POLITECNICO DI TORINO
Repository ISTITUZIONALE

A comparison of machine learning based surrogate models for multi-site performance assessment of wave energy converters

Original
A comparison of machine learning based surrogate models for multi-site performance assessment of wave energy converters / Fontana Crespo, R.N., Giorcelli, F., Pasta, E., Aliberti, A., Bottaccioli, L., Sirigu, S.A., Mattiazzo, G., Patti, E.. - In: SUSTAINABLE ENERGY, GRIDS AND NETWORKS. - ISSN 2352-4677. - ELETTRONICO. - 48:(2026). [10.1016/j.segan.2026.102551]

Availability:
This version is available at: 11583/3016552 since: 2026-10-14T14:23:55Z

Publisher:
Elsevier

Published
DOI:10.1016/j.segan.2026.102551

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

05 November 2026

Sustainable Energy, Grids and Networks 48 (2026) 102551

Contents lists available at ScienceDirect

Sustainable Energy, Grids and Networks

journal homepage: www.elsevier.com/locate/segan

A comparison of machine learning based surrogate models for multi-site performance assessment of wave energy converters

Rafael Natalio Fontana Crespo [a,*], Filippo Giorcelli [b,c], Edoardo Pasta [d], Alessandro Aliberti [e], Lorenzo Bottaccioli [e,f], Sergej Antonello Sirigu [b,c], Giuliana Mattiazzo [b,c], Edoardo Patti [a,f]

[a] Department of Control and Computer Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129, Torino, Italy
[b] Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129, Torino, Italy
[c] Marine Offshore Renewable Energy Lab (MOREnergy Lab), Politecnico di Torino, Via Paolo Borsellino 38/16, 10138, Torino, Italy
[d] Department of Electronics, Information and Bioengineering, Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133, Milano, Italy
[e] Interuniversity Department of Regional and Urban Studies and Planning, Politecnico di Torino, Viale Pier Andrea Mattioli, 39, 10125, Torino, Italy
[f] Energy Center Lab, Politecnico di Torino, Via Paolo Borsellino 38/16, 10138, Torino, Italy

ARTICLE INFO

Keywords:
Renewable energy
Performance assessment
Wave energy converter
Surrogate model
Machine learning

ABSTRACT

Wave energy is emerging as a promising renewable resource. Nonetheless, the levelized cost of energy for current Wave Energy Converters (WECs) still exceeds that of mature technologies like wind and solar. Bridging this gap demands an extensive assessment of possible designs and deployment sites. Therefore, we developed a machine learning (ML) based surrogate model that accelerates WEC performance evaluation. Although the framework is broadly applicable, its effectiveness is demonstrated here with the Pendulum Wave Energy Converter (PEWEC). The surrogate model receives the parameters describing single WEC setups as input and outputs the device's general site-independent power matrix. The surrogate architecture comprises two main components: a classification and a regression algorithm. The first determines whether the parameter combination yields a feasible device setup, while the latter predicts the site-independent power matrix. For the classifier, we compared the performance of five ML algorithms: Logistic Regression, Random Forest (RF), Support Vector Classifier, Extreme Gradient Boosting, and Feedforward Neural Network (FNN). For the regressor, we compared four ML algorithms: RF, Gaussian Process Regression, Radial Basis Function Neural Network, and FNN. The FNN algorithm showed the best results for the classification and regression tasks. Lastly, we assessed the performance of the FNN-based surrogate model for WEC performance prediction in four representative deployment sites. Regarding annual productivity, the surrogate achieved relative errors below 7.1% at all four sites, limited the absolute deviation from the reference numerical model to under 2.1 MWh, and cut the computational time per device assessment from 222 s to 0.11 s.

Notation

In this work, we adopt the following notational conventions. The set of real numbers is indicated by $\mathbb{R}$, while $\mathbb{C}$ represents the complex numbers (in which Re refers to the real part and Im to the imaginary part). The imaginary unit is denoted by j. The symbol 0 is used to represent the null element, irrespective of its dimensionality. The set of the first n positive natural numbers is expressed as $\mathbb{N}_n = \{1, 2, \dots, n\}$. Bold uppercase and lowercase letters are used to represent matrices and vectors, respectively. For a matrix $\mathbf{M} \in \mathbb{R}^{r \times c}$ with r rows and c columns, the element located at row a and column b is written as $M_{a,b}$, where $a \in \mathbb{N}_r$ and $b \in \mathbb{N}_c$. Likewise, for a vector $\mathbf{v} \in \mathbb{R}^s$ containing s components, the a-th element is denoted as v_a with $a \in \mathbb{N}_s$. The transpose of a matrix or vector is indicated by the superscript ^{T}.

1. Introduction

In order to be an active player in facing the global climate crisis, the European Union governments strive to reach net-zero pollution and carbon-free energy supplies, and declare their intention to achieve such ambitious results in [1]. Moreover, given the global scale of the challenge, the new technologies capable of providing clean energy must

* Corresponding author.
Email address: rafael.fontana@polito.it (R.N. Fontana Crespo).

https://doi.org/10.1016/j.segan.2026.102551
Received 25 March 2026; Received in revised form 21 September 2026; Accepted 6 October 2026
Available online 8 October 2026
2352-4677/© 2026 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

Nomenclature

A Attenuator
AEP Annual Energy Production
ANN Artificial Neural Network
BEM Boundary Element Method
CDF Cumulative Distribution Function
CFD Computational Fluid Dynamics
DB Data-based
DD Data-driven
DoF Degrees of Freedom
FD Frequency-Domain
FNN Feedforward Neural Network
GPR Gaussian Process Regression
LCoE Levelized Cost of Energy
LHS Latin Hypercube Sampling
LR Logistic Regression
MAE Mean Absolute Error
MAPE Mean Absolute Percentage Error
MAX Maximum Error
MBE Mean Bias Error
ML Machine Learning
MSE Mean Squared Error
NRMSE Normalized Root Mean Squared Error
OT Overtopping
OWC Oscillating Water Column
PA Point Absorber
PEWEC Pendulum Wave Energy Converter
PTO Power Take-Off
PR AUC Precision-Recall Area Under the Curve
R Correlation Coefficient
R^2 Coefficient of Determination
RBF Radial Basis Function
RBFNN Radial Basis Function Neural Network
RF Random Forest
RMSE Root Mean Squared Error
ROC AUC Receiver Operating Characteristic Area Under the Curve
RSM Response Surface Methodology
SHAP SHapley Additive exPlanations
SVR Support Vector Regressor
SVC Support Vector Classifier
SVGP Stochastic Variational Gaussian Process
T Terminator
TD Time-Domain
TRL Technology Readiness Level
WEC Wave Energy Converter
XGBoost Extreme Gradient Boosting

meet the different operating conditions arising from the wide variety of climatic contexts in which they must operate. Therefore, various efforts have been made concerning the diversification of the renewable energy sources to be exploited.

In this context, emerging wave energy is receiving a significant amount of interest. The attractive aspect of such a form of renewable energy is not only its yet untapped high potential [2–4], but also that it holds significant potential in areas where the availability of land for energy production is severely limited, *e.g.*, remote islands or coastal regions facing the dual challenges of rising sea levels and escalating population density and urbanization pressures [5,6].

The class of technologies aimed at harvesting wave motion and transforming it into electrical energy is called Wave Energy Converters (WECs). Compared to different forms of renewable technologies (*i.e.*, solar and wind technologies), the Levelized Cost of Energy (LCoE) of WECs is still not competitive, but projections predict a decrease, reaching a comparable level with offshore wind in the 2030 s [7]. On the other hand, WEC technology concepts have not yet converged toward a main archetype; instead, a galaxy of different prototypes with different working principles has been presented in the past century [8]. One of the most common classifications for this diversified technology is according to its working principle, dimension and orientation with respect to the incoming wave [9]:

- **OWC:** a WEC capable of exploiting the motion induced by waves of a compressed air volume column within a chamber to spin a turbine and generate electrical energy is called an Oscillating Water Column (OWC).
- **A** and **T:** Attenuator (A) and Terminator (T) WECs are systems parallelly-oriented or perpendicularly-oriented with respect to the incoming wave, respectively.
- **PA:** Point Absorber (PA) WECs are usually small-sized devices (their main dimension is smaller than the incoming wavelength) which harvest wave energy by exploiting the heave or pitch motion, *i.e.*, along a single degree of freedom (DoF), or multiple DoFs. WECs capable of collecting wave energy only via a single DoF but with more than one excited motion are defined as *underactuated* [10].
- **OT:** Overtopping (OT) devices, lastly, are WECs designed as a system capable of filling a water reservoir and then exploiting the potential wave energy generated by the difference in water volume height-pressure in order to produce electricity often by means of conventional low-head hydro-turbines [8].

However, this lack of convergence affects the Technology Readiness Level (TRL) of WECs, which is still highly variegated and generally presents early-stage values [8]. It follows that, for each WEC, its setup is extensively assessed under multiple environmental conditions, with the aim of searching for an efficient and economically viable system design [11,12]. In various recent review papers [8,12–14] authors stress the importance of directing efforts towards extensive techno-economic assessment studies, in order to increase the reliability of WECs' potential analyses and, consequently, the assessment of their economic viability in the process towards a higher TRL of the technology.

In the current stage of WEC technological development, progressing towards reliable and commercially viable devices requires comprehensive assessment campaigns under different configurations and operating setups. Generally, such an assessment should evaluate the device's energy productivity, economic viability, and overall feasibility under realistic sea conditions, which are strictly site-dependent. This process relies on different mathematical models [15,16], whose accuracy strongly depends on both the level of physical detail and the site-specific environmental conditions considered.

Typically, the performance assessment process of a WEC involves four key steps [17] (see blue lines and boxes in Fig. 1): geometric and parametric definition, hydrodynamic curves and dynamic response estimation, Power Take-Off (PTO) control for energy maximization, and performance evaluation.

However, performing such extensive performance analyses can be computationally demanding and time-consuming, especially when several device configurations are investigated in different locations. Employing accurate mathematical models may require considerable computational time, which can be unaffordable for performance assessment purposes on different sites. Moreover, WEC's performance assessment process remains strongly affected by the degree of accuracy of the employed models [8,12,16]. As a result, this trade-off

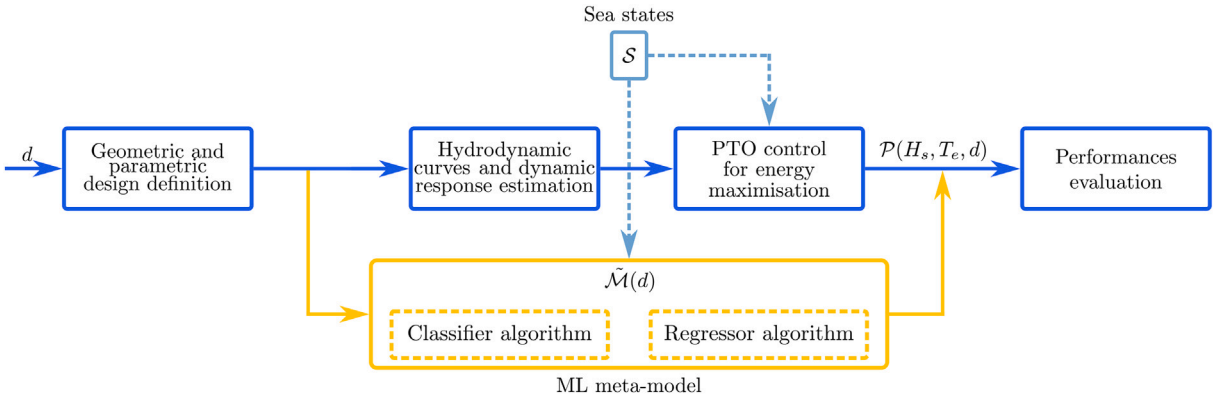


Fig. 1. Schematization of the DD/DB general surrogate model approach.

between computational efficiency and model accuracy must be carefully evaluated, and the suitability of the modeling approach may require experimental validation. In fact, the trade-off between model simplicity and predictive reliability must be carefully managed, as it directly affects the robustness of the resulting performance assessment.

In this sense, exploiting the advantages of data-driven (DD) and data-based (DB) approaches can help maintain a low computational burden while preserving high model fidelity. These methodologies enable a systematic exploration of the design space by efficiently capturing the nonlinear relationships between design parameters, environmental conditions, and device performance. By relying on a structured dataset rather than repeated high-fidelity simulations, such approaches allow for fast and accurate estimations of WEC performance metrics across a broad range of configurations. Therefore, developing accurate yet computationally efficient surrogate models represents a crucial step toward enabling extensive and systematic evaluations of WEC across different design setups and deployment sites [18]. However, as discussed in the following, for the purpose of the present study, learning the input–output mapping generated by a selected numerical workflow aims to enable the rapid estimation of WEC performance across broad design spaces, while its predictive validity remains conditional on the data-generating model. Indeed, in the present study, the surrogate is not intended to recover hydrodynamic effects that are absent from the reference model. Rather, it is used to bypass the repeated geometry-dependent hydrodynamic characterization, dynamic-response calculation, and PTO-control optimization required during techno-economic preliminary WEC performance assessment.

Recently, the Machine Learning (ML) field has shown substantial progress in advanced and efficient algorithms that demonstrate excellent capability in addressing complex engineering problems [19,20]. In this scenario, an ML-based surrogate model that can replace the computationally intensive "*white-box*" models can significantly reduce the computational cost of WEC design performance assessment. Surrogate models, also known as meta-models, have emerged as a prominent research focus in engineering applications [21,22], offering an effective strategy for mitigating prohibitive computational costs. These models are cost-effective accurate substitutes for complex and expensive computational tasks [23].

Surrogate models have been widely employed in the WEC literature for different applications, such as control [24,25], system identification [26], WEC farm motion prediction [27] and layout optimization [23,28–33], extreme load forecasting [34–36], control co-design [37–40], as well as for WEC design performance assessment [41–62]. These surrogate models have shown significant advantages in terms of computational cost while maintaining accuracy. Some of the most commonly employed ML algorithms for building the surrogates within the WEC literature are: the Radial Basis Function Neural Network (RBFNN), Feedforward Neural Network (FNN), Extreme Gradient Boosting (XGBoost), Gaussian Process Regression (GPR), Kriging, and Random Forest (RF). Regarding the use of surrogate models for WEC design performance assessment, many authors have employed them to reduce the computational burden of one of the steps of the WEC design procedure [41–57]. Other works have employed the surrogate to evaluate the WEC design only for particular sea conditions [58–62]. Nonetheless, to the best of the author's knowledge, there is no reference to employing surrogate models that account for a holistic approach to WEC design and extensive assessment, which considers the interactions among the different steps involved in the WEC design procedure.

Increasing WEC's economic viability, moving towards a higher TRL for the technology, and shrinking the LCoE gap with other mature renewable technologies such as solar and wind are still key challenges that need to be addressed. Bridging this gap demands research into WEC design, which must adopt a holistic design approach that considers the mutual interactions of the multiple design steps. As mentioned before, this holistic design approach is commonly based on detailed numerical models. However, the WEC design workflows commonly require exploring extensive design spaces, which are significantly constrained and generally time expensive due to the computational burden imposed by these detailed numerical models. Moreover, the design procedure for assessing the WEC technology in a particular location requires evaluating different WEC setup combinations. Therefore, this necessarily leads to running the detailed numerical model multiple times. Consequently, this computational time may significantly increase when aiming to explore multiple locations, making the multi-site-oriented design computationally prohibitive. In this context, substituting the detailed numerical model with an ML-based surrogate model can significantly reduce such computational burden, overcoming the "white-box" model's high computational costs while maintaining accuracy.

This work presents a surrogate model ($\tilde{\mathcal{M}}(d)$, generally transferable for any WEC design) for the WEC modeling phase. The proposed meta-model maps the design parameter space to the performance space (in terms of power production) for a set of sea states. The surrogate model considers the interactions between the various steps of the WEC design process. The multi-input/multi-output meta-model $\tilde{\mathcal{M}}(d)$ accepts as input the system's design setup $d \subset \mathcal{D}$ (where $\mathcal{D} \in \mathbb{R}^{N_D}$ is the design space) and returns the complete power matrix $\mathcal{P}(H_s, T_e, d) \in \mathbb{R}^{N_{H_s} \times N_{T_e}}$ for a given subset of sea states (defined by each waves' significant wave height H_s and energetic period T_e) of interest $\mathcal{S} \in \mathbb{R}^{N_{H_s} \times N_{T_e}}$. In this formulation N_{H_s} and N_{T_e} are the numbers of explored significant wave heights and energetic periods, defining the resolution of the sea-state space $\mathcal{S}$. A scheme of the proposed surrogate model is presented in Fig. 1.

In this way, the surrogate model's main characteristics are:

- $\tilde{\mathcal{M}}(d)$ can be applied to different sites, depending on the span of the explored sea states space S.
- $\tilde{\mathcal{M}}(d)$ can be applied to different WEC designs, depending on the span of the explored design space D.

Despite its general applicability, in this paper, the approach has been validated on a specific case study. For the particular WEC device studied, the design vector consists of thirteen WEC design parameters, *i.e.*, $d \in \mathbb{R}^{N_D}$ with the design space dimension N_D equal to thirteen. Given that many combinations of the design parameters may lead to infeasible combinations during the design process, the surrogate model comprises two main components: a classifier algorithm combined with a regressor algorithm. The classifier identifies whether the particular parameter combinations lead to a feasible design. For those feasible combinations, the regressor algorithm predicts a device's power matrix, covering an appropriately defined range of the sea-state space, which can be considered representative of both North Sea and Mediterranean applications. In particular, for the classifier algorithm, we design and compare the performance of five machine learning algorithms that have shown excellent performance in the literature for binary classification problems (*i.e.*, determining if the design is feasible or not): Logistic Regression (LR), RF, Support Vector Classifier (SVC), XGBoost, and FNN. On the other hand, for the regression algorithm, we compare four algorithms that have been extensively used in the literature for regression tasks as well as for WEC design optimization, and have demonstrated promising results: RF, FNN, RBFNN, and GPR.

The novelty of this work is a design-parameterized and site-agnostic surrogate framework that maps a physically meaningful WEC design layout directly to a complete site-independent power matrix, rather than approximating isolated modeling blocks or selected sea states, allowing a quick assessment of the device's techno-economic performance. For the particular WEC device studied, the classifier-regressor architecture first filters infeasible configurations (investigating the geometric and mechanical consistency of the design candidate, as well as its hydrodynamic stability) and then predicts the power response, enabling rapid multi-site techno-economic screening without executing the complete geometry-dependent hydrodynamic and PTO-optimization workflow for every new candidate design. In Section 2.3, a description of the paper's novelty and positioning within the current literature is given.

The rest of the article is organized as follows. Section 2 provides an overview of WEC mathematical modeling. Moreover, it reviews works in the literature that employ data-based and data-driven modeling approaches for WECs performance estimation. Section 3 presents the proposed methodology for the development of the two-stage surrogate model for multi-site WEC design performance assessment. Section 4 discusses the experimental results for the classification and regression tasks. Furthermore, it evaluates and compares the performance obtained by employing the reference numerical model of the PEWEC and the proposed surrogate for four representative deployment sites. Finally, Section 5 presents the concluding remarks.

2. Literature review and related work

2.1. Overview of WEC mathematical modeling and design optimization

The numerical modeling of WECs is at the core of the research activities for WEC development [8]. As anticipated in the introduction, both the performance accuracy and the computational affordability of the assessment performed during the WEC's design process strongly depend on the mathematical formulation of the physical WEC problem [11,16]. The impact of this aspect becomes evident when considering a widely used schematic framework for evaluating the performance of a WEC, which typically involves the four key steps mentioned above: geometric and parametric definition, hydrodynamic characterization and dynamic response estimation, PTO control for energy maximization, and performance evaluation. Therefore, a wide range of WEC research & development efforts tend to study the WEC hydrodynamics and develop high-fidelity models [63,64].

Generally, it is possible to summarize the numerous mathematical models employed in the literature to evaluate the hydrodynamic parameters of a WEC [65]: Frequency-Domain (FD) and Time-Domain (TD) models based on the Boundary Element Method (BEM), *i.e.*, potential flow models [16,64], and TD Computational Fluid Dynamic (CFD) models. FD models are generally based on the assumptions of linear potential flow: constant mean wetted surface (*i.e.*, small floater motion), an incompressible and irrotational fluid. On the other hand, BEM-based TD models include the effects of non-linearities [65].

Usually, in order to model the wave-structure interaction, it is typical in WEC research to describe the WEC's motion using the Cummins' equation [66] as the main governing equation:

$$[\mathbf{M} + \mathbf{A}_\infty]\ddot{\mathbf{u}}(t) + \int_0^t \mathrm{h}_r(t-\tau)\dot{\mathbf{u}}(t)d\tau + \mathbf{K}_h\mathbf{u}(t) = \mathbf{f}_{\mathbf{ext}}(t). \tag{1}$$

Deriving from Newton's Second Law for a body with N_{DoF} DoFs and described by the time-dependent coordinate vector $\mathbf{u}(t) \in \mathbb{R}^{N_{DoF}}$, on the left hand side of Eq. (1):

- the inertial properties of the whole WEC are expressed by the inertia matrix $\mathbf{M} \in \mathbb{R}^{N_{DoF}\times N_{DoF}}$ and the added mass at infinite frequency $\mathbf{A}_\infty \in \mathbb{R}^{N_{DoF}\times N_{DoF}}$,
- $\mathbf{K}_h \in \mathbb{R}^{N_{DoF}\times N_{DoF}}$ is the hydrostatic-restoring stiffness, which contributes to the balance between gravity and buoyancy forces as $\mathbf{f}_{hs}(t) = \mathbf{K}_h u(t)$,
- $\mathrm{h}_r \in \mathbb{R}^{N_{DoF}\times N_{DoF}}$ is the radiation impulse response function, contributing to the convolution term $\int_0^t \mathrm{h}_r(t-\tau)\dot{\mathbf{u}}(t)d\tau$ (which can be approximately computed by means of a state space representation as in [25,67]) to the radiation forces $\mathbf{f}_r(t) = \mathbf{A}_\infty + \int_0^t \mathrm{h}_r(t-\tau)\dot{\mathbf{u}}(t)d\tau$.

Then, on the right-hand side, the excitation forces vector can be expressed as:

$$\mathbf{f}_{ext}(t) = \mathbf{f}_{wave}(t) + \mathbf{f}_{wind}(t) + \mathbf{f}_{current}(t) + \mathbf{f}_{II}(t) + \mathbf{f}_{PTO}(t) + \mathbf{f}_{moor}(t), \tag{2}$$

where:

- $\mathbf{f}_{wave}(t)$ is the wave excitation force component, given by the sum of Froude-Krylov excitation and diffraction forces.
- $\mathbf{f}_{wind}(t)$ and $\mathbf{f}_{current}(t)$ are the wind and current forces, often included by means of a single component, *i.e.*, the drag force $\mathbf{f}_D(t)$ [65].
- $\mathbf{f}_{II}(t)$ are the second-order forces [68].
- $\mathbf{f}_{PTO}(t)$ and $\mathbf{f}_{moor}(t)$ are the PTO and mooring loads, respectively.

All the force components are elements $\mathbf{f}(t) \in \mathbb{R}^{N_{DoF}}$.

Alternatively, the WEC's hydrodynamics can be computed via an FD approach, whose formulation can be related to Eq. (1) by means of Ogilvie's equations [69]. However, it is common-practice in WEC research to assume the validity of linear wave theory [70] together with small floater oscillations (*i.e.*, the wetted surface remains mainly constant during the floater motion). Therefore, the FD model formulation yields:

$$\{-\omega^2[\mathbf{M} + \mathbf{A}(\omega)] + j\omega\mathbf{B}(\omega) + \mathbf{K}_h\}u(\omega) = \mathbf{f}_{wave}(\omega)\mathbf{A}(\omega) + \mathbf{f}_{PTO}(\omega), \tag{3}$$

where $\{\mathbf{A}(\omega), \mathbf{B}(\omega)\} \subset \mathbb{R}^{N_{DoF}\times N_{DoF}}$ are the hydrodynamic frequency-dependent (*i.e.*, ω) matrices of added mass and radiation damping, respectively. The Fourier transform of the device displacements and rotations is denoted by $\mathbf{u}(\omega) \in \mathbb{C}^{N_{DoF}}$. The corresponding elements in the FD of $\mathbf{f}_{wave}(t)$ and $\mathbf{f}_{PTO}(t)$ are $\mathbf{f}_{PTO}(\omega) \in \mathbb{C}^{N_{DoF}}$ and $\mathbf{f}_{wave}(\omega)A(\omega) \in \mathbb{C}^{N_{DoF}}$, with $A(\omega)$ denoting the wave amplitude coefficients.

The computation of both FD and TD BEM-based models can be performed via various tools, *e.g.*, WAMIT [71], OrcaWave [72] and NEMOH [73] for FD, while OrcaFlex [74], WECSim [75] or MOORDYN [76] for TD. Among the possible mathematical models of WECs, a trade-off between their computational burden and their fidelity is desired [16]. TD CFD-based models are able to fully describe the total non-linearities and the complex hydrodynamic interactions between the WEC and waves,

but with a high time requirement. TD BEM-based approaches are capable of including non-linear effects, *e.g.*, second-order forces and mooring effects, at a higher computational cost than FD BEM-based approaches.

On the other hand, the trade-off between computational efficiency and model accuracy should be carefully evaluated to select the most appropriate model, with respect to the purpose of the employment of a convenient mathematical model. For these reasons, in recent years, DD and DB models have gained popularity in the WEC field, both for the mathematical modeling [16] and control [18,77] applications.

A surrogate model $\tilde{\mathcal{M}}$ is a mathematical model able to mimic the behavior of a system that is determined only from input/output data. Via DD and DB models it is also possible to include unmodeled hydrodynamics and non-linearities but with a very reduced computational time [16].

Likewise for WEC control applications [77], the same definitions of DB and DD approaches given in [77,78] can be used in the WEC modeling field: a DB approach relies on a static dataset with no integration of online data flow, while DD approaches employ a dynamic dataset, which involves the use of online data during the process but does not exclude the presence of offline data collected beforehand.

Usually, the employed data arise from high-fidelity simulators (*e.g.*, CFD-based numerical wave tanks) or directly from experimental campaigns. With roots in the field of system identification [79], DD/DB models can be achieved in several ways, *e.g.*, via machine learning (ML) techniques. Thanks to their trade-off between accuracy and low computational burden, DD and DB models are an interesting alternative in WEC optimization or for developing new system control philosophies [77,80,81]. Although comprehensive reviews on DD and DB modeling approaches for WECs are available in the literature, a brief overview is provided in Appendix A to position the present work within the broader research landscape. For an in-depth analysis, the reader is referred to [77,82], which offer extensive surveys of DD and DB modeling techniques applied to WECs.

2.2. Surrogate models for WEC design performance assessment

The WEC design performance assessment category comprises studies that aim to evaluate the design performance of an individual WEC as well as of a group of WECs (also known as a WEC farm or array), as reported in Tables 1 and 2. Refs. [45,46] employed a Kriging surrogate model to replace the BEM-based model (*i.e.*, NEMOH WECSim model), and estimate the rated power of a multi-float WEC array based on the number of floats and their spacing, the separation between floats and reactors, and the PTO stiffness and damping. Similarly, Refs. [37–39] aimed to overcome the computational costs associated with employing BEM-based models for determining the hydrodynamic coefficients of an array of PA WECs. Therefore, the authors constructed a FNN-based surrogate model for estimating these hydrodynamic coefficients. To prepare the training data for the FNNs, Ref. [38] employed a Latin Hypercube Sampling (LHS) approach. On the other hand, Refs. [37,39] used Query by Committee as an active learning sampling strategy. The results showed a substantial improvement in terms of computational time relative to employing multi-scattering (*e.g.*, a 33 times improvement reported in [37]), while closely matching the solution in terms of accuracy. Recent contributions have also explored more advanced hybrid and meta-learning strategies for wave-energy power prediction, especially at the wave-farm scale. Neshat et al. proposed a meta-learner framework in which surrogate models are combined with a meta-learner to predict the total absorbed power of WEC farms at different Australian sites [83]. A subsequent study further developed this research direction by introducing a cooperative ensemble model, aimed at improving the accuracy and robustness of wave-farm power prediction [84]. These works show the increasing interest in hybrid and meta-learning models for wave-energy applications, particularly when farm layouts, inter-device interactions, and site-dependent conditions have to be represented.

Regarding the design performance assessment of individual WECs, many works aimed to reduce the computational burden of WEC design performance assessment by substituting the reference hydrodynamic evaluation models with surrogate ones [40–44,47–62]. Refs. [42–44] utilized an RBFNN to forecast the capture factor of a terminator WEC. The surrogate model substituted a numerical smoothed particle hydrodynamics model. The works in [51,54] constructed surrogate models for predicting the power generation of an attenuator WEC based on its submergence depth and length. In those cases, researchers employed LHS to sample the dataset. Ref. [54] used an FNN-based surrogate, while [51] employed an XGBoost-based one. Other works [48,50,53,59,61] studied the design performance of OWC WEC devices. Ref. [53] employed a Kriging-based surrogate model to predict the torque coefficient of a well's turbine based on four turbine design parameters as input. The surrogate model showed only a 0.06% relative error compared to the CFD model results. Instead, Ref. [48] constructed an RBFNN-based surrogate to estimate the diodicity of the fluidic diode of an OWC, in which the sample points were obtained using LHS. Moreover, Ref. [59] employed an FNN to determine the capture factor of an L-shaped OWC based on five of its chamber design parameters. Similarly, Ref. [61] compared RF-based and FNN-based surrogate models to predict the power generation of an OWC WEC based on the wave period, front wall, and chamber length. LHS was employed to sample the data. The results indicated that the FNN-based surrogate outperformed the RF one. Likewise, Ref. [50] utilized LHS for data sampling. They built and compared surrogate models using FNN and XGBoost algorithms to predict the average annual power of an OWC WEC, with the FNN demonstrating better performance.

Other authors used surrogate models to evaluate the design performance of PA WEC devices [40,41,47,49,52,55–58,60,62]. Ref. [56] studied the performance of a Response Surface Methodology (RSM) as a surrogate model that employed the design parameters of a PA double dumbbell flux-switching linear generator as input and predicted the peak power output of the generator. Furthermore, Ref. [60] used an FNN-based surrogate model to estimate the captured power of a two-body PA based on nine design parameters and wave frequency. In like manner, Ref. [47] utilized an FNN to construct a surrogate model that predicts with 93% accuracy the maximum heave displacement and additional mass of a buoy. Similarly, Ref. [62] employed an FNN to build a surrogate model that reduced the computational cost of hydrodynamic simulation during WEC design optimization. Likewise, Ref. [52] constructed an FNN surrogate to replace the hydrodynamic simulations performed in AQWA software, predicting the output power of the WEC based on key geometric variables. The results indicated that the computational time was reduced by over 99% by employing the surrogate, with an average error of 1.5%. Moreover, Ref. [55] employed an FNN to estimate the mean extracted power of a WEC based on its ballast weights and position, wave frequency, PTO damping and viscosity. They utilized LHS to obtain the dataset for training and evaluating the surrogate. Additionally, Ref. [57] extended the study by comparing the performance of employing an FNN as a surrogate with a Support Vector Regressor (SVR) and XGBoost, with the XGBoost-based surrogate showing the best performance. On the other hand, Ref. [41] compared using surrogates based on RSM, RBFNN, and an Elliptical Basis Function Neural Network to estimate the mean annual power for a buoy-shaped WEC design. They employed LHS for data sampling. The RSM showed the worst accuracy, while the Elliptical Basis Function Neural Network outperformed the RBFNN. Moreover, Ref. [49] built and compared surrogate models which replace the WEC hydrodynamic model and estimate the CWR and the costs of a PA based on the WEC outer radius, bottom radius and draft. The authors compared employing RSM and an FNN for building the surrogate, with the latter model showing better results.

However, the previously highlighted works employed surrogate models to substitute one of the steps of the WEC design process in isolation. To the best of our knowledge, there is no reference in the literature

Table 1
Literature on WEC surrogate models.

Author and Year	Data	WEC	Surrogate Model	Methodology	Metrics
Colby et al. [52]	Num. Sim.	PA	FNN	Constructed a FNN surrogate to replace the hydrodynamic simulations, estimating the output power of the WEC based on key geometric variables.	Percentage Error
Faraggiana et al. [45]	Num. Sim.	PA	Kriging	Kriging surrogate to determine the rated power of a multi-float WEC array based on five design parameters. The surrogate model replaced the NEMOH WECSim BEM-based models. Employed LHS for obtaining dataset.	NRMSE, MAX
Liu et al. [42]	Exp. & Num. Sim.	T	RBFNN	Utilized a RBFNN based surrogate to substitute the smoothed particles hydrodynamic model of a terminator WEC. The surrogate predicts the WEC capture factor based on nine design parameters	RMSE, R^2, Hit rate
Bao et al. [60]	Exp. & Num. Sim.	PA	FNN	Employed a FNN surrogate model. The metamodel predicted the captured power of a two-body PA based on nine geometric parameters and wave frequency	R
Khatri and Wang [56]	Num. Sim.	PA	RSM	Utilized a RSM surrogate to predict the peak output power of a double dumbbell flux-switching linear generator of a PA based on the generator design parameters.	R^2
Liu et al. [41]	Num. Sim.	PA	RSM, RBFNN, Elliptical Basis Function Neural Network	Compared surrogate models based on RSM, RBFNN, and Elliptical Basis Function Neural Network to optimize a PA. The surrogate replaced the hydrodynamic model, by predicting the mean annual power based on the design parameters. Employed LHS for data sampling.	R^2
V. and Koley [54]	Num. Sim.	A	FNN	Employed a FNN surrogate model to predict the power generation of an attenuator WEC based on its submergence depth and length. The surrogate was utilized to substitute the hydrodynamic model. LHS was employed as sampling strategy	R^2
Azad and Herber [38]	Num. Sim.	PA	FNN	Employed FNN surrogate to estimate the model hydrodynamic coefficients of an array of PA WECs. Used LHS as sampling strategy.	MSE
Kotb et al. [53]	Num. Sim.	OWC	Kriging	Kriging surrogate to predict torque coefficient of a Wells turbine based on four design parameters of the turbine blade. Surrogate model substituted hydrodynamic model.	RMSE, R^2
Poguluri et al. [55]	Num. Sim.	PA	FNN	Employed a FNN surrogate to predict the mean extracted power of the rotor of a PA. The surrogate utilized in input the rotor's ballast position and weight, viscosity, and PTO damping. Used LHS as sampling strategy.	RMSE, R^2
Lin et al. [59]	Exp. & Num. Sim.	OWC	FNN	Constructed a FNN surrogate to determine the capture factor of an L-shaped OWC. The surrogate employed in input five of the WEC's chamber design parameters.	MSE
Hithaish et al. [48]	Num. Sim.	OWC	RBFNN	Employed a RBFNN based surrogate model. The surrogate was employed to estimate the diodicity of a fluidic diode of an OWC. Utilized LHS for obtaining the dataset	Grid Convergence Index
V. et al. [50]	Num. Sim.	OWC	FNN, XGBoost	Compared XGBoost and FNN surrogate models for predicting the average annual power of an OWC. Used LHS for data sampling.	MAE, RMSE, MAPE, R^2
Jiang et al. [47]	Num. Sim.	PA	FNN	Used a FNN based metamodel to replace a PA hydrodynamic model and predict maximum heave displacement and additional mass of the buoy based on its geometric parameters	Relative Error

to the implementation of surrogate models that substitute for the combination of steps within the WEC design process, in such a way that takes into consideration the interaction of these steps.

2.3. Motivations, scientific novelty and contributions

Enhancing WEC's economic viability and, therefore, narrowing the LCoE disparity with more established renewable sources (*e.g.*, wind and solar) remain critical challenges. Addressing these challenges requires research into WEC design, following a holistic design philosophy that accounts for the interdependencies among the steps involved within the WEC design workflow. As previously noted, this comprehensive design approach typically relies on detailed numerical models. Nonetheless, WEC design workflows often entail the exploration of vast design spaces and extensive assessment of their power capabilities, which can be computationally intensive due to the time requirements of the employed numerical models. Defining the proper WEC designs for a specific site may require hundreds or thousands of simulations, potentially extending assessment processes over several days. This computational demand escalates considerably when considering multiple geographical sites, making multi-site design operationally impractical. In this context, substituting the computationally intensive numerical models with an ML-based surrogate model offers a promising alternative and an effective trade-off. This approach substantially reduces the computational burden while maintaining satisfactory predictive accuracy, particularly during the initial phase of WEC design.

In this work, we present a novel methodology that employs surrogate models (generally transferable to any WEC design) for WEC performance and design assessment. The approach is then applied for the Pendulum Wave Energy Converter (PEWEC), as an example. The proposed surrogate model maps the PEWEC parametric design and performance space (quantified via power output) for multivariate sea state conditions. The surrogate explicitly accounts for the interdependencies among the various steps involved in the WEC design process. The model has a multi-input/multi-output architecture that, for the PEWEC case, receives thirteen design parameters (which will be introduced in the next Section 3.1.1) as input and outputs the general site-independent power matrix of the PEWEC. During the WEC design process, it is common for numerous design parameter combinations to lead to infeasible design solutions. Therefore, the surrogate architecture comprises two main components: a classification algorithm and a regression algorithm. The classifier determines whether a given set of design parameters yields a feasible design solution. For the viable combinations, the regressor predicts the site-independent power matrix. For the classifier algorithm, we design and evaluate the performance of five machine learning algorithms that have shown promising results for binary classification problems [85–89]: LR, RF, SVC, XGBoost, and FNN. Instead, we employ four algorithms for the regression task that have demonstrated excellent

Table 2
Literature on WEC surrogate models part 2.

Author and Year	Data	WEC	Surrogate Model	Methodology	Metrics
Azad et al. -[37]	Num. Sim.	PA	FNN	Utilized FNN surrogate to determine hydrodynamic coefficients of a PA WEC. Used Query By Committee as sampling strategy.	MSE
Poguluri and Bae [57]	Num. Sim.	PA	FNN, SVR, XGBoost	Compared FNN, SVR and XGBoost surrogate model for determining extracted power of a PA. The surrogate utilized in input the rotor's ballast position and weight, viscosity, and PTO damping. Employed LHS as sampling strategy	RMSE, R^2, MAE
Wang and Liu [43]	Exp. & Num. Sim.	T	RBFNN	Used a RBFNN metamodel to estimate the capture factor of a terminator WEC based on nine design parameters. The surrogate model substituted an smoothed particles hydrodynamics model.	RMSE
Wang et al. [44]	Exp. & Num. Sim.	T	RBFNN	Employed a RBFNN-based surrogate model for substituting an smoothed particles hydrodynamics model of a terminator WEC. The surrogate used in input nine design parameters to predict the WEC's capture factor.	RMSE
V. and Koley [51]	Num. Sim.	A	XGBoost	Used an XGBoost surrogate model to predict the power generation of an attenuator WEC based on its submergence depth and length. The surrogate was utilized to substitute the hydrodynamic model. LHS was employed as sampling strategy	R^2, MAPE, MAE, RMSE
Rao et al. [61]	Num. Sim.	OWC	FNN, RF	Compared RF and FNN based surrogate model for predicting the power generation of an OWC based on wave period, depth submersion, front wall and chamber length. Employed LHS for data sampling	MAE, RMSE, MAPE, R^2
Lin et al. [62]	Num. Sim.	PA	FNN	Employed a FNN based metamodel to substitute a PA hydrodynamic model. The surrogate was used to determine the hydrodynamic behavior of the buoy based on its design parameters	Percentage Error
Azad et al. [39]	Num. Sim.	PA	FNN	Utilized an FNN surrogate to estimate hydrodynamic coefficients of a WEC array. Employed Query By Committee as sampling strategy.	MSE
Arrosyid et al. [49]	Num. Sim.	PA	RSM, FNN	Compared RSM and FNN-based surrogate models for estimating CWR and the costs of a PA based on the WEC outer radius, bottom radius and draft. The surrogate model replaced the WEC hydrodynamic model	MSE, R^2

behavior as surrogates for WEC modeling: RF [61], GPR [90], RBFNN [41–44,48], and FNN [37–39,47,49,50,52,54,55,57,59–62].

The distinctive contribution of the present work is not limited to the use of machine-learning algorithms as computationally efficient substitutes for numerical models. Rather, the proposed framework introduces a design-parametrized, and site-agnostic surrogate architecture for preliminary WEC techno-economic performance assessment. In contrast with most existing surrogate-based WEC studies, which generally approximate a single modeling block or predict performance under selected wave conditions, the present approach maps a physically meaningful WEC design vector directly to a complete site-independent power matrix. Such an input design vector includes parameters that uniquely define a preliminary WEC configuration. For the PEWEC case study, this vector includes the main parameters defining a preliminary device layout, namely hull geometry, inertial properties, ballast and pendulum arrangement, number and position of pendulum units, and PTO selection. The resulting power matrix is independent of a specific deployment site and can therefore be combined with different wave-occurrence matrices to estimate site-specific performance indicators. Considering that the hydrodynamic and PTO-optimization workflows are required to generate the design-dependent power matrix, the aim of the surrogate is to predict this matrix for a new candidate design. The evaluation of different deployment sites only requires its combination with the corresponding occurrence matrices (see Fig. 1). Moreover, the proposed classifier-regressor coupling explicitly accounts for the physical feasibility of candidate configurations before predicting their power response, which is particularly relevant when large and highly constrained WEC design spaces are explored. In this sense, the framework is intended to support rapid techno-economic screening and multi-site assessment of possible WEC design configurations, while retaining the interactions among the main steps of the conventional design and performance-evaluation chain.

The practical relevance of the proposed framework does not rely on treating the adopted model as a high-fidelity simulator. Rather, it derives from replacing a complete and repeatedly executed design-dependent numerical workflow. For each new PEWEC candidate, the conventional procedure requires a geometry-dependent hydrodynamic characterization, the calculation of the dynamic response, and the optimization of the PTO-control parameters over the representative sea-state domain. The proposed surrogate directly maps the design vector to the resulting site-independent power matrix, bypassing all these intermediate calculations during inference. Its offline development cost can therefore be amortized over the large number of candidate designs typically considered in preliminary design-space exploration, optimization, and multi-site techno-economic assessment. The adopted linear FD formulation is commonly used in preliminary techno-economic WEC design and optimization (*e.g.*, [11,12,16,91]) because it provides a practical compromise between accuracy and computational burden. Nevertheless, its validity depends on the WEC concept, operating regime, and motion amplitude, and may decrease when energetic sea states or control actions produce responses outside the small-motion range (see the so-called WEC modeling paradox [92]). Consequently, the screening performance and candidate rankings reported in this study remain conditional on the adopted linear reference model and should not be interpreted as high-fidelity predictions.

3. Methodology

This section describes the methodology employed to develop the surrogate model. The workflow is depicted in Fig. 2. We propose a novel approach employing surrogate models for WEC assessment, which is generally transferable to any WEC setup. In particular, as anticipated, we studied the assessment and design of the PEWEC device. The *surrogate model* maps the PEWEC design parameters to the general site-independent power matrix of the device. Given that during the PEWEC design process (and WECs in general), many design parameter combinations may lead to infeasible design solutions, the surrogate architecture comprised two main components: a *classification algorithm* and a *regression algorithm*. The *classification algorithm* assesses whether a specific combination of design parameters leads to a feasible design configuration. To this end, we designed and compared the performance of five

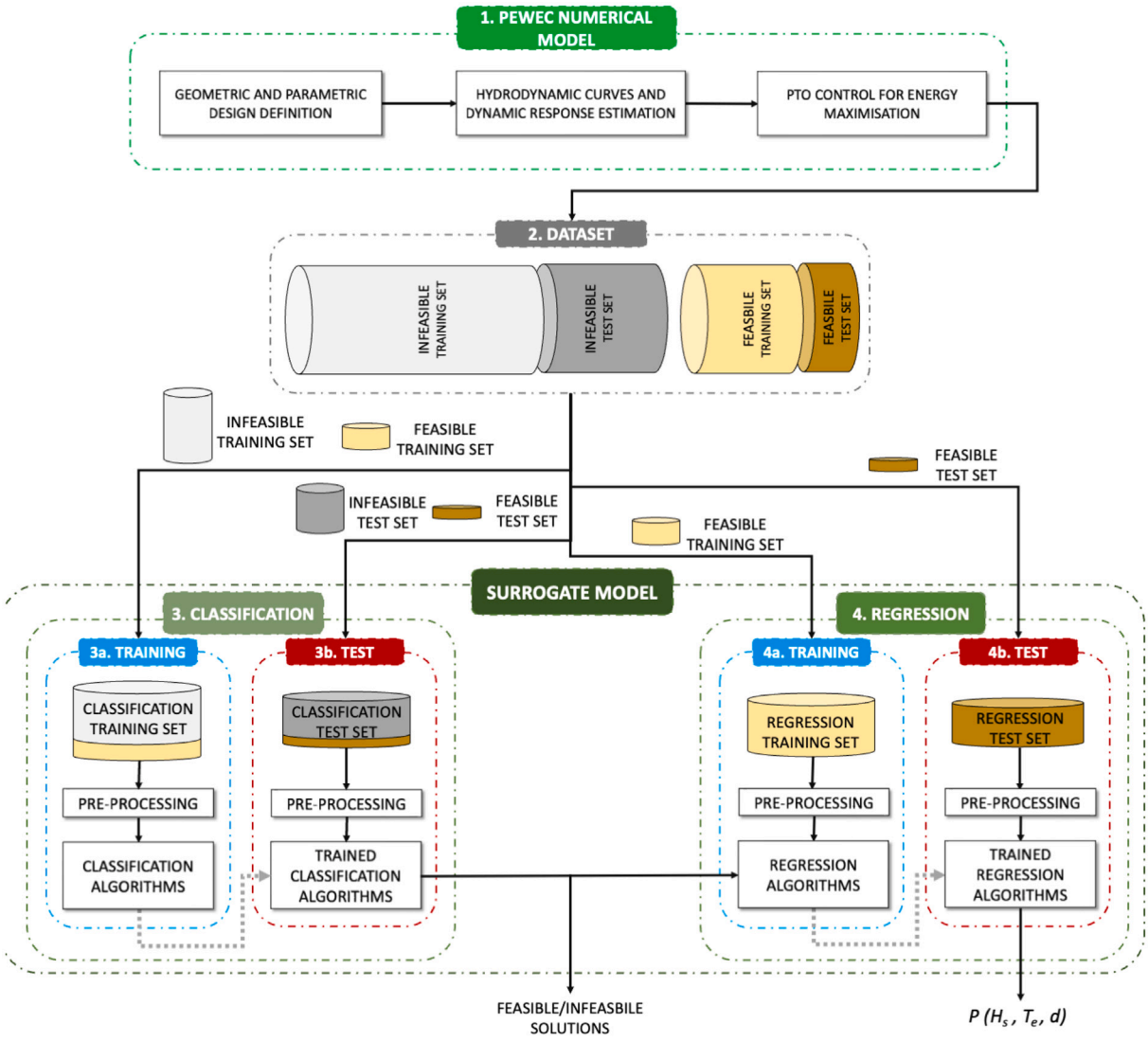


Fig. 2. Schema of the proposed methodology.

ML algorithms that have demonstrated excellent results in the literature for binary classification tasks: LR, RF, SVC, XGBoost, and FNN. Furthermore, the *regression algorithm* determines the site-independent power matrix for the feasible designs. For the regression task, we compared the performance of four ML algorithms that have shown promising results as surrogates in WEC extensive assessment and design: RF, GPR, RBFNN, and FNN.

3.1. PEWEC numerical model

The PEWEC is a floating, inertia-based device in which the hull's pitch motion, induced by incident waves, drives an enclosed pendulum about its rotational axis, ε. This pendular motion is transmitted, via a gearbox, to a PTO that converts mechanical rotation into electrical power. The fully enclosed design enhances reliability under harsh marine conditions, while a four-line mooring system simultaneously ensures station-keeping and allows the required pitch motion in both operational and extreme sea states [10,63,93]. A graphical 3D render of the PEWEC device is shown in Fig. 3 (c).

Assuming the device is aligned with the predominant wave direction, the state of motion can be represented by

$$\mathbf{u} = \begin{bmatrix} x & z & \delta & \varepsilon \end{bmatrix}^T,$$

where x is the surge displacement along the x-axis, z is the heave displacement along the z-axis, δ is rotation about the transverse (y) axis, and ε is the pendulum's rotation about its own axis. The reference system of the PEWEC is depicted in Fig. 3 (b).

A state-of-the-art linear frequency-domain model of the PEWEC is employed to balance computational efficiency and fidelity. Under potential-flow assumptions, the coupled dynamics of hull and pendulum are written as:

$$-\omega^2\left[\mathbf{M}+\mathbf{A}(\omega)\right]\mathbf{u}(\omega) + j\,\omega\,\mathbf{B}(\omega)\,\mathbf{u}(\omega) + \left[\mathbf{K}_h+\mathbf{K}_p\right]\mathbf{u}(\omega) = a_\omega(\omega)\,\mathbf{F}_\omega(\omega) + \mathbf{T}_{\mathrm{PTO}}(\omega), \quad (4)$$

where:

- $\mathbf{M} \in \mathbb{R}^{4\times4}$ is the combined mass-inertia matrix of hull and pendulum;
- $\mathbf{A}(\omega) \in \mathbb{R}^{4\times4}$ and $\mathbf{B}(\omega) \in \mathbb{R}^{4\times4}$ are the frequency-dependent added mass and radiation damping matrices, respectively, obtained via a BEM solver, in this study NEMOH [73];
- $\mathbf{K}_h \in \mathbb{R}^{4\times4}$ is the hydrostatic stiffness matrix;
- $\mathbf{K}_p \in \mathbb{R}^{4\times4}$ is the pendulum's restoring force matrix;
- $a_\omega(\omega)\,\mathbf{F}_\omega(\omega) \in \mathbb{C}^4$ is the wave excitation vector (with $a_\omega(\omega)$ denoting the wave amplitude and $\mathbf{F}_\omega(\omega)$ the wave excitation coefficients);

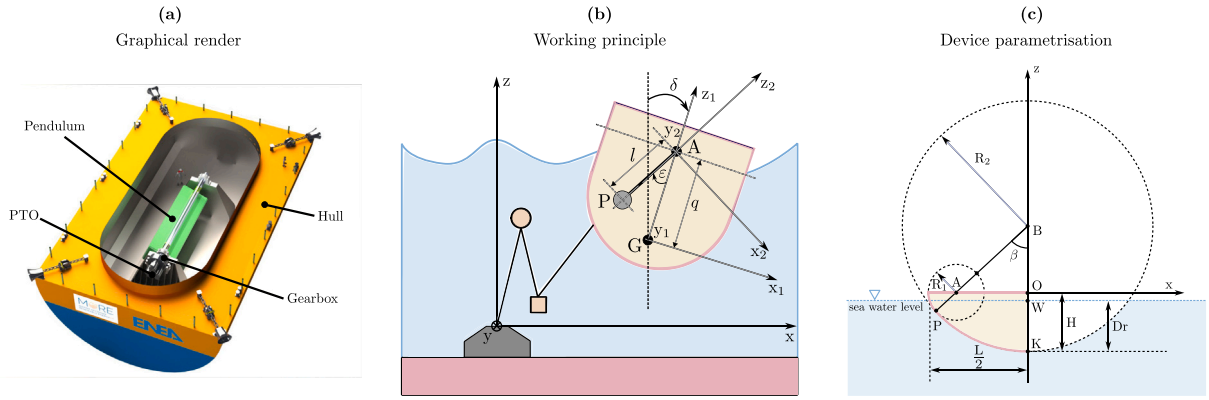


Fig. 3. PEWEC graphical rendering (a), working principle (b), and device parametrization (c).

- $\mathbf{T}_{\text{PTO}} \in \mathbb{R}^4$ is the vector of PTO torques, acting only on the pendulum degree of freedom;
- $\mathbf{u}(\omega) \in \mathbb{C}^4$ denotes the spectral amplitudes of the motion.

The mass-inertia coupling between hull and pendulum is given by:

$$\mathbf{M} = \begin{bmatrix} m_h + m_p & 0 & m_p(q-l) & -m_p l \\ 0 & m_h + m_p & 0 & 0 \\ m_p(q-l) & 0 & I_h + I_p + m_p(q-l)^2 & I_p + m_p l^2 - m_p q l \\ -m_p l & 0 & I_p + m_p l^2 - m_p q l & I_p + m_p l^2 \end{bmatrix}, \tag{5}$$

where:

- m_h and m_p are the hull and pendulum masses, respectively;
- I_h and I_p are the moments of inertia of hull and pendulum about their centers of gravity;
- q is the vertical distance from the hull's center of gravity to the pendulum pivot;
- l is the pendulum arm length.

The pendulum's restoring matrix is

$$\mathbf{K}_p = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -g\, m_p(q-l) & g\, m_p l \\ 0 & 0 & g\, m_p l & g\, m_p l \end{bmatrix}, \tag{6}$$

where g is the gravitational acceleration. The PTO control law is modeled as a first-order transfer function:

$$\mathrm{T}_{ctrl}(\omega) = \frac{j\omega\alpha_{PTO}}{j\omega + \beta_{PTO}}, \tag{7}$$

$$T_{\text{PTO}}(\omega) = \frac{T_{\text{ctrl}}(\omega)}{\tau_{\text{gear}}}, \tag{8}$$

$$\mathbf{T}_{\text{PTO}}(\omega) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ T_{\text{PTO}}(\omega) \end{bmatrix}, \tag{9}$$

where α_{PTO} and β_{PTO} are the two parameters optimized during the control internal optimization loop, and are defined as positive real numbers (*i.e.*, $\{\alpha_{PTO}, \beta_{PTO}\} \in \mathbb{R}^+$). The gearbox ratio is τ_{gear}, instead the absorbed power can be formulated as:

$$P_w(\omega) = \int_0^\infty \mathrm{Re}\left\{\mathrm{T}_{ctrl}\mathrm{H}_{\dot{\varepsilon}\dot{\varepsilon}}(\omega)\right\} d\omega, \tag{10}$$

where $\mathrm{T}_{ctrl}(\omega)$ describe the PTO control action and $\mathrm{H}_{\dot{\varepsilon}\dot{\varepsilon}}(\omega)$ refers to the the power spectral density [94] of the ε-axis rotational velocity, which is formulated in FD as:

$$\dot{\varepsilon} = j\omega\varepsilon(\omega). \tag{11}$$

A control-parameter optimization is performed over the set S of N_c representative sea states, each modeled by a Joint North Sea Wave Project (JONSWAP) spectrum [95,96], to maximize mean absorbed power. After identifying optimal control parameters for each sea state, operational constraints (maximum generator speed and PTO torque) are enforced to discard infeasible parameter pairs. The Annual Energy Production (AEP) is then calculated as

$$\mathrm{AEP} = \frac{3600 \cdot 24 \cdot 365}{100} \sum_{i=1}^{N_w} \left(\mathrm{Occ}_i \cdot P_w^i(\omega)\right), \tag{12}$$

where Occ_i is the percentage occurrence of the i^{th} sea state, and $P_w^i(\omega)$ is the mean absorbed power under the corresponding optimal control parameters [25].

From a computational standpoint, this workflow is design-dependent. Any variation of the PEWEC design vector requires a new geometry-dependent hydrodynamic characterization, followed by the calculation of the device response and the PTO-control optimization over the complete set of representative sea states. The surrogate developed in this work emulates the final design-to-power mapping produced by this complete numerical workflow, thereby avoiding the repetition of these intermediate calculations for every new candidate design.

To ensure computational tractability, the model neglects second-order drift forces, viscous damping, nonlinear mooring effects, and PTO losses.

It should be noted that the surrogate performance reported in this work is therefore conditional on the fidelity and assumptions of the adopted reference model. The ML algorithms do not learn the true device physics directly, but rather the input–output mapping generated by the selected numerical workflow. If additional nonlinear effects were included, such as viscous damping, nonlinear mooring loads, second-order wave forces, or PTO losses, the resulting design-to-power mapping could change. Consequently, the relative performance of each tested algorithm may also change and should therefore be interpreted as an empirical result for the present PEWEC dataset and modeling assumptions, rather than as a general statement valid for all hydrodynamic model fidelities. Accordingly, the computational speed-up reported in this work concerns the emulation of the adopted reference workflow and should not be interpreted as a correction of its model-form limitations.

3.1.1. PEWEC design parametrization

This section introduces the thirteen PEWEC design parameters. The PEWEC device, comprising its geometry and internal components, is fully defined via design parametrization as done in a previous PEWEC

design investigation [91]. Referring to the longitudinal section (x–z plane), the hull profile consists of a bottom circle tangent to two additional symmetrical circles at the bow and stern (Fig. 3 (a)). The three-dimensional hull is generated by extruding this profile along the y-axis, resulting in symmetry about both the x–z and y–z planes. Generally, the hull parameters include:

1. Hull length L (dimension along the x-axis);
2. Hull width W (dimension along the y-axis);
3. Bow curvature ratio $h = x_A/(L/2)$;
4. Height-to-draft ratio $k = x_A/(H/Dr)$, where H is the overall hull height and Dr is the draft;
5. Draft-based pitch limit α, which bounds the maximum pitch angle to avoid green water on deck;
6. Ballast filling ratio (BFR), a scalar in $[0, 1]$ that sets the fore-to-aft ballast distribution among bow, stern, and keel tanks, influencing roll and pitch inertia. When BFR is equal to 0 all the ballast is in the keel, while at BFR equal to 1 all the ballast is in the fore and aft tanks.

Referring to previous studies [91], the structural mass of the hull, m_h, is computed as 130 times the total hull volume, assuming a uniform steel shell (with density $\rho_h = 7800\ \mathrm{kg/m^3}$). The equivalent shell thickness is determined from the hull's surface area. Sand (density $\rho_b = 1400\ \mathrm{kg/m^3}$) is used as ballast due to its favorable cost-to-weight ratio [97]. Then, the pendulum unit is described by:

7. Pendulum mass m_p, defined by a rectangular steel block (density $\rho_p = \rho_h$) of cross-section $\phi \times \psi$ and length l_p, located a vertical distance l below the hull's center of gravity. A 40% mass allowance is added to m_p to account for housing, PTO, and support structure [91].
8. Pendulum arm length l.
9. Pendulum block width ϕ.
10. Pendulum block height ψ.
11. Number of identical pendulum units N_u.
12. Fulcrum position λ_0, defining the vertical placement of the pendulum axis to ensure full rotation without hull interference.
13. PTO unit identifier, selected from a discrete library of 37 commercially available permanent-magnet synchronous generators (each with a specified nominal speed, gearbox ratio, and torque).

Thus, each PEWEC design is fully specified by $d \in \mathbb{R}^{13}$, with $d = [L, W, h, k, \alpha, \mathrm{BFR}, m_p, l, \phi, \psi, N_u, \lambda_0, \mathrm{PTO\ ID}]$. The design variable space $\mathcal{D}$ is explored within the bounds reported in Table 3.

Once the key elements of the *PEWEC Numerical Model* block (see Fig. 2) have been specified, we can detail how these are employed in the classical framework, as reported with blue lines in Fig. 1:

- **Geometric and parametric design definition.** Let $d \in \mathbb{R}^{13}$ denote a candidate set of the thirteen PEWEC design variables within the design space $\mathcal{D}$. As we described, d fully defines the device's hull geometry, inertial properties, and pendulum parameters.
- **Hydrodynamic curves and dynamic response estimation.** With the geometry and mass-inertia characteristics fixed, the next step is to perform hydrodynamic and hydrostatic analyses, via the NEMOH software in this investigation.
- **PTO control for energy maximization.** From these results, the device's dynamic response is computed for a suite of wave conditions representative of the target site. For each sea state in this representative set, a nested optimization loop adjusts the PTO control parameters to maximize P_w, resulting in optimal control parameter vectors $\left\{\boldsymbol{\alpha}_{PTO}^{opt}, \boldsymbol{\beta}_{PTO}^{opt}\right\} \subset \mathbb{R}^{N_c}$.

Lastly, the **Performance Evaluation** block of Fig. 1 will be employed in the same manner for both the numerical model framework and the surrogate one. By interpolating the absorbed-power outputs across a grid of significant wave height H_s and energy period T_e, a continuous power surface is reconstructed, *i.e.*, the power matrix $\mathcal{P}(H_s, T_e, d)$. Evaluating this surface at all non-zero occurrence sea states (N_w) yields the expected power capture for each condition and the AEP evaluation, following Eq. (12).

Table 3
PEWEC thirteen design parameters.

Design Parameter	Symbol	Unit	Lower Bound	Upper Bound
Hull length	L	(m)	10	30
Hull width	W	(m)	5	25
Bow curvature ratio	h	(/)	0.4	1
Height-to-draft ratio	k	(/)	0.4	1
Draft-based pitch limit	α	(/)	15	25
Ballast filling ratio	BFR	(/)	0.1	1
Pendulum Mass	m_p	(kg)	5000	50,000
Pendulum Arm Length	l	(m)	0.5	5
Pendulum block width	ϕ	(m)	0.2	2.5
Pendulum block height	ψ	(m)	0.2	2.5
Number of Pendulum	N_u	(/)	1	4
Fulcrum position	λ_0	(m)	1	1
PTO ID	PTO ID	(/)	1	37

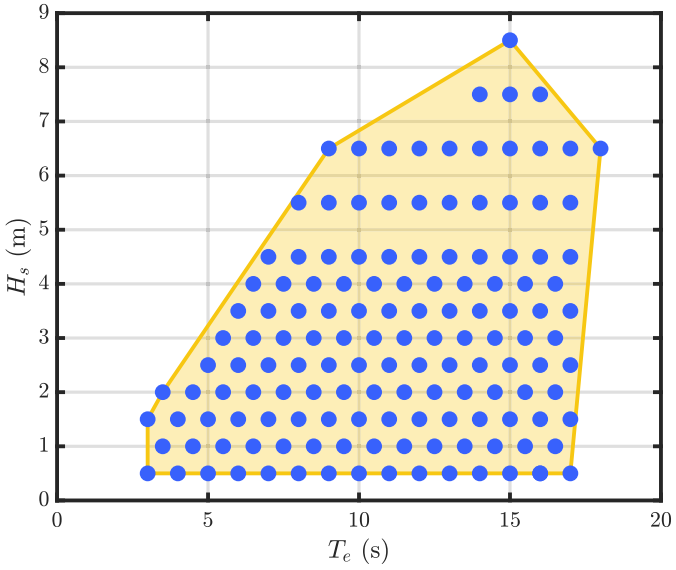


Fig. 4. Set of the 141 representative waves. In blue dots are shown the 141 sea states and in yellow the covered area. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

3.2. Dataset and pre-processing

This section presents a detailed description of the *Dataset* block within the proposed methodology, depicted in Fig. 2. As introduced in Section 3.1, thirteen design parameters of the PEWEC are adopted to uniquely characterize each possible device configuration. For a given subset of sea states of interest $S \in \mathbb{R}^{N_{H_s} \times N_{T_e}}$, the PEWEC reference numerical model returns the corresponding power matrix $\mathcal{P}(H_s, T_e, d) \in \mathbb{R}^{N_{H_s} \times N_{T_e}}$.

In the present study, the power matrix is built from N_c 141 sea states, which are representative of European oceanic sites. These N_c 141 sea states represent the number of H_s and T_e combinations. In particular, the H_s spans the range [0.5–8.5] m while T_e is in [3–18] s. A power value is associated with each of these combinations, resulting in a 141 × 3 matrix. The region covered by the resulting set of 141 (N_c) waves is shown in Fig. 4.

A broad design space was explored for the device parameters in order to cover all feasible solutions for every design variable. Lower and upper bounds were defined based on expert design experience (listed in Table 3). A Latin Hypercube Sampling (LHS) space-filling strategy was adopted to generate a total of 175,000 samples, of which 144,505 turned out to be *infeasible* design solutions and 30,495 *feasible* ones. LHS [98] is a stratified sampling technique that splits the range of each input variable into N equally probable intervals and draws exactly one sample from each interval, ensuring uniform coverage of the input space with fewer samples than purely random approaches. The full dataset generation process required 2621.25 min (*i.e.*, 1 day, 19 hours, 41 min and 15 s) on 70 workers running in parallel.

3.2.1. Classification task

We aim to identify the *feasible* and *infeasible* solutions during the *classification* task. As previously stated, from the 175,000 samples, 83% of the samples are *infeasible* solutions, and 17% are *feasible* ones. Based on the ratio (almost one *feasible* solution per five *infeasible* solutions, *i.e.*, 1:5 ratio), we can state that it is an imbalanced classification problem with a severe imbalance [99]. In our binary classification problem, we assign the minority class (*i.e.*, *feasible* solutions) the class 1 and the majority (*i.e.*, *infeasible* solutions) the class 0 in such a way that we are compliant with the evaluation metrics [89].

There are three main approaches for overcoming the challenge of imbalanced distribution [99]: i) data-level techniques, which adjust the dataset by adding/removing samples to balance the class distributions; ii) algorithm-level approaches, which modify the internal learning mechanisms of the algorithms to mitigate the bias towards the majority class and improve their performance on skewed data; and iii) hybrid strategies, which aim to combine the strengths of the two techniques. There are two main approaches for data-level techniques: adding new samples to the minority class (*i.e.*, oversampling) and removing samples from the majority one (*i.e.*, undersampling). For the algorithm-level techniques, the cost-sensitive approach is one of the most popular [100]. This approach employs a larger penalty (or class weights) for the minority class, boosting the minority class's importance during the training phase. This is because the algorithm should focus on minimizing the global cost associated with classification errors. In this work, we employ the cost-sensitive approach to handle class imbalance.

Classification data split and pre-processing. The 175,000 samples dataset was divided into *classification training set* (80%—140,000 samples) and *classification test set* (20%—35,000 samples) using a stratified holdout split. We employed a stratified split to preserve the proportional distribution of feasible and unfeasible samples between the *classification training set* and the *classification test set*. Moreover, a 10-fold repeated stratified cross-validation scheme was adopted during the *classification training phase*.

Before training the classifier algorithms, we normalized the *classification training* dataset based only on the training folds of each cross-validation split. This was achieved using the *pipeline* method of the *sklearn* Python library [101]. We normalize the *classification training set* of each split to the range 0 to 1 using *Min-Max* normalization:

$$x_{scaled} = \frac{x - \min(x)}{\max(x) - \min(x)}, \tag{13}$$

where x denotes the vector of values to be normalized, and $\min(x)$ and $\max(x)$ correspond to the minimum and maximum values for each parameter in the *classification training* set.

3.2.2. Regression task

The dataset of *feasible* designs comprises 30,495 samples. We aim to predict the site-independent power matrix $\mathcal{P}(H_s, T_e, d) \in \mathbb{R}^{N_{H_s} \times N_{T_e}}$ of a viable design for a particular set of the thirteen PEWEC design parameters (see Table 3). To generalize the problem and make the algorithms agnostic to the H_s and T_e, we decided to structure problem so that the algorithm input consists of the thirteen PEWEC design parameters together with H_s and T_e and, as output, the corresponding power. Therefore, the original problem, which consisted of input n x 13 and output of n x 141 x 3, where 13 is the number of PEWEC design parameters, and 141 correspond to the combinations of H_s and T_e, is transformed to input n x 15 and output n x 1, where the output is the power. In this manner, the number of samples is increased by 141 (*i.e.*, the number of H_s and T_e combinations sampled), resulting in 4,299,795 samples.

Regression data split and pre-processing. A *designID* auxiliary column was added to identify rows belonging to the same input design. We employed a simple holdout approach with a grouped split by *designID* to divide the dataset into the *regression training set* (80% - 3439836 samples) and the *regression test set* (20% - 859959 samples). This grouping strategy ensures that no design is shared between the two sets. Therefore, the regression algorithms are trained on one subset of designs (*i.e.*, the *regression training set*) and evaluated on entirely unseen designs (*i.e.*, the *regression test set*). During the *regression training phase*, we performed preliminary experiments comparing 10-fold grouped cross-validation with the holdout approach, but observed no differences. Consequently, we kept the holdout grouped split by *designID* and used the last 10% of the *regression training set* as the *regression validation set*. As with the classification tasks, *Min-Max* normalization was applied to scale all input features in the range from 0 to 1 prior to training the regression algorithms.

3.3. Surrogate model - classifier algorithms

This section will provide a detailed explanation of the *Classification algorithms* employed within the *Surrogate Model* module of the proposed methodology depicted in Fig. 2. To identify whether the WEC design parameter combination leads to a feasible design or not, we designed and compared the performance of five ML algorithms that have shown excellent performance in the literature for binary classification problems: Logistic Regression (LR), Random Forest (RF) classifier, Support Vector Classifier (SVC), Extreme Gradient Boosting (XGBoost) and Feedforward Neural Network (FNN) classifier.

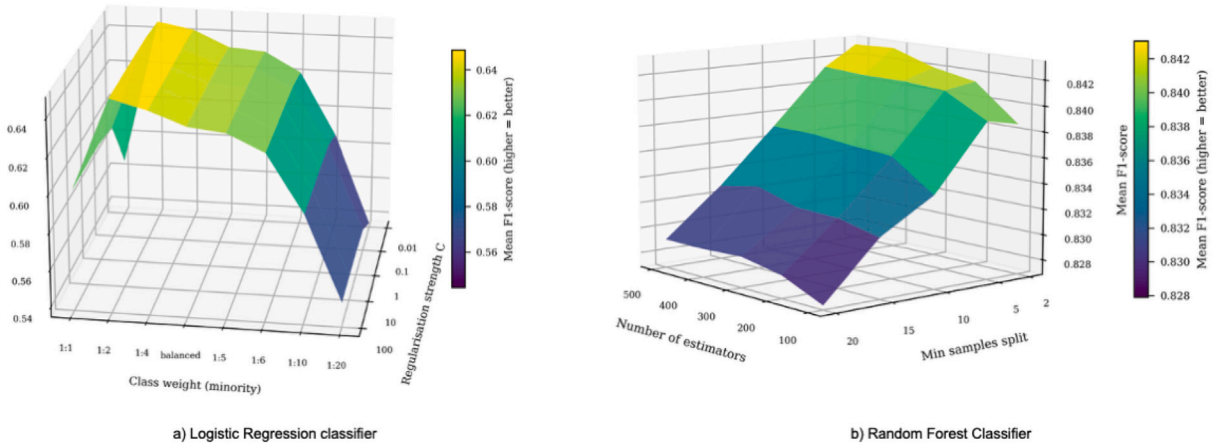
As shown in Fig. 2, during the *classification training phase*, the *classification algorithms* were trained with the corresponding *classification training set* after *pre-processing (normalization)*. Furthermore, a 10-fold repeated stratified cross-validation scheme was adopted, as already introduced in Section "Classification Data Split and Pre-processing". We searched and identified the best architecture for each of the proposed *classification algorithms* via a trial-and-error approach with a grid search algorithm, evaluating several possible hyperparameter configurations for each *classification algorithm*. Despite the fact that grid search can be computationally expensive (particularly when dealing with high-dimensional hyperparameter spaces or large datasets), it is one of the most widely used methodologies in the literature for hyperparameter optimization in machine learning [83,102,103]. This is due to its guaranteed repeatability as well as its simplicity and exhaustiveness compared to other more complex methodologies [103,104]. During the fine-tuning of the hyperparameters of each algorithm, the best hyperparameter combination of each algorithm was chosen based in the highest mean F1-score on the 10-fold repeated stratified cross-validation scheme.

During the *classification test phase* the *trained classification algorithms* were fed with the *classification test set* accordingly, determining whether the WEC design parameter combination led to feasible or unfeasible solutions (as shown in Fig. 2).

3.3.1. Logistic regression classifier

LR is one of the most employed statistical algorithms for modeling the probability of a binary problem based on one or more predictor variables [105]. The target variable can take only two possible values, such as yes/no or success/failure. More details of the LR structure can be found in [88].

The LR classifier was implemented in Python using the *sklearn* library [101]. We tuned the hyperparameters through a trial-and-error



a) Logistic Regression classifier b) Random Forest Classifier

Fig. 5. Adjusting hyperparameters of a) Logistic regression and b) Random Forest using grid search.

Table 4
LR classifier hyperparameters.

Hyperparameter	Value
C	0.1
Solver	*lbfgs*
Penalty	*L2*
Class Weight	1:2

Table 5
RF classifier hyperparameters.

Hyperparameter	Value
Number of estimators	500
Max Features	*sqrt*
Min Samples Split	2
Class Weight	1:1

approach using a simple grid search. For the *C* hyperparameter, we searched over values from 0.01 to 100. For the *solver*, we explored solutions employing *saga*, *lbfgs*, *newton-cg*, and *newton-cholesky*. Regarding the *penalty*, we tested solutions with no penalty, *L1*, and *L2* regularization, as well as *Elastic-Net* regularization. It is important to highlight that the *Elastic-Net* and *L1* regularization are only applicable when employing the *saga* solver. Additionally, as previously explained, given that the dataset is unbalanced, we searched for different *class weights* for the algorithm. We tested *class weights* from 1:1 to 1:10 and *balanced* weighting. The *balanced* weighting means assigning weights that are inversely proportional to the class frequencies observed in the input data. Fig. 5(a) shows an example of the landscape of the grid search for tuning the LR *class weight* and the regularization *C* in terms of the F1-score. As can be observed, the performance of the models was affected mainly by the *class weights*, and for this case study, the best performance was achieved with *C* equal to 0.1 and the *class weight* of 1:2 (as reported in Table 4). Table 4 describes the LR hyperparameter configuration that showed the best performance. The LR model configuration that showed the best performance employed the *lbfgs solver*, *L2* regularization, *C* equal to 0.1 and a *class weight* of 1:4.

3.3.2. Random forest classifier

RF is an ensemble learning method composed of multiple Decision Trees. By combining the forecasts of many decision tree classifiers, the RF algorithm can perform a more accurate classification than that obtained with any singular tree. Within the RF architecture, each singular decision tree is constructed independently via a deterministic procedure that selects a random subset of the training data as input. Typically, when ensemble Decision Trees are employed for classification tasks, they are known as Random Forest Classifiers. More details of the RF algorithm employed in this study can be found in [106].

The RF classifier algorithm was implemented in Python using the *sklearn* library [101]. The RF classifier has four parameters to be optimized: i) *number of estimators*, which determines the number of trees; ii) *max features*, which represents the maximum number of features a tree can employ; iii) *min samples split*, which represents the minimum number of samples required to split a node; and iv) *class weight*, which allows balancing the class weights. As before, we adopted a grid search approach to find the best hyperparameters for the RF. We searched values ranging from 100 to 500 for the *number of estimators*. The *min samples split* hyperparameter was explored with values in the interval from 1 to 20. Lastly, for the *max features*, we tested the *sqrt* and *log2* cases, which means considering the square root or the $\log_2$ of the number of features as *max features*. For the *class weight*, we tested *balanced* and *balanced subsample* weights, as well as weights in the interval from 1:1 to 1:10. The *balanced* weight means assigning weights to the classes inversely proportional to the class frequencies observed in the input data. On the other hand, the *balanced subsample* weight differs from the *balanced* case in that the weights are calculated for the bootstrap sample of every tree. Fig. 5(b) depicts the landscape analysis of model performance when varying the *number of estimators* and *min samples split*, based on the F1-score. We observed that smaller *min samples split* values and larger *number of estimators* led to better results. Nonetheless, it is important to highlight that across the various levels, the difference between *number of estimators* 400 and 500 is quite small, and it seems like a plateau. The RF classifier model that demonstrated the best performance in our case employed 500 *estimators*, *min samples split* equal to 2, *sqrt* for *max features*, and a *class weight* of 1:1 (as reported in Table 5).

3.3.3. Extreme gradient boosting classifier

Extreme Gradient Boosting (XGBoost) is a supervised ML algorithm that leverages an ensemble framework built upon the gradient boosting algorithm. XGBoost employs decision trees as base estimators and integrates regularization techniques to improve the model's generalization capabilities. The XGBoost algorithm is widely employed to obtain state-of-the-art results on many machine learning challenges, showing good results with low computational cost [107,108]. A detailed description of the XGBoost algorithm is provided in [109].

We implemented the XGBoost algorithm in Python using the *xgboost* library [109]. The XGBoost algorithm has eight main parameters to tune: i) *Min Child Weight*, ii) *Max Depth*, iii) γ, iv) λ, v) α, vi) *subsample*, vii) *Col Sample by Tree*, and viii) *Scale Pos Weight*. For tuning

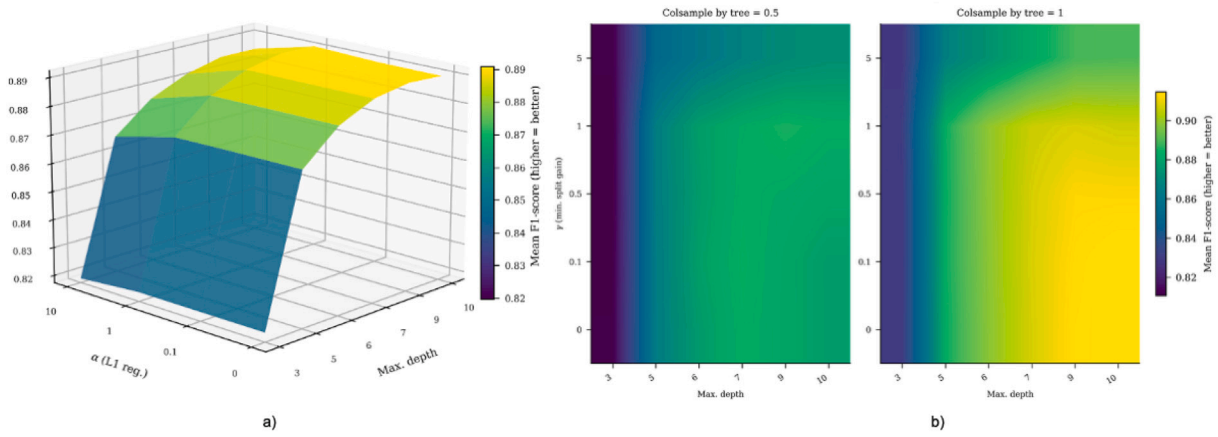


Fig. 6. Adjusting hyperparameters of the XGBoost classifier using grid search.

Table 6
XGBoost classifier hyperparameters.

Hyperparameter	Value
Min Child Weight	3
Max Depth	10
γ	0.1
λ	0
α	1
Subsample	1
Col Sample by Tree	1
Scale Post Weight	3

the hyperparameters, we exploited the same grid search approach previously highlighted. We searched *Min Child Weight* in the range from 1 to 10. For the *Max Depth*, we tested values ranging from 3 to 10. Furthermore, we explored α and λ in the interval from 0.0 to 10.0. For γ, we searched values in the range from 0.0 to 5.0. Additionally, for *subsample* and *Col Sample by Tree*, we tested values of 0.5 and 1.0. For *Scale Pos Weight*, we explored values in the interval from 1 to 10 and 25, 50 and 75. Fig. 6 illustrates the landscape of grid search for tuning the *max depth* and α (Fig. 6a) as well as *max depth*, γ and *Col Sample by Tree* (Fig. 6b). Fig. 6(a) clearly depicts that performance improves significantly for larger values of *Max Depth*. Similar behavior is observed in Fig. 6(b). Additionally, a slight improvement is observed for bigger values of α up to 1, which then worsens for α equal to 10, as shown in Fig. 6(a). For the *Col Sample by Tree* (see Fig. 6b), it can be seen that better results are achieved overall with this hyperparameter equal to 1.0 rather than 0.5. Instead, similar performance is observed for γ values ranging from 0 to 1, which degrades for values equal to 5, as depicted in Fig. 6(b). Table 6 reports the XGBoost hyperparameter configuration that showed the best performance. The XGBoost model that presented the best performance has *Col Sample*, *subsample* and α equal to 1, λ 0, γ 0.1, *Max Depth* 10, *Min Child Weight* 3 and *Scale Pos Weight* 3.

3.3.4. Support vector classifier

SVC is a nonlinear classification algorithm designed to find a hyperplane in a high-dimensional space that best captures the underlying relationship between input variables and corresponding outputs. This is achieved by mapping the input data x into a high-dimensional feature space using a nonlinear mapping, followed by performing a linear classification within the transformed space. For a detailed description of the SVC structure, refer to [110].

The SVC was implemented in Python using the *sklearn* library [101]. We employed the Radial Basis Function (RBF) kernel. For tuning the hyperparameters (*i.e.*, C penalty, RBF kernel coefficient γ, and *class weights*), we utilized a grid search approach. We searched for values ranging from 0.01 to 50 for the C penalty parameter. For the RBF kernel coefficient γ, we tested values ranging from 0.01 to 10. We also tested γ equal to *auto* and *scale*. In the first case, γ is equal to $1/n_{features}$, while in the second case it is $1/(n_{features} * \mathrm{Var}(x))$, where $n_{features}$ is the number of input features, and $\mathrm{Var}(x)$ is the variance of the input dataset. Additionally, as previously explained, given that the dataset is unbalanced, we searched for different *class weights* for the algorithm. We tested *class weights* from 1:1 to 1:10 and *balanced* weighting. Fig. 7 shows as an example the landscape of the grid search for tuning the SVC's γ and C regularization. As observed, for growing γ and C better results were observed. Nonetheless, there was a plateau when reaching C equal to 50 as well as γ reaching 1. In the case of γ, after we found the best hyperparameter configuration, which employed γ of 1 and C of 50, we tried increasing γ further, which did not produce better results. The SVC hyperparameter configuration that showed the best performance employed the RBF kernel, C equal to 50, γ of 1 and *class weight* 1:1 (as reported in Table 7).

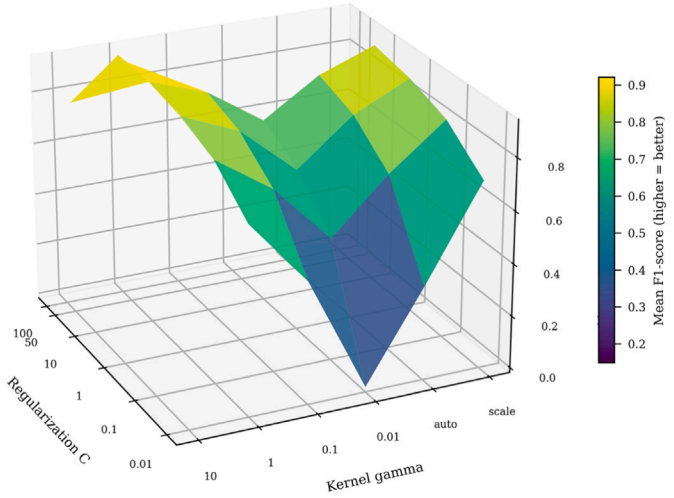


Fig. 7. Adjusting hyperparameters of a Support vector classifier using grid search.

3.3.5. Feedforward neural network classifier

The FNN (or Multilayer Perceptron) is a neural network architecture characterized by unidirectional information flow: from the input layer

Table 7
SVC classifier hyperparameters.

Hyperparameter	Value
Kernel	RBF
C	50
γ	1
Class Weight	1:1

through the hidden layers to the output layer. An FNN consists of an input layer, one or more hidden layers, and an output layer. The output of each neuron in the FNN is computed by applying an activation function to the weighted sum of its inputs. Typically, a bias term is added to this weighted sum to enhance the model's flexibility. The output layer employs a $sigmoid$ activation function to produce binary outputs in a binary classification problem. A detailed description of the FNN algorithm structure is provided in [111].

The FNN was built in Python using the *Keras* library with TensorFlow backend [112]. We employed the Rectified Linear Unit ($ReLu$) activation function for the hidden layer due to its simplicity, extensive use and proven efficacy [113]. For the output layer, we employed the $sigmoid$ activation function and employed the binary cross-entropy as the loss function. We searched for different architectures, employing from one to three hidden layers. We tuned the hyperparameters, *i.e.*, *number of units*, *batch size*, and *class weight*, through a trial-and-error approach using a grid search algorithm, as was previously mentioned. We explored *batch sizes* in the range 16 to 64. For the *class weight*, we tested models with *class weights* in the interval 1:1 to 1:5. Furthermore, we searched the *number of units* in the range from 16 to 128. For the models employing two and three hidden layers, we tested two main different configurations: one in which the n hidden layer employed the same number of units as in the previous hidden layer, and the second configuration in which the n hidden layer employed half the units of the previous layer. For example, when the first hidden layer employs 16 units, we tested values of 16 and 8 units for the second hidden layer. Similarly, when the second hidden layer utilizes 16 units, we evaluated the third hidden layer employing 16 and 8 units. Furthermore, when the second hidden layer uses 8 units, we tested the third hidden layer with 8 and 4 units. We adopted the Adaptive Moment Estimation (Adam optimizer) [114] as the optimization algorithm. Additionally, the early stopping strategy [115] was employed to prevent overfitting. This strategy stops the training process when no further improvement is observed in the validation set after a predefined number of iterations. Apart from reducing the risk of overfitting, early stopping also contributes to a significant reduction in the overall training time. Fig. 8 depicts the landscape of the grid search for tuning the FNN's *number of hidden layers*, *number of units* (or neurons) per layer, and *batch size*. As observed, the architectures with only 16 neurons in Layer 1 led to suboptimal F1-scores. The highest F1-scores were consistently achieved by deeper and more densely parameterized architectures, highlighting that, for this use case, both the depth and having enough "intermediary" width (*i.e.*, neurons in the middle hidden layers) are required for the algorithm to capture the decision boundaries effectively. The FNN model showed the best performance using a *batch size* of 64, *class weight* of 1:2, and 3 hidden layers, with 32 *units* in each of the three hidden layers. Table 8 reports the FNN classifier hyperparameters employed which showed the best performance, while Table 9 describes the best FNN classifier layers configuration found.

3.4. Surrogate model - regression algorithms

In this section, we will provide a detailed explanation of the *regression algorithms* utilized within the *Surrogate Model* of the proposed solution illustrated in Fig. 2. After the classifier determines whether a given set of design parameters yields a feasible design or not (detailed in Section 3.3), for those feasible, the *regression algorithms* predict the site-independent power matrix. During inference, the regression stage is activated only when the classifier identifies the candidate configuration as feasible. If the candidate is classified as infeasible, the regression stage is not executed, the corresponding power matrix is considered undefined, and no derived performance indicator is calculated. If an actually infeasible configuration is incorrectly classified as feasible, the regressor may still return a numerical power matrix because it does not independently enforce the original geometric, mechanical, and hydrodynamic feasibility constraints. To this end, we employed and compared four ML regression algorithms that demonstrated excellent behavior as surrogates for WEC modeling: Random Forest (RF) regressor, Gaussian Process Regression (GPR), Radial Basis Function Neural Network (RBFNN), and Feedforward Neural Network (FNN) regressor.

As depicted in Fig. 2, during the *regression training phase*, the *regression algorithms* were trained on the corresponding *regression training set* after *pre-processing* (as detailed in Section "Regression Data Split and Pre-processing"). We employed a trial-and-error approach following the grid-search technique to find the optimal architecture for each of the proposed *regression algorithms*. During the fine-tuning of the hyperparameters of each algorithm, we compared and chose the best hyperparameter configuration for each model based on its performance on the *regression validation set* according to the highest Coefficient of determination (R^2) and lowest Normalized Root Mean Squared Error ($NRMSE$). During the *regression test phase*, the *trained regression algorithms* were fed the *regression test set* to obtain the site-independent power matrix $\mathcal{P}(H_s, T_e, d) \in \mathbb{R}^{N_{H_s} \times N_{T_e}}$, as shown in Fig. 2.

3.4.1. Random forest regressor

Similar to the RF algorithm introduced in Section 3.3.2, when Decision Trees are employed for regression tasks, they are typically referred to as Regression Trees, while their ensemble counterpart is known as the Random Forest Regressor. A detailed description of the RF regressor structure used in this study can be found in [106].

We implemented the RF regressor algorithm in Python using the *sklearn* library [101]. Different from the RF classifier, the RF regressor has three hyperparameters to be optimized: i) *number of estimators*, ii) *max features*, and iii) *min samples split*. We adopted a trial-and-error approach with a grid search algorithm to fine-tune these hyperparameters. We searched for values ranging from 100 to 800 for the *number of estimators*. The *min samples split* hyperparameter, which is fundamental as a stopping criterion to prevent overfitting, was explored from 0.005% to 100% of the total samples. Lastly, we investigated values of *max features* ranging from 10% to 100%. Fig. 9 illustrates the landscape of the grid search tuning the RF's hyperparameters based on NRMSE. As can be observed in Fig. 9(a), the results improved for an increasing percentage of *max features*, but after 70% the results worsened. On the other hand, there is almost no noticeable improvement in terms of *number of estimators*. Likewise, as illustrated in Fig. 9(b), there is no appreciable improvement for varying *number of estimators* for fixed *min samples split*. Nonetheless, Fig. 9(b) highlights that the NRMSE significantly improved when decreasing *min samples split*. The RF model configuration showing the best performance was: *number of estimators* equal to 600; *max features* 70%; and *min samples split* 0.005% (as described in Table 10). Despite the fact that there was no visible improvement in Fig. 9 for a given configuration of *min samples split* and *max features*, there was a slight improvement with increasing *number of estimators* but only up to a certain limit. This is the reason why the best performance was shown by *number of estimators* 600 and not at the limit we investigated (800). It is also worth mentioning that, when employing a larger *number of estimators* for fixed *min samples split* and *max features*, the training time of the models increased.

3.4.2. Gaussian process regression

GPR is an ML method grounded in Bayesian learning principles. The fundamental idea underlying Gaussian processes is that any finite collection of observations x_i, $i = 1, .., n$, follows a joint normal distribution, characterized by a mean function $m(x)$ and a covariance matrix or kernel

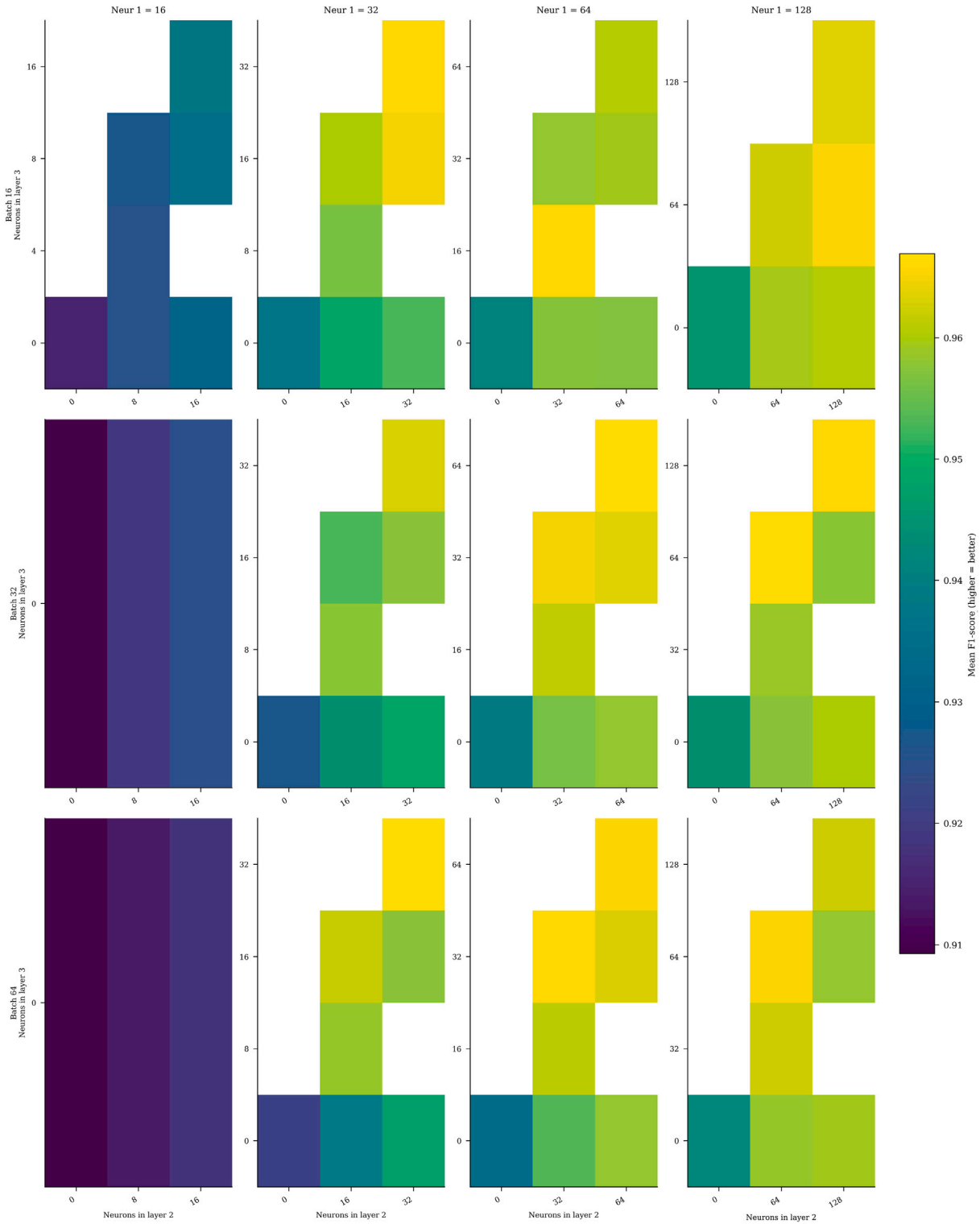


Fig. 8. Adjusting the hyperparameters of a Feedforward Neural Network classifier using grid search.

$k(x, x')$. The hyperparameters of the kernel are optimized by maximizing the log-marginal likelihood. Given that the likelihood maximization is a non-convex optimization problem, standard gradient-based methods are used to perform the optimization. However, when applying a Gaussian process to a dataset of size n, its inference leads to a computational complexity of $O(n^3)$, representing a significant drawback for "large datasets" (*i.e.*, those that comprise over a thousand data points) [116]. Considering that the *regression training set* consisted of 3,439,836 samples, we adopted the Stochastic Variational Gaussian Process (SVGP) model [116] to address this limitation. The SVGP unlocks scalability by introducing inducing points within a stochastic variational inference framework. These inducing points can be initialized from the training dataset using the K-Means clustering algorithm. A detailed description of the structure of the SVGP model employed can be found in [116].

Table 8
FNN classifier hyperparameters.

Hyperparameter	Value
Optimizer	Adam
Epochs	100
Batch Size	64
Stopping Criteria	early stopping with patience equal to 10
Class Weight	1:2

Table 9
FNN classifier layers.

Layer Type	Units	Activation
Dense	32	ReLu
Dense	32	ReLu
Dense	32	ReLu
Dense	1	sigmoid

Table 10
RF regressor hyperparameters.

Hyperparameter	Value
Number of estimators	600
Max Features	70%
Min Samples Split	0.005%

Table 11
SVGP regressor hyperparameters.

Hyperparameter	Value
Optimizer	Adam
Epochs	2000
Kernel	RBF
Batch Size	1500
Inducing Points	2000

We implemented SVGP in Python using the *GPflow* library [117]. For tuning the SVGP hyperparameters, *i.e.*, *inducing points*, *kernel*, and *batch size*, we exploited the grid search approach previously highlighted. We searched for values ranging from 1000 to 2000 for the *inducing points*. For the *kernel*, we tested the RBF, *squared exponential*, and *Mattern 32*. For the *batch sizes*, we explored values ranging from 1000 to 2000. We trained the SVGP using the Adam optimizer. Fig. 10 depicts the landscape of the grid search for tuning the SVGP's hyperparameters based on NRMSE. We observed that more inducing points led to better results (but at the expense of larger training times). Furthermore, it can be observed that *inducing points* was the parameter that most affected the performance. The variation based on the *batch size* and *kernel* was less significant. However, for a given number of *inducing points*, better results were seen for larger *batch sizes* and *rbf* as the *kernel* (despite the fact that it is not distinguishable in Fig. 10). Nonetheless, it is important to highlight that this improvement in terms of larger *batch sizes* for the *rbf* case was seen up to a certain point, which was a *batch size* equal to 1500, after which the results reached a plateau and did not improve further. The SVGP configuration that showed the best performance employed *batch size* 1500, a RBF kernel, and 2000 *inducing points* (as reported in Table 11).

3.4.3. Feedforward neural network regressor

FNN, already introduced in Section 3.3.5, can also be employed for regression tasks. Differently from its implementation for classification tasks, when employed for regression, typically the loss function is the *Mean Squared Error*, the activation function of the output layer is a *linear* function and the output values are continuous. For a detailed description of the FNN model, refer to [111].

In our work, we used the *Keras* library with TensorFlow backend [112] for developing the FNN in Python. We employed the hyperbolic tangent (tanh) as the activation function for the hidden layers because of its extensive use and demonstrated effectiveness [118]. On the other hand, we used a *linear* activation function for the output layer. We searched for different architectures, ranging from one to three hidden layers. We tuned the hyperparameters, *i.e.*, *number of units* and *batch size*, through a trial-and-error approach using a grid search algorithm, as previously mentioned. We explored *batch sizes* of 128 and 256. Furthermore, we searched the *number of units* in the range from 16 to 512. For the models employing two and three hidden layers, we tested two main configurations as explained in Section 3.3.5: one in which the n hidden layer employed the same number of units as the previous hidden layer, and the second configuration in which the n hidden layer employed half the units of the previous layer. We adopted the Adam Optimizer as the optimization algorithm. Additionally, the early stopping strategy was employed to prevent overfitting. Fig. 11 shows the landscape of the grid search for tuning the FNN's hyperparameters. We observed that across all the subplots employing only one layer (thus Neurons Layer 2 and Neurons Layer 3 have 0 neurons) yielded the worst NRMSE (indicated in purple). Furthermore, when increasing the number of neurons in Layer 1 (up to 64 neurons) the overall performance across

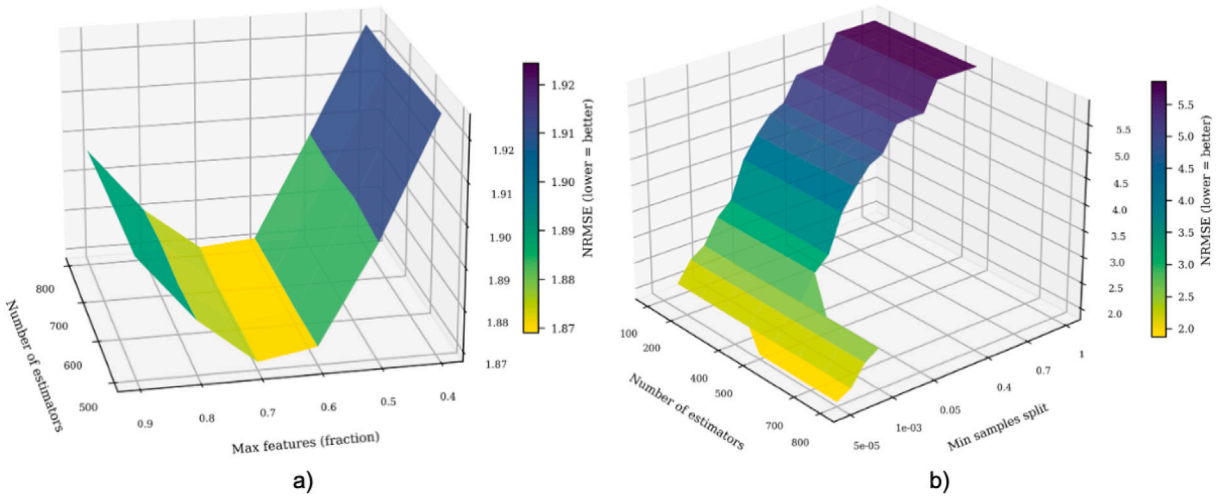


Fig. 9. Adjusting the hyperparameters of a Random Forest regressor using grid search.

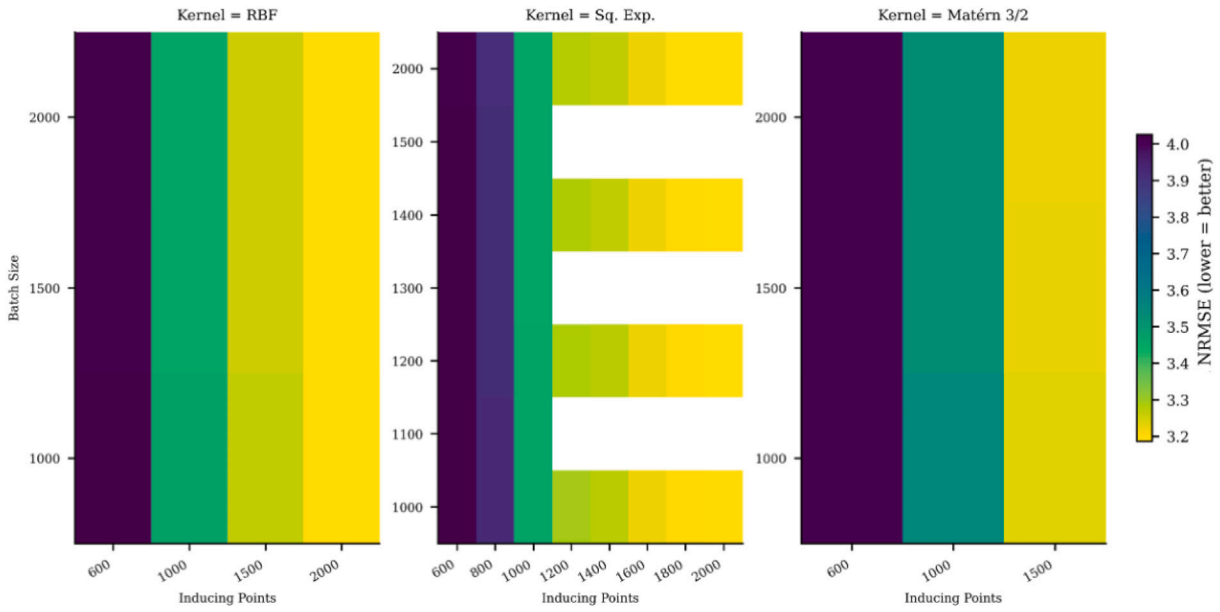


Fig. 10. Adjusting hyperparameters of Stochastic Variational Gaussian process model using grid search.

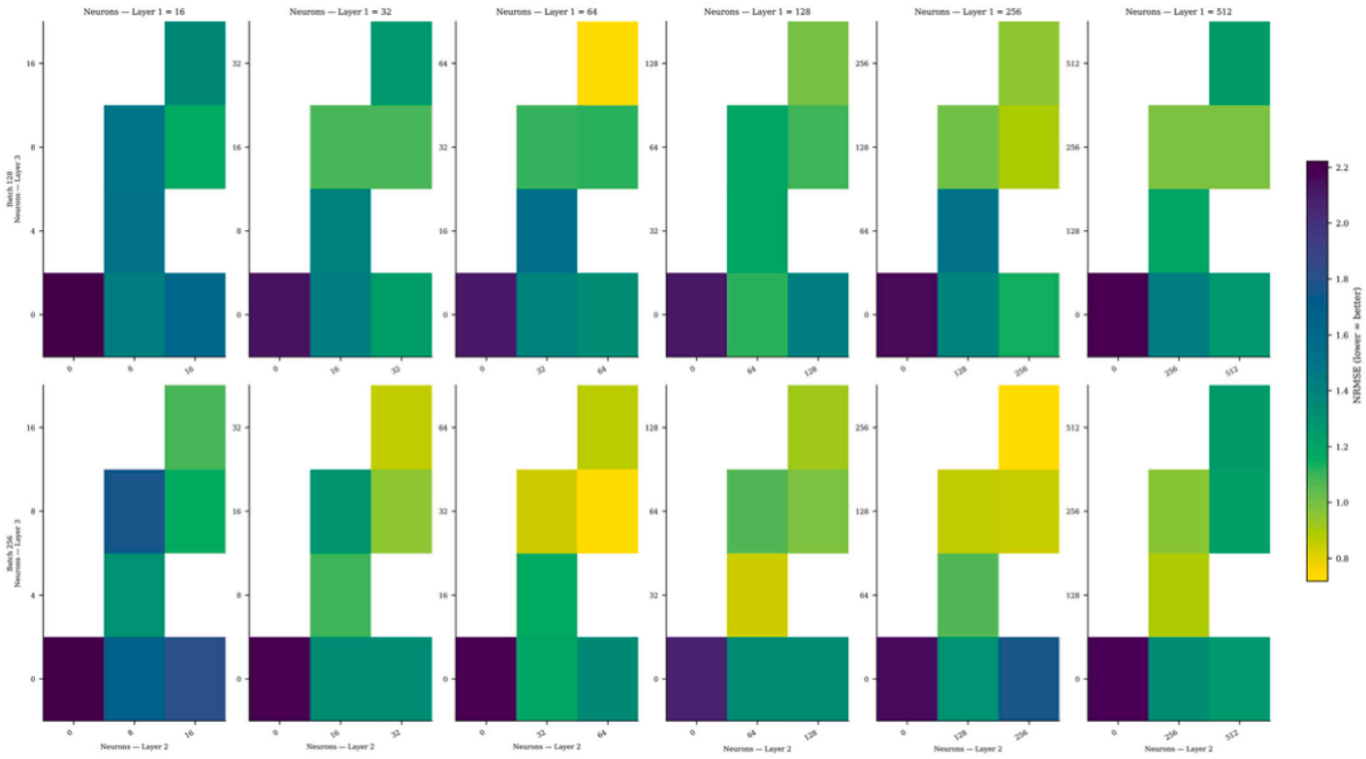


Fig. 11. Adjusting hyperparameters of a Feedforward Neural Network regressor using grid search.

the hyperparameter space improved. Nonetheless, when moving to the largest number of neurons evaluated in Layer 1 (*i.e.*, 512 neurons), a degradation in the overall performance, which may be due to overfitting the training data. The FNN hyperparameter configuration showed the best performance using a *batch size* of 128, 3 hidden layers, with 64 *units* in each of the three hidden layers. Table 12 reports the FNN regressor hyperparameters employed that showed the best performance, while Table 13 describes the best FNN regressor layers configuration found.

Table 12
FNN regressor hyperparameters.

Hyperparameter	Value
Optimizer	Adam
Epochs	100
Batch Size	128
Stopping Criteria	early stopping with patience equal to 10

Table 13
FNN regressor layers.

Layer Type	Units	Activation
Dense	64	tanh
Dense	64	tanh
Dense	64	tanh
Dense	1	linear

Table 14
RBFNN regressor hyperparameters.

Hyperparameter	Value
Optimizer	Adam
Epochs	100
Batch Size	256
RBF units	100
β	1.0
Stopping Criteria	early stopping with patience equal to 10

3.4.4. Radial basis function neural network

RBFNN is a particular type of ANN, characterized by its generalization ability and simplicity compared to other networks [119]. The RBFNN is a three-layer architecture: the input layer, the hidden layer, and the output layer. The hidden layer comprises several *RBF units*, typically employing a Gaussian kernel. More details on the structure of the RBFNN employed can be found in [120].

The RBFNN was implemented in Python using the open-source *RBF layer* [121] and the *Keras* library with TensorFlow backend [112]. As for the other ML algorithms, we adopted a grid search approach to find the best hyperparameters (*i.e.*, *batch size*, β and *number of RBF units*) for the RBFNN. We searched for *batch sizes* of 128 and 256. For β, we studied values ranging between 0.1 and 2.0. Moreover, we investigated values from 50 to 600 for the *number of RBF units*. Like in the FNN, we used Adam as the optimization algorithm and employed the early stopping technique. Fig. 12 illustrates the landscape of the grid search of the RBFNN's hyperparameters. We can observe a strong dependence on the *batch size*, with models employing *batch size* equal to 256 showing lower NRMSE error overall. Furthermore, it can be seen that the algorithms achieved optimal performance for *RBF units* in the range 100–200 and β lower than 1.5. Additionally, when employing more than 300 *RBF units* and larger β (upper right part of both subplots), the models showed severe performance degradation. The RBFNN configuration that demonstrated the best performance was: β equal to 1.0, 100 *RBF units* and *batch size* 256 (as reported in Table 14).

Table 15
Confusion matrix for a binary classifier.

		Predicted Class	
		Positive	Negative
Observed Class	**Positive**	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

4. Results and discussion

This section presents the experimental results obtained for the proposed methodology. Firstly, we introduce the metrics for assessing the surrogate model components (*i.e.*, classifier and regressor algorithms). Furthermore, we show the results obtained for the classification and regression tasks. Last but not least, we evaluate and compare the performance of employing the reference PEWEC numerical model and the proposed surrogate model to assess the performance of the PEWEC device for four representative deployment sites.

4.1. Performance metrics

In a binary classification scenario, the results of a classification process can be categorized into four groups, as illustrated in the confusion matrix presented in Table 15: the samples that are correctly classified as positives (*True Positive - TP*) and as negatives (*True Negative - TN*), and the samples that are misclassified as positives (*False Positive - FP*) and negatives (*False Negative - FN*).
The *accuracy* (see Eq. 14):

$$accuracy = \frac{TP + TN}{TP + FN + TN + FP} \tag{14}$$

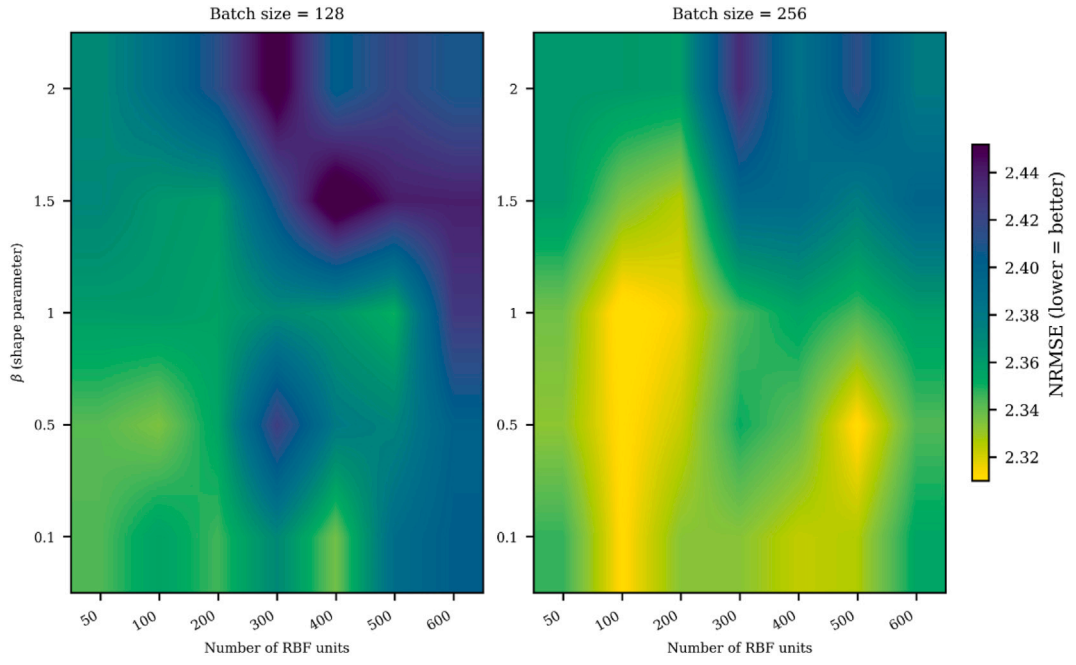


Fig. 12. Adjusting the hyperparameters of a Radial Basis function Neural Network regressor using grid search.

Accuracy is generally the most widely employed metric for assessing classification performance. Nonetheless, in an imbalanced classification problem in which there are many more samples of one class (majority) than of the other (minority), accuracy may not be a proper measure since it is disproportionately influenced by the majority class, rendering the contribution of the minority class nearly negligible. Therefore, other metrics derived from the confusion matrix (shown in Table 15) are also employed. First, the *recall*, or true positive rate (see Eq. 15):

$$recall = \frac{TP}{TP + FN}. \tag{15}$$

Then, the *precision*, or positive predictive value (see Eq. 16):

$$precision = \frac{TP}{TP + FP}. \tag{16}$$

Furthermore, the *F1-Score* or *F1-measure* provides a unified metric that integrates both *precision* and *recall*:

$$F1\text{-}score = 2 * \frac{precision * recall}{precision + recall}. \tag{17}$$

The *F1-score*, which assigns equal weights to *precision* and *recall*, is the most commonly adopted metric when addressing imbalanced classification problems [122]. Moreover, the *receiver operating characteristic area under the curve* (ROC AUC) provides a score for directly comparing binary classifiers. However, for severely imbalanced classification problems, the *precision-recall area under the curve* (PR AUC) score is recommended, since the ROC AUC may yield an overly optimistic assessment of model performance [123]. The scores for both ROC AUC and PR AUC range from 0.0 to 1.0, with 1.0 being the best classifier:

$$ROC\ AUC = \int_0^1 \mathrm{TPR}(\mathrm{FPR})\, d(\mathrm{FPR}) \tag{18}$$

$$PR\ AUC = \int_0^1 \mathrm{PPV}(\mathrm{TPR})\, d(\mathrm{TPR}) \tag{19}$$

where TPR is the *true positive rate* or *recall*, FPR is the *false positive rate* which is equal to $FPR = FP/(FP + TN)$, and PPV is the *positive predictive value* or *precision*.

On the other hand, to evaluate the performance of the regression algorithms, we relied on three metrics commonly employed for surrogate modeling [50,124]: i) the *Coefficient of determination* (R^2), ii) *Normalized Root Mean Squared Error* (*NRMSE*), and iii) *Mean Absolute Error* (*MAE*). R^2 quantifies the percentage of the variance in the true sample that can be explained by the predicted one. The $NRMSE$ is the normalized discrepancy between the true sample and the predicted one. The MAE measures the average of the absolute errors. These metrics are described by the following equations:

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_{test,i} - y_{fore,i})^2}{\sum_{i=1}^{n} (y_{test,i} - \overline{y}_{test})^2} \tag{20}$$

$$NRMSE = \frac{100}{y_{test,\max} - y_{test,\min}} \sqrt{\frac{\sum_{i=1}^{n} (y_{fore,i} - y_{test,i})^2}{n}} \tag{21}$$

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |y_{fore,i} - y_{test,i}| \tag{22}$$

where $\overline{y}_{test}$ denotes the mean value, $y_{test,\max}$ and $y_{test,\min}$ are the maximum and minimum observed values, n is the number of predictions, and y_{test} and y_{fore} are the observed and forecasted values, respectively. R^2 determines the correlation between the observed and the predicted values, where 1 represents a total correlation, and smaller values indicate weaker correlations. On the other hand, the $NRMSE$ is expressed as a percentage, with smaller values indicating that the surrogate model predicts better. According to [124], surrogate models with an $NRMSE$ lower than 10% have good predictive capabilities, while those with an $NRMSE$ lower than 2% demonstrate excellent predictive capabilities. Last but not least, the MAE quantifies the average absolute error, with lower values indicating better performance. In the case under study, the PEWEC observed power values ranged up to 151 kW.

Table 16
Classification and regression machine learning algorithms technical settings.

Acronym	Full name	Type	Hyper-parameters
LR	Logistic Regression	Classification	C = 0.1, solver = 'lbfgs', penalty = 'l2', class_weight = {0:1.0, 1:2.0}
RF	Random Forest	Classification	n_estimators = 500, max_features = 'sqrt', min_samples_split = 2, class_weight = {0:1.0, 1:1.0}
XGBoost	Extreme Gradient Boosting	Classification	min_child_weight = 3, max_depth = 10, gamma = 0.1, lambda = 0, alpha = 1, subsample = 1, colsample_bytree = 1, scale_pos_weight = 3.0
SVC	Support Vector Classifier	Classification	kernel = 'rbf', C = 50.0, gamma = 1, class_weight = {0:1.0, 1:1.0}
FNN	Feedforward Neural Network	Classification	optimizer = 'adam', loss = 'binary_crossentropy', epochs = 100, activation = 'relu', batch_size = 64, hidden layer sizes = (32,32,32,1), callbacks = EarlyStopping (monitor = 'val_loss', patience = 10), class_weight = {0:1.0, 1:2.0}
RF	Random Forest	Regression	n_estimators = 600, max_features = 0.7, min_samples_split = 0.00005
SVGP	Stochastic Variational Gaussian Process	Regression	optimizer = 'adam',epochs = 2000, kernel = 'rbf', batch_size = 1500, inducing_points = 2000
FNN	Feedforward Neural Network	Regression	optimizer = 'adam', loss = 'mean_squared_error', epochs = 100, activation = 'tanh', batch_size = 128, hidden layer sizes = (64,64,64,1), callbacks = EarlyStopping (monitor = 'val_loss', patience = 10)
RBFNN	Radial Basis Function Neural Network	Regression	optimizer = 'adam', epochs = 100, batch_size = 256, rbf_units = 100, betas = 1.0, callbacks = EarlyStopping(monitor = 'val_loss', patience = 10)

4.2. Surrogate model evaluation

To train and validate the classifier and regressor models of the surrogate, we run the simulations in Python 3.9.20 on a server equipped with a Intel Xeon Gold 6238R 2.20 GHz, an Nvidia Quadro RTX 6000 24 GB, and 256 GB of RAM. Table 16 reports the hyperparameters and settings of the classification and regression ML algorithms employed in this study.

4.2.1. Classifier algorithms results

We chose the hyperparameter configuration that demonstrated the best performance for each of the classifier models evaluated (see Table 16), as introduced in Section 3.3. We compared the performance of these models on the *classification test set*.

Table 17
Surrogate classifier models results based on PR AUC, ROC AUC, F1-score (%), accuracy (%), precision (%) and recall (%).

Model	Training Time [s]	Inference Time [s]	Metrics					
			PR AUC	ROC AUC	F1-score	accuracy	precision	recall
LR	0.589	0.0008	0.4800	0.8021	65.23	87.05	61.29	69.72
RF	178.004	0.0137	0.7712	0.8878	85.38	95.30	93.21	78.77
SVC	885.805	0.0013	0.8729	0.9549	92.72	97.47	92.99	92.46
XGBoost	6.368	0.0207	0.8578	0.9583	92.01	97.16	90.31	93.79
FNN	223.057	0.0524	**0.9956**	**0.9991**	**96.22**	**98.65**	**94.30**	**98.21**

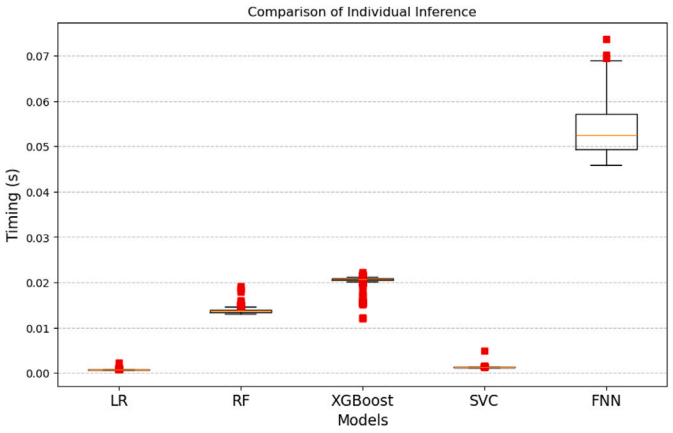


Fig. 13. Comparison of classifier algorithms inference time.

Table 17 presents the classification performance of the models in terms of *PR AUC*, *ROC AUC*, *F1-score*, *accuracy*, *recall*, and *precision*. Moreover, we compared the time it takes for each model to determine whether a particular set of PEWEC design parameters leads to a feasible design or not (*i.e.*, inference time). Fig. 13 illustrates the box plot of one thousand inferences of each model, and Table 17 reports the median inference time of each model. As observed in Table 17, the FNN presented the best classifier performance among the evaluated models, with a *PR AUC* of 0.9956, *ROC AUC* of 0.9991 and an *F1-score* of 96.22%. From an operational perspective, the FNN precision of 94.30% indicates that 5.70% of the configurations classified as feasible in the test set are false positives. For these cases, the regressor may return numerical power values, but such predictions are not physically valuable. Conversely, the recall of 98.21% indicates that 1.79% of the actually feasible configurations are incorrectly rejected and are therefore not evaluated by the regressor. The coupled architecture should consequently be regarded as a preliminary screening tool, while shortlisted configurations should be verified using higher-fidelity models before subsequent detailed design stages. Nonetheless, as can be seen in Fig. 13, it had the largest inference time, with a median of 0.0524 s. The SVC classifier also performed well, but it was outperformed by the FNN in terms of all metrics, with a *PR AUC* of 0.8729, *ROC AUC* of 0.9549, *F1-score* of 92.72%, 97.47% *accuracy*, 92.99% *precision*, and 92.46% *recall*. The training time of the FNN (223.057 s) was lower than that of the SVC (885.805 s), but the median inference time of the SVC was 0.0013 s against 0.0524 s of the FNN. The XGBoost classifier demonstrated comparable performance to the SVC classifier, even slightly outperforming it in terms of *ROC AUC* (0.9583) and *recall* (93.79%). Nonetheless, the median inference time of the SVC (0.0013 s) was lower than that of the XGBoost (0.0207 s) despite the fact that the training time of the XGBoost (6.368 s) was significantly less than that of the SVC (885.805 s). Furthermore, the XGBoost classifier was outperformed by the SVC in terms of *PR AUC* (0.8578), *F1-score* (92.01%), *accuracy* (97.16%) and *precision* (90.31%). The LR classifier presented the worst performance among the models, with a *PR AUC* of 0.4800, *ROC AUC* of 0.8021, an *F1-score* of 65.23%, 87.05% *accuracy*, 61.29% *precision*, and 69.72% *recall*. Nonetheless, it was the model with the shortest training time (0.589 s) and fastest inference (0.0008 s). Interestingly, the RF classifier, with a training time of 178.004 s and inference time of 0.0137 s, performed better than the LR in terms of all metrics and slightly outperformed the XGBoost and SVC classifiers in *precision* (93.21%). Nonetheless, the RF demonstrated lower performance than the other models (*i.e.*, FNN, XGBoost, and SVC).

It is important to highlight that the training time is a cost only incurred once and can be done offline, in the sense that the model can be trained before it is in the "production" pipeline. Instead, the inference time is the time taken by the trained model for each evaluation. All things considered, for the classifier, we chose the FNN model due to its better trade-off between classification performance and lightweight computational cost. Despite having a training time of around 4 min, the FNN classifier had an inference time of 0.0524 s and significantly outperformed all the other models in terms of classification performance, as already described. In case reducing the inference time, the SVC could be a good alternative, which, despite showing slightly worse performance than the FNN classifier and having a larger training time (885.805 s compared to the 223.057 s of the FNN), had an inference time of 0.0013 s (one order of magnitude lower than the FNN).

Table 18
Surrogate regression model results based on NRMSE (%), R^2, and MAE (kW).

Model	Training Time [s]	Inference Time [s]	Metrics		
			NRMSE	R^2	MAE
RF	293.58	0.2753	1.719	0.895	1.225
SVGP	17,972.75	0.2340	2.912	0.698	2.319
FNN	5790.26	0.0595	**0.914**	**0.970**	**0.635**
RBFNN	2860.67	0.0589	2.127	0.839	1.515

4.2.2. Regression algorithms results

For each regression model, we employed the hyperparameter configuration that demonstrated the best performance (see Table 16), as already introduced in Section 3.4. We compared the models' performance based on their predictive performance on the *regression test set*.

Table 18 shows the prediction performance of the four regression models evaluated. Furthermore, we compared the time it takes for each model to determine the power matrix for the 141 N_c sea states for a particular PEWEC design (*i.e.*, inference time). Fig. 15 reports the box plot of one thousand inferences for each model. As observed in Table 18, the FNN demonstrated the best prediction performance among the models, with an error of 0.914% in terms of NRMSE, 0.970 of R^2 and MAE of 0.635 kW. Despite having a large training time (5790.26 s, almost 1.5 hours), it exhibited one of the fastest inference times among the models (as can be seen in Fig. 15), with a median of 0.0595 s. It is important to highlight that the training time is a cost only incurred once and can be done offline, in the sense that the model can be trained before it is in the "production" pipeline. Instead, the inference time is the time taken for the trained model by each evaluation. Interestingly, the RF also showed good prediction performance, with an NRMSE of 1.719%, an R^2 of 0.895 and MAE of 1.225 kW. Additionally, the training time of the

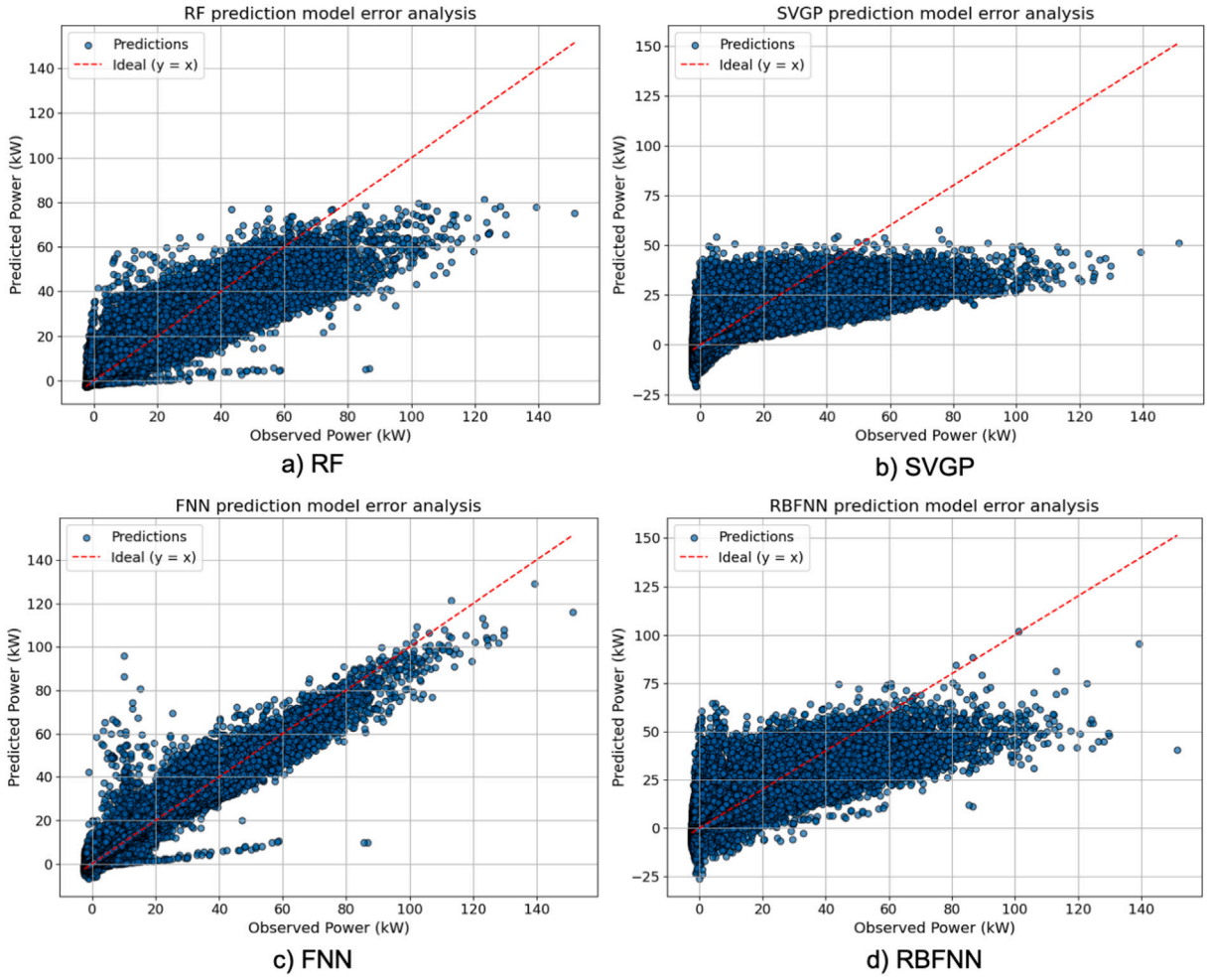


Fig. 14. Comparison of regression models, power predictions, and observed values of the test set.

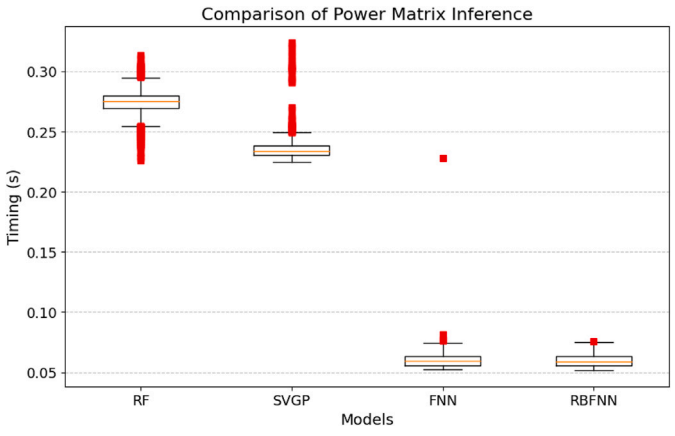


Fig. 15. Comparison of regression models inference times for the power matrix of the 141 N_c sea states.

RF (293.58 s) was significantly lower than that of the FNN (5790.26 s). Nonetheless, the RF showed the largest inference time (as illustrated in Fig. 15) with a median of 0.2753 s. On the other hand, the SVGP presented the worst performance, with an NRMSE of 2.912%, R^2 of 0.698 and MAE of 2.319 kW. Moreover, it showed one of the longest inference times (only surpassed by the RF) with a median inference time of 0.234 s. Furthermore, the RBFNN performed slightly better than the SVGP, with an NRMSE of 2.127%, R^2 of 0.839 and MAE of 1.515 kW, and both its training time (2860.67 s) and inference time (0.0589 s) were significantly lower than those of the SVGP, which presented a training time of 17,972.75 s and an inference time of 0.2753 s. Nonetheless, the RBFNN showed lower performance than the other models. The better fit of the FNN compared to the other models can be clearly observed in Fig. 14, which shows the observed power and the predicted power for the test set for the four models. As illustrated, all four models showed a good fit for low values up to 30 kW of the observed power. Nonetheless, the SVGP did not follow the ideal fit (y = x depicted by the red dashed line) for observed values higher than 40 kW, as shown in Fig. 14(b). Similar behaviors were observed for the RF and RBFNN above 60 kW (as shown in Fig. 14a and d). Instead, the FNN (see Fig. 14c) showed a good fit overall, confirmed by the R^2 of 0.970 and MAE of 0.635 kW (considering that the PEWEC observed power values ranged up to 151 kW as shown in Fig. 14c).

Based on the benchmark indicated in [124], the NRMSE of all the models are lower than 10%, which proves that all of the surrogates presented had good predictive capabilities. Moreover, the NRMSE error for RF and FNN is lower than 2%, demonstrating that these models showed very good predictive capabilities.

For the regressor, we chose the FNN model as the best model due to its better trade-off between prediction performance and lightweight computational cost. For example, the RF model training time (293.58 s) was one order of magnitude lower than that of the FNN (5790.26 s). On

the contrary, the FNN model inference time is one order of magnitude lower than that shown by the RF, which also shows good predictive performance (but worse than that of the FNN). Despite the fact that the training time of the FNN is higher than that of the RF model, the training is only performed once. Instead, although the inference times of both the FNN and RF are low at first glance (0.0595 s and 0.2753 s median, respectively), when performing hundreds or thousands of evaluations (which typically occurs during techno-economic assessments), this order of magnitude difference in the inference time between the FNN and RF models counts. On the other hand, the RBFNN reports faster computational times than the RF model but worse prediction accuracy.

Compared to the time that the described FD numerical model of the PEWEC takes to evaluate the performance for a given WEC (*i.e.*, for a given d), which is in the order of 10^2s on average (it depends on the size of the hull to be meshed during the hydrodynamic coefficients evaluation), the proposed surrogate (FNN classifier combined with FNN regressor) significantly reduces the computational load. The resulting inference time when combining these two models is in the order of 10^{-1}s (3 orders of magnitude lower). Even when considering the combination of other models, for example, the SVC classifier (inference time of 0.0013 s) and the RF regressor (0.2753 s), the inference time is still in the order of 10^{-1}s (3 orders of magnitude lower than the numerical model).

Furthermore, if we consider a larger design space in terms of "denser sampling" of the existing thirteen parameter space, this will raise the number of training designs, and hence the training cost of the surrogate model (both classifier and regressors), but the inference cost remains unchanged. These scalability scenarios reinforce the motivation for the surrogate, where the FD solver must be invoked for a rapidly growing number of geometries. The surrogate is most valuable because the training cost is incurred once and amortized, while each subsequent design is evaluated in approximately 0.11 s (considering both classifier and regressors). On the other hand, when considering the scenario of higher-resolution sea-state grids, the classifier algorithms are not affected (since the inputs are the design parameters) while the regressors algorithm training time will be affected. The training time will increase due to having a larger dataset. Nonetheless, the inference time may only increase marginally. Despite the fact that the regressor predicts the power matrix row-by-row over the H_s and T_e grid and therefore depends on the number of rows, the current reported time is mainly dominated by internal Python function call overhead rather than the mathematical matrix computation. On the contrary, in the FD numerical model, the computational cost of grid refinement is not negligible, since the PTO control parameters are optimized for each considered sea state. Consequently, the order of magnitude advantage of the surrogate is preserved, since the training cost is an offline, one-time cost and does not affect deployment-time inference.

In conclusion, the results highlight the surrogate models' ability to accurately determine WEC performance while substantially reducing the computational cost. Furthermore, this computational advantage compared to the FD numerical model is robust to both grid refinement and design-space growth, with the inference time being almost unaffected, while the training cost, which grows is a one-time "investment" that the surrogate is designed to amortize. Consequently, the proposed surrogate framework paves the way for performing multi-site WEC design (typically limited by computational constraints). By enabling the efficient exploration of large design spaces across multiple locations, the surrogate model emerges as a key tool in narrowing the LCoE gap between WECs and other mature renewable technologies.

4.2.3. Feature importance and interpretability analysis

To quantify the sensitivity of the proposed FNN surrogate model $\tilde{\mathcal{M}}(d)$ (which was the one showing the best performance as discussed in Section 4.2.1 and Section 4.2.2) to the design parameters, we complemented the performance evaluation with a feature-importance analysis of the two FNN components (the *classifier* and the *regressor*) to understand the influence of the various design parameters on the surrogate behavior. To this end, we employed two techniques: permutation importance [106] and SHapley Additive exPlanations (SHAP) values [125]. When applying permutation importance, each input feature is randomly shuffled across the held-out test set (*i.e.*, the *classification test set* and the *regression test set* for the classifier and regressor, respectively) and the resulting degradation of the model's reported metric is measured, *i.e.*, the *F1-score* for the classifier and the *NRMSE* for the regressor. A larger degradation indicates a more influential feature. For the case of the FNN regressor, since the same design (*i.e.*, thirteen design parameters) was unchanged across the 141 sea states combinations (*i.e.*, combination of H_s and T_e) (as explained in Section "Regression Data Split and Pre-processing"), the thirteen design parameters were permuted at the design level (one consistent value per design, grouped by *designID*) to avoid evaluating the regressor on physically inconsistent inputs, whereas H_s and T_e were permuted per row. Furthermore, each estimate was averaged over thirty repetitions.

On the other hand, we computed the SHAP values, which compute one attribution per row (*i.e.*, input features) relative to a training-set background distribution. For the case of the regressor, the per-row attributions were averaged at the design level (so that each design counts once), and the SHAP values were rescaled to physical power units (kW).

Fig. 16 reports the importance of the design parameters for the *classifier*, which takes as inputs the thirteen design parameters. As can be observed, the pendulum arm length l and the height-to-draft ratio k are the most influential features (permutation Δ*F1-score* of 0.42 and 0.33 and SHAP values of 0.208 and 0.142, respectively), followed by the hull length L and width W. This ranking is physically meaningful, as these parameters directly constrain the geometric and inertial feasibility of the device. Furthermore, the permutation and SHAP rankings agree on the leading parameters.

Fig. 17 illustrates the importance of the inputs to the FNN *regressor*, separating the thirteen design parameters (Fig. 17a) from the two sea-state descriptors, *i.e.*, H_S and T_e (Fig. 17b). As can be seen in Fig. 17, the sea-state variables H_s and T_e dominate the prediction by a wide margin (permutation ΔNRMSE of 5.5% and 5.1%, and SHAP values of 5.0 kW and 4.2 kW, respectively), which is expected since the absorbed power of the device is strictly governed by the incident wave resource. Among the

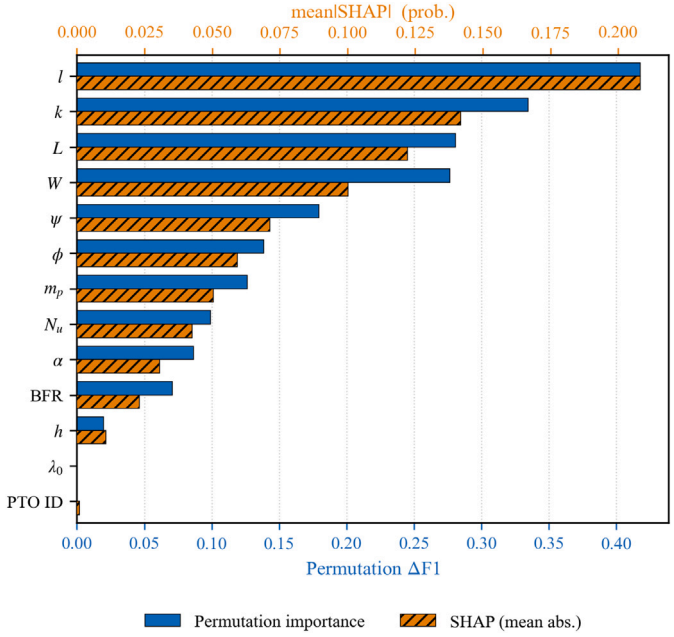


Fig. 16. Feature importance of the FNN classifier, obtained by permutation importance bottom axis) and by SHAP (top axis), for the thirteen design parameters.

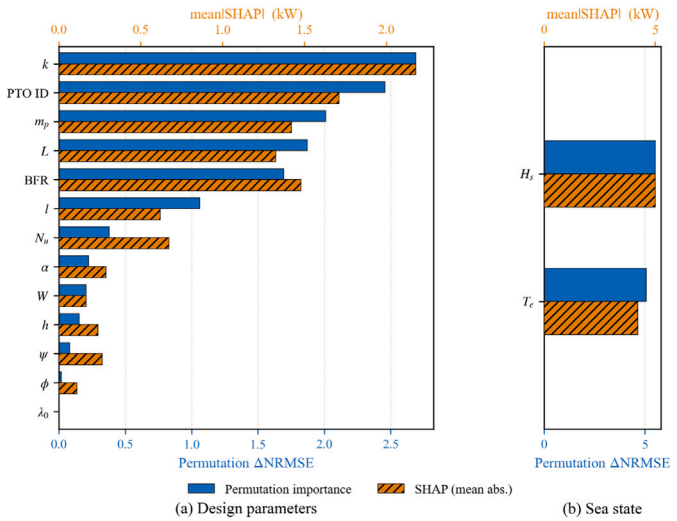


Fig. 17. Feature importance of the FNN regressor, obtained by permutation importance bottom axis) and by SHAP (top axis), for (a) the thirteen design parameters and (b) H_s and T_e.

design parameters, the height-to-draft ratio k, the PTO ID, the pendulum mass m_p, the hull length L and the ballast filling ratio (BFR) emerge as the leading drivers, while the fulcrum position λ_0 has no influence, consistent with the fact that it is held constant across the design space (see Table 3). Notably, the permutation and SHAP rankings are in close agreement, specifically for the leading drivers such as H_s, T_e, k and PTO ID. It is worth noting that the most influential parameters differ between the two tasks: the FNN classifier is governed mainly by the pendulum and hull geometry (l, k, L, W), whereas power production is additionally sensitive to the PTO ID and the pendulum mass m_p, reflecting the distinct physical mechanisms that each surrogate component approximates.

4.3. Test cases

To evaluate the accuracy of the proposed FNN surrogate model $\tilde{\mathcal{M}}(d)$ (which was the one showing the best performance as discussed in Sections 4.2.1 and 4.2.2) for WEC performance prediction, four representative deployment sites were selected, each characterized by distinct meteo-oceanographic conditions: the European Marine Energy Center (EMEC, in the North Sea), the Biscay Marine Energy Platform (BIMEP, in the Bay of Biscay), a site located offshore the coast of Sicily, designated for the MedWind project [126] (in the present paper we refer to it as CDS, in the Sicily's channel of the Mediterranean Sea) and Capo Caccia (west coast of Sardinia). The data for the locations have been obtained from ERA5 [127] and Copernicus Marine Service [128]. The geographical locations of the sites are shown in Fig. 18, while their wave energy resource in terms of mean occurrences and energy density per meter of wave front are reported in Figs. 19–22. Each plot presents a bivariate distribution of sea states in terms of H_s and T_e. The colormap of the left plot shows the annual percentage occurrence of each sea state bin, while the right plot reports the corresponding annual wave energy density (in MWh/m), allowing for a combined understanding of both frequency and energetic contribution of wave conditions at the site. Moreover, the sites' most frequent and most energetic H_s and T_e are reported in Table 19 together with the site total power density.

We ran the simulations of the expensive numerical model $\mathcal{M}(d)$ in MATLAB on a PC equipped with an Intel i7-12700 H CPU at 2.30 GHz and 32 GB of RAM. To integrate the FNN surrogate model $\tilde{\mathcal{M}}$, which was developed using *Keras* with a TensorFlow backend in Python (as explained in Sections 3.3.5 and 3.4.3), we first converted the FNN classifier and FNN regressor to Open Neural Network Exchange (ONNX) format using the *tf2onnx* Python tool [129] and later imported them into MATLAB using the *importNetworkFromONNX* built-in function from the *Deep Learning Toolbox*. The simulation results are summarized in Table 20 and in Figs. 23 to 26. For each site, the AEP estimated by the surrogate model was compared against that obtained from the expensive numerical model $\mathcal{M}(d)$. The surrogate model demonstrated consistently good performance across all four locations, with AEP relative errors (Δ_{AEP_r}) of 7.1%, 5.7%, 1.7% and 3.2% for BIMEP, EMEC, CDS and Capo Caccia, respectively. The highest deviation was observed for BIMEP.

Although the oceanic test sites showed a higher relative error than the Mediterranean case study, the discrepancy in terms of AEP (Δ_{AEP}) between the surrogate and the numerical model remained below 2.1 MWh (as reported in Table 20). This small gap both justifies using the proposed meta-model for preliminary multi-site analyses and encourages further model development.

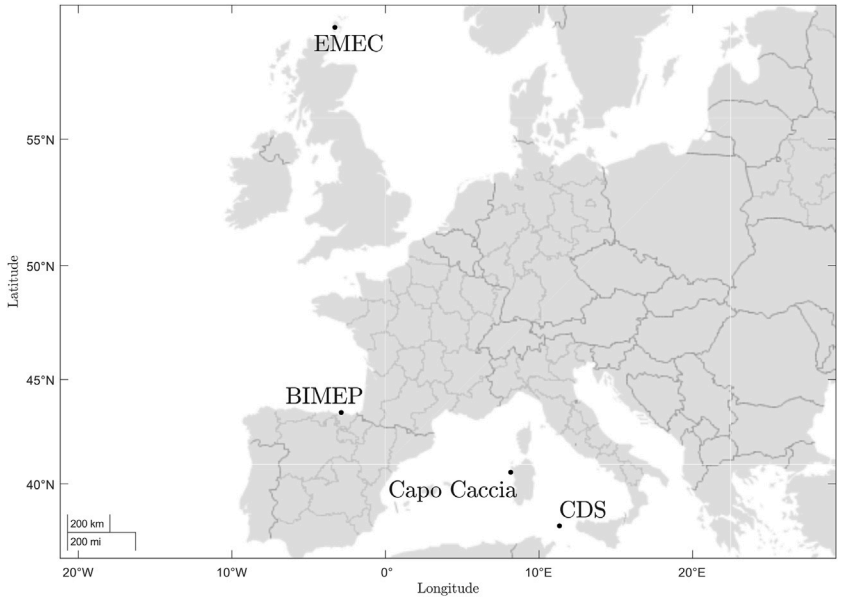


Fig. 18. North sea (BIMEP, EMEC) and Mediterranean (CDS and Capo Caccia) sites analyzed.

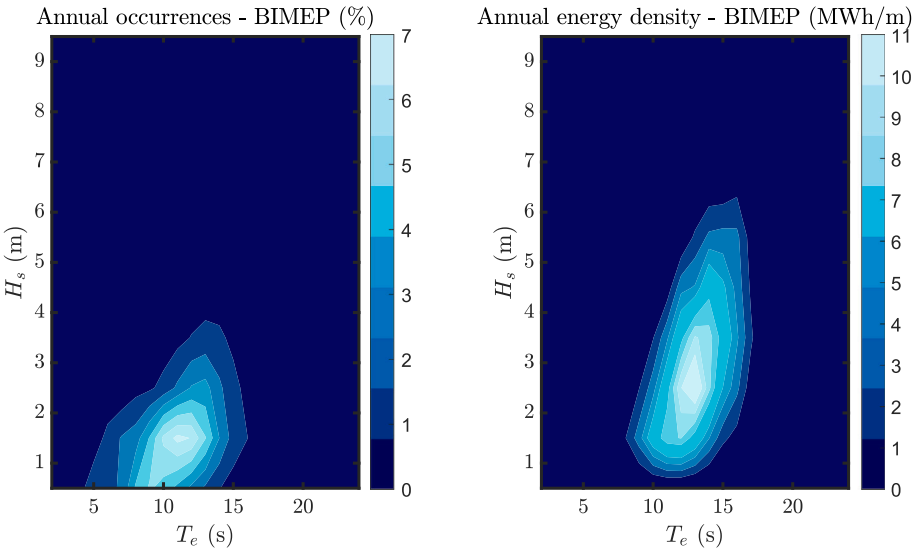


Fig. 19. Wave energy occurrences and energy density matrices for the BIMEP test site.

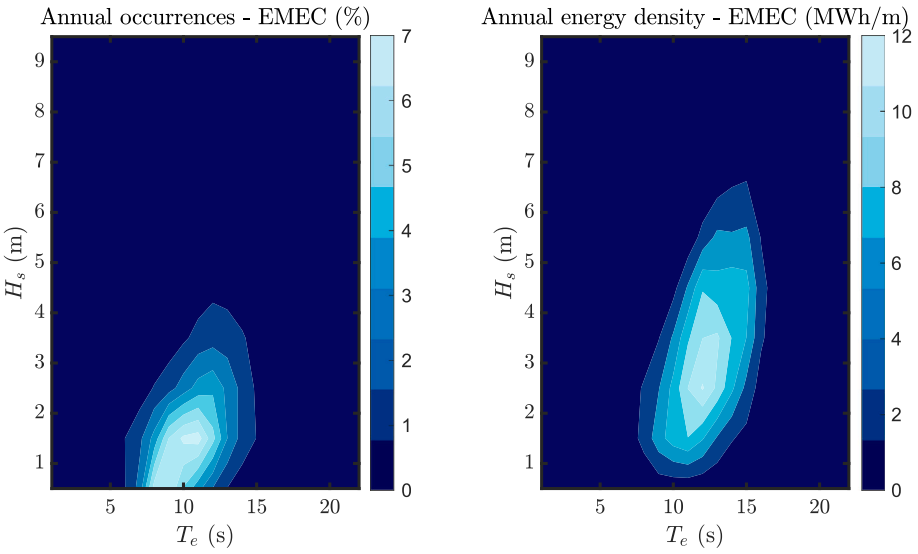


Fig. 20. Wave energy occurrences and energy density matrices for the EMEC test site.

These results confirm the capability of the surrogate model to reliably approximate site-specific WEC performance while significantly reducing computational requirements. As such, the model is particularly well-suited for techno-economic assessment workflows, where repeated evaluations across multiple locations and device configurations are needed. In particular, its integration within a design-dependent exploration framework (*i.e.*, where site-specific productivity is evaluated as a function of the WEC design vector d) enables robust multi-site analyses and informed design choices that reflect both environmental variability and system-level constraints.

Referring to the nominal numerical model of the PEWEC ($\mathcal{M}(d)$) (introduced and described in Section 3.1), the benchmark for the required computational time is 3.7 min with respect to the design employed in the test cases. In this sense, the meta-model significantly outperforms the timing benchmark, cutting the elapsed time from 3.7 min (*i.e.*, 222 s) to almost 0.11 s. The meta-model timing includes pre-processing and post-processing, *i.e.*, scaling, reshaping and inverse scaling. Moreover, the computational time required for the numerical model evaluation is strictly dependent on the parameter values of the WEC design candidate, *i.e.*, device dimensions (different WEC sizes would imply different meshes for the hydrodynamic evaluation), while the meta-model is uncorrelated with such parameters. In the case of considering larger design spaces, in the sense of denser sampling of the existing thirteen WEC parameter space, as already discussed in Section 4.2.2, the surrogate model inference cost remains unchanged, while for the numerical model the cost will depend almost linearly on the number of designs evaluated, which reinforces the motivation for the surrogate. However, it is fair to highlight that the surrogate model also required offline computational time, such as for dataset generation (2621.25 min in our case for the 175000 samples) and training of the model (223.057 s and 5790.26 s for the FNN classifier and regressor, respectively). In the case of employing a larger design space, both the dataset generation time and the training cost of the model will increase. Nonetheless, considering that both of these processes (dataset generation and training of the models) are done once, once the surrogate is available, the more design evaluations performed employing it, the more convenient its implementation becomes. In particular, given that evaluating WEC performance for a single site requires thousands of design evaluations, the development

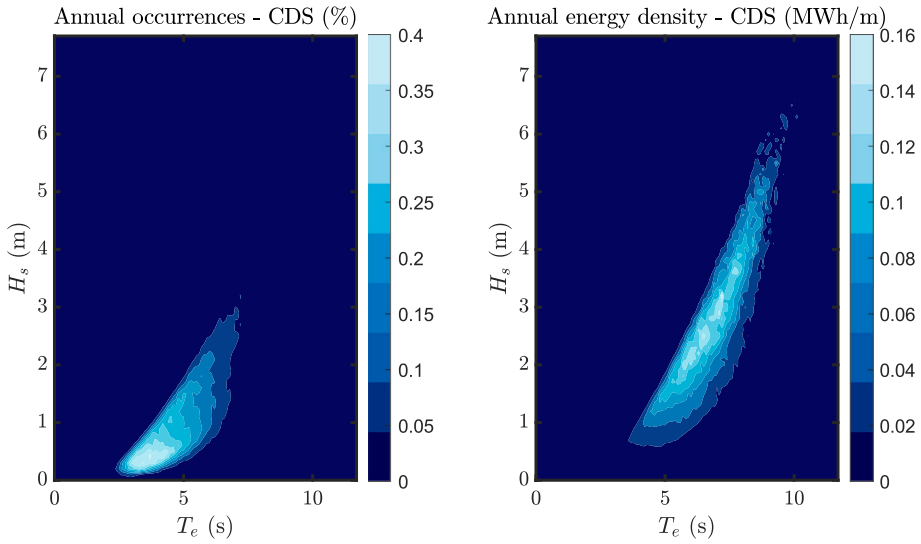


Fig. 21. Wave energy occurrences and energy density matrices for the CDS test site.

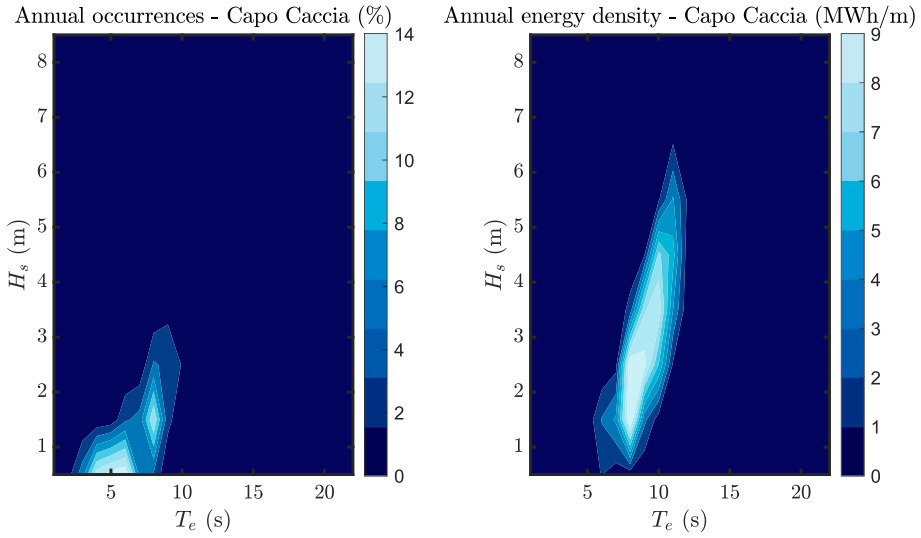


Fig. 22. Wave energy occurrences and energy density matrices for the Capo Caccia test site.

Table 19
Overview of the main meteomarine data for the four examined test sites.

Parameter	Unit	BIMEP	EMEC	CDS	Capo Caccia
Most occurrent H_s	(m)	1.5	1.5	0.5	0.5
Most occurrent T_e	(s)	12	11	4.6	8
Most energetic H_s	(m)	2.5	2.5	2.5	2.5
Most energetic T_e	(s)	13	12	7.1	10
Wave power density J_{wave}	(kW/m)	23.0	25.2	7.4	9.54

Table 20
AEP comparison between meta-model and numerical model for a selected PEWEC design.

Parameter	Unit	BIMEP	EMEC	CDS	Capo Caccia
AEP from $\mathcal{M}(d)$	(MWh)	25.8	37.7	76.0	48.1
AEP from $\tilde{\mathcal{M}}(d)$	(MWh)	23.9	35.6	74.8	46.6
Δ_{AEP}	(MWh)	1.8	2.1	1.2	1.5
Δ_{AEP_r}	(%)	7.1	5.7	1.7	3.2

of a surrogate model is highly beneficial, especially considering that the ideal scenario involves analyzing multiple locations rather than just one.

The trained surrogate provides a compact cross-design mapping from the design-parameter vector to the corresponding site-independent power matrix. Once trained, a new candidate design can therefore be evaluated directly, without re-running the full hydrodynamic analysis, dynamic-response computation, and PTO-control optimization. This capability should be interpreted only within the prescribed PEWEC design domain used for dataset generation.

4.4. Sensitivity analysis

To further assess the generalization capability of the trained surrogate, an additional sensitivity analysis was performed on sea states not belonging to the original representative power-matrix grid. Two complementary cases were considered. First, a set of internal sea states was generated inside the convex hull of the original sea-state domain by shifting the original grid points, thus testing the interpolation capability of the surrogate. Second, a set of external sea states was generated by moving selected boundary points outward from the center of the covered

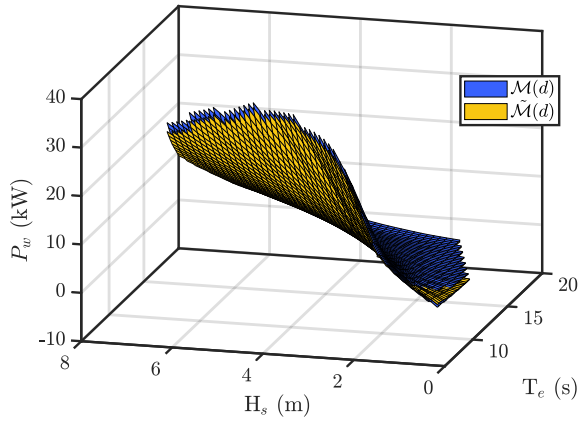


Fig. 23. $\mathcal{P}(H_s, T_e, d)$ comparison between the numerical model $\mathcal{M}(d)$ and the meta-model $\tilde{\mathcal{M}}(d)$ for a selected PEWEC design, in the BIMEP case.

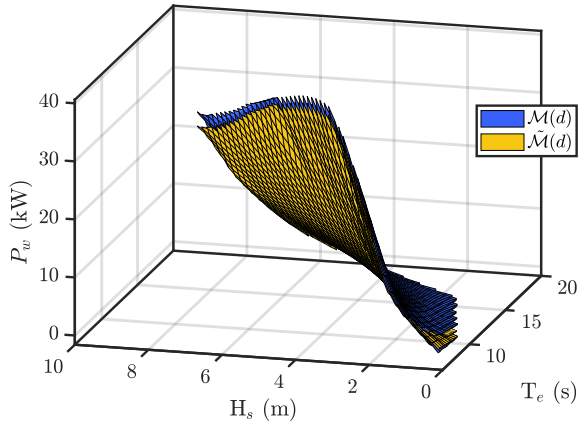


Fig. 24. $\mathcal{P}(H_s, T_e, d)$ comparison between the numerical model $\mathcal{M}(d)$ and the meta-model $\tilde{\mathcal{M}}(d)$ for a selected PEWEC design, in the EMEC case.

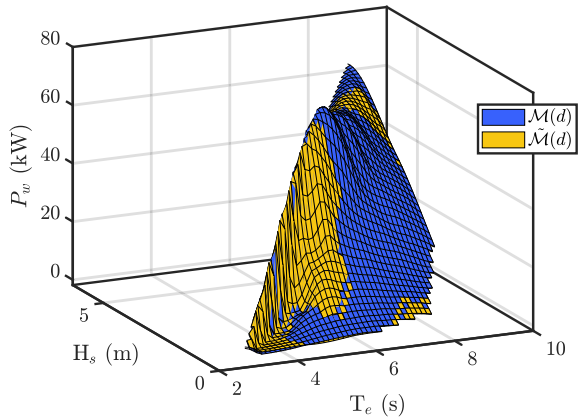


Fig. 25. $\mathcal{P}(H_s, T_e, d)$ comparison between the numerical model $\mathcal{M}(d)$ and the meta-model $\tilde{\mathcal{M}}(d)$ for a selected PEWEC design, in the CDS case.

domain, thus providing a preliminary indication of the surrogate behavior under linear extrapolation. The two sets of sampled sea states are graphically illustrated in Fig. 27.

For each additional sea state, the power predicted by the surrogate model, $P_w^{\tilde{\mathcal{M}}}$, was compared against the corresponding value obtained from the reference PEWEC numerical model, $P_w^{\mathcal{M}}$.

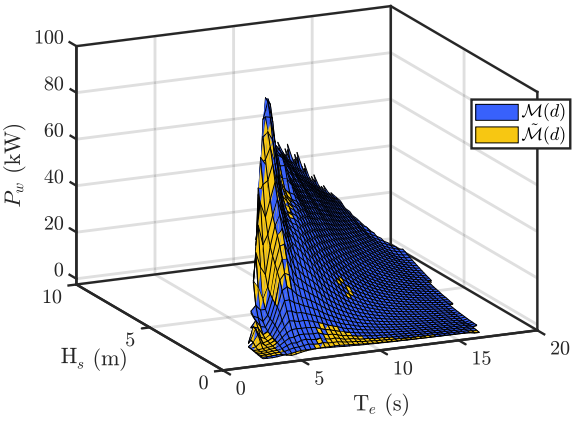


Fig. 26. the $\mathcal{P}(H_s, T_e, d)$ comparison between the numerical model $\mathcal{M}(d)$ and the meta-model $\tilde{\mathcal{M}}(d)$ for a selected PEWEC design, Capo Caccia case.

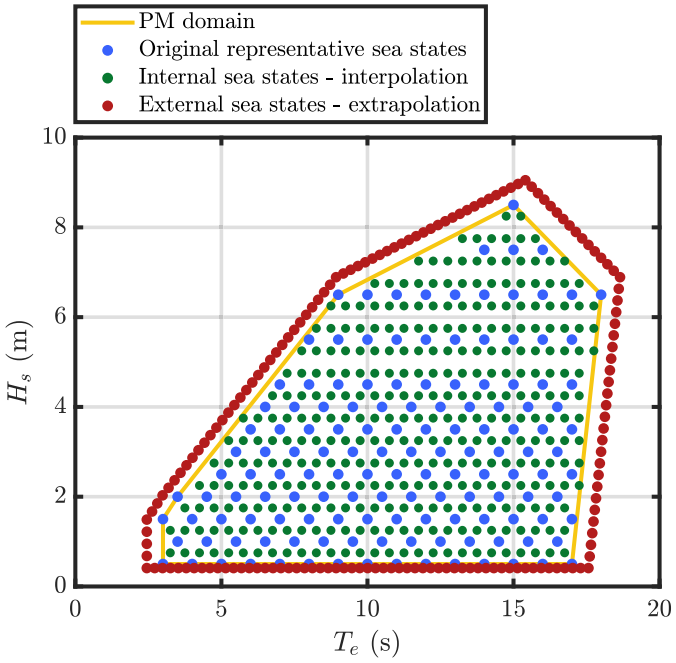


Fig. 27. Sets of sea states employed for the sensitivity analysis.

The results are shown in Fig. 28 and the statistical indices are summarized in Table 21. For the internal interpolation set, the surrogate presents a bias of 0.78 kW, an MAE of 0.91 kW, and an RMSE of 1.37 kW. For the external linear-extrapolation set, the corresponding values are −0.69 kW, 0.69 kW, and 0.98 kW, respectively. Although the external case presents lower MAE and RMSE in absolute power units, the two datasets sample different regions of the power-response surface and are associated with different characteristic magnitudes of the reference power. Therefore, the absolute metrics alone cannot be used to conclude that extrapolation performs better than interpolation. The normalized indices clarify this result. For the internal set, the Mean Bias Error (MBE), $\text{MAE}_{\%}$, and $\text{RMSE}_{\%}$ are 1.99%, 9.22%, and 14.86%, respectively, whereas the external set presents values of −13.23%, 13.23%, and 20.48%. Thus, the external case exhibits poorer relative accuracy. This behavior is also visible in Fig. 29 (where $\varepsilon_{\mathcal{P}_w}$ is the relative error between $P_w^{\mathcal{M}}$ and $P_w^{\tilde{\mathcal{M}}}$): the Cumulative Distribution Function (CDF) shows that the interpolation errors are more concentrated at low values, while the extrapolation distribution presents a more pronounced upper tail.

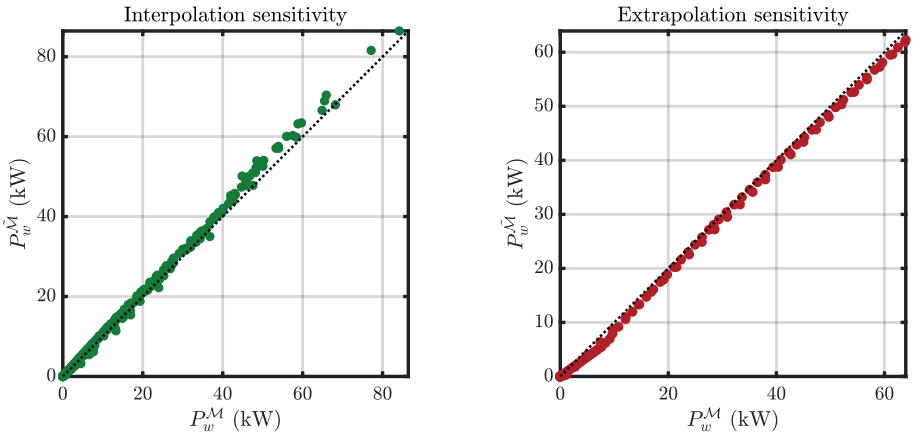


Fig. 28. Sensitivity analysis outcomes, comparing power production of the surrogate model and the reference numerical model for the two investigated sets of sea states.

Table 21
Comparative sensitivity analysis of the proposed FNN surrogate model and the SVGP baseline on unseen sea states.

	Bias (kW)	MAE (kW)	RMSE (kW)	MBE (%)	$MAE_{\%}$ (%)	$RMSE_{\%}$ (%)
FNN surrogate internal interpolation	0.78	0.91	1.37	1.99	9.22	14.86
FNN surrogate external extrapolation	−0.69	0.69	0.98	−13.23	13.23	20.48
SVGP internal interpolation	−6.08	6.24	10.67	23.15	75.65	535.19
SVGP external extrapolation	−4.61	5.75	9.26	48.31	108.31	483.97

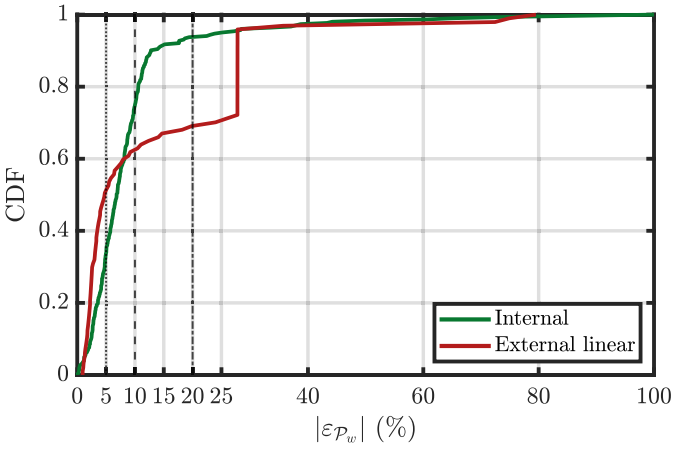


Fig. 29. CDFs of the absolute relative power error for the internal-interpolation and external linear-extrapolation sensitivity cases. The vertical lines indicate absolute relative error thresholds of 5%, 10%, and 20%.

Nevertheless, the errors remain limited in terms of absolute power production, with an MAE below 1 kW and an RMSE below 1.4 kW for both sets.

Furthermore, we aimed to benchmark the proposed surrogate with more classical interpolation algorithms like GPR to establish whether the proposed architecture offers an advantage over these classical methods. However, as highlighted in Section 3.4.2, we employed SVGP for the regression task as a variant of GPR to address the limitations imposed by the large scale of our dataset. While the SVGP showed acceptable results in the *regression test set*, it was outperformed by the FNN, as reported in Table 18 and discussed in Section 4.2.2. Therefore, we considered the SVGP model the benchmark for our case and evaluated this model on the additional set of internal and external sea states not belonging to the original representative power-matrix grid.

Table 21 compares the performance of these models on these additional sea states. As observed in Table 21, the proposed FNN significantly outperforms the baseline SVGP across all metrics. Although the SVGP showed acceptable results across the "trained" sea states (as reported in Table 18), its MAE increased drastically from 2.319 kW in the *regression test set* to 6.24 kW for internal interpolation, and 5.75 kW for external extrapolation. Furthermore, Fig. 28 illustrates the results for the FNN while Fig. 30 depicts the SVGP case. As observed in Fig. 28, the FNN model exhibits a strong fit overall and closely tracks the "ideal" diagonal in both the interpolation and extrapolation scenarios. This strong fit of the FNN is in line with the model's behavior on the *regression test set*, as can be observed in Fig. 14(c). On the contrary, the SVGP model does not follow the ideal fit (y = x), severely underpredicting the target values in both the interpolation and extrapolation scenarios for observed powers greater than 40 kW. This underpredictive behavior of the SVGP was already observed when analyzing its performance in the *regression test set* (Fig. 14b). These quantitative and visual results, alongside the significantly lower training time and inference time of the FNN with respect to the baseline SVGP (as reported in Table 18) demonstrate the superior performance of the developed FNN based surrogate model. Furthermore, this confirms that the proposed architecture successfully outperforms traditional interpolation algorithms like GPR (adapted here as SVGP due to dataset size constraints).

Overall, these results validate the use of the proposed surrogate and its advantage over classical methods. Additionally, the findings support the use of the surrogate within the original sea-state envelope, whereas predictions outside this domain should still be interpreted with caution. Indeed, this limitation is inherent to the adopted modeling and control framework. In the reference numerical workflow, the PTO control parameters are optimized over a predefined set of representative sea states, selected to cover the most relevant (occurring) operating conditions for the analyzed deployment scenarios. Consequently, the surrogate is trained to reproduce the model response within this domain, rather than to provide reliable predictions under arbitrary or extreme sea states. Under these assumptions, the proposed model is therefore suitable for preliminary techno-economic performance assessment and rapid design screening, while detailed analyses under extreme or out-of-domain sea conditions should still rely on dedicated high-fidelity simulations. In this sense, an advantage of the proposed surrogate model lies in its capability to skip both the hydrodynamic analysis and the control parameters inner optimization loop when more than one design of the PEWEC is tested. In

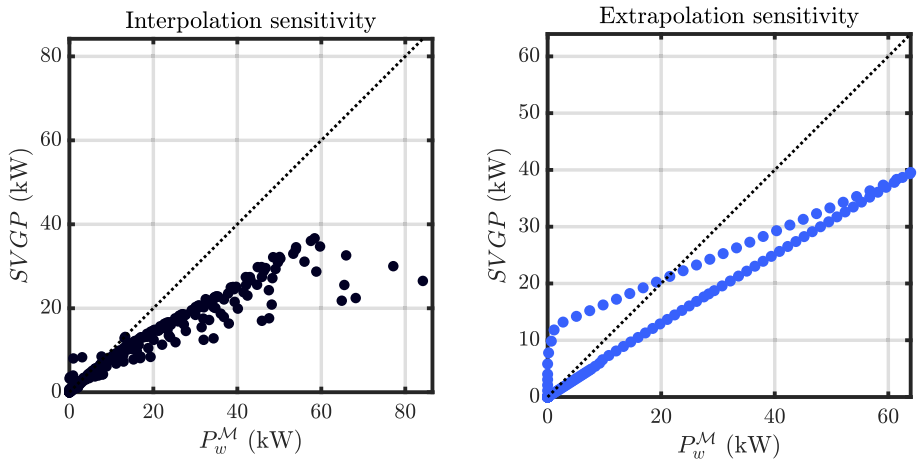


Fig. 30. Sensitivity analysis of the SVGP baseline, comparing the power production of the SVGP model and the reference numerical model for the two investigated sets of sea states.

practice, assessing the techno-economic performance of multiple PEWEC designs would require performing hydrodynamic analyses and optimizing the control parameters for each design candidate configuration. With the proposed surrogate model, this computationally demanding workflow is bypassed: the design vector can be directly provided as input to obtain the corresponding device power matrix with a satisfactory degree of accuracy.

5. Conclusions, limitations and further work

In this work, we presented a methodology for the development of a ML-based surrogate model that accelerates performance evaluation of a WEC design. The meta-model has been designed to map the input design parameters of the WEC to the performance space (in terms of power production) for a set of sea states. The surrogate model explicitly accounts for the existing interdependencies among the various steps involved in the performance evaluation of a WEC design by adopting a holistic design approach. In particular, the method's effectiveness is demonstrated here on the PEWEC case study, but the methodology is generally transferable to any WEC design. The surrogate model receives as input the WEC design parameters (*i.e.*, thirteen design parameters in the PEWEC case) and outputs the general site-independent power matrix of the device. Since many design parameter combinations throughout the WEC design process may lead to infeasible designs, the surrogate architecture integrates two main components: a classifier and a regressor. The classifier aims to determine whether a given set of design parameters yields a feasible design, while the regressor predicts the site-independent power matrix for the feasible designs. Therefore, for the classification task, we compared the performance of five ML algorithms used for binary classification tasks that have shown excellent results in the literature: LR, RF, SVC, XGBoost, and FNN. Moreover, for the regression case, we compared four ML algorithms that have demonstrated promising results as surrogates for WEC design assessment: RF, GPR, RBFNN, and FNN.

The FNN exhibited the best performance for the classification task among the evaluated classifier algorithms. The SVC and XGBoost algorithms also performed well, but their performance was slightly worse than the FNN. In contrast, the LR presented the worst performance. The RF displayed better performance than the LR, and even slightly outperformed the XGBoost classifier in precision, but its performance was lower compared to the FNN, SVC, and XGBoost.

On the other hand, the FNN also demonstrated the best performance for the regression task, followed by the RF. In contrast, the SVGP displayed the worst performance. The RBFNN outperformed the SVGP, but its performance was worse than that of the FNN and the RF. Given that the NRMSE obtained by all algorithms was lower than 10%, the surrogates demonstrated good predictive capabilities as indicated in [124]. Furthermore, the FNN and RF NRMSE values were lower than 2%, which suggests that these surrogates have outstanding predictive capabilities. Moreover, we compared the time it took for each model to predict the power matrix, and the FNN and RBFNN were the fastest, with medians of 0.0595 s and 0.0589 s, respectively. The SVGP and RF inference times were longer, with medians of 0.234 s and 0.2753 s, respectively.

Additionally, we quantified the sensitivity of the proposed FNN surrogate model $\tilde{\mathcal{M}}(d)$ to the design parameters. Thus, we employed permutation importance and SHAP values. In the case of the classifier, the pendulum arm length l and the height-to-draft ratio k were the most influential parameters, with a permutation Δ*F1-score* of 0.42 and 0.33 and SHAP values of 0.208 and 0.142, respectively. On the other hand, in the case of the regressor, the sea-state variables H_s and T_e were the most dominant features, with a permutation ΔNRMSE of 5.5% and 5.1%, and SHAP values of 5.0 kW and 4.2 kW, respectively. Within the design parameters, the height-to-draft ratio k, the PTO ID, and the pendulum mass m_p emerged as the design parameters with greater influence. It is worth noting that the most influential parameters differed between the two tasks: the FNN classifier was governed mainly by the pendulum and hull geometry (l, k, L, W), whereas power production was mainly driven by the sea-states H_s and T_e followed by the PTO ID and the pendulum mass m_p, reflecting the distinct physical mechanisms that each surrogate component approximates.

We further assessed the obtained FNN regression component of the surrogate model by implementing it for assessing the PEWEC device power performance in four representative deployment sites: the EMEC, the BIMEP, Capo Caccia (west coast of Sardinia), and a site off the coast of Sicily (referred to as CDS). We compared the AEP obtained for each site using the reference PEWEC numerical model with the one obtained using the surrogate model. For all sites, the surrogates exhibited consistently good performance, with AEP relative errors of 1.7%, 3.2%, 5.7% and 7.1% for CDS, Capo Caccia, EMEC, and BIMEP, respectively. In all cases, the highest discrepancy between the surrogate and the reference numerical model remained below 2.1 MWh. However, the numerical model required 3.7 min (*i.e.*, 222 s) while the surrogate model (combining the classifier and the regressor) required only 0.11 s. In the case of considering larger design spaces, in the sense of denser sampling of the existing thirteen WEC parameter space, the surrogate model inference cost remains unchanged, while for the numerical model the cost will depend almost linearly on the number of designs evaluated, which reinforces the motivation for the surrogate. Nevertheless, it is fair to acknowledge the offline computational overhead inherent to

the surrogate modeling approach. Specifically, generating the 175,000 dataset required 2621.25 min, while the training of the FNN classifier and regressor took 223.057 s and 5790.26 s, respectively. In the case of employing a larger design space, both the dataset generation time and the training cost of the model will increase. Nonetheless, since both data generation and model training constitute one-time preliminary processes, the relative computational advantage of the surrogate model scales proportionally with the number of design evaluations. Consequently, the surrogate model proves particularly well-suited for techno-economic assessment workflows, in which repeated evaluations (generally on the order of thousands) of device configurations are needed. Moreover, the developed surrogate model unlocks the possibility of performing multi-site assessment and design of WECs, which are generally constrained due to computational burdens. By enabling the rapid exploration of these extensive design spaces and multiple locations during WEC design, the surrogate model can be a key enabler towards reducing the gap between the levelized cost of energy of WECs and other mature renewable technologies, such as wind and solar.

An additional sensitivity analysis on unseen sea states was also carried out to further assess the generalization capability of the trained surrogate. The model showed good agreement with the reference numerical model for internal interpolation cases, with a bias of 0.78 kW, MAE of 0.91 kW, and RMSE of 1.37 kW. The extrapolation test on external sea states also showed limited errors, with a bias of -0.69 kW, MAE of 0.69 kW, and RMSE of 0.98 kW, although predictions outside the original sea-state envelope should still be interpreted with caution considering the relative errors reported in Table 21. Furthermore, we benchmarked the generalization capability of the trained surrogate against a more classical regression algorithm like GPR. In particular, we compared it with the SVGP model, adopted in this work as a variant of GPR due to dataset size constraints. The results demonstrated the superior performance of the proposed surrogate over this classical regression algorithm in both the interpolation and extrapolation scenarios.

Overall, these results reinforce the suitability of the surrogate for preliminary techno-economic screening of multiple designs within the representative operating domain, while out-of-domain or extreme sea-state analyses require dedicated numerical evaluations. It should also be clarified that a sensitivity analysis concerning noisy input data is not directly applicable to the present surrogate for the same sense as in sensor-driven or forecasting models. The direct inputs of the model are deterministic PEWEC design variables, defined during the design stage, together with prescribed sea-state coordinates used to reconstruct the site-independent power matrix. Therefore, they do not represent online measurements affected by sensor noise or missing channels. Possible uncertainty would instead arise at a different level, *e.g.*, manufacturing tolerances or uncertain metocean statistics. Since the surrogate predicts a site-independent power matrix, such uncertainties would mainly propagate through the subsequent site-specific AEP calculation in the same way as they would be propagated using the numerical model, but taking into account, in addition, those derived from the surrogate itself.

5.1. Limitations and further work

The objective of the designed meta-model is the mapping of the design parameter space to the performance space (in terms of power production) for a set of sea states (*i.e.*, site-independent power matrix). The developed surrogate model considers the interactions between the various steps of the WEC design process. Nonetheless, the surrogate model does not directly approximate the underlying hydrodynamic solvers nor does it reproduce intermediate physical steps of the system, such as motion responses or PTO dynamics. It is also important to underline that the surrogate performance reported in this work is conditional on the fidelity and assumptions of the adopted reference numerical model. In the present PEWEC case study, the training dataset was generated using a linear FD model, selected as a compromise between computational efficiency and physical fidelity for preliminary design-space exploration and techno-economic performance assessment. If additional nonlinear effects were included, the resulting design-to-power mapping could change and consequently, the relative performance of the tested ML algorithms may also change. The superior performance of the FNN observed in this study should therefore be interpreted as an empirical result for the present PEWEC dataset and modeling assumptions. The present dataset does not allow the model-form deviation introduced by the linear hydrodynamic assumptions, or its effect on candidate ranking, to be quantified. A systematic assessment would require paired high-fidelity evaluations covering the same design and sea-state domains investigated in this work. Accordingly, a direct future research direction is the extension of the training and validation datasets using progressively higher-fidelity reference models, including nonlinear time-domain simulations, CFD calculations, and, where available, experimental measurements. These data would allow the surrogate to represent physical effects currently neglected by the linear FD model. This extension would also make it possible to reassess the prediction accuracy, the relative performance of the considered ML algorithms, and the computational advantage of the surrogate when more complex physical effects are represented. In practical applications, the present surrogate can be used for broad preliminary techno-economic design screening, while a set of promising and closely ranked configurations should subsequently be verified using partially nonlinear potential-flow, nonlinear time-domain, CFD, and, in conclusion experimental models for the detailed design and test phase within the development procedure of a WEC. Overall, the classifier-regressor framework is expected to remain applicable to high-fidelity datasets because its input and output structure would remain unchanged. However, the presently trained models cannot be transferred directly and would require new training, tuning, and independent validation. Since high-fidelity simulations are more computationally demanding, the potential inference-time advantage of the surrogate is expected to be greater, but both its accuracy and computational acceleration remain to be quantified. Future work will therefore evaluate the same regression algorithms considered in this study using data generated by nonlinear high-fidelity models.

The discrepancies observed for the relative errors of the AEP for the different evaluated sites can be explained by the fact that the FNN model was trained using a mean squared error loss function, which aims to reduce larger deviations. Therefore, the absolute error for those sea states and oceanic sites where the power production of the PEWEC is lower (*e.g.*, BIMEP) may be similar to that for those where the power production of the PEWEC is higher (*e.g.*, CDS). Nonetheless, the relative error can vary significantly. For this reason, in future work, we aim to compare the performance of the FNN model trained using the mean squared error with one using a relative error-based loss function.

In this study, we proved the effectiveness of the proposed methodology for modeling and assessing the PEWEC, but the methodology is generally transferable to any WEC design. To this end, our methodology can be updated, by replacing the *PEWEC hydrodynamic curves and dynamic response estimation* as well as the *PTO control for energy maximization* blocks with those corresponding to the desired WEC device to be studied. In this case, the proposed workflow remains unchanged. Nonetheless, it is important to highlight that for different WEC devices, the number and type of WEC design parameters to be considered to describe the whole device may vary. Therefore, the optimum hyperparameter configuration of the surrogate model classifier and regression algorithms may differ for varying WEC devices. Consequently, the term “transferability” should be understood at the workflow level, rather than at the level of the trained PEWEC surrogate model. The proposed structure can, in principle, be applied to other WEC archetypes by replacing the PEWEC-specific design parametrization, hydrodynamic and dynamic response model, PTO-control block, and feasibility criteria with those corresponding to the target device. In each case, a new dataset would have to be generated, and the classifier and regressor would need to be retrained, re-tuned, and revalidated. Accordingly, the present results should not be interpreted as a direct validation of the trained

PEWEC surrogate on other WEC concepts, nor as evidence that the FNN would necessarily remain the best-performing algorithm for all WEC archetypes. Different devices may lead to different design-space dimensionalities, feasibility constraints, nonlinearities, and output distributions, potentially modifying the relative performance of the tested ML algorithms. A dedicated validation on fundamentally different WEC archetypes represents an important future research direction.

The investigation compares the performance of different ML algorithms for developing the proposed surrogate model's two main components (*i.e.*, classifier and regressor). Furthermore, we employed an LHS filling strategy for data sampling in this work. We plan to investigate further and compare using active sampling strategies, such as Query by Committee. Additionally, in the present work, we used a class weighting approach to address the imbalanced distribution in the classification task. Future work may include studying and comparing other approaches, like data-level techniques for removing samples, *i.e.*, undersampling, and/or oversampling (*e.g.*, Synthetic Minority Oversampling Technique - SMOTE). A distinct future research direction concerns the residual risk of false-positive feasibility classifications. Future work will investigate possible architectures to improve the reliability and physical consistency of the resulting power predictions. Moreover, for the present analysis, we employed "black-box" machine learning algorithms for the regression task, meaning algorithms that have opaque decision-making processes for humans. In future studies, we aim to employ and compare the results obtained using physics-informed neural networks, such as DeepONet, which integrate physics knowledge with data-driven algorithms. Lastly, another relevant research direction concerns the extension of the presented work to advanced hybrid or meta-learning models, which have recently shown promising performance in wave-farm power prediction.

CRediT authorship contribution statement

Rafael Natalio Fontana Crespo: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Filippo Giorcelli:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Data curation, Conceptualization. **Edoardo Pasta:** Writing – review & editing, Validation, Supervision, Methodology, Conceptualization. **Alessandro Aliberti:** Writing – review & editing, Validation, Supervision, Methodology. **Lorenzo Bottaccioli:** Writing – review & editing, Validation, Supervision, Methodology. **Sergej Antonello Sirigu:** Writing – review & editing, Validation, Supervision, Project administration. **Giuliana Mattiazzo:** Supervision, Project administration. **Edoardo Patti:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Methodology, Conceptualization.

Acknowledgments and funding

The authors express their gratitude for the funds received from the Ministry of University and Research and from PNRR's PhD scholarship program defined by DM-352/2022 in collaboration with Eni S.p.A. This publication is part of the PNRR-NGEU project.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. WECs data-driven and data-based models literature overview

The predominant focus of Data-driven (DD) and Data-based (DB) approaches has been their integration into control strategies for WECs, rather than their use for comprehensive design capability assessment. In contrast, the present study specifically targets applications of these modeling techniques within the context of WEC performance modeling, *e.g.*, studies regarding WECs' system identification (both linear and nonlinear), power extraction prediction and electricity production forecasting, and system diagnosis. Besides, the remaining works aim to develop DD and DB control strategies to enhance information, suggesting alternative model-free control strategies (*i.e.*, capable of avoiding errors deriving from unmodeled hydrodynamics) and to exploit real-time data to describe WEC's behavior [77]. Relevant contributions aligning with the current investigation's focus are summarized and classified in Table A.22. The collected works have been classified with respect to three categories: the WEC archetype that is the object of the study (*i.e.*, Attenuator (A), Oscillating Water Column (OWC), Point Absorber (PA), and Terminator (T)), the nature of the data employed in the DD/DB framework (*i.e.*, Experimental (*Exp.*) or Numerical Simulation (*Num. Sim.*)), and whether the research is conducted on a fixed WEC design or for a set of possible system designs.

Fig. A.31 shows that the majority of the studies focus on PA (63.8%) and OWC (21.2%) devices, while only 10% and 5% develop DD and DB approaches for T and A type WECs, respectively.

Looking at the nature of the data employed to design and/or validate the meta-model approach, Fig. A.32 denotes that the share of numerical

Table A.22
List of the works analyzed in the study.

Reference	Data		WEC	Fixed Design
	Exp.	Num. Sim.		
[52]		•	PA	•
[130]	•		PA	•
[131]	•		OWC	•
[132]	•		PA	•
[133]	•		PA	•
[134]	•		PA	•
[135]	•		PA	•
[136]	•		OWC	•
[137]	•		PA	•
[79]	•		PA	
[138]		•	OWC	•
[139]	•	•	OWC	•
[140]	•	•	PA	•
[141]	•		PA	•
[142]	•		OWC	•
[60]	•	•	PA	
[143]	•		OWC	•
[56]		•	PA	
[48]		•	OWC	
[144]	•		A	•
[145]	•		T	•
[50]		•	OWC	
[55]		•	PA	
[54]		•	A	
[146]	•		OWC	•
[20]		•	PA	•
[147]	•		PA	•
[148]		•	PA	•
[47]		•	PA	
[149]		•	PA	•
[59]	•	•	OWC	
[150]	•		T	•
[151]		•	PA	•
[152]		•	T	•
[153]	•		OWC	•
[154]		•	PA	•
[155]	•		PA	•
[156]		•	OWC	•
[57]		•	PA	
[157]		•	PA	•
[158]	•		PA	•
[51]		•	A	
[159]		•	PA	•

(*continued on next page*)

Table A.22 *(continued)*

Reference	Data		WEC	Fixed Design
	Exp.	Num. Sim.		
[61]		•	OWC	
[160]		•	PA	•
[161]	•		A	•
[90]		•	PA	•
[62]		•	PA	
[162]		•	PA	•
[163]		•	PA	•
[164]	•		PA	•
[165]		•	PA	•
[166]		•	PA	•
[167]		•	PA	•
[26]	•		T	•
[49]		•	PA	
[168]		•	PA	•
[169]		•	PA	•
[170]		•	PA	•
[171]		•	T	•
[172]	•		PA	•
[173]		•	PA	•
[174]		•	PA	•
[175]	•		PA	•
[176]		•	OWC	•
[177]	•		OWC	•
[178]	•		OWC	•
[179]		•	PA	•
[180]		•	PA	•
[58]		•	PA	
[45]		•	PA	•
[42]	•	•	T	
[181]		•	PA	•
[41]		•	PA	
[38]		•	PA	
[53]		•	OWC	•
[37]		•	PA	
[43]	•	•	T	
[44]	•	•	T	
[40]		•	PA	
[39]		•	PA	

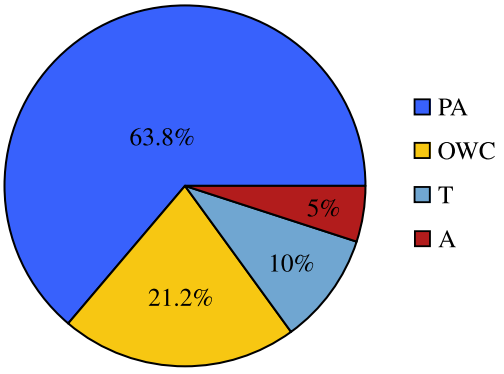


Fig. A.31. Percentage of analyzed types of WECs, classified according to their main working principle, dimension and orientation with respect to the incoming wave.

simulation results usage is greater than that of experimental data, *i.e.*, 58.7% versus 32.5% respectively. The choice of using only experimental data could constrain the range of possible design and operating sea states investigated due to the high economic and time costs required by an experimental campaign. Therefore, a portion of works (8.8%) employs both experimental and numerical data, *e.g.*, first using numerical simulations in order to train the DD/DB surrogate model, which is afterward validated via properly designed experimental tests.

Moreover, as illustrated in Fig. A.33, only 27.5% of the examined articles present a framework that spans a set of possible designs of the

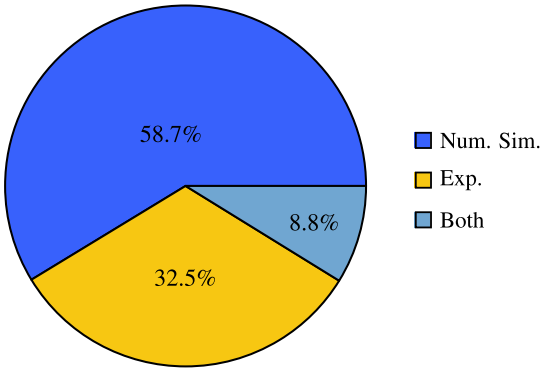


Fig. A.32. Percentage of employed data sources in the data-driven WEC surrogate models presented in the reviewed articles.

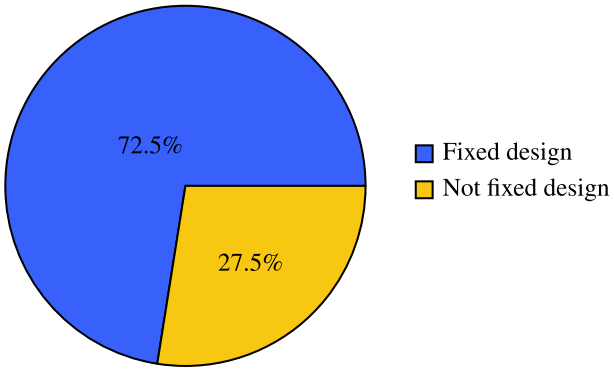


Fig. A.33. Percentage share between data-based models of WECs covering a set of design parameters (in yellow) (*i.e.*, non-fixed WEC design) and those developed for a specific design (in blue) (*i.e.*, fixed WEC design). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

WEC that is the object of the research. Varying geometrical or design parameters allows one to cover a wide range of possible solutions and this could help in design optimization frameworks, but at the cost of increasing the exploration space dimensions and therefore also the necessity to increase the amount of training data. In this context, the present study proposes a DB model-free strategy which applies to any WEC class.

Data availability

The data that have been used are confidential.

References

[1] E. Commission, D.-G. for Research, Innovation, Horizon Europe, Budget – Horizon Europe – The Most Ambitious EU Research & Innovation Programme Ever, Publications Office of the European Union, 2021, https://doi.org/10.2777/202859
[2] G. Mork, S. Barstow, A. Kabuth, M.T. Pontes, Assessing the global wave energy potential, in: International Conference on Offshore Mechanics and Arctic Engineering, vol. 49118, 2010, pp. 447–454.
[3] A. Ilyas, S.A. Kashif, M.A. Saqib, M.M. Asad, Wave electrical energy systems: implementation, challenges and environmental issues, Renew. Sustain. Energy Rev. 40 (2014) 260–268.
[4] T. Thorpe, A Brief Review of Wave Energy, a Report Produced for the UK Department of Trade and Industry, 1999.
[5] W. Adger, T. Hughes, C. Folke, S.R. Carpenter, J. Rockstrom, Social-ecological resilience to coastal disasters, Science 309 (5737) (2005) 1036–1039.
[6] A. Hoballah, C. Peter, P. Des Nations Unies Pour l'environnement, Sustainable, Resource Efficient Cities: Making It Happen!, UNEP, 2012.
[7] R. Satymov, D. Bogdanov, M. Dadashi, G. Lavidas, C. Breyer, Techno-economic assessment of global and regional wave energy resource potentials and profiles in hourly resolution, Appl. Energy 364 (2024) 123119.
[8] B. Guo, J.V. Ringwood, A review of wave energy technology from a research and commercial perspective, IET Renew. Power Gener. 15 (14) (2021) 3065–3090.

[9] G. Giorgi, How to generate electricity from waves – principles of wave energy converters, in: M.R. Rahimpour (Ed.), Encyclopedia of Renewable Energy, Sustainability and the Environment, First ed., Elsevier, Oxford, 2024, pp. 17–26, https://doi.org/10.1016/B978-0-323-93940-9.00238-3

[10] F. Carapellese, E. Pasta, B. Paduano, N. Faedo, G. Mattiazzo, Intuitive LTI energy-maximising control for multi-degree of freedom wave energy converters: the PeWEC case, Ocean Eng. 256 (2022) 111444.

[11] B. Guo, J.V. Ringwood, Geometric optimisation of wave energy conversion devices: a survey, Appl. Energy 297 (2021) 117100.

[12] A. Garcia-Teruel, D. Forehand, A review of geometry optimisation of wave energy converters, Renew. Sustain. Energy Rev. 139 (2021) 110593.

[13] A. Trueworthy, B. DuPont, The wave energy converter design process: methods applied in industry and shortcomings of current practices, J. Mar. Sci. Eng. 8 (11) (2020) 932.

[14] D. Golbaz, R. Asadi, E. Amini, H. Mehdipour, M. Nasiri, B. Etaati, S.T.O. Naeeni, M. Neshat, S. Mirjalili, A.H. Gandomi, Layout and design optimization of ocean wave energy converters: a scoping review of state-of-the-art canonical, hybrid, cooperative, and combinatorial optimization methods, Energy Rep. 8 (2022) 15446–15479.

[15] M. Penalba, J.V. Ringwood, A high-fidelity wave-to-wire model for wave energy converters, Renew. Energy 134 (2019) 367–378.

[16] M. Penalba, G. Giorgi, J.V. Ringwood, Mathematical modelling of wave energy converters: a review of nonlinear approaches, Renew. Sustain. Energy Rev. 78 (2017) 1188–1207.

[17] F. Carapellese, B. Paduano, N. Faedo, Performance optimisation of wave energy converters, in: Reference Module in Earth Systems and Environmental Sciences, 1st ed, Elsevier, 2024, pp. 89–104.

[18] J.V. Ringwood, S. Zhan, N. Faedo, Empowering wave energy with control technology: possibilities and pitfalls, Annu. Rev. Control 55 (2023) 18–44.

[19] C. Janiesch, P. Zschech, K. Heinrich, Machine learning and deep learning, Electron. Mark. 31 (3) (2021) 685–695.

[20] C. Ren, J. Tan, Y. Xing, ALK-PE: an efficient active learning kriging approach for wave energy converter power matrix estimation, Ocean Eng. 286 (2023) 115566, https://doi.org/10.1016/j.oceaneng.2023.115566, https://linkinghub.elsevier.com/retrieve/pii/S0029801823019509.

[21] P. Westermann, M. Welzel, R. Evins, Using a deep temporal convolutional network as a building energy surrogate model that spans multiple climate zones, Appl. Energy 278 (2020) 115563.

[22] Z. Miao, L. Margetts, A.N. Vasileiou, H. Yin, Surrogate model development using simulation data to predict weld residual stress: a case study based on the NeT-TG1 benchmark, Int. J. Press. Vessels Pip. 206 (2023) 105014.

[23] Z. Zhang, Z. Zhang, Z. Lei, R. Xiong, J. Cheng, S. Gao, Surrogate-assisted differential evolution for wave energy converters optimization, IEEE Trans. Emerg. Top. Comput. Intell. (2024) 1–10, https://doi.org/10.1109/TETCI.2024.3451612.

[24] E. Pasta, F. Carapellese, N. Faedo, P. Brandimarte, Data-driven control of wave energy systems using random forests and deep neural networks, Appl. Ocean Res. 140 (2023) 103749, https://doi.org/10.1016/j.apor.2023.103749, https://linkinghub.elsevier.com/retrieve/pii/S0141118723002900.

[25] D.G. Gioia, E. Pasta, P. Brandimarte, G. Mattiazzo, Data-driven control of a pendulum wave energy converter: a Gaussian process regression approach, Ocean Eng. 253 (2022) 111191.

[26] M. Ayyad, L. Yang, A. Ahmed, A. Shalaby, J. Huang, J. Mi, R. Datla, L. Zuo, M.R. Hajj, System identification of oscillating surge wave energy converter using physics-informed neural network, Appl. Energy 378 (2025) 124703, https://doi.org/10.1016/j.apenergy.2024.124703, https://linkinghub.elsevier.com/retrieve/pii/S0306261924020865.

[27] C. Stavropoulou, E. Katsidoniotaki, N. Faedo, M. Göteman, Multi-fidelity surrogate modeling of nonlinear dynamic responses in wave energy farms, Appl. Energy 380 (2025) 125011.

[28] A.P. Kyprioti, J. Zhang, A.A. Taflanidis, Adaptive design of experiments for global kriging metamodeling through cross-validation information, Struct. Multidiscip. Optim. 62 (2020) 1135–1157.

[29] M. Neshat, S. Mirjalili, N.Y. Sergiienko, S. Esmaeilzadeh, E. Amini, A. Heydari, D.A. Garcia, Layout optimisation of offshore wave energy converters using a novel multi-swarm cooperative algorithm with backtracking strategy: a case study from coasts of Australia, Energy 239 (2022) 122463.

[30] M. Neshat, B. Alexander, N.Y. Sergiienko, M. Wagner, Optimisation of large wave farms using a multi-strategy evolutionary framework, in: Proceedings of the 2020 Genetic and Evolutionary Computation Conference, 2020, pp. 1150–1158.

[31] J. Zhang, A.A. Taflanidis, J.T. Scruggs, Surrogate modeling of hydrodynamic forces between multiple floating bodies through a hierarchical interaction decomposition, J. Comput. Phys. 408 (2020) 109298.

[32] M. Neshat, E. Abbasnejad, Q. Shi, B. Alexander, M. Wagner, Adaptive neuro-surrogate-based optimisation method for wave energy converters placement optimisation, in: Neural Information Processing: 26th International Conference, ICONIP 2019, Sydney, NSW, Australia, December 12–15, 2019, Proceedings, Part II 26, Springer, 2019, pp. 353–366.

[33] M. Neshat, B. Alexander, N.Y. Sergiienko, M. Wagner, New insights into position optimisation of wave energy converters using hybrid local search, Swarm Evol. Comput. 59 (2020) 100744.

[34] C. Eskilsson, J. Palm, P. Johannesson, G.M. Paredes, Sensitivity analysis of extreme loads acting on a point-absorbing wave energy converter, Int. Mar. Energy J. 5 (1) (2022) 91–101.

[35] P.T. Nguyen, L. Manuel, R.G. Coe, On the development of an efficient surrogate model for predicting long-term extreme loads on a wave energy converter, J. Offshore Mech. Arct. Eng. 141 (6) (2019) 061103.

[36] E. Katsidoniotaki, F. Psarommatis, M. Göteman, Digital twin for the prediction of extreme loads on a wave energy conversion system, Energies 15 (15) (2022) 5464.

[37] S. Azad, D.R. Herber, S. Khanal, G. Jia, Site-dependent solutions of wave energy converter farms with surrogate models, control co-design, and layout optimization, in: 2024 American Control Conference (ACC), IEEE, 2024, pp. 2572–2579.

[38] S. Azad, D.R. Herber, Concurrent probabilistic control co-design and layout optimization of wave energy converter farms using surrogate modeling, in: International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, vol. 87318, American Society of Mechanical Engineers, 2023, V03BT03A035.

[39] S. Azad, D.R. Herber, S. Khanal, G. Jia, Concurrent geometry, control, and layout optimization of wave energy converter farms in probabilistic irregular waves using surrogate modeling, Ocean Eng. 320 (2025) 120183.

[40] Y. Peña-Sanchez, M. Centeno-Telleria, A. Zarketa-Astigarraga, M. Penalba, D. García-Violini, N. Faedo, J. Ringwood, Efficient wave energy converter optimisation via control co-design: a comparison of AI-based algorithms, in: Innovations in Renewable Energies Offshore, CRC Press, 2024, pp. 535–542.

[41] T. Liu, Y. Liu, S. Huang, G. Xue, Shape optimization of oscillating buoy wave energy converter based on the mean annual power prediction model, Energies 15 (20) (2022) 7470.

[42] Z. Liu, Y. Wang, X. Hua, Prediction and optimization of oscillating wave surge converter using machine learning techniques, Energy Convers. Manag. 210 (2020) 112677.

[43] Y. Wang, Z. Liu, A bionic design of oscillating wave surge energy converter based on scallops, Energy 304 (2024) 132236.

[44] Y. Wang, Z. Liu, T. Tao, A novel oscillating wave surge converter by hollowing out the flap bottom: numerical studies using smoothed particle hydrodynamics, Ocean Eng. 303 (2024) 117694.

[45] E. Faraggiana, J. Chapman, A. Williams, I. Masters, Genetic based optimisation of the design parameters for an array-on-device orbital motion wave energy converter, Ocean Eng. 218 (2020) 108251.

[46] E. Faraggiana, M. Sirigu, A. Ghigo, E. Petracca, G. Mattiazzo, G. Bracco, Conceptual design and optimisation of a novel hybrid device for capturing offshore wind and wave energy, J. Ocean Eng. Mar. Energy 10 (1) (2024) 35–56.

[47] C. Jiang, F. Cao, D. Li, Z. Wei, H. Shi, A double-objective prediction and optimization method for buoys performance based on the artificial neural network, Ocean Eng. 282 (2023) 114969, https://doi.org/10.1016/j.oceaneng.2023.114969, https://linkinghub.elsevier.com/retrieve/pii/S0029801823013537.

[48] D. Hithaish, T.K. Das, M. Takao, A. Samad, Design optimization of a fluidic diode for a wave energy converter via artificial intelligence-based technique, Arab. J. Sci. Eng. 48 (9) (2023) 11407–11423.

[49] W.A. Arrosyid, K.T. Waskito, Y. Nasruddin, Sholahudin, A. Geraldi, Y. Zhao, Hydrodynamic performance and multi-objective optimization of multi-cylinder floating point absorber wave energy converter, Ocean Eng. 317 (2025) 120040, https://doi.org/10.1016/j.oceaneng.2024.120040, https://linkinghub.elsevier.com/retrieve/pii/S002980182403378X.

[50] V. Vipin, K. Trivedi, S. Koley, Optimization of parameters of the OWC wave energy converter device using MLP and XGBoost models, Results Phys. 55 (2023) 107163, https://doi.org/10.1016/j.rinp.2023.107163, https://linkinghub.elsevier.com/retrieve/pii/S2211379723009567.

[51] V. Vipin, S. Koley, Power generation prediction and optimization of parameters of piezoelectric wave energy converter device, in: 2024 Fourth International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), IEEE, 2024, pp. 1–7, https://doi.org/10.1109/ICAECT60202.2024.10469022, https://ieeexplore.ieee.org/document/10469022/.

[52] M.K. Colby, E.M. Nasroullahi, K. Tumer, Optimizing ballast design of wave energy converters using evolutionary algorithms, in: Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation, 2011, pp. 1739–1746.

[53] A.T. Kotb, M.A. Nawar, Y.A. Attai, M.H. Mohamed, Performance optimization of a modified wells turbine for wave energy conversion, Ocean Eng. 280 (2023) 114849.

[54] V. Vipin, S. Koley, Design optimization of a submerged piezoelectric wave energy converter device using an artificial neural network model, Energy Rep. 9 (2023) 322–326, https://doi.org/10.1016/j.egyr.2023.09.036, https://linkinghub.elsevier.com/retrieve/pii/S235248472301274X.

[55] S.K. Poguluri, D. Kim, Y. Lee, J.-H. Shin, Y.H. Bae, Design optimization of asymmetric wave energy converter using artificial neural network model, Int. J. Nav. Archit. Ocean Eng. 15 (2023) 100529, https://doi.org/10.1016/j.ijnaoe.2023.100529, https://linkinghub.elsevier.com/retrieve/pii/S2092678223000183.

[56] P. Khatri, X. Wang, Design parameter sensitivity analysis and performance prediction of a novel direct drive double dumbbell flux switching linear generator, J. Ocean Eng. Mar. Energy (2022) 1–18.

[57] S.K. Poguluri, Y.H. Bae, Enhancing wave energy conversion efficiency through supervised regression machine learning models, J. Mar. Sci. Eng. 12 (1) (2024) 153, https://doi.org/10.3390/jmse12010153, https://www.mdpi.com/2077-1312/12/1/153.

[58] H.-J. Koh, W.-S. Ruy, I.-H. Cho, H.-M. Kweon, Multi-objective optimum design of a buoy for the resonant-type wave energy converter, J. Mar. Sci. Technol. 20 (2015) 53–63.

[59] C.-C. Lin, Y.-C. Chow, D.T. Nguyen, Geometrical design of the L-shaped oscillating water column using artificial neural network, Proc. Eur. Wave Tidal Energy Conf. 15 (2023), https://doi.org/10.36688/ewtec-2023-555, https://submissions.ewtec.org/proc-ewtec/article/view/555.

[60] X. Bao, W. Xiao, S. Li, G. Iglesias, Parametric study and optimization of a two-body wave energy converter, IET Renew. Power Gener. 15 (14) (2021) 3319–3330.

[61] R. Rao, H. Raj, V. Vipin, S. Koley, Power generation prediction of OWC device using deep neural network model, in: 2024 Third International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT), IEEE, 2024, pp. 1–8, https://doi.org/10.1109/ICEEICT61591.2024.10718532, https://ieeexplore.ieee.org/document/10718532/.

[62] W. Lin, X. Li, L. Zuo, Shape optimization of a point absorber wave energy converter for reduced current drag and improved wave energy capture using neural networks and genetic algorithms, IEEE Trans. Sustain. Energy 15 (4) (2024) 2758–2768, https://doi.org/10.1109/TSTE.2024.3443117, https://ieeexplore.ieee.org/document/10638322/.

[63] O. Dell'edera, F. Niosi, P. Casalone, M. Bonfanti, B. Paduano, G. Mattiazzo, Understanding wave energy converters dynamics: high-fidelity modeling and validation of a moored floating body, Appl. Energy 376 (2024) 124202.

[64] M. Folley, Numerical Modelling of Wave Energy Converters: State-of-the-Art Techniques for Single Devices and Arrays, Academic Press, 2016.

[65] B. Paduano, L. Parrinello, F. Niosi, O. Dell'edera, S.A. Sirigu, N. Faedo, G. Mattiazzo, Towards standardised design of wave energy converters: a high-fidelity modelling approach, Renew. Energy 224 (2024) 120141.

[66] W. Cummins, The Impulse Response Function and Ship Motion, Report 1661, Department of the Navy, David W. Taylor Model Basin, Hydromechanics Laboratory, Research and Development Report, Oct 1962.

[67] N. Faedo, Y. Peña-Sanchez, J.V. Ringwood, Finite-order hydrodynamic model determination for wave energy applications using moment-matching, Ocean Eng. 163 (2018) 251–263.

[68] O. Faltinsen, Sea Loads on Ships and Offshore Structures, vol. 1, Cambridge University Press, 1993.

[69] T.F. Ogilvie, Recent progress toward the understanding and prediction of ship motions, in: Proc. of the 5th Symposium on Navan Hydrodynamics, Bergen, Norway, 1964, pp. 3–79.

[70] G.B. Airy, Tides and waves, in: Encyc. Metrop, vol. 192, William Clowes and Sons, 1845, pp. 241–396.

[71] I. Wamit, et al., Wamit User Manual, WAMIT Inc.: Chestnut Hill, MA, USA, 2006 527.

[72] Orcina, Orcina validation report, available at 2015. https://www.orcina.com/resources/validation/.

[73] A. Babarit, G. Delhommeau, Theoretical and numerical aspects of the open source BEM solver NEMOH, in: 11th European Wave and Tidal Energy Conference (EWTEC2015), 2015.

[74] Orcina, OrcaFlex—Documentation, 10.1 b Edition (2016.

[75] K. Ruehl, A. Keester, D. Forbush, J. Grasberger, S. Husain, J. Leon, D. Ogden, M. Shabara, WEC-Sim V6.1, Sep 2024, https://doi.org/10.5281/zenodo.10023797

[76] M. Hall, MoorDyn User's Guide, Department of Mechanical Engineering, University of Maine: Orono, ME, USA, 2015 15.

[77] E. Pasta, N. Faedo, G. Mattiazzo, J.V. Ringwood, Towards data-driven and data-based control of wave energy systems: classification, overview, and critical assessment, Renew. Sustain. Energy Rev. 188 (2023) 113877.

[78] Z.-S. Hou, Z. Wang, From model-based control to data-driven control: survey, classification and perspective, Inf. Sci. 235 (2013) 3–35.

[79] E. Pasta, G. Papini, Y. Peña-Sanchez, F.D. Mosquera, F. Ferri, N. Faedo, Data-based modelling of arrays of wave energy systems: experimental tests, models, and validation, Control Eng. Pract. 148 (2024) 105949, https://doi.org/10.1016/j.conengprac.2024.105949, https://www.sciencedirect.com/science/article/pii/S0967066124001096.

[80] M. Masoumi, B. Estejab, F. Henry, Implementation of machine learning techniques for the analysis of wave energy conversion systems: a comprehensive review, J. Ocean Eng. Mar. Energy (2024) 1–30.

[81] A. Shadmani, M.R. Nikoo, A.H. Gandomi, R.-Q. Wang, B. Golparvar, A review of machine learning and deep learning applications in wave energy forecasting and WEC optimization, Energy Strategy Rev. 49 (2023) 101180.

[82] E. Pasta, B. Paduano, G. Mattiazzo, N. Faedo, J.V. Ringwood, On data-based control-oriented modelling applications in wave energy systems, Proc. Eur. Wave Tidal Energy Conf. 15 (2023), https://doi.org/10.36688/ewtec-2023-409, https://submissions.ewtec.org/proc-ewtec/article/view/409.

[83] M. Neshat, N.Y. Sergiienko, A. Rafiee, S. Mirjalili, A.H. Gandomi, J. Boland, Meta wave learner: predicting wave farms power output using effective meta-learner deep gradient boosting model: a case study from Australian coasts, Energy 304 (2024) 132122.

[84] M. Neshat, S. Dadgar, S. Ziaee, S. Mirjalili, N. Markovska, A.H. Gandomi, Effective and accurate wave farm power prediction using cooperative multi-objective covariance adaptation gradient ensemble model with comprehensive comparative experiments, Renew. Energy (2026) 125695.

[85] R. Kumari, S.K. Srivastava, Machine learning: a review on binary classification, Int. J. Comput. Appl. 160 (7) (2017).

[86] Y.-C. Chang, K.-H. Chang, G.-J. Wu, Application of eXtreme gradient boosting trees in the construction of credit risk assessment models for financial institutions, Appl. Soft Comput. 73 (2018) 914–920.

[87] C.-W. Wang, Y.-C. Lee, E. Calista, F. Zhou, H. Zhu, R. Suzuki, D. Komura, S. Ishikawa, S.-P. Cheng, A benchmark for comparing precision medicine methods in thyroid cancer diagnosis using tissue microarrays, Bioinformatics 34 (10) (2018) 1767–1773.

[88] C.M. Bishop, N.M. Nasrabadi, Pattern Recognition and Machine Learning, vol. 4, Springer, 2006.

[89] J. Brownlee, Imbalanced Classification with Python: Better Metrics, Balance Skewed Classes, Cost-Sensitive Learning, Machine Learning Mastery, 2020.

[90] F. Giorcelli, S.A. Sirigu, G. Giorgi, N. Faedo, M. Bonfanti, J. Ramello, E. Giorcelli, G. Mattiazzo, Measuring the robustness of optimal design solutions for wave energy converters via a stochastic approach, J. Mar. Sci. Eng. 12 (3) (2024) 482.

[91] S.A. Sirigu, L. Foglietta, G. Giorgi, M. Bonfanti, G. Cervelli, G. Bracco, G. Mattiazzo, Techno-economic optimisation for a wave energy converter via genetic algorithm, J. Mar. Sci. Eng. 8 (7) (2020) 482.

[92] C. Windt, N. Faedo, M. Penalba, F. Dias, J.V. Ringwood, Reactive control of wave energy devices–the modelling paradox, Appl. Ocean Res. 109 (2021) 102574.

[93] B. Paduano, F. Carapellese, E. Pasta, S.A. Sirigu, N. Faedo, G. Mattiazzo, Data-based control synthesis and performance assessment for moored wave energy conversion systems: the PeWEC case, IEEE Trans. Sustain. Energy 15 (1) (2023) 355–367.

[94] M. Bonfanti, S.A. Sirigu, Spectral-domain modelling of a non-linear wave energy converter: analytical derivation and computational experiments, Mech. Syst. Signal Process. 198 (2023) 110398.

[95] A. Pecher, J.P. Kofoed, Handbook of Ocean Wave Energy, Springer Nature, 2017.

[96] K. Hasselmann, T.P. Barnett, E. Bouws, H. Carlson, D.E. Cartwright, K. Enke, J. Ewing, A. Gienapp, D. Hasselmann, P. Kruseman, et al., Measurements of Wind-Wave Growth and Swell Decay During the Joint North Sea Wave Project (JONSWAP), Ergaenzungsheft zur Deutschen Hydrographischen Zeitschrift, Reihe A (1973.

[97] E. Giglio, E. Petracca, B. Paduano, C. Moscoloni, G. Giorgi, S.A. Sirigu, Estimating the cost of wave energy converters at an early design stage: a bottom-up approach, Sustainability 15 (8) (2023) 6756.

[98] M.D. McKay, R.J. Beckman, W.J. Conover, A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, Technometrics 42 (1) (2000) 55–61.

[99] B. Krawczyk, Learning from imbalanced data: open challenges and future directions, Prog. Artif. Intell. 5 (4) (2016) 221–232.

[100] Z.-H. Zhou, X.-Y. Liu, On multi-class cost-sensitive learning, Comput. Intell. 26 (3) (2010) 232–257.

[101] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-Learn: machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.

[102] J. Bergstra, R. Bardenet, Y. Bengio, B. Kégl, Algorithms for hyper-parameter optimization, Adv. Neural Inf. Process. Syst. 24 (2011).

[103] M. Neshat, From Grid Search to Modern Deterministic Ideas: A Brief Review of Hyper-Parameter Optimisation in Deep and Foundation Model Era, 2026.

[104] R. Khalid, N. Javaid, A survey on hyperparameters optimization algorithms of forecasting models in smart grid, Sustain. Cities Soc. 61 (2020) 102275.

[105] F. Salehi, E. Abbasi, B. Hassibi, The impact of regularization on high-dimensional logistic regression, Adv. Neural Inf. Process. Syst. 32 (2019).

[106] L. Breiman, Random forests, Mach. Learn. 45 (2001) 5–32.

[107] Z.E. Aydin, Z.K. Ozturk, Performance analysis of XGBoost classifier with missing data, Manch. J. Artif. Intell. Appl. Sci. (MJAIAS) 2 (2) (2021) 2021.

[108] A. Aliberti, Y. Xin, A. Viticchié, E. Macii, E. Patti, Comparative analysis of neural networks techniques to forecast airfare prices, in: 2023 IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC), IEEE, 2023, pp. 1023–1029.

[109] T. Chen, C. Guestrin, XGBoost: a scalable tree boosting system, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785–794.

[110] C.-C. Chang, C.-J. Lin, LIBSVM: a library for support vector machines, ACM Trans. Intell. Syst. Technol. 2 (3) (2011) 1–27.

[111] T.L. Fine, Feedforward Neural Network Methodology, Springer Science & Business Media, 2006.

[112] F. Chollet, et al., Keras, 2015. https://keras.io., (Accessed: 25 October 2023).

[113] G. Wang, G.B. Giannakis, J. Chen, Learning ReLU networks on linearly separable data: algorithm, optimality, and generalization, IEEE Trans. Signal Process. 67 (9) (2019) 2357–2370.

[114] D.P. Kingma, J. Ba, Adam: a method for stochastic optimization, arXiv preprint arXiv:1412.6980, 2014.

[115] L. Prechelt, Early stopping-but when? in: Neural Networks: Tricks of the Trade, Springer, 2002, pp. 55–69.

[116] J. Hensman, N. Fusi, N.D. Lawrence, Gaussian processes for big data, arXiv preprint arXiv:1309.6835, 2013.

[117] A.G.D.G. Matthews, M. Van Der Wilk, T. Nickson, K. Fujii, A. Boukouvalas, P. Le, Z. Ghahramani, J. Hensman, et al., GPflow: a Gaussian process library using TensorFlow, J. Mach. Learn. Res. 18 (40) (2017) 1–6.

[118] B. Karlik, A.V. Olgac, Performance analysis of various activation functions in generalized MLP architectures of neural networks, Int. J. Artif. Intell. Expert Syst. 1 (4) (2011) 111–122.

[119] J. Chenou, G. Hsieh, T. Fields, Radial basis function network: its robustness and ability to mitigate adversarial examples, in: 2019 International Conference on Computational Science and Computational Intelligence (CSCI), IEEE, 2019, pp. 102–106.

[120] R. Neruda, P. Kudová, Learning methods for radial basis function networks, Future Gener. Comput. Syst. 21 (7) (2005) 1131–1142.

[121] P. Vidnerová, RBF-keras: an RBF layer for keras library, 2019. https://github.com/PetraVidnerova/rbf_keras., (Accessed: 23 May 2025).

[122] H. He, Y. Ma, Imbalanced Learning: Foundations, Algorithms, and Applications, John Wiley & Sons, 2013.

[123] J. Davis, M. Goadrich, The relationship between precision-recall and ROC curves, in: Proceedings of the 23rd International Conference on Machine Learning, 2006, pp. 233–240.
[124] A. Forrester, A. Sobester, A. Keane, Engineering Design via Surrogate Modelling: A Practical Guide, John Wiley & Sons, 2008.
[125] S.M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, Adv. Neural Inf. Process. Syst. 30 (2017).
[126] R. Group, MedWind project, https://medwind.it/.
[127] J. Muñoz-Sabater, E. Dutra, A. Agustí-Panareda, C. Albergel, G. Arduini, G. Balsamo, S. Boussetta, M. Choulga, S. Harrigan, H. Hersbach, et al., ERA5-land: a state-of-the-art global reanalysis dataset for land applications, Earth Syst. Sci. Data 13 (9) (2021) 4349–4383.
[128] C.M. Service, Mediterranean Sea waves analysis and forecast, https://data.marine.copernicus.eu.
[129] O.N.N. Exchange, tensorflow-onnx https://github.com/onnx/tensorflow-onnx (Accessed: 8 August 2025)..
[130] S. Chandrasekaran, A. Amarkarthik, K. Sivakumar, D. Selvamuthukumaran, S. Sidney, Experimental investigation and ANN modeling on improved performance of an innovative method of using heave response of a non-floating object for ocean wave energy conversion, Front. Energy 7 (2013) 279–287.
[131] I. López, G. Iglesias, Efficiency of OWC wave energy converters: a virtual laboratory, Appl. Ocean Res. 44 (2014) 63–70.
[132] H. Huo, Y. Ji, S. Wang, X. Kuang, C. Yang, The research on AdaBoost-BPNN model of point absorber wave energy converter, in: 2014 IEEE International Conference on Mechatronics and Automation, IEEE, 2014, pp. 1762–1766.
[133] S. Nagulan, J. Selvaraj, A. Arunachalam, K. Sivanandam, Performance of artificial neural network in prediction of heave displacement for non-buoyant type wave energy converter, IET Renew. Power Gener. 11 (1) (2017) 81–84.
[134] C. Ni, X. Ma, Prediction of wave power generation using a convolutional neural network with multiple inputs, Energies 11 (8) (2018) 2097.
[135] H.M. Deberneh, I. Kim, Predicting output power for nearshore wave energy harvesting, Appl. Sci. 8 (4) (2018) 566.
[136] M.P. Fernandes, S.M. Vieira, J.C. Henriques, D. Valério, L.M. Gato, Short-term prediction in an oscillating water column using artificial neural networks, in: 2018 International Joint Conference on Neural Networks (IJCNN), IEEE, 2018, pp. 1–7.
[137] C. Ni, X. Ma, J. Wang, Integrated deep learning model for predicting electrical power generation from wave energy converter, in: 2019 25th International Conference on Automation and Computing (ICAC), IEEE, 2019, pp. 1–6.
[138] T.K. Das, A. Samad, Surrogate-based optimization of a biplane wells turbine, in: Proceedings of the Fourth International Conference in Ocean Engineering (ICOE2018), vol. 2, Springer, 2018, pp. 639–648.
[139] T. Chakraborty, M. Majumder, Impact of extreme events on conversion efficiency of wave energy converter, Energy Sci. Eng. 8 (10) (2020) 3441–3456.
[140] F. Niosi, L. Parrinello, B. Paduano, E. Pasta, F. Carapellese, G. Bracco, On the influence of mooring in wave energy converters productivity: the pewec case, in: 2021 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME), IEEE, 2021, pp. 1–6.
[141] C. Ni, Data-driven models for short-term ocean wave power forecasting, IET Renew. Power Gener. 15 (10) (2021) 2228–2236.
[142] D. Seo, T. Huh, M. Kim, J. Hwang, D. Jung, Prediction of air pressure change inside the chamber of an oscillating water column–wave energy converter using machine-learning in big data platform, Energies 14 (11) (2021) 2982.
[143] J.M. Silva, S.M. Vieira, D. Valério, J.C. Henriques, P.D. Sclavounos, Air pressure forecasting for the mutriku oscillating-water-column wave power plant: review and case study, IET Renew. Power Gener. 15 (14) (2021) 3485–3503.
[144] J. Zhang, X. Zhao, D. Greaves, S. Jin, Modeling of a hinged-raft wave energy converter via deep operator learning and wave tank experiments, Appl. Energy 341 (2023) 121072, https://doi.org/10.1016/j.apenergy.2023.121072, https://linkinghub.elsevier.com/retrieve/pii/S0306261923004361.
[145] C. Ni, W. Peng, An integrated approach using empirical wavelet transform and a convolutional neural network for wave power prediction, Ocean Eng. 276 (2023) 114231, https://doi.org/10.1016/j.oceaneng.2023.114231, https://linkinghub.elsevier.com/retrieve/pii/S0029801823006157.
[146] M. Rosati, J. Ringwood, H. Bingham, B. Joensen, K. Nielsen, A Data-Based Modelling Approach for a Vented Oscillating Water Column Wave Energy Converter, 1st ed, CRC Press, 2022, pp. 339–347. https://doi.org/10.1201/9781003360773-39
[147] Z. Shahroozi, M. Göteman, J. Engström, A neural network approach to minimize line forces in the survivability of the point-absorber wave energy converters, in: Volume 8: Ocean Renewable Energy, American Society of Mechanical Engineers, 2023, V008T09A065. https://doi.org/10.1115/OMAE2023-102422, https://asmedigitalcollection.asme.org/OMAE/proceedings/OMAE2023/86908/V008T09A065/1167316.
[148] Z. Liu, C. McGregor, S. Ding, X. Wang, Study of a three-dimensional model simulation of a speed amplified flux switching linear generator for wave energy conversion and its design optimization in the ocean environment, Energy 284 (2023) 128625, https://doi.org/10.1016/j.energy.2023.128625, https://linkinghub.elsevier.com/retrieve/pii/S0360544223020194.
[149] Y. Wu, W. Sheng, C.J. Taylor, G. Aggidis, X. Ma, Long-short term memory based TALOS wave energy converter power output prediction with numerical modelling, in: ISOPE International Ocean and Polar Engineering Conference, ISOPE, 2023, pp. ISOPE–I.
[150] M. Ayyad, A. Ahmed, L. Yang, M.R. Hajj, R. Datla, L. Zuo, System identification of OSWEC response using physics-informed neural network, in: OCEANS 2023 - Limerick, IEEE, 2023, pp. 1–5, https://doi.org/10.1109/OCEANSLimerick52467.2023.10244631, https://ieeexplore.ieee.org/document/10244631/.
[151] R.N. Fontana Crespo, A. Aliberti, L. Bottaccioli, E. Macii, G. Fighera, E. Patti, LSTM for grid power forecasting in short-term from wave energy converters, in: 2023 IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC), IEEE, 2023, pp. 1495–1500.
[152] B. Lydon, B. Polagye, S. Brunton, Nonlinear WEC modeling using sparse identification of nonlinear dynamics (SINDy), Proc. Eur. Wave Tidal Energy Conf. 15 (2023), https://doi.org/10.36688/ewtec-2023-383, https://submissions.ewtec.org/proc-ewtec/article/view/383.
[153] F. M'zoughi, J. Lekube, I. Garrido, A.J. Garrido, Machine learning-based OWC diagnosis using real measured data from wave power plants, in: 2023 IEEE PES 15th Asia-Pacific Power and Energy Engineering Conference (APPEEC), IEEE, 2023, pp. 1–6, https://doi.org/10.1109/APPEEC57400.2023.10561986, https://ieeexplore.ieee.org/document/10561986/.
[154] B.-C. Chen, Y.-H. Yu, A preliminary study of learning a wave energy converter system using physics-informed neural network method, in: Volume 8: Ocean Renewable Energy, American Society of Mechanical Engineers, 2023, V008T09A077. https://doi.org/10.1115/OMAE2023-105123, https://asmedigitalcollection.asme.org/OMAE/proceedings/OMAE2023/86908/V008T09A077/1167364.
[155] I. McLeod, D. Clemente, P.R. Santos, F.T. Pinto, M. Rosati, J.V. Ringwood, Mathematical modelling of a TENG-powered data buoy, in: OCEANS 2023 - Limerick, IEEE, 2023, pp. 1–10, https://doi.org/10.1109/OCEANSLimerick52467.2023.10244462, https://ieeexplore.ieee.org/document/10244462/.
[156] X. Bao, G. Huang, M. Liu, H. Sun, G. Iglesias, Turbine fault diagnosis of the oscillating water column wave energy converter based on multi-lead residual neural networks, Ocean Eng. 291 (2024) 116429, https://doi.org/10.1016/j.oceaneng.2023.116429, https://linkinghub.elsevier.com/retrieve/pii/S0029801823028135.
[157] J. Kang, X. Zhu, L. Shen, M. Li, Fault diagnosis of a wave energy converter gearbox based on an Adam optimized CNN-LSTM algorithm, Renew. Energy 231 (2024) 121022, https://doi.org/10.1016/j.renene.2024.121022, https://linkinghub.elsevier.com/retrieve/pii/S0960148124010905.
[158] Z. Shahroozi, M. Göteman, J. Engström, Neural network survivability approach of a wave energy converter considering uncertainties in the prediction of future knowledge, Renew. Energy 228 (2024) 120662, https://doi.org/10.1016/j.renene.2024.120662, https://linkinghub.elsevier.com/retrieve/pii/S0960148124007304.
[159] S. Xu, S. Wang, C. Guedes Soares, Prediction of mooring tensions of a wave energy converter considering the effects of nonlinear axial stiffness by a deep learning neural network, Ocean Eng. 305 (2024) 117810, https://doi.org/10.1016/j.oceaneng.2024.117810, https://linkinghub.elsevier.com/retrieve/pii/S002980182401148X.
[160] Y. Zhang, S. Liu, Q. Shen, L. Zhang, Y. Li, Z. Hou, R. Chen, Short-term prediction model of wave energy converter generation power based on CNN-BiLSTM-DELA integration, Electronics 13 (21) (2024) 4163, https://doi.org/10.3390/electronics13214163, https://www.mdpi.com/2079-9292/13/21/4163.
[161] L. Wang, C. Wen, S. Wu, S. Wu, Electric power prediction of a two-body hinge-barge wave energy converter using machine learning techniques, Ocean Eng. 305 (2024) 117935, https://doi.org/10.1016/j.oceaneng.2024.117935, https://linkinghub.elsevier.com/retrieve/pii/S0029801824012733.
[162] R.N. Fontana Crespo, A. Aliberti, L. Bottaccioli, E. Pasta, S.A. Sirigu, E. Macii, G. Mattiazzo, E. Patti, A comparative analysis of machine learning techniques for short-term grid power forecasting and uncertainty analysis of wave Energy converters, Eng. Appl. Artif. Intell. 138 (2024) 109352, https://doi.org/10.1016/j.engappai.2024.109352, https://linkinghub.elsevier.com/retrieve/pii/S0952197624015100.
[163] C. Quilodrán-Casas, Q. Li, N. Zhang, S. Cheng, S. Yan, Q. Ma, R. Arcucci, Exploring unseen 3D scenarios of physics variables using machine learning-based synthetic data: an application to wave energy converters, Environ. Model. Softw. 177 (2024) 106051, https://doi.org/10.1016/j.envsoft.2024.106051, https://linkinghub.elsevier.com/retrieve/pii/S1364815224001129.
[164] S. Xu, S. Wang, C. Guedes Soares, A hybrid deep learning approach to predict dynamic mooring tension of a wave energy converter, in: Volume 7: Ocean Renewable Energy, American Society of Mechanical Engineers, 2024, V007T09A078. https://doi.org/10.1115/OMAE2024-129671, https://asmedigitalcollection.asme.org/OMAE/proceedings/OMAE2024/87851/V007T09A078/1202746.
[165] A. Subramanian, S. Zou, K. Zhou, Y. Su, Data-driven fault diagnosis of mooring systems in wave energy converters, in: OCEANS 2024 - Halifax, IEEE, 2024, pp. 1–10, https://doi.org/10.1109/OCEANS55160.2024.10753944, https://ieeexplore.ieee.org/document/10753944/.
[166] T. Kawagishi, M. Murai, J. Han, On the applicability of DNN-based models trained under the fixed spar condition to control force estimation of a TLP moored PA-WEC, in: Volume 4: Ocean Space Utilization, American Society of Mechanical Engineers, 2024, V004T05A019. https://doi.org/10.1115/OMAE2024-122100, https://asmedigitalcollection.asme.org/OMAE/proceedings/OMAE2024/87813/V004T05A019/1202441.
[167] O.M. Gursel, P. Yanik, H.B. Karayaka, Integration of machine learning for enhanced wave energy converter power output estimation, in: 2024 56th North American Power Symposium (NAPS), IEEE, 2024, pp. 1–6, https://doi.org/10.1109/NAPS61145.2024.10741837, https://ieeexplore.ieee.org/document/10741837/.
[168] Q. Zhang, H. Zhang, X. Zhao, J. Ding, D. Xu, Multiple-input operator network prediction method for nonlinear wave energy converter, Ocean Eng. 317 (2025) 120106, https://doi.org/10.1016/j.oceaneng.2024.120106, https://linkinghub.elsevier.com/retrieve/pii/S0029801824034449.

[169] M.M. Abaei, E. Arzaghi, M. Bao, V. Garaniya, N. Abdussamie, A. Pichard, R. Abbassi, Performance evaluation of point-absorber wave energy converters; energy extraction and structural integrity aspects, Ocean Eng. 317 (2025) 119983, https://doi.org/10.1016/j.oceaneng.2024.119983, https://linkinghub.elsevier.com/retrieve/pii/S0029801824033213.

[170] S. Giorgi, J. Davidson, J.V. Ringwood, Identification of wave energy device models from numerical wave tank data—Part 2: Data-based model determination, IEEE Trans. Sustain. Energy 7 (3) (2016) 1020–1027.

[171] M.A.A. Bakar, D.A. Green, A.V. Metcalfe, N.M. Ariff, Unscented Kalman filtering for wave energy converters system identification, in: AIP Conference Proceedings, vol. 1602, American Institute of Physics, 2014, pp. 304–310.

[172] G. Bacelli, R.G. Coe, D. Patterson, D. Wilson, System identification of a heaving point absorber: design of experiment and device modeling, Energies 10 (4) (2017) 472.

[173] J. Davidson, S. Giorgi, J.V. Ringwood, Linear parametric hydrodynamic models for ocean wave energy converters identified from numerical wave tank experiments, Ocean Eng. 103 (2015) 31–39.

[174] M. Farajvand, D. García-Violini, J.V. Ringwood, Representative linearised models for a wave energy converter using various levels of force excitation, Ocean Eng. 270 (2023) 113635.

[175] S. Giorgi, J. Davidson, M. Jakobsen, M. Kramer, J.V. Ringwood, Identification of dynamic models for a wave energy converter from experimental data, Ocean Eng. 183 (2019) 426–436.

[176] G. Gkikas, G. Athanassoulis, Development of a novel nonlinear system identification scheme for the pressure fluctuation inside an oscillating water column-wave energy converter Part I: Theoretical background and harmonic excitation case, Ocean Eng. 80 (2014) 84–99.

[177] M. Rosati, T. Kelly, J.V. Ringwood, Nonlinear data-based hydrodynamic modeling of a fixed oscillating water column wave energy device, IEEE Access 9 (2021) 149756–149765.

[178] B. Stappenbelt, M. Fiorentini, P. Cooper, S.-P. Zhu, J.-R. Nader, System identification of a floating oscillating water column wave energy converter, in: International Conference on Offshore Mechanics and Arctic Engineering, vol. 44373, 2011, pp. 521–527.

[179] N. Faedo, F.J.D. Piuma, G. Giorgi, G. Bracco, J.V. Ringwood, G. Mattiazzo, Data-driven nonlinear model reduction by moment-matching for the ISWEC system, in: 2021 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME), IEEE, 2021, pp. 1–6.

[180] G. Papini, F.J.D. Piuma, N. Faedo, J.V. Ringwood, G. Mattiazzo, Nonlinear model reduction by moment-matching for a point absorber wave energy conversion system, J. Mar. Sci. Eng. 10 (5) (2022) 656.

[181] E. Amini, H. Mehdipour, E. Faraggiana, D. Golbaz, S. Mozaffari, G. Bracco, M. Neshat, Optimization of hydraulic power take-off system settings for point absorber wave energy converter, Renew. Energy 194 (2022) 938–954.