Floating Spar-Type Offshore Wind Turbine Hydrodynamic Response Characterisation: a Computational Cost Aware Approach

1th Andrea Coraddu, 3rd Miltos Kalikatzarakis, 5th Maurizio Collu
Naval Architecture,
Ocean & Marine Engineering
Strathclyde University
{andrea.coraddu, miltiadis.kalikatzarakis, maurizio.collu}@strath.ac.uk

2nd Luca Oneto, 4th Davide Ilardi
DIBRIS - University of Genoa
{luca.oneto, davide.ilardi}@unige.it

***Abstract*—The hydromechanics analysis of floating offshore wind turbines is a fundamental and time consuming part of the design process, traditionally analysed with methods of computational fluid dynamics. In this work, an alternative computational framework is suggested, able to significantly accelerate the design process with minimal accuracy loss. Through the use of a state-of-the-art potential-flow code, a surrogate model is developed with the aim to approximate the Response Amplitude Operators of any arbitrary floating offshore wind turbine of the spar buoy type. The results, measured in terms of accuracy and computational effort, demonstrate that this approach is able to approximate the potential-flow solver with very high accuracy at a fraction of the computational cost.**

***Index Terms*—Surrogate models, Extreme Learning Machines, Response Amplitude Operators, Floating Offshore Wind Turbine, Hydromechanics Analysis**

I. Introduction

Wind energy is expected to make significant contributions in the achievement of energy policy commitments in the global power sector. Currently, onshore wind farms are the major contributor, however, recent trends in the renewable energy industry indicate that offshore wind farms will soon be at the forefront of the fight against climate change [1]. Therefore, far, offshore wind farms have been developed in near-shore and shallow waters, utilising mostly fixed bottom structures. Nonetheless, it has been identified that the best wind resources exist beyond the reach of these structures, in deeper locations where the wind speed is higher and more consistent, larger areas are available, and the use of large and costly installation vessels can be potentially avoided. These locations can be accessed by Floating Offshore Wind Turbines (FOWT), however, the harsh operating conditions and the lower level of maturity pose significant engineering challenges in the design and construction of FOWTs. Among these challenges is the development of accurate modelling tools with increased flexibility, to facilitate the complex and iterative design process [2]–[4].

In order to shorten the lengthy design cycles, part of the research community has focused on reducing the computational time requirements of the necessary modelling tools. To this end, several researches have proposed the use of surrogate models in the design and optimisation of wind turbines. Surrogate models have had several successful applications in the field of Computational Fluid Dynamics (CFD), and are known for their high computational efficiency and the ability to represent functions of very high complexity [5]–[9]. In [10], a surrogate optimisation framework was developed for the aerodynamic design of wind-turbine rotors, using a three-dimensional viscous-inviscid interaction code. The proposed methodology was compared to the blade element-momentum theory approach, with the final results showing remarkable similarities at a fraction of the computational cost. A Response Surface Model (RSM) was developed in [11] to reduce the computational time of a complex two-step, multi-objective optimisation of a wind turbine blade. RSMs were also utilised in [12], [13], where expensive numerical computations of the Navier-Stokes equations were originally used to design wind-turbine airfoil profiles. Similarly, Kriging and Artificial Neural Networks (ANNs) were also shown to be preferable in the design of wind turbine airfoils than traditional CFD methods in [14], [15], respectively.

Although these approaches clearly demonstrate that surrogate models exhibit a favourable trade-off between computational complexity and accuracy, the design process of FOWT structures poses additional challenges that remain to be addressed. One of these challenges, and a fundamental part of the conceptual and preliminary design, is the assessment of the hydromechanics characteristics of these structures, and more precisely: the added mass matrix, the radiation damping matrix, and the wave load transfer functions, as functions of the frequency of oscillation.

In practice, several configurations are initially considered, and their dynamic response to the environmental loads is assessed. Based on these initial evaluations, one or more new configurations are proposed and assessed once again. The current state-of-the-art high accuracy approaches for this analysis are based on CFD and Higher Order Boundary Element Methods (HOBEM) [16]–[18]. However, their high computational cost and the large number of load cases that need to be considered restrict the full exploration of the design space due to practical limitations: the standard resources that are usually available to engineers involved in FOWT analysis and design are not sufficient to sustain the necessary iterations [19]. This severely limits the speed of the design process, and as such, the quality of the final configuration.

Despite the increasing efforts to incorporate surrogate models in several areas of wind turbine design and analysis, their application on hydrodynamic response estimation of FOWT structures has yet to be addressed. For this reason, this work demonstrates the feasibility and assesses the performance of a surrogate model in this particular design aspect. Namely, the hydrodynamic response of FOWT foundation structures of the spar-buoy type. For the reasons that are discussed in the sections that follow, a particular sub-family of ANNs has been employed, known as Extreme Learning Machines (ELMs), developed utilising a dataset of simulations from a state-of-the-art, potential-flow-based computational code [20]. The dataset will serve as input to the Model Selection (MS) and Error Estimation (EE) phases, while the remaining part of the data will be used for validation and verification. In particular, to assess the ability of the ELM to approximate the results of the computationally expensive potential-flow solver, we evaluated two different scenarios: an interpolation scenario, in which the design parameters lie within the design space sampled to develop the ELM, and an extrapolation scenario, in which the foundation parameters lie outside the boundaries of the original design space.

This paper is organised as follows: Section II gives an overview of the computational code employed to generate a limited dataset of the hydrodynamic responses, in terms of Response Amplitude Operators (RAOs), of FOWT structures with different geometries. Section III describes the sampling methodology used to select specific geometries from the design space as input to the ELM, and Section IV discusses the process of learning the RAOs. Finally, Section V discusses the performance and time requirements of the proposed method, and Section VI gives the conclusion and recommendations for further work.

II. Hydromechanics analysis

The system of equations of motion for a floating body in regular waves is [21]:

$$\sum_{j=1}^{6} \xi_j \left[-\omega^2 \left(M_{kj} + a_{kj}\right) + i\omega b_{kj} + c_{kj}\right] = \eta X_k, \quad k = 1, ..., 6 \tag{1}$$

where $\boldsymbol{M}$ is the total system matrix, $\boldsymbol{a_{kj}}$ is the hydrodynamic added mass coefficient, $\boldsymbol{b_{kj}}$ is the radiation damping coefficient (no viscous forces have been considered), and $\boldsymbol{c_{kj}}$ is the sum of the hydrostatic and mooring system stiffness coefficients. $\boldsymbol{\xi_j}$ is the j-th degrees of freedom displacement (rigid platform global response), $\boldsymbol{\eta}$ is the wave amplitude, and $\boldsymbol{X_k}$ is the first order wave load transfer function.

These are six simultaneous linear equations of motion, which can be solved to obtain the body displacement in the j-th DOF:

$$\xi_j = \eta \sum_{k=1}^{6} \frac{X_k}{-\omega^2 \left(M_{kj} + a_{kj}\right) + i\omega b_{kj} + c_{kj}} \tag{2}$$

The complex response transfer function between the amplitude of the wave and the amplitude of the oscillation of the system in the j-th DOF is therefore:

$$H_j = \frac{\xi_j}{\eta} = \sum_{k=1}^{6} \frac{X_k}{-\omega^2 \left(M_{kj} + a_{kj}\right) + i\omega b_{kj} + c_{kj}} \tag{3}$$

The RAO in the j-th DOF is defined as the complex magnitude of the transfer function $\boldsymbol{H_j}$:

$$RAO_j = |H_j| \tag{4}$$

A. Hydrodynamics analysis

The software NEMOH is a Boundary Element Methods (BEM) code calculating the wave loads on offshore structures (added mass, radiation, damping, diffraction forces), developed by researchers at Ecole Centrale de Nantes. In the present work, it has been used to estimate the dynamic response of the floating support structures analysed. Differently from other BEM softwares, NEMOH's approach decouples the resolution of the linear free surface Boundary Value Problem (BVP) and the definition of the boundary condition on the body (body condition). For further details, please refer to [20].

In particular, for the present work, NEMOH has been used to derive

- the $\boldsymbol{A_{6\times 6}}(\omega)$ and $\boldsymbol{B_{6\times 6}}(\omega)$ added mass and radiation damping matrices, function of the frequency;
- the $\boldsymbol{X_k}(\omega)$ first order wave load transfer functions (incident and diffraction potential);
- the $\boldsymbol{C_{6x6}}$ hydrostatic restoring matrix.

It should be noted that, in the present work, being the floating support structures analysed always axisymmetric, only one incident wave direction has been considered.

B. Mass matrix and centre of gravity

At this preliminary design stage, the support structure mass is usually estimated as a fraction of the mass of displace water [22], and in the present case this fraction is equal to 0.26. Then, the support structure mass matrix $\boldsymbol{M^{ss}_{6x6}}$ is estimated by considering each frustum of a cone element mass distributed on the external panel, also allowing the estimation of its centre of gravity $\boldsymbol{CoG}^{ss}_{1\times 3}$.

The wind turbine mass matrix, $\boldsymbol{M^{wt}_{6x6}}$, and its centre of gravity, $\boldsymbol{CoG}^{wt}_{1\times 3}$, is an input to the problem, and the ones assumed here are those of the open access NREL 5 MW reference offshore wind turbine [23].

The ballast mass can be therefore calculated using the floatability requirement, i.e., the sum of the wind turbine weight, the support structure weight, and the ballast weight has to be equal to the total buoyancy force. The ballast mass matrix $\boldsymbol{M_{6x6}^{bal}}$ and the ballast centre of gravity $\boldsymbol{CoG}^{bal}_{1\times3}$ are then derived, considering the ballast filling the lowest part of the support structure. The material considered for the ballast is a mix of seawater and heavy solid material, with a density equal to ρ = 1800 $\frac{kg}{m^3}$.

Therefore, the mass matrix and the centre of gravity of the whole system are derived as follow

$$M_{6x6} = M_{6x6}^{ss} + M_{6x6}^{wt} + M_{6x6}^{bal} \tag{5}$$

$$CoG_{1\times3} = \frac{m^{ss}CoG^{ss} + m^{wt}CoG^{wt} + m^{bal}CoG^{bal}}{m} \tag{6}$$

where $\boldsymbol{m^i = M^i(1,1)}$ is the mass of the $i-th$ subsystem.

C. Mooring system restoring forces

The mooring system stiffness matrix has been considered constant and equal to the one provided for the open access OC3 Spar floating offshore wind turbine [24].

III. Dataset Creation

Our ability to generate input data for the learning phase can be exploited to positively affect the performance of the surrogate model. As such, special consideration was given in the careful sampling of the design space, to generate a limited set of representative geometries. We defined the design space on the basis of the foundation's draft and the external radii of the axi-symmetric platform, at six different depths along the vertical axis of the structure. To sample the design space, the external radii were varied between one and six meters, with a discretization interval of one meter, whereas the foundation draft was varied between 50 and 140 meters, with a discretization interval of 10 meters. A full factorial design of the experiments was employed to allow us to capture the joint effects of all the design parameters on each of the RAOs. Given the discretization used, a total of 2187 geometries were generated.

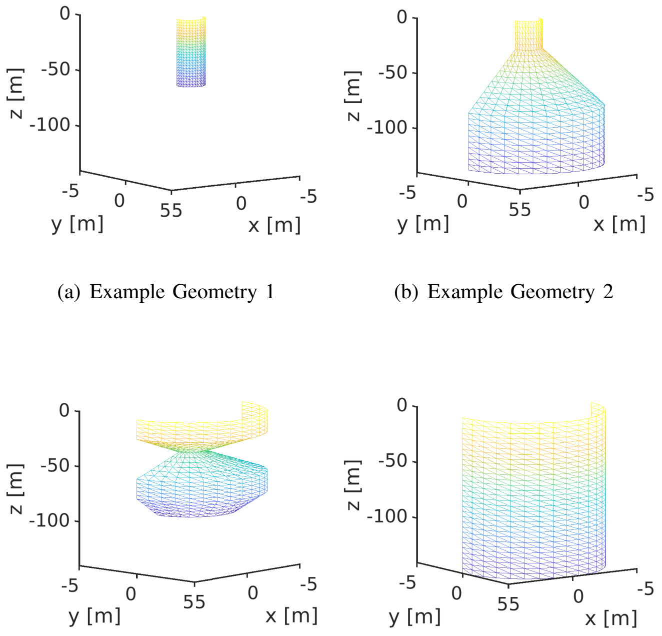
Finally, the mesh of each geometry, along with the load cases to be evaluated were inserted into the potential-flow based code to derive the global rigid-body response of each platform quantified by the 6DOF RAOs. Examples of generated geometries are given in Fig. 1.

IV. Learning the RAO

The problem of learning the RAO based on the data described in Section III can be easily mapped in the now-classical Machine Learning regression problem [25]. However, our scope is broader, and we want to learn a model with limited computational requirements, namely, the computational effort to compute the model output given its inputs should be as limited as possible. As we will see in this section, this constraint will guide us to a particular solution.

Let us recall the now-classical Machine Learning regression problem [25]. Let $\mathcal{X} \subseteq \mathbb{R}^d$ be the input space consisting of f features, and let $\mathcal{Y} = \mathbb{R}$ be the output space. Let $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$, where $X_i \in \mathcal{X}$ and $Y_i \in \mathcal{Y}$ $\forall i \in \{1, \cdots, n\}$, be a sequence of $n \in \mathbb{N}^*$ samples drawn independently from an unknown probability distribution μ over $\mathcal{X} \times \mathcal{Y}$. Let us consider a model (function) $f : \mathcal{X} \to \mathcal{Y}$ chosen from a set $\mathcal{F}$ of possible ones. An algorithm $\mathscr{A}_{\mathcal{H}} : \mathcal{D}_n \times \mathcal{F} \to f$ characterised by its hyperparameters $\mathcal{H}$ selects a model inside a set of possible ones based on the available dataset.

Note that many algorithms for solving regression problems exist in the literature [25]. In particular, it is possible to identify three main different families of methods which are mostly effective in practice [25]–[27]: the kernel methods, the ensemble methods, and the neural networks. As we will see later, our constraints on the computational requirements of the final learned model will guide us to the use of a particular sub-family of neural networks called ELM [28], [29]. In fact,



(a) Example Geometry 1

(b) Example Geometry 2

(c) Example Geometry 3

(d) Example Geometry 4

Fig. 1. Sample of geometries for the learning phase.

TABLE I
List of Features in the Dataset

Name	Description
$\boldsymbol{r}_{1\times6}$	External radii of the platform
T	Foundation draft
$\boldsymbol{CoG}^{wt}_{1\times3}$	Wind turbine center of gravity
z_G^{ss}	Support structure center of gravity - vertical position
z_G^{ts}	Total structure center of gravity - vertical position
$\boldsymbol{M}^{ss}_{6\times6}$	Support structure mass matrix
$\boldsymbol{M}^{ts}_{6\times6}$	Total structure mass matrix
$\boldsymbol{C}^{m}_{6\times6}$	Mooring system restoring matrix
$\boldsymbol{C}^{hst}_{6\times6}$	Hydrostatics and ballast restoring matrix
$\boldsymbol{\omega}_{1\times207}$	Frequency vector

kernel methods [30] and ensemble methods [31], [32] are known to be very effective, but the computational requirements grow a lot if we want to increase the accuracy. ELM, instead, allows to easily build and learn highly non-linear regressors with the use of random projection. Even if this approach is quite effective, still the computational requirement of an ELM can be high since many random projection may be not very useful for increasing the accuracy. For this reason, we will purpose to clean up the number of random projections by keeping just the ones which are really useful and effective in improving the regressors accuracy.

The error of f in approximating $\mathbb{P}\{Y \mid X\}$ is measured by a prescribed metric $M : \mathcal{F} \to \mathbb{R}$. Note that many different metrics are available in the literature for regression, which may provide insights on the performance of the model [33]. In our case, we will make use of the Mean Absolute Percentage Error (MAPE). In order to define it, let us first consider a subset of the available data $\mathcal{T}_t$, also called test set, coming from μ but different from $\mathcal{D}_n$ since the data that have been used to learn f should be different to the ones exploited to evaluate its performance so to avoid overfitting [34]. Then we can define the MAPE as

$$\text{MAPE}(f){=}\text{MAPE}(f,\mathcal{T}_t){=}\frac{100}{t}\sum_{(X,Y)\in\mathcal{T}_t}\left|\frac{f(X)-Y}{Y}\right| \quad (7)$$

Finally, to tune the performance of the $\mathscr{A}_{\mathcal{H}}$, namely to select the best set of hyperparameters, and to estimate the performance of the final model according to the desired metrics, a MS and EE phase needs to be performed [34]. Finally we will also measure the time, in seconds, of computing $f(X)$ (TIME(f)) on the same infrastructure where the BEM Model has been run (see Section II).

A. The Proposed Model

As we previously described, the method that we propose is based on ELM, plus a smart cleaning strategy of the random generated features in the ELM.

The ELM approach was originally introduced to overcome problems posed by back-propagation training algorithm in classical neural networks; specifically, potentially slow convergence rates, the critical tuning of optimisation parameters, and the presence of local minima that call for multi-start and re-training strategies [28], [29]. ELM was originally developed for the single-hidden-layer feed-forward neural networks and then generalised in order to cope with cases in which the ELM is not neuron alike.

$$f(X) = \sum_{j=1}^{h} \boldsymbol{w}_j g_j(X), \quad (8)$$

where $g_j : \mathbb{R}^d \to \mathbb{R},\ j \in \{1, \cdots, h\}$ is the hidden-layer output corresponding to the input sample X and $\boldsymbol{w}$ is the output weight vector between the hidden layer and the output layer.

In our case, the input layer has d neurons and connects to the hidden layer (having h neurons) through a set of weights

$$\boldsymbol{v}_j \in \mathbb{R}^d, \quad j \in \{1, \cdots, h\}, \quad (9)$$

the j-th hidden neuron embeds a bias term,

$$v_j^0, \quad j \in \{1, \cdots, h\}, \quad (10)$$

and a nonlinear activation function, $\varphi : \mathbb{R} \to \mathbb{R}$ (in our case the hyperbolic tangent). Thus, the neuron's response to an input stimulus, X, is

$$\varphi(\boldsymbol{v}_j \cdot X + v_j^0), \quad j \in \{1, \cdots, h\}. \quad (11)$$

Note that Eq. (11) can be further generalised to include a wider class of functions; therefore, the response of a neuron to an input stimulus X can be generically represented by any nonlinear piece-wise continuous function characterised by a set of parameters. In ELM, these parameters ($\boldsymbol{v}_j$ and v_j^0) are set randomly (in our case these parameters are sampled from the Normal distribution). A vector of weighted links, $\boldsymbol{w} \in \mathbb{R}^h$, connects the hidden neurons to the output neuron without any bias. The overall output function, $f(X)$, of the network is

$$f(X) = \sum_{j=1}^{h} \boldsymbol{w}_j \varphi(\boldsymbol{v}_j \cdot X + v_j^0). \quad (12)$$

It is convenient to define an activation matrix $V \in \mathbb{R}^{n\times h}$, such that the entry $V_{i,j}$ is the activation value of the j-th hidden neuron for the i-th input pattern. The V matrix is

$$V{=}\begin{bmatrix}\varphi(\boldsymbol{v}_1{\cdot}X_1{+}v_1^0) & \cdots & \varphi(\boldsymbol{v}_h{\cdot}X_1{+}v_h^0)\\ \vdots & \ddots & \vdots\\ \varphi(\boldsymbol{v}_1{\cdot}X_n{+}v_1^0) & \cdots & \varphi(\boldsymbol{v}_h{\cdot}X_n{+}v_h^0)\end{bmatrix}{=}\begin{bmatrix}\boldsymbol{\phi}^T(X_1)\\ \vdots\\ \boldsymbol{\phi}^T(X_n)\end{bmatrix}. \quad (13)$$

In the ELM model, the quantities $\{\boldsymbol{v}_j, v_j^0\}$ in Eq. (11) are set randomly and are not subject to any adjustment, and the quantity $\boldsymbol{w}$ in Eq. (12) is the only degree of freedom. Hence, the training problem reduces to minimisation of the convex cost

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \|V\boldsymbol{w} - \boldsymbol{y}\|^2. \quad (14)$$

A matrix pseudo-inversion yields the unique L_2 solution

$$\boldsymbol{w}^* = V^+\boldsymbol{y}. \quad (15)$$

The simple, efficient procedure to train an ELM therefore involves the following steps

1) Randomly generate hidden node parameters (in or case $\boldsymbol{v}_i$ and bias v_i^0) for each hidden neuron;
2) Compute the activation matrix V, of Eq. (13);
3) Compute the output weights by solving the pseudo-inverse problem of Eq. (15).

Despite the apparent simplicity of the ELM approach, the crucial result is that even random weights in the hidden layer endow a network with notable representation ability. Moreover, regularisation strategies can further improve the approach's generalisation performance. As a result, the cost function of Eq. (14) is augmented by a regularisation factor as follows

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \|V\boldsymbol{w} - \boldsymbol{y}\|^2 \quad \text{and} \quad \|\boldsymbol{w}\|, \quad (16)$$

where $\|\boldsymbol{w}\|$ can be any suitable norm of the output weights. A common approach is then to use the L_2 regularizer

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \|V\boldsymbol{w} - \boldsymbol{y}\|^2 + \lambda \|\boldsymbol{w}\|^2 , \tag{17}$$

and consequently the vector of weights $\boldsymbol{w}^*$ is then obtained as follows

$$\boldsymbol{w}^* = (V^T V + \lambda I)^{-1} V^T \boldsymbol{y}, \tag{18}$$

where $I \in \mathbb{R}^{h \times h}$ is an identity matrix.

Note that, also intuitively, not all the random projection will be useful for actually improving the accuracy of the model. For this reason, we propose to incrementally add random projection. In particular, instead of contemporary extracting all the h projections (Step 2 in the ELM), the model is built incrementally adding n_h projections at the time and keeping only if the increment in accuracy is at least $p_h\%$. This simple strategy helps the model in generating only informative random projection, reducing the number of parameters required by the regressor to reach a good accuracy.

B. Model Selection and Error Estimation

MS and EE deal with the problem of tuning and assessing the performance of a learning algorithm [34]. Resampling techniques like k-fold cross validation and non-parametric bootstrap are often used by practitioners because they work well in many situations [35]. Other alternatives exist, which represent bases in the Statistical Learning Theory and give more insight into the learning process. Examples of methods in this last category are: the seminal work of the Vapnik-Chervonenkis Dimension, its improvement with the Rademacher Complexity, the theory of compression, the Algorithmic Stability breakthrough, the PAC-Bayes theory, and more recently the Differential Privacy theory [34].

In this work, we will exploit the resampling techniques which rely on a simple idea: the original dataset $\mathcal{D}_n$ is resampled once or many (n_r) times, with or without replacement, to build three independent datasets called learning, validation and test sets, respectively $\mathcal{L}_l^r$, $\mathcal{V}_v^r$, and $\mathcal{T}_t^r$, with $r \in \{1, \cdots, n_r\}$. Note that $\mathcal{L}_l^r \cap \mathcal{V}_v^r = \oslash$, $\mathcal{L}_l^r \cap \mathcal{T}_t^r = \oslash$, $\mathcal{V}_v^r \cap \mathcal{T}_t^r = \oslash$, and $\mathcal{L}_l^r \cup \mathcal{V}_v^r \cup \mathcal{T}_t^r = \mathcal{D}_n$ for all $r \in \{1, \cdots, n_r\}$.

Then, to select the best combination of the hyperparameters $\mathcal{H}$ in a set of possible ones $\mathfrak{H} = \{\mathcal{H}_1, \mathcal{H}_2, \cdots\}$ for the algorithm $\mathscr{A}_{\mathcal{H}}$ or, in other words, to perform the MS phase, the following procedure has to be applied

$$\mathcal{H}^* : \quad \arg\min_{\mathcal{H} \in \mathfrak{H}} \sum_{r=1}^{n_r} M(\mathscr{A}_{\mathcal{H}}(\mathcal{L}_l^r), \mathcal{V}_v^r), \tag{19}$$

where $\mathscr{A}_{\mathcal{H}}(\mathcal{L}_l^r)$ is a model built with the algorithm $\mathscr{A}$ with its set of hyperparameters $\mathcal{H}$ and with the data $\mathcal{L}_l^r$ and where $M(f, \mathcal{V}_v^r)$ is a desired metric. Since the data in $\mathcal{L}_l^r$ are independent from the ones in $\mathcal{V}_v^r$, the idea is that $\mathcal{H}^*$ should be the set of hyperparameters that allows to achieve a small error on a data set that is independent from the training set.

Then, to evaluate the performance of the optimal model, which is $f_{\mathscr{A}}^* = \mathscr{A}_{\mathcal{H}^*}(\mathcal{D}_n)$ or, in other words, to perform the EE phase, the following procedure has to be applied

$$\text{MAPE}(f_{\mathscr{A}}^*) = \frac{1}{n_r} \sum_{r=1}^{n_r} \text{MAPE}(\mathscr{A}_{\mathcal{H}^*}(\mathcal{L}_l^r \cup \mathcal{V}_v^r), \mathcal{T}_t^r). \tag{20}$$

Since the data in $\mathcal{L}_l^r \cup \mathcal{V}_v^r$ are independent from the ones in $\mathcal{T}_t^r$, $M(f_{\mathscr{A}}^*)$ is an unbiased estimator of the true performance, measured with the metric M, of the final model [34].

If $n_r = 1$, if l, v, and t are aprioristically set such that $n = l + v + t$, and if the resampling procedure is performed without replacement, the hold out method is obtained [34]. For implementing the complete nested k-fold cross validation, instead, it is needed to set $n_r \leq \binom{n}{k}\binom{n-\frac{n}{k}}{k}$, $l = (k-2)\frac{n}{k}$, $v = \frac{n}{k}$, and $t = \frac{n}{k}$ and the resampling must be done without replacement [35]. Finally, for implementing the nested non-parametric bootstrap, $l = n$ and $\mathcal{L}_l^r$ must be sampled with replacement from $\mathcal{D}_n$, while $\mathcal{V}_v^r$ and $\mathcal{T}_t^r$ are sampled without replacement from the sample of $\mathcal{D}_n$ that has not been sampled in $\mathcal{L}_l^r$ [35]. Note that for the bootstrap procedure, $n_r \leq \binom{2n-1}{n}$. In this paper, the complete nested k-fold cross validation is exploited because it represents the state-of-the-art approach [34], [35].

V. Experiments Results

We built a model using the MS strategy where we set $k = 10$ and $n_r = 1000$. During the MS, we searched the hyperparameters using the following ranges: $h \in 10^{\{1.0, 1.2, 1.4, \cdots, 4.0\}}$, $\lambda \in 10^{\{} - 6.0, -5.8, \cdots, +4.0\}$, $n_h = {}^{h}/\{10, 20, 40, 80, 160\}$, and $p_h \in \{0.5, 1.0, 2.0, 4.0, 8.0, 16\}$.

All the developed models have a TIME (f) of a few milliseconds maximum.

Two modelisation scenarios have been investigated

- Interpolation Scenario: in this case the models try to predict the RAOs in various, but different, configurations of the design parameters, as discussed in Section III, within the ones exploited for building the model. In other words, this scenario is accounting for configurations that belong inside the search space used to build the dataset;
- Extrapolation Scenario: in this scenario, the models try to predict the RAOs in various, but different, configurations of the design parameters, with respect to those exploited for building the model. This second scenario, instead is considering configurations that belong outside of the search space used to build the dataset.

Basically the two scenarios just differ in the definitions of $\mathcal{L}_l$, $\mathcal{V}_v$, and $\mathcal{T}_t$, that are the subset of data exploited for building, tuning, and testing the models.

The interpolation scenario is obviously the simplest one. In this scenario $\mathcal{L}_l$, $\mathcal{V}_v$, and $\mathcal{T}_t$ have been created by splitting randomly all the samples of the datasets described in Section III. In this way the models have been tested in their ability to predict the RAO in various, but different, conditions within those exploited for building the models.

On the contrary, the extrapolation scenario tests the capability of the models to predict the RAO for cases not included in the variable domain of the data used to build them. The practical application of the extrapolation scenario is to demonstrate that the regression model can be utilised outside the limits of the design variables (external radii and foundation drafts). As such, the proposed model can be embedded within a broader optimisation framework, having a much broader search space for the design variables. In order to obtain an indication of the extrapolation performance, the authors performed a number of extrapolations scenarios, one for each design variable ($r_i,\ i \in 1, ..., 6$, and T), and one additional scenario, for the extrapolation of all design variables simultaneously. The results are reported in Table II.

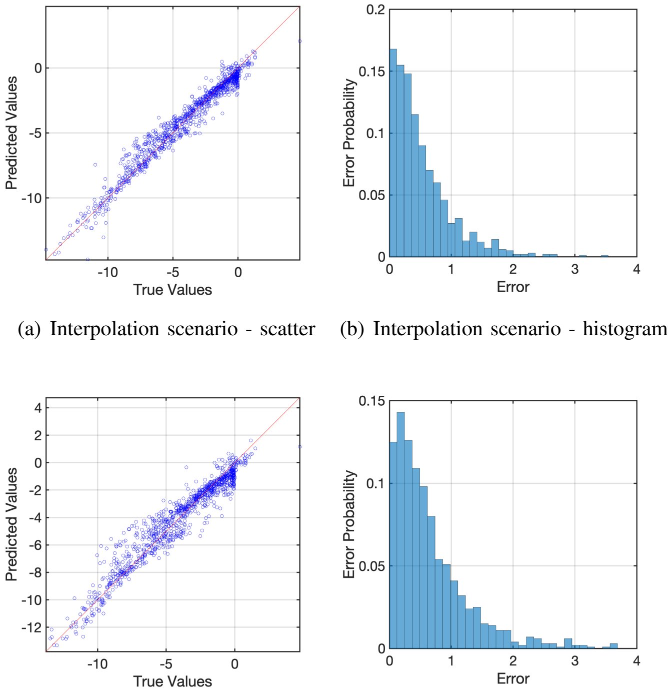
From the results, we can note that an average MAPE of 2% is achieved across the different RAOs for the interpolation scenario. As expected, a higher MAPE is observed for all the considered extrapolation scenarios. The highest MAPE is obtained when all design variables lie outside the initial search space $\mathcal{X}$. Nonetheless, it is worth noting that, in the worst case scenario, an average MAPE of 5% is reported. A more detailed description of the results for the surge, heave, and pitch motions is given in Figs. 2 - 4. We can observe that for the majority of the geometries, the ELM forecast is very close to the BEM-based solution (vertical distance between the blue circles and the red line). Moreover, the error distribution for the surge and heave motions for both the interpolation and extrapolation scenarios reveals that for 98% of the geometries, the MAPE is less than 2%. While, for the pitch motion, the same error is less than 1%, which is also verified by the smaller variance presented in Fig. 4.

To better illustrate the ability of the regression model to accurately predict the 6 RAOs, and to provide the reader with a visual impression of the results, the authors report the RAOs predicted with the high-fidelity BEM solver and the ELM in Fig. 5, for the geometry of Fig. 6, characterised by the following design parameters: $\mathcal{X} = \{1, 1, 5, 5, 1, 5, 120\}$ (all the quantities are reported in meters).

Due to the fact that the wet geometry is axisymmetric, the only incident wave direction considered has been 0 deg (i.e., aligned with x), and therefore the only RAO considered are those in surge, heave, pitch, and yaw, since the sway RAO will be equal to the surge RAO, and the one in roll equals the pitch RAO. Again, due to the wet geometry being axysimmetric, the excitation in yaw is minimal, negligible with respect to the others. Nevertheless, we can safely state that the ELM is able to predict the behaviour of the BEM solver, even purely in terms of numerical noise. Moreover, the RAOs for surge, heave, and pitch motions are well captured for the entire frequency spectrum. It is worth noting that the learning phase has been carried out in a base-10 logarithmic scale, to properly capture the behaviour of the RAOs near the resonance frequencies for all geometries. For the sake of consistency, we have reported the results of Fig. 5 in base-10 logarithmic scale.

VI. Conclusions & Discussion

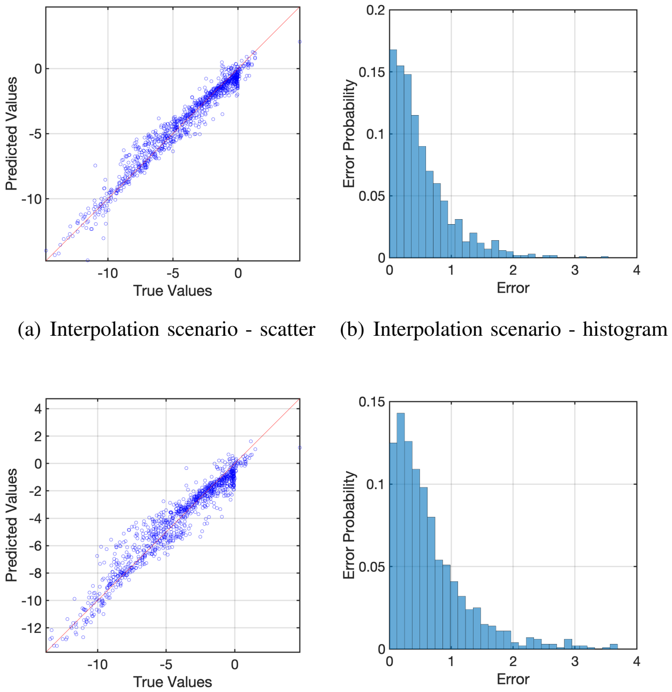
In this paper, the authors presented a surrogate model for the hydromechanics analysis of floating offshore wind



(a) Interpolation scenario - scatter (b) Interpolation scenario - histogram

(c) Extrapolation on $\mathcal{X}$ - scatter (d) Extrapolation on $\mathcal{X}$ - histogram

Fig. 2. Surge motion ELM forecast



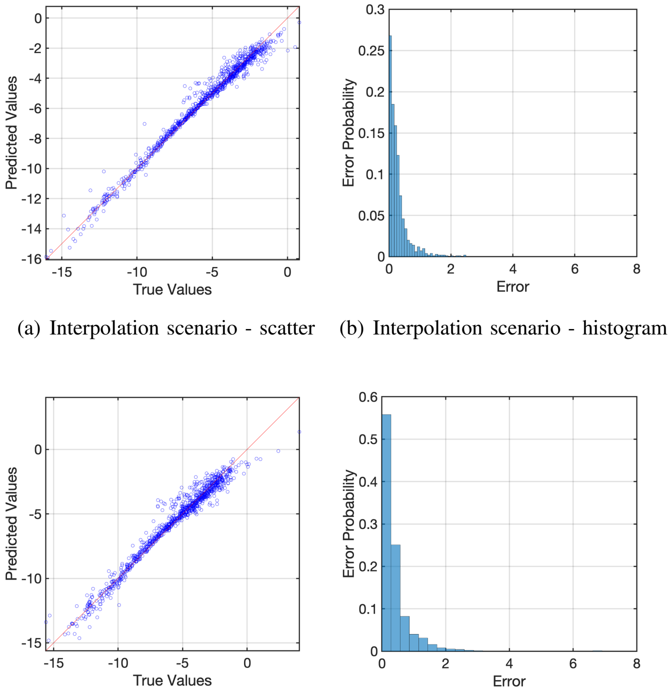
(a) Interpolation scenario - scatter (b) Interpolation scenario - histogram

(c) Extrapolation on $\mathcal{X}$ - scatter (d) Extrapolation on $\mathcal{X}$ - histogram

Fig. 3. Heave motion ELM forecast

TABLE II
MEAN ABSOLUTE PERCENTAGE ERROR

Scenario	Surge	Heave	Pitch
Interp	2.52 ± 0.30	2.52 ± 0.33	1.43 ± 0.20
r_1	3.51 ± 0.43	4.11 ± 0.33	2.11 ± 0.23
r_2	4.11 ± 0.39	3.76 ± 0.47	2.00 ± 0.22
r_3	3.45 ± 0.25	3.62 ± 0.48	2.17 ± 0.25
r_4	3.65 ± 0.34	3.45 ± 0.30	2.22 ± 0.15
r_5	3.41 ± 0.43	3.57 ± 0.30	2.07 ± 0.21
r_6	3.82 ± 0.44	3.36 ± 0.29	2.02 ± 0.22
T	3.40 ± 0.28	3.82 ± 0.51	2.00 ± 0.16
$\mathcal{X}$	5.78 ± 0.27	5.51 ± 0.62	3.75 ± 0.49

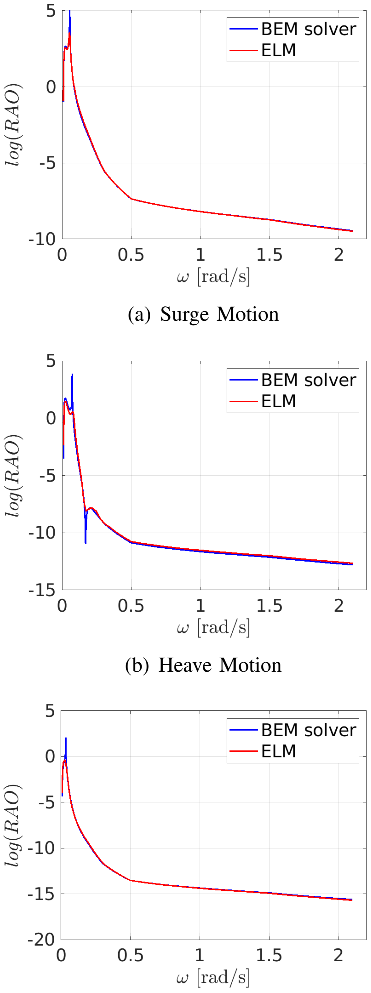


(a) Interpolation scenario - scatter (b) Interpolation scenario - histogram

(c) Extrapolation on $\mathcal{X}$ - scatter (d) Extrapolation on $\mathcal{X}$ - histogram

Fig. 4. Pitch motion ELM forecast

turbines. The authors proposed an alternative computational framework to accelerate the design process with minimal accuracy loss and minimal computational requirements. Based on the results of a state-of-the-art potential-flow code on a limited set of geometries, an ELM-based surrogate model has been developed to approximate the Response Amplitude Operators of any arbitrary floating offshore wind turbine of the spar buoy type. The results demonstrate the feasibility of replacing the computationally expensive BEM solver with a fast, yet accurate surrogate model. More specifically, the surrogate ELM-based model can predict the RAOs of any FOWT geometry with an average MAPE of 2% across all the DOFs, when the design variables are within the limits of the search space used to learn the surrogate model. In the more challenging extrapolation scenario, in which the design variables lie outside the limits of the original search



(a) Surge Motion

(b) Heave Motion

(c) Pitch Motion

Fig. 5. Interpolation scenario results

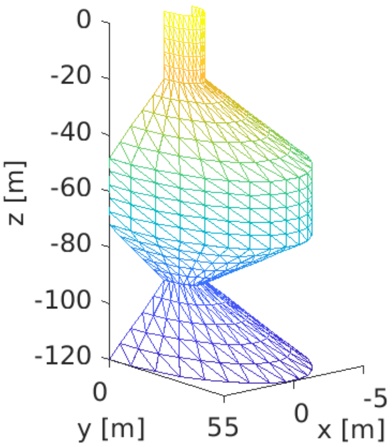


Fig. 6. Interpolation scenario geometry

space, the average MAPE increases up to 5% for all the DOFs. The current state-of-the-art, high fidelity approaches for this analysis are based on CFD and HOBEM. However, the computational cost of these methods limits their application in optimisation, restricting the number of configurations that can be investigated, thus limiting the assessed designs. The proposed ELM-based surrogate model, when exploited within an optimisation framework, can overcome this limitation by enabling fast, robust, and highly accurate hydrodynamics analysis, built-on high-accuracy numerical simulation data. We expect that the proposed approach will pave the way towards innovative support platform geometries investigation and design, optimising the hydrodynamics response, which can substantially enhance the performance of FOWTs.

VII. Acknowledgements

This work is supported by the UK Engineering and Physical Sciences Research Council UK (EPSRC) and the Natural Environment Research Council UK (NERC), through grant EP/R007497/1 and EP/R007497/2

References

[1] M. Hannon, E. Topham, J. Dixon, D. McMillan, and M. Collu, *Offshore wind, ready to float? Global and UK trends in the floating offshore wind market*. University of Strathclyde, 2019.

[2] H. Yang and Y. Zhu, "Robust design optimization of supporting structure of offshore wind turbine," *Journal of Marine Science and Technology*, vol. 20, no. 4, pp. 689–702, 2015.

[3] M. Borg, A. Shires, and M. Collu, "Offshore floating vertical axis wind turbines, dynamics modelling state of the art. part i: Aerodynamics," *Renewable and Sustainable Energy Reviews*, vol. 39, pp. 1214–1225, 2014.

[4] M. Borg and M. Collu, "Offshore floating vertical axis wind turbines, dynamics modelling state of the art. part iii: Hydrodynamics and coupled modelling approaches," *Renewable and sustainable energy reviews*, vol. 46, pp. 296–310, 2015.

[5] S. L. Brunton, B. R. Noack, and P. Koumoutsakos, "Machine learning for fluid mechanics," *Annual Review of Fluid Mechanics*, vol. 52, 2019.

[6] A. Pollard, L. Castillo, L. Danaila, and M. Glauser, *Whither turbulence and big data in the 21st century?* Springer, 2016.

[7] C. W. Rowley and S. T. M. Dawson, "Model reduction for flow analysis and control," *Annual Review of Fluid Mechanics*, vol. 49, 2017.

[8] J. Ling and J. Templeton, "Evaluation of machine learning algorithms for prediction of regions of high reynolds averaged navier stokes uncertainty," *Physics of Fluids*, vol. 27, no. 8, 2015.

[9] A. P. Singh, S. Medida, and K. Duraisamy, "Machine-learning-augmented predictive modeling of turbulent separated flows over airfoils," *AIAA Journal*, pp. 2215–2227, 2017.

[10] M. Sessarego, N. Ramos-García, H. Yang, and W. Z. Shen, "Aerodynamic wind-turbine rotor design using surrogate modeling and three-dimensional viscous–inviscid interaction technique," *Renewable Energy*, vol. 93, pp. 620–635, 2016.

[11] K.-H. Lee, K.-H. Kim, D.-H. Lee, K.-T. Lee, and J.-P. Park, "Two-step optimization for wind turbine blade with probability approach," *Journal of Solar Energy Engineering*, vol. 132, no. 3, 2010.

[12] R. Bourguet, G. Martinat, G. Harran, and M. Braza, "Aerodynamic multi-criteria shape optimization of vawt blade profile by viscous approach," in *Wind Energy*. Springer, 2007, pp. 215–219.

[15] A. Ribeiro, A. Awruch, and H. Gomes, "An airfoil optimization technique for wind turbines," *Applied Mathematical Modelling*, vol. 36, no. 10, pp. 4898–4907, 2012.

[13] H. Sun, "Wind turbine airfoil design using response surface method," *Journal of Mechanical Science and Technology*, vol. 25, no. 5, p. 1335, 2011.

[14] Z. Han, K. Zhang, W. Song, and J. Liu, "Surrogate-based aerodynamic shape optimization with application to wind turbine airfoils," in *51st AIAA aerospace sciences meeting including the new horizons forum and aerospace exposition*, 2013, p. 1108.

[16] T. T. Tran and D. H. Kim, "The coupled dynamic response computation for a semi-submersible platform of floating offshore wind turbine," *Journal of wind engineering and industrial aerodynamics*, vol. 147, pp. 104–119, 2015.

[17] ——, "Fully coupled aero-hydrodynamic analysis of a semi-submersible fowt using a dynamic fluid body interaction approach," *Renewable energy*, vol. 92, pp. 244–261, 2016.

[18] S. Netzband, C. W. Schulz, U. Göttsche, D. Ferreira González, and M. Abdel-Maksoud, "A panel method for floating offshore wind turbine simulations with fully integrated aero-and hydrodynamic modelling in time domain," *Ship Technology Research*, vol. 65, no. 3, pp. 123–136, 2018.

[19] M. Muskulus, "Designing the next generation of computational codes for wind-turbine simulations," in *The Twenty-first International Offshore and Polar Engineering Conference*. International Society of Offshore and Polar Engineers, 2011.

[20] A. Babarit and G. Delhommeau, "Theoretical and numerical aspects of the open source bem solver nemoh," in *LHEEA*, 2015.

[21] J. N. Newman, *Marine hydrodynamics*. MIT press, 2018.

[22] O. Anaya-Lara, J. O. Tande, K. Uhlen, and K. Merz, *Offshore Wind Energy Technology*. Wiley Online Library, 2018.

[23] J. Jonkman, S. Butterfield, W. Musial, and G. Scott, "Definition of a 5-mw reference wind turbine for offshore system development," National Renewable Energy Lab.(NREL), Golden, CO (United States), Tech. Rep., 2009.

[24] J. Jonkman, "Definition of the floating system for phase iv of oc3," National Renewable Energy Lab.(NREL), Golden, CO (United States), Tech. Rep., 2010.

[25] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[26] M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we need hundreds of classifiers to solve real world classification problems?" *The journal of machine learning research*, vol. 15, no. 1, pp. 3133–3181, 2014.

[27] M. Wainberg, B. Alipanahi, and B. J. Frey, "Are random forests truly the best classifiers?" *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 3837–3841, 2016.

[28] G. Huang, G. B. Huang, S. Song, and K. You, "Trends in extreme learning machines: A review," *Neural Networks*, vol. 61, pp. 32–48, 2015.

[29] S. Ding, H. Zhao, Y. Zhang, X. Xu, and R. Nie, "Extreme learning machine: algorithm, theory and applications," *Artificial Intelligence Review*, vol. 44, no. 1, pp. 103–115, 2015.

[30] J. Shawe-Taylor and N. Cristianini, *Kernel methods for pattern analysis*. Cambridge university press, 2004.

[31] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[32] R. Blaser and P. Fryzlewicz, "Random rotation ensembles," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 126–151, 2016.

[33] C. C. Aggarwal, *Data Mining: the Textbook*. Springer, 2015.

[34] L. Oneto, *Model Selection and Error Estimation in a Nutshell*. Springer, 2019.

[35] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *International Joint Conference on Artficial Intelligence*, 1995.