances the field through two primary contributions: the establishment of a systematic, reproducible methodology for OWT model development and subsequent calibration and validation, and, equally significant, its successful application to actual operational turbine data. The synthesis of methodological innovation with practical implementation using field measurements from commercial OWTs represents, to the authors' knowledge, a novel advancement.

3. Dataset

The dataset analyzed in this study was obtained from a 6 MW offshore wind turbine (OWT) supported by a monopile foundation, managed by Ørsted energy company. The instrumentation setup on this turbine includes a variety of strain gauges and accelerometers, as well as data from a SCADA system. Accelerometer, strain gauges and SCADA data are recorded simultaneously. A summary of the available measurement channels and their sampling rates is provided in Table 2.

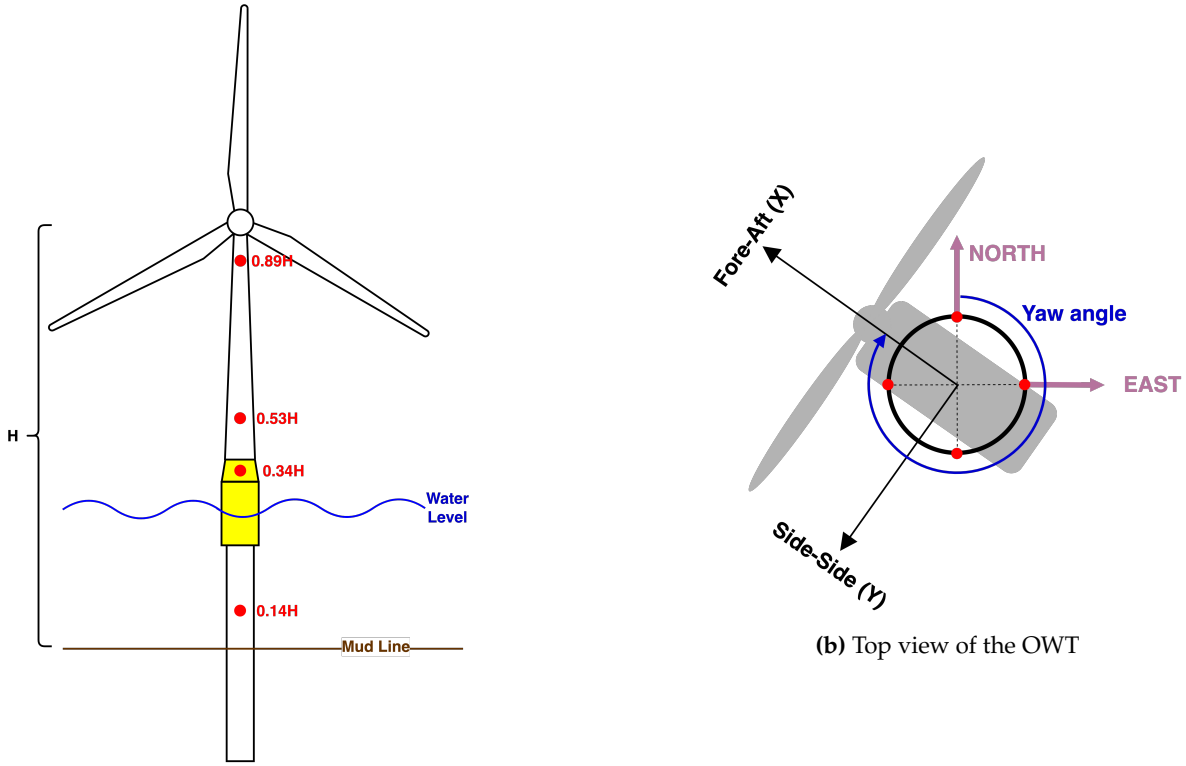
Table 2. Summary of measurement channels installed on the 6 MW offshore wind turbine.

Measurement Type	Details	Sampling Frequency
Accelerometers	12 sensors: 3 at 4 distinct elevations	25 Hz
Strain gauges	16 sensors: 4 at 4 distinct elevations	25 Hz
SCADA measurements	Wind speed (m/s), power output (kW), rotor speed (rpm), pitch angle (deg), yaw angle (deg)	10 Hz

The recorded strain gauge data have undergone calibration and are subsequently processed to derive bending moment components along two perpendicular global coordinates (aligned with North and East directions). Afterwards, by utilizing the yaw angle readings from SCADA the bending moments in global reference coordinates (North and East) transformed into the local coordinate system of the turbine, Fore-Aft (FA) and Side-Side (SS). Figure 2 provides a visual depiction of the strain gauges configuration employed in this study.

Measurements were recorded at 10-minute windows and filtered to ensure the turbine was operating under normal conditions. All measurements with a 10-minute average generated power below $100\,kW$ were excluded from the analysis to reduce the chance of having any non-operational data.

 doi:10.20944/preprints202604.2052.v1



(a) Strain gauges location relative to H

(b) Top view of the OWT

Figure 2. Strain gauges location along the OWT support structure, and the top view of the OWT with the strain gauge locations on the circumference of the support structure. H is the hub distance from the mudline.

After applying these filters, 1443 complete 10-minute measurement sets remained, each containing all required data fields.

In this study, we consider the strains and moments at 0.53 H as the quantity of interest and focus only on the location measurements and simulation results The same dataset is utilized in the work of Moynihan et al. for the 6MW turbine system identification and Mehrjoo et al. for wind load estimation on an OWT.

4. Results and Discussion

In Section 2 we explained the framework that we utilized in this work. In the results section, we start by explaining the specific tools that we used for this work considering the limitations and constraints we had. Later, we show how the simulation results compare with the measurements. Next, we describe the regression process we used and present the results.

4.1. WISDEM Framework for Optimizing the Blade Profile

Through an optimization procedure, we developed a blade that generates the same power, thrust and torque as the 6MW turbine for which we have access to the data. For the optimization of the blade, we employed Wind-Plant Integrated System Design & Engineering Model (WISDEM) developed by NLR [42]. WISDEM provides a set of integrated modules designed to evaluate the overall behavior of wind turbines and wind plants. In this framework, any parameter can be configured as a design variable, objective function, or constraint within a multidisciplinary optimization process. WISDEM has different submodules that they are used to optimized different part of a wind turbine. As we are dealing with the blade we only used RotorSE submodule for optimizing the blade [43]. RotorSE represents the comprehensive rotor analysis module within WISDEM, designed to model horizontal axis wind turbine rotors using steady-state computational approaches. The aerodynamic performance is calculated through blade element momentum theory, while the structural characteristics of composite rotor blades are determined using a specialized crossectional analysis tool. Structural deformations

 doi:10.20944/preprints202604.2052.v1

and load responses are computed with employing Timoshenko beam theory for accurate deflection predictions. Beyond the core aerodynamic and structural analyses, RotorSE incorporates turbine control system modeling through regulation trajectory implementation, enabling comprehensive annual energy production calculations. The framework also provides critical design assessment capabilities, including ultimate load predictions, blade tip deflection analysis, and strain distribution evaluation throughout the rotor structure. To support complete rotor design optimization, RotorSE features an integrated economic analysis component through its detailed Blade Cost Model, which estimates manufacturing costs based on blade geometry, materials, and manufacturing complexity. This multidisciplinary approach enabled us simultaneously optimize rotor performance, structural integrity, and economic viability within a unified design environment. Further description of the WISDEM and RotorSE is available at [42,43]

For our case, WISDEM is used as the framework that connects the optimizer to the wind turbine blade design. For the optimizer, we opted for `OpenMDAO` toolbox in `python` [44]. As the initial design, we considered the IEA-10MW with 198m rotor diameter [45] because the Cp of this turbine at rated wind speed was the closest to the 6MW wind turbine for which we have access to the data. The design variables (DVs) for the optimization of the blade are 10 control points for the chord distribution, 10 for the twist distribution, and 58 for the internal structural layup (including spar caps, shear webs, shell layers, leading edge and trailing edge reinforcement thicknesses at multiple spanwise stations). The chord and twist are parameterized using 10 control points each, distributed along the blade span, with intermediate values obtained through spline interpolation. This resolution was selected because fewer control points (e.g., 5) produced insufficient flexibility to match the target aerodynamic coefficients across the full operating range, while more control points (e.g., 15) increased the dimensionality of the search space without meaningful improvement in the objective function, as verified through preliminary optimization runs. The 58 internal DVs follow directly from the RotorSE module in WISDEM, which parameterizes the composite layup at multiple spanwise stations for each structural component; these variables are necessary to satisfy the blade mass and stiffness constraints (Table 1) while the chord and twist are adjusted to meet the aerodynamic targets. The total of 78 DVs represents the minimum set required to simultaneously satisfy all constraints in Table 1 while keeping the DE optimization computationally tractable. However, there are two important constraints that we added to the constraints in the table which come from our expert knowledge. First, the slope of the blade chord is always negative after it reaches its maximum in a wind turbine blade. Second, avoiding stall, by defining a 5° stall margin, in the operational condition is also added to the constraints list. We selected DE as the optimization algorithm in this work. We ended up with this decision after trying different gradient-based optimization algorithms without success. The gradient-based algorithms did not work as in our tests they all got stuck in unfeasible local minima. This is expected as we have a large number of DVs and an initial design which is relatively far from the optimum design. Although DE optimization algorithm provides the chance of reaching global optima, it comes with its own drawbacks. As DE starts with generating a large sample size and tests all of them for the feasibility of the design, it is computationally expensive. To reduce the computational burden, we divided the optimization into three stages of exploration, refinement and exploitation [46–48]. The summary of parameter settings for each phase is presented in Table 3. The complete three-phase optimization process required approximately 6 days of computational time, with the exploration phase taking 3 days, refinement 2 days, and exploitation 1 day.

Table 3. DE phases parameter settings for the blade optimization

Phase	Generations	population size	Differential weight (F)	Crossover probability (CF)
Exploration	1–900	30	0.7	0.3
Refinement	901–2100	20	0.5	0.5
Exploitation	2101–3000	15	0.3	0.7

 doi:10.20944/preprints202604.2052.v1

The three-phase adaptive strategy is used to balance exploration and exploitation throughout optimization. The approach begins with a large population (30) and high differential weight F (0.7) but low crossover CR (0.3) to broadly explore the search space through large moves. As promising regions are identified, the algorithm transitions to a refinement phase with reduced population (20) and balanced parameters (F=0.5, CR=0.5) to investigate these areas more thoroughly. Finally, the exploitation phase employs a small population (15) with low F (0.3) for precise movements and high CR (0.7) for multi-dimensional fine-tuning to converge on optimal solutions. These adaptive and phase base strategy is used as the fixed parameter could not handle the needs the optimizations. By utilizing the phased based approach, the computational resources could have been used more efficiently as well. The output of the optimizer after the three stages is chord and twist of the optimized blade.

The optimized blade chord and twist along the blade length is presented in Figure 3. The optimized chord and twist distributions exhibit unconventional characteristics compared to typical wind turbine blades, which is an expected outcome given the constraints of this study. The optimizer was initialized with a fixed set of publicly available airfoils originally selected for the IEA-10MW reference turbine, which features a 198m rotor diameter and generates 10MW at rated wind speed. In contrast, the target 6MW turbine employs proprietary, specifically optimized airfoils arranged in a rotor with diameter D. Since the available IEA-10MW airfoils were not designed for the 6MW turbine configuration, the optimization process produced unconventional blade geometries to match the target power, thrust, and torque characteristics. While the resulting blade design may not be manufacturable using current production methods and would be unsuitable for structural or dynamic analyses of the blades, it successfully replicates the aerodynamic performance metrics required for this study. This aerodynamic fidelity is sufficient for the objectives of this piece of work, despite the geometric differences from conventional blade designs.

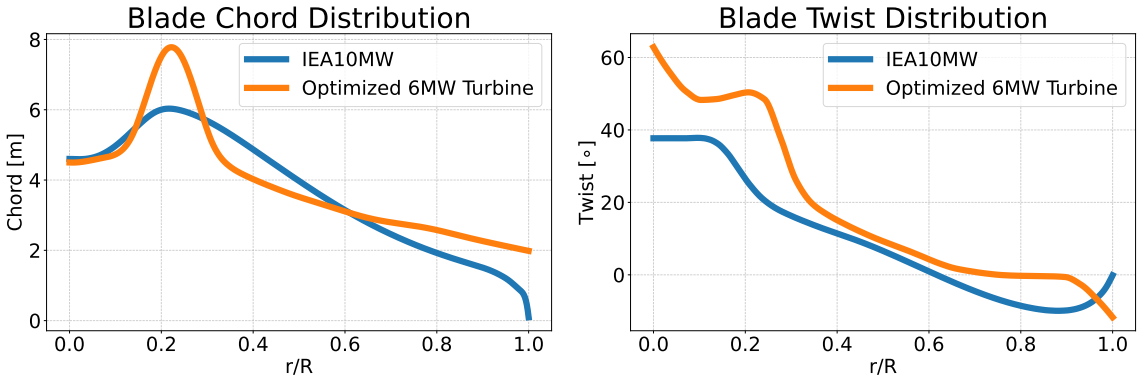


Figure 3. Optimization output vs the initial design for the chord and twist of the blade

Converting the optimized blade chord and twist into a part of an aero-hydro-servo-elastic model is not straightforward. Multiphysics models are often complex and have specific input requirements that vary by software platform. To bridge this gap, we used Wind Energy with Integrated Servo-Control (WEIS) toolbox to build the aerodynamic part of the aero-hydro-servo-elastic model [42]. WEIS is an open-source framework from NLR that brings together multiple wind turbine design tools for coupled optimization of turbine components and control systems. It handles multidisciplinary design optimization for both fixed-bottom and floating offshore wind turbines by combining aerodynamic, structural, and control modeling. In our case, WEIS takes the optimized blade geometry and converts it into the aerodynamic component of our aero-hydro-servo-elastic model, which can then be used for analysis.

4.2. Aero-Hydro-Servo-Elastic Model and Simulator

For this study, we built the Aero-hydro-servo-elastic model in OpenFAST [49]. OpenFAST is an open-source, physics-based engineering tool developed by the NLR for simulating the coupled

 doi:10.20944/preprints202604.2052.v1

dynamic response of wind turbines. The software integrates aerodynamic, hydrodynamic, control system, and structural dynamics modules to provide aero-hydro-servo-elastic modeling capabilities for both land-based and offshore wind turbine systems. OpenFAST comprises several interconnected modules that work together to capture the complex physics of wind turbine operation. AeroDyn computes aerodynamic loads on the blades using blade element momentum theory. ElastoDyn serves as the primary structural dynamics module, modeling the flexible behavior of blades, tower, and drivetrain components through modal or multibody approaches. For offshore applications, HydroDyn calculates hydrodynamic loading from waves and currents on the foundation. SubDyn provides a finite-element-based structural dynamics solution for fixed-bottom support structures. The ServoDyn and ROSCO [50] module implements turbine control systems, including blade pitch, generator torque, and yaw control algorithms. TurbSim [51] is a stochastic turbulence simulator that serves as a preprocessor tool for generating realistic turbulent wind inflow conditions used in OpenFAST simulations. These modules communicate through a "glue code" that allows for coupled time-domain simulations, enabling the wind energy community to analyze wind turbine performance, loads, and stability under various operating conditions.

To build the aero-hydro-servo-elasto model in OpenFAST, all the submodules are required:

- The rotor aerodynamics in AeroDyn is modeled by taking the direct output of WEIS. This assures us that we have the optimized blade aerodynamics in the OpenFAST model.
- The support structure is modeled as an integrated piece and in SubDyn. Modeling the entire support structure (tower, transition piece and foundation) in SubDyn is recommended because it ensures consistent finite element formulation throughout the structure, accurately captures coupled tower-foundation dynamics [52,53]. In SubDyn, the finite element model of the support structure consists of 77 Timoshenko beam elements. We model the support structure in SubDyn using apparent fixity depth (provided by the developer), as we did not have access to soil stiffness information. The apparent fixity method models the monopile-soil system by extending the pile to a fictitious depth below the mudline where it is assumed to be rigidly fixed, effectively replacing the complex soil-structure interaction with an equivalent cantilever beam [54,55]. We model the substructure in SubDyn using Craig-Bramton reduction, utilizing first 10 modes. The structural damping is set to 1% critical damping for the first mode as stated by the developer. Before running the OpenFAST simulation, one of the important checks is how close is the first FA and SS natural frequencies to the measurement. Table 4 shows the comparison between the first FA and SS natural frequencies of the OpenFAST model and the measurement [40]. The OpenFAST model and measurement first natural frequencies are in good agreement.

Table 4. FA and SS first natural frequencies comparison.

	OpenFAST [Hz]	Measurement [Hz]	Difference [%]
FA	0.224	0.230	2.6
SS	0.225	0.235	4.4

- As we model the support structure fully in SubDyn, ElastoDyn covers the blade model. The structural properties of the blade are also an output of WEIS. The dimensions and weights of the OWT components are set in ElastoDyn as well. This includes RNA center of mass and moment of inertia, hub mass and moment of inertia, and the hub height. These data were provided to us by the developer.
- For the hydrodynamic part, we did not have access to wave measurements. However, the developer provided us with a limited number of wave elevation time series for each wind speed. We utilized those wave elevation time series as the input to HydroDyn to calculate the hydrodynamic loads on the support structure. Also, we did not have access to the hydrodynamic coefficients of the monopile. Therefore, we kept them at the default values suggested in HydroDyn [56].

 doi:10.20944/preprints202604.2052.v1

- The controller settings tuning was performed in ROSCO. We tuned the controller settings to follow the turbine behavior in terms of power generation and rotor speed as closely as possible. In other words, we used ROSCO to tune the gains in the controller in a manner that the turbine follows specific power and rotor speed curves that we extracted from the SCADA data. For below-rated wind speeds, the torque control is active, while for above-rated wind speeds the pitch controller is active. After tuning, the ROSCO output is a controller settings file which is the input to ServoDyn. ROSCO tunes the controller gains based on the aerodynamic coefficients of the rotor.
- To run the aero-hydro-servo-elastic simulations we need two environmental inputs: time histories of wind speed and wave height. For OpenFAST, the turbulent wind time series is generated in TurbSim. To mimic the conditions under which the 6MW turbine operated, we generated the TurbSim output based on the wind time series readings from SCADA. For every 10-minute measurement, we calculated the mean and standard deviation. Then we calculated the turbulence intensity (TI) for the 10 minutes:

$$TI = \frac{\sigma_{wind}}{\bar{U}} \tag{11}$$

where σ_{wind} is the standard deviation of the wind time series, and $\bar{U}$ is the mean wind speed for that 10-minute measurement. For every TurbSim output we generated 36 unique realizations (seeds). Also, we set the wind shear $\alpha = 0.11$, and the grid height and width for the generated turbulent wind field to 200m by 200m divided into a 21 by 21 grid. Each generated turbulent wind time series is 800 seconds long.

To this point, we have set up the model in OpenFAST. We visualize the process explained above in Figure 4. As mentioned, we generated 36 realizations (seeds) per measurement. The OpenFAST simulations are set to have a 25 Hz sampling frequency, which is the same as the measurement. The OpenFAST simulations are set to have 800s length, where the first 200s is not recorded as it is considered the initialization phase. The turbulent wind time series was also set to 800s to cover the 200s initialization. We have 1,443 ten-minute measurement data sets, and by running 36 realizations per measurement we end up with 51,948 ten-minute OpenFAST simulated time series. Each OpenFAST simulation requires approximately 2 hours of computational time, resulting in 72 hours per measurement case. The simulations are run for normal operation conditions, DLC1.2, and zero wind-wave misalignment (WWMA) according to IEC standards [18]. We ran the simulations in 1,443 batches of 36 simulations on Tufts University HPC, leveraging parallel computing to reduce the total wall-clock time.

4.3. OpenFAST Simulation Results

To this point, we build a database of 51,948 OpenFAST simulations. This includes 36 realizations (seeds) of the wind statistics for each of 1443 measurements. After running the simulations the initial step is checking the power output and rotor speed simulation outputs against measurement (Figure 5).

In Figure 5, every data point is the 10-minutes average from simulations or the measurements. The figure illustrates that the generated power output of the simulation aligns closely with the measurements. The rotor speed output follows the pattern, however it has a wider range than the measurements. For the pitch angle, the simulations follow the measurement above the rated wind speed (when pitch controller is active) very well. The reason behind rotor speed wide range, and pitch angle below rated not matching, rests in the optimized blade design. It seems the large chord can produce more torque than the 6MW turbine at the wind speeds below rated wind speed, therefore for the rotor speeds less than the measurement it can generate the same power. After the rated wind speed, when the pitch controller kicks in, the behavior is closer to the measurement, as the larger chord helps to compensate for the aerodynamic discrepancy.

 doi:10.20944/preprints202604.2052.v1

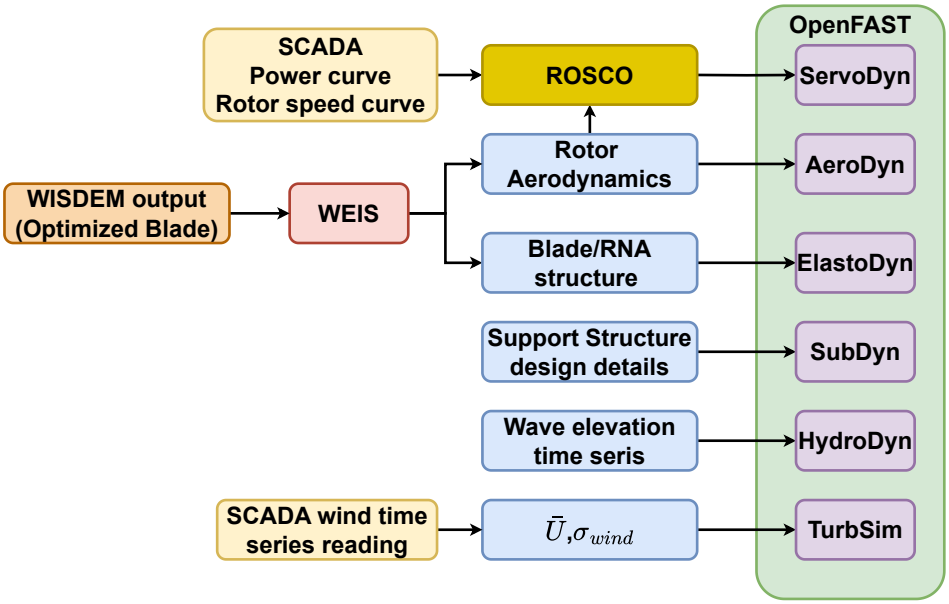


Figure 4. Schematic of the OpenFAST model assembly process. The optimized blade from WISDEM is converted through WEIS into rotor aerodynamics (AeroDyn) and blade structural properties (ElastoDyn). The support structure design is modeled in SubDyn, wave elevation time series are input to HydroDyn, and SCADA-derived wind statistics feed TurbSim for synthetic turbulent wind generation. The controller (ServoDyn) is tuned via ROSCO using the measured power and rotor speed curves. Arrows indicate data flow between modules.

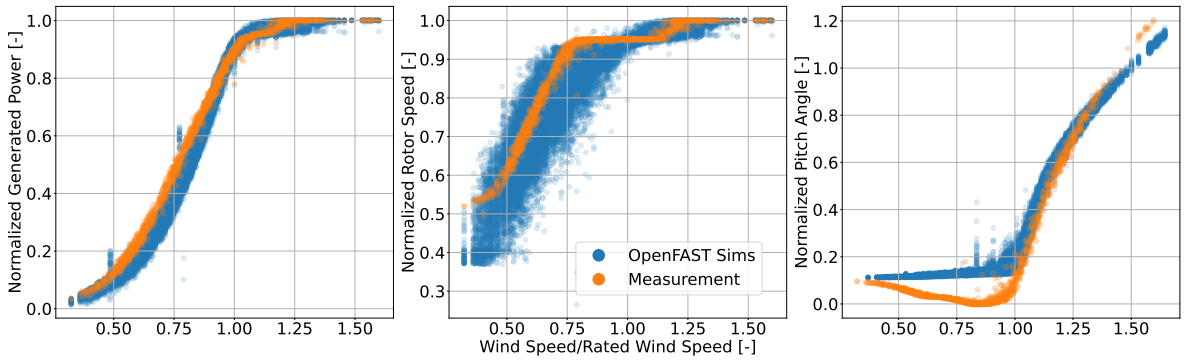


Figure 5. Comparison of SCADA-measured and simulated generated power, rotor rotational speed and pitch angles. Each datapoint presents 10-minutes average of a simulation or measurement. The x- and y-axes are normalized.

As mentioned before, for this study, we consider the tower moments in FA and SS at height 0.53H as a quantity of interest. Figure 6 shows how the measured moments compared with the simulations. Each data point on the plots is the 10-minutes average of the measurement or simulations.

 doi:10.20944/preprints202604.2052.v1

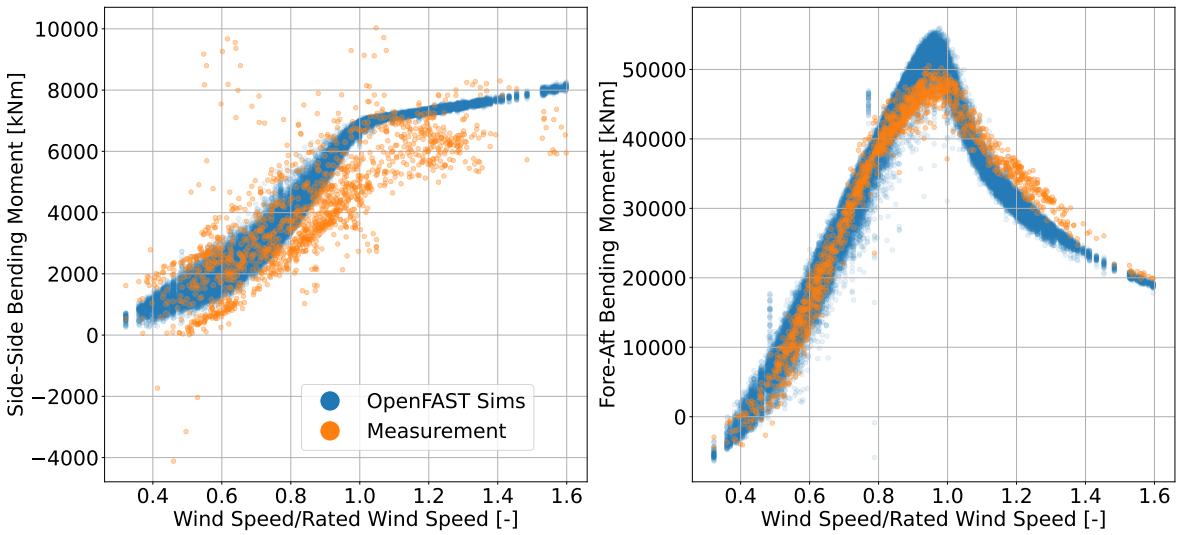


Figure 6. SS and FA bending moments at 0.53H location along the support structure. Each datapoint presents 10-minutes average of a simulation or measurement.

In Figure 6, the SS moment simulations follow the expected pattern; however, the measured moments in the SS direction show considerable spread. There is a weak pattern for the measurements below rated wind speed, but for wind speeds above rated, this pattern disappears. The SS moment along the tower is dominantly from the rotor torque, where the simulation and measurement match reasonably well in terms of mean trends. We speculate this spread may be caused by factors not captured in our model, such as wind-wave misalignment, yaw misalignment, or potential issues in the measurement system. On the other hand, the FA moments match the measurements closely. This shows the optimized rotor generates similar thrust to the 6MW turbine, especially before the rated wind speed. At the rated wind speed, we are overestimating the FA moment, while after that, we are slightly underestimating the FA moment. This happens due to the large chord of the optimized blade. At the rated wind speed, when the pitch angle is zero, the thrust is maximum, and as we start to pitch out the blade, due to the chord length, the blade design reduces C_t faster than the 6MW blade. In other words, we have a blade which mimics the aerodynamic coefficients of the 6MW turbine, but when it is in operation, the details of the design affect its behavior.

Heretofore, we have built a database that consists of 51,948 simulations representing different EOCs of the wind turbine. In this database, for each measurement we have 36 counterpart simulations. The next steps are calculating the absolute error between the simulations and measurements for both the 10-minute average and 10-minute standard deviation. As explained in Section 2, this is done by simply finding the difference between the values (Eq. (7)). Figure 7 presents the error $\boldsymbol{E}$ between measured and simulated 10-minute mean bending moments across the full dataset. The FA errors exhibit a clear wind-speed-dependent structure, with small errors below rated, positive bias near rated, and negative bias above rated—consistent with the thrust behavior discussed in the context of Figure 6. The SS errors are more symmetric around zero but with substantially larger scatter, reflecting the unmodeled factors identified earlier. Notably, visualizing the data as errors rather than raw moments motivates the regression-based error modeling in Section 4.4.

As explained in Section 2, one may consider the error per measurement a random variable, where the mean and standard deviation of those come from the 36 errors per measurement. In some cases [26], it is reported that the modeling error follows a normal distribution as in Equation (8). To check this assumption, we employed quantile-quantile (Q-Q) plots, which provide a graphical method for assessing whether data follows a specific theoretical distribution [57]. A Q-Q plot compares the quantiles of the observed data against the quantiles of the theoretical distribution (in our case, normal distribution). When the data follows the assumed distribution, the points in the Q-Q plot align approximately along a straight diagonal line. Systematic deviations from this line indicate departures

 doi:10.20944/preprints202604.2052.v1

from normality: S-shaped patterns suggest heavier or lighter tails than the normal distribution, while outliers appear as points that deviate significantly from the line at the extremes. Figure 8 shows representative Q-Q plots for the error distributions at different wind speeds, confirming that the normality assumption is reasonable for most operating conditions.

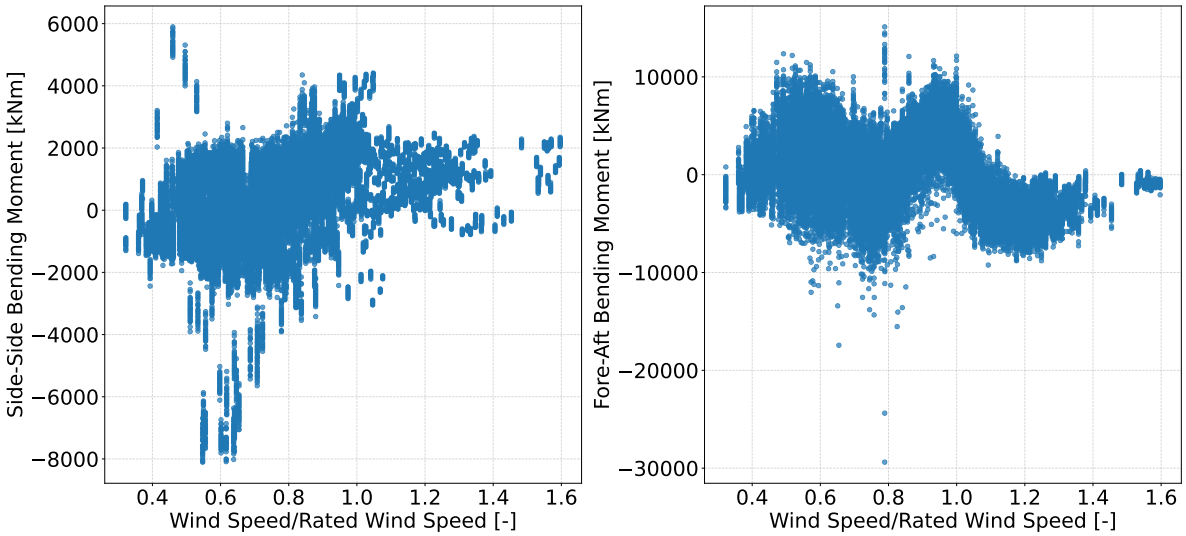


Figure 7. Error E between measured and simulated 10-minute mean bending moments in SS and FA directions as a function of normalized wind speed. Each data point represents the difference between a measured 10-minute mean and its corresponding simulation, totaling 51,948 data points (36 realizations $\times$ 1,443 measurements).

This observed normality emerges naturally from aggregating multiple independent stochastic sources and was verified after ruling out systematic biases. While normal distributions adequately describe our errors, the framework generalizes to any parametric distribution, ensuring broad applicability regardless of the error distribution form.

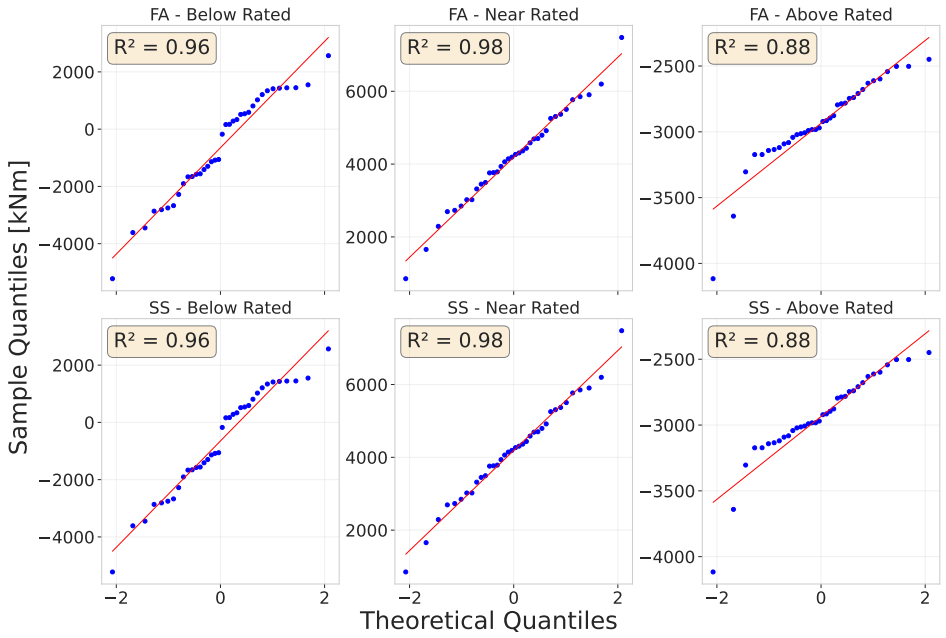


Figure 8. Q-Q plots for error distributions at representative wind speeds. SS E and FA E at below-rated wind speed, near-rated wind speed, and above-rated wind speed. The diagonal line represents a perfect normal distribution. Points following this line indicate that the error distributions are approximately normal.

 doi:10.20944/preprints202604.2052.v1

4.4. Error Modeling

For the error modeling, our aim is to predict the mean and standard deviation of the error at different EOC. Table 5 shows the ANN architecture and training details. The ANN architecture and its hyperparameters were determined experimentally. The database size for training is the same as the number of measurement data points (1,443 data points). We use 80% of the database for training and 20% for testing. Figure 9 presents the architecture of the employed ANN. We developed, trained and tested the ANNs in `PyTorch` [58]. The training process was computationally efficient, requiring approximately 3 minutes per ANN model on an NVIDIA A100 GPU, resulting in a total training time of less than 30 minutes for all four regression models.

The choice of ANNs over simpler regression methods was motivated by the expected non-linear relationships between EOCs and simulation errors. Initial tests with linear regression showed significantly lower R^2 values (not shown), justifying the use of more complex models.

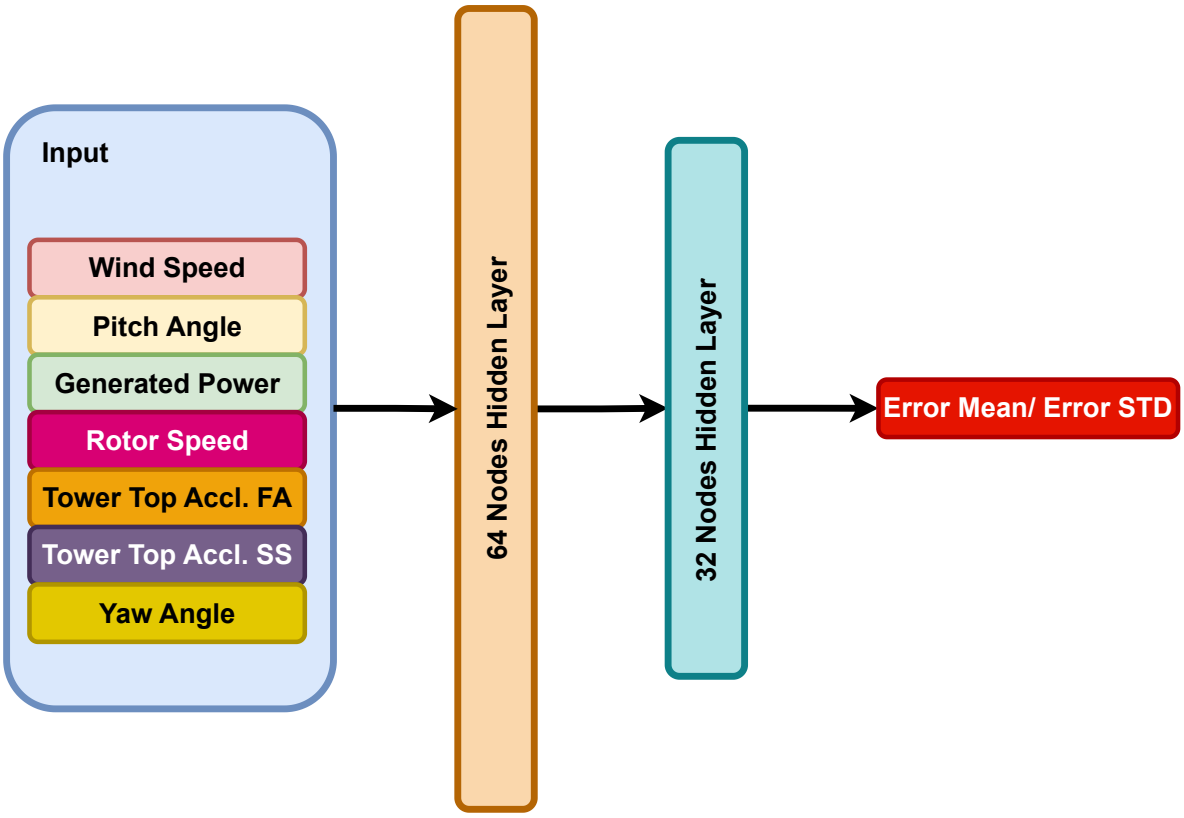


Figure 9. Architecture of the artificial neural network used for error regression. The network takes seven environmental and operational condition inputs and outputs one of the four error distribution parameters.

Table 5. The utilized ANN details.

Number of layers	2
Neurons per layer	64, 32
Batch Size	64
Optimizer	Adam
Epochs patience	100
Learning rate	0.001
Drop out rate	0.2

The aim of the regression model here is to be able to predict the mean and standard deviation of the error. As we are taking into account two outputs, bending moments in FA and SS directions, we need four regression models. For brevity, we denote these regression models as μ_{SS}, σ_{SS}, μ_{FA}, and σ_{FA}, corresponding to the mean and standard deviation of error mean in the SS and FA directions,

 doi:10.20944/preprints202604.2052.v1

respectively. The testing results for the aforementioned ANN model, for the four ANN models with feature importance analysis utilizing the permutation importance method [39] are presented in Figure 10.

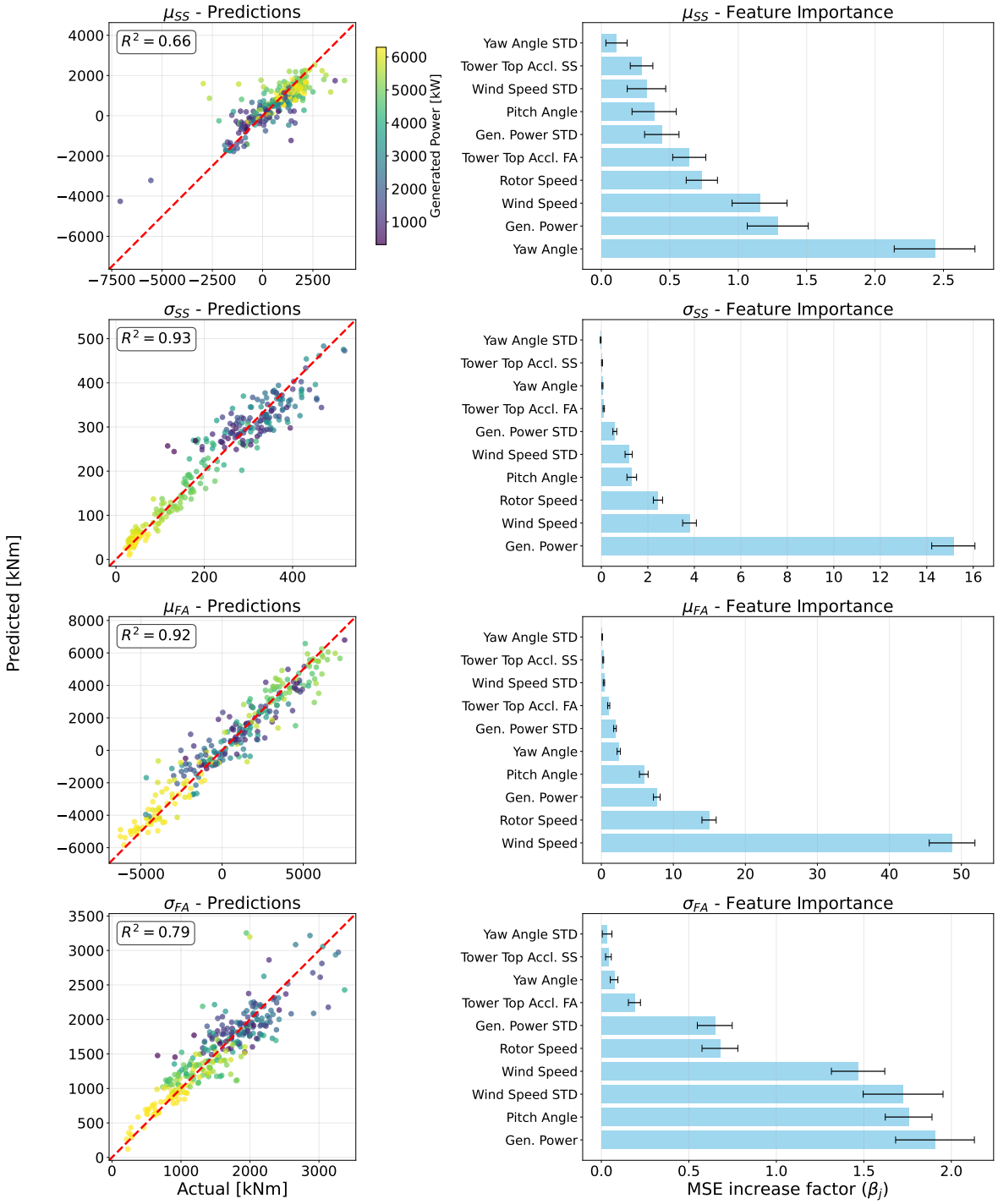


Figure 10. ANN regression results and permutation feature importance analysis for the four error models: μ_{SS}, σ_{SS}, μ_{FA}, and σ_{FA}. Left panels show predicted versus actual error values on the test dataset (20% of 1,443 data points), colored by generated power, with R^2 values indicated. Right panels show the MSE increase factor β_j for each input feature, with error bars representing the standard deviation across 100 permutation repetitions. Higher β_j values indicate greater feature influence on model predictions.

The permutation importance analysis reveals distinct predictive patterns across the four ANN models, with R^2 values ranging from 0.65 to 0.93. The μ_{SS} and μ_{FA} demonstrate direction-specific feature dependencies, where μ_{SS} dominated by yaw angle, while μ_{FA} primarily driven by wind speed.

 doi:10.20944/preprints202604.2052.v1

For σ_{SS} and σ_{FA}, the models are more sensitive to the inputs from the wind turbine, as σ_{SS} is most sensitive to generated power, whereas σ_{FA} is predominantly influenced by pitch angle. The consistently moderate importance of rotor speed across all models suggests its role as a fundamental coupling parameter between environmental and operational conditions.

The four ANN regression models demonstrate varying degrees of predictive accuracy across different error metrics. The models achieved R^2 values of 0.65 for μ_{SS}, 0.93 for σ_{SS}, 0.90 for μ_{FA}, and 0.81 for σ_{FA} on the test dataset. These results indicate that the ANNs successfully capture the complex relationships between EOCs and simulation errors, with particularly strong performance for the FA direction mean errors and SS standard deviation predictions. The permutation importance analysis provides crucial insights into which EOCs most strongly influence model predictions. The error bars in the feature importance plots represent the standard deviation across multiple permutation repetitions (100 times for this study), providing confidence in the importance rankings. For μ_{SS}, yaw angle emerges as the dominant feature with an importance score exceeding 2.0, suggesting that yaw misalignment or yaw dynamics significantly affect side-side loading predictions. In contrast, μ_{FA} shows the highest sensitivity to wind speed (importance $\sim$50), which directly drives the thrust forces on the rotor. The σ_{SS} and σ_{FA} models reveal similar dependencies: σ_{SS} and σ_{FA} are primarily influenced by generated power, while the influence of generated power on σ_{SS} is significantly larger. These patterns align with the physical understanding of wind turbine dynamics, where thrust-related quantities (FA) correlate with environmental conditions, while torque-related quantities (SS) correlate with operational parameters.

The trained ANN models enable practical uncertainty quantification for OpenFAST simulations by predicting the error distribution parameters μ_{FA}, σ_{FA}, μ_{SS}, and σ_{SS} for any given EOC. When conducting future simulations, these models can be used to establish confidence bounds on the predictions. Specifically, for a simulation with known EOCs, the ANNs predict μ_{FA} and μ_{SS}, which represent the expected bias between simulation and reality, and σ_{FA} and σ_{SS}, which quantify the variability of this bias across EOCs. This approach transforms deterministic OpenFAST outputs into probabilistic predictions that can account for the model uncertainty.

The superior predictability of FA errors compared to SS errors (R^2 of 0.92 vs. 0.66 for μ_{FA} and μ_{SS}) can be attributed to several factors. First, FA loads are primarily driven by wind thrust forces, which our optimized blade design was specifically calibrated to match. The optimization process ensured that the rotor produces the same thrust-wind speed relationship as the actual turbine, leading to accurate FA load predictions. In contrast, SS loads are influenced by multiple factors not fully captured in our modeling framework. The assumption of zero wind-wave misalignment has a significant impact on SS dynamics, as wave-induced loads can excite SS vibrations when waves approach from oblique angles. Additionally, the low importance scores for most EOCs in the μ_{SS} model suggest that unmeasured or unmodeled factors, such as actual wind-wave misalignment angles or yaw control dynamics, may be driving the SS loading variability. The relatively high importance of yaw angle for μ_{SS} supports this hypothesis, as yaw misalignment directly affects the lateral loading on the rotor.

5. Conclusions

This study presented a comprehensive framework for developing and calibrating aero-servo-hydro-elastic models of offshore wind turbines when proprietary blade and controller design information are unavailable and limited. The methodology successfully addresses a critical gap in the wind energy research community by demonstrating that accurate operational models can be developed using only publicly available reference designs and operational measurements.

The key contribution of this work are: a) the development of an inverse-design optimization approach using differential evolution to reconstruct blade aerodynamic characteristics from operational data, without any knowledge about the aerodynamic and elastic properties of the rotor b) quantification of modeling errors at different EOC, and c) application of teh proposed approach on an operational 6MW OWT using in-situ measurements. The methodology enabled the creation of a complete aero-

 doi:10.20944/preprints202604.2052.v1

servo-hydro-elastic model in OpenFAST that accurately reproduces the dynamic behavior of a 6MW OWT. Furthermore, the implementation of a comprehensive uncertainty quantification framework using ANN successfully predicts simulation errors based on environmental and operational conditions, achieving R^2 values between 0.66 and 0.93 for different error metrics. The analysis revealed that yaw angle and wind speed are the primary drivers of μ_{SS} and μ_{FA}, providing valuable insights for future model improvements. The results indicate that while the optimized blade design may not physically represent the actual turbine blade, it successfully replicates the aerodynamic performance necessary for accurate support structure load predictions. The methodology can be effectively applied to various turbines, provided there is access to operational measurements, an understanding of support structure design, careful selection of a suitable reference turbine for initial design, and tailored re-optimization and validation for each specific turbine.

While this study demonstrates a practical framework for developing wind turbine models without proprietary design data, several limitations should be acknowledged. The optimized blade design, while aerodynamically equivalent, represents a non-physical geometry that cannot be manufactured using current production methods. The unconventional chord and twist distributions result from optimizing publicly available airfoils to match the proprietary turbine performance, making this approach unsuitable for blade-specific structural analyses, while we show the strength of this method to predict the bending moments along the support structure. The absence of direct wave measurements constrains the model's ability to capture side-side loading accurately, as we relied on limited wave elevation time series and assumed zero wind-wave misalignment throughout the simulations. This simplification likely contributes to the lower prediction accuracy for SS moments compared to FA moments. Another potential source of discrepancy between OpenFAST model predictions and measured data is the lack of information about active damping strategies in the proprietary controller. This could explain discrepancies in the predictions, as modern wind turbines often employ sophisticated damping controllers that are not captured in our simplified ROSCO implementation.

The framework also requires significant computational resources, with the blade optimization process alone requiring 6 days of computational time and the generation of 51,948 ten-minute simulations requiring approximately 104,000 CPU hours. The choice of 36 seeds per measurement represents a balance between statistical convergence and computational feasibility, though more seeds would provide better statistical estimates at the cost of increased computation time.

Future work should focus on incorporating wind-wave misalignment effects through either direct measurements or statistical models to improve SS moment predictions. Investigation of advanced controller features, particularly active damping strategies, would help better capture dynamic response characteristics. The methodology could be extended to floating offshore wind turbines where platform dynamics introduce additional complexity.

This framework provides a practical pathway for developing validated numerical models of operational offshore wind turbines, enabling improved structural health monitoring, remaining useful life assessments, and optimization of operations and maintenance strategies. As the offshore wind industry continues to expand, such approaches will become increasingly valuable for managing assets where complete design documentation is unavailable.

Data Availability Statement: All the data used in these studies, including measurement and simulation input files and results, are proprietary and cannot be shared with the public. The goal of this paper is not to share data from a utility-scale operational offshore wind turbine but to present a framework that demonstrates how a functioning wind turbine model can be achieved if the data were available.

Acknowledgments: Partial support of this study was provided by the National Science Foundation grant 2230630 and Tufts Institute for Artificial Intelligence (TIAI). The authors would like to express their gratitude to Finn Rüdinger and Ross McAdam for providing the data and their technical support. Additionally, we would like to thank Jason Jonkman and Pietro Bortolotti from NLR for introducing WISDEM and WEIS to us. The authors acknowledge the Tufts University High Performance Compute Cluster (https://it.tufts.edu/high-performance-computing), which was utilized for the research reported in this manuscript.

 doi:10.20944/preprints202604.2052.v1

References

1. F. Driscoll, J. Jonkman, A. Robertson, S. Sirnivas, B. Skaare, F. G. Nielsen, Validation of the FAST model of the Statoil-Hywind demo floating wind turbine, in: Proceedings of the Deep Wind Conference 2016 (published in Energy Procedia), volume 94, 2016, pp. 3–19. doi:10.1016/j.egypro.2016.09.181.
2. K. Brown, P. Bortolotti, E. Branlard, M. Chetan, S. Dana, N. deVelder, P. Doubrawa, N. Hamilton, H. Ivanov, J. Jonkman, C. Kelley, D. Zalkind, One-to-one aeroservoelastic validation of operational loads and performance of a 2.8 mw wind turbine model in openfast, Wind Energy Science 9 (2024) 1791–1810. doi:10.5194/wes-9-1791-2024.
3. S. Guntur, J. Jonkman, R. Sievers, M. A. Sprague, S. Schreck, Q. Wang, A validation and code-to-code verification of FAST for a megawatt-scale wind turbine with aeroelastically tailored blades, Wind Energy Science 2 (2017) 443–468. doi:10.5194/wes-2-443-2017.
4. M. Kretschmer, J. Jonkman, V. Pettas, P. W. Cheng, Fast.farm load validation for single wake situations at alpha ventus, Wind Energy Science 6 (2021) 1247–1262. doi:10.5194/wes-6-1247-2021.
5. J. Jonkman, A. Robertson, Offshore Code Comparison Collaboration Continuation (OC4), Phase I - Results of Coupled Simulations of an Offshore Wind Turbine with Jacket Support Structure, 2012. URL: https://docs.nrel.gov/docs/fy12osti/54124.pdf.
6. D. Kaufer, P. W. Cheng, Validation of an integrated simulation method with high-resolution load measurements of the offshore wind turbine REpower 5m at alpha ventus, Journal of Ocean and Wind Energy 1 (2014) 30–40.
7. B. Sanderse, V. V. Dighe, K. Boorsma, G. Schepers, Efficient Bayesian calibration of aerodynamic wind turbine models using surrogate modeling, Wind Energy Science 7 (2022) 759–781. doi:10.5194/wes-7-759-2022.
8. G. M. Stewart, M. A. Lackner, A. Robertson, J. M. Jonkman, A. J. Goupee, Calibration and validation of a FAST floating wind turbine model of the DeepCwind scaled tension-leg platform, Technical Report NREL/CP-5000-54822, National Renewable Energy Laboratory, 2013. NREL Technical Report (preprint).
9. Y. Ma, et al., Blade inverse design through CFD and experimental torque data, 2023. Unpublished work.
10. K. Xu, et al., Structural model calibration using particle swarm optimization based on vibration data, 2023. Unpublished work.
11. P. Veers, C. L. Bottasso, L. Manuel, et al., Grand challenges in the design, manufacture, and operation of future wind turbine systems, Wind Energy Science 8 (2023) 1071–1131.
12. D. T. Griffith, et al., Large Offshore Rotor Development: Design and Analysis of the Sandia 100-meter Wind Turbine Blade, Technical Report, Sandia National Laboratory, 2012.
13. P. Sharma, B. Gupta, M. Pandey, A. K. Sharma, R. Nareliya Mishra, Recent advancements in optimization methods for wind turbine airfoil design: A review 47 (2021) 6556–6563. doi:10.1016/j.matpr.2021.02.231.
14. X. Yang, Engineering Optimization: An Introduction with Metaheuristic Applications, 1 ed., Wiley, 2010. doi:10.1002/9780470640425.
15. A. D. Belegundu, T. R. Chandrupatla, Optimization Concepts and Applications in Engineering, Cambridge University Press, 2019. doi:10.1017/9781108347976.
16. R. Storn, K. V. Price, Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces, Journal of Global Optimization 11 (1997) 341–359. doi:10.1023/A:1008202821328.
17. E. Bossanyi, T. Burton, D. Sharpe, N. Jenkins, Wind Energy Handbook, New York: Wiley, 2011.
18. IEC, Wind energy generation systems – part 3-1: Design requirements for fixed offshore wind turbines, 2019.
19. J. Jonkman, P. Sclavounos, Development of Fully Coupled Aeroelastic and Hydrodynamic Models for Offshore Wind Turbines, in: 44th AIAA Aerospace Sciences Meeting and Exhibit, American Institute of Aeronautics and Astronautics, 2012. doi:10.2514/6.2006-995.
20. A. D. Wright, L. J. Fingersh, Advanced Control Design for Wind Turbines; Part I: Control Design, Implementation, and Initial Tests, 2008. doi:10.2172/927269.
21. P. L. C. van der Valk, S. N. Voormeeren, P. C. de Valk, D. J. Rixen, Dynamic models for load calculation procedures of offshore wind turbine support structures: Overview, assessment, and outlook 10 (2015) 041013. doi:10.1115/1.4028136.
22. F. D. Bianchi, H. De Battista, R. J. Mantz, Wind turbine control systems: principles, modelling and gain scheduling design, Springer Science & Business Media, 2006.
23. B. Sanderse, et al., Surrogate models for the blade element momentum aerodynamic model using non-intrusive polynomial chaos expansions, Wind Energy Science Discussions 2021 (2021) 1–25. doi:10.5194/wes-2021-110.

 doi:10.20944/preprints202604.2052.v1

24. P. S. Veers, Three-dimensional wind simulation, Sandia National Labs., Albuquerque, NM (USA), 1988. URL: https://www.osti.gov/biblio/6633902.
25. E. Marino, A. Giusti, L. Manuel, Offshore wind turbine fatigue loads: The influence of alternative wave modeling for different turbulent and mean winds 102 (2017) 157–169. doi:10.1016/j.renene.2016.10.023.
26. M. H. Kutner, C. J. Nachtsheim, J. Neter, W. Li, Applied Linear Statistical Models, 5th ed., McGraw-Hill, 2005.
27. C. J. Roy, W. L. Oberkampf, A comprehensive framework for verification, validation, and uncertainty quantification in scientific computing, Computer Methods in Applied Mechanics and Engineering 200 (2011) 2131–2144. doi:10.1016/j.cma.2011.03.016.
28. M. C. Kennedy, A. O'Hagan, Bayesian calibration of computer models, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 63 (2001) 425–464. doi:10.1111/1467-9868.00294.
29. I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016. http://www.deeplearningbook.org.
30. G. Cybenko, Approximation by superpositions of a sigmoidal function, Mathematics of Control, Signals and Systems 2 (1989) 303–314. doi:10.1007/BF02551274.
31. K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural Networks 2 (1989) 359–366. doi:10.1016/0893-6080(89)90020-8.
32. S. Haykin, Neural Networks: A Comprehensive Foundation, 2nd ed., Prentice Hall, Upper Saddle River, NJ, 1999.
33. C. M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, Oxford, UK, 1995.
34. D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning representations by back-propagating errors, Nature 323 (1986) 533–536. doi:10.1038/323533a0.
35. J. Schmidhuber, Deep learning in neural networks: An overview, Neural Networks 61 (2015) 85–117. doi:10.1016/j.neunet.2014.09.003.
36. B. R. Deshpande, A. V. Karhade, C. E. Hwang, N. N. Trivedi, S. Gwilym, P. Jayakumar, Artificial neural networks outperform linear regression in estimating 9-month patient-reported outcomes after upper extremity fractures with increasing number of variables, OTA International 7 (2024) e284. doi:10.1097/OI9.0000000000000284.
37. M. West, J. Harrison, H. S. Migon, A comparison of linear regression, neural networks and non-parametric methods for forecasting: evidence from simulation, International Journal of Forecasting 13 (1997) 281–293.
38. D. Bansal, K. S. Evans, B. Jones, Comparison of the decision tree, artificial neural network, and linear regression methods based on the number and types of independent variables and sample size, Expert Systems with Applications 32 (2007) 1227–1234. doi:10.1016/j.eswa.2006.02.017.
39. A. Fisher, C. Rudin, F. Dominici, All Models are Wrong, but Many are Useful: Learning a Variable's Importance by Studying an Entire Class of Prediction Models Simultaneously 20 (2019) 1–81. URL: http://jmlr.org/papers/v20/18-760.html.
40. B. Moynihan, A. Mehrjoo, B. Moaveni, R. McAdam, F. Rüdinger, E. Hines, System identification and finite element model updating of a 6 MW offshore wind turbine using vibrational response measurements 219 (2023) 119430. doi:10.1016/j.renene.2023.119430.
41. A. Mehrjoo, E. M. Tronci, B. Moynihan, B. Moaveni, F. Rüdinger, R. McAdam, E. Hines, Recursive Bayesian estimation of wind load on a monopile-supported offshore wind turbine using output-only measurements 224 (2025) 112183. doi:10.1016/j.ymssp.2024.112183.
42. N. J. Abbas, G. Barter, P. Bortolotti, J. Jasa, R. M. Mudafort, J. Nunemaker, M. Chetan, E. Quon, D. S. Ramos, Y. Liao, C. Frontin, Y. H. Lee, A. K. Sundarrajan, E. Gaertner, S. Ryu, E. Branlard, S. Mulders, A. Gupta, A. Key, D. Heffernan, G. Motes, J. Rinker, P.-E. Meunier, X. Du, WISDEM/WEIS: V1.5.2 release, 2025. doi:10.5281/zenodo.15133488.
43. RotorSE — WISDEM 2.0 documentation, ???? URL: https://wisdem.readthedocs.io/en/master/wisdem/rotorse/index.html.
44. J. S. Gray, J. T. Hwang, J. R. R. A. Martins, K. T. Moore, B. A. Naylor, OpenMDAO: An open-source framework for multidisciplinary design, analysis, and optimization, Structural and Multidisciplinary Optimization 59 (2019) 1075–1104. doi:10.1007/s00158-019-02211-z.
45. P. Bortolotti, H. C. Tarres, K. Dykes, K. Merz, L. Sethuraman, D. Verelst, F. Zahle, IEA Wind Task 37 on Systems Engineering in Wind Energy – WP2.1 Reference Wind Turbines, Technical Report, International Energy Agency, 2019. URL: https://www.nrel.gov/docs/fy19osti/73492.pdf.
46. J. Brest, M. Sepesy Maučec, Population size reduction for the differential evolution algorithm 29 (2008) 228–247. doi:10.1007/s10489-007-0091-x.

 doi:10.20944/preprints202604.2052.v1

47. R. Poláková, J. Tvrdík, P. Bujok, Differential evolution with adaptive mechanism of population size according to current population diversity 50 (2019) 100519. doi:10.1016/j.swevo.2019.03.014.
48. S. X. Zhang, W. S. Chan, K. S. Tang, S. Y. Zheng, Adaptive strategy in differential evolution via explicit exploitation and exploration controls 107 (2021) 107494. doi:10.1016/j.asoc.2021.107494.
49. B. Jonkman, A. Platt, R. M. Mudafort, E. Branlard, L. Wang, D. Slaughter, M. Sprague, H. Ross, J. Jonkman, M. Chetan, R. Davies, M. Hall, G. Vijayakumar, M. Buhl, P. Bortolotti, R. Bergua, S. Ananthan, J. Rood, A. Gupta, F. H. Bhuiyan, A. Sharma, OpenFAST/openfast: V4.1.0, 2025. doi:10.5281/zenodo.15694513.
50. N. J. Abbas, D. S. Zalkind, L. Pao, A. Wright, A reference open-source controller for fixed and floating offshore wind turbines 7 (2022) 53–73. doi:10.5194/wes-7-53-2022.
51. B. J. Jonkman, M. L. Buhl Jr, TurbSim user's guide, Technical Report, National Renewable Energy Lab.(NREL), Golden, CO (United States), 2006.
52. J. Jonkman, A. Robertson, G. Hayman, Development of SubDyn for analysis of offshore wind turbine support structures, in: Proceedings of the 24th International Ocean and Polar Engineering Conference, ISOPE, Busan, Korea, 2014.
53. H. Song, R. Damiani, A. Robertson, J. Jonkman, Dynamic Modeling and Load Analysis of Offshore Wind Turbines, Technical Report NREL/CP-5000-58089, National Renewable Energy Laboratory, Golden, CO, 2013.
54. P. Passon, M. Kühn, State-of-the-art and development needs of simulation codes for offshore wind turbines, in: Copenhagen Offshore Wind Conference, Copenhagen, Denmark, 2012.
55. I. B. Løken, A. M. Kaynia, Effect of foundation type and modelling on dynamic response and fatigue of offshore wind turbines, Wind Energy 22 (2019) 1667–1683. doi:10.1002/we.2394.
56. J. M. Jonkman, A. Robertson, G. J. Hayman, Hydrodyn user's guide and theory manual, National Renewable Energy Laboratory 610 (2014).
57. M. B. Wilk, R. Gnanadesikan, Probability plotting methods for the analysis for the analysis of data, Biometrika 55 (1968) 1–17.
58. A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, PyTorch: An Imperative Style, High-Performance Deep Learning Library, 2019. doi:10.48550/arXiv.1912.01703. arXiv:1912.01703.