

Article

Fast Endpoint-Aware Residual Surrogate Modeling for Preliminary Static Screening of Deep-Water Catenary Mooring Lines

Xuhao Liu, Jiang Li, Hujie Shen and Qinhui Liu *

College of Mechanical and Electrical Engineering, Harbin Engineering University, Harbin 150001, China; liuxvhao@hrbeu.edu.cn (X.L.); li_jiang@hrbeu.edu.cn (J.L.)

* Correspondence: liuqinhui@hrbeu.edu.cn

Abstract

Within a single-depth, fixed line property, single-line synthetic quasi-static scope, this study develops an endpoint-prioritized two-stage training framework for full catenary geometry and terminal tension prediction. The global three-dimensional (3D) reference configuration first learns the coupled output and then applies validation-only selection for endpoint-focused refinement. An axisymmetry-informed radial plane representation enhancement is evaluated as the preferred accuracy configuration within the frozen seed-42 scope, yielding geometry root mean square error (RMSE) of 0.360 m, endpoint RMSE of 0.344 m, and tension mean absolute error (MAE) of 1641.8 N. Software-level cross-validation against MoorPy completed 1000 of 1000 requested cases, while a five-seed global-3D ensemble produced an empirical ensemble spread indicator whose geometry spread error Spearman correlation was 0.701. Runtime depended on batch size: central processing unit (CPU) batch-1 inference provided no advantage over the reference solver, whereas CPU batch-1000 throughput was 29.78× faster on the same machine. Expanded screening retained 19 unresolved solver failures. These results support preliminary static screening only. This software-level cross-validation does not constitute experimental, field, or external physical validation. The evidence does not establish dynamic, multiline, or calibrated uncertainty validation.

Keywords: deep-water mooring lines; static catenary prediction; surrogate modeling; residual neural network; endpoint-prioritized fine-tuning; quasi-static analysis; preliminary design screening

Academic Editor: Decheng Wan

Received: 23 June 2026

Revised: 19 July 2026

Accepted: 24 July 2026

Published: 27 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Deep-water floating offshore systems depend on mooring lines for station keeping and for controlling structural response under environmental loading. Catenary systems remain widely used because their restoring force is governed by self-weight, geometric nonlinearity, and seabed contact within mature design frameworks.

Static mooring prediction matters because the equilibrium profile controls suspended shape, touchdown behavior, seabed contact, and terminal tensions. Engineering workflows often require many equilibrium evaluations as fairlead positions and line lengths vary during screening, optimization, uncertainty propagation, or design exploration. Conventional catenary solvers are reliable and physically interpretable, but repeated calls can be ineffi-

cient when many static states must be considered. This motivates engineering surrogates that preserve sufficient physical fidelity while reducing computational cost.

The static analysis used in this study builds on a long line of cable and mooring equilibrium work. Peyrot and Goulois established a cable structure analysis framework that remains relevant for catenary systems, and Masciola et al. implemented a multisegment quasi-static cable model for mooring applications [1,2]. In the *Journal of Marine Science and Engineering* (JMSE), Li et al. formulated quasi-static response prediction through mechanical energy minimization, Touzon et al. compared numerical fidelity levels for wave-energy-converter mooring design, and Lim et al. used time-domain analysis within Bayesian mooring optimization [3–5]. These studies show why static analysis is still useful as an early equilibrium screening layer, even when final designs require dynamic verification.

Recent finite-element studies of offshore wind support structures have examined soil property sensitivity, nonlinear dynamic integrity, and seismic demand under coupled environmental loading, illustrating the broader structural response context in which efficient computational screening is valuable [6–8].

Surrogate modeling has also been introduced to reduce repeated offshore simulation and optimization costs [9]. Christiansen et al. and de Pina et al. used neural network surrogates for mooring line and time domain simulation tasks [10,11]. More recent JMSE studies by Li et al., Nikkhah et al., and Li et al. applied artificial neural network (ANN)- or long short-term memory (LSTM)-based models to mooring tension and floating production, storage, and offloading (FPSO) mooring system prediction [12–14]. Related work by Zhao, Payenda, Xie, Yan, and Ye and their co-workers further developed deep learning models for mooring tension prediction under floating platform or floating wind settings [15–19]. These studies mainly target scalar tensions, time histories, monitoring signals, optimization responses, or system-level characteristics [20–24]. They do not directly address full static catenary geometry together with terminal tensions, which is the output needed when endpoint closure and line shape admissibility must both be checked.

Physics-informed neural networks (PINNs) provide a different route. Raissi et al. introduced PINNs by embedding differential equation residuals in neural network training, and subsequent reviews by Karniadakis and Cai and their co-workers consolidated their use in scientific machine learning and fluid mechanics [25–27]. That formulation is powerful when governing equations and residual terms can be imposed cleanly, but it also requires differentiable residual construction and automatic differentiation. The present study therefore takes a narrower engineering route: physical knowledge enters through solver-generated static data, a structured output that includes geometry and terminal tensions, and selective supervision of the fairlead endpoint.

A remaining gap is the joint prediction of structured full-line geometry and terminal tensions while respecting an engineering-critical endpoint. This nonlinear, regime-dependent task is more demanding than scalar or time-series tension prediction, yet a full residual equation formulation is not required for the targeted screening role.

We therefore retain endpoint-prioritized two-stage training as the method spine. Standard mean squared error (MSE) pretraining learns the global configuration before static endpoint weighting is introduced for targeted refinement; this ordering is curriculum-style staging, but it is not a claim to a new curriculum-learning algorithm. The endpoint weight is chosen by validation-only selection, with no test-set reselection.

The axisymmetry-informed radial plane representation enhancement addresses redundant azimuthal variation in the fixed-property, single-depth, single-line problem. It is a representation change rather than a new training algorithm and is positioned only as the preferred accuracy configuration within the frozen seed-42 scope. The global three-dimensional (3D) reference configuration remains the evidence base for sensitivity, ensemble spread, runtime, and screening analyses.

The contributions of this study are to: (i) formulate structured geometry-plus-tension prediction; (ii) evaluate endpoint-prioritized two-stage training against static weighting and architecture alternatives; (iii) quantify a bounded radial plane seed-42 enhancement; and (iv) assess software-level MoorPy agreement, empirical ensemble spread, batch-dependent runtime, and preliminary screening. All claims remain limited to synthetic quasi-static data at one depth with fixed line properties and one line.

2. Materials and Methods

2.1. Static Catenary Problem and Output Quantities

A catenary mooring line in static equilibrium is governed by the balance between axial tension, submerged weight, and boundary constraints. Let the line centerline be described by the position vector $\mathbf{r}(s) = [x(s), y(s), z(s)]^T$, where s denotes the arc-length coordinate measured from the anchor point to the fairlead. Under static conditions, the internal tension $T(s)$ satisfies the force-balance relation [1,28]

$$\frac{dT(s)}{ds} + \mathbf{q}(s) = 0, \quad (1)$$

where $\mathbf{q}(s)$ is the distributed external load per unit unstretched length, primarily determined by the submerged weight of the line. In the present study, hydrodynamic drag and dynamic inertial effects are neglected, so that the problem is restricted to static geometric equilibrium.

For a fixed water depth and line property set, the equilibrium configuration is determined by the two end conditions and the total line length. In engineering analysis, this configuration is usually obtained with numerical or semi-analytical equilibrium solvers. These methods are reliable and physically interpretable, but repeated evaluations become costly when many static states must be analyzed.

The static catenary problem is parameterized by four inputs: the fairlead coordinates (x_f, y_f, z_f) and the total unstretched line length L . In the simplified single-line setting, these variables define the boundary geometry required by the equilibrium solver. The anchor point is fixed on the seabed, while the fairlead position changes with the prescribed operating condition.

The outputs of interest are the full 3D line configuration and the terminal tensions. We represent the line shape with discrete nodal coordinates sampled along the arc length and include terminal tensions to retain engineering relevance for mooring analysis. The resulting task is a high-dimensional regression problem because the surrogate must predict the full equilibrium geometry rather than a single scalar response.

The fairlead endpoint is particularly important because it represents the prescribed terminal boundary condition. A low average nodal error is therefore insufficient if the predicted line fails to terminate consistently at the fairlead.

Deep-water catenary mooring lines may exhibit qualitatively different equilibrium characteristics under different boundary conditions. In general, the present problem spans three representative regimes: slack, operational, and taut configurations.

In the slack regime, a substantial part of the line may rest on the seabed, depending on the boundary condition. In the operational regime, the line forms a typical suspended catenary with moderate curvature and tension. In the taut regime, the suspended part becomes more stretched, the curvature distribution changes, and terminal tension increases rapidly.

These regime transitions make the mapping from boundary inputs to line shape and tension strongly nonlinear. The surrogate must therefore approximate a high-dimensional output and a regime-dependent response with distinct physical behaviors. This is one reason static catenary prediction remains difficult for purely data-driven learning.

Based on the above physical setting, the surrogate learning task is defined in this study as a nonlinear mapping

$$\mathbf{y} = \mathbf{F}(\mathbf{x}), \mathbf{x} = [x_f, y_f, z_f, L]^T. \quad (2)$$

where $\mathbf{x}$ denotes the input vector of boundary condition variables, and $\mathbf{y}$ denotes the corresponding high-dimensional output vector containing the nodal coordinates of the equilibrium line profile together with the terminal tensions.

The objective is to build a computationally efficient surrogate approximation, defined in this study as

$$\hat{\mathbf{y}} = \mathbf{f}_{\boldsymbol{\varphi}}(\mathbf{x}), \quad (3)$$

where $\boldsymbol{\varphi}$ denotes the trainable parameters of the surrogate model. The desired approximation should satisfy two requirements simultaneously: it should remain computationally efficient for high-throughput prediction, and it should preserve sufficient physical fidelity for engineering use.

Accordingly, the present approach does not embed governing equilibrium equations directly into the network as residual constraints. Instead, it treats the problem as supervised surrogate learning from physics-generated static equilibrium data, with the learning objective designed to reflect the engineering importance of selected outputs. Under this formulation, the methodological question is how to improve boundary-aware prediction while preserving global geometry, terminal tension accuracy, and computational efficiency.

2.2. Dataset Generation and Physical Settings

The surrogate is trained on data from a physics-based static catenary solver. For each sample, the solver computes the equilibrium configuration of a deep-water mooring line under prescribed boundary conditions. The dataset therefore provides physically consistent input and output pairs for supervised learning, without embedding governing equation residuals in the neural network training process.

The in-house data-generation solver is a quasi-analytical static catenary solver with separate suspended line and touchdown line branches. For the fully suspended branch, it first determines the catenary parameter by solving the arc-length constraint with the Brent root-finding algorithm [29], then solves for the horizontal offset required to satisfy the prescribed vertical endpoint separation. For touchdown configurations, it solves a touchdown catenary residual equation, again using Brent root finding, to determine suspended length, grounded length, and nodal geometry. A unified robust solver first attempts the suspended solution; if the profile violates the seabed admissibility tolerance, the touchdown branch is used. Near lift-off cases return the suspended solution when the computed grounded length becomes negligible. The accepted output consists of 41 nodal coordinates, top tension, bottom tension, touchdown status, and grounded length.

We use a representative 800 m water depth with fixed line properties and a fixed anchor location. The setting is informed by recent floating wind mooring design studies, including the 800 m Humboldt Bay reference depth used by Lozon et al. [30], but it is not intended to reproduce a site-specific design. Dataset variation enters only through the fairlead coordinates and the total unstretched line length L .

To make the dataset generation procedure reproducible, the sampling strategy is described explicitly here. For each candidate sample, the total unstretched line length L is sampled uniformly from 850 m to 1200 m, and the fairlead vertical coordinate z_f is sampled uniformly from −20 m to −10 m. The lower and upper bounds of the horizontal sampling interval are defined as

$$R_{\min} = \max(0.1, L - D),$$

$$R_{\max} = 0.95 \sqrt{L^2 - (D - z_f)^2},$$

where $D = 800$ m is the water depth. Candidate samples with $R_{\min} \geq R_{\max}$ are discarded before solver evaluation.

The frozen sampling protocol used a lower horizontal-span bound of 0.1 m and applied a factor of 0.95 to the geometric upper limit. These values are protocol parameters used in constructing the present synthetic dataset and are not proposed as universal engineering constants.

A stratified sampling strategy is used to cover slack-like, operational-like, and taut-like catenary configurations. A normalized horizontal distance parameter p is sampled from regime-dependent intervals: $p \in [0.0, 0.4]$ for Slack, $p \in [0.4, 0.9]$ for Operational, and $p \in [0.9, 1.0]$ for taut configurations.

$$R = R_{\min} + p (R_{\max} - R_{\min}).$$

The intervals $p \in [0, 0.4]$, $[0.4, 0.9]$, and $[0.9, 1.0]$ are protocol-defined sampling strata used to construct slack-like, operational-like, and taut-like subsets; they should not be interpreted as universal physical regime boundaries.

An azimuth angle θ is sampled uniformly from 0 to 2π , and the fairlead horizontal coordinates are obtained as

$$x_f = R \cos \theta, \quad y_f = R \sin \theta.$$

Therefore, x_f and y_f are not sampled independently; they are generated from the sampled horizontal distance and azimuth angle.

An exact replay of the frozen generator requested 10,380 candidate solutions, accepted 10,000, and recorded 380 solver failures, corresponding to a rejection rate of 3.6609%. This audit reproduces the frozen dataset rather than adding samples. Because acceptance depends on numerical solution and endpoint-quality gates, residual rejection bias near difficult boundary states remains possible.

The regime labels are used as sampling strata, not as post hoc solver classifications. Touchdown status is recorded separately and used only for subset analysis. Samples are discarded when the solver does not return a valid solution, produces invalid coordinates, or fails the endpoint quality gate. The target number of accepted samples was prescribed for each stratum; therefore, the final regime counts are not intended to be proportional to the widths of the p intervals. After rejection filtering, sampling was continued within each stratum until the prescribed accepted counts were reached, giving 10,000 valid samples: 3000 slack, 4000 operational, and 3000 taut configurations. Training, validation, and test subsets are then formed by stratified sampling, with an 8:1:1 ratio in each regime. The overall methodology is illustrated in Figure 1.

Line-level static catenary surrogate workflow



Figure 1. Overall framework of the proposed surrogate modeling methodology. Blue denotes data, green surrogate, orange fine-tuning, yellow screening; arrows show progression.

2.3. Input–Output Representation

Each sample is represented by the four-dimensional input vector defined in this study:

$$x = [x_f, y_f, z_f, L]^T,$$

where (x_f, y_f, z_f) denotes the fairlead position and L denotes the total unstretched line length.

The output vector is defined in this study to contain the full 3D nodal geometry of the equilibrium catenary profile and the terminal tensions. The line profile is discretized with 41 nodes along the arc length, with three spatial coordinates at each node. The geometric output therefore has $41 \times 3 = 123$ dimensions. Two scalar quantities are added for the top tension at the fairlead node and the bottom tension at the anchor node, giving a total output dimension of 125. The full output vector is written as

$$y = [x_1, y_1, z_1, \dots, x_{41}, y_{41}, z_{41}, T_{top}, T_{bottom}]^T \in \mathbf{R}^{125}.$$

This formulation defines the learning task as a high-dimensional nonlinear regression problem.

Before training, all input and output variables are normalized with statistics computed only from the training set. The same parameters are then applied to the validation and test sets. This train-only normalization avoids information leakage and improves optimization stability. The same split, normalization procedure, and evaluation scripts are used across all compared models, and the reported supplementary settings identify the training protocol needed to reproduce the fixed-seed comparisons.

The main dataset and representation settings used in the present study are summarized in Tables 1 and 2.

Table 1. Dataset generation and input–output representation settings used in the present study.

Category	Item	Value/Description
Physical setting	Water depth	800 m
Dataset	Total samples	10,000
Dataset split	Train/validation/test	8000/1000/1000
Input	Variables	Fairlead coordinates (x_f, y_f, z_f) and total line length L
Input	Dimension	4
Sampling	Line length L	Uniformly sampled from 850 m to 1200 m
Sampling	Fairlead vertical coordinate z_f	Uniformly sampled from −20 m to −10 m
Sampling	Azimuth angle θ	Uniformly sampled from 0 to 2π
Sampling	Horizontal distance R	$R = R_{\min} + p(R_{\max} - R_{\min})$
Sampling	Slack stratum	$p \in [0, 0.4]$
Sampling	Operational stratum	$p \in [0.4, 0.9]$
Sampling	Taut stratum	$p \in [0.9, 1.0]$
Quality control	Endpoint quality gate	Final-node error < 0.1 m
Output geometry	Nodal coordinates	41 nodes $\times$ 3 coordinates
Output tension	Terminal response	Top tension at the fairlead and bottom tension at the anchor
Output	Dimension	125
Preprocessing	Normalization	Training-set statistics only

Table 2. Fixed physical settings used for static catenary dataset generation.

Parameter	Value	Description
Water depth	800 m	Representative deep water case
Anchor location	(0, 0, −800) m	Fixed seabed origin
Effective submerged line weight	3924 N/m	Constant line weight in water used for solver generation and MoorPy-aligned validation
Axial stiffness, EA	1.8×10^9 N	Constant extensional stiffness of the line

Line type assumption	Extensible chain mooring	Single-material extensible chain mooring with constant EA used across all samples; no specific commercial chain grade or nominal diameter is assumed
Geometry discretization	41 nodes	Nodal representation of the equilibrium line profile
Variable boundary inputs	(x_f, y_f, z_f, L)	Fairlead coordinates and total unstretched line length

The line is therefore treated as an idealized extensible chain mooring represented by its effective submerged weight and axial stiffness, rather than as a specific commercial chain grade or catalogue product.

These physical settings remain fixed for all generated samples. Variation is introduced only through the boundary inputs, namely, the fairlead coordinates (x_f, y_f, z_f) and the total unstretched line length L . This controlled setting isolates how boundary variation affects static equilibrium geometry and terminal tensions.

Axisymmetry-Informed Radial Plane Representation

The fixed-property, single-depth, single-line quasi-static setting is horizontally rotation-symmetric. The axisymmetry-informed radial plane representation enhancement removes redundant azimuthal variation for this bounded problem, but it is not generalized to non-axisymmetric multiline systems.

$$R = \sqrt{x_f^2 + y_f^2}$$

For $R \geq 1 \times 10^{-8}$, define $c_\theta = \frac{x_f}{R}$ and $s_\theta = \frac{y_f}{R}$; otherwise, set $(c_\theta, s_\theta) = (1, 0)$. Each node is represented by $r_i = x_i c_\theta + y_i s_\theta$, and z_i , and reconstructed by $(x_i, y_i, z_i) = (r_i c_\theta, r_i s_\theta, z_i)$.

$$\tilde{\mathbf{y}} = [r_1, z_1, \dots, r_{41}, z_{41}, T_{\text{top}}, T_{\text{bottom}}]^T \in \mathbb{R}^{84}.$$

The canonical boundary input vector is unchanged. The comparison uses the same frozen dataset and split, train-only scaling, a 512×4 ResNet, endpoint-prioritized two-stage training with $\lambda_{\text{end}} = 50$, seed 42, and validation-only selection.

This representation is not a new training algorithm. It remains a single-seed synthetic quasi-static analysis for fixed line properties, one water depth, and one line; no radial-specific multi-seed uncertainty analysis, complete sensitivity sweep, or physical or field evidence is available.

2.4. Network Architecture

Two feedforward neural network architectures are considered: a multilayer perceptron (MLP) baseline and a fully connected residual neural network (ResNet). The MLP has three fully connected hidden layers of width 512 with Tanh activation, followed by a linear output layer that produces the 125-dimensional prediction vector. The main model uses a fully connected ResNet to improve optimization stability for this high-dimensional nonlinear regression problem [31].

Let $\mathbf{h}^{(l)}$ denote the feature vector at layer l . A residual block updates the feature representation according to the residual learning formulation [31] in Equation (4):

$$\mathbf{h}^{(l+1)} = \mathbf{h}^{(l)} + G^{(l)}(\mathbf{h}^{(l)}), \quad (4)$$

where $G^{(l)}(\cdot)$ denotes the nonlinear transformation implemented within the block.

All fully connected layers within the ResNet use a hidden dimension of 512, with Tanh activation throughout. An input projection layer maps the four-dimensional input to the 512-dimensional hidden space. Four residual blocks follow, each containing two linear layers with an additive skip connection. The output layer maps the final hidden rep-

resentation to the 125-dimensional target vector without activation. The network structures are illustrated in Figure 2, and detailed architecture settings are provided in Table S1.

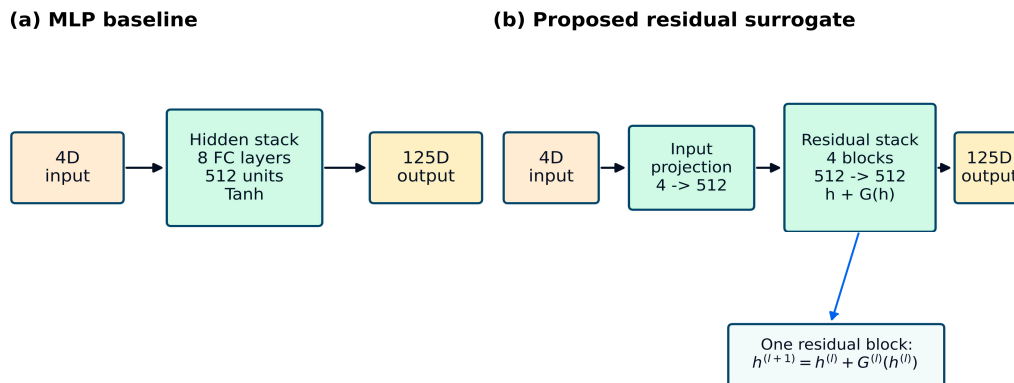


Figure 2. Comparison between the MLP baseline and the proposed ResNet surrogate. Panels (a) and (b) show the MLP and ResNet. Beige marks inputs/outputs, green hidden/residual layers, black arrows data flow, and blue the residual block.

2.5. Loss Functions and Endpoint-Prioritized Training

Baseline supervised loss.

The surrogate is first trained with a standard MSE objective over the standardized output vector. To improve fairlead consistency, the endpoint-prioritized stage modifies this objective by assigning extra weight to the three fairlead endpoint coordinates. The loss definitions are given below.

Let $\hat{y}$ denote the network prediction for the target vector y . The baseline training objective is the mean squared error (MSE),

$$L_{\text{MSE}} = \frac{1}{D} \sum_{i=1}^D (\hat{y}_i - y_i)^2, \quad (5)$$

where $D = 125$ is the output dimension.

Although this loss is simple and effective for generic regression, it treats all output dimensions equally. In the present catenary problem, the engineering importance of different output components is not uniform. The fairlead endpoint has a disproportionately important role in determining whether the predicted line shape is physically acceptable.

Weighted loss formulation.

To include this engineering preference in supervised learning, we define in this study a physics-guided weighted loss,

$$L_w = \frac{1}{D} \sum_{i=1}^D w_i (\hat{y}_i - y_i)^2, \quad (6)$$

where w_i is a prescribed scalar weight assigned to the i -th output component.

All loss functions were evaluated in the standardized output space using statistics fitted only on the training set. The prescribed weights therefore control relative supervision priorities rather than raw dimensional scales. During evaluation, predictions were transformed back to physical units before computing geometric errors in meters and terminal tension errors in Newtons.

Different weighting strategies can be defined by assigning larger values of w_i to selected important outputs. Here, the fairlead endpoint coordinates receive special attention, which gives the proposed endpoint-prioritized supervision strategy. Under this formulation, endpoint-related geometric components are emphasized during training, while all remaining outputs retain unit weight.

The goal is not to impose hard physical constraints in the manner of physics-informed neural networks. Instead, the supervised objective is biased toward quantities that matter most in engineering checks. The method therefore remains a data-driven surrogate trained on physics-generated data, with domain knowledge entering through selective output prioritization.

Endpoint-prioritized supervision.

Among the candidate weighting strategies, endpoint-prioritized supervision is the principal methodological design examined in this work. Let $\mathcal{I}_{\text{end}}$ denote the index set corresponding to the fairlead endpoint coordinates in the output vector. The loss can then be derived from the baseline MSE by partitioning output indices as

$$\mathcal{L}_{\text{end}} = \frac{1}{D} [\sum_{i \notin \mathcal{I}_{\text{end}}} (\hat{y}_i - y_i)^2 + \lambda_{\text{end}} \sum_{i \in \mathcal{I}_{\text{end}}} (\hat{y}_i - y_i)^2], \quad (7)$$

where λ_{end} controls the relative emphasis placed on endpoint accuracy.

This design tests whether one physically important boundary target can be more useful than uniform output weighting or several competing objectives. In the proposed two-stage model, endpoint-prioritized supervision is used only in Stage 2, after standard MSE pretraining has learned a stable global representation. In implementation, the fairlead endpoint coordinates receive extra weight during fine-tuning, while all other outputs keep unit weight. We refer to this procedure as endpoint-prioritized fine-tuning.

Full multitask ablation.

A full multitask variant is used as an ablation baseline to test whether several physically motivated targets improve performance when combined. The comparison also reflects the known difficulty of balancing losses in multitask learning [32]. In this variant, z-coordinate weighting, endpoint weighting, and terminal tension weighting are applied simultaneously through the comparison objective defined in this study as follows:

$$\mathcal{L}_{\text{multi}} = \frac{1}{D} \sum_{i=1}^D w_i^{(\text{multi})} (\hat{y}_i - y_i)^2, \quad (8)$$

where $w_i^{(\text{multi})}$ is the prescribed weight for the output component. The weighting rule is defined in this study as $w_i^{(\text{multi})} i - th$

$$w_i^{(\text{multi})} = \begin{cases} \lambda_{\text{end}}, & i \in \mathcal{I}_{\text{end}}, \\ \lambda_z, & i \in \mathcal{I}_z \setminus \mathcal{I}_{\text{end}}, \\ \lambda_{\text{ten}}, & i \in \mathcal{I}_{\text{ten}}, \\ 1, & \text{otherwise}, \end{cases} \quad (9)$$

The z-weighted and full multitask ablations retain their stated component weights. Endpoint weights $\lambda_{\text{end}} = 0, 1, 10, 25, 50$, and 100 were tested on the frozen seed-42 split. $\lambda_{\text{end}} = 0$ is an endpoint-masked negative control, whereas $\lambda_{\text{end}} = 1$ corresponds to ordinary endpoint weighting. $\lambda_{\text{end}} = 50$ was selected because it minimized validation endpoint root mean square error (RMSE) among positive weights; the held-out test set was not used for selection or reselection. These coefficients are dataset-specific settings, not general constants.

Two-stage endpoint-prioritized training strategy.

Motivation.

Applying strong endpoint weighting from random initialization may overemphasize local boundary accuracy before the network has learned a globally consistent catenary representation. In that case, optimization may converge to solutions with poor overall shape consistency, as shown by the direct endpoint-weighted ablation reported in Section 3. The present study therefore adopts a two-stage strategy in which standard supervised learning first establishes a stable global representation, followed by endpoint-prioritized refinement. This staged training idea is conceptually related to curriculum-style optimization, where easier or more global objectives are introduced before more targeted refinements [33].

Stage 1: standard MSE-based global-geometry pretraining.

In Stage 1, the ResNet is trained using the standard MSE loss over the complete output vector, as defined in Equation (5). All geometric coordinates and terminal tension outputs are treated with unit weight during this stage. This stage establishes a stable global representation of the full catenary configuration before endpoint-prioritized refinement is introduced.

By avoiding output-specific weighting at this stage, the network first learns the coupled 3D geometry and terminal responses without biasing the representation toward a single coordinate direction or response quantity. The best checkpoint from this MSE-based pretraining stage is retained and used to initialize Stage 2.

Stage 2: endpoint-prioritized fine-tuning.

In Stage 2, the best MSE-pretrained checkpoint is used as initialization, and the model is fine-tuned using endpoint-prioritized loss weighting. Specifically, the fairlead endpoint coordinates are assigned a weight of $\lambda_{\text{end}} = 50$, while all non-endpoint outputs retain unit weights. This corresponds to the practical implementation of endpoint-prioritized supervision in the proposed model.

Because the model already approximates the global catenary geometry after Stage 1, endpoint-prioritized fine-tuning acts as a targeted boundary-aware refinement rather than the sole driver of representation learning.

Ablation cases for training sequence and initialization.

Additional ablation cases isolate the effects of training sequence and initialization. Direct endpoint weighting from random initialization tests whether endpoint-focused learning is stable without global pretraining. The z-initialized endpoint-prioritized variant tests whether z-coordinate-weighted pretraining improves or degrades the final endpoint-focused surrogate. These variants are compared with the proposed MSE-pretrained endpoint-prioritized model in Section 3.

The proposed training sequence and the principal ablation pathways are summarized in Figure 3.

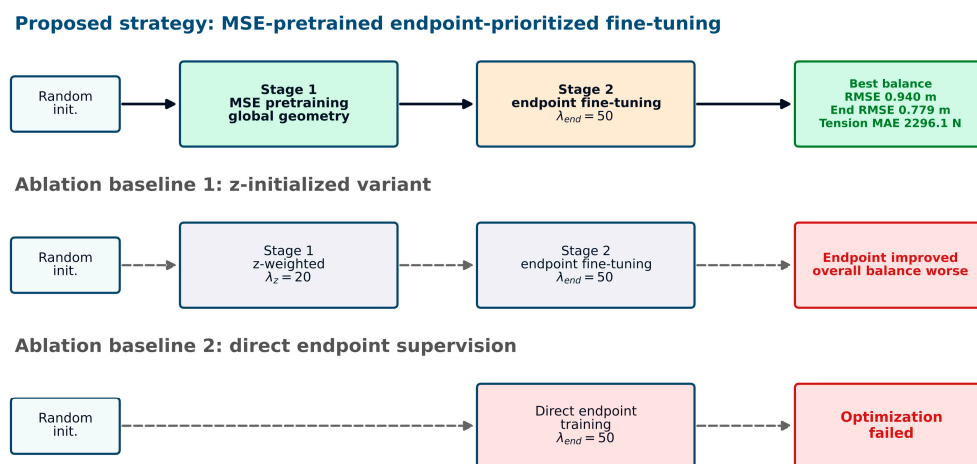


Figure 3. Proposed two-stage endpoint-prioritized training strategy and the main ablation variants. Green and orange mark the proposed stages, gray the z-initialized ablation, and red unfavorable outcomes; solid arrows show the proposed path and dashed arrows the ablations.

2.6. Optimization Settings

All models were trained using mini-batch gradient-based optimization with AdamW [34]. In Stage 1, the ResNet was trained with standard MSE loss, a learning rate of 5×10^{-4} , a batch size of 256, and a maximum of 1000 epochs. In Stage 2, the best MSE-pretrained

model was fine-tuned with endpoint-prioritized loss using a learning rate of 5×10^{-5} for up to 500 epochs. The endpoint weight was $\lambda_{\text{end}} = 50$, while non-endpoint outputs retained unit weights. Detailed architecture and training hyperparameters are provided in Tables S1 and S2. Formal graphics processing unit (GPU) experiments used Python 3.9.25, PyTorch 1.12.0+cu113, and NumPy 1.21.5; the frozen central processing unit (CPU) revision analyses used Python 3.9.12, PyTorch 2.8.0+cpu, NumPy 2.0.2, SciPy 1.13.1, scikit-learn 1.6.1, and MoorPy 1.3.0.

2.7. Evaluation Metrics

Model performance was evaluated exclusively on the held-out test set. During evaluation, test inputs were normalized using the input scaler fitted on the training set, and the network predictions were inverse-transformed to physical units using the corresponding train-set output scaler before metric calculation. All reported geometric errors are therefore expressed in meters (m), and all terminal tension errors are expressed in Newtons (N).

The primary geometric metrics were computed using the 41-node three-dimensional catenary representation. For the n -th test sample, let $\hat{\mathbf{r}}_{n,j}$ and $\mathbf{r}_{n,j}$ denote the predicted and reference position vectors of the j -th node. The nodal Euclidean error is defined as

$$e_{n,j} = \|\hat{\mathbf{r}}_{n,j} - \mathbf{r}_{n,j}\|_2, \quad j = 1, \dots, 41. \quad (10)$$

The global geometric RMSE and mean absolute error (MAE) are defined as

$$\text{RMSE}_{\text{geo}} = \sqrt{\frac{1}{N_{\text{test}} N_{\text{nodes}}} \sum_{n=1}^{N_{\text{test}}} \sum_{j=1}^{N_{\text{nodes}}} e_{n,j}^2}, \quad N_{\text{nodes}} = 41, \quad (11)$$

$$\text{MAE}_{\text{geo}} = \frac{1}{N_{\text{test}} N_{\text{nodes}}} \sum_{n=1}^{N_{\text{test}}} \sum_{j=1}^{N_{\text{nodes}}} e_{n,j}, \quad N_{\text{nodes}} = 41. \quad (12)$$

The maximum nodal error was defined as the maximum value of $e_{n,j}$ across all test samples and nodes, and the p99 error was defined as the 99th percentile of this nodal-error distribution.

Boundary-condition consistency was evaluated using the fairlead endpoint error. Since the 41st node corresponds to the fairlead connection, the endpoint error for sample n is defined as $e_{n,\text{end}} = e_{n,41}$. The endpoint RMSE and subset-wise endpoint maximum error are defined as

$$\text{RMSE}_{\text{end}} = \sqrt{\frac{1}{N_{\text{test}}} \sum_{n=1}^{N_{\text{test}}} e_{n,41}^2}, \quad e_{\text{end,max}}(\mathcal{S}) = \max_{n \in \mathcal{S}} e_{n,\text{end}}, \quad (13)$$

where $\mathcal{S}$ denotes the corresponding regime, touchdown, or non-touchdown subset.

Terminal-tension accuracy was evaluated using the mean absolute error over the two terminal-tension outputs. Let $\hat{T}_{n,k}$ and $T_{n,k}$ denote the predicted and reference terminal tensions for the n -th sample, respectively, where $k \in \{\text{top}, \text{bottom}\}$. The tension MAE is defined as

$$\text{MAE}_{\text{ten}} = \frac{1}{2N_{\text{test}}} \sum_{n=1}^{N_{\text{test}}} \sum_{k=1}^2 |\hat{T}_{n,k} - T_{n,k}|. \quad (14)$$

In addition to aggregate test-set metrics, the same evaluation metrics were computed for slack, operational, and taut regimes, and separately for touchdown and non-touchdown samples. These subset evaluations were used to examine whether prediction accuracy changes across physical response regimes and seabed-contact conditions. Detailed subset metrics are reported in Tables S3 and S4.

2.8. Additional Revision Analyses and Verification Protocols

2.8.1. Generator Rejection Audit

The frozen generator is replayed with identical inputs and acceptance gates; attempts, accepted cases, solver failures, and rejection partitions are recorded without creating a new dataset.

2.8.2. Endpoint-Weight and Architecture-Sensitivity Protocol

Endpoint-weight and architecture configurations are compared on the frozen split. Train-only scaling and validation-only selection are retained, and architecture results are treated as robustness evidence rather than a new main-model selection.

2.8.3. Conventional and Gaussian Process Baselines

Principal component analysis (PCA) with radial basis function support vector regression (RBF-SVR), Gaussian process regression, and a sample-matched ResNet are evaluated with explicitly matched or caveated training-set sizes. No best configuration is declared in this protocol description.

2.8.4. Empirical Ensemble Spread Indicator

Five global-3D checkpoints are combined to relate cross-seed prediction spread to error. The resulting quantity is an empirical ensemble spread indicator, not calibrated uncertainty or an interval estimate.

2.8.5. MoorPy Comparison Protocol

A frozen stratified and boundary-enriched set of cases is reconstructed in MoorPy under aligned line properties. Agreement is assessed at software level. No experimental, field, or external physical validation is inferred.

2.8.6. Runtime Benchmarking

CPU timing is repeated by batch size and separated from GPU timing. Reported speed comparisons are hardware-, software-, thread-, and batch-dependent, with solver and surrogate timing scopes stated explicitly.

2.8.7. Expanded Screening Verification

A fixed candidate set is triaged by the global-3D reference configuration, followed by stratified and boundary-enriched conventional-solver checks. Surrogate screening does not replace solver verification.

3. Results

3.1. Reference-Data Consistency and Generator Rejection Audit

Because the surrogate is trained on solver-generated static catenary solutions rather than direct experimental measurements, the credibility of the reference dataset must be assessed before model comparison. This section examines whether the generated samples are numerically consistent, physically admissible, and reproducible by an independent quasi-static implementation. Two validation levels are used. Internal checks examine tension ranges, regime coverage, touchdown behavior, geometric admissibility, arc-length consistency, endpoint closure, and representative catenary shapes. Software-level cross-validation against MoorPy is then conducted under aligned physical settings. This software-level cross-validation does not constitute experimental, field, or external physical validation. These checks establish the credibility of the reference targets used for supervised learning; surrogate predictive performance is evaluated separately in the model-comparison results.

Internal physical consistency checks.

The exact frozen-dataset replay required 10,380 attempts for 10,000 accepted cases and recorded 380 solver failures (3.6609%). The accepted dataset remains unchanged; the audit also motivates caution that solver rejection may underrepresent difficult boundary states.

The generated reference dataset was first examined through internal physical consistency checks. Across the 10,000 accepted samples, the top tension ranges from 3062.1 kN to 5495.5 kN with a mean value of 3675.8 kN, whereas the bottom tension ranges from 0.0 kN to 2400.7 kN with a mean value of 595.7 kN. No negative-tension artifacts are observed in the retained dataset. The key tension statistics are summarized in Table 3.

The final dataset preserves the prescribed stratified regime distribution, with 3000 slack, 4000 operational, and 3000 taut samples. Touchdown behavior is physically consistent with the intended regime transition: all slack samples show touchdown, most operational samples show touchdown, and only a minority of taut samples show touchdown.

Geometric admissibility and endpoint closure were also verified. The reconstructed polyline length has a mean discrepancy of 0.5274 m and a mean relative discrepancy of 0.051% relative to total line length, mainly due to finite-node discretization of curved catenary profiles. All accepted samples satisfy the 0.1 m endpoint quality gate, and endpoint errors are zero to the reported precision. The regime and touchdown distributions are shown in Figure 4. Full tension distributions, numerical consistency checks, and representative reference profiles are provided in Figures S1–S3.

Table 3. Key tension statistics of the validated reference dataset.

Quantity	Range	Mean
Top tension	3062.1–5495.5 kN	3675.8 kN
Bottom tension	0.0–2400.7 kN	595.7 kN

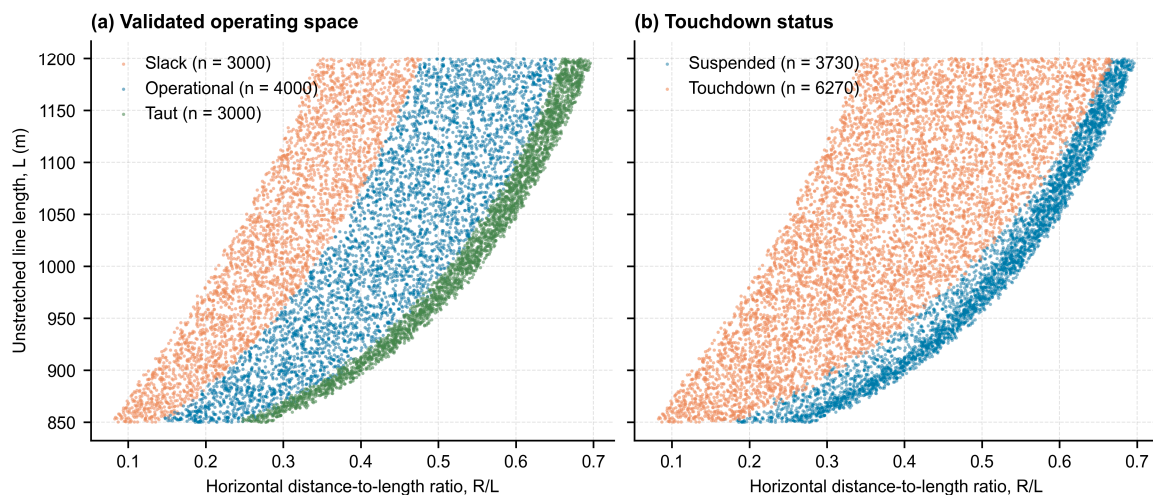


Figure 4. Operating-space and touchdown distributions for the 10,000 validated samples. (a) Regime distribution. (b) Geometric display classification using at least two non-anchor nodes within 0.1 m of the seabed (6270 touchdown; 3730 suspended). Solver metadata records 7195 touchdown cases under a different definition.

3.2. MoorPy Software-Level Cross-Validation

Software-level cross-validation against MoorPy was expanded to 1000 requested cases under aligned physical settings; all 1000 solved successfully, and none failed. The comparison evaluates agreement among independent quasi-static implementations and the surrogate. This software-level cross-validation does not constitute experimental, field, or external physical validation. MoorPy is cited as the independent software implementation [28,35]; related mooring-model validation literature is retained only as background and is not evidence of experimental, field, or external physical validation for this study.

Across the successful cases, the internal-solver/MoorPy geometry RMSE was 0.845 m, and the surrogate/MoorPy geometry RMSE was 1.288 m. The corresponding endpoint discrepancies were 0.000155 m and 0.671 m. These aggregate results support software-level consistency while preserving differences between reference-solver agreement and surrogate prediction error.

The aggregate software-level comparisons are summarized in Table 4.

Table 4. MoorPy aggregate software-level comparison.

Comparison	Geometry RMSE (m)	Endpoint Error (m)	Status
Internal solver/MoorPy	0.845	0.000155	1000/1000 successful
Global-3D surrogate/MoorPy	1.288	0.671	same aligned cases

Regime- and touchdown-resolved comparisons are provided in the Supplementary Materials. The evidence remains bounded to the aligned, single-line quasi-static model and cannot establish physical validation.

3.3. Global-3D Reference Performance and Training Hierarchy

Building on the validated reference dataset, this section assesses whether the surrogate provides useful engineering accuracy for repeated static mooring-line evaluation. Table 5 reports the broad fixed-seed comparison, including architecture baselines, direct weighted-loss variants, two-stage variants, and the proposed model. Table 6 reports the five-seed stability assessment for the main ResNet-based variants.

Under identical MSE supervision, replacing the MLP with a ResNet improves full-line geometry, endpoint accuracy, and tension prediction. The MLP + MSE baseline gives an overall RMSE of 1.829 m and endpoint RMSE of 2.782 m, whereas the ResNet + MSE baseline reduces these values to 1.184 m and 1.692 m, respectively.

Among all fixed-seed variants, the proposed MSE-pretrained endpoint-prioritized model provides the best overall balance, with RMSE = 0.940 m, endpoint RMSE = 0.779 m, and tension MAE = 2296.1 N. The five-seed results show the same performance hierarchy, indicating that the improvement is not an artifact of a single random initialization.

Table 5. Comprehensive fixed-seed performance comparison across all model variants on the test set.

Model	RMSE (m)	MAE (m)	Max (m)	p99 (m)	End RMSE (m)	Tension MAE (N)
MLP + MSE	1.829	1.488	9.918	4.907	2.782	5824.7
MLP + z	3.777	3.068	26.269	9.651	6.068	9909.4
ResNet + MSE	1.184	0.931	10.675	3.328	1.692	2785.9
ResNet + z	2.062	1.630	19.901	5.740	2.595	4765.8
Direct endpoint-weighted variant	9.464	7.756	54.536	26.092	2.737	82,045.4
Z-initialized endpoint-prioritized variant	1.259	0.993	14.244	3.581	0.829	2884.0
Full multitask variant	1.303	1.031	11.685	3.577	1.192	2984.1
Proposed	0.940	0.739	10.606	2.675	0.779	2296.1

For clarity, reference tension magnitudes are reported in kN during reference-data validation, whereas prediction errors are reported in N to preserve resolution for model comparison.

Table 6. Five-seed stability assessment of the main ResNet-based variants.

Model	RMSE (m)	End RMSE (m)	Tension MAE (N)
ResNet + MSE	1.085 ± 0.071	1.401 ± 0.127	2569.8 ± 240.1
Proposed	0.947 ± 0.074	0.904 ± 0.176	2340.8 ± 235.6
Z-initialized endpoint-prioritized variant	1.375 ± 0.093	1.119 ± 0.111	3194.1 ± 308.2
Full multitask variant	1.210 ± 0.086	1.112 ± 0.079	2843.1 ± 355.7

3.4. Axisymmetry-Informed Radial Plane Accuracy Comparison

The radial plane and global-3D configurations use the same frozen dataset and split, seed 42, train-only scaling, and validation-only selection; the test set was not used to reselect either configuration.

The seed-42 global-3D and radial-plane accuracy comparison is reported in Table 7.

Table 7. Global-3D and radial plane seed-42 accuracy comparison.

Metric	Global-3D Reference	Radial Plane
Geometry RMSE (m)	0.940	0.360
Geometry MAE (m)	0.739	0.257
Endpoint RMSE (m)	0.779	0.344
Tension MAE (N)	2296.1	1641.8

Relative to the global-3D reference, the radial plane configuration reduces geometry RMSE by 61.7%, endpoint RMSE by 55.9%, and tension MAE by 28.5% from the unrounded authority values. It is therefore the preferred accuracy configuration within the frozen seed-42 scope.

The global-3D reference configuration remains the evidence base for sensitivity, ensemble spread, runtime, and screening analyses. No radial runtime advantage, calibrated uncertainty, cross-seed robustness, or transfer beyond the single-depth fixed-property synthetic quasi-static problem is inferred.

3.5. Endpoint Weight and Architecture Sensitivity

Direct z-coordinate weighting degrades performance for both architectures. For the MLP, RMSE increases from 1.829 m to 3.777 m; for the ResNet, RMSE increases from 1.184 m to 2.062 m. This result supports treating the catenary response as a coupled 3D equilibrium problem rather than optimized through a single coordinate direction.

The endpoint-prioritized strategy gives the best balance among the tested weighting schemes. Direct endpoint weighting from random initialization performs poorly, which means endpoint emphasis should not dominate before the model has learned the global catenary shape. Once MSE pretraining provides a stable representation, endpoint-prioritized finetuning improves boundary-sensitive prediction without degrading the full geometry.

The endpoint-weight sweep tested λ_{end} values 0, 1, 10, 25, 50, and 100. $\lambda_{\text{end}} = 50$ gave the lowest positive-weight validation endpoint RMSE (0.806 m) and was selected without test-set reselection. Architecture sensitivity across 256-, 512-, and 1024-wide networks and two, four, or six residual blocks shows that the training behavior is not confined to one architecture; these runs are robustness analysis rather than a new main-model selection.

The endpoint-weight and architecture-sensitivity results are summarized in Table 8.

Table 8. Endpoint-weight and architecture sensitivity highlights.

Analysis	Configuration	Validation Evidence	Interpretation
Endpoint weight	$\lambda_{\text{end}} = 50$	endpoint RMSE 0.806 m	validation-only selection
Architecture	1024×4	geometry RMSE 1.075 m	robustness evidence
Architecture	512×2	endpoint RMSE 0.993 m	cost/accuracy evidence

The full multitask variant, which combines z-coordinate, endpoint, and tension weighting, also underperforms the proposed method. Adding more physically meaningful weighted terms therefore does not automatically improve a structured surrogate. This behavior is consistent with multitask learning studies in which competing objectives can lead to suboptimal convergence under simple weighted aggregation [36–38]. Here, selective end-

point prioritization is more effective than broad fixed weight aggregation of partially competing objectives. The overall ablation trends are summarized quantitatively in Table 5.

3.6. Hard Endpoint Clamping Versus Endpoint-Prioritized Fine-Tuning

Hard endpoint clamping was evaluated to distinguish exact endpoint closure from learned boundary-aware representation. In this diagnostic baseline, the predicted fairlead endpoint coordinates were replaced after inference by the prescribed fairlead coordinates, while all other nodal coordinates and terminal tension predictions were left unchanged. No retraining was involved.

The hard endpoint-clamping comparison is reported in Table 9.

Table 9. Hard endpoint-clamping comparison between post-processed endpoint closure and endpoint-prioritized learning.

Model	RMSE (m)	MAE (m)	Max (m)	p99 (m)	End RMSE (m)	Tension MAE (N)
ResNet + MSE	1.184	0.931	10.675	3.328	1.692	2785.9
ResNet + MSE + endpoint clamping	1.154	0.895	10.675	3.289	0.000	2785.9
Proposed endpoint-prioritized	0.940	0.739	10.606	2.675	0.779	2296.1
Proposed + endpoint clamping	0.932	0.722	10.606	2.674	0.000	2296.1

3.7. Regime-Wise and Touchdown-Dependent Performance

The proposed model remains accurate across slack, operational, and taut regimes, as well as across touchdown and non-touchdown configurations. On the held-out test set, the RMSE and tension MAE are 1.126 m and 1887.8 N for slack samples ($n = 300$), 0.850 m and 2027.3 N for operational samples ($n = 400$), and 0.847 m and 3062.6 N for taut samples ($n = 300$). For touchdown and non-touchdown subsets, the corresponding RMSE values are 0.975 m and 0.841 m, with tension MAE values of 2012.7 N and 3035.5 N, respectively. Detailed subset metrics are provided in Tables S3 and S4.

The slack regime is the most challenging for geometry reconstruction because it involves stronger seabed interaction and local transitions near touchdown or lift-off. In contrast, the taut regime shows the largest tension MAE, which points to mechanically stretched configurations being more sensitive to small geometric deviations. A similar geometry-tension trade-off is observed between touchdown and non-touchdown subsets: touchdown cases are more difficult in terms of geometry, whereas non-touchdown cases show larger tension errors.

Poor-case analysis shows that top-tension relative MAE is 0.0631%, while 14 bottom-tension predictions are negative only near a zero reference value. Clipping those predictions to zero is reported as a diagnostic and is not applied to the primary metrics. The hardest geometric cases cluster near regime and touchdown boundaries, so error distributions and worst cases remain necessary alongside aggregate metrics.

The held-out error distributions and the ten largest per-case geometry RMSE values are shown in Figure 5.

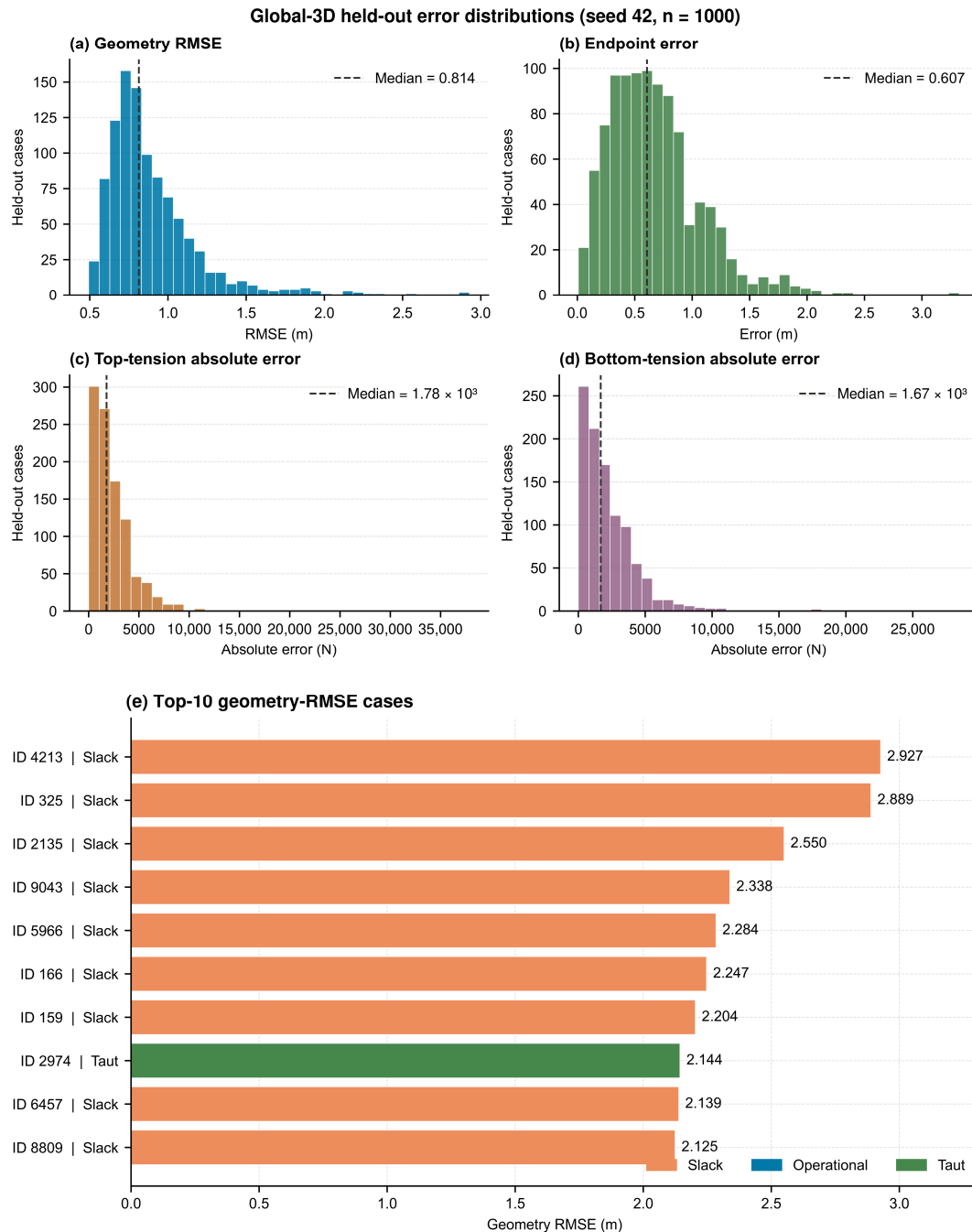


Figure 5. Error distributions and the ten largest per-case geometry RMSE values for the frozen global-3D seed-42 test set ($n = 1000$). The upper block shows geometry, endpoint, and terminal-tension error distributions; the lower block ranks the ten worst geometry cases. Per-case geometry RMSE is distinct from the global maximum nodal error.

3.8. Representative Line-Shape Predictions

Representative stored line-shape comparisons are shown for slack, operational, and taut configurations. The historical node-level source bundle and original case-selection record were unavailable; therefore, no claim is made that the original selection procedure was reproduced. Figure 6 shows the stored slack, operational, and taut profiles.

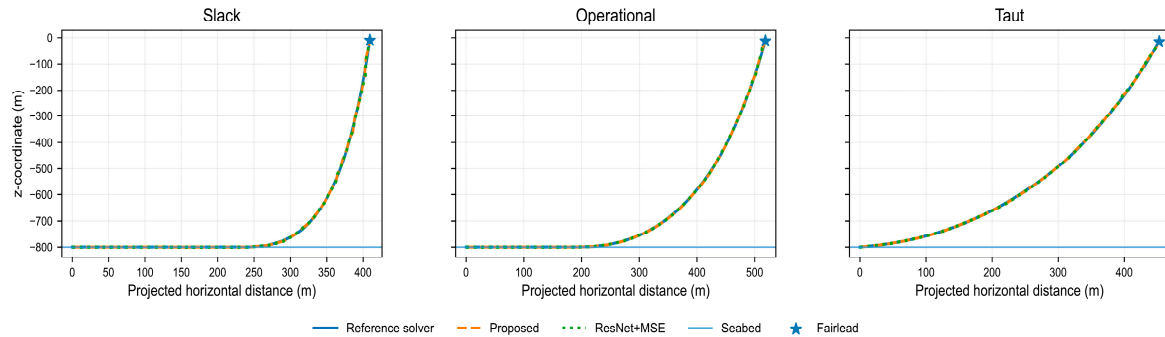


Figure 6. Representative stored line-shape comparisons on the held-out test set for slack, operational, and taut configurations.

3.9. Classical and Gaussian-Process Surrogate Baselines

Classical baselines were extended beyond the original PCA regressors. PCA RBF-SVR (PCA-50, $C = 100$, $\gamma = 0.1$, $\epsilon = 0.01$) produced test geometry RMSE of 10.094 m, endpoint RMSE of 2.032 m, and tension MAE of 50,173.0 N. The Gaussian process regression (GPR) regressors used 2000 matched training IDs and achieved test geometry RMSE of 0.820 m, endpoint RMSE of 0.263 m, and tension MAE of 6224.2 N; the ResNet trained on the same 2000 IDs achieved 2.477 m, 2.300 m, and 7421.6 N, respectively. However, the PCA basis used by GPR was fitted on all 8000 frozen training outputs, so this was not a fully self-contained 2000-sample pipeline. The full-data global-3D ResNet is background only, not an equal-size comparator. CPU/GPU fit, inference, and memory measurements are descriptive rather than a direct hardware-matched resource comparison. Table 10 summarizes these baselines.

Table 10. Classical, Gaussian-process, and matched neural baselines.

Model	Training n	Geometry RMSE (m)	Endpoint RMSE (m)	Tension MAE (N)
PCA RBF-SVR	8000	10.094	2.032	50,173.0
GPR	2000	0.820	0.263	6224.2
Matched ResNet	2000	2.477	2.300	7421.6
Global-3D reference	full data	0.940	0.779	2296.1

3.10. Empirical Ensemble Spread Indicator and Poor-Case Analysis

Five global-3D checkpoints with seeds 13, 21, 42, 84, and 168 were averaged. The ensemble geometry RMSE was 0.588 m, endpoint RMSE was 0.405 m, and tension MAE was 1628.2 N. Cross-seed spread correlated with geometry error (Spearman's $\rho = 0.701$), but correlations were weak for endpoint error (0.177) and tension error (0.242).

Deferring the highest-spread geometry cases reduced accepted-set risk from 0.410 m at full coverage to 0.361 m at 80% coverage. This supports geometry risk ranking only; it is an empirical ensemble spread indicator, not calibrated uncertainty, a confidence interval, or a prediction interval.

The ensemble aggregate metrics and spread–error correlations are summarized in Table 11.

Table 11. Global-3D ensemble aggregate and spread error correlations.

Quantity	Value
Ensemble geometry RMSE (m)	0.588
Ensemble endpoint RMSE (m)	0.405
Ensemble tension MAE (N)	1628.2

Geometry Spearman's ρ	0.701
Endpoint Spearman's ρ	0.177
Tension Spearman's ρ	0.242

3.11. Batch-Dependent Computational Efficiency

Runtime was evaluated separately for repeated CPU batches and single-sample GPU execution. On the audited CPU, batch-1 end-to-end inference required 0.9596 ms, compared with 0.7715 ms for the repeated reference solver, so no single-query speed advantage was observed. At batch 1000, the surrogate required 25.909 ms in total (0.02591 ms per sample), corresponding to a same-machine throughput speed-up of 29.78 $\times$. These values are hardware-, software-, thread-, and batch-dependent.

The CPU measurements are summarized in Table 12, and the RTX 2060 batch-1 measurements are reported separately in Table 13.

Table 12. CPU runtime by batch scope.

Scope	Total (ms)	Per Sample (ms)	Relative to Solver
Reference solver repeated median	0.7715	0.7715	1.00
Surrogate batch 1 end-to-end	0.9596	0.9596	no advantage
Surrogate batch 1000 end-to-end	25.909	0.02591	29.78 $\times$ throughput

Table 13. RTX 2060 single-sample runtime.

Scope	Time (ms)	Boundary
Pure inference	1.0247	batch 1
End-to-end	1.1582	batch 1; separately reported

3.12. Expanded Preliminary Static Screening Verification

Expanded preliminary static screening evaluated 10,000 candidates with the global-3D reference configuration. The surrogate classified 6099 as feasible. A stratified and boundary-enriched sequence attempted 59 conventional-solver checks: 40 resolved and 19 remained unresolved solver failures.

Among the 40 resolved checks, the confusion matrix comprised true positives (TP = 20), true negatives (TN = 12), false positives (FP = 0), and false negatives (FN = 8). The checks are not a random prevalence sample, the criteria are illustrative rather than code requirements, and surrogate triage does not replace conventional solver or dynamic verification.

The screening counts are summarized in Table 14, and the corresponding candidate map is shown in Figure 7.

Table 14. Expanded screening flow and resolved confusion matrix.

Quantity	Count
Candidates	10,000
Surrogate-feasible	6099
Solver attempts	59
Resolved/unresolved	40/19
TP/TN/FP/FN	20/12/0/8

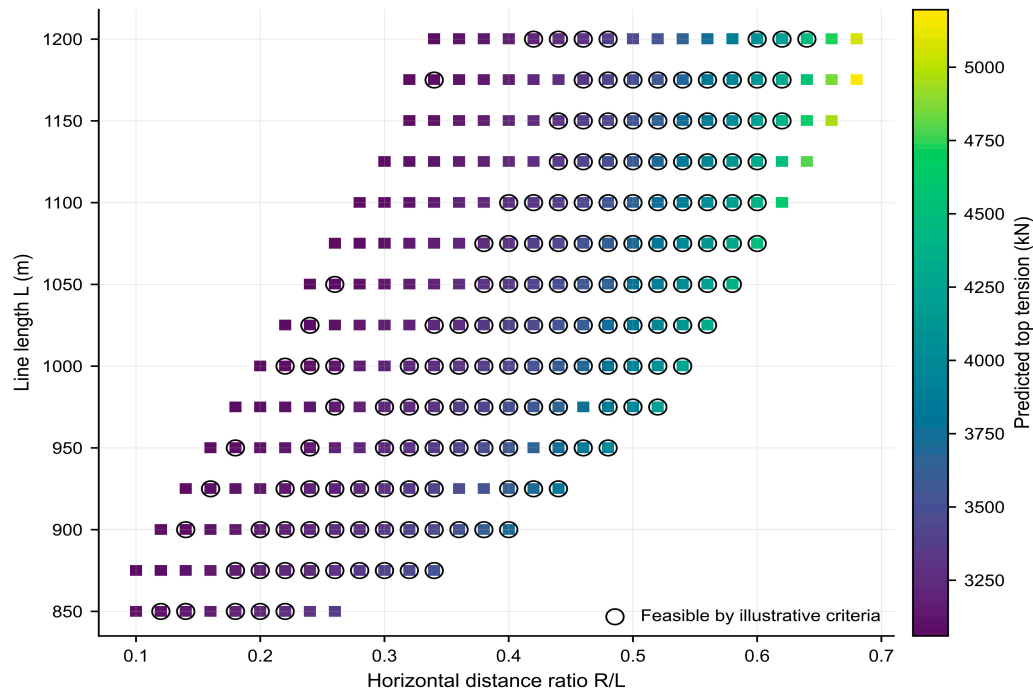


Figure 7. Preliminary static design-screening map predicted by the proposed surrogate. The color denotes predicted top tension, and the outlined markers indicate candidates satisfying the illustrative feasibility criteria.

4. Discussion

4.1. Training Hierarchy and Representation Roles

Endpoint-prioritized two-stage training is the method spine: global MSE pretraining establishes a coupled representation before endpoint emphasis is introduced. The lambda and architecture analyses show why endpoint weighting must be interpreted together with training order and validation-only selection, rather than as a universal static coefficient.

4.2. Radial Accuracy and Global-3D Evidence Boundaries

The axisymmetry-informed radial-plane representation enhancement substantially improves seed-42 accuracy by removing redundant azimuthal variation in the bounded single-line problem. This does not transfer the global-3D evidence for sensitivity, ensemble spread, runtime, or screening to the radial configuration, nor does it establish cross-seed robustness.

4.3. Verification, Spread, and Poor-Case Interpretation

Software-level cross-validation against MoorPy supports implementation consistency. This software-level cross-validation does not constitute experimental, field, or external physical validation. The empirical ensemble spread indicator is informative for geometry risk ranking, while weak endpoint and tension correlations preclude a calibrated uncertainty interpretation. Near-zero negative tension predictions and boundary-associated poor cases further motivate solver checks.

The GPR regressors used 2000 matched training IDs, but the PCA basis was fitted on all 8000 frozen training outputs, so this was not a fully self-contained 2000-sample pipeline. The full-data global-3D ResNet is background only, and CPU/GPU fit, inference, and memory measurements are descriptive rather than a direct hardware-matched resource comparison.

4.4. Batch-Dependent Screening Utility

The practical runtime benefit is a batched-throughput result, not a universal single-query acceleration: CPU batch 1 showed no advantage, whereas batch 1000 supported faster repeated evaluation on the same machine. The screening demonstration is therefore a preliminary triage workflow whose unresolved cases and false negatives require conventional verification.

4.5. Scope Limitations and Next Steps

The evidence is synthetic and quasi-static, with one water depth, fixed line properties, one line, and no vessel dynamics, hydrodynamic forcing, multiline coupling, fatigue, experimental observations, or field data. Future work should separate training and test ranges across depths and properties, assess radial representations across seeds, calibrate uncertainty only with an appropriate protocol, and validate final designs with established solvers and physical evidence.

5. Conclusions

1. Endpoint-prioritized two-stage training provides the methodological contribution: global MSE pretraining precedes boundary-focused refinement, and selection remains validation-only.
2. The axisymmetry-informed radial plane representation is the preferred accuracy configuration within the frozen seed-42 scope, with 0.360 m geometry RMSE, 0.344 m endpoint RMSE, and 1641.8 N tension MAE; global-3D retains the broader evidence assignments.
3. Software-level cross-validation against MoorPy solved 1000 of 1000 requested cases and supports implementation agreement. It does not constitute experimental, field, or external physical validation.
4. Runtime benefit is batch-dependent: CPU batch 1 showed no advantage, while CPU batch 1000 delivered a 29.78× same-machine throughput speed-up.
5. Expanded screening produced TP = 20, TN = 12, FP = 0, and FN = 8 among 40 resolved checks, with 19 unresolved solver failures; preliminary surrogate triage therefore requires conventional verification.
6. All findings are limited to single-depth, fixed line property, single-line synthetic quasi-static data. Dynamic, multiline, cross-property, calibrated uncertainty, and physical validation remain future work.

Supplementary Materials: The following supporting information can be downloaded at: <https://www.mdpi.com/article/10.3390/jmse14151377/s1>, Figure S1: Distribution of top and bottom tensions across the validated catenary dataset; Figure S2: Numerical consistency checks for the validated dataset; Figure S3: Representative static catenary profiles from the validated dataset; Table S1: Network architecture settings of the surrogate model; Table S2: Two-stage training hyperparameters used in the present study; Table S3: MoorPy aggregate software-level comparison; Table S4: Regime-wise performance; Table S5: Touchdown/non-touchdown performance; Table S6: Poor-case and near-zero tension diagnostic; Table S7: Performance of PCA-based traditional surrogate baselines on the held-out test set; Table S8: Representative candidates selected from the preliminary static design-screening map and checked using the reference static solver; Table S9: Frozen generator rejection audit; Table S10: Complete endpoint-weight sweep (seed 42); Table S11: Complete architecture sensitivity (seed 42); Table S12: Radial transform and seed-42 aggregate metrics; Table S13: PCA RBF-SVR, GPR, and matched-ResNet details; Table S14: Global-3D ensemble spread-error correlations; Table S15: Geometry risk coverage from the empirical ensemble-spread indicator; Table S16: CPU runtime details; Table S17: RTX 2060 runtime details; Table S18: Expanded screening criteria and flow; Table S19: Screening confusion matrix and unresolved failures.

Author Contributions: Conceptualization, X.L. and Q.L.; methodology, X.L., J.L. and H.S.; software, X.L.; validation, X.L., J.L. and H.S.; formal analysis, X.L.; investigation, X.L.; resources, Q.L.; data curation, X.L.; writing—original draft preparation, X.L.; writing—review and editing, X.L., J.L., H.S. and Q.L.; visualization, X.L.; supervision, J.L., H.S. and Q.L.; project administration, Q.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The scripts, reference results, trained models, and generated datasets supporting this study are available from the corresponding author upon reasonable request.

Acknowledgments: The authors thank the developers of MoorPy for providing an open-source quasi-static mooring analysis tool used for software-level cross-validation. During the preparation of this manuscript, artificial intelligence (AI)-assisted tools were used only for English-language editing, wording refinement, and formatting checks. The authors reviewed, edited, and approved all scientific content, analyses, interpretations, and conclusions, and take full responsibility for the content of the publication.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Peyrot, A.H.; Goulois, A.M. Analysis of cable structures. *Comput. Struct.* **1979**, *10*, 805–813. [https://doi.org/10.1016/0045-7949\(79\)90044-0](https://doi.org/10.1016/0045-7949(79)90044-0).
2. Masciola, M.; Jonkman, J.; Robertson, A. Extending the capabilities of the Mooring Analysis Program: A survey of dynamic mooring line theories for integration into FAST. In Proceedings of the International Conference on Offshore Mechanics and Arctic Engineering, San Francisco, CA, USA, 8–13 June 2014; Volume 9A: Ocean Renewable Energy, V09AT09A032. <https://doi.org/10.1115/OMAE2014-23508>.
3. Li, C.B.; Chen, M.; Choung, J. The quasi-static response of moored floating structures based on minimization of mechanical energy. *J. Mar. Sci. Eng.* **2021**, *9*, 960. <https://doi.org/10.3390/jmse9090960>.
4. Touzon, I.; Nava, V.; de Miguel, B.; Petuya, V. A comparison of numerical approaches for the design of mooring systems for wave energy converters. *J. Mar. Sci. Eng.* **2020**, *8*, 523. <https://doi.org/10.3390/jmse8070523>.
5. Lim, J.; Choi, M.; Lee, S. A Bayesian optimization algorithm for the optimization of mooring system design using time-domain analysis. *J. Mar. Sci. Eng.* **2023**, *11*, 507. <https://doi.org/10.3390/jmse11030507>.
6. Shahzad, M.M.; Khan, M.H.F.; Lee, D. Parametric sensitivity analysis of soil properties for retrofitted offshore wind turbine jacket substructures under combined environmental loading. *Ocean Eng.* **2025**, *341*, 122690. <https://doi.org/10.1016/j.oceaneng.2025.122690>.
7. Shahzad, M.M.; Khan, M.H.F.; Lee, D. Analyzing the structural integrity of retrofitted jacket substructures for an offshore wind turbine under nonlinear dynamic loading. *Appl. Ocean Res.* **2025**, *165*, 104862. <https://doi.org/10.1016/j.apor.2025.104862>.
8. Shahzad, M.M.; Wang, X.; Khan, M.H.F.; Lee, D. Seismic demand of jacket and monopile offshore wind turbine support structures under combined loading: Effect of ground motion significant duration. *Ocean Eng.* **2026**, *363*, 126746. <https://doi.org/10.1016/j.oceaneng.2026.126746>.
9. Forrester, A.I.J.; Keane, A.J. Recent advances in surrogate-based optimization. *Prog. Aerosp. Sci.* **2009**, *45*, 50–79. <https://doi.org/10.1016/j.paerosci.2008.11.001>.
10. Christiansen, N.H.; Voie, P.E.T.; Winther, O.; Høgsberg, J. Optimization of neural networks for time-domain simulation of mooring lines. *Proc. Inst. Mech. Eng. Part M J. Eng. Marit. Environ.* **2016**, *230*, 434–443. <https://doi.org/10.1177/1475090215573090>.
11. de Pina, A.C.; de Pina, A.A.; Albrecht, C.H.; de Lima, B.S.L.P.; Jacob, B.P. ANN-based surrogate models for the analysis of mooring lines and risers. *Appl. Ocean Res.* **2013**, *41*, 76–86. <https://doi.org/10.1016/j.apor.2013.03.003>.
12. Li, P.; Jin, C.; Ma, G.; Yang, J.; Sun, L. Evaluation of dynamic tensions of single point mooring system under random waves with artificial neural network. *J. Mar. Sci. Eng.* **2022**, *10*, 666. <https://doi.org/10.3390/jmse10050666>.
13. Nikkhah, E.; Fernandes, A.C.; Caprace, J.-D. Design of experiments technique applied to artificial neural network models for FPSO mooring system analysis. *J. Mar. Sci. Eng.* **2023**, *11*, 2194. <https://doi.org/10.3390/jmse1112194>.

14. Li, Y.; Zhong, Q.; Zhang, J.; Wang, X. Prediction of mooring line top-tensions incorporated with identification of lost clump weights. *J. Mar. Sci. Eng.* **2025**, *13*, 631. <https://doi.org/10.3390/jmse13040631>.
15. Zhao, Y.; Dong, S.; Jiang, F.; Incecik, A. Mooring tension prediction based on BP neural network for semi-submersible platform. *Ocean Eng.* **2021**, *223*, 108714. <https://doi.org/10.1016/j.oceaneng.2021.108714>.
16. Payenda, M.A.; Wang, S.; Jiang, Z.; Prinz, A. Prediction of mooring dynamics for a semi-submersible floating wind turbine with recurrent neural network models. *Ocean Eng.* **2024**, *313*, 119490. <https://doi.org/10.1016/j.oceaneng.2024.119490>.
17. Xie, Y.; Tang, H. A deep learning approach based on domain generalization for mooring tension prediction in floating structures. *Ocean Eng.* **2024**, *306*, 118128. <https://doi.org/10.1016/j.oceaneng.2024.118128>.
18. Yan, C.; Shi, W.; Jiang, Z.; Han, X.; Li, X.; Soares, C.G. A domain-adaptive deep neural network for real-time mooring tension prediction in floating wind turbines. *Ocean Eng.* **2026**, *343*, 123259. <https://doi.org/10.1016/j.oceaneng.2025.123259>.
19. Ye, M.; Cao, K.; Wang, J.; Wan, D. Mooring tension prediction of floating offshore wind turbines using end-to-end convolutional neural network models with attention mechanisms. *Ocean Eng.* **2026**, *346*, 123939. <https://doi.org/10.1016/j.oceaneng.2025.123939>.
20. Walker, J.; Coraddu, A.; Collu, M.; Oneto, L. Digital twins of the mooring line tension for floating offshore wind turbines to improve monitoring, lifespan, and safety. *J. Ocean Eng. Mar. Energy* **2022**, *8*, 1–16. <https://doi.org/10.1007/s40722-021-00213-y>.
21. Karim, F.R.B.; Mishra, I.; Rotea, M.A.; Griffith, D.T.; Zhang, J. Wavelet-aided learning for condition monitoring of floating offshore wind turbine mooring systems. *Ocean Eng.* **2026**, *347*, 123907. <https://doi.org/10.1016/j.oceaneng.2025.123907>.
22. Festa, O.; Charles, J.; Gourvenec, S. A flexible neural network-based surrogate model for optimisation of floating wind moorings with LRDs. *Ocean Eng.* **2025**, *332*, 119767. <https://doi.org/10.1016/j.oceaneng.2024.119767>.
23. Ye, X.; Zheng, P.; Qiao, D.; Zhao, X.; Zhou, Y.; Wang, L. Multi-objective optimization design of a mooring system based on the surrogate model. *J. Mar. Sci. Eng.* **2024**, *12*, 1853. <https://doi.org/10.3390/jmse12101853>.
24. Seo, J.; Park, D.-W. Prediction of mooring system characteristics of the floating barge using deep neural networks. *J. Mar. Sci. Technol.* **2024**, *32*, 4. <https://doi.org/10.51400/2709-6998.2745>.
25. Raissi, M.; Perdikaris, P.; Karniadakis, G.E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.* **2019**, *378*, 686–707. <https://doi.org/10.1016/j.jcp.2018.10.045>.
26. Karniadakis, G.E.; Kevrekidis, I.G.; Lu, L.; Perdikaris, P.; Wang, S.; Yang, L. Physics-informed machine learning. *Nat. Rev. Phys.* **2021**, *3*, 422–440. <https://doi.org/10.1038/s42254-021-00314-5>.
27. Cai, S.; Mao, Z.; Wang, Z.; Yin, M.; Karniadakis, G.E. Physics-informed neural networks (PINNs) for fluid mechanics: A review. *Acta Mech. Sin.* **2021**, *37*, 1727–1738. <https://doi.org/10.1007/s10409-021-01148-1>.
28. Hall, M. Generalized quasi-static mooring system modeling with analytic Jacobians. *Energies* **2024**, *17*, 3155. <https://doi.org/10.3390/en17133155>.
29. Brent, R.P. *Algorithms for Minimization Without Derivatives*; Prentice-Hall: Englewood Cliffs, NJ, USA, 1973.
30. Lozon, E.; Lekkala, M.R.; Sirkis, L.; Hall, M. Reference mooring and dynamic cable designs for representative U.S. floating wind farms. *Ocean Eng.* **2025**, *322*, 120473. <https://doi.org/10.1016/j.oceaneng.2025.120473>.
31. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. <https://doi.org/10.1109/CVPR.2016.90>.
32. Kendall, A.; Gal, Y.; Cipolla, R. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7482–7491. <https://doi.org/10.1109/CVPR.2018.00781>.
33. Bengio, Y.; Louradour, J.; Collobert, R.; Weston, J. Curriculum learning. In Proceedings of the 26th International Conference on Machine Learning, Montreal, QC, Canada, 14–18 June 2009; pp. 41–48. <https://doi.org/10.1145/1553374.1553380>.
34. Loshchilov, I.; Hutter, F. Decoupled weight decay regularization. In Proceedings of the 7th International Conference on Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
35. Hall, M.; Goupee, A. Validation of a lumped-mass mooring line model with DeepCwind semisubmersible model test data. *Ocean Eng.* **2015**, *104*, 590–603. <https://doi.org/10.1016/j.oceaneng.2015.05.035>.
36. Rudner, S. An overview of multi-task learning in deep neural networks. *arXiv* **2017**, arXiv:1706.05098.

37. Chen, Z.; Badrinarayanan, V.; Lee, C.-Y.; Rabinovich, A. GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In Proceedings of the 35th International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; Proceedings of Machine Learning Research, Volume 80, pp. 794–803.
38. Yu, T.; Kumar, S.; Gupta, A.; Levine, S.; Hausman, K.; Finn, C. Gradient surgery for multi-task learning. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 5824–5836.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.