



Learning structural damage fields from time-domain wave propagation data with Bayesian neural networks

Bernardo Feijó Junqueira¹ · Gabriel L. S. Silva² · Rafael de O. Teloli³ · Ricardo Leiderman⁴ · Daniel Alves Castello²

Received: 23 October 2025 / Accepted: 1 June 2026 / Published online: 2 September 2026
© The Author(s) 2026

Abstract

This study investigates the application of Bayesian Neural Networks for regression tasks aimed at recovering structural damage parameters from time-domain wave propagation data. The raw wave propagation data are simulated in a one-dimensional elastic bar using the Spectral Finite Element Method, considering data collection from a limited number of sensors. The damage fields are modeled as rectangular profiles, described by three parameters: position, size (or width), and magnitude. The inverse problem of damage recovery is formulated within a Bayesian framework, where the wave-field data collected by the sensors serve as inputs, while the outputs consist of probabilistic distributions of the damage parameters, allowing the quantification of uncertainty in the damage parameters and providing a stochastic characterization of the damage state. The training data include deterministic values for system properties, while the validation and test datasets incorporate uncertainties in the elastic bar's properties, thereby simulating modeling errors during the inference stage of the inverse model. The main contribution of this work is to demonstrate the effectiveness of Bayesian Neural Networks as surrogate inverse models for damage characterization, addressing challenges related to model training and generalization. The results highlight the potential of this approach for reliable damage assessment in Structural Health Monitoring applications under uncertainty.

Keywords Structural health monitoring · Deep learning · Bayesian neural network · 1D wave propagation · Inverse problem · Surrogate model

1 Introduction

Structural Health Monitoring (SHM) is an essential process for maintenance, anomaly detection, and improving the serviceability of structures through the management and repair of damage, which is often manifested by changes

in mechanical properties, such as stiffness. Recently, with the fast advances of Artificial Intelligence (AI), and more specifically machine learning (ML) and deep learning (DL) techniques, automation of SHM has become prominent in the engineering field [1–3]. Neural Network (NN) models have contributed to extensive analysis of datasets

Technical Editor: Geng Wang.

✉ Bernardo Feijó Junqueira
bernardo.junqueira@eng.uerj.br

Gabriel L. S. Silva
gabriellucassousadasilva@gmail.com

Rafael de O. Teloli
rafael.teloli@femto-st.fr

Ricardo Leiderman
leider@ic.uff.br

Daniel Alves Castello
castello@mecanica.coppe.uff.br

¹ Department of Mechanical Engineering, Rio de Janeiro State University (UERJ), Fonseca Teles Street, 121, Rio de Janeiro 20940-200, RJ, Brazil

² Department of Mechanical Engineering, Federal University of Rio de Janeiro (UFRJ), Rio de Janeiro 21941-594, RJ, Brazil

³ SUPMICROTECH, CNRS, institut FEMTO-ST, Université Marie et Louis Pasteur, 25000 Besançon, France

⁴ Computer Science Department, Fluminense Federal University (UFF), Passo da Pátria Street, 156, Niterói 24210-240, RJ, Brazil

and intricate system behaviors through automated feature extraction. However, there are still ongoing challenges, such as the lack of interpretability regarding uncertainties in the network's output and robustness to structural changes [4], both of which are crucial for the decision-making process in an effective SHM system.

The Bayesian modeling framework can be a valuable tool for investigating uncertainty in DL models. In this context, a Bayesian Neural Network (BNN) is an NN that takes into account uncertainty in one or more of its weights by treating them as random variables rather than deterministic entities. Therefore, a BNN provides predictive uncertainty estimates in the form of full output distributions rather than single-point predictions, enabling the quantification of confidence intervals and uncertainty bounds for inferred damage parameters, thereby enhancing the interpretability and reliability of the surrogate model's response and contributing to the reduction of false positives and false negatives in practical SHM applications [5–7]. BNNs also learn effectively from small and noisy datasets and are more robust to overfitting than traditional NNs. This makes them particularly suitable for applications such as SHM, where uncertainty estimates can build a cost-informed decision-making process [8]. In this regard, several studies in the literature have explored using BNNs to support SHM procedures [7–13], with most of them, to the best of the authors' knowledge, focusing on damage detection, condition classification or damage evolution (severity). The exception is the work by [12], which not only detects damage but also locates it in Carbon Fiber Reinforced Polymer (CFRP) composites using ultrasonic Lamb waves, although not fully characterizing the damage profile.

The main objective of the present work is to propose the use of BNNs as a surrogate inverse model to fully characterize damage profiles using measured wave propagation data. Additionally, this study addresses challenges related to BNN model training and generalization, particularly in SHM regression tasks. Unlike classification problems, where posterior predictive distributions can be accurately approximated despite imperfect sampling, regression tasks pose greater difficulty for Markov Chain Monte Carlo (MCMC) methods to converge when sampling from the posterior distribution of network parameters, due to the continuous nature of regression outputs [14]. The measured data were synthetically generated using Spectral Finite Element modeling of one-dimensional wave propagation in an elastic bar, with data collected from a limited number of sensors distributed along the bar's length. The damage profiles are modeled as rectangular shapes, fully characterized by their center position, size (or width), and magnitude. To assess the robustness of the proposed approach, the surrogate inverse model was also tested with profiles generated

from a model that includes property uncertainties, as well as with Gaussian damage profiles (which were not used during the BNN training procedure).

The remainder of this work is organized as follows: In the *Mathematical Modeling* section, the mathematical formulation of the forward and inverse problems is presented. In the *Numerical Experiments: Simulated Guided Wave Propagation Data* section, the datasets used to train, validate, and test the proposed inverse surrogate are described. The *BNN-Based Damage Characterization* section presents and discusses the results for different damage profiles. The *Discussion* section analyzes and discusses the advantages and limitations of the proposed model. Finally, the last section draws the conclusions.

2 Mathematical modeling

This section presents the proposed mathematical models for the forward and inverse problems, which are designed in conjunction to recover damage profiles from time-domain wave propagation data collected in a one-dimensional elastic medium.

2.1 Forward model - 1D wave propagation in an elastic bar

Under the geometric linearity assumption (infinitesimal extension strains and small rotations), the equations of motion for a one-dimensional continuous bar in axial extension can be derived from Hamilton's principle within the context of linear elastodynamics:

$$\frac{\partial}{\partial x} \left(E(x)A(x) \frac{\partial u(x, t)}{\partial x} \right) = \rho A(x) \frac{\partial^2 u(x, t)}{\partial t^2}, \quad (1)$$

where $u(x, t)$ denotes the longitudinal displacement, $E(x)$ is the Young modulus (local), ρ is the material density, and $A(x)$ is the bar cross-section area. For non-dispersive waves, Eq. (1) describes the wave propagation phenomena of bars with linear isotropic property and negligible lateral inertia effects [15]. The one-dimensional bar used to generate samples for training has material and geometric properties defined in Table 1. The boundary conditions for the problem in Eq. (1) are: $u(x = 0) = 0$, $(EA \partial_x u)|_{x=L} = f(t)$, where $f(t)$ is a 2-cycle Hanning-windowed tone burst, with central frequency of 25 kHz

The space-discretized equations of motion are obtained using the spectral finite element method, as detailed in [16]. Equation (1) is discretized using $N_e = 120$ non-overlapping spectral elements of order $N_O = 6$, where the order is chosen following the rule of thumb from [17], resulting in a

Table 1 Physical system properties and their associated uncertainties

Parameters	Nominal values	Considered uncertainties
L	1 [m]	0.1%
A	3.6×10^{-5} [m ²]	1.0%
E	200×10^9 [Pa]	2.0%
ρ	7556 [kg/m ³]	2.0%
F_0	1 [N]	—

system of $ngdl = 720$ degrees of freedom. The resulting system of linear ordinary differential equations (ODEs) is as follows:

$$M\ddot{u} + Ku = f(t), \quad (2)$$

where $u(t) \in \mathbb{R}^n$ is the displacement vector of the discrete governing equations, M and K are the global mass and stiffness matrices of the discretized system, which carry information regarding inertia and stiffness of the proposed system, and f is the global force vector. On the element level, M_{ij}^e , K_{ij}^e and f_i^e are computed as follows:

$$K_{ij}^e = \int_{\Omega_e^h} \rho A \psi_i \psi_j dx, \quad (3)$$

$$K_{ij}^e = \int_{\Omega_e^h} EA \psi_i' \psi_j' dx, \quad (4)$$

$$f_i^e = (f \psi_i) \Big|_{\Omega_e^h \cap x=L}. \quad (5)$$

where Ω_e^h is the element domain of integration and $\psi(x)$ are the Lagrange Polynomials defined on the internal nodes of the element; for the SFEM, these nodes are the points of the Gauss-Lobatto-Legendre quadrature [17], which are the same used for integration.

The ODEs in Eq. (2) are then integrated with the Composite Explicit Method proposed by [18], with a time step $\delta t = 5 \times 10^{-8}$ s over a time span of $T = 2 \times 10^{-3}$ s.

The direct problem consists of determining the system's response given a damage field $d(x|\alpha)$. In this study, damage is modeled as a scalar-valued 1D field $d(x, \alpha) : x \in [0, L] \mapsto [0, 1]$, stationary in time, corresponding to a local reduction in stiffness; damage is parameterized by $\alpha = (x_c, dl, D)$, as given by Eq. (6):

$$d(x, \alpha) = \begin{cases} D, & |x - x_c| \leq dl/2 \\ 0, & \text{otherwise} \end{cases}, \quad (6)$$

where x_c is the location of the damage's center, dl is its width and D represents the local reduction in the Young's Modulus of the bar, that is to say $E(x) = (1 - d(x))E_0$.

This simple model is commonly used in the literature, as can be seen in [19–25]. The properties of the intact material are provided in Table 1, along with the uncertainties associated with each parameter.

The uncertainty values for the properties, also presented in Table 1, indicate that each parameter can vary from its nominal value by an amount of $\text{nominal value} \times \text{uncertainty} \times (1 + \gamma)$ for each generated sample of the model. Here, F_0 represents the amplitude of the applied excitation, while $\gamma \sim \mathcal{N}(0, 1)$ represents a sample from an univariate Gaussian distribution with mean $\mu = 0$ and standard deviation $\sigma = 1$. In other words L, A, E and ρ are random variables such that $L \sim \mathcal{N}^+(L, 0.1\%L)$, $A \sim \mathcal{N}^+(A, 1.0\%A)$, $E \sim \mathcal{N}^+(E, 2.0\%E)$ and $\rho \sim \mathcal{N}^+(\rho, 2.0\%\rho)$; where $\mathcal{N}^+(\mu, \sigma)$ is shorthand for $\mathbb{I}_{[0, +\infty]} \times \mathcal{N}(\mu, \sigma)$, with $\mathbb{I}_{[0, +\infty]}$ being the unitary support over the real positive line.

2.2 Inverse model - Bayesian neural network

The procedure used in this work to recover damage profiles from time-domain wave propagation data is based on a BNN [5, 6]. As BNNs are stochastic artificial networks trained with Bayesian inference, they naturally allow for quantifying uncertainties associated to deep neural network predictions. The basic characteristics of BNNs are briefly presented next.

Let us assume a set of data $\mathcal{D}$ is associated to a set of network parameters $\theta = (W, b)$, where W and b are the weights and biases. In a Bayesian framework variables and parameters are modeled as random variables whose characteristics are described by means of PDFs (probability density functions). In this sense, as data $\mathcal{D}$ and model parameters are modeled as random variables, one may relate them by the Bayes rule as shown next:

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta) p_0(\theta)}{p(\mathcal{D})}, \quad (7)$$

where $p(\theta|\mathcal{D})$ is the posterior PDF of model parameters given a set of data $\mathcal{D}$; $p(\mathcal{D}|\theta)$ is the likelihood function, which quantifies how well the model explains the observed data $\mathcal{D}$ given specific set of model parameters θ ; and the denominator $p(\mathcal{D})$ is the evidence or marginal likelihood, acting as a normalizing constant. As for $p_0(\theta)$, it is the prior distribution, encoding assumptions about model parameters before observing the data $\mathcal{D}$. Briefly speaking, Eq. 7 provides the full stochastic model to infer about model parameters θ given data $\mathcal{D}$. In practice, instead of modeling all the layers as stochastic, one usually chooses only a few layers as stochastic, usually placed at the end of the networks. As for the BNN design used in this work, it is described next.

The first step in the process is to define the BNN architecture, depicted in Fig. 1. The dropout rate (fixed at 5%) and the activation functions – Rectified Linear Unit (ReLU) for all hidden standard layers (dense and convolutional) and no activation for the Bayesian layer – were predetermined and held constant during hyperparameter tuning. Hyperparameter optimization was performed via grid search, which involved fine-tuning several hyperparameters of the deterministic layers, such as layer types, the number of layers, and the number of units or filters in each layer. Regarding the Bayesian layers, it is important to note that they are prone to overparameterization due to the added complexity of modeling distributions over weights. In particular, experiments with architectures containing multiple Bayesian layers demonstrated that propagating uncertainty through successive probabilistic layers creates a cascading effect, which substantially increases the complexity of the loss landscape [14, 26, 27]. This added complexity makes optimization significantly more challenging, often rendering the loss function excessively difficult to be minimized. Consequently, the grid search was constrained to architectures with a single Bayesian layer, positioned as the final hidden layer (refer to Fig. 1) and limited to a maximum of 16 units. This configuration was selected to reduce model complexity and improve training stability while still capturing the probabilistic nature of the problem. Moreover, even with this constrained design, the Bayesian layer contributes to improved generalization by enabling the model to quantify uncertainty in its predictions, a particularly valuable

property in settings with limited or noisy data [28]. The network architecture presented in Fig. 1 employs the *DenseReparameterization* layer as the layer immediately preceding the outputs, from *TensorFlow Probability*¹. During training, this layer samples weights using the reparameterization strategy, which expresses each trainable parameter sample as $\theta_i = \mu_i + \sigma_i \cdot \epsilon$, where $\epsilon \sim \mathcal{N}(0, 1)$. This formulation allows stochastic sampling to be expressed as a differentiable operation, enabling efficient gradient-based optimization while propagating uncertainty through the model. In other words, this layer incorporates trainable parameters μ_i and σ_i for both weights and biases, while also allowing for the specification of a prior distribution $p_0(\theta_i) \sim \mathcal{N}(0, 1)$, while the posterior is modeled using a mean-field normal distribution. The divergence function, used for regularization during training, is defined with the Kullback-Leibler (KL) divergence, which measures the difference between the prior and the posterior distributions [29]:

$$\text{KL}(p(\Theta | \mathcal{D}) || p_0(\Theta)) \cong \frac{1}{N} \sum_{j=1}^N \left[\log \frac{p(\Theta^{(j)} | \mathcal{D})}{p_0(\Theta^{(j)})} \right] \quad (8)$$

where $\Theta^{(j)}$ are random draws from the posterior.

The final output of the network is generated through the *IndependentNormal* layer. This layer models the output distribution, where the event shape corresponds to the number of independent normal variables. The distribution

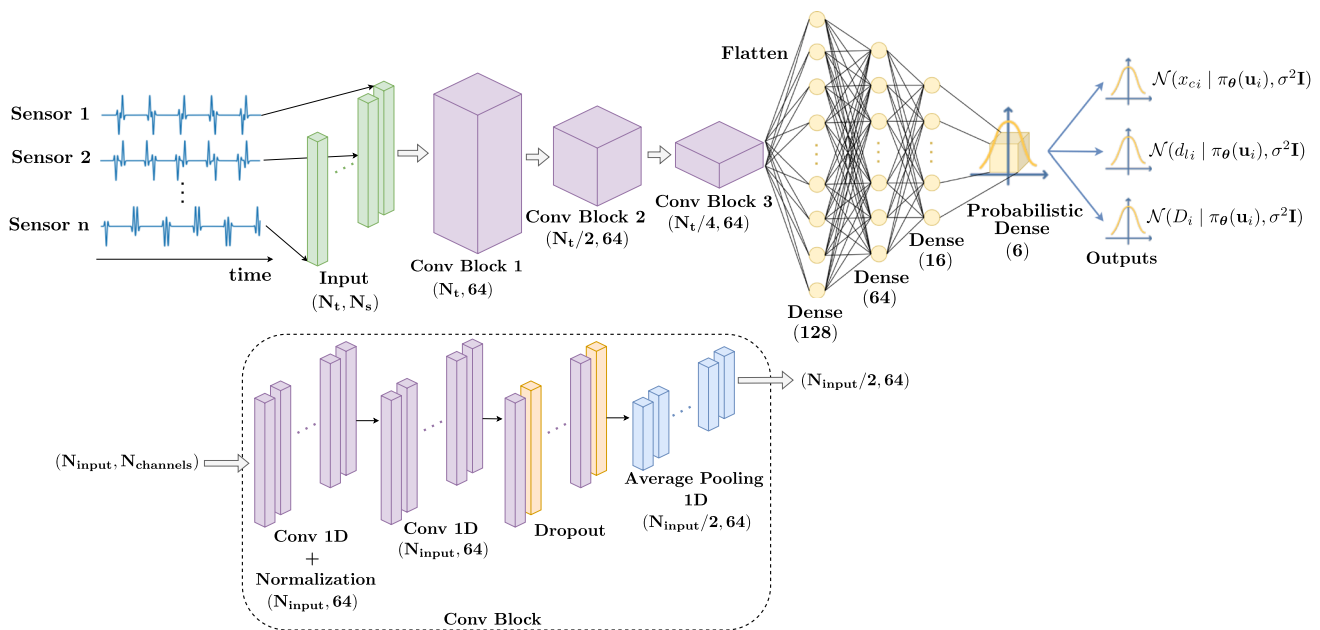


Fig. 1 Proposed architecture of the BNN. Here, N_s denotes the number of sensors used to measure the time-domain response, while N_t represents the number of time points, given the duration of the measurement

¹ Version: 0.26.0-dev20250103.

is then sampled, converting it into a tensor for downstream processing.

Regarding the loss function $\mathcal{L}(\theta)$, the Negative Log-Likelihood [30] is used in conjunction with the KL divergence as a regularization term [5]:

$$\mathcal{L}(\theta) = - \sum_{i=1}^n \log p(\alpha_i | u_i, \theta) + \text{KL}(p(\Theta | \mathcal{D}) || p_0(\Theta)), \quad (9)$$

where, in the context of the approached problem, α_i are samples of the damage profile parameters (or the outputs of the model), u_i denotes the samples of observed time-domain displacement data collected from the sensors (or the inputs of the model) and $p(\alpha_i | u_i, \theta)$ is the likelihood of the damage profile parameters α_i given an input sample u_i and the model parameters θ . In this work, the likelihood is assumed to follow a Gaussian distribution with fixed variance, which corresponds to the common assumption of Gaussian observation noise. Accordingly, the likelihood can be expressed analytically as:

$$p(\alpha_i | u_i, \theta) = \mathcal{N}(\alpha_i | \pi_{\theta}(u_i), \sigma^2 \mathbf{I}), \quad (10)$$

where $\pi_{\theta}(u_i)$ denotes the model prediction (i.e., the mean of the Gaussian), and $\sigma^2 \mathbf{I}$ is the assumed covariance, representing homoscedastic Gaussian noise.

As for the optimization algorithm used during the back-propagation procedure of the proposed model, two were considered: Adaptive Moment Estimation (ADAM) and Stochastic Gradient Descent (SGD). Experiments were conducted with both algorithms, and their performance in minimizing the cost function will be discussed in the BNN-Based Damage Characterization section. For both optimizers, the following standard hyperparameter values were used: for ADAM, the exponential decay rate for the first moment estimates was set to $\beta_1 = 0.9$, and for the second moment estimates to $\beta_2 = 0.999$; for SGD, the momentum was set to 0.0 (i.e., no momentum). For ADAM, a fixed learning rate of 0.005 was used. In contrast, for SGD the learning rate followed an exponential decay schedule, starting at 0.01 with a decay rate of 0.96 every 10,000 steps (i.e., minibatch updates), which, given the dataset size, corresponds to approximately one decay every 10 epochs. The learning rate decay strategy was not applied to ADAM, as its internal adaptive mechanism already adjusts the learning rate during training, making external scheduling redundant. In contrast, the decay schedule was beneficial for SGD, which lacks such adaptivity and therefore can benefit from a gradual reduction in the learning rate to improve convergence stability.

Note that, when using a BNN for prediction, the output is typically obtained using a Markov Chain Monte Carlo

(MCMC) approach. In this method, a large set of parameters θ_i is sampled from the posterior distribution and used to compute a series of possible outputs α_i . These outputs correspond to samples from the marginal distribution $p(\alpha | u, \theta)$, where u corresponds to the current measurement of the system [5].

In the context of the proposed problem, given a set of system responses u , a sufficient number of samples are drawn to estimate the underlying distribution of the parameters α , which describe the damage field (see the *Forward Model - 1D Wave Propagation in an Elastic Bar* section). The measured responses, obtained from N_s sensors uniformly spaced along the length of the bar. Furthermore, during the inference sampling procedure, samples that violate the physical domain of the problem are discarded. Specifically, for a damage sample $\alpha = (x_c, d\ell, D)$ to be considered valid, the support of the damage field must lie entirely within the spatial domain $x \in [0, L]$. This imposes the following constraint:

$$x_c - \frac{d\ell}{2} \geq 0 \quad \text{and} \quad x_c + \frac{d\ell}{2} \leq L. \quad (11)$$

Samples that do not satisfy this condition are discarded from the posterior estimate.

3 Numerical experiments: simulated guided wave propagation data

Synthetic data were generated using the forward model described in the *Forward Model - 1D Wave Propagation in an Elastic Bar* section, with both the excitation signal and the damage profile provided as inputs. The resulting displacement field was recorded by eight sensors uniformly distributed along the length of the bar, spanning from $x = 0.1\text{m}$ to $x = 1.0\text{m}$, on the free end where the guided wave is applied, as illustrated in Fig. 2. Measurements were collected over a 2 ms time window, with a sampling frequency of $f_s = 400\text{kHz}$.

In the simulations, sensor readings incorporate a signal-to-noise ratio (SNR) of 20 dB. As detailed in *Inverse Model - Bayesian Neural Network* section, these time-domain sensor measurements capture the system response and serve as inputs to the inverse model, illustrated in Fig. 1. Therefore, the inputs and outputs of the BNN have shapes of (N, N_t, N_s) and (N, N_{α}) , respectively, respectively, where N , N_t , N_s , N_{α} represent the number of samples, time points, sensors, and damage parameters.

Narrowband signals are recognized as the preferred method for exciting guided wave signals in non-destructive testing applications [31, 32]. Thus, for this study, a 2-cycle

Fig. 2 Schematic representation of the simulated experiment, showing the distribution of the eight sensors along the bar

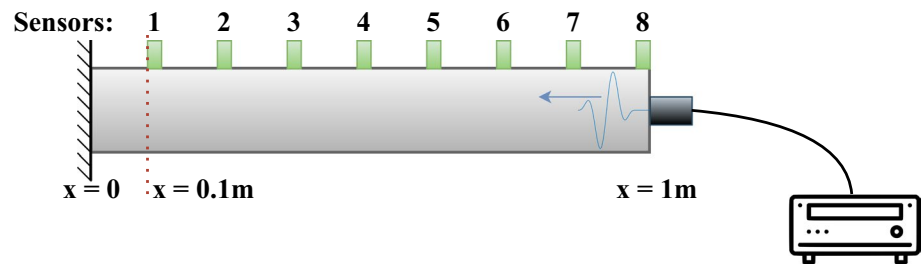


Fig. 3 Two-cycle Hanning-windowed pulse. **a** Time domain signal $F(t)$. **b** Absolute value of the Fourier transform of the signal $F(t)$

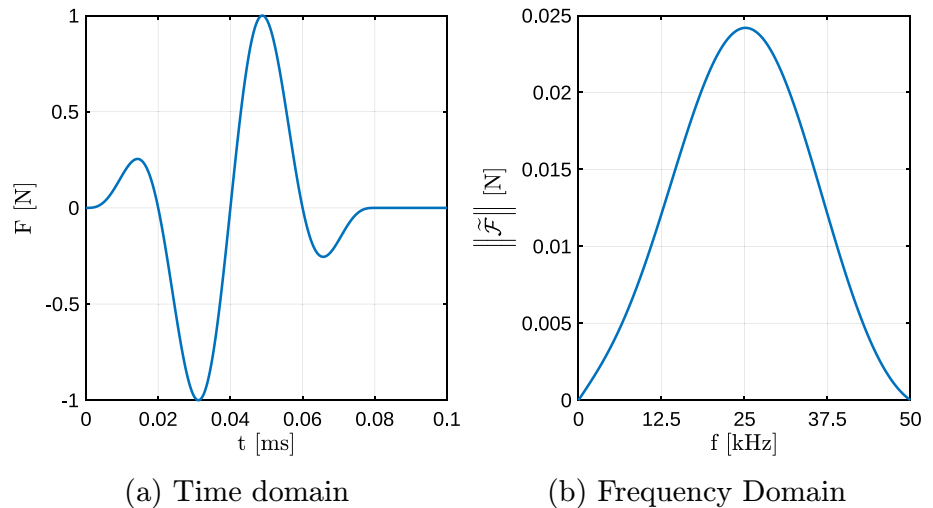


Table 2 Dataset division for training and evaluating the BNN model

Dataset	Damage profile	Properties uncertainties?	Number of samples
Training	Rectangular	No	20000
Validation/test	Rectangular	Yes	5000
Test	Gaussian	No	1000

Hanning-windowed tone burst with a central frequency $f_c = 25$ kHz is employed as the excitation signal. The waveform is defined by

$$f(t) = 0.5 \left(1 - \cos \left(\frac{2\pi f_c t}{n} \right) \right) \sin(2\pi f_c t), \quad 0 < t < 2f_c, \quad n = 2. \quad (12)$$

This excitation lasts for 0.1 ms and is sampled at a rate of 4 MHz to generate the signal. The resulting time-domain waveform and its frequency spectrum are shown in Fig. 3.

This excitation signal, depicted in Fig. 3, is used as input to the forward model to solve the direct problem for three distinct scenarios. Each scenario corresponds to a different dataset configuration, summarized in Table 2.

As can be seen in Table 2, a total of 26,000 synthetic samples were generated through computational simulations. For training, 20,000 samples were created using rectangular damage profiles. Each profile is randomly generated with uniform sampling and fully defined by three parameters: position (center), size (width), and magnitude, representing the percentage reduction in bar stiffness within the damaged

region. An additional 5,000 samples with rectangular profiles are used for validation and testing, incorporating uncertainties in the bar properties (see Table 1). This asymmetric splitting of the training, validation, and test datasets was adopted to assess the robustness of the proposed inverse surrogate when trained under idealized reference conditions and subsequently evaluated on data affected by realistic uncertainties in the physical properties. In a related but more challenging setting, training with data that include physical uncertainties was adopted for damage assessment in plate-like structures [33], where it contributed to improving the DL-based model's performance. Furthermore, 1,000 Gaussian-shaped damage profiles are generated for testing only. These profiles are parameterized as

$$d(x, \alpha) = D \exp \left(-\frac{(x - x_c)^2}{2\sigma^2} \right), \quad (13)$$

where $\sigma = d_\ell/6$ and sampled in the same way as the rectangular ones, enabling evaluation of the model's generalization capability to damage shapes not seen during training. Figure 4 illustrates the considered damage profiles, and the parameter ranges are summarized in Table 3. The range of the position parameter x_c is chosen to ensure that all damage profiles lie within the structure. The upper bounds for width and intensity are arbitrary, as larger values would only make identification easier, while the lower bounds for d_ℓ

Fig. 4 Graphical representation of parameterized damage profiles: **a** rectangular and **b** Gaussian

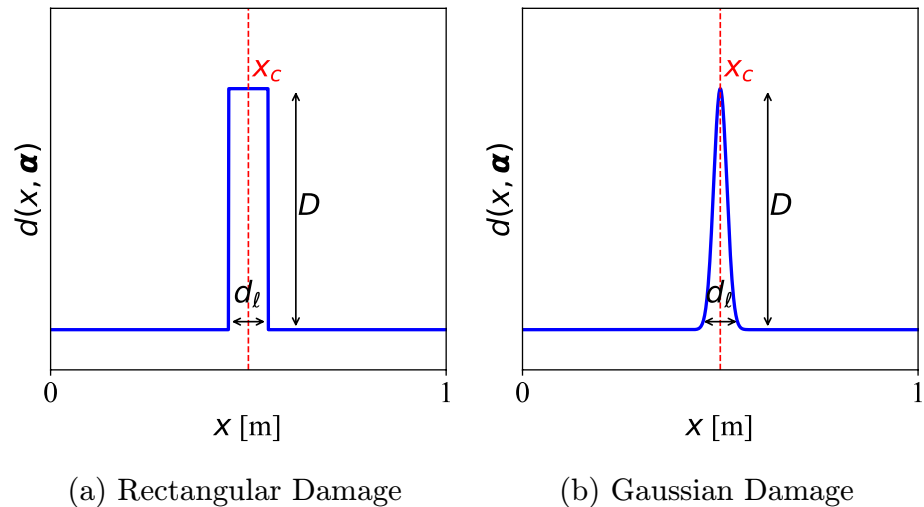


Table 3 Damage profiles parameters and their limits

Parameter	Notation	Minimum	Maximum
Position [m]	x_c	0.01	0.99
Size [m]	d_ℓ	0.05	0.15
Magnitude	D	0.1	0.5

and D correspond to the smallest values that remain distinguishable from noise given the 20, dB signal-to-noise ratio.

The dataset size was empirically chosen to ensure convergence and generalization of the inverse model. Although generating such a dataset experimentally may be impractical due to time and resource limitations, it is feasible in simulation-based workflows, where large datasets can be generated offline. In real-world applications, this approach supports a trade-off: by leveraging simulated data, it becomes possible to reduce the number of physical sensors while still enabling damage characterization through data-driven models.

Furthermore, additional care is taken to ensure that the damage does not exceed the physical spatial domain. Specifically, the extent of the damage, considering both its position and size, is constrained to stay within the bar's length. For instance, if a damage profile is located at 0.01 m on the bar and has a size of 0.1 m, it will not extend beyond the left end of the bar (at $x = 0$ m). In such cases, the portion of the damage that exceeds the physical domain is disregarded, and a parameterization correction is applied to adjust its position and size accordingly.

4 BNN-based damage characterization

In this section, the results are presented and discussed regarding the performance of the proposed BNN model in recovering damage field profiles, along with challenges associated with its training. First, note that the only preprocessing step applied to the raw data was normalizing both

the input and output data to the range $[0, 1]$ via min-max scaling.

Training a BNN is a non-trivial task. Compared to traditional NNs, BNNs are more prone to convergence issues due to their highly non-convex, multimodal posterior distribution [5]. This makes the optimization process sensitive to initialization and stochasticity during training. To mitigate these issues, multiple training runs with different random seeds should be performed, followed by careful seed selection to ensure reproducibility and generalization.

To illustrate this, 100 training experiments were conducted using both the Adam and Mini-batch SGD optimizers. Each run employed a different random seed to control the data splits, dropout masks, and weight/bias initializations. All models were trained for 100 epochs with a batch size of 32. For weights and biases in the standard layers, He initialization (also known as Kaiming initialization) was applied, which is designed for activation functions like ReLU. It samples weights from a normal distribution with variance inversely proportional to the number of input neurons n_{in} [34]:

$$W \sim \mathcal{N}\left(0, \frac{2}{n_{in}}\right). \quad (14)$$

This approach compensates for the variance reduction introduced by the ReLU function, helping to maintain a consistent signal magnitude across layers, which aids both gradient flow and convergence.

For Bayesian layers such as the ReparameterizedDense used in this work, which do not use an activation function, Xavier (Glorot) initialization was applied to the mean of the weight distributions. Specifically, weights were sampled from a uniform distribution with support limits proportional to the inverse square root of the sum of the number of input n_{in} and output neurons n_{out} [35]:

$$W \sim \mathcal{U}\left(-\sqrt{\frac{6}{n_{\text{in}} + n_{\text{out}}}}, \sqrt{\frac{6}{n_{\text{in}} + n_{\text{out}}}}\right). \quad (15)$$

This initialization is better suited for linear layers as it maintains stable activation variances. The uncertainty-related parameters (e.g., variances or log-variances) were initialized with small values to reflect prior assumptions and promote stable convergence.

Figure 5 presents the training curves averaged over the 100 runs for each optimizer configuration, with shaded regions representing the 95% confidence intervals.

In 91% of the experiments, the Adam optimizer failed to reach a satisfactory minimum, with the mean relative error (MRE) across the predicted damage parameters in the validation dataset ranging from approximately 78% to 164%. These runs are referred to as “bad seeds” due to their poor convergence behavior (see Fig. 5a). In contrast, Fig. 5b illustrates a subset of runs that successfully converged, with the MRE dropping to values between 5% and 14%, demonstrating that Adam can perform well under favorable initializations. Meanwhile, the SGD optimizer (Fig. 5c) showed consistent convergence across all runs, with final loss values stabilizing around -3 and MRE ranging from approximately 13% to 22%. Although convergence was slower compared to the best Adam runs, SGD proved to be significantly more robust to initialization. [36] explain this behavior, stating that SGD has more generalization power than Adam in DL. Specifically, (1) the opportunity to escape a minimum for

both SGD and Adam depends positively on the “size” of the basin (the region around the minimum) and negatively on the gradient noise; (2) for the same basin, SGD tends to escape faster than Adam. This is mainly because: (a) Adam adapts the gradient using a scaling factor, which makes the basin “larger” and smoother, whereas (b) Adam’s averaging technique results in lighter gradient noise compared to SGD. Therefore, SGD is more unstable in sharp minima (defined as minima with small “size” of their basins), but can escape to flatter minima (which often generalize better) more easily. Furthermore, additional training runs were performed with Adam using reduced sensor configurations, namely 4 sensors (sensors 1, 3, 5, and 7; see Fig. 2) and 1 sensor (sensor 8; see Fig. 2). For the 4-sensor configuration, the model converged in 73% of the runs, whereas for the single-sensor configuration it converged in 100% of the runs, representing an improvement in convergence. However, this increased stability comes at the cost of degraded final predictive performance, as well as reduced generalization to other field shapes, such as Gaussian profiles.

A summary of the average and standard deviation of the MRE, as well as absolute error metrics (MAE and RMSE), for the predicted damage parameters under different training regimes is provided in Table 4. Absolute errors are reported in physical units, computed from de-normalized predictions and respecting the admissible parameter bounds. MAE is reported together with the standard deviation of absolute

Fig. 5 Training curves for the BNN using both the Adam and SGD optimizers. Each plot shows the mean and 95% confidence interval for three experimental settings, ordered from top to bottom: Adam with bad seeds (poor convergence), Adam with good seeds (successful convergence), and SGD

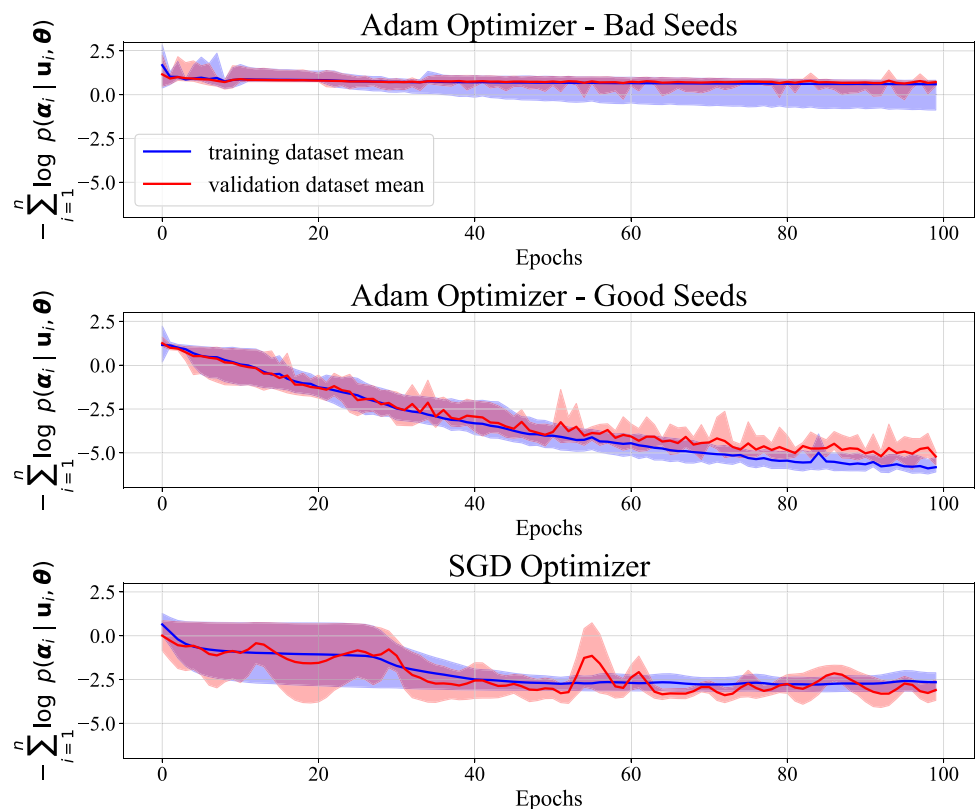


Table 4 Relative (MRE) and absolute (MAE and RMSE) error metrics of the predicted damage parameters for different training regimes, reported in physical units after de-normalization

Optimizer/metric	Parameter		
	Position (x_c)	Width (d_ℓ)	Magnitude (D)
<i>MRE [%]</i>			
Adam with bad seeds	199.84 ± 42.99	132.58 ± 14.42	124.09 ± 3.79
Adam with good seeds	5.52 ± 2.26	11.22 ± 11.40	3.73 ± 3.37
SGD	10.51 ± 3.55	28.43 ± 4.02	19.66 ± 4.45
<i>MAE</i>			
Adam with bad seeds	0.456 ± 0.212 [m]	0.085 ± 0.066 [m]	0.244 ± 0.012 [-]
Adam with good seeds	0.007 ± 0.015 [m]	0.010 ± 0.009 [m]	0.010 ± 0.009 [-]
SGD	0.093 ± 0.022 [m]	0.038 ± 0.004 [m]	0.034 ± 0.017 [-]
<i>RMSE</i>			
Adam with bad seeds	0.625 [m]	0.088 [m]	0.344 [-]
Adam with good seeds	0.017 [m]	0.014 [m]	0.013 [-]
SGD	0.212 [m]	0.074 [m]	0.117 [-]

Fig. 6 Best training runs of the BNN using Adam and SGD optimizers. Blue lines represent training loss, and red lines represent validation loss

errors, while RMSE highlights the impact of larger deviations on the validation/test dataset.

With poor initialization, the Adam optimizer effectively loses damage localization capability, yielding position errors approaching half of the structural span and absolute errors close to the admissible parameter bounds. When properly initialized, Adam achieves the most accurate predictions for the damage location x_c , with absolute errors remaining small relative to the admissible spatial range. The damage magnitude D is estimated with moderate accuracy, exhibiting larger normalized errors than the location parameter but remaining well below the admissible bounds. In contrast, the damage width d_ℓ represents the most challenging parameter, as indicated by its comparatively high MAE and RMSE relative to its narrow admissible range, revealing occasional deviations approaching the parameter limits.

Overall, a well-initialized Adam optimizer outperforms SGD in terms of training and validation accuracy, as shown

in Fig. 6, making it the preferred choice for the final model. Nevertheless, when multiple training runs to identify favorable random seeds are computationally prohibitive, SGD represents a viable alternative due to its more stable behavior across different initializations. Furthermore, although the mean absolute error for the damage width d_ℓ remains moderate in absolute terms, its magnitude becomes significant when compared to the narrow admissible range of this parameter. The elevated RMSE further indicates occasional larger deviations approaching the admissible bound, highlighting the increased sensitivity of width estimation under more challenging scenarios.

The selected BNN model, whose architecture is depicted in Fig. 1, was trained for 100 epochs using the Adam optimizer, with a carefully chosen random seed that yielded stable convergence and robust generalization (see the first plot in Fig. 6). To quantify the impact of the Bayesian layer, a matched deterministic network was trained under identical

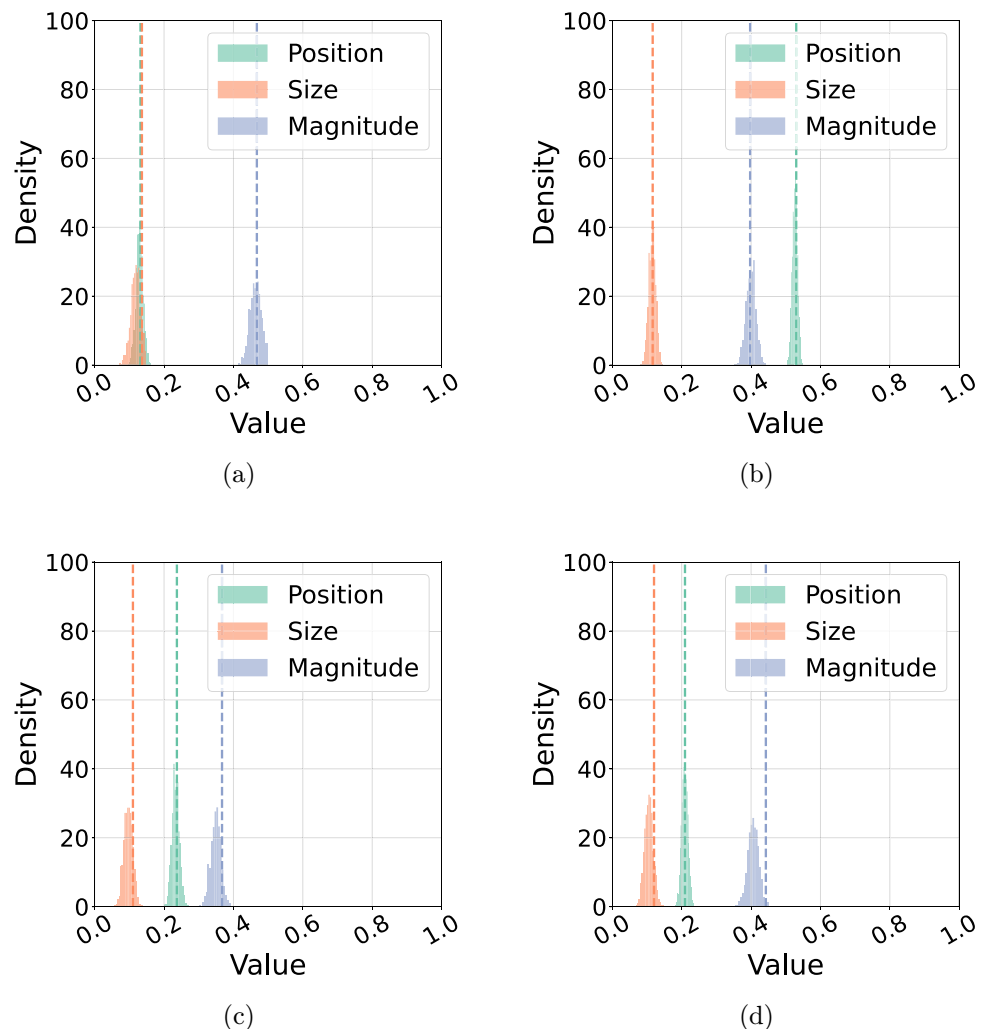
conditions, differing only by the use of an MSE loss instead of the negative log-likelihood. The BNN achieves systematic accuracy gains, reducing the position error by approximately $2.1\times$ in MAE and $1.7\times$ in RMSE, the width error by about $1.3\times$, and the magnitude error by about $1.4\times$ relative to the deterministic model. These improvements are most pronounced in ill-conditioned regimes with weak and narrow defects at 20dB SNR, where the inverse problem is highly ambiguous: the deterministic network tends to regress toward a biased mean, whereas the Bayesian model remains more robust by marginalizing over multiple plausible damage configurations [37]. The only practical advantage of the deterministic network is its faster and more stable convergence during training.

For the trained BNN model inference, 1000 samples were drawn from the model's output for each test/validation sample. These samples correspond to measurements from a bar with rectangular damage profiles considering properties uncertainties (see Tables 1 and 3). In other words, for each input data, 1000 samples were drawn from the estimated distributions of the damage parameters in order to analyze

the stochastic nature of the output. The impact of enforcing the physical constraint imposed by Eq. 11 was assessed through 1000 independent inference trials with 1000 posterior samples each in near-boundary regimes ($x_c < 0.15$ and $x_c > 0.85$), revealing a low average rejection rate ($\approx 4\%$). Although truncation slightly clips the posterior tails and marginally shifts the posterior mean near the boundaries, this effect was found to be negligible and does not introduce any practically relevant bias in the inferred statistics. Despite the relatively large number of forward passes, the inference process remains computationally efficient, as the sampling is performed over the predictive distribution (i.e., model outputs) rather than over the parameter space, as in traditional MCMC methods. To better illustrate the predictions, Fig. 7 shows some examples of the predicted distributions of the parameters alongside their true values.

As shown in the examples of Fig. 7, the predicted distributions for the position parameter closely align with the target, with the mean of the Gaussian distribution being very close to the true value in all cases. Furthermore, it is evident that the magnitude parameter has a higher error in the

Fig. 7 Four examples of the damage parameters estimation using the BNN approach for four different test cases: (a)–(d). The vertical dashed lines represent the true values of the damage parameters, while the bins represent the distributions of the output of the BNN regarding each parameter. Note: Number of samples is equal to 1000



prediction (see Fig. 7d, for instance). Interestingly, the size parameter, although often affected by higher relative errors, tends to vary within a small absolute range, which mitigates its impact on the physical characterization of the damage. Importantly, the model's ability to recover meaningful parameter distributions, even under test conditions that differ from the training data, indicates robust generalization capabilities. In particular, the reliable inference of position, combined with credible estimates of size and magnitude despite the increased variability, highlights the strength of the Bayesian approach in capturing uncertainty and adapting to new conditions. To further visualize how these parameter uncertainties affect the recovered spatial profile of the damage, Fig. 8 presents examples of the reconstructed damage fields derived from the posterior predictions of the BNN.

To further evaluate the generalization capability of the proposed BNN-based inverse model, Fig. 9 presents examples of damage profile reconstructions for a test set featuring a different class of damage shapes from those seen during training. The corresponding relative and absolute error

metrics for this Gaussian test set are reported in Table 5, reported in physical units after de-normalization. Specifically, while the training dataset was generated using rectangular damage profiles, the test samples in this case exhibit Gaussian-shaped damage. This setting aims to emulate real-world conditions, where the actual damage in a structure may not conform to idealized shapes.

From Fig. 9, it is evident that the model can effectively estimate Gaussian profiles. However, it is important to highlight the model's difficulty in estimating certain cases, which is more apparent when recovering Gaussian profiles. This tendency is also reflected in the error metrics summarized in Table 5. In Fig. 9b, three cases stand out in which the model struggles to recover the damage. The first (on the far left) shows a low-magnitude damage, while the second (in the middle) and the third (on the far right) correspond to damages close to the edge of the bar ($x = 1\text{m}$). The first case is indeed near the lower limit of the intensity parameter range, where the scattered response is weak. On the other hand, the second and third cases are affected by multiple

Fig. 8 Examples of damage profiles estimated with the BNN approach for representative test cases. In each panel, (a)–(d), three damage scenarios are considered. The black continuous line represents the true damage profile, while the red continuous line is the mean of the predicted damage profile, accompanied by a 95% confidence envelope

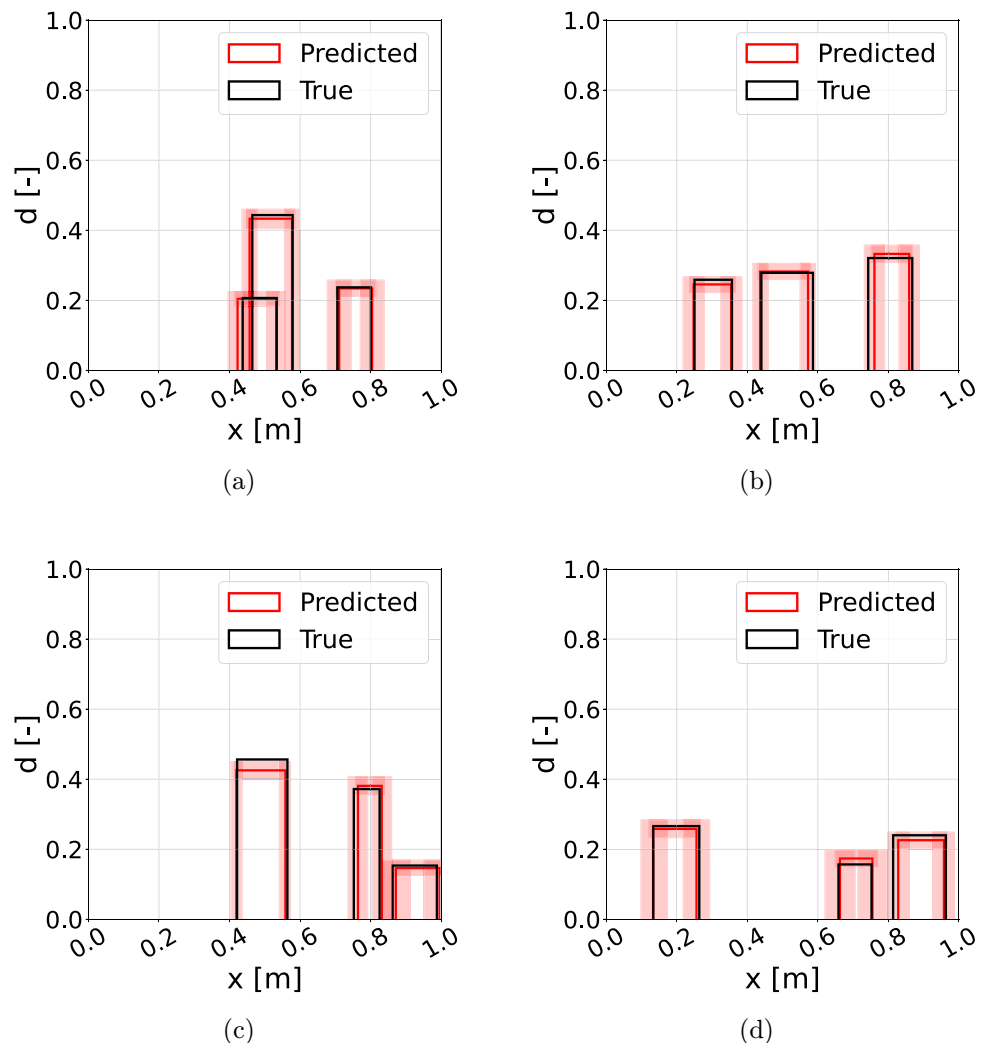


Fig. 9 Examples of Gaussian damage profiles estimated with the BNN approach, trained with rectangular damage profile, for the **a** best and **b** worst scenarios. The black continuous line represents the true damage profile, while the red continuous line is the mean of the predicted damage profile, accompanied by a 95% confidence envelope. The red dashed line links the predicted damage to the true profile, with darker lines indicating larger discrepancies

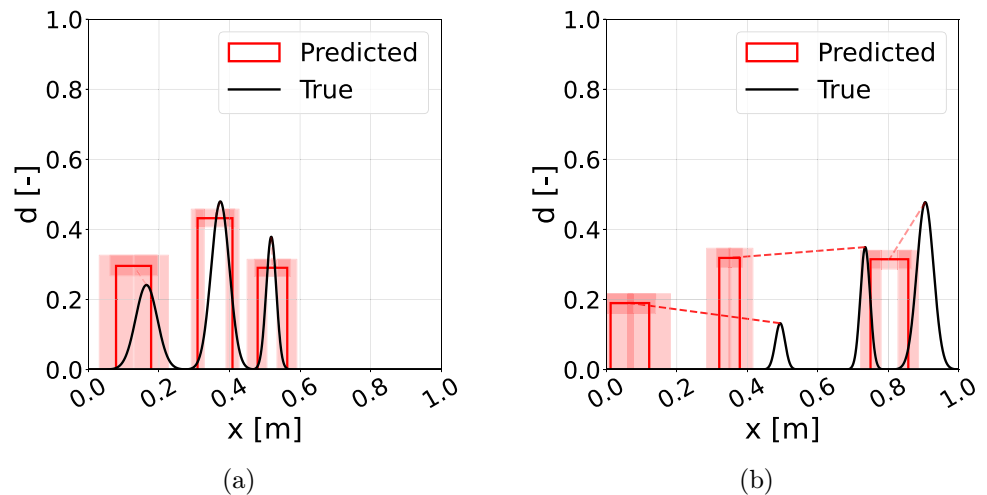
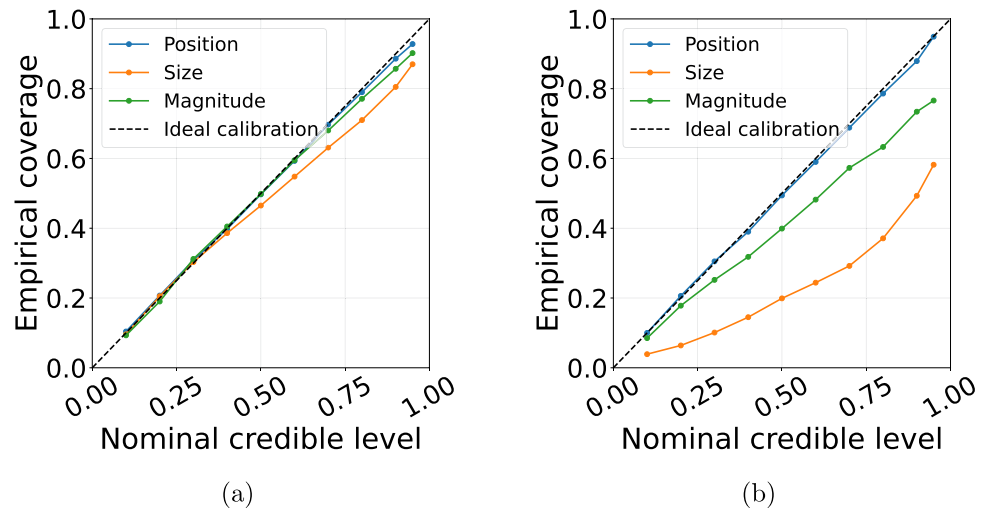


Table 5 Relative (MRE) and absolute (MAE and RMSE) error metrics for Gaussian-shaped damage profiles not seen during training (Adam with good seeds)

Metric	Parameter		
	Position (x_c)	Width (d_ℓ)	Magnitude (D)
MRE [%]	6.33 ± 4.59	31.34 ± 29.07	20.95 ± 21.95
MAE	0.008 ± 0.016 [m]	0.031 ± 0.025 [m]	0.051 ± 0.039 [-]
RMSE	0.022 [m]	0.040 [m]	0.064 [-]

Fig. 10 Reliability diagrams of the BNN for the three damage parameters (position, size, and magnitude). The dashed line denotes perfect calibration, where empirical coverage equals the nominal credible level. **a** rectangular damage fields with uncertain bar properties. **b** Gaussian damage fields



reflections due to the system's boundary effect, as well as the proximity to the excitation location.

In order to evaluate the probabilistic calibration of the BNN, i.e., whether the posterior uncertainty produced by the model correctly reflects the true prediction errors, reliability diagrams are shown in Fig. 10. For a set of nominal credible levels $c \in (0, 1)$, posterior credible intervals $[\hat{\alpha}^-(c), \hat{\alpha}^+(c)]$ for each damage parameter are constructed from the $(1 - c)/2$ and $1 - (1 - c)/2$ quantiles of the posterior distribution. The empirical coverage at level c is then defined as

$$\frac{1}{N} \sum_{i=1}^N 1\{\alpha_i \in [\hat{\alpha}_i^-(c), \hat{\alpha}_i^+(c)]\}, \quad (16)$$

i.e., the fraction of test cases for which the true parameter α lies inside the corresponding credible interval. A perfectly calibrated model therefore satisfies empirical coverage $(c) = c$, corresponding to the diagonal reference line in the diagram [38, 39].

The reliability curves in Fig. 10 show a clear and physically consistent hierarchy among the three damage parameters. For rectangular damage fields with uncertain bar properties, both the damage position and the damage

magnitude lie very close to the ideal calibration line, indicating that the BNN produces well-calibrated posterior uncertainties for these two parameters. In contrast, the damage size exhibits a systematic under-coverage, with empirical coverages lying below the ideal line, revealing an overconfident posterior. This behavior is consistent with the fact that the size is the least identifiable parameter and therefore the most difficult to infer reliably. From a physical standpoint, the damage width parameter d_ℓ affects the structural response in a more diffuse and spatially distributed manner than the position x_c or the magnitude D . While the position primarily controls where the perturbation in stiffness is localized and the magnitude governs the intensity of the stiffness reduction, the width mainly alters how this reduction is spatially spread. Consequently, different combinations of d_ℓ and D may produce similar dynamic responses, leading to partial parameter correlation and reduced identifiability. Moreover, localized damage signatures are typically more strongly reflected in modal features than the precise spatial extent of the damaged region, especially under limited sensor and modal resolution. Moreover, localized damage signatures are typically more strongly reflected in the wave response than the precise spatial extent of the damaged region, especially when the measurements are obtained from a limited number of spatially distributed sensors. As a result, variations in d_{ell} tend to produce subtler changes in the measured wavefield, making this parameter intrinsically harder to identify reliably.

When the model is evaluated on Gaussian-shaped damage fields, a stronger distributional shift is observed. Remarkably, the posterior for the damage position remains

well calibrated, demonstrating that the spatial localization of damage is robust to the change in damage morphology. The magnitude shows a noticeable degradation in calibration, while the size suffers the largest deterioration, becoming strongly overconfident across all credible levels. These results indicate that while the BNN has learned a robust and transferable uncertainty model for the damage position, the uncertainty associated with shape-dependent parameters—particularly the size—is far more sensitive to model mismatch and suffers significantly under out-of-distribution conditions. Furthermore, a global sensitivity performed using the first order Sobol indices over the parameter range from Table 3 shows that, based on their indices distribution, the parameters can be sorted in the following order of decreasing sensitivity overall: x_c , D and d_ℓ – the top and lower 40% percentiles for these parameters can be seen to satisfy this sorting. This is compatible with the results from the Reliability diagrams (Fig. 10) and Table 4.

To better illustrate how different damage profiles affect the structural response captured by the sensors—and, consequently, the complexity of the inverse problem—Figs. 11, 12, 13, 14, 15 show the displacement fields recorded by the eight sensors for five representative scenarios: a healthy structure, and both rectangular and Gaussian damage profiles with identical parameters, at two different damage magnitudes (high and low). These examples help contextualize the degree of similarity or overlap in sensor responses caused by distinct damage profiles, or the absence of damage, which can significantly contribute to prediction uncertainty.

From a mechanical perspective, variations in the position, size, and magnitude of the damage field directly affect the

Fig. 11 Sensor readings for a healthy bar system

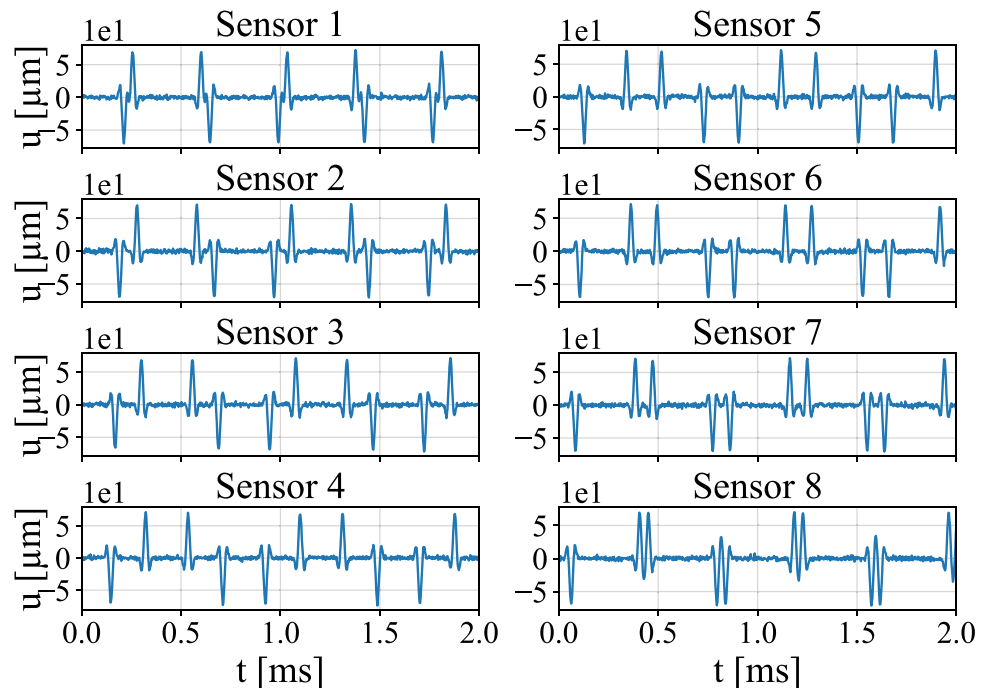


Fig. 12 Sensor readings and model prediction for a rectangular damage profile with high magnitude (a favorable case for model prediction)

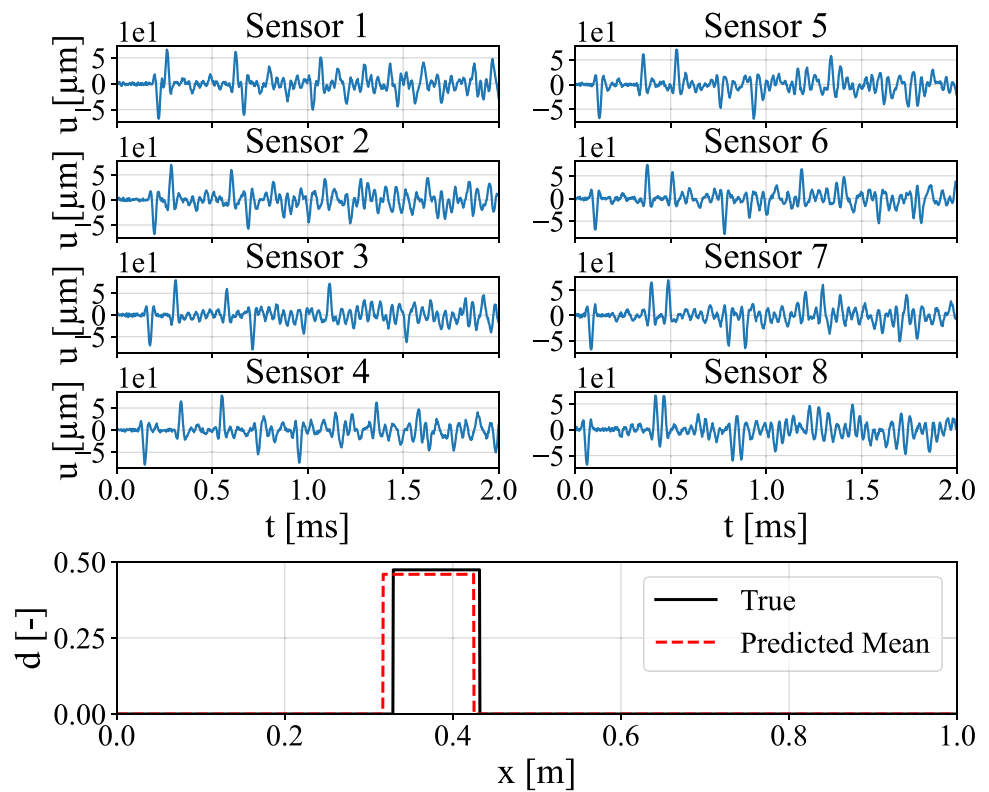


Fig. 13 Sensor readings and model prediction for a Gaussian damage profile with high magnitude (a favorable case for model prediction)

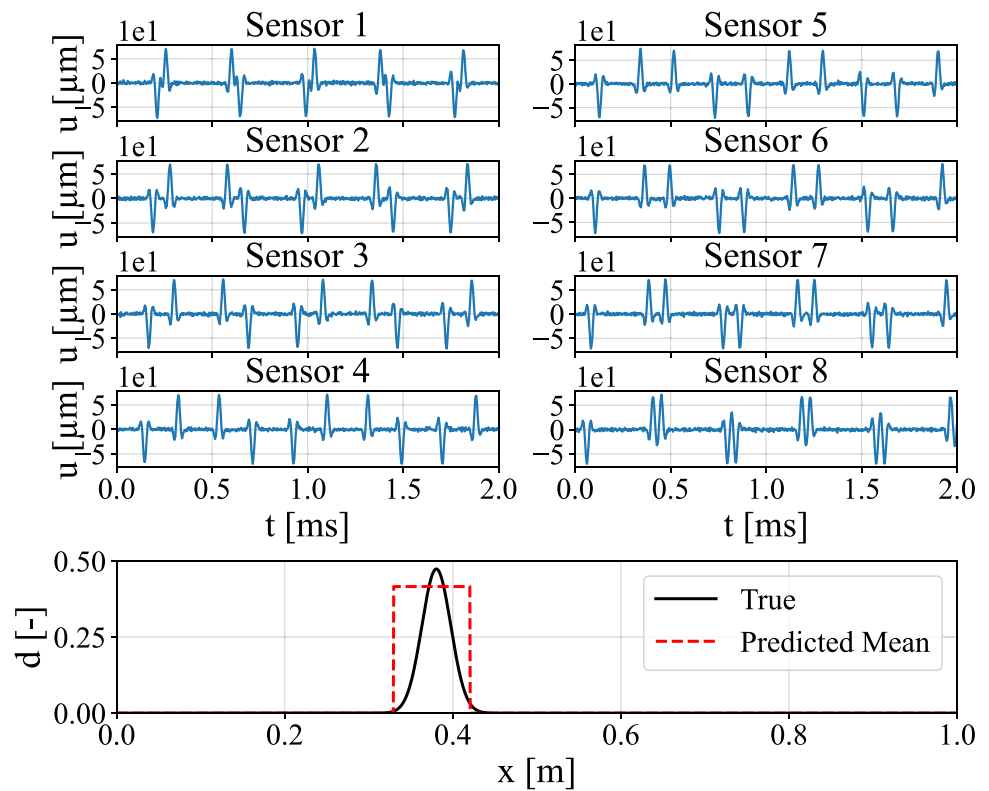


Fig. 14 Sensor readings and model prediction for a rectangular damage profile with low magnitude (an unfavorable case for model prediction)

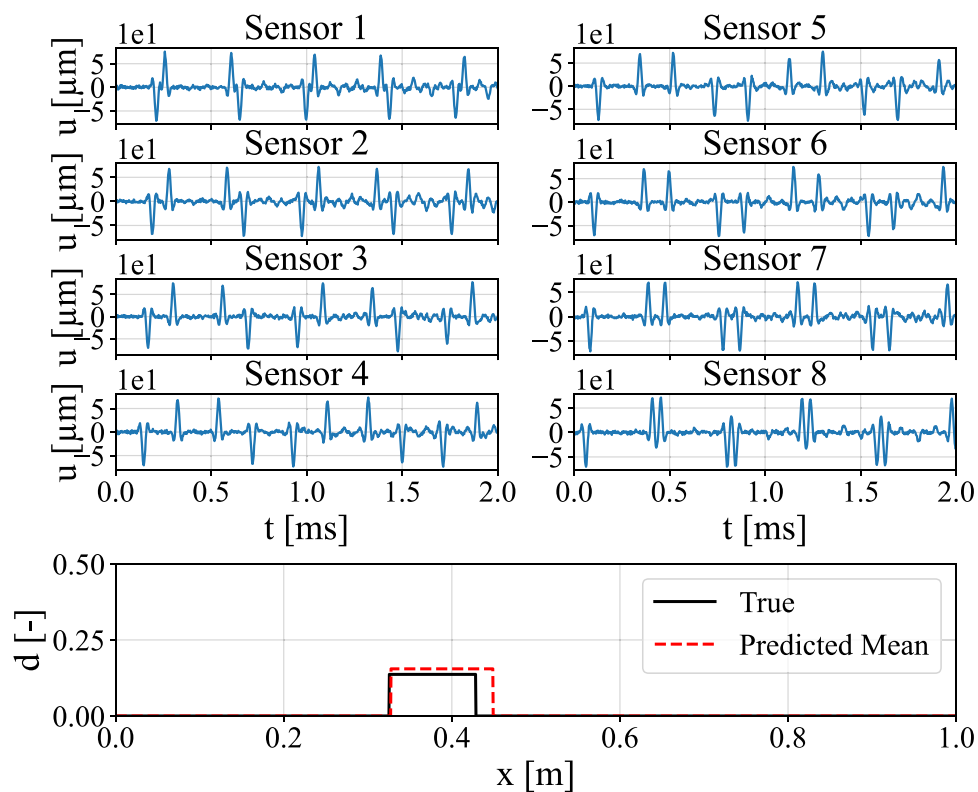
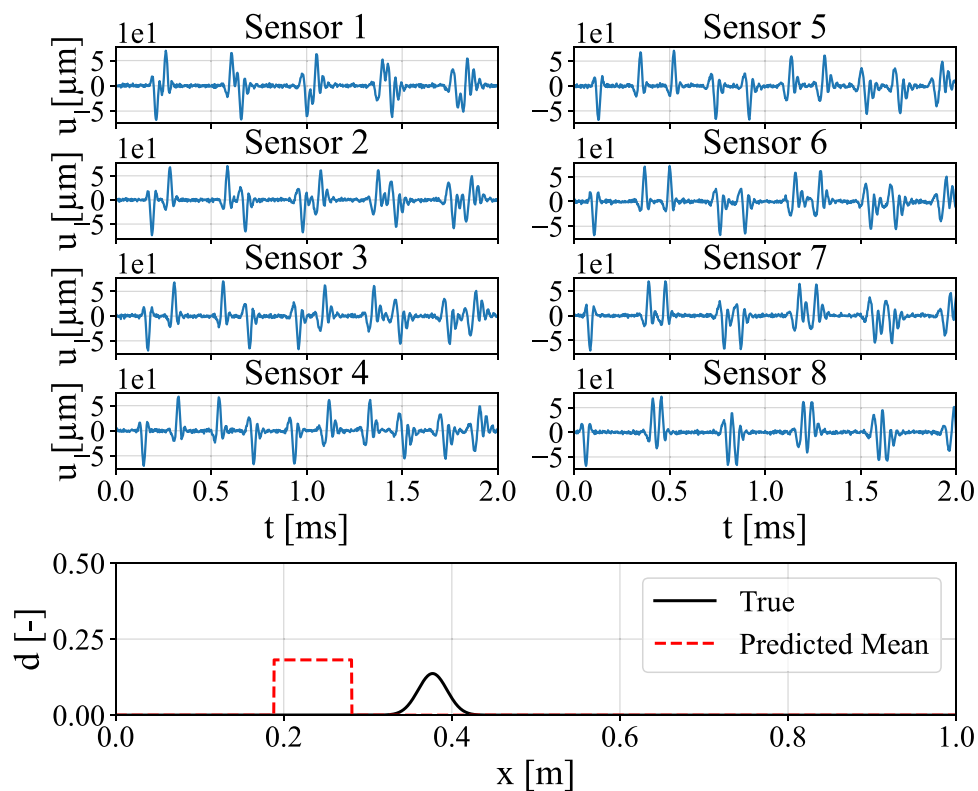


Fig. 15 Sensor readings and model prediction for a Gaussian damage profile with low magnitude (an unfavorable case for model prediction)



local stiffness and mass distribution of the structure, thereby modifying wave propagation characteristics such as reflection and scattering captured by the sensors; these systematic changes in the measured signals are precisely what enable SHM, as they encode information that allows the characterization of the damage field. In particular, sensor signals from damaged structures can exhibit notably different patterns depending on whether the damage follows a rectangular or Gaussian profile, even when the parameters are identical. For example, signals associated with the rectangular profile often display more erratic temporal behavior, especially at higher damage magnitudes. These discrepancies in signal behavior across different damage shapes, despite identical parameter values, introduce additional uncertainty into the inverse inference process, potentially accounting for the higher prediction variances observed in some cases. This phenomenon underscores both the inherent complexity of structural damage characterization and the ability of the BNN to generalize across varying damage profile scenarios.

To the best of the authors' knowledge, in practical acoustic inspections, the most critical parameter to estimate accurately is the damage position. This type of inspection is understood here as a form of non-destructive evaluation (NDE) in which guided waves propagate through the structure to detect and characterize damage. The location of the damage is crucial because it directly informs where technicians should focus subsequent inspection or maintenance efforts. Although size and magnitude can provide insight into the severity of the damage, in large structures, failing to accurately estimate the location of the damage can make the technician's work far less effective, with the risk of failing to detect any damage. Therefore, to assess the effectiveness of the proposed model in supporting the SHM procedure for the analyzed system, a detailed evaluation of the position inference was conducted based on the test data, for both rectangular and Gaussian damage profiles (Figs. 16 and 17). In addition to global error metrics, joint residual analyses were performed to investigate potential performance degradation across different regions of the parameter space.

For rectangular damage profiles with uncertainty in the bar properties, the residual analysis in Fig. 16 shows that the damage position is, on average, estimated with high accuracy, with most residuals concentrated near zero. In quantitative terms, poorly predicted samples related to the position parameter—defined here as cases in which the absolute discrepancy between the predicted posterior mean and the true position exceeds 5 cm—account for only 0.8% of the total test set. However, the joint and conditional residual plots indicate that the localization error is not uniform across the parameter space. Larger position errors tend to occur for narrow damages (small d_ℓ) and for low-magnitude defects (D), reflecting the fact that such damage profiles

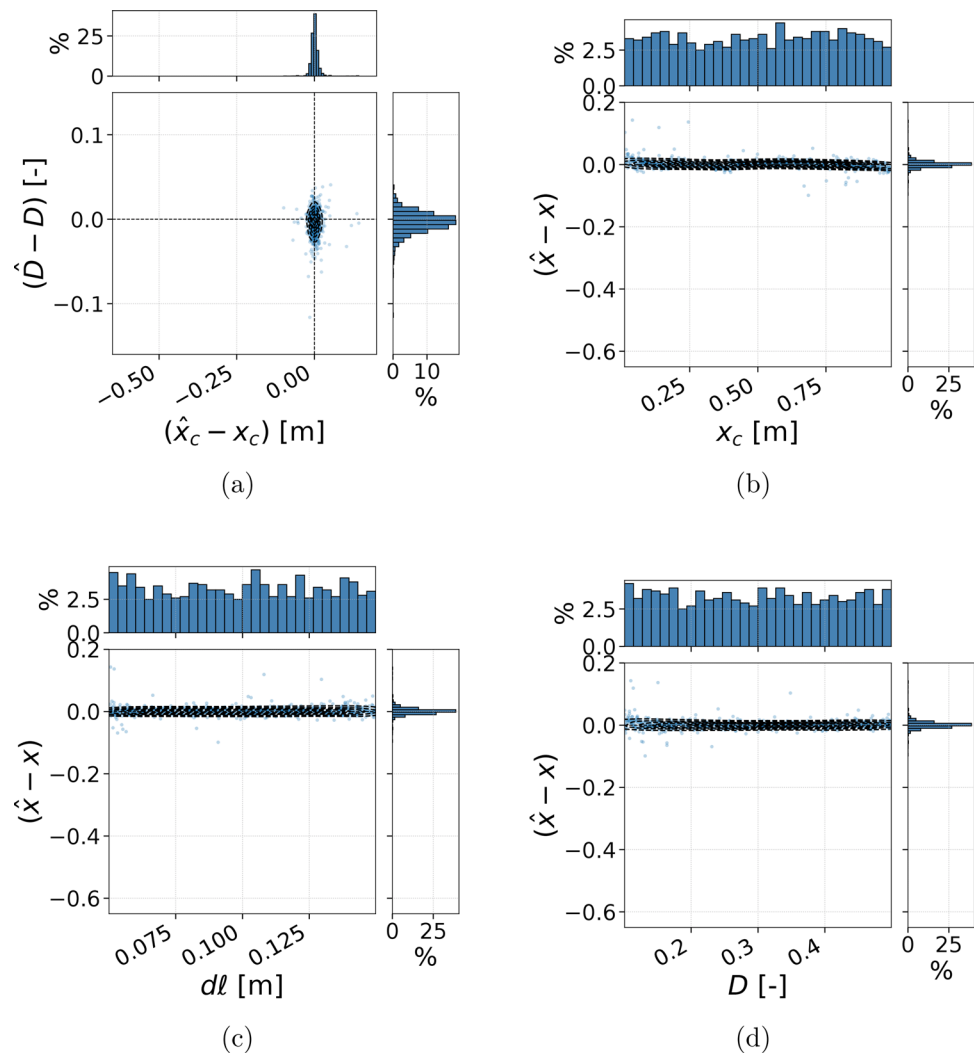
generate weaker and more spatially compact scattering patterns, which are inherently harder to detect and localize. This behavior is visible in Fig. 16c, d, where the spread of the position residual increases as both the damage size and magnitude decrease, with a particularly strong dependence on the magnitude parameter. Conversely, for wider and higher-magnitude damages, the residuals collapse tightly around zero, apart from a small number of outliers, indicating a well-conditioned localization regime in these regions of the parameter space.

For Gaussian damage profiles, the residual analysis in Fig. 17 indicates that the model is still able to localize damage positions with good overall accuracy, as the position residuals remain largely concentrated around zero across most of the parameter space. However, the joint residual distributions in Fig. 17a reveal a systematic distortion in the coupled inference of the damage parameters: while the position residuals remain centered, the magnitude residual axis exhibits a noticeable shift, and the contour levels associated with the size residuals are also displaced toward negative values. This indicates a tendency of the model to underestimate both the damage magnitude and its spatial extent for Gaussian profiles, even when the position is correctly recovered. In line with the behavior observed for rectangular damage profiles, the Gaussian case also exhibits clear and physically interpretable regimes of degraded localization performance, albeit in a more pronounced manner. In quantitative terms, poorly predicted samples related to the position parameter—defined here as cases in which the absolute discrepancy between the predicted posterior mean and the true position exceeds 5 cm—account for 5.5% of the Gaussian test set, representing a substantial increase relative to the rectangular case. As shown in Fig. 17b–d, larger position errors tend to occur for narrow damage profiles (small d_ℓ), for low-magnitude defects ($D \lesssim 0.2$), and for damages located near the boundaries of the domain. This structured pattern of errors suggests that the observed performance degradation is driven primarily by intrinsic physical limitations of the wave-based inspection problem, rather than by a global lack of generalization of the Bayesian neural network.

5 Discussion

This approach provides an alternative to estimating parameters within an inverse problem framework, effectively creating a surrogate model for inversion that offers enhanced analytical capabilities compared to deterministic NNs. For instance, when dealing with uncertainty and incorporating probabilistic reasoning, a digital twin could benefit from the stochastic nature of a BNN, something a deterministic

Fig. 16 Residual analysis of rectangular damage profile predictions for validation/test samples with uncertain bar properties. **a** Joint distribution of magnitude and position residuals, with contour lines given by a kernel density estimate weighted by the absolute width residual. **b** Position residual as a function of the true damage position. **c** Position residual as a function of the true damage width. **d** Position residual as a function of the true damage magnitude. Marginal histograms indicate the percentage of samples



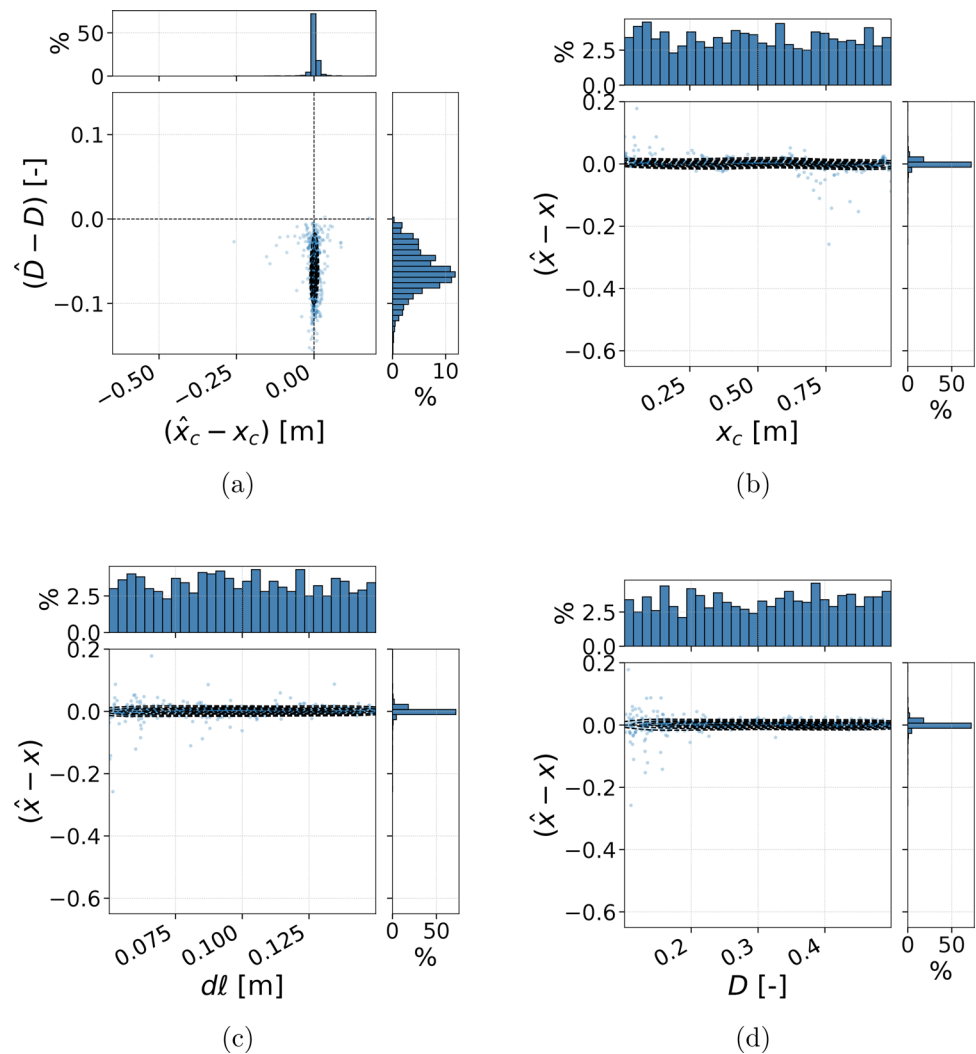
model cannot provide [40, 41]. Furthermore, for wave-propagation-based SHM, DL models enable near-real-time inference once trained, offering a key practical advantage over classical inversion approaches [25].

Due to the nature of data-driven models, the trained neural network is specific to the excitation, boundary conditions, and damage parameterization represented in the training data, and cannot be directly generalized to different structural configurations without retraining. Accordingly, the present approach is restricted to a one-dimensional bar under guided-wave propagation, where dispersive effects and boundary reflections are implicitly controlled by the excitation bandwidth and time-windowing strategy, and its applicability is therefore bounded by these assumptions. Nevertheless, BNNs have been successfully applied to other structures, such as beams and plates [7, 12], and the present work further demonstrates their viability for modeling wave-damage interactions in localized regions.

The present BNN addresses non-identifiability at the posterior level by marginalizing over multiple plausible

solutions. However, as previously observed, its training remains sensitive to initialization, and additional strategies are required to improve convergence behavior. More generally, related hybrid ML frameworks have proposed complementary approaches, including metaheuristic optimization methods (e.g., grey wolf optimizer and gene expression programming–invasive weed optimization hybrids) [42, 43] and preconditioning techniques such as Random Forest and Genetic Algorithm (RF-GA) and sensitivity-based methods [44], which improve convergence by globally optimizing network parameters or conditioning the input space prior to training. In addition, practical strategies such as multiple-seed training and ensemble averaging can help mitigate sensitivity to random initialization, although they may significantly increase computational cost. Alternatively, recent studies have proposed improved optimization schemes, including methods based on Halpern’s iteration [45] and modified variants of the Adam algorithm [46]. These strategies could be combined with the proposed framework in

Fig. 17 Residual analysis for gaussian damage profile predictions. **a** Joint distribution of magnitude and position residuals, with contour lines given by a kernel density estimate weighted by the absolute width residual. **b** Position residual as a function of the true damage position. **c** Position residual as a function of the true damage width. **d** Position residual as a function of the true damage magnitude. Marginal histograms indicate the percentage of samples



future work to enhance convergence robustness while preserving its probabilistic characterization.

6 Conclusions

In the analysis of 1D wave propagation in an elastic bar, the behavior of a BNN applied to regression was investigated to model a simple damage field by estimating the probability distributions of the parameters that fully characterize the damage. The proposed inverse surrogate was evaluated in relation to test data that included uncertainties in the physical system properties and damage profile shapes not seen during training. The goal was to recover damage parameters, such as the center, width, and magnitude. The main findings of this study can be summarized as follows: (i) the BNN is capable of accurately estimating damage position and magnitude from sensor readings of wavefields, with position being the most reliably recovered and practically relevant parameter for SHM applications; (ii) the convergence of

the BNN remains challenging, even for the simple architecture adopted in the present work, showing sensitivity to parameter initialization and optimizer choice; in particular, SGD exhibited greater stability than Adam, which was more prone to convergence to poor local minima, consistent with known optimization difficulties in BNN training [14]; (iii) the estimation of damage size is less accurate, which may be attributed to reduced sensitivity of the measured signals to this parameter, reflecting either limitations of the current formulation or inherent identifiability constraints of the problem; (iv) the model performance degrades in specific regions of the parameter space, particularly for low-magnitude, narrow, or boundary-adjacent damage; and (v) these limitations are physically interpretable, arising from reduced wave–damage interaction, boundary effects, and overlapping wavefields, rather than reflecting a general loss of model robustness.

The proposed approach serves as an auxiliary tool for acoustic inspections and can be extended to more complex systems. Although the training stage is computationally

demanding (particularly due to the generation of training samples through spectral finite element simulations and the need to identify suitable parameter initializations for stable BNN convergence), the inference stage is extremely efficient once the network is trained. It requires only a forward pass through the BNN to estimate the posterior distributions of the damage parameters. In contrast, traditional inverse-modeling approaches typically require thousands of evaluations of the forward model within optimization or sampling procedures to obtain comparable uncertainty estimates. Therefore, despite the high offline computational cost, the proposed surrogate-based framework offers substantial advantages for near real-time SHM applications. Future work will focus on sensitivity analyses with respect to the number of sensors, placement, and noise levels to better assess applicability under realistic conditions. Additionally, investigating the placement of probabilistic layers may improve training stability. Ongoing studies involving experimental data aim to assess the effects of measurement noise and modeling errors, thereby providing further validation of the proposed framework and guiding refinements of both the sensing configuration and the BNN architecture.

Acknowledgements Daniel Alves Castello would like to express his gratitude for the financial support provided by CNPq under grant number 312669/2023-2.

Author contributions Bernardo F. Junqueira: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Visualization, Writing – original draft, review and editing. Gabriel L. S. Silva: Conceptualization, Formal analysis, Investigation, Writing – original draft, review and editing. Rafael de O. Teloli: Conceptualization, Investigation, Writing – review and editing. Ricardo Leiderman: Conceptualization, Investigation. Daniel A. Castello: Conceptualization, Writing – original draft, Writing – review and editing.

Funding The Article Processing Charge (APC) for the publication of this research was funded by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) (ROR identifier: 00x0ma614).

Data availability Datasets generated during the current study are available from the corresponding author on reasonable request.

Declarations

Conflict of 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.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate

if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Bao Y, Li H (2021) Machine learning paradigm for structural health monitoring. *Struct Health Monit* 20(4):1353–1372
2. Koptika N, Blikharsky Y (2024) Digital image correlation for assessment of bridges' technical state and remaining resource. *Struct Control Health Monit* (1):1763285
3. Okur FY, Altunışık AC, Kalkan Okur E (2025) A novel approach for anomaly detection in vibration-based structural health monitoring using autoencoders in deep learning. *Structural Control and Health Monitoring* (1):5602604
4. Cha Y-J, Ali R, Lewis J, Büyük"oztürk O (2024) Deep learning-based structural health monitoring. *Autom Constr* 161:105328
5. Jospin LV, Laga H, Boussaid F, Buntine W, Bennamoun M (2022) Hands-on bayesian neural networks-a tutorial for deep learning users. *IEEE Comput Intell Mag* 17(2):29–48
6. Goan E, Fookes C (2020) Bayesian neural networks: An introduction and survey. *Case Studies in Applied Bayesian Data Science: CIRM Jean-Morlet Chair. Fall 2018*. Springer, Cham, pp 45–87
7. Vashisht R, Viji H, Sundararajan T, Mohankumar D, Sumitra S (2019) Structural health monitoring of cantilever beam, a case study-using bayesian neural network and deep learning. *Structural Integrity Assessment: Proceedings of ICONS 2018*. Springer, Singapore, pp 749–761
8. Vega MA, Todd MD (2022) A variational bayesian neural network for structural health monitoring and cost-informed decision-making in miter gates. *Struct Health Monit* 21(1):4–18
9. Lei X, Sun Z, Wang A, Guo T, Nagayama T (2025) Estimation of bridge girder cumulative displacement for component operational warning using bayesian neural networks. *Struct Control Health Monit* (1):9974584
10. Yu H, Seno AH, Sharif Khodaei Z, Aliabadi MF (2022) Structural health monitoring impact classification method based on bayesian neural network. *Polymers* 14(19):3947
11. Hlaing N, Morato PG, Santos FDN, Weijtjens W, Devriendt C, Rigo P (2024) Farm-wide virtual load monitoring for offshore wind structures via bayesian neural networks. *Struct Health Monit* 23(3):1641–1663
12. Luo K, Zhu J, Li Z, Zhu H, Li Y, Hu R, Fan T, Chang X, Zhuang L, Yang Z (2024) Ultrasonic lamb wave damage detection of cfrp composites using the bayesian neural network. *J Nondestr Eval* 43(2):48
13. Hamadneh M, Mustapha S, Fakhri MA (2024) Impact-force identification using deep learning and bayesian inference with application on pipeline structures. *Struct Health Monit* 14759217241297572
14. Papamarkou T, Hinkle J, Young MT, Womble D (2022) Challenges in markov chain monte carlo for bayesian neural networks. *Stat Sci* 37(3):425–442
15. Graff KF (2012) *Wave Motion in Elastic Solids*. Courier Corporation, New York
16. Silva GL, Castello DA (2023) Parametric reduced order models for wave propagation in 1d media containing defects. *J Sound Vib* 558:117771

17. Komatitsch D, Tsuboi S, Tromp J, Levander A, Nolet G (2005) The spectral-element method in seismology. *Geophysical Monograph-American Geophysical Union* 157:205
18. Kim W, Lee JH (2018) An improved explicit time integration method for linear and nonlinear structural dynamics. *Computers & Structures* 206:42–53
19. Moaveni B, Behmanesh I (2012) Effects of changing ambient temperature on finite element model updating of the dowing hall footbridge. *Eng Struct* 43:58–68
20. Venanzi I, Kita A, Cavalagli N, Ierimonti L, Ubertini F (2019) Continuous oma for damage detection and localization in the sciri tower in perugia, italy. *Proceedings of the International Operational Modal Analysis Conference (IOMAC 2019)*. pp 127–136 Unknown
21. Lu Z, Law S (2010) Differentiating damage effects in a structural component from the time response. *Mech Syst Signal Process* 24(8):2914–2928
22. Silva GL, Castello DA, Borges L, Kaipio JP (2020) Damage identification in plates under uncertain boundary conditions. *Mech Syst Signal Process* 144:106884
23. Chen C-T, Gu GX (2021) Learning hidden elasticity with deep neural networks. *Proc Natl Acad Sci* 118(31):2102721118
24. Silva GL, Castello DA, Kaipio JP (2021) Damage identification under uncertain mass density distributions. *Comput Methods Appl Mech Eng* 376:113672
25. Junqueira BF, Leiderman R, Castello DA (2024) Damage recovery in composite laminates through deep learning from acoustic scattering of guided waves. *Ultrasonics* 139:107293
26. Blundell C, Cornebise J, Kavukcuoglu K, Wierstra D (2015) Weight uncertainty in neural network. In: *International Conference on Machine Learning*. pp 1613–1622 PMLR
27. Pearce T, Leibfried F, Brintrup A (2020) Uncertainty in neural networks: Approximately bayesian ensembling. In: *International Conference on Artificial Intelligence and Statistics*. pp 234–244 PMLR
28. Magris M, Iosifidis A (2023) Bayesian learning for neural networks: an algorithmic survey. *ArtifIntell Rev* 56(10):11773–11823
29. Ji S, Zhang Z, Ying S, Wang L, Zhao X, Gao Y (2020) Kullback-leibler divergence metric learning. *IEEE Trans Cybern* 52(4):2047–2058
30. Lampinen J, Vehtari A (2001) Bayesian approach for neural networks-review and case studies. *Neural Netw* 14(3):257–274
31. Takiy AE, Kitano C, Higuti RT, Granja SC, Prado VT, Elvira L, Martínez-Graullera O (2017) Ultrasound imaging of immersed plates using high-order lamb modes at their low attenuation frequency bands. *Mech Syst Signal Process* 96:321–332
32. Su Z, Ye L, Lu Y (2006) Guided lamb waves for identification of damage in composite structures: A review. *J Sound Vib* 295(3–5):753–780
33. Silva GLS, Junqueira BF, Castello DA, Leiderman R (2026) Machine learning inverse surrogates for damage identification in plates based on lamb waves. *Ultrasonics* 159:107838. <https://doi.org/10.1016/j.ultras.2025.107838>
34. He K, Zhang X, Ren S, Sun J (2015) Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: *Proceedings of the IEEE international conference on computer vision*. pp 1026–1034
35. Glorot X, Bengio Y (2010) Understanding the difficulty of training deep feedforward neural networks. In: *Proceedings of the Thirteenth international conference on artificial intelligence and statistics*. pp 249–256
36. Zhou P, Feng J, Ma C, Xiong C, Hoi SCH et al (2020) Towards theoretically understanding why sgd generalizes better than adam in deep learning. *Adv Neural Inf Process Syst* 33:21285–21296
37. Yang L, Meng X, Karniadakis GE (2021) B-pinns: Bayesian physics-informed neural networks for forward and inverse pde problems with noisy data. *J Comput Phys* 425:109913
38. Martin R, Ning B (2020) Empirical priors and coverage of posterior credible sets in a sparse normal mean model. *Sankhya A* 82(2):477–498
39. Lee JE, Nicholls GK, Ryder RJ (2019) Calibration procedures for approximate bayesian credible sets. *Bayesian Anal* 14(4)
40. Yu J, Song Y, Tang D, Dai J (2021) A digital twin approach based on nonparametric bayesian network for complex system health monitoring. *J Manuf Syst* 58:293–304
41. Yang J, Langley RS, Andrade L (2022) Digital twins for design in the presence of uncertainties. *Mech Syst Signal Process* 179:109338
42. Ahmadi M, Ebadi-Jamkhaneh M, Dalvand A, Rezazadeh Eidgahee D (2024) Hybrid bio-inspired metaheuristic approach for design compressive strength of high-strength concrete-filled high-strength steel tube columns. *Neural Comput Appl* 36(14):7953–7969
43. Shariati M, Mafipour MS, Ghahremani B, Azarhomayun F, Ahmadi M, Trung NT, Shariati A (2022) A novel hybrid extreme learning machine-grey wolf optimizer (elm-gwo) model to predict compressive strength of concrete with partial replacements for cement. *Eng Comput* 38(1):757–779
44. Sharghi A, Komasi M, Ahmadi M (2025) Variable sensitivity analysis in groundwater level projections under climate change adopting a hybrid machine learning algorithm. *Environ Model Soft* 183:106264
45. Colao V, Foglia KR, Giordano A, Ritacco E, Spataro W (2026) An optimizer derived from halpern's method for enhanced neural network convergence and reduced carbon emissions. *J Intell Inf Syst* 64(1):77–96
46. Reyad M, Sarhan AM, Arafa M (2023) A modified adam algorithm for deep neural network optimization. *Neural Comput Appl* 35(23):17095–17112

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.