IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 21 July 2022, accepted 8 August 2022, date of publication 16 August 2022, date of current version 23 August 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3199009

RESEARCH ARTICLE

Neural Network Meta-Models for FPSO Motion Prediction From Environmental Data With Different Platform Loads

LUCAS P. COTRIM[1], RODRIGO A. BARREIRA[2], ISMAEL H. F. SANTOS[2], EDSON S. GOMI[1], (Senior Member, IEEE), ANNA H. REALI COSTA[1,3], (Member, IEEE), AND EDUARDO A. TANNURI[1]

[1]Numerical Offshore Tank, Universidade de São Paulo (USP), São Paulo 05508-900, Brazil
[2]Petrobras, Rio de Janeiro 20031-912, Brazil
[3]Intelligent Techniques Laboratory, Universidade de São Paulo (USP), São Paulo 05508-900, Brazil

Corresponding author: Lucas P. Cotrim (lucas.cotrim@usp.br)

This work was supported in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES Finance Code 001), Brazil; in part by ANP/PETROBRAS, Brazil, under Project 21721-6; and in part by CNPq under Grant 310085/2020-9 and Grant 310127/2020-3.

ABSTRACT The current design process of mooring systems for Floating Production, Storage, and Offloading units (FPSOs) depends on the availability of the platform's mathematical model and the accuracy of dynamic simulations. These simulations then provide the FPSO's time series motion which is evaluated according to design constraints. This process can be time-consuming and present inaccurate results due to the mathematical model's limitations and the overall complexity of the vessel's dynamics. We propose a Neural Simulator, called NeuroSim, a set of data-based surrogate models with environmental data as input, each model specialized in predicting different motion statistics relevant to mooring system design: Maximum Roll, Platform Offset, and Fairlead Displacements. The surrogate models are trained by current, wind, and wave data given in 3 hours periods at a Brazilian Offshore Basin from 2003 to 2010, and the associated dynamic response of a spread-moored FPSO is obtained through time-domain simulations using the Dynasim software. Hyperparameter Optimization techniques are performed to obtain optimal Artificial Neural Network (ANN) models specialized in different platform drafts. Finally, the proposed models are shown to correctly capture platform dynamics, providing good results when compared to motion statistics obtained from Dynasim. We conclude that an ANN surrogate model can be trained directly on actual measured metocean conditions and corresponding FPSO motion statistics to provide increased accuracy and reduced computational time over traditional methods based on dynamic simulation. Moreover, the proposed architecture can be integrated into an automated learning framework: The data-based surrogate models can be continuously fine-tuned and updated with newly measured data, improving accuracy over time.

INDEX TERMS Artificial neural networks, floating offshore platforms, hyperparameter optimization, neural architecture search, surrogate models.

I. INTRODUCTION

Current FPSO's Mooring System Design consists in the measurement of local environmental conditions over a representative time period and subsequent dynamic simulation of the FPSO model subject to combinations of the extreme winds, waves, and currents expected in the next 10 to 100 years of operation, which are obtained from statistical projections of the measured environmental conditions. The maximum offsets and mooring line tensions are then obtained and verified to remain within project limits safely.

This process, however, relies on the numerical simulation of a dynamic model on software such as Dynasim, which multiplies the approximated wave energy spectrum and the

The associate editor coordinating the review of this manuscript and approving it for publication was Jamshid Aghaei.

FPSO's RAOs (Response Amplitude Operators) to obtain the expected vessel's movement. This process can be both time consuming and present slightly inaccurate responses compared to the actual measured movement.

Recently, the increasing performance of data-based machine learning models in various domains in conjunction with the high computational times of traditional models and the unprecedented availability of data have motivated the study and development of alternative models, denominated surrogates, or meta-models. The primary motivation behind such models is to directly model complex, computationally costly dynamics through available data. Meta-models have been successfully implemented as alternatives for Finite Element (FE) and Computational Fluid Dynamics (CFD) models for predicting mooring line tensions and submerged riser's vibration responses. Gumley *et al.* [7] successfully implemented a neural network capable of predicting the hourly mean offset of a turret-moored FPSO from environmental conditions, showing that differences in mooring configuration result in differences between predicted and measured offset, which can be used to monitor mooring system integrity.

A. OBJECTIVES

The main objective of this research is to design and validate a set of data-based meta-models capable of predicting relevant statistics associated with an FPSO's dynamic response to generic environmental conditions. The models are trained and validated through data obtained from the simulation of a spread-moored platform subject to 6 years of measured currents, waves, and winds.

In the proposed framework, meta-models are trained to correctly predict the maximum roll, offset and fairlead displacements, obtained through dynamic simulation rather than real FPSO responses. This process allows for validating the proposed architecture's performance without the interference of sensor noise on measured platform responses. Results indicate that the proposed set of meta-models correctly captures the simulated platform's responses and suggest that a similarly structured neural simulator trained on real FPSO responses can be more accurate than traditional dynamic simulation methods.

As the proposed set of artificial neural network meta-models is trained directly on environmental conditions and associated FPSO motion, no meaningful physical interpretation of its underlying operations is available. However, this allows the models to capture complex nonlinear dynamics, such as varying mooring line damping and second-order wave drift, approximated in dynamic simulation methods. Overall, a data-based approach is expected to present three main advantages in comparison to traditional methods:

- **Increased accuracy:** Training directly on real measured environmental conditions and the corresponding platform movement responses avoids several approximations and simplifications of physical phenomena implemented on traditional simulation software. Moreover, the availability of a considerable volume of data (over 18 thousand 3h periods) improves the accuracy of trained data-based models.
- **Automated Learning:** The resulting system is designed to be integrated with other design tools and is continuously updated with newly measured environmental and platform motion data. These updates continuously improve the meta-model's accuracy over time.
- **Reduced Computational Time:** After training, the computational time associated with evaluating a neural network prediction is significantly shorter than that associated with time integration of the system's dynamic equations. As a result, given a set of different environmental conditions, a set of data-based meta-models can obtain relevant platform movement information faster than traditional methods.

The proposed framework has two main applications: First, in response-based Mooring System Design, it can be used in conjunction with traditional simulation software to reduce the computational time associated with the simulation of hundreds of metocean conditions. Second, as a Digital Twin monitoring tool, training with actual measured data and predicting real-time motion statistics given incident environmental conditions.

The main contributions of this work are the proposal, development and validation of a data-driven motion statistics simulator, called NeuroSim, designed to be used as an auxiliary tool in Mooring System Design and Seakeeping applications. In comparison to similar research, the proposed framework is trained on significantly more data (over 18 thousand environmental conditions), utilizes a robust hyperparameter optimization algorithm in order to determine optimal ANN architectures and considers the issue of the FPSO's dynamic response being strongly dependent on platform load (represented in this work by platform draft).

II. LITERATURE REVIEW

Over the past few decades, data-driven Machine Learning (ML) models have presented an increasing performance in various applications, from image recognition and product recommendation to medical diagnosis and language translation. In areas such as Offshore Engineering, where complex non-linear physical phenomena are involved, and there is a high availability of data due to existing sensor measurements required to ensure the safety of operations, this approach has been particularly successful when compared to analytical hydrodynamic models.

Machine Learning applications in Offshore Engineering can be divided into two main categories: Mooring Failure Detection and Response-based Seakeeping Analysis. The former can be seen as a classification problem, in which models are trained to correctly identify mooring line breakage given the vessel's motion, allowing for the appropriate repair of compromised lines while saving expenses associated with

ROV inspection. The latter is considered a regression task, in which data-driven meta-models are trained to accurately predict FPSO motion given incident metocean conditions to verify that vessel's responses such as roll, offset and line tensions remain safely within design criteria.

Initial efforts in FPSO motion prediction through ML models have focused on using simple Artificial Neural Networks (ANNs) to predict a small number of motion statistics. Mazaheri [1] has trained Multi-Layer Perceptrons (MLPs) with 1 Hidden Layer to predict an FPSOs surge, sway, and total excursion over a 3h period given the following 6 environmental variables: significant wave height, wave direction, current speed, current direction, wind speed, and wind direction. The ANN models were trained on data obtained by a Hydrodynamic mathematical model to reduce the computational time associated with time-series simulations in response-based FPSO design.

In the following decade, with the introduction and rise in popularity of Digital Twins, ML models have been applied to time-series motion prediction of FPSO, focusing on real-time prediction of individual variables: such as heave [2], roll [3] and coupled heave-pitch motion [4]. ML-based time-series prediction has also been successfully applied to the prediction of mooring line tensions [5], with the initial segments of the tension time-series obtained by FE and the remaining by Wavelet Neural Network meta-models, which resulted in a significant reduction of computational time. de Pina *et al.* [6] presented a comparison of different types of surrogate models in the prediction of mooring line tension, classified according to the time-series input. The models are Purely Autoregressive, which utilizes the top tension series itself exclusively; exogenous models that use surge, sway and heave time-series; and Nonlinear Autoregressive models with Exogenous inputs, which combine both present and past values of exogenous series as well as past values of the desired series itself.

Gumley *et al.* [7] have successfully used ANNs to predict a turret-moored FPSO's mean offset given input variables such as significant wave height, current speed, and vessel draft. By changing mooring line configuration, a significant difference between ANN's predicted and measured motion statistics is observed, which can be used to monitor the integrity of an FPSOs mooring system. They determined the optimal input variable combinations to be used to predict each motion statistic.

Also with the objective of monitoring the integrity of an FPSO mooring system, Saad *et al.* [8] compared two ANN models. They show that the models were able to detect mooring line failure in near real-time based on the comparison between the FPSO movement measured by sensors and the movement predicted by ANNs.

It is essential to notice that there is no universal standardization of data-driven meta-models, unlike traditional methods based on hydrodynamic equations. Their structure varies according to application and is often determined through trial and error or hyperparameter optimization (Section VI-B), which is exemplified by different authors presenting different choices of input variables to be considered in each model as shown in the papers presented above. The choice of these variables is a significant challenge to ML applications overall, with considerable research efforts directed towards developing more general methods, reducing the necessity of domain-specific knowledge and human intervention. With this in mind, the framework described in IV attempts to be as general as possible, making use of all available variables with no assumption of their relevance to different output variables.

III. THEORETICAL CONCEPTS

This section aims to overview the theoretical foundations required to understand the problem definition and proposed methods presented in the next section. Section III-A describes Dynasim's Hydrodynamic model of an FPSO and how it is used to generate simulations of platform motion when subject to arbitrary environmental conditions. Section III-B illustrates the concept of ANNs, particularly Multi-Layer Perceptrons (MLPs), focusing on the process of learning from data. Finally, section III-C refers to Hyperparameter Optimization methods.

A. PLATFORM DYNAMIC MODEL

Mooring systems are used to provide stability to FPSOs subject to incident currents, wind, and waves by anchoring the platform through the use of mooring lines attached to the sea bed. Such systems are not only a key component to ensure the safety of offshore operations but also pose several engineering challenges, from project and conception to operation and maintenance. Dynamic simulation is a crucial step in the design and subsequent monitoring of mooring systems, allowing for predicting FPSO motion given incident metocean conditions and platform-specific variables such as draft and the mooring configuration itself.

The dynamic simulation requires a hydrodynamic model of the FPSO and parameterized wind and wave energy spectrums: The former is obtained by calculating hydrostatic properties of the vessel given its geometry, such as its Mass Matrix and Center of Gravity (CG) position, which is used in conjunction with the vessel's mesh to determine hydrodynamic coefficients of added mass, radiation damping, first order and mean drift wave forces for each platform draft. A hydrodynamic model then calculates the vessel's Response Amplitude Operators (RAOs), which are essentially transfer functions used along with wave energy spectrums to determine the amplitude of FPSO motion. The latter is obtained by data that comes from a hindcast model of the area, calibrated by a large number of measurements by wave radars, anemometers, wave buoys, and current meters that are either installed in the platforms or moored to the seabed.

Simulation results consist in the 6 Degrees of Freedom (DoF) time-series motion of the platform, which are measured at a North-East-Down (NED) geographic reference frame and, given the platform's heading, can be converted to the following 6 DoF (Figure 1):

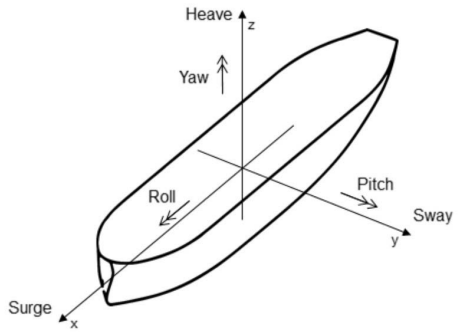

FIGURE 1. Illustration of the 6 DoF of a marine vessel.

FIGURE 2. FPSO platform at Brazilian coast (source: https://petroleohoje.editorabrasilenergia.com.br/p-50-passa-por-reparos).

1) Surge: Longitudinal motion,
2) Sway: Sideways motion,
3) Yaw: Rotation about the vertical axis,
4) Roll: Rotation about the longitudinal axis,
5) Pitch: Rotation about the transverse axis,
6) Heave: Vertical motion.

Simulations are performed on a model of a spread-moored FPSO located at the Brazilian coast similar to the platform illustrated in Figure 2, the platform's main variables are given by Table 1. Figure 3 illustrates the simulated platform as viewed in the Dynasim software interface, where mooring lines are presented in blue and risers in green.

TABLE 1. Main variables of simulated FPSO.

Variable	Value
Vessel Type	FPSO
Mooring	Spread-Moored
Length (L_{pp})	320m
Breath	54.5m
Heading	206.16°
Draft Ballasted Condition	8m
Draft Full-Load Condition	21m
Displacement Ballasted Condition	102931 ton
Displacement Full-Load Condition	302918 ton

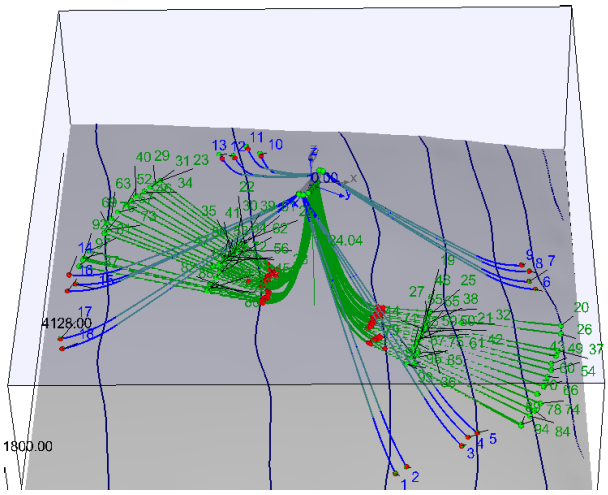

FIGURE 3. FPSO platform in Dynasim interface.

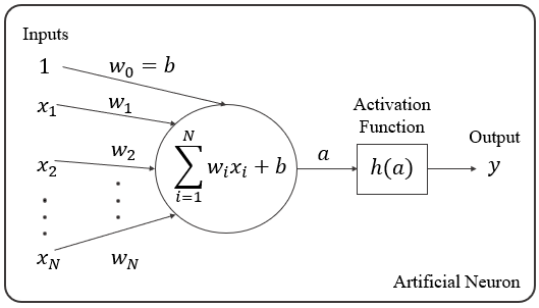

FIGURE 4. Diagram of an artificial neuron.

B. ARTIFICIAL NEURAL NETWORKS

Artificial Neural Networks (ANNs) are mathematical models inspired by biological neural networks in which neurons are interconnected in a layered structure. Their output signals are obtained by applying a nonlinear activation function to the weighted sum of their input signals. Figure 4 illustrates the model of a single artificial neuron.

Let $\mathbf{x} \in \mathbb{R}^N$ be an input vector and $f_{\mathbf{W}}(\mathbf{x})$ be a single-hidden-layer ANN parameterized by weights $\mathbf{W} = \{\mathbf{W^{(1)}}, \mathbf{W^{(2)}}\}$, where $\mathbf{W^{(1)}} \in \mathbb{R}^{M\times N}$ are the weights associated with connections from input layer to hidden layer and $\mathbf{W^{(2)}} \in \mathbb{R}^{K\times M}$ are the weights associated with connections from hidden layer to the output vector $\mathbf{y} \in \mathbb{R}^K$. For a regression MLP with activation function h in the hidden layer and a linear activation function in the output layer, the network's k-th output, y_k, is given by:

$$y_k = f_k(\mathbf{x}|\mathbf{W}) = \sum_{j=1}^{M} w_{kj}^{(2)} h\Big(\sum_{i=1}^{N} w_{ji}^{(1)} x_i + w_{j0}^{(1)}\Big) + w_{k0}^{(2)}, \quad (1)$$

where the weights $\mathbf{W^{(1)}} = \{w_{ji}^{(1)}\}$ and $\mathbf{W^{(2)}} = \{w_{kj}^{(2)}\}$ are shown in scalar form in Equation 1 in order to explicitly illustrate the calculation of output y_k.

Given a dataset $\mathbb{D} = \{\mathbf{x^{(i)}}, \mathbf{y^{(i)}}\}_{i=1}^{N_D}$ comprised of N_D samples and with known inputs $\mathbf{x^{(i)}} \in \mathbb{R}^N$ and outputs $\mathbf{y^{(i)}} \in \mathbb{R}^K$,

the process of training an ANN consists in finding the weights $\mathbf{W}$ that minimize a loss function, typically the Mean Squared Error (MSE) given by:

$$L(\mathbf{W}) = \frac{1}{N_D}\sum_{i=1}^{N_D}\left[\frac{1}{K}\sum_{k=1}^{K}\left(y_k^{(i)} - f_k(\mathbf{x^{(i)}}|\mathbf{W})\right)^2\right], \qquad (2)$$

where K is the number of output neurons and f_k is given by Eq.(1). This can be done by performing gradient descent on the loss function and updating the weights in the direction that minimize it: $\mathbf{W} \leftarrow \mathbf{W} - \alpha\nabla L(\mathbf{W})$, where α is the learning rate. The gradient of the loss function with respect to the weights $\nabla L(\mathbf{W})$ is found by back-propagation of the error through the network.

C. HYPERPARAMETER OPTIMIZATION

Hyperparameters are parameters used to control the training process itself. Instead of being learned as are the network weights, they are fixed during training and define the model's architecture or learning algorithm itself. The appropriate choice of hyperparameters is problem-specific and of fundamental importance in developing robust models with high generalization capability. Examples of ANN hyperparameters are:

- **Network architecture**: Number of hidden layers and number of neurons in each layer.
- **Activation Function**: The activation function $h : \mathbb{R} \rightarrow \mathbb{R}$ used in hidden-layer neurons.
- **Number of Epochs**: The number of times the entire training dataset is passed to the network during training.
- **Batch size**: The number of samples in each training batch used to approximate the gradient of the loss function.
- **Optimizer**: The gradient estimation algorithm implemented during training.

Hyperparameter Optimization (HO) consists of the general process of determining a model's optimal hyperparameters for a given task. Finding the optimal hyperparameters for an ML model has always been an essential yet demanding and time-consuming task. Recently, new methods have been developed to automate these tasks, significantly reducing the human effort required to optimize models [9] and thus creating the field of Automated Machine Learning (AutoML). The positive results obtained by ANNs and Deep Learning in recent years motivated the development of Neural Architecture Search (NAS), a sub-field of AutoML dedicated to optimizing ANN architectures specifically.

NAS methods can be defined by three dimensions [10]: a *search space* $\mathcal{S}_\mathcal{S}$, which bounds the possible architectures evaluated during NAS; a *performance estimation strategy*, which defines an *objective function* C used to evaluate model performance during the search process; and a *search strategy*, which determines how the algorithm explores the search space. The search space and performance estimation strategy are manually defined to obtain the best optimization performance for the shortest execution times and to ensure that the models generated by this process can generalize to unseen data.

Bayesian Optimization (BO) is one of the NAS techniques and it consists of two key components: a probabilistic surrogate model S of the objective function C; and a policy P, denoted as *acquisition function*, for selecting new parameters based on the surrogate model. In each trial, an evaluation of C updates the surrogate model S, allowing P to select a new architecture $\mathcal{M}$ most likely to achieve the objective of the optimization for the subsequent trial. The general procedure is shown in Algorithm 1.

Algorithm 1: Bayesian Optimization

Input: search space $\mathcal{S}_\mathcal{S}$, objective function C, initial architecture $\mathcal{M}_0$, surrogate model S, acquisition function P
Output: search history $\mathcal{H}$
$\mathcal{M} \leftarrow \mathcal{M}_0$;
$\mathcal{H} \leftarrow \{\langle \mathcal{M}_0, C(\mathcal{M}_0)\rangle\}$;
Initialize the surrogate model S;
for $i = 1, 2, \ldots, k-1$ **do**
  $\mathcal{M} \leftarrow \arg\max_{\mathcal{M}' \in \mathcal{S}_\mathcal{S}} P(\mathcal{M}', S)$;
  $\mathcal{H} \leftarrow \mathcal{H} \cup \{\langle \mathcal{M}, C(\mathcal{M})\rangle\}$;
  $S \leftarrow$ `updateSurrogate`$(S, \langle \mathcal{M}, C(\mathcal{M})\rangle)$;
end

The surrogate model is used to estimate the objective function C, and can be generated in several ways. Often Gaussian Process regression is used [11]. However, there has been a rise in the use of Tree-structured Parzen Estimator (TPE), as it is more flexible to non-numerical search spaces, can scale to bigger search spaces with a smaller computational cost, and has been shown to achieve better performance on some optimization problems [9], [12]. Various choices are also available for the acquisition function, the most common being Expected Improvement (EI). This method is based on selecting the model with the best estimated performance at each turn. EI is calculated using the surrogate model, and the architecture with the most significant EI is selected for evaluation in the subsequent trial. A more thorough discussion on Bayesian Optimization can be found in [11] and [12].

IV. NeuroSim ARCHITECTURE

NeuroSim is an intelligent, data-driven FPSO Dynamics Simulator based on Artificial Neural Networks with two main applications in offshore engineering:

- Mooring System Design Tool: The proposed Neural Simulator can be trained on several highly accurate simulations of centenary critical metocean conditions, which are typically time-consuming to obtain through traditional time-series dynamic simulation. After training, NeuroSim can provide design variables such as maximum FPSO roll or center of gravity offset associated with unseen conditions instantaneously, without computing the entire motion and extracting these values.

As a result, NeuroSim is a valuable auxiliary design tool as it allows for a higher number of metocean conditions to be evaluated.

- Monitoring of an Operational FPSO: On operational platforms, NeuroSim can be used to convert a batch of future expected metocean conditions into expected motion statistics. This applicability provides an extra layer of safety as critical short-term motion is predicted, and appropriate measures can be taken. In this case, NeuroSim's meta-models are continuously trained through FPSO motion already measured by IMU and GPS systems.

This section aims to illustrate NeuroSim's overall architecture, explaining its functionality both in response-based design and in seakeeping applications as well as detailing inputs and outputs of the system. We present the proposed framework first as a high-level, black box model capable of predicting FPSO motion statistics from incident environmental conditions in Figure 5. We then illustrate individual NeuroSim modules and one of the ANN models that constitute them in Figures 6 and 7.

During loading and offloading operations, FPSO tank levels go through significant variations. This process results in variations to platform load, which can be measured by the draft, the vertical distance between the waterline and the bottom of the vessel's hull, also known as the keel. An FPSO's dynamic response to incident metocean conditions is highly dependent on its draft, as it is related to fundamental properties of the vessel, such as its mass, natural period, and the position of its Center of Gravity (CG). As a result, data-driven predictive models need to account for the draft's influence in FPSO motion.

Figure 5 illustrates a diagram of the proposed NeuroSim's architecture as an extension of the ML framework presented in [13]. The current framework differs from it in three main ways:

- Consideration of different platform loads: NeuroSim is trained on FPSO motion associated with a wide spectrum of loads represented by the 8m to 21m draft range.
- Utilization of a more efficient HO method in Bayesian Optimization over the previously implemented Grid Search approach, which investigated only a small number of ANN architectures.
- Extended analysis and additional experiments: The current work presents ensemble regression results, analyzes the sensitivity of the trained models to draft variations and applies them to unseen, harsh environmental conditions expected in 1, 10, 30, 50 and 100 years periods.

The system is composed of a set of 14 individual NeuroSim Modules, each specialized in the prediction of platform dynamics when subject to a specific draft, ranging from $8m$ to $21m$. NeuroSim takes a batch of measured environmental conditions representative of 3h time periods as input. A switching mechanism then takes the measured platform draft and directs the environmental conditions to

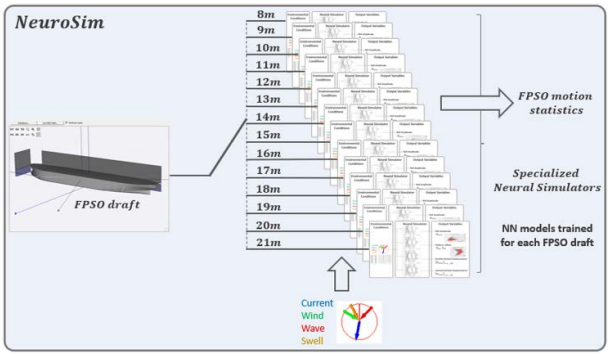

FIGURE 5. NeuroSim architecture diagram.

the appropriate NeuroSim module, which has been trained and validated to predict motion statistics given the measured draft accurately. This module provides each of its internal meta-models with the incident metocean conditions, and each of them predicts one of the four output variables that describe FPSO motion statistics: Maximum Roll, Maximum Center of Gravity Offset, and Maximum Fairlead Displacements.

This modular architecture is highly scalable, as it allows for new meta-models to be added in the future to predict other relevant variables without compromising previously trained ones. It also provides the system with more flexibility, as different modules have different hyperparameters which define their structure and help them specialize in the accurate prediction of FPSO motion for each draft. A single model appropriate for all platform drafts would not perform as well in the entire spectrum.

Figure 6 illustrates the structure of an individual draft-specific NeuroSim Module, specialized in the prediction of motion statistics given incident metocean conditions for a single given platform draft. The NeuroSim Module comprises four Neural Network meta-models, one for each of the desired output variables. During the Hyperparameter Optimization pipeline, the architectures of each meta-model are determined through Bayesian Optimization to ensure each

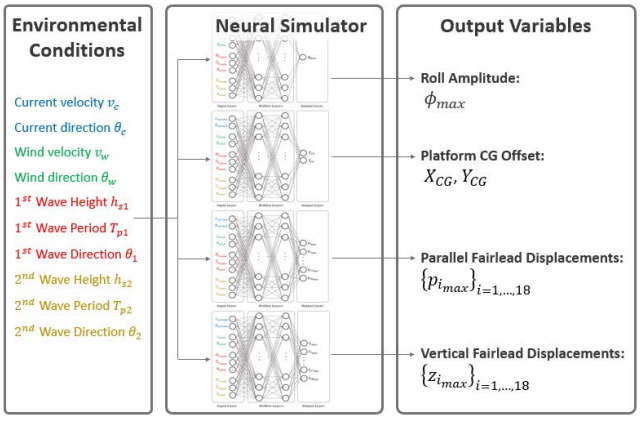

FIGURE 6. Diagram of NeuroSim Module specialized in the prediction of FPSO's dynamic responses for individual draft.

is appropriate for predicting its corresponding output variable given the module's FPSO draft.

It is crucial to notice that, while some input variables are more relevant for predicting specific output variables, the same environmental conditions are applied identically to all four meta-models. For instance, Maximum Roll Amplitude is more affected by significant wave height, period, and direction than by wind. However, the same 10 input variables described in the Data Preparation pipeline in section V-B are used in all models. This procedure is done because, during training, the Neural Network meta-models automatically learn an abstract representation of inputs related to this physical behavior, while manually changing the input variables of different meta-models through expert domain knowledge may lead to biased or inaccurate models.

Finally, a draft-specific NeuroSim Module is comprised of four individual ANNs, each designed to predict one of the four output variables: Maximum Roll, Maximum Center of Gravity Offset, and Maximum Parallel and Vertical Fairlead Displacements. Figure 7 illustrates the Multi-Layer Perceptron ANN specialized in the prediction of Maximum FPSO Offset for a given platform draft. In this work, MLPs with three hidden layers are considered and Bayesian Optimization is employed in order to determine the optimal number of neurons in each hidden layer for each of the 56 ANN models (1 module for each of the 14 drafts with 4 ANN models each).

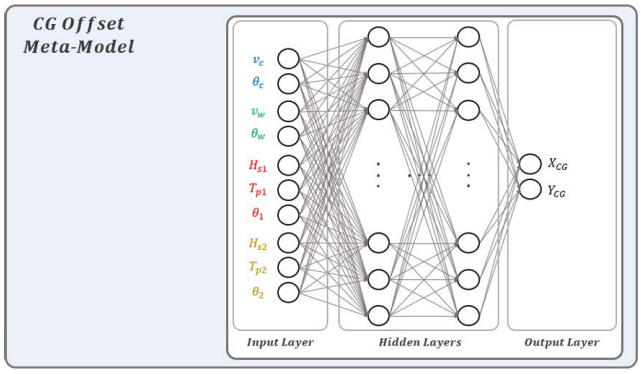

FIGURE 7. Diagram of an MLP Neural Network designed to predict the Maximum Center of Gravity Offset of the FPSO given incident metocean conditions.

V. PROPOSED METHODS

This section describes the methodology adopted in this research, indicating how the theory discussed in section III is applied to the problem of FPSO motion prediction through data-driven meta-models. The main objective of this section is to illustrate the project workflow adopted and how the NeuroSim architecture presented in Section IV is built, from the obtention of the measured environmental conditions to the preparation of the NeuroSim training dataset and validation of results. Project Workflow decomposition is highly correlated to the implemented code architecture, with each stage associated with its corresponding Kedro pipeline module.

First, Section V-A details the metocean conditions simulated. Then, Section V-B describes each of the project's pipelines following the simulation of metocean conditions on Dynasim. Finally, Sections V-C and V-D describe the implemented HO algorithm and the error metrics used to validate trained models, which are used to discuss results in section VII. Figure 8 depicts the project's workflow in its five sequential pipelines:

1) Post Processing.
2) Data Preparation.
3) Hyperparameter Optimization.
4) NeuroSim Training.
5) NeuroSim Test.

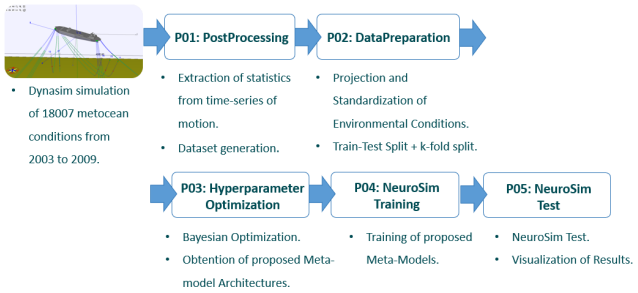

FIGURE 8. Project Workflow Diagram illustrating five pipelines: Post-processing, Data Preparation, Hyperparameter Optimization, NeuroSim Training, NeuroSim Test.

The NeuroSim Project workflow is subdivided into modular sequential pipelines not only as a way to better segment code in terms of functionality, resulting in easier debugging and less intrusive updates, but also to allow the execution of individual desired pipelines.

A. METOCEAN CONDITIONS

Petrobras Oceanography Group provided the environmental data in 3h periods from November 2003 to December 2009 at the Brazilian coast. The data comes from a hindcast model of the area, calibrated by many measurements by wave radars, anemometers, wave buoys, and current meters that are either installed in the platforms or moored to the seabed. The following variables can describe Metocean conditions:

1) **Current velocity**: Mean current velocity v_c (m/s).
2) **Current direction**: Current propagation angle θ_c (°).
3) **Wind velocity**: Mean wind velocity v_w (m/s).
4) **Wind direction**: Wind incidence angle θ_w (°).
5) **First Wave component height**: Significant wave height $H_{s1}(m)$ corresponding to highest energy wave.
6) **First Wave component period**: Peak Period $T_{p1}(s)$ corresponding to highest energy wave.
7) **First Wave component direction**: Incidence angle θ_1 (°) corresponding to highest energy wave.
8) **Second Wave component height**: Significant wave height $H_{s2}(m)$ corresponding to second highest energy wave.

TABLE 2. Samples of measured metocean conditions.

Index	v_c (m/s)	θ_c (°)	v_w (m/s)	θ_w (°)	Hs_1 (m)	Tp_1 (s)	θ_1 (°)	Hs_2 (m)	Tp_2 (s)	θ_2 (°)
1	0.11	118.33	5.47	161.1	1.73	7.10	132.5	0.61	3.74	180.5
2	0.13	133.33	7.46	186.9	1.68	7.61	152.7	1.08	5.62	198.2
.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.
18006	0.55	178.10	11.41	1.4	2.56	7.46	9.6	0.85	8.07	55.3
18007	0.54	178.88	10.21	3.9	2.54	7.35	9.1	0.00	0.00	0.0

9) **Second Wave component period**: Peak Period $T_{p2}(s)$ corresponding to second highest energy wave.
10) **Second Wave component direction**: Incidence angle θ_2 (°) corresponding to second highest energy wave.

Table 2 shows samples of the measured metocean data and the corresponding values of each input variable. Due to sensitive data restrictions, the complete dataset comprised of all 3h periods used in this work cannot be fully disclosed. It is important to notice that two wave components were chosen as input variables in addition to wind and current variable since sea states on the Brazilian coast are typically bimodal. Applying the same framework for FPSOs in different locations may be simplified to include a single wave component without significant performance loss.

B. PROJECT WORKFLOW

1) P01: POST-PROCESSING

After Dynasim simulation of the 18k measured metocean conditions and storage of the corresponding FPSO dynamic responses in h5 files, the Post-processing pipeline performs time-series analysis and stores the desired motion statistics in a csv file.

Figure 9 illustrates 20 minutes of the roll angle time-series obtained from the Dynasim simulation of environmental condition 2680. During the first seconds, the resulting motion is highly dependent on initial configuration, while subsequent dynamics are governed by the incident environmental conditions. In order to isolate the effects of environmental conditions, a cutoff time t_{cutoff} of 3600s was implemented,

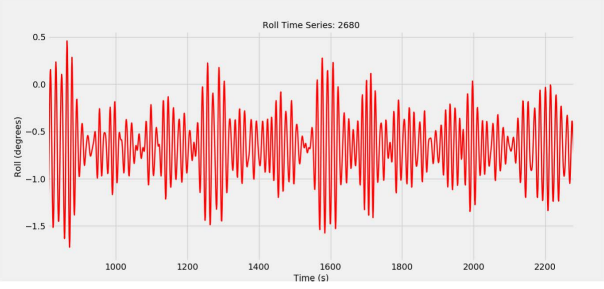

FIGURE 9. Roll angle time-series corresponding to simulation 2680 obtained from dynasim.

and this pipeline analyzes the remaining 6 DoF time-series to extract the following output variables:

- **Roll Amplitude** ϕ_{max}: Maximum absolute roll angle given by:

$$\phi_{max} = \max_{t>t_{cutoff}} |\phi(t) - \phi_{eq}| \tag{3}$$

where $\phi(t)$ is the roll motion time-series of the vessel and ϕ_{eq} is the equilibrium position of the roll degree of freedom, given by the vessel's average roll angle when no incident environmental conditions are present.

- **Platform Offset** X_{CG}, Y_{CG}: Global X and Y positions of the platform's center of gravity associated with maximum observed offset:

$$\begin{cases} X_{CG} = X(t*) \\ Y_{CG} = Y(t*), \end{cases} \quad \text{where}$$
$$t^* = \underset{t>t_{cutoff}}{\arg\max} \left\| (X(t), Y(t)) - (X_{eq}, Y_{eq}) \right\|_2 . \tag{4}$$

$(X(t), Y(t))$ and (X_{eq}, Y_{eq}) describe the time-series of the FPSO's center of gravity over time and its equilibrium position when it is subject to no environmental conditions respectively, measured on a global North-South/East-West reference frame.

- **Maximum Parallel Fairlead Displacements** $p_{i_{max}}$: Maximum parallel displacements of each fairlead in the direction parallel to the XY-projection of the corresponding mooring line during equilibrium, as illustrated in 10.

$$p_{i_{max}} = \max_{t>t_{cutoff}} p_i(t) \tag{5}$$

where $p_i(t)$ is the time-series motion of the i-th fairlead projected on the vector $\overrightarrow{v}_{\mathbf{a_i},\mathbf{f_i}} = \mathbf{f_{i,eq}} - \mathbf{a_i}$, with $\mathbf{f_{i,eq}}$ and $\mathbf{a_i}$ denoting the equilibrium position of the i-th fairlead and the fixed position of the corresponding anchor on the seabed respectively.

- **Maximum Vertical Fairlead Displacements** $z_{i_{max}}$: Maximum vertical displacements of each fairlead, as illustrated in Figure 10, with

$$z_{i_{max}} = \max_{t>t_{cutoff}} z_i(t), \tag{6}$$

where $z_i(t)$ is the time-series motion of the i-th fairlead projected on the global vertical axis Z.

Figure 10 shows a p_i, w_i, z_i reference frame in which Maximum Fairlead Displacements are determined. Fairlead

motion in this reference frame is obtained by applying a time dependent transformation of coordinate frames to the 6 DoF $\{X(t), Y(t), Z(t), \phi(t), \theta(t), \psi(t)\}$ time-series generated by Dynasim simulation at each time step.

The product of pipeline P01: Post-processing is a csv file for each platform draft from 8m to 21m in which environmental conditions and the corresponding four variables above that represent the resulting FPSO motion are stored. Table 3 illustrates the output variables obtained in Pipeline P01 for the environmental conditions in Table 2.

TABLE 3. Samples of output variables obtained through Pipeline P01.

Index	ϕ_{max}	X_{CG}	Y_{CG}	$p1_{max}$	$z1_{max}$	...
1	0.118°	-3.527m	4.235m	4.218m	0.389m	...
2	0.108°	-0.012m	4.490m	3.866m	0.366m	...
⋮	⋮	⋮	⋮	⋮	⋮	...
18006	0.119°	-2.389m	-19.157m	-9.645m	0.357m	...
18007	0.121°	-2.409m	-17.552m	-8.712m	0.240m	...

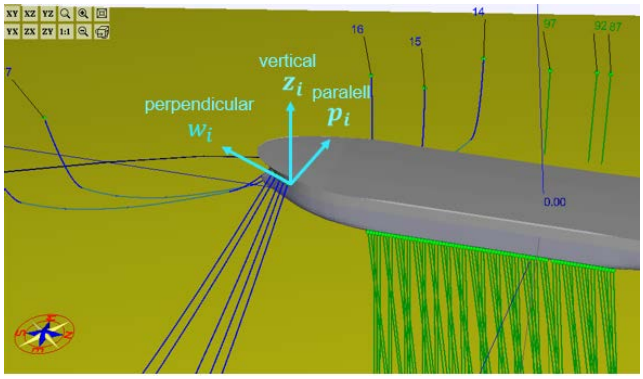

FIGURE 10. Visualization of a local $\{p_i, w_i, z_i\}$ fairlead reference frame.

2) P02: DATA PREPARATION

In order to improve the numerical convergence of learning algorithms, several data preparation techniques are applied to the data prior to training. Let $\mathbf{e} = (v_c, \theta_c, v_w, \theta_w, H_{s1}, T_{p1}, \theta_1, H_{s2}, T_{p2}, \theta_2)$ denote a measured metocean condition as described in Section V-A. As angular variables are defined in $[0°, 360°]$, their periodic property implies that values such as 0.1° and 359.9° are functionally close despite being numerically distant. This discrepancy can cause slow convergence of the ANN as similar seastates may be far apart in the network input space. As a result, the projections of current velocity, wind velocity, and significant wave height in the N-S and E-W directions were used, rather than their magnitude and incidence angle, so that the same seastate can be represented as:

$$\begin{aligned}\mathbf{e^{proj}} = (&v_c \sin(\theta_c), v_c \cos(\theta_c), \ v_w \sin(\theta_w), v_w \cos(\theta_w),\\ &H_{s1} \sin(\theta_1), H_{s1} \cos(\theta_1), T_{p1},\\ &H_{s2} \sin(\theta_2), H_{s2} \cos(\theta_2), T_{p2}).\end{aligned}$$

This process, known as the projection of environmental conditions, is illustrated in Figure 11. Since the values of different input variables have different orders of magnitude, e.g., local wind velocity can be as high as 20 m/s while current velocity is lower than 1 m/s, a Gaussian Standardization method was applied to transform the projected environmental conditions into the actual model's input data:

$$x_i = \frac{e_i^{proj} - \mu_i}{\sigma_i}, \quad i = 1, \ldots, 10,$$

where μ_i and σ_i are the mean and standard deviation of the i-th variable on the complete dataset. This process scales input variables and improves numerical convergence.

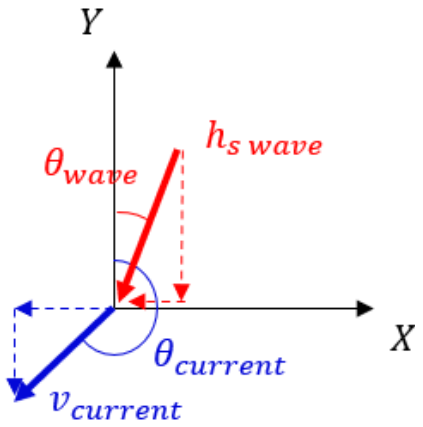

FIGURE 11. Visualization of Environmental Condition Projection for incident current (blue) and wave (red).

3) P03: HYPERPARAMETER OPTIMIZATION

The third pipeline is responsible for determining the optimal Neural Network Architectures for each meta-model and each FPSO draft through a Hyperparameter Optimization algorithm known as Bayesian Optimization (BO), described in Section V-C. In this project, BO was applied to determine the optimal number of neurons in each of the three ANN hidden layers, while remaining hyperparameters such as activation function, optimizer, batch size, and the number of training epochs were kept fixed at appropriate values found through trial and error and expert knowledge [13].

4) P04: NEUROSIM TRAINING

Once the optimal model architectures, defined by the proposed set of hyperparameters, have been determined extensive training with 90% of the entire dataset is performed, and the trained models are saved for testing in the following pipeline. During this training, the Train+Validation dataset was used, and proposed neural networks described in Tables 9 and 10 were trained for 5000 epochs as opposed to the 200 epochs used during Hyperparameter Optimization.

5) P05: NEUROSIM TEST

Finally, the proposed models for each desired output variable and each FPSO draft are tested in 10% of the original dataset, which is comprised exclusively of unseen metocean conditions. This test ensures the resulting metrics are representative of the proposed models' generalization performance when applied to realistic scenarios where measured environmental conditions were not used during training and validation.

The main objective of this pipeline is to generate analytic results to verify NeuroSim's performance, comparing error metrics with those of traditional simulation methods and validating the proposed approach for a data-driven FPSO dynamics simulator. These results are visualized numerically and graphically, as discussed in Section VI.

C. NEURAL ARCHITECTURE SEARCH

This section aims to detail the Hyperparameter Optimization algorithm implemented in pipeline P03. Initially, a preliminary study indicated appropriate values of hyperparameters such as activation function, batch size, and the number of training epochs [13], [15], showing that an optimization of the ANN architecture (NAS) presented the highest potential for model performance improvement. As a result, subsequent research focused on NAS methods in order to determine the optimal number of neurons in each hidden layer of the final ANN models [14], where the following methods were compared for the CG Offset meta-model:

- Random Search (RS)
- Simulated Annealing (SA)
- Bayesian Optimization (BO)

Figure 12 illustrates a comparison of the three investigated NAS methods in [14], showing BO and RS performed a more comprehensive search of the proposed search space. At the same time, SA's initial architecture tends to impact evaluated architectures over its optimization trials. Overall, BO yielded the best models and was ultimately chosen as the NAS method for this work. In Section VI-B we detail the adopted procedure.

D. ERROR METRICS

The definition of precise error metrics allows for parameter tuning during training, the numerical comparison of different ML models during validation, and indicates expected error ranges for the final models during testing. As presented in Section III-C, error metrics are also used to define the Objective Function used in BO. As a result, despite the insights provided by visualization tools, error metrics remain of paramount importance and must be precisely defined. In this work, meta-model performance is evaluated by the following error metrics:

- **MSE**: The Mean Squared Error is obtained by averaging the squared differences between model predictions ($\hat{y}_i$) and true values (y_i): $MSE = \frac{1}{N_D}\sum_{i=1}^{N_D}(y_i - \hat{y}_i)^2$
- **RMSE**: The Root Mean Squared Deviation is analogous to the standard deviation and is obtained by taking the square root of the Mean Squared Error. It has the same unit as the output variable: $RMSE = \sqrt{MSE}$
- **MAE**: The Mean Absolute Error is the average of the absolute differences between true and predicted values and has the same unit as the output variable: $MAE = \frac{1}{N_D}\sum_{i=1}^{N_D}|y_i - \hat{y}_i|$
- **Max Error**: Overall maximum error observed over test set, highly sensitive to outliers: $MaxError = \max_{i=1,\ldots,N_D}|y_i - \hat{y}_i|$.

VI. EXPERIMENTS

The main objective of this section is to describe the experiments performed according to the Methodology illustrated in Section V-B. Firstly, Section VI-A details the dataset generation process performed in pipelines P01 and P02, specifying the implemented train-validation-test split. Then, Section VI-B illustrates the results of pipeline P03, namely the optimal ANN architectures for each of the 4 meta-models and 14 different platform drafts, while Section VI-D provides a local and global analysis of model sensitivity to draft variations.

A. DATASET GENERATION

The motion of a spread-moored FPSO with 18 mooring lines was simulated in Dynasim subject to each metocean condition and for the 14 draft values between $8m$ and $21m$. The generated time series were then processed according to the process described in Section V-B for pipeline P01. For each metocean condition, the corresponding 4 output variables of interest were extracted: Maximum Roll Angle, CG Offset, and Maximum Parallel and Vertical Fairlead Displacements.

Before constituting the meta-models Dataset, the output variables were plotted as a function of the most relevant environmental variables to check for potential errors in the simulation process and verify cohesion in terms of the underlying physical phenomena. Figures 13 and 14 illustrate, for all metocean conditions, the maximum observed roll angle and CG offset, respectively. It is possible to notice that Roll Amplitude tends to increase as significant wave height increases and as its period approaches the FPSO natural period, while the two peaks in the direction plot correspond to perpendicularly incident waves relative to the platform's heading of 210°. In the CG Offset plot, higher offset values are observed in the third quadrant (South-East), which is compatible with typical metocean conditions.

After the extraction of output variables and preliminary analysis of their values, metocean conditions are projected and normalized according to the process described in pipeline P02 (Section V-B). This procedure results in 14 different datasets, one for each platform draft, with 18007 metocean conditions and the corresponding motion statistics of interest obtained through dynamic simulation. In order to appropriately evaluate the performance of trained models in the upcoming stages of the project, this dataset was split into a Training+Validation set (used for 5-fold

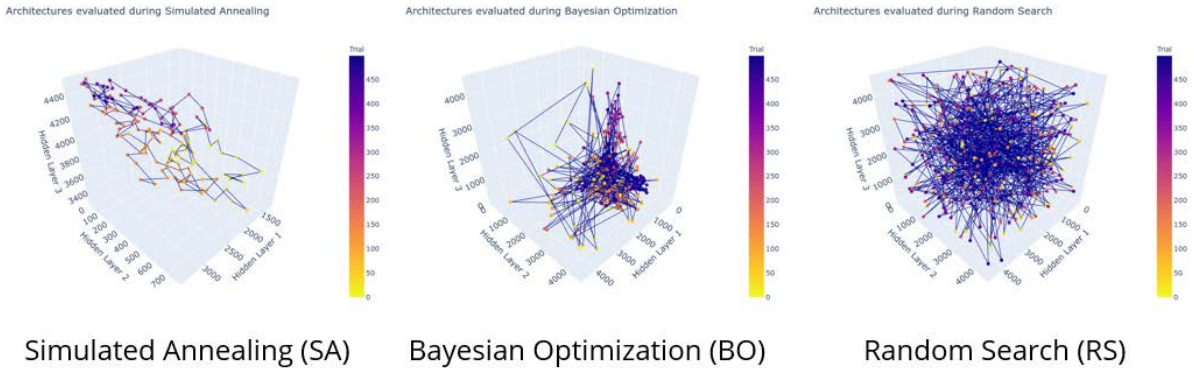

FIGURE 12. Visualization of evaluated ANN architectures over optimization trials (colorbar) for three different NAS methods: SA (left), BO (center) and RS (right). The 3 axis correspond to the number of neurons in each ANN Hidden Layer.

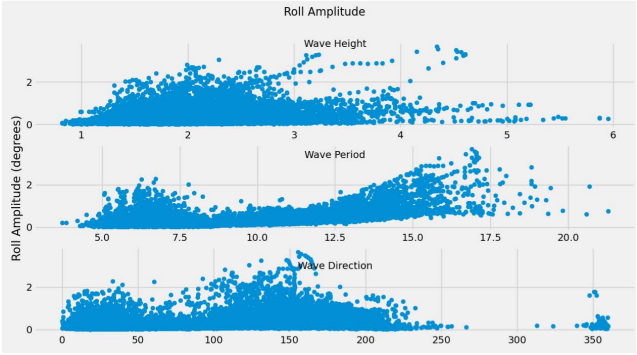

FIGURE 13. Visualization of Roll Amplitude values as a function of significant wave height (top), wave peak period (middle) and wave direction (bottom).

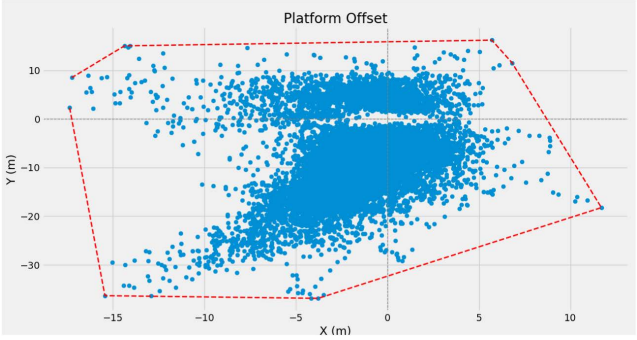

FIGURE 14. Visualization of CG Offset values for all 18k environmental conditions in global X (East-West) and Y (North-South) axis.

Cross-Validation) and a separate Test set used exclusively to assess the performance of the final models (Section VI-C). Figure 15 illustrates the implemented dataset split, indicating the number of samples in each set.

B. BAYESIAN OPTIMIZATION

After processing the inputs through projection and standardization and splitting the Dataset into Training, Validation, and Test sets, NAS methods were used to determine the

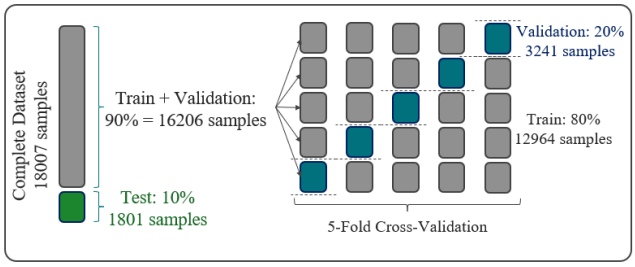

FIGURE 15. Diagram of dataset split for 5-fold cross-validation.

optimal meta-model architectures for each output variable and platform draft. As each of the 14 draft-specific NeuroSim modules contains 4 ANNs, each one responsible for predicting one of the four relevant motion statistics, a total of 56 ANN architectures were optimized in pipeline P03.

Initially, a simple Grid Search algorithm was used, in which a reduced number of candidate architectures are trained, and the model with minimum Cross-Validation MSE was chosen [13]. This approach, however, is highly time-consuming and evaluates only a small number of architectures. Then, Simulated Annealing (SA) was implemented to encourage the exploration of new architectures during the initial trials and convergence to optimal regions of the search space. SA presented better results, exploring better quality ANN architectures overall, but with highly correlated to the initial random architecture (Figure 12).

Finally, Bayesian Optimization (BO) was executed through the Python Optuna library. It solved the issue of exploring only ANN architectures close to the initial random model. The first 10 trials are randomly generated and subsequent architectures explored are not necessarily close to each other but correspond to regions of the search space in which the Expected Improvement (EI) is maximized, given the current approximation of the objective function (section V-C). It is possible to notice that ANN architectures vary considerably with platform draft and meta-model output variable, section VI-D explores the sensitivity of each model to draft variations.

TABLE 4. Summary of Hyperparameters used in Bayesian optimization. HL 1-3 denote the number of neurons in Hidden Layers 1-3. *: Parallel FD meta-model exceptionally uses a sigmoid activation Function instead of ReLU due to better observed results [13].

Hyperparameter	Value
Number of Optimization Trials	500
Objective Function	Average Cross-Validation MSE
Number of Hidden Layers	3
Number of Training Epochs	200
Batch Size	800
Optimizer	Adam
Learning Rate	0.001
Activation Function	ReLU*
HL 1	Optimized through BO
HL 2	Optimized through BO
HL 3	Optimized through BO

Since BO showed better results, yielding ANN architectures that presented smaller Cross-Validation MSEs in reduced computational time [14], it was chosen as the main NAS method for pipeline P03 in this work. It is important to notice that, while BO has been applied in this research to optimize exclusively the number of neurons in each of the three hidden layers of the ANN models, it is a general optimization algorithm which can be applied to the optimization of any other hyperparameters. Ma *et al.* [16] have successfully applied BO in the problem of NAS for Convolutional Neural Networks (CNNs), which are significantly more complex than MLPs. Additionally, there has been significant effort in recent work to determine the importance of different hyperparameters in NAS for MLPs, allowing for the reduction of the search space and more efficient algorithms [17].

Table 4 summarizes parameter values used for Bayesian Optimization. An initial analysis of the model's training curves indicated that 200 training epochs during HO is an appropriate value, providing representative measurements of a model's performance without leading to unfeasible training times. Hyperparameters such as batch size, optimizer, and activation functions were found in [13]. After 500 BO trials, optimal ANN architectures were found for each model and platform draft, as illustrated in Tables 9 and 10 in Appendix VIII.

C. NeuroSim TEST

Once the optimal ANN architectures were found, the corresponding models are trained with the Train+Validation dataset (Figure 15) for 5000 training epochs, and their performance is evaluated using the proposed metrics (section V-D) applied to the Test dataset, comprised of unseen metocean conditions.

1) MAXIMUM ROLL

The Maximum Roll meta-model's performance is illustrated in Figure 16, which presents three scatter plots of Roll Amplitude as a function of significant wave height, period, and direction, respectively. In each plot, individual points represent metocean conditions, values predicted by Dynasim are shown in blue while NeuroSim predictions are shown in orange.

In order to better visualize error regions as a function of the same set of input variables, Figure 17 illustrates a colormap of the associated absolute error between Dynasim and NeuroSim. In this plot, red indicates higher errors (higher than 0.20°) while blue indicates prediction errors in the range of 0° to 0.1°. An analysis of NeuroSim's error plots for the 14m draft Roll Amplitude meta-model leads to two immediate conclusions:

- Good performance on critical cases: All metocean conditions which resulted in maximum platform roll of over 2.0° were accurately predicted with NeuroSim. This result is positive, as highly accurate predictions of roll motion for calm conditions are not as relevant in mooring system design.
- Localized error spikes: NeuroSim's Roll Amplitude meta-model's highest errors were observed on a minimal number of incident metocean conditions. In other words, there is a large gap between the first and second highest errors. The condition which presented the highest error corresponds to a 2.5m high-frequency wave with an incidence direction parallel to the platform's main axis. In this case, NeuroSim's predicted roll was higher than the actual maximum roll measured by Dynasim. This result is preferable to having multiple conditions with poor prediction performance, as higher errors tend to be highly uncommon.

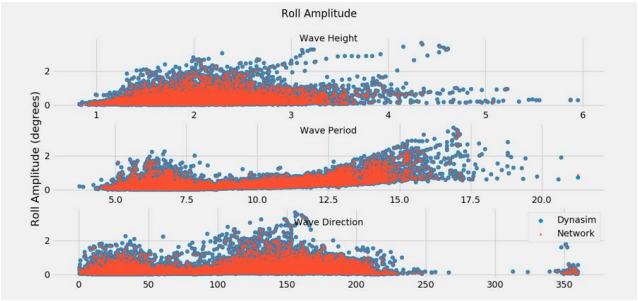

FIGURE 16. Roll amplitude meta-model results as a function of significant wave height (top), wave peak period (middle) and wave direction (bottom) for 14m FPSO draft. Comparison between NeuroSim meta-model (orange) and Dynasim values (blue).

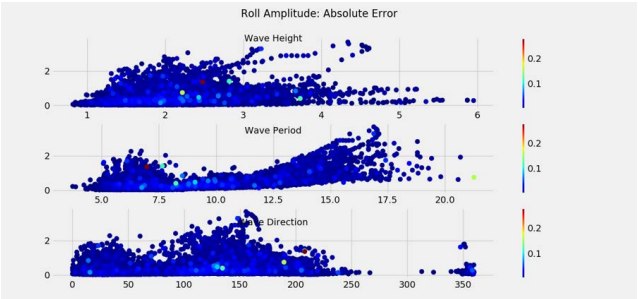

FIGURE 17. NeuroSim roll amplitude meta-model error plot.

2) MAXIMUM CENTER OF GRAVITY OFFSET

Similarly to the first meta-model, NeuroSim's CG Offset prediction results for an FPSO draft of 14m are plotted in Figures 18 and 19. The former illustrates Dynasim and NeuroSim predicted values for the X and Y position of the maximum center of gravity offset, while the latter displays a colormap of the prediction error. The convex hull that encapsulates all offsets across the entire dataset is represented as the dotted red line in both figures.

NeuroSim's CG Offset meta-model presented relatively low absolute errors in cases where the offset was more significant, which tend to occur when incident currents present higher velocity. This result is a positive outcome similar to the Roll Amplitude meta-model, where accurate predictions occur in critical cases. One possible explanation for this behavior is that the neural network models find it more challenging to correctly predict the desired variables in calm sea states as similar environmental conditions may lead to more varied dynamic responses, while critical conditions may have a narrower range of responses. This hypothesis is supported by Figure 19 where points closer to the origin tend to display more significant absolute errors ($>3m$).

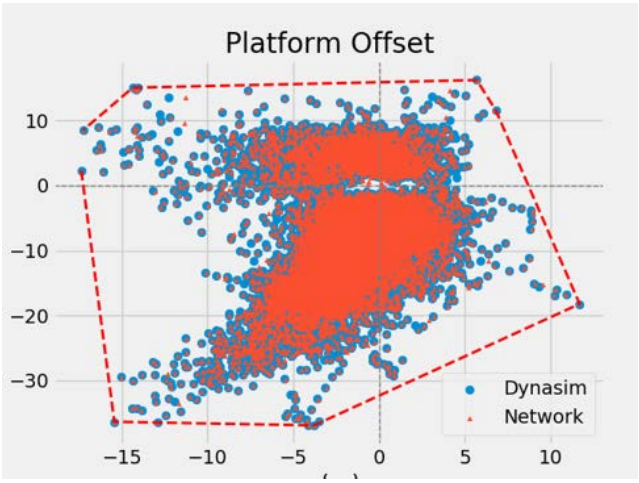

FIGURE 18. CG Offset meta-model results at global XY reference frame for 14m FPSO draft. Comparison between NeuroSim meta-model (orange) and Dynasim values (blue).

Overall, NeuroSim's CG Offset meta-model's error is within an acceptable range for almost all of the metocean conditions tested. Error spikes of 8m are observed for only two conditions in less densely populated areas of the plot, indicating the model's error is due to insufficient similar training samples and resulting generalization issues in these regions.

3) MAXIMUM PARALLEL FAIRLEAD DISPLACEMENTS

The last two meta-models, maximum parallel and vertical fairlead displacements have predicted variables of higher dimension than the previous two. Both outputs are 18-dimensional, as they contain numerical values for the maximum displacements of each of the 18 fairleads in the

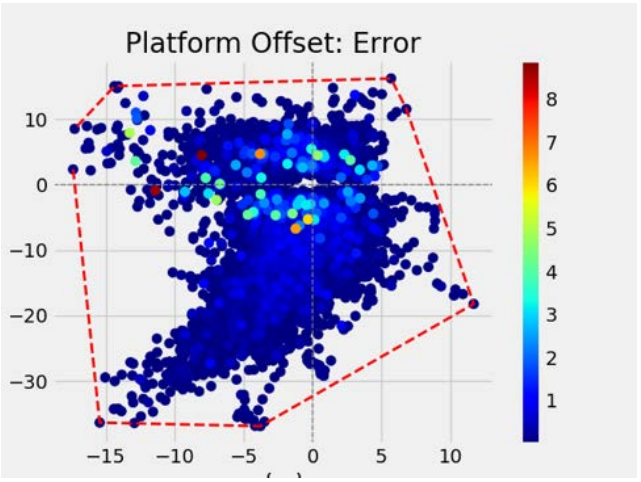

FIGURE 19. NeuroSim CG Offset meta-model error plot.

corresponding local reference frames. This result leads to visualization issues when trying to plot NeuroSim's performance similarly to previous meta-models. As a result, the Parallel Fairlead Displacement prediction results are plotted in terms of the average value among mooring lines in a given group. The 18 mooring lines can be divided into four different groups:

- Group 1: Lines 1-5 (bow-port)
- Group 2: Lines 6-9 (stern-port)
- Group 3: lines 10-13 (stern-starboard)
- Group 4: Lines 14-18 (bow-starboard)

Figure 20 contains four 3D scatter plots of the group-average maximum parallel fairlead displacements obtained through Dynasim simulation (blue) and predicted by NeuroSim (orange). The output variable is plotted as a function of the most relevant inputs, current velocity, and direction for the more straightforward physical interpretation of results. It is worth noting that taking the average maximum displacements among line groups is a good approach for visualization as the output variable presents slight variations inside each group.

As expected, the typical local seastates and the resulting FPSO dynamic motion are reflected in the overall behavior of the Parallel Fairlead Displacement (FD) meta-model for each line group. Groups 2 and 3 (stern) tend to display mostly positive values of maximum parallel FD, while groups 1 and 4 show negative output variable values. This behavior is appropriate given typical local seastates and the FPSO heading. Additionally, NeuroSim presented predictions with small errors when compared to Dynasim. These results are numerically analyzed in chapter VII.

4) MAXIMUM VERTICAL FAIRLEAD DISPLACEMENTS

Finally, the performance of the final meta-model is evaluated as the previous one. However, since no group-specific behavior was observed, the maximum vertical fairlead displacements (FD) are plotted as an average across all 18 lines, as a

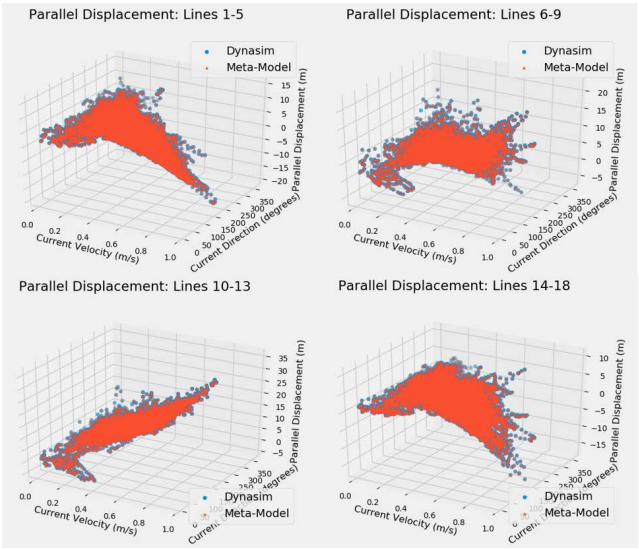

FIGURE 20. Maximum parallel fairlead displacement meta-model results for each line group as a function of current velocity and direction for 14m FPSO draft.

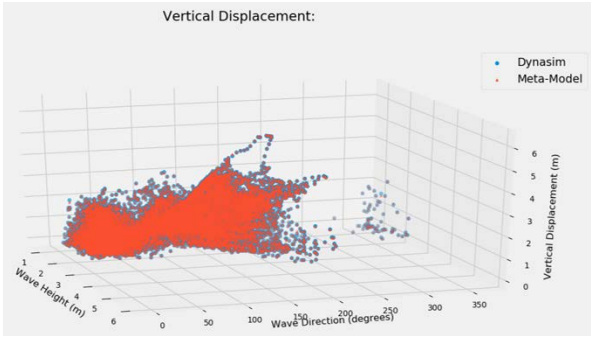

FIGURE 21. Maximum vertical fairlead displacements meta-model results as a function of significant wave height and direction for 1m draft. Comparison between NeuroSim meta-model (orange) and Dynasim values (blue).

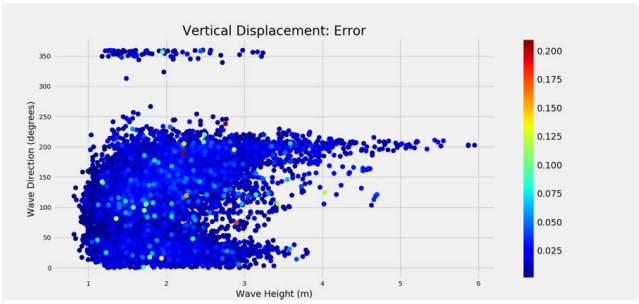

FIGURE 22. NeuroSim maximum vertical fairlead displacements meta-model error plot.

function of the environmental variables it is most sensitive to, significant wave height and direction (Figure 21).

Similar to CG Offset prediction, error spikes were observed on a small number of metocean conditions while most conditions were predicted with high accuracy. Unlike the first meta-models, however, NeuroSim's prediction of maximum vertical FD presented no regions of error concentration in the input space, even in plots of different combinations of input variables. A positive result is that even the maximum observed test error (approx. $0.2m$) is sufficiently small to allow for accurate line tension estimation in the future.

D. DRAFT SENSITIVITY ANALYSIS

Platform draft, similar to metocean conditions, is obtained through a model that receives several sensor measurements. As a result, there is an associated uncertainty with draft values, and measured motion may correspond to a different actual platform draft. Given the switching mechanism present in NeuroSim's architecture (section IV), this might lead to incorrect predictions made on the assumption that the FPSO had a different load, and therefore different dynamic response, at the time the metocean condition was measured. Motivated by these issues, NeuroSim's draft sensitivity was analyzed, both locally (section VI-D1) for small draft variations, and globally (section VI-D2), where each meta-model's predictions are compared to observed output variables for all 14 considered platform drafts.

1) LOCAL SENSITIVITY ANALYSIS

During operation, the most common platform draft for the studied FPSO is 14m, as extreme values only occur directly after loading or offloading. This analysis aims to evaluate the performance of the 14m meta-models when predicted roll amplitude and offset are compared to values obtained from simulations of the FPSO subject to the same metocean conditions but under 13m and 15m draft. As draft measurement errors higher than 1m are uncommon, this analysis can verify whether trained meta-models remain accurate and how specialized they are in the draft they were trained on.

Figure 23 illustrates the sensitivity of the 14m draft Roll Amplitude meta-model to 1m draft variations. An error scaling process can be observed, in which specific metocean conditions with the highest errors in the center column plots had their errors amplified and remained the highest error conditions when the model was applied to predict the motion of a 13m draft FPSO (left column). However, metocean conditions, which lead to higher than average roll amplitudes (such as waves with peak period close to 16s), had their errors significantly amplified when the 14m model predictions were compared to the 15m dataset (right column). Interestingly, this leads to an asymmetrical sensitivity to draft, where the 14m Roll Amplitude meta-model seems to be better suited to predict FPSO roll motion when subject to an actual draft of 13m rather than 15m.

Similarly, the 14m draft CG Offset meta-model's sensitivity to draft (Figure 24) presented the same error scaling behavior, in which a few problematic metocean conditions remain with high errors when the model is applied to the 13m and 15m datasets. However, the model presented a generalized increase in the errors of predictions with small offset values close to the center, preserving the accuracy of

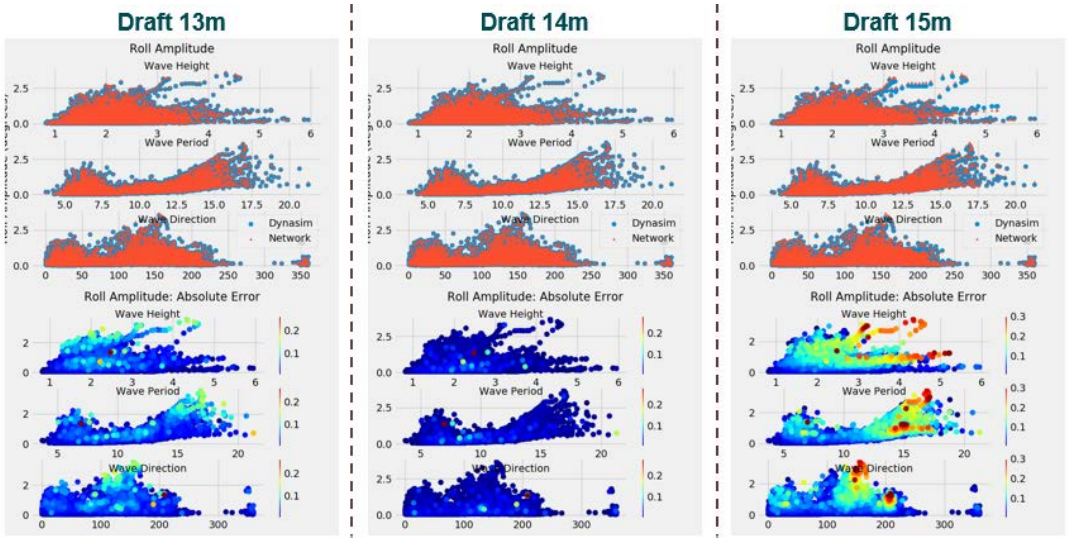

FIGURE 23. Local sensitivity analysis of 14m draft roll amplitude meta-model when compared to actual drafts of 13m (left) and 15m (right). Plots are defined as Figure 16.

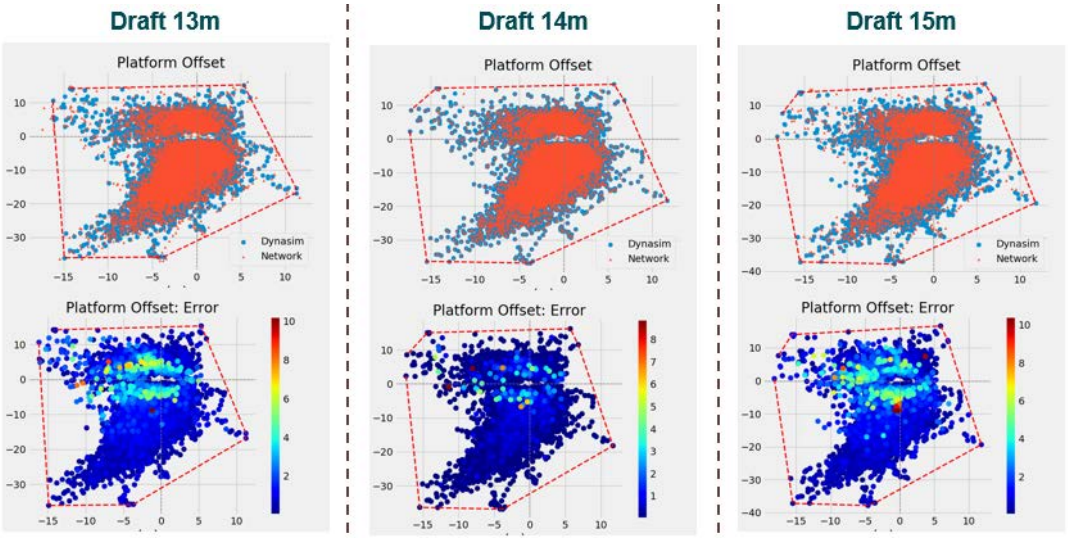

FIGURE 24. Local sensitivity analysis of 14m draft roll amplitude meta-model when compared to actual drafts of 13m (left) and 15m (right). Plots are defined as Figure 18.

predictions with higher offset values. This result is a positive outcome since such metocean conditions tend to be the most extreme and relevant during response-based design, as they are more likely to lead to mooring line breakage.

2) GLOBAL SENSITIVITY ANALYSIS

The sensitivity analysis presented above is limited to the evaluation of the 14m draft model and only to variations of 1m in draft measurements. While providing valuable insights, the strictly visual interpretation of results should be complemented with a numerical, quantitative evaluation. The objective of this section is to investigate the performance of each of the 14 meta-models when applied to the 14 datasets corresponding to observed FPSO motion statistics when subject to each draft. By assigning an error metric such as the test Mean Absolute Error (MAE) to each combination of models (columns) and datasets (rows), we can construct a Draft Sensitivity Matrix (DSM) that compactly presents the expected meta-model performance for all possible drafts.

Figure 25 illustrates the Roll Amplitude meta-model DSM. The left plot presents the MAE values associated with applying each draft-specific meta-model to each of the 14 datasets. In contrast, the right plot identifies the combinations which result in errors smaller than 10% of the average Roll Amplitude across the entire dataset (0.0366°). This threshold is applied to verify which models allow for draft measurements without significantly compromising prediction

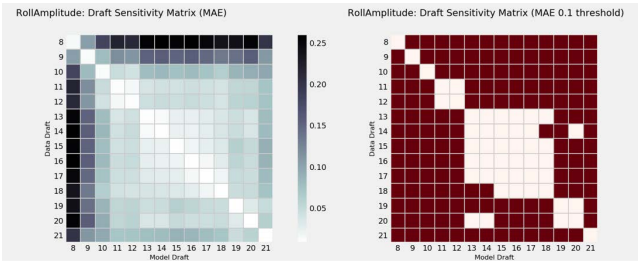

FIGURE 25. Roll amplitude meta-model MAE DSM (left) where columns and rows correspond to draft-specific ANN models and datasets respectively. Error threshold DSM (right) indicating in red combinations where the MAE exceeds 10% of the average Roll Amplitude (0.0366°).

accuracy. We notice that ANN models trained to predict maximum roll for drafts of 8m and 9m tend to perform worst when their predictions are compared to the maximum roll observed for an FPSO subject to different drafts. However, the 13m to 18m meta-models are more robust to draft measurement errors and display a wide range of MAE values under the specified threshold.

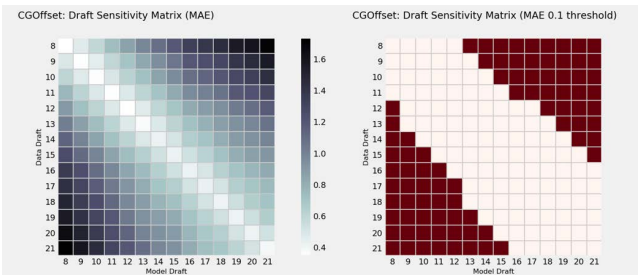

FIGURE 26. CG Offset meta-model MAE DSM (left) where columns and rows correspond to draft-specific ANN models and datasets respectively. Error threshold DSM (right) indicating in red combinations where the MAE exceeds 10% of the average CG offset (0.8942*m*).

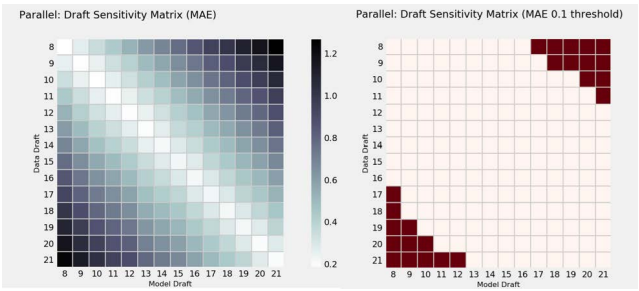

FIGURE 27. Parallel meta-model MAE DSM (left) where columns and rows correspond to draft-specific ANN models and datasets respectively. Error Threshold DSM (right) indicating in red combinations where the MAE exceeds 10% of the average parallel fairlead displacement (0.8911*m*).

Similarly, Figures 26, 27 and 28 exhibit the MAE DSM and the corresponding Error Threshold DSM for the CG Offset, Parallel and Vertical Fairlead Displacement meta-models. We notice the sensitivity errors are more symmetrically distributed when compared to the Roll Amplitude meta-model, which means, for example, that the MAE observed when

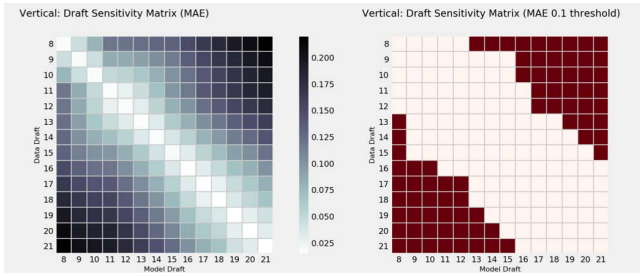

FIGURE 28. Vertical meta-model MAE DSM (left) where columns and rows correspond to draft-specific ANN models and datasets respectively. Error threshold DSM (right) indicating in red combinations where the MAE exceeds 10% of the average vertical fairlead displacement (0.1213*m*).

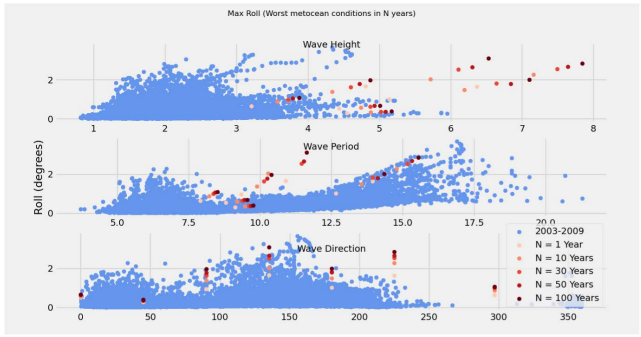

FIGURE 29. Maximum roll as a function of significant wave height (top), wave peak period (middle) and wave direction (bottom) for 14m FPSO draft. Comparison between 2003-2009 dataset (blue) and extreme environmental conditions (red) in 1, 10, 30, 50 and 100 years.

comparing the 12m ANN model predictions with the 14m dataset is similar to the MAE observed when comparing the 14m ANN model with the 12m dataset. A positive result is that small MAE values are obtained across the main diagonal of the DSM, which corresponds to cases where the ANN model predictions are compared to FPSO motion subject to the same draft. The Parallel and Vertical Fairlead Displacement meta-models presented wider ranges of acceptable draft measurement error, indicating models with higher flexibility to draft errors.

E. EXTREME ENVIRONMENTAL CONDITIONS

In Mooring System Design it is common to analyze typical local environmental conditions in order to generate statistical worst-case scenario estimates expected in the future. For a 100 year period these are known as centenary environmental conditions. In order to analyze the performance of NeuroSim in such conditions, Dynasim simulations were performed for a set of the seven worst-case scenario environmental conditions expected in 1, 10, 30, 50 and 100 years time periods was simulated in Dynasim. Each environmental condition corresponds to current, wind and wave aligned in one of the seven directions: N, NE, E, SE, S, SW, WNW, resulting in 35 total simulations. Figure 29 illustrates the most extreme environmental conditions expected in the future in comparison to the original 2003-2009 dataset used to train NeuroSim.

Table 5 illustrates the numerical error metrics obtained by NeuroSim when applied to the set of extreme environmental conditions for the Roll Amplitude meta-model. Both the Mean Absolute Error and the Maximum Error increase as the time period considered increases. Errors increase substantially even for the 1 year time period due to the lack of similar environmental conditions in the 2003-2009 dataset with which NeuroSim was trained. As presented in VII, NeuroSim achieves good performance in the test dataset comprised of samples from the 2003-2009 time period in which it was trained. As a result, the addition of extreme environmental conditions in the training set would increase its performance in the prediction of other extreme conditions. The application of the proposed models directly in the prediction of extremely harsh metocean conditions without retraining is not advised.

TABLE 5. NeuroSim's roll amplitude error metrics for the set of expected extreme future environmental conditions.

Roll Amplitude Test Results		
	MAE	Max Error
2003-2009 Dataset	0.0099°	0.2512°
Roll Amplitude Results: Extreme Environmental Conditions		
Future time period (N)	MAE	Max Error
1 Year	0.240°	0.510°
10 Years	0.394°	0.840°
30 Years	0.437°	0.675°
50 Years	0.458°	0.716°
100 Years	0.549°	0.944°

VII. RESULTS AND DISCUSSION

Previous sections presented the theoretical concepts behind this research, displayed our proposal, detailed the adopted methodology, and illustrated the experiments performed in chronological order from the obtention of data to the analysis of trained and validated models. This section aims to expose a summary of the results obtained by NeuroSim, provide a numerical overview of the relevant error metrics, and finally, a discussion of results.

Data-driven approaches to FPSO motion prediction and response-based design are relatively new and currently in the stage of validation, working as additional tools to traditional dynamic simulation methods. As a result, the research performed thus far constitutes a proof of concept of the proposed framework as an auxiliary tool to response-based design and, in the future, an addition to FPSO digital twins in seakeeping applications. Future steps include training meta-models with real measured FPSO motion, rather than the use of Dynasim to generate the predicted variables. As described in Section V-B, 18k metocean conditions measured over 6 years at the Brazilian coast were analyzed and simulated in a dynamic simulation software known as Dynasim, which generated the time-series motion of a spread-moored FPSO subject to these conditions. Output motion statistics were then extracted from the 6 DoF time-series and associated with the corresponding metocean conditions according to Equations 3 to 6, which were processed to generate Training, Validation, and Test datasets. The datasets were then used to perform a Hyperparameter Optimization technique known as Bayesian Optimization, which provided the optimal ANN architectures for each meta-model and platform draft. Finally, the proposed models were trained and tested. Some of the results obtained are visually presented in section VI-C.

TABLE 6. NeuroSim roll amplitude test error metrics for each FPSO draft.

Roll Amplitude		Error Metrics			
Draft	Model	MSE	RMSE(°)	MAE(°)	MaxError(°)
8	[1716,431,869]	0.00051	0.02259	0.01394	0.314
9	[3896,535,689]	0.00046	0.02156	0.01235	0.355
10	[2766,326,361]	0.00038	0.01938	0.01146	0.301
11	[2248,280,278]	0.00027	0.01636	0.01050	0.200
12	[4212,1214,1181]	0.00029	0.01715	0.01042	0.206
13	[1555,1096,341]	0.00027	0.01642	0.00972	0.212
14	[3341,1120,1644]	0.00025	0.01575	0.00943	0.279
15	[2274,319,536]	0.00026	0.01624	0.00981	0.209
16	[1311,501,1059]	0.00024	0.01565	0.00911	0.264
17	[3314,186,278]	0.00022	0.01489	0.00864	0.253
18	[1878,288,4092]	0.00027	0.01632	0.00865	0.343
19	[1790,332,704]	0.00017	0.01306	0.00798	0.163
20	[2393,230,462]	0.00019	0.01384	0.00838	0.207
21	[2256,215,1065]	0.00020	0.01406	0.00864	0.210
Average		**0.00028**	**0.01666**	**0.00993**	**0.251**

Table 6 illustrates the test errors of NeuroSim's Roll Amplitude meta-model, with the results from similar tables for the remaining three meta-models being summarized in Table 7. The model column indicates the optimal ANN architecture obtained through BO for each platform draft.

The average Roll Amplitude test MAE is 0.00993° which corresponds to 2.71% of the average maximum roll in the complete dataset (0.366°). For the CG Offset meta-model, the test MAE of 0.3906m is 4.37% of the output variable's average value 8.942m while its maximum test error of 6.3912m, when compared to the water depth at the platform's location, corresponds to only 0.51%. These results indicate NeuroSim's errors when predicting motion statistics for unseen environmental conditions are within an acceptable range.

TABLE 7. NeuroSim's average test error metrics across all platform drafts.

Meta-Model	Error Metrics			
	MSE	RMSE	MAE	MaxError
Roll Amplitude	0.00028	0.01666°	0.00993°	0.251°
CG Offset	0.5495	0.7373m	0.3906m	6.3912m
Parallel FD	0.0859	0.2924m	0.2019m	1.1623m
Vertical FD	0.00072	0.0267m	0.0167m	0.2581m

Since BO generates a history of trials or candidate ANN architectures, an alternative approach that better utilizes this set of architectures was implemented and compared to the results above. We tested a technique is known as Ensemble Regression, in which the final prediction is given by a composition of the prediction of multiple models. The 5 and 10 best ANN architectures obtained during the 500 trials of BO were used in ensembles, and the average prediction of its models gave the final predicted value. Figure 30 shows that ensembles reduce Test MAE across all platform drafts for the

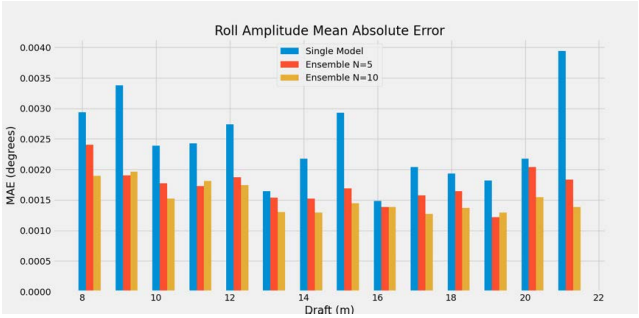

FIGURE 30. Test MAE for roll amplitude ensemble as a function of N for each FPSO draft, where N is the number of ANN models in the ensemble.

TABLE 8. NeuroSim's test error metrics for different ensemble approaches.

Meta-Model	Approach					
	Single Model		Ensemble ($N = 5$)		Ensemble ($N = 10$)	
	MAE	Max Error	MAE	Max Error	MAE	Max Error
Roll Amplitude	0.0099°	0.2512°	0.0088°	0.2357°	0.0086°	0.2326°
CG Offset	0.3906m	6.3912m	0.3678m	6.3766m	0.3652m	6.3141m
Parallel FD	0.2019m	1.1623m	0.1721m	1.0061m	0.1679m	1.0063m
Vertical FD	0.0167m	0.2581m	0.0143m	0.2208m	0.0140m	0.2174m

Roll Amplitude meta-model and Table 8 presents the results for all meta-models.

Overall, the optimized models obtained through BO and trained on simulated motion data presented good test results compared to the average accuracy of traditional dynamic simulation methods. The Roll Amplitude (Table 6) and Vertical Fairlead Displacement meta-models presented a slightly worse performance for lower draft values. In contrast, the CG Offset and Vertical Fairlead Displacement meta-models presented the inverse behavior, performing better for drafts of $8m$-$13m$. This result is possibly explained by the characteristics of the underlying motion, as roll and vertical fairlead displacements are vertical motions, whereas platform offset and parallel fairlead displacements are horizontal, parallel to the water plane.

Across all four meta-models, the Maximum Error metric displayed the highest variance, as it depends on a small number of specific, problematic metocean conditions, as illustrated in Figures 16 to 21. However, this metric is helpful in order to determine the absolute highest prediction errors given by NeuroSim across the entire Test set, providing valuable insights into high confidence upper boundaries of errors for unseen metocean conditions.

From Table 8 we conclude that ensembles of 5 and 10 models performed better across all meta-models, which is visually illustrated in Figure 30. There is a trade-off in terms of additional required memory and training times relative to error reduction for ensembles regression, diminishing returns as the number N of models in the ensemble increases. Interestingly, the performance increase when using ensembles compared to the single best model approach varies significantly for different values of FPSO draft.

VIII. CONCLUSION

In this work, we proposed a framework for developing, training, and validating a data-driven, neural network-based FPSO dynamics simulator called NeuroSim. The adopted methodology is comprehensively described in Section V-B and makes use of best practices of modern ML development to ensure scalability and ease of maintenance during production. NeuroSim's project code is based on the Kedro framework, which significantly organizes it in modular pipelines and efficiently stores all project data that flows through them. A state-of-the-art hyperparameter optimization algorithm is used to determine optimal ANN architectures for each of NeuroSim's models, which are then trained and tested.

The results obtained (Section VII) indicate NeuroSim is capable of capturing an FPSO's dynamic response to arbitrary incident metocean conditions, as mean errors are within an acceptable range. These results solidify the proposed framework as a successful proof of concept for the future development and training of NeuroSim on real, measured FPSO motion statistics and possible integration with other monitoring tools as part of a complete digital twin of an operating FPSO.

From the results obtained, three main issues are identified with the framework proposed in this paper: First, the ANN models are trained exclusively on simulated motion, rather than data obtained from a real operational FPSO. While this

TABLE 9. Optimal model architectures obtained through Bayesian Optimization in Pipeline P03 for roll amplitude and CG offset meta-models.

Meta-Model	Draft	HL 1	HL 2	HL 3	MSE
Roll Amplitude	8	1716	431	869	0.00114
	9	3896	535	689	0.00095
	10	2766	326	361	0.00056
	11	2248	280	278	0.00052
	12	4212	1214	1181	0.00054
	13	1555	1096	341	0.00052
	14	3341	1120	1644	0.00055
	15	2274	319	536	0.00051
	16	1311	501	1059	0.00057
	17	3314	186	278	0.00056
	18	1878	288	4092	0.00057
	19	1790	332	704	0.00046
	20	2393	230	462	0.00055
	21	2256	215	1065	0.00058
CG Offset	8	3425	702	32	0.6840
	9	2520	4243	2986	0.7111
	10	3637	2696	2622	0.7812
	11	4016	3098	1298	0.8428
	12	3623	1098	874	0.8922
	13	4202	1025	356	0.9656
	14	2140	1313	891	1.0073
	15	2387	2157	1454	1.1128
	16	2918	455	989	1.1878
	17	1347	4480	449	1.1993
	18	1192	845	965	1.2127
	19	1125	3446	1383	1.2358
	20	2937	2985	857	1.2466
	21	3186	1253	1055	1.2966

TABLE 10. Optimal model architectures obtained through Bayesian Optimization in Pipeline P03 for parallel and vertical maximum fairlead displacements.

Meta-Model	Draft	HL 1	HL 2	HL 3	MSE
Parallel FD	8	1641	3085	1693	0.1397
	9	976	1637	1046	0.1713
	10	977	703	2262	0.1771
	11	1050	4190	1262	0.1923
	12	1270	3827	1066	0.0944
	13	1687	4126	3460	0.1133
	14	1375	4082	1298	0.1066
	15	794	2975	897	0.1225
	16	1435	4261	1617	0.1143
	17	926	3744	1181	0.1198
	18	1036	3957	1099	0.1167
	19	777	1595	858	0.1270
	20	1318	4167	1532	0.1167
	21	954	2992	1185	0.1182
Vertical FD	8	2372	2507	2684	0.0010
	9	1977	2450	4005	0.0014
	10	1265	3392	891	0.0018
	11	855	1491	1786	0.0011
	12	600	765	4311	0.0012
	13	2277	1773	711	0.0011
	14	3426	3070	2397	0.0011
	15	1484	957	167	0.0012
	16	2441	4114	1750	0.0010
	17	1670	660	3090	0.0009
	18	1075	3626	3356	0.0009
	19	1818	3830	2677	0.0009
	20	2059	4205	2355	0.0010
	21	2041	3165	2187	0.0009

approach is interesting as a proof of concept, it needs to be validated on noisy, real-world data. Second, NeuroSim's test metrics, particularly the maximum errors (0.251° and 6.39m for the roll and offset outputs), can be further improved. Since Hyperparameter optimization has already been performed on the ANN architectures, a natural next step is to validate different types of regression models such as ensembles. Third, modern Mooring System Design techniques require additional motion statistics not predicted in the current version of NeuroSim, such as the standard deviation of the roll motion, the average CG offset and maximum line tensions.

In order to address these three issues, future work includes:

- Fine-tuning the proposed framework on real measured FPSO motion and testing NeuroSim in seakeeping applications as an auxiliary tool for real-time motion statistics prediction. Results will then be compared to values obtained through traditional simulation methods, indicating whether the meta-models can capture unmodeled dynamics.
- Further investigate Ensemble Regression techniques, such as Boosting or weighted average prediction of ANN models specialized in different regions of the input space, rather than using the simple average of the best ANN models. Different strategies such as Physics Informed Models can also be researched in order to reduce errors through the use of domain knowledge and known hydrodynamic equations.
- Increment the proposed framework to include additional motion statistics to be predicted, such as line tensions.

APPENDIX
BAYESIAN OPTIMIZATION RESULTS

This section presents the optimal Neural Network architectures found during Bayesian Optimization in Pipeline P03 for each of the four meta-models and 14 different platform drafts. Tables 9 and 10 summarize the set of hyperparameters that corresponded to the minimum value of the average Cross-Validation MSE after 500 optimization trials for the Roll Amplitude, Maximum Center of Gravity Offset, and Maximum Fairlead Displacements, respectively. Columns HL1, HL2 and HL3 indicate the number of neurons in each of the three ANN hidden layers.

REFERENCES

[1] S. Mazaheri, "The usage of artificial neural networks in hydrodynamic analysis of floating offshore platforms," *Int. J. Maritime Eng.*, vol. 3, no. 4, pp. 48–60, Sep. 2006.

[2] G. De Masi, F. Gaggiotti, R. Bruschi, and M. Venturi, "Ship motion prediction by radial basis neural networks," in *Proc. IEEE Workshop Hybrid Intell. Models Appl.*, Apr. 2011, pp. 28–32, doi: 10.1109/HIMA.2011.5953967.

[3] J.-C. Yin, A. N. Perakis, and N. Wang, "A real-time ship roll motion prediction using wavelet transform and variable RBF network," *Ocean Eng.*, vol. 160, pp. 10–19, Jul. 2018, doi: 10.1016/j.oceaneng.2018.04.058.

[4] B. Huang, J. Jiang, and Z. Zou, "Online prediction of ship coupled heave-pitch motions in irregular waves based on a coarse-and-fine tuning fixed-grid wavelet network," *J. Mar. Sci. Eng.*, vol. 9, no. 9, p. 989, Sep. 2021, doi: 10.3390/jmse9090989.

[5] A. C. de Pina, C. H. Albrecht, B. S. L. P. de Lima, and B. P. Jacob, "Wavelet network meta-models for the analysis of slender offshore structures," *Eng. Struct.*, vol. 68, pp. 71–84, Jun. 2014, doi: 10.1016/j.engstruct.2014.02.039.

[6] A. C. de Pina, A. A. de Pina, C. H. Albrecht, B. S. L. P. de Lima, and B. P. Jacob, "ANN-based surrogate models for the analysis of mooring lines and risers," *Appl. Ocean Res.*, vol. 41, pp. 76–86, Jun. 2013, doi: 10.1016/j.apor.2013.03.003.

[7] J. M. Gumley, M. J. Henry, and A. E. Potts, "A novel method for predicting the motion of moored floating bodies," in *Proc. Int. Conf. Offshore Mech. Arctic Eng.*, Jun. 2016, pp. 19–24, doi: 10.1115/OMAE2016-54674.

[8] A. M. Saad, F. Schopp, R. A. Barreira, I. H. F. Santos, E. A. Tannuri, E. S. Gomi, and A. H. R. Costa, "Using neural network approaches to detect mooring line failure," *IEEE Access*, vol. 9, pp. 27678–27695, 2021, doi: 10.1109/ACCESS.2021.3058592.

[9] M. Feurer and F. Hutter, "Hyperparameter Optimization," in *Automated Machine Learning*. Cham, Switzerland: Springer, 2019, doi: 10.1007/978-3-030-05318-5.

[10] T. Elsken, J. H. Metzen, and F. Hutter, "Neural architecture search: A survey," 2018, *arXiv:1808.05377*.

[11] P. I. Frazier, "A tutorial on Bayesian optimization," 2018, *arXiv:1807.02811*.

[12] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in *Proc. 24th Int. Conf. Neural Inf. Process. Syst.*, Granada, Spain, Dec. 2011, pp. 2546–2554.

[13] L. P. Cotrim, H. B. Oliveira, A. N. Q. Filho, I. H. F. Santos, R. A. Barreira, E. A. Tannuri, A. H. R. Costa, and E. S. Gomi, "Neural network meta-models for FPSO motion prediction from environmental data," in *Proc. Int. Conf. Offshore Mech. Arctic Eng.*, Jun. 2021, pp. 1–10, doi: 10.1115/OMAE2021-62674.

[14] T. M. Suller, E. O. Gomes, H. B. Oliveira, L. P. Cotrim, A. M. Saad, I. H. F. Santos, R. A. Barreira, E. A. Tannuri, E. S. Gomi, and A. H. R. Costa, "Evaluation of neural architecture search approaches for offshore platform offset prediction," in *Proc. Encontro Nacional de Inteligência Artificial E Computacional (ENIAC)*, 2021, pp. 326–337, doi: 10.5753/eniac.2021.18264.

[15] L. P. Cotrim, H. B. Oliveira, A. N. Q. Filho, A, E. A. Tannuri, A. H. R. Costa, E. S. Gomi, I. H. F. Santos, and R. A. Barreira, "Neural network meta-model for FPSO roll motion prediction from environmental data," in *Proc. Ibero-Latin-Amer. Congr. Comput. Methods Eng. (CILAMCE)*, 2021.
[16] L. Ma, J. Cui, and B. Yang, "Deep neural architecture search with deep graph Bayesian optimization," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell.*, Oct. 2019, pp. 500–507, doi: 10.1145/3350546.3360740.
[17] F. Itano, M. A. D. A. de Sousa, and E. Del-Moral-Hernandez, "Extending MLP ANN hyper-parameters optimization by using genetic algorithm," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2018, pp. 1–8, doi: 10.1109/IJCNN.2018.8489520.

LUCAS P. COTRIM received the bachelor's degree in mechatronic engineering from the Universidade de São Paulo, in 2019. He is currently pursuing the master's degree in control and mechanical automation engineering. In the years 2021 and 2022, he has been one of the professors to teach a reinforcement learning graduate course at the Instituto Mauá de Tecnologia. His research interests include artificial intelligence, reinforcement learning, and robotics.

RODRIGO A. BARREIRA graduated from the Federal University of Rio de Janeiro (UFRJ), in 2004. He received the Master of Science degree from UFRJ, in 2006. In 2005, he joined as a Naval Engineer at Petrobras. Since then, he has been working at the Petrobras Research Centre (Cenpes), where he has been leading research and development projects internally and in partnership with universities. He is currently a Naval Architect. His research interest includes the detection of abrupt change in time series with application in mooring line break detection systems of offshore platforms.

ISMAEL H. F. SANTOS was born in Rio de Janeiro, Brazil, in 1963. He received the bachelor's degree in electrical engineering and the Master of Science degree in applied mathematics from the Federal University of Rio de Janeiro and the Ph.D. degree in computer graphics/virtual reality from the Pontifical Catholic University of Rio de Janeiro. At the Petrobras Research Centre (Cenpes), he has been leading research and development projects using machine learning and data analytics techniques, since 2014. His research interests include the combination of artificial intelligence, computer graphics, and physics-informed machine learning to develop hybrid simulations (data and model driven) in the E&P segment, mainly production, offshore engineering, reservoir simulation, and geophysics.

EDSON S. GOMI (Senior Member, IEEE) received the B.Sc. degree in electrical engineering from the Escola Politécnica, Universidade de São Paulo (USP), in 1984, and the Ph.D. degree from The University of Tokyo, in 1996. He is currently an Assistant Professor with the Department of Computer Engineering, Escola Politécnica, USP. He also works on research projects in the areas of oil and gas, glaucoma diagnosis, and law and information technology. His research interests include artificial intelligence and machine learning.

ANNA H. REALI COSTA (Member, IEEE) received the Ph.D. degree from the Universidade de São Paulo (USP), Brazil. She investigated robot vision as a Research Scientist at the Karlsruhe Institute of Technology, Germany. She was a Guest Researcher at Carnegie Mellon University, USA, working in the integration of machine learning, planning, and execution in intelligent systems. She is currently a Full Professor in computer engineering at USP. She is the Scientific Director of the Data Science Center (C2D), a partnership between USP and the Itaú-Unibanco Bank. She is a member of the Center for Artificial Intelligence (C4AI), USP, a partnership with IBM and FAPESP. Her scientific contributions include artificial intelligence and machine learning, in particular reinforcement learning.

EDUARDO A. TANNURI was born in São Paulo, Brazil, in 1976. He received the degree in mechatronics engineering and the Ph.D. degree in control from the Universidade de São Paulo (USP), Brazil, in 1998 and 2003, respectively. He is currently a Full Professor with the Department of Mechatronics Engineering, USP. He has published 188 papers in conferences and 38 articles in journals. His research interests include non-linear dynamics and control, with applications in autonomous maritime vehicles and ship maneuverability. He has been a member of the International Towing Tank Conference (ITTC) Maneuvering Committee, since 2011. He has received the Best Student Paper Award in the IFAC CAMS Conference, in 2001, and the National Prizes—the Brazilian Navy Engineering Merit Award, in 2017, and the National Agency for Petroleum, Natural Gas and Biofuels Innovation Prize, in 2019.