A physics-informed neural networks framework for model parameter identification of beam-like structures

Rafael de O. Teloli [*], Roberta Tittarelli, Maël Bigot, Lucas Coelho, Emmanuel Ramasso, Patrice Le Moal, Morvan Ouisse

SUPMICROTECH, Université de Franche-Comté, CNRS, Institut FEMTO-ST, Besançon, F-25000, France

ARTICLE INFO

ABSTRACT

This study introduces an innovative approach that employs Physics-Informed Neural Networks (PINNs) to address inverse problems in structural analysis. Specifically, this technique is applied to the 4th order partial differential equation (PDE) of the Euler–Bernoulli formulation to estimate beam displacement and identify structural parameters, including damping and elastic modulus. The methodology incorporates PDEs into the neural network's loss function during training, ensuring it adheres to physics-based constraints. This approach simplifies complex structural analysis, even when explicit knowledge of boundary conditions is unavailable. Importantly, the method reliably captures structural behavior without resorting to synthetic noise in data — an experimental application is put forward to validate the framework. This study represents a pioneering effort in utilizing PINNs for inverse problems in structural analysis, offering potential inspiration for other fields. The characterization of damping, a typically challenging task, underscores the versatility of methodology. The strategy is initially assessed through numerical simulations utilizing data from a finite element solver and subsequently applied to experimental datasets. The presented methodology successfully identifies structural parameters using experimental data and validates its accuracy against state-of-the-art techniques. This work opens new possibilities in engineering problem-solving, positioning Physics-Informed Neural Networks as valuable tools in addressing practical challenges in structural analysis.

1. Introduction

Decision-making processes are intrinsic to engineering systems, crucial for crafting designs that balance performance, reliability, safety, and cost-effectiveness. Engineers rely on decision support mechanisms that integrate physics-based simulations, acting as virtual prototyping tools [1]. These simulations enable the quantification of complex structure behavior under various multiphysical loads. Through virtual prototyping, mathematical models are formulated using partial differential equations (PDEs), offering a comprehensive understanding of the dynamic interactions within engineering structures. Solving PDEs, particularly when analytical solutions are unavailable (a common scenario), requires the use of numerical methods, as finite elements (FE), finite differences, spectral methods and others. However, these classical numerical algorithms, while workhorses in simulating physical systems in modern engineering, face challenges like sensitivity to boundary conditions, meshing problems, and computational time dependency on the complexity of the analyzed model, among others.

From a data-centric perspective, the increasing precision and capacity to gather data have led to the incorporation of machine learning algorithms in engineering practices. Application fields involve classification tasks such as image recognition, regression of

---

physical process, structural monitoring analysis or surrogate modeling. While these data-driven techniques demonstrate capabilities in managing extensive datasets and flexibility in adjusting their structures (nonlinearities inherent to these representations are capable to provide universal approximations) to accommodate quantities of interest from diverse systems, they often pose challenges due to their lack of interpretability. In the context of system identification, both data and often some physical hypotheses are available [2]. In the worst-case scenario, where physical interactions are not well understood, all the relevant information about the system must be present in data. Unfortunately, although the data deluge one is able to acquire, data often do not capture all behaviors of interest, and might be corrupted or noisy. Therefore, the lack of interpretability, along with challenges like forecasting outside the training domain and instability [3], restricts their extensive utilization in modeling engineering structures — specially the high consequence ones. Given the significance of decision support in engineering endeavors, there exists a need for models and outcomes that prioritize parsimony, interpretability and consistency.

Bridging the gap between physics-based models and data-driven approaches, recent algorithms in machine learning focus on inducing learning bias towards physically consistent solutions, giving rise to the paradigm known as physics-informed machine learning [4] (alternatively termed physics-augmented machine learning, among others). Notably, Physics-informed neural networks (PINNs) [5] have gained attention in computational engineering, finding applications like in micro-continuum mechanics [6], thermochemical process [7] or fluid dynamics [8]. In PINN architectures, physical constraints are included during training through the loss function, ranging from conservative energy laws (weak form) to PDEs (strong form).

The application of PINNs in the mechanics community has expanded recently. Roy and Guha [9] introduced a PINN framework for solving von Mises plasticity problems under plane-strain assumption. For integrating physics into the training process of the neural network, a multi-objective loss function is introduced, accounting constitutive relations and boundary conditions. As'ad and Farhat [10] put forward a computational strategy for training a mechanics-informed neural network, with the objective of modeling nonlinear viscoelastic problems and, at the same time, ensuring the agnostic form of the model to have dynamic and material stability, as well as consistency. Rezaei et al. [11] compared the potential of PINNs as solver in solid mechanics with the FE method. Their physics-informed loss function captured spatial gradients (strong form) and the energy form (weak form). Worthy noting that both methods share similarities when using nonlinear interpolations – neural networks for PINNs and shape functions for FE methods – to solve PDEs and element nodes, respectively. Drawing inspiration from classical numerical integration schemes, Zhai et al. [12] introduced the Runge–Kutta PINN to model evolutionary nonlinear dynamical systems. In other words, instead of addressing partial differential equations (PDEs), their methodology tackles problems represented by ordinary differential equations (ODEs). A similar approach was proposed by [13], which exploited the use of neural ordinary differential equations. These works represent a bird's eye view of the most recent studies focusing on the application of PINNs to solve forward problems in mechanical systems — where inputs, parameters, and physical models are known, and the objective is to employ PINN algorithms to determine the corresponding outputs. However, if the number of papers investigating inverse problems based on PINN formulation for mechanical systems is increasing, there is still a lack of contributions for solving inverse problems with PINNs in structural dynamics, where frequency domain analysis is common and damping plays a major role in the estimation of the quantities of interest.

In a broader context than the framework proposed by Roy and Gua, Haghighat et al. [14] explored the application of PINNs for modeling nonlinear von Mises elastoplasticity and simultaneously conducting parameter identification. A mixed formulation was employed, wherein independent neural networks modeled the displacement and components of the Cauchy stress tensor. Additionally, the Lamé parameters were treated as unknowns, thus setting up the core of an inverse problem configuration. Yucesan et al. [15], in turn, introduced a PINN framework to refine a torsional vibration damper model based on experimental data. In contrast to the works addressed so far that explored PDEs, they employed the neural network within a direct graph representation to update the stiffness and damping coefficients. The graph model takes observed frequency response function (FRF) as input, generating an estimated FRF based on the updated coefficients, forming the basis for the loss function. Guo and Fang [16] proposed a parameter estimation framework incorporating training data derived from a validated FE model, where modal parameters served as physical constraints for the PINNs. Despite being a commonly employed method in vibration-based system identification, modal analysis is resource-intensive, incurring significant costs, and providing only global information about the system under analysis, with potential uncertainties stemming from estimation biases. Additionally, PINNs have been investigated in the context of structural health monitoring, specifically for solving inverse problems related to monitoring damage-sensitive parameters. This is exemplified by the work of Di Lorenzo et al. [17] and Zhou and Xu [18].

This paper aims to present a physics-informed approach, encompassing inverse problems, to effectively identify beam-like structures using the PINNs framework. At the moment, there are few studies on this topic in the literature. Kapoor et al. [19] applied PINNs to solve PDEs of the Euler–Bernoulli and Timoshenko models. The formulation adopted by the authors involves expressing the PDEs of the beam models as time-dependent, thereby increasing the analysis dimension and potentially necessitating specific vibration tests to acquire the appropriate response for the training stages—significantly augmenting the volume of required training data. Faroughi et al. [20], in turn, discussed a forward application of PINNs to predict the output of non-uniform beam structures under different load scenarios, evaluating the impact of the order of derivation in the training process. Meanwhile, Wu et al. [21] employed PINNs to address inverse problems involving linear elastic and hyperelastic materials in beam-like structures, examining the influence of hard and soft constraints in the solution accuracy. None of the studies within the beam-type structures context validated the methodologies with experimental data, nor did they consider the presence of damping in their models. The 4th order PDE of the constitutive model is probably one of the reasons that explains the lack of works reporting PINN applications on such structures. Additionally, in the context of system identification, this 4th order derivative is known to be responsible for the high sensitivity of the procedure to noise, which is inherent to the experimental data used as inputs. Filtering procedures are required to obtain consistent results [22,23].

In this work, a novel approach is proposed, employing the PINN framework to identify the transverse displacement and complex elastic modulus of a beam structure through a loss function that incorporates the PDE of the Euler–Bernoulli model (physics informed component) and the observed data from the system under interest (data-driven component). In particular, the transverse displacement and the complex elastic modulus are captured by separate neural networks, proving advantageous, especially with non-uniform parameters. Although this work considers elastic modulus as the unknown, other structural parameters could have been taken into account. The displacement output from the neural network is injected into the PDE, such that the higher-order derivatives are obtained through automatic differentiation. The approach is somehow close to the technique recently discussed in [17], which is not validated using experimental data. The main contribution lies in using the PINN framework to identify the elastic modulus, being homogeneous or space-dependent, together with the system damping; to the best of the authors' knowledge, this is a pioneering work using the PINN approach to estimate damping in structural dynamics. Moreover, this work's novelty is reinforced by formulating the inverse problem directly in the frequency domain, which simplifies the analysis and allows the estimation of damping, especially for homogeneous domains This involves transforming the PDE into the frequency domain, eliminating the need for a vast amount of training data associated with the time domain, as this is the case in [17]. Operating directly in the frequency domain simplifies the analysis, requiring knowledge of just one operating frequency. Consequently, the PDE comprises two coupled equations — one for the real part and another for the imaginary part. Two scenarios are considered: when the elastic modulus is non-uniform and real, and when it is complex to account for damping effects. Experimental validation is conducted for the latter scenario. Notably, the approach reveals that explicit knowledge of boundary conditions is unnecessary; only local conditions within the analysis zone need attention, which is a strong benefit of this kind of technique.

Towards this background, this paper is organized as follows: Section 2 presents a high-level overview of the PINN. Section 3, in turn, presents the Euler–Bernoulli model, deriving the PDEs used to compose the physics-informed loss function, thus including a physics learning bias in the training procedure of neural networks. Specifically, this work proposes independent neural networks to approximate the displacement of the beams, as well as the parameters of interest in an inverse problem configuration. Then, Section 4 addresses numerical applications of the proposed framework, passing through the case where the elastic modulus is considered non-uniform, and then addressing its complex formulation that takes damping into account. Section 5 validates the methodology considering an experimental application involving a cantilever steel beam, in which the elastic modulus and damping are identified through the PINN framework. Finally, Section 6 presents the concluding remarks and proposes the next steps for future work.

2. Physics informed neural networks (PINNs)

The aim of this section is to provide a high-level overview of neural networks, specifically emphasizing the incorporation of physics-informed terms in the loss function, distinguishing them as physics-informed networks. Physics-related terms can be derived from various sources like the stress tensor, energy balance, etc. However, this section primarily concentrates on presenting a loss function derived from partial differential equations, see *e.g.* [5].

A neural network is a mathematical model that represents a linear composition of nonlinear functions. In the context of this work, the fully connected architecture is considered, a common and versatile structure in neural networks. This architecture consists of interconnected layers of neurons, with depth of L hidden layers and width of N_ℓ neurons in each layer, where each neuron in a layer is connected to every neuron in the subsequent layer by an activation function $\sigma(\cdot)$. Let $\boldsymbol{x} \in \mathbb{R}^D$ be the input, and $y \in \mathbb{R}$ a target function. Thus, the approximated output $\hat{y}$ of the neural network is given by

$$y(\boldsymbol{x}) \approx \hat{y}(\boldsymbol{x}) = \mathcal{H}^{(L)} \circ \mathcal{H}^{(L-1)} \circ \ldots \mathcal{H}^{(1)} \circ \mathcal{H}^{(0)}(\boldsymbol{x}), \text{ for } \boldsymbol{x} \in \mathbb{R}^D, \tag{1}$$

in which the function $\mathcal{H}$ is written as

$$\begin{aligned}
\boldsymbol{x}_0 &:= \mathcal{H}^{(0)}(\boldsymbol{x}), \\
\boldsymbol{x}_\ell &:= \mathcal{H}^{(\ell)}(\boldsymbol{x}_{l-1}) := \sigma(\boldsymbol{W}_\ell \boldsymbol{x}_{\ell-1} + \boldsymbol{b}_\ell), \quad 0 < \ell < L, \\
\hat{y}(\boldsymbol{x}; \boldsymbol{W}, \boldsymbol{b}) &:= \mathcal{H}^{(L)}(\boldsymbol{x}_{L-1}) := \boldsymbol{W}_L \boldsymbol{x}_{L-1} + \boldsymbol{b}_L.
\end{aligned} \tag{2}$$

In above equations, $\boldsymbol{W}_\ell \in \mathbb{R}^{N_\ell \times N_{\ell-1}}$ and $\boldsymbol{b}_\ell \in \mathbb{R}^{N_\ell}$ are the weights and biases of the ℓth-layer, respectively, and σ denotes the activation function that acts component-wise. Since the problem addressed in this work concerns a regression, the output layer comes directly from the weighting with bias of the previous neurons. The output of the neural network $\hat{y}(\boldsymbol{x}) = \hat{y}(\boldsymbol{x}; \boldsymbol{W}, \boldsymbol{b})$ depends implicitly on the choice of the weights and biases — to simplify the notation, $\theta := \{\boldsymbol{W}, \boldsymbol{b}\}$.

In a classic data-driven approach using neural networks, for a given architecture with L layers and N neurons, the problem corresponds to a process of minimizing an objective function that measures the distance between the neural network's output and the observed target using a proper discrepancy metric. This loss function related to data observations, named $\mathcal{R}_d$, can be written as

$$\mathcal{R}_d(\hat{y}(\cdot; \theta)) = \frac{1}{N} \sum_{n=1}^{N} |\hat{y}(\boldsymbol{x}_n; \theta) - y(\boldsymbol{x}_n)|^2, \tag{3}$$

where N corresponds to the number of observed data samples. This loss corresponds to the (squared) discrete L2 norm.

In a physics-informed context, the loss function incorporates, in addition to the term $\mathcal{R}_d$ related to the data, residual terms associated with the system under analysis. Consider the following general PDE of a mechanical system in steady state regime of motion,

$$\mathcal{D}^\alpha(y(\boldsymbol{x}); \lambda) = f(\boldsymbol{x}), \quad \boldsymbol{x} \in \Omega, \tag{4}$$

where $D^{\alpha}(\bullet)$ corresponds to the differential operator of order α, λ represents the model parameters, and $f(\boldsymbol{x})$ is the external excitation.

From the perspective of forward analysis, the PDE is integrated as a component of the loss function to create a learning bias during training process, i.e., the approximated solution obtained by neural networks ideally has to satisfy pointwise the PDE. Therefore, a loss function related to the PDE residual, commonly known as the physically-informed loss and named $\mathcal{R}_f$, is typically defined as

$$\mathcal{R}_f(\hat{y}(\cdot;\boldsymbol{\theta})) = \frac{1}{M}\sum_{m=1}^{M}|D^{\alpha}(\hat{y}(\boldsymbol{x}_m;\boldsymbol{\theta});\lambda) - f(\boldsymbol{x}_m)|^2, \tag{5}$$

where M corresponds to the number of the discretization points (also known as collocation points) linked to the PDE residual.

The above equation involves partially deriving the neural network's output according to the order α. These gradients can be solved by using the automatic differentiation [24], which considers symbolic rules for differentiation instead of numerical differentiation, being a widely used tool in deep learning. Subsequently, the total loss function is defined as

$$\mathcal{L} = \gamma\mathcal{R}_d(\boldsymbol{\theta}) + \beta\mathcal{R}_f(\boldsymbol{\theta}), \tag{6}$$

where γ and β are weights that ensure the importance of the terms in the composite loss function. Additional terms could be included in the composition of the loss function, such as initial conditions and boundary conditions (e.g., Dirichlet-type), which are imposed on the problem as weakly constraints.

From a typical perspective of inverse problems, point observations are utilized to formulate the PDE, and then the model parameters are derived. However, depending on the application domain, this simple idea might present limitations, specially when handling experimental data. Numerically estimating high-order partial derivatives, particularly in contexts involving spatial measurements of quantities of interest (e.g., displacement field) along a structure, introduces a source of systematic error that can impact the parameter estimation procedure in inverse problem applications. In addition, experimental campaigns can also have limited access to the term on the right-hand side of Eq. (4), to the boundary conditions that may be uncertain, and the measurements are often corrupted by noise effects. Within the PINN framework, the objective is two-fold: identify the solution $y(x)$ and, at the same time, the model parameters λ that minimize the loss function. More specifically, this work presents a framework in which these parameters can be considered as space-dependent, i.e. $\lambda = \lambda(\boldsymbol{x})$. Therefore, an additional neural network is proposed to approximate the parameters of interest $\lambda(\boldsymbol{x}) \approx \hat{\lambda}(\boldsymbol{x};\boldsymbol{\xi})$, where $\boldsymbol{\xi}$ represents the parameters of this new neural network. Hence, the physics informed loss function used in this work is rewritten as

$$\mathcal{R}_f(\hat{y}(\cdot;\boldsymbol{\theta}),\hat{\lambda}(\cdot;\boldsymbol{\xi})) = \frac{1}{M}\sum_{m=1}^{M}|D^{\alpha}(\hat{y}(\boldsymbol{x}_m;\boldsymbol{\theta});\hat{\lambda}(\boldsymbol{x}_m;\boldsymbol{\xi})) - f(\boldsymbol{x}_m)|^2. \tag{7}$$

Several domains within structural dynamics can benefit from this physics-informed approach, particularly in constructing maps of mechanical properties for structures or objects. Additionally, it offers the potential to estimate characteristics related to discontinuities, e.g. structural health monitoring applications for locating and quantifying damage, as in the case of cracks. However, it is crucial to acknowledge that this approach has recently been used in the literature in different contexts (see [18]). This work addresses the elasticity and damping properties of beam-like structures; and the problem is introduced in the next section.

3. On the partial differential equation: The Euler–Bernoulli model

Beam problems may be viewed as basic examples, but their significance in proposing and validating new methodologies should not be underestimated. This section introduces the PDE of the Euler–Bernoulli model. Subsequently, the formulation is adapted for the two scenarios investigated in this work: first, considering a non-uniform elastic modulus,[1] and then, incorporating the complex elastic modulus — this enables the assessment of dissipative effects by including terms related to both the real and imaginary components in the physics-informed loss function.

Consider the schematic representation of the cantilever beam presented in Fig. 1. The equation of motion of the Euler–Bernoulli model considering a harmonic motion at angular frequency ω is given by [25]

$$\frac{\partial^2}{\partial x^2}\left(E(x)I\frac{\partial^2 w(x,\omega)}{\partial x^2}\right) - \rho S\omega^2 w(x,\omega) = f(x,\omega), \tag{8}$$

where $w(x)$ is the transversal deflection of the beam; I [m^4], and S [m^2] are the geometrical properties, corresponding to the flexural inertia $\frac{b \times h^3}{12}$ and the area of the cross section $h \times b$ of the beam, respectively; $E(x)$ [GPa $\equiv 1 \times 10^9$ N/m^2] and ρ [kg/m^3] are the material properties, representing the elastic modulus and density, respectively; $f(x)$ is the applied loading. In this formulation, $E(x)$ can be considered as dependent on the x position of the beam. The PDE (8) is a strong formulation, meaning the equation must be satisfied over the entire domain of the structure. This study focuses on a specific region of the beam where no external force is applied, to recover the parameters of the equation. An engineering field that can benefit from this formulation is non-destructive analysis of structures, where output-only methodologies are welcome. Once the model parameters are obtained, the proposed approach may be extended to force identification, which constitutes a perspective to this work.

---

[1] All along this paper, we refer to E as elastic modulus, however it is considered as a complex number, its imaginary part being non null if losses are considered in the model. Additionally, it may be frequency-dependent, as all computations are performed in the frequency domain for a specific frequency. It hence applies to viscoelastic modulus as well.
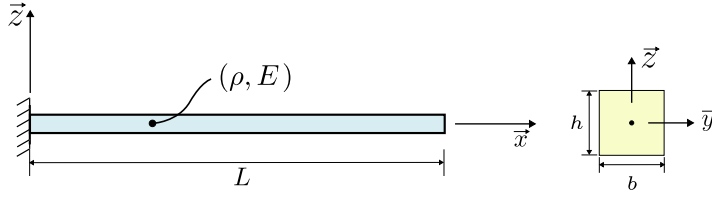
Fig. 1. Schematic representation of the bending beam; L, b, and h stand for the length, width and thickness respectively.

Assuming that the elastic modulus is non-uniform and differentiable along with the length x of the beam, and the material is linear viscoelastic, one has $E(x) = E_r(x) + jE_i(x)$, and the transversal deflection of the beam is also a complex quantity $w(x, \omega) = w_r(x, \omega) + jw_i(x, \omega)$.[2] The loss factor η, which characterizes the material damping to be identified, is given by:

$$\eta = \frac{E_i}{E_r}. \tag{9}$$

Taking into account the real and imaginary parts of each term, Eq. (8) yields

$$\frac{I}{\rho S \omega^2} \left((E_r + jE_i) \left(\frac{\partial^4 w_r}{\partial x^4} + j\frac{\partial^4 w_i}{\partial x^4} \right) + \left(\frac{\partial^2 E_r}{\partial x^2} + j\frac{\partial^2 E_i}{\partial x^2} \right) \left(\frac{\partial^2 w_r}{\partial x^2} + j\frac{\partial^2 w_i}{\partial x^2} \right) \right) +$$
$$\frac{2I}{\rho S \omega^2} \left(\frac{\partial E_r}{\partial x} + j\frac{\partial E_i}{\partial x} \right) \left(\frac{\partial^3 w_r}{\partial x^3} + j\frac{\partial^3 w_i}{\partial x^3} \right) - (w_r + jw_i) = 0. \tag{10}$$

In a scenario aiming to determine the deflection of the beam and the non-constant elastic modulus, both of which are complex quantities, it is advisable to propose a problem-solving framework that includes a total of four independent neural networks, i.e. $\hat{w}_r \approx w_r$, $\hat{w}_i \approx w_i$, $\hat{E}_r \approx E_r$ and $\hat{E}_i \approx E_i$, with parameters $\theta_1, \theta_2, \xi_1$ and ξ_2, respectively. For simplifying notation, $\mathcal{W} := (\hat{w}_r(\cdot; \theta_1), \hat{w}_i(\cdot; \theta_2))$, and $\mathcal{E} := (\hat{E}_r(\cdot; \xi_1), \hat{E}_i(\cdot; \xi_2))$. Therefore, Eq. (10) can thus be decomposed into two parts that form the physics-informed loss function, one considering only the real terms:

$$\mathcal{R}_f^{(\text{real})}(\mathcal{W}, \mathcal{E}) = \frac{1}{M} \sum_{m=1}^{M} \left| \frac{I}{\rho S \omega^2} \left(\hat{E}_{r,m} \frac{\partial^4 \hat{w}_{r,m}}{\partial x^4} - \hat{E}_{i,m} \frac{\partial^4 \hat{w}_{i,m}}{\partial x^4} + \frac{\partial^2 \hat{E}_{r,m}}{\partial x^2} \frac{\partial^2 \hat{w}_{r,m}}{\partial x^2} - \frac{\partial^2 \hat{E}_{i,m}}{\partial x^2} \frac{\partial^2 \hat{w}_{i,m}}{\partial x^2} \right) + \right.$$
$$\left. \frac{2I}{\rho S \omega^2} \left(\frac{\partial \hat{E}_{r,m}}{\partial x} \frac{\partial^3 \hat{w}_{r,m}}{\partial x^3} - \frac{\partial \hat{E}_{i,m}}{\partial x} \frac{\partial^3 \hat{w}_{i,m}}{\partial x^3} \right) - \hat{w}_{r,m} \right|^2, \tag{11}$$

and other involving the imaginary terms:

$$\mathcal{R}_f^{(\text{imag})}(\mathcal{W}, \mathcal{E}) = \frac{1}{M} \sum_{m=1}^{M} \left| \frac{I}{\rho S \omega^2} \left(\hat{E}_{r,m} \frac{\partial^4 \hat{w}_{i,m}}{\partial x^4} + \hat{E}_i \frac{\partial^4 \hat{w}_{r,m}}{\partial x^4} + \frac{\partial^2 \hat{E}_{r,m}}{\partial x^2} \frac{\partial^2 \hat{w}_{i,m}}{\partial x^2} + \frac{\partial^2 \hat{E}_{i,m}}{\partial x^2} \frac{\partial^2 \hat{w}_{r,m}}{\partial x^2} \right) + \right.$$
$$\left. \frac{2I}{\rho S \omega^2} \left(\frac{\partial \hat{E}_{r,m}}{\partial x} \frac{\partial^3 \hat{w}_{i,m}}{\partial x^3} + \frac{\partial \hat{E}_{i,m}}{\partial x} \frac{\partial^3 \hat{w}_{r,m}}{\partial x^3} \right) - \hat{w}_{i,m} \right|^2. \tag{12}$$

In a similar manner, residual terms related to data observations can also be defined concerning real and imaginary components, resulting in

$$\mathcal{R}_d(\mathcal{W}) = \frac{1}{N_x} \sum_{n=1}^{N_x} \left| \hat{w}_{r,n} - w_{r,n} \right|^2 + \frac{1}{N_x} \sum_{n=1}^{N_x} \left| \hat{w}_{i,n} - w_{i,n} \right|^2, \tag{13}$$

where $\hat{w}_{\cdot,n}$ and $w_{\cdot,n}$ are the displacements evaluated at x_n, and N_x corresponds to the number of points used for displacement measurements along the length of the beam. Thus, the loss function is given by

$$\mathcal{L} = \gamma \mathcal{R}_d(\mathcal{W}) + \beta \left(\mathcal{R}_f^{(\text{real})}(\mathcal{W}, \mathcal{E}) + \mathcal{R}_f^{(\text{imag})}(\mathcal{W}, \mathcal{E}) \right), \tag{14}$$

where (γ, β) are determined heuristically in this work, being $\gamma = 1$ and $\beta = 10^{-2}$. These parameters serve as scalar values that scale the terms of the loss function, ensuring that the data-driven loss and the physics-informed loss are of comparable magnitude. The physics-informed loss preserves physical constraints due to the strong formulation adopted.

Another aspect worth mentioning about the methodology presented in this work is that the predicted outputs of neural networks are normalized. More specifically, the outputs $\hat{E}_r$ and $\hat{E}_i$ are multiplied by a factor 10^{11} when inserted in the loss function, whereas the outputs of the networks for predicting $\hat{w}_r$ and $\hat{w}_i$ are normalized between -1 and 1, but when included in the loss function, both are multiplied by the factor used in the normalization. This procedure is carried out to reduce instability problems during the network training phase.

Furthermore, these equations can be simplified for different analysis scenarios, as summarized in Table 1. Out of the four possible scenarios, this work specifically addresses three of them, while the remaining case with non-uniform $\hat{E}_i$ will be the focus of ongoing

---

[2] To simplify the notation, the dependence of the displacement and the complex elastic modulus on (x, ω) are omitted from now on.

Table 1

Possible cases of analysis involving the quantities of interest. ✓indicates cases addressed in this work.

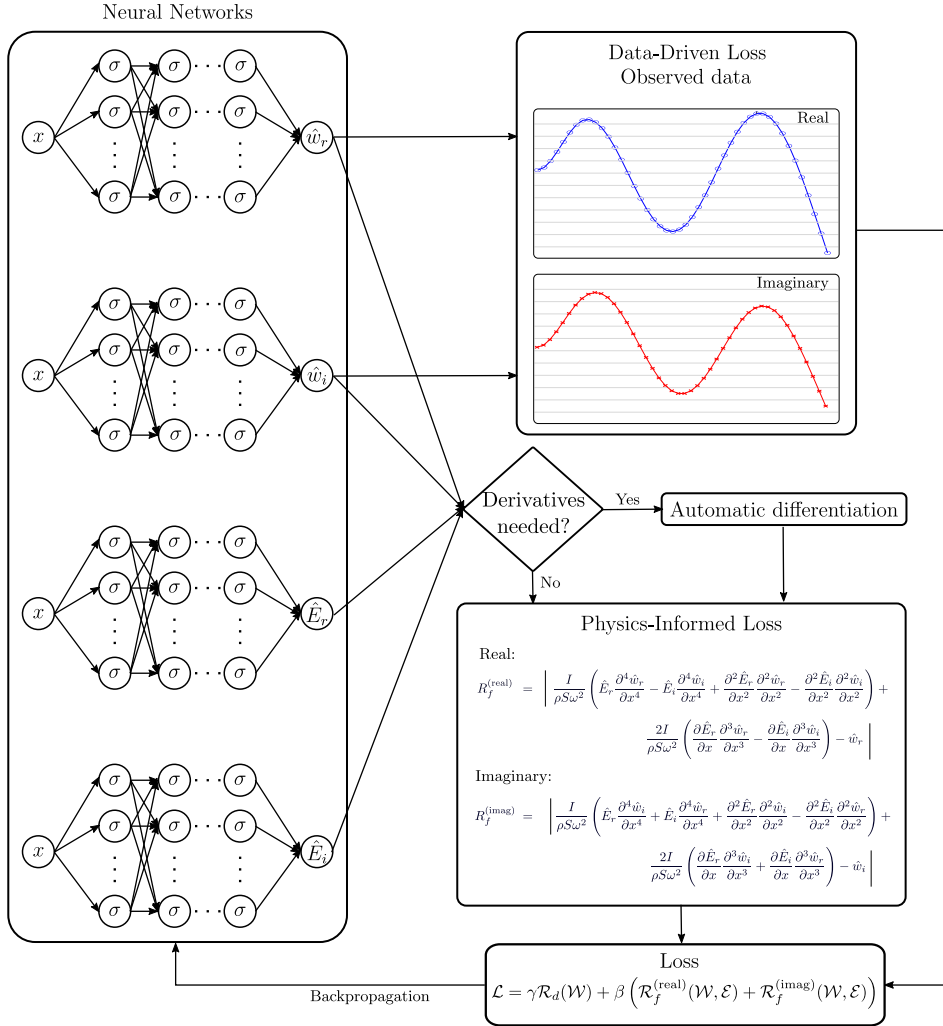| Quantities of interest	$\hat{E}_r$	$\hat{E}_i$
Uniform	✓	✓
Non-uniform	✓	x

Fig. 2. Schematic representation of the algorithm framework involving four independent neural networks. From this general framework, simplifications can be made depending on the complexity of the problem being analyzed.

works. Among the three cases of interest, two of them do not involve a complex E, meaning that the user is not interested in the damping identification. This is feasible only for very low damping levels. In these scenarios, the loss function is constructed without considering $\hat{w}_i$ and $\hat{E}_i$, taking into account only the contributions from the real components of Eqs. (11) and (13).

Fig. 2 introduces a general schematic representation of the proposed identification framework. From this representation, various simplifications can be implemented, such as suppressing the neural networks related to the imaginary terms. Notably, in cases where the elastic modulus and damping are uniform, they can be represented by constant values. In this scenario, instead of using neural networks with multiple layers and several neurons, a simplified approach involves using just one bias for each parameter. Then, biases can be optimized along with the neural network parameters of $\hat{w}_r$ and $\hat{w}_i$ during the backpropagation process.

Table 2

Geometrical and material properties used to construct the FE model.

FE Parameters	Specifications
Length [m]	0.70
Width [m]	0.02
Thickness [m]	0.01
Nominal elastic modulus: E [GPa]	70
Structural damping: η [%]	1
Poisson ratio	0.30
Density [kg/m^3]	2700
Number of elements	100
Larger element size [m]	7×10^{-3}
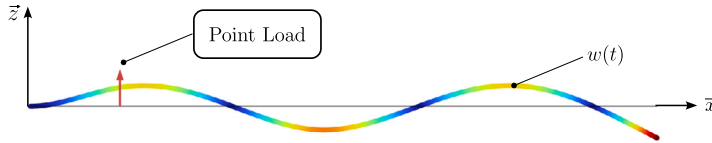
Fig. 3. Displacement field of the FE model for a harmonic excitation force of 1 N at 500 Hz.

4. Numerical applications

This section presents the identification procedure introduced in the previous section through numerical studies. In this algorithm, model parameters and the displacement of the beam are described by the PINN framework. To facilitate this demonstration, a FE model is constructed to generate data for the numerical experiments. The model represents a clamped-free aluminum beam with dimensions of $700 \times 20 \times 10$ mm, where an excitation force is applied close to the clamped-end, at 100 mm. Eq. (8), which represents the physics-informed term in the neural networks' loss, must be satisfied across the entire domain of the structure. However only a specific region (between 240 mm and 620 mm) is considered, excluding the zone where the force is applied. The FE model, built using COMSOL software, employs the beam model with a nominal elastic modulus of 70 GPa, a density of 2700 kg/m^3, and structural damping of 1%. Additionally, a frequency study is conducted on the structure at 500 Hz, with an excitation amplitude of 1 N, to generate the displacement field data used in the training steps, particularly in the data-driven loss function. All this information and supplementary data are summarized in Table 2, and Fig. 3 depicts the displacement field of FE model at 500 Hz. It is important to point out that the excitation frequency is between the 4th and 5th bending modes, corresponding to 292.98 and 560.39 Hz, respectively.

The excitation amplitude of 1 *N* is chosen arbitrarily since, in linear structural dynamics, the amplitude of the response is directly proportional to the excitation force, while the shape of the response remains unaffected. The displacements are normalized to the maximum response amplitude, meaning that the specific choice of force amplitude does not influence the results. Similarly, the excitation frequency, selected near one of the eigenfrequencies, is chosen to avoid significant noise interference, though other frequencies could have been used.

In the continuation of this section, two cases are addressed: the first involving non-uniform elastic modulus without the presence of damping, and the second considering complex and uniform elastic modulus along the length of the beam. These applications enable one to explore the different architectures of the identification framework introduced in Section 3.

4.1. Case 1: Non-uniform elastic modulus

This case serves to present a simplification of the architecture of the identification framework introduced in Fig. 2, where the terms related to the imaginary components are not considered in the formulation, since there is no presence of damping. In the scenario where a non-uniform elastic modulus is explored, a spatially varying elastic modulus is introduced to emulate the heterogeneous mechanical properties of the beam. The non-uniform distribution is defined by

$$E(x) = \left(1 - \frac{c}{u\sqrt{2\pi}} \cdot \exp\left(-\frac{1}{2}\left(\frac{x-m}{u}\right)^2\right)\right) \times E, \tag{15}$$

where the parameters c, u and m where set to 0.025, 0.030 and 0.320, respectively.

Therefore, two independent neural networks are proposed — one dedicated to capturing displacement and the other to modeling the elastic modulus. Both networks have a similar architecture, comprising 9 hidden layers, each with 10 neurons. The hyperbolic tangent function is employed as the activation function throughout this work. Each network includes input and output layers, and despite their different purposes, both networks take the x-coordinate of the beam as input. The Xavier Glorot method [26] initializes the neural networks, and the optimization process employs the Adam optimizer [27]. This initialization strategy, as well as the optimizer, are used throughout this work. For learning rate updating, the OneCycle policy is adopted [28], achieving a maximum
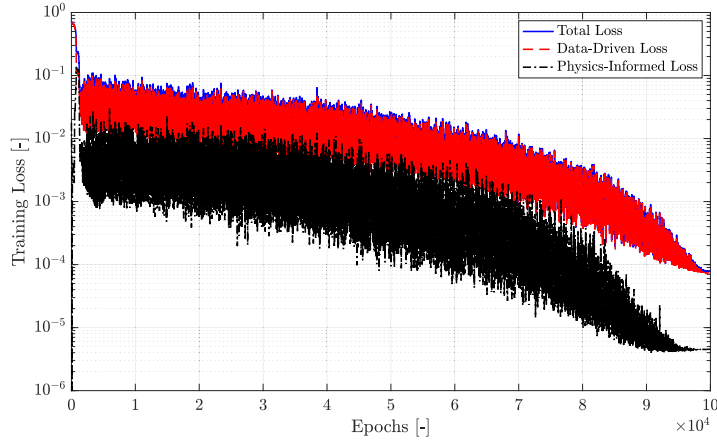
Fig. 4. Evolution of the different loss functions for a non-uniform elastic modulus over the number of epochs.

(a) Displacement, with NMSE=0.03 %

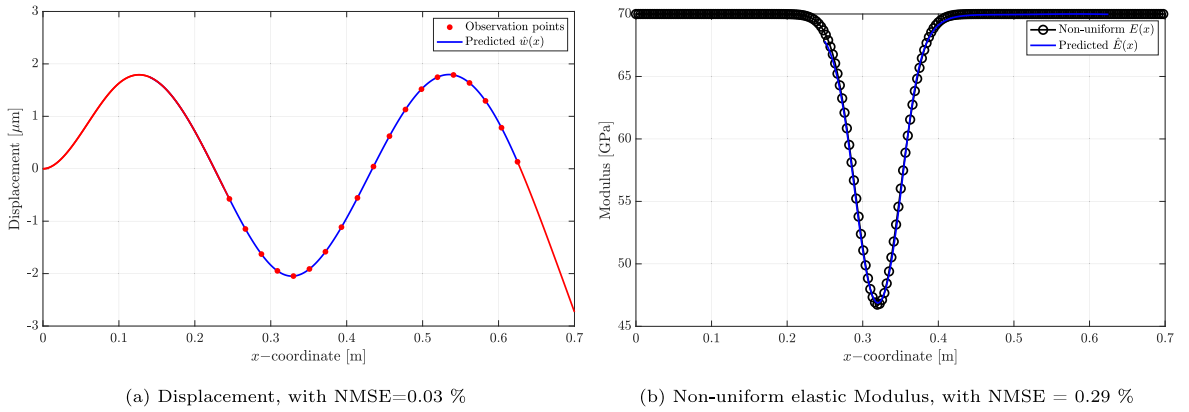(b) Non-uniform elastic Modulus, with NMSE = 0.29 %

Fig. 5. Predictions made by neural networks on $\hat{w}(x)$ and non-uniform $\hat{E}(x)$ compared to data generated by numerical experiments. In Fig. 5(a), ▬ represents the neural network's predictions on the generalization points, • represents observation points within the zone of interest, and ▬ is the numerical points generated by COMSOL across the remaining domain not utilized by the algorithm. In Fig. 5(b), ▬ represents the neural network's predictions, whereas in Fig. 5(b), ▬○▬ is the non-uniform $\hat{E}(x)$.

learning rate of 0.005 at 10% of the total number of epochs — considered as 10^5. To construct the PDE, 1024 collocation points are uniformly generated along the beam's area of interest (see Table 2), whereas 20 observation points are used for the data-driven loss.

Fig. 4 depicts the evolution of the total loss, including the components related to observation points and the PDE. At the beginning of the training process, the physics-informed loss rapidly achieved values around 10^{-2}. This initial behavior can be attributed to the optimizer exploring the solution space and initially converging to the trivial solution, which minimizes the loss function as well. In this application, and for the gamma and beta values used, it can be seen that the loss function related to the observation data drives the progress of the total loss function. In the last iteration, the total loss reached a value of 7.80×10^{-5}.

In Fig. 5, the (a) predicted displacement and (b) the predicted non-uniform elastic modulus are presented. It is noteworthy that both outputs of the PINNs, $\hat{w}(x)$ and $\hat{E}(x)$, demonstrated precision in predicting the numerical experiment, with a normalized mean square error (NMSE) of 0.03% for the displacement – estimated considering 200 generalization points not used during the training stages – and 0.29% for the non-uniform elastic modulus. These results underscore the potential of the proposed methodology in accurately capturing the intricate behavior of beam-like structures with varying properties.

4.2. Case 2: Complex modulus

In this example, a uniform structural damping of 1% and a uniform elastic modulus of 70 GPa along the beam are assumed (see Table 2). To assess the predictive capacity of the algorithm, two scenarios are considered. First, both the real and imaginary parts of the complex modulus are approximated by neural networks with multiple hidden layers and neurons. Then, since η and E_r are uniform throughout the structure, the parameters $\hat{E}_r$ and $\hat{E}_i$ are treated as constant values. The independent neural networks are simplified to single biases, optimized during the backpropagation stage of the training process.
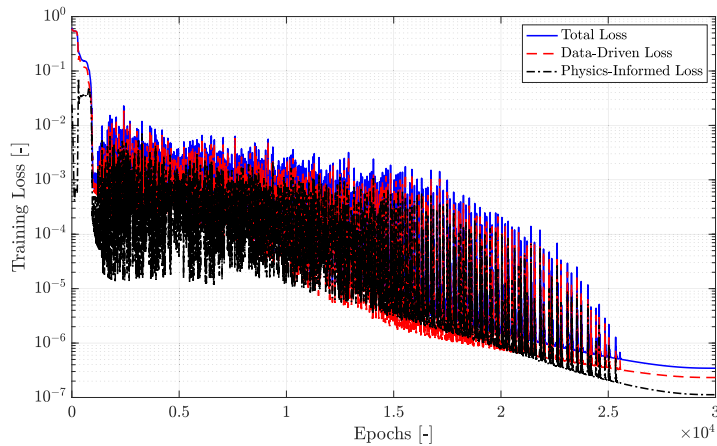
Fig. 6. Evolution of the different loss functions for an uniform elastic modulus and damping over the number of epochs.

(a) Real displacement, with NMSE=0.06 %

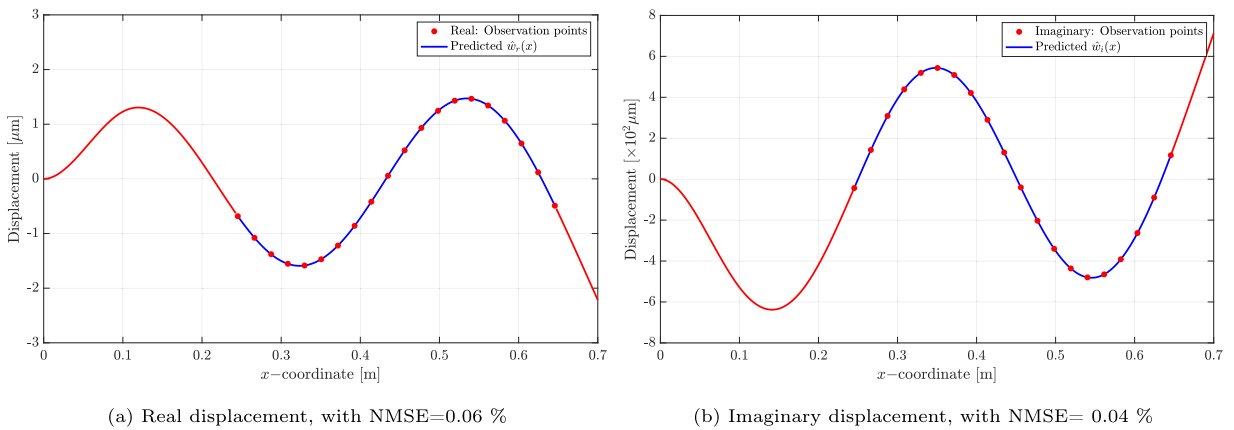(b) Imaginary displacement, with NMSE= 0.04 %

Fig. 7. Predictions made by neural networks on the real $\hat{w}_r(x)$ and imaginary $\hat{w}_i(x)$ components of the beam displacement compared to data generated by numerical experiments. — represents the neural network's predictions on the generalization points, whereas • represents observation points within the zone of interest; — is the numerical points generated by COMSOL across the remaining domain not utilized by the algorithm.

In the first scenario, four independent neural networks are introduced to capture the real and imaginary parts of the displacement and elastic modulus. All four networks have the same architecture, featuring 9 hidden layers, each with 10 neurons, and take the x-coordinate as input. The OneCycle strategy is employed for learning rate scheduling, with a maximum learning rate set to 0.005 at 10% of the total number of epochs, which is defined as 3×10^4. To construct the PDE, 1024 collocation points are uniformly generated along the beam's area of interest (see Table 2), whereas 20 observation points are used for the data-driven loss.

Fig. 6 presents the training loss as function of the number of epochs. In the last iteration, the total loss reached a value of 3.45×10^{-7}. For the uniform elastic modulus, the algorithm requires a smaller number of epochs to reach convergence. In Fig. 7, both the (a) real and (b) imaginary parts of displacement from numerical experiments are compared with the predictions obtained from the neural networks. For each part of the complex displacement, the NMSE is estimated using 200 generalization points not utilized during the training process, resulting in an error below 0.1%.

Fig. 8 presents the predictions of $\hat{E}_r$ and $\hat{\eta}$ made by the PINN framework, comparing them with the material parameters used in the numerical experiments. Although η is not directly used as an output of the neural network, Eq. (9) enables its direct derivation from the outputs E_r and E_i. Consequently, for ease of interpretation, the parameter η is presented. Despite some deviations from the exact values used in the numerical model, the most notable error is observed in predicting damping, with an NMSE of 0.48%. Nonetheless, these results indicate that the model effectively predicts numerical data in good agreement.

In practical applications, the observation points represent the number of sensors available for taking measurements. Table 3 provides a comparison of the neural networks' predictions for the parameters $\hat{E}_r$ and η, as a function of the number of observation points used in the data-driven loss function. To ensure a fair analysis, the same network architecture and training configurations were used across all cases. The results indicate that a lower number of observation points leads to a decline in the quality of the model's predictions. This is because displacement estimates directly affect the fourth-order derivative, which is critical for calculating the physical loss function. Consequently, inaccuracies in this estimate may influence the predictions of $\hat{E}_r$ and η as these parameters can act in a compensatory manner to minimize the PDE loss across the entire analysis interval.

(a) Normalized mean square error: 0.03 %

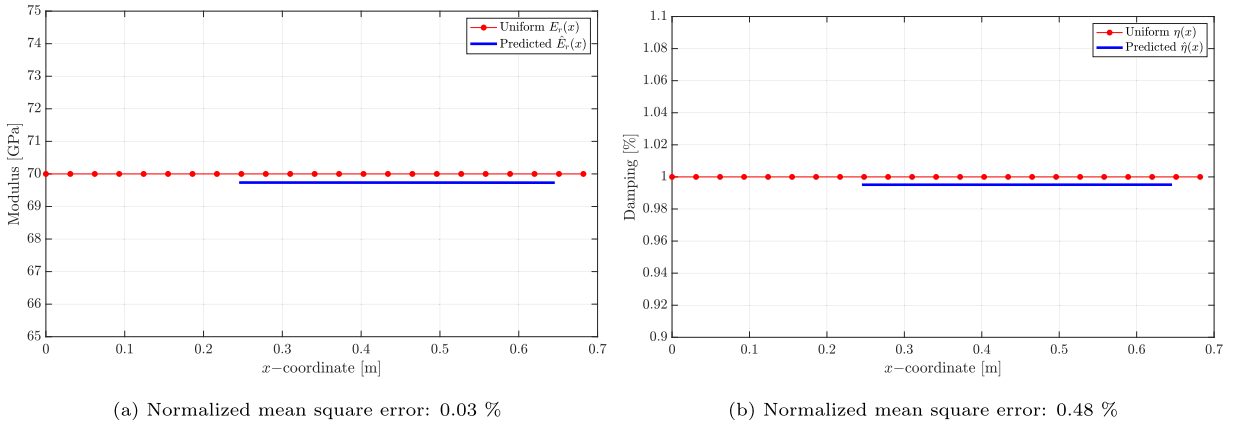(b) Normalized mean square error: 0.48 %

Fig. 8. Predictions made by neural networks on the elastic modulus $\hat{E}_r(x)$ and damping $\hat{\eta}(x)$ compared to material properties utilized by numerical experiments. ━ represents the neural network's predictions, whereas • represents the behavior of the material properties across the beam.
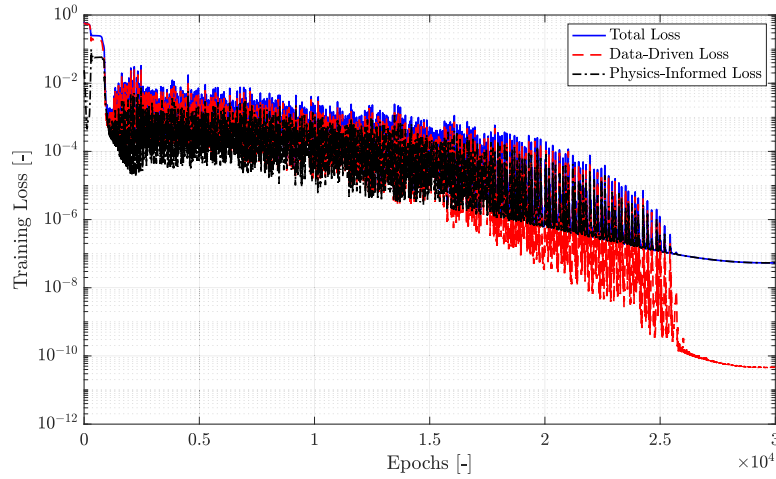

Fig. 9. Evolution of the different loss functions, when the neural networks of $\hat{E}_r$ and $\hat{E}_r$ are composed by a single biases, over the number of epochs.

Table 3
NMSE for the predictions of $\hat{E}_r$ and $\hat{\eta}$ as a function of the number of observation points used in the data-driven loss function.

Observation points	NMSE ($\hat{E}_r$, E_r) [%]	NMSE ($\hat{\eta}$, η) [%]
10	0.42	1.18
20	0.03	0.48
40	0.005	0.03

In the second scenario, the architecture of the $\hat{E}_r$ and $\hat{E}_i$ neural networks is simplified to one bias each. These biases are optimized simultaneously with the parameters of the neural networks for $\hat{w}_r(x)$ and $\hat{w}_i(x)$. Employing the same training strategy introduced earlier and integrating $\hat{E}_r$ and $\hat{E}_i$ into the parameters for optimization, Fig. 9 illustrates the total training loss, reaching values of 5.37×10^{-8} for the total loss in the final iteration.

Finally, Fig. 10 presents the evolution of $\hat{E}_r$ and also of η as a function of the number of epochs used in training. Both values are initialized far from the optimal values, being 110 GPa for $\hat{E}_r$ and the equivalent of 10% loss factor for $\hat{\eta}$. Additionally, from 25000 epochs onward, both values reach convergence. In addition to offering a relatively lower total loss value than that obtained for the case with $\hat{E}_r$ and $\hat{E}_i$ described as independent neural networks, convergence of the algorithm is achieved with fewer epochs. This outcome is expected, considering that the optimizer has to deal with a reduced number of parameters to optimize. These results indicate that, when dealing with uniformly distributed E_r and η, this strategy delivers the best trade-off between predictive performance and convergence.

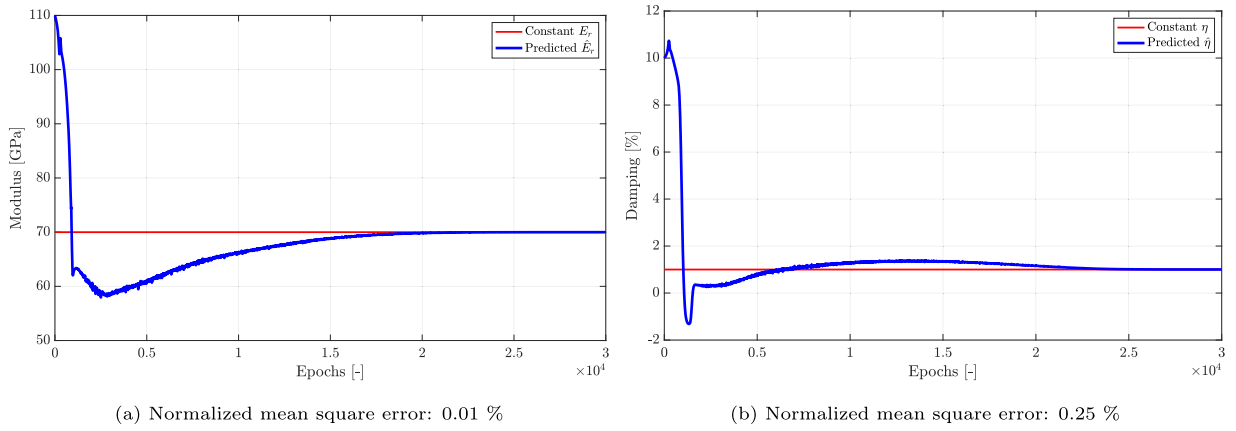(a) Normalized mean square error: 0.01 %   (b) Normalized mean square error: 0.25 %

Fig. 10. Evolution of the $\hat{E}_r$ and $\hat{\eta}$ parameters over the training iterations. ▬ represents evolution through iterations, and ▬ is the reference value.
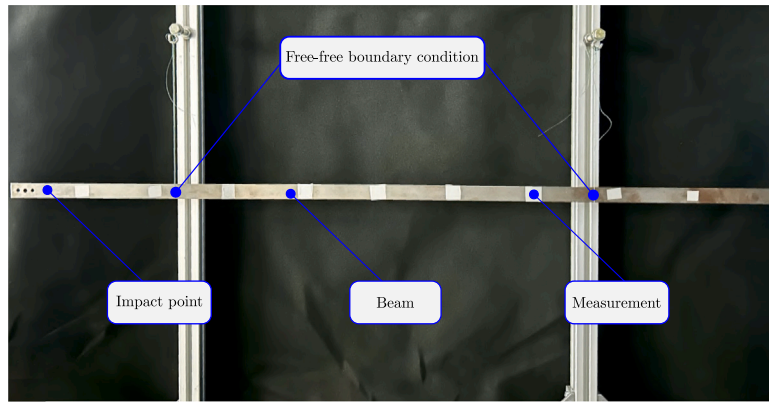

Fig. 11. Test rig in free-free boundary conditions.

5. Experimental case study — bending beam

5.1. Description of the experimental setup

The objective of this section is to demonstrate the applicability of the methodology proposed in this work, based on the use of PINNs, to solve inverse problems in beam-like structures when considering experimental data. The target beam is made of steel, with dimensions of $1000 \times 20 \times 2$ [mm], and density of 7870 kg/m^{-3}. To first characterize the vibrational behavior of this structure in order to obtain some reference values, impact hammer tests are conducted under free-free boundary conditions. Fig. 11 illustrates the testing structure. Excitation is performed at 55 mm, whereas the measurement is performed at 700 mm from the total lenght of the structure. The impact is carried out with a Modal Shop hammer model 086C01, and the velocity at the is measured using a PSV-500 Xtra laser. A sampling frequency of 16 kHz is considered, with an acquisition time of 8 seconds until the free response dies out. Fig. 12 illustrates the mobility of the structure, computed from the H_1 estimator, 4 averaging measurements, and exponentially decaying window. These results highlight several bending modes within a frequency range from 10 to 1000 Hz.

These experimental results are used to calibrate a FE model of the structure, considering the free-free boundary condition. The model is created in COMSOL. The quantities related to geometry and density can be directly inferred with traditional experimental apparatus, being the elastic modulus the only parameter identified during the model updating procedure. This calibration step identified a elastic modulus of 205 GPa. In Table 4, the first eleven resonant frequencies of the bending motion of the FE model are compared with the ones obtained experimentally. For each vibration mode of the structure under free-free boundary conditions, the experimental modal damping ratio ζ was identified, as also presented in Table 4. This data will be used to evaluate the damping results obtained with the PINN approach, as for an isotropic homogeneous structure with loss factor η, one has $\eta = 2\zeta$. It is noteworthy that, after the updating procedure, the model accurately reproduces the frequencies of the vibration modes of the structure. Therefore, this initial result involving the elastic modulus provides a reference value for comparison with the modulus to be estimated using the PINN framework. Table 5 summarizes the specifications used for constructing the FE model.
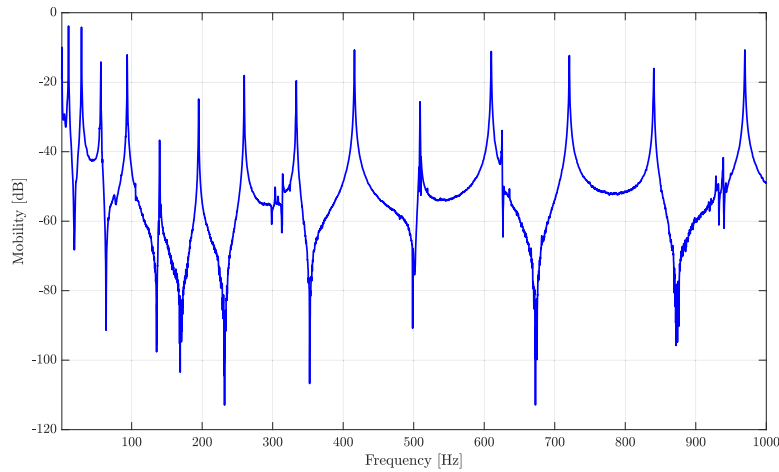
Fig. 12. Mobility of the beam considering single point measurements.

Table 4

Resonant frequencies experimentally measured and those estimated by the FE model, considering the first 11 bending modes. The modal damping ratios associated with each experimental mode are also presented. Only symmetric modes are identified as the sensor is located in the middle of the beam.

Bending Mode	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th
Experimental [Hz]	10.50	28.86	56.50	93.62	139.62	195.10	259.62	333.53	416.44	509.10	610.53
ζ [%]	0.25	0.24	0.18	0.20	0.19	0.22	0.12	0.17	0.19	0.20	0.24
Numerical [Hz]	10.49	28.92	56.70	93.73	140.01	195.56	260.36	334.42	417.73	510.31	612.14

Table 5

Geometrical and material properties of the beam.

Length [m]	1
Width [m]	0.02
Thickness [m]	0.002
Nominal elastic modulus: E [GPa]	205
Poisson ratio	0.30
Density [kg/m^3]	7870

5.2. Complex modulus estimation using the PINN

For the estimation of the complex elastic modulus using the PINN framework, the beam is mounted in a clamped-free boundary condition. In this setup, a length of 40 mm is fixed on a steel support to ensure the clamped condition. Excitation is conducted using a DEWESoft DS-MS-20 shaker and, to minimize interaction effects between the shaker and the structure, the input is applied at a distance of 45 mm from the clamped end of the beam. A PCB force and acceleration transducer, model 288D01, is used to measure the input at the driving point. Once again, the laser PSV-500 Xtra is utilized to measure the velocity field of the structure. Sweep sine tests are performed between 10 and 1000 Hz, considering a frequency increment rate of approximately 193 Hz/s, a total acquisition time of 5.12 s, and a sampling rate of 6250 Hz. Fig. 13 illustrates the experimental setup considered in this work.

The laser scanning is performed over a mesh of 78×3 measurement points spanning the longitudinal surface of the beam, organized in three longitudinal measurement lines — one precisely positioned over the neutral axis of the beam and one at each lateral extremity. To minimize the potential effects of torsion on velocity measurements along the neutral x-axis, as the focus is solely on the bending behavior of the structure, the average of the three columns of measurements is considered. This resulted in a total of 78 measurement points covering from the clamped end to the free end of the beam. Given that the identification framework aims to identify parameters within a specific zone of interest, among the 78 observation points, a dense quantity of 44 measurement points are acquired between 440 mm and 880 mm — referred to as the zone of interest. This choice ensured a measurement point at every 10 mm. Fig. 14 illustrates the mobility of the structure at a measurement point located at the free end of the beam, calculated using the $\mathcal{H}_1$ estimator, considering 5 averaging measurements, and rectangular window. In the following, time-domain measurements are processed using H1 estimator.

From the velocity measurements, it is possible to determine the displacement of the beam at each measurement point through numerical integration performed in the frequency domain. For the application of the identification strategy, the displacement field of the structure measured at a frequency of 530 Hz is chosen. This frequency is in the vicinity of the 11th bending resonant frequency, which is located at 534 Hz.

Since the displacement field is complex, two independent neural networks are proposed, namely $\hat{w}_r(x)$ and $\hat{w}_i(x)$. They have equal architecture, consisting of 10 hidden layers with 9 neurons per layer, and they also consider the $x-$coordinate as the input variable.
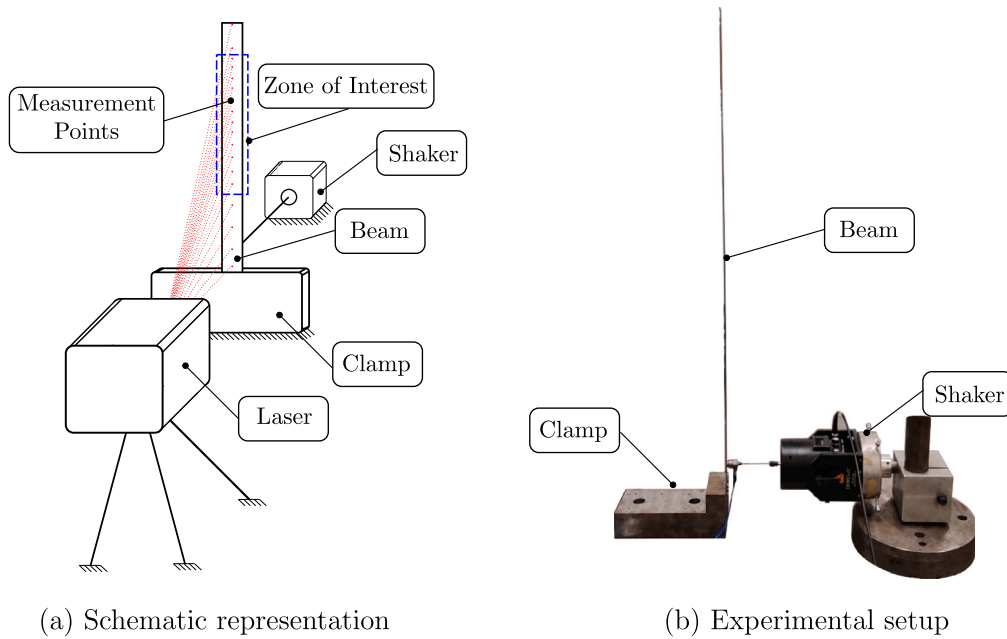
(a) Schematic representation

(b) Experimental setup

Fig. 13. Schematic representation and experimental setup illustrating the cantilever beam.
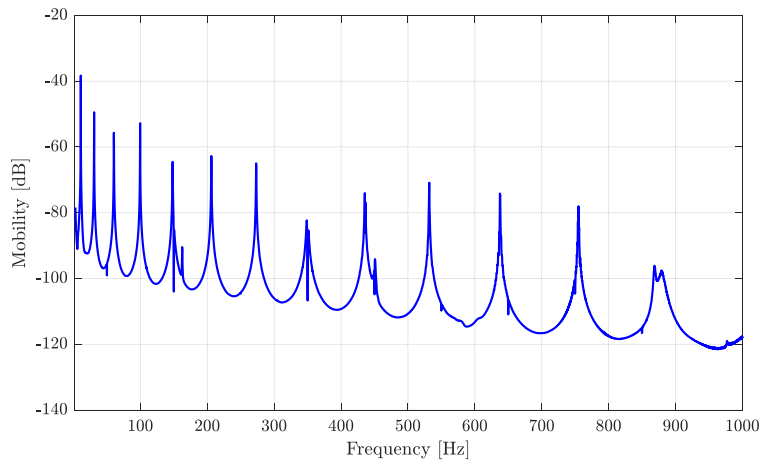

Fig. 14. Mobility calculated considering the measurement point located at the free end of the beam.

In an initial scenario of applying the methodology, $\hat{E}_r$ and $\hat{E}_i$ are also approximated by neural networks of same architectures. For the optimization process, the OneCycle strategy is employed for learning rate scheduling, with a maximum learning rate set to 0.005 at 10% of the total number of epochs, defined as 3×10^4. To construct the PDE, 1024 collocation points are uniformly generated along the beam's area of interest, whereas 22 observation points are used for the data-driven loss — the remaining 22 samples are used as generalization points are used for testing the algorithm.

Fig. 15 illustrates the training loss evolution, indicating stabilization of the total loss function after 22,000 epochs. The total loss value at the end of training is 8.25×10^{-5}. Fig. 16, in turn, presents the predictions of $\hat{w}_r(x)$ and $\hat{w}_i(x)$ generated by the neural networks in direct comparison with the experimental data (generalization points). Considering the testing data, the highest NMSE found is 1.29% on the imaginary displacement. These findings indicate that the methodology handles experimental data, even when confronted with inherent contamination from measurement noise.

Finally, Fig. 17 illustrates the predictions of $\hat{E}_r$ and $\hat{\eta}$ after the training procedure. Both neural networks identified uniform properties, yielding values of 205.5 GPa for the elastic modulus (marginally deviating from the previously identified 205 GPa) and 0.48% for structural damping. The corrected force analysis technique (CFAT) [23] is also employed to compare with the estimate obtained by the PINN method. For a beam in bending, the CFAT method can be developed from the equation of motion of the Euler–Bernoulli model, and the material properties, such as Young's modulus and damping, can be spatially estimated through an inverse
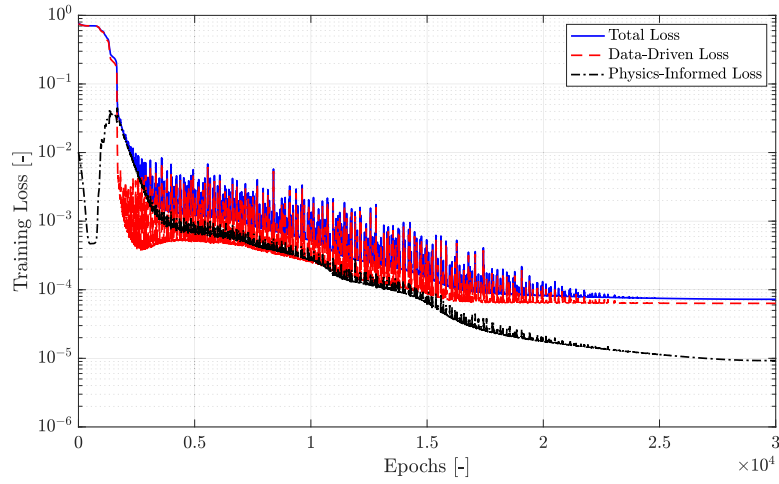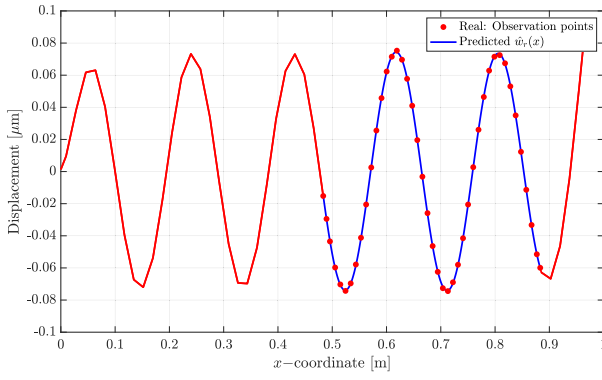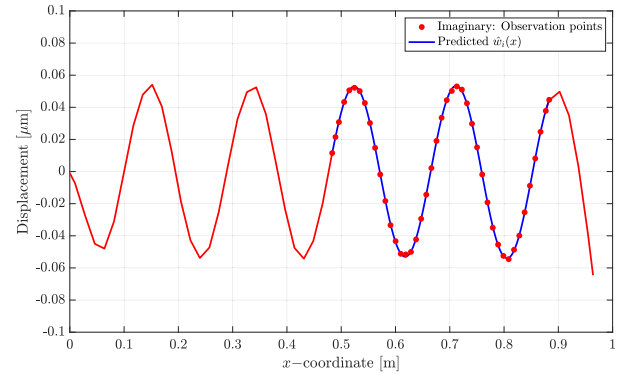
Fig. 15. Evolution of the different loss functions when training is performed considering experimental data. Here, the neural networks of $\hat{E}_r$ and $\hat{E}_r$ are composed by a single biases.
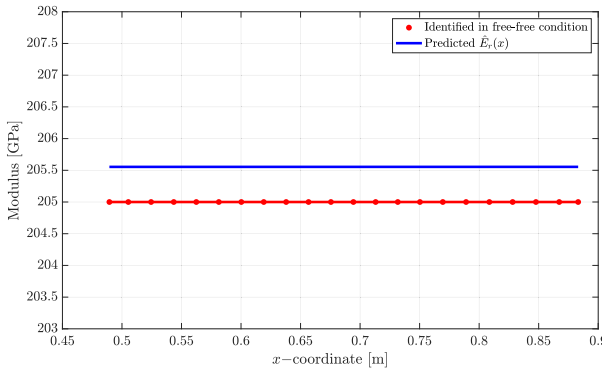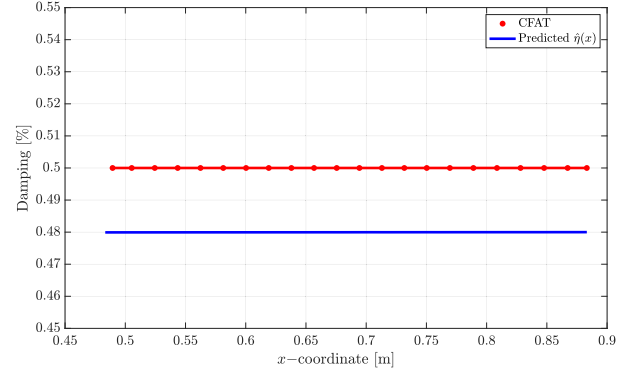

(a) Normalized mean square error: 0.69 %

(b) Normalized mean square error: 1.29 %

Fig. 16. Predictions performed by neural networks on the real $\hat{w}_r(x)$ and imaginary $\hat{w}_i(x)$ components of the beam displacement compared to experimental data around the 11th bending mode. ▬ represents the neural network's predictions on the generalization points, whereas • represents observation points within the zone of interest; ▬ is the experimental points across the remaining domain not utilized by the algorithm.

(a) $\hat{E}_r \approx 205.2$ GPa.

(b) $\hat{\eta} \approx 0.48\%$.

Fig. 17. Predictions of $\hat{E}_r$ and $\hat{\eta}$ parameters when considering the experimental data around the 11th bending mode. ▬ represents the neural network's predictions, whereas • represents the behavior of the material properties across the beam identified by benchmark methods.
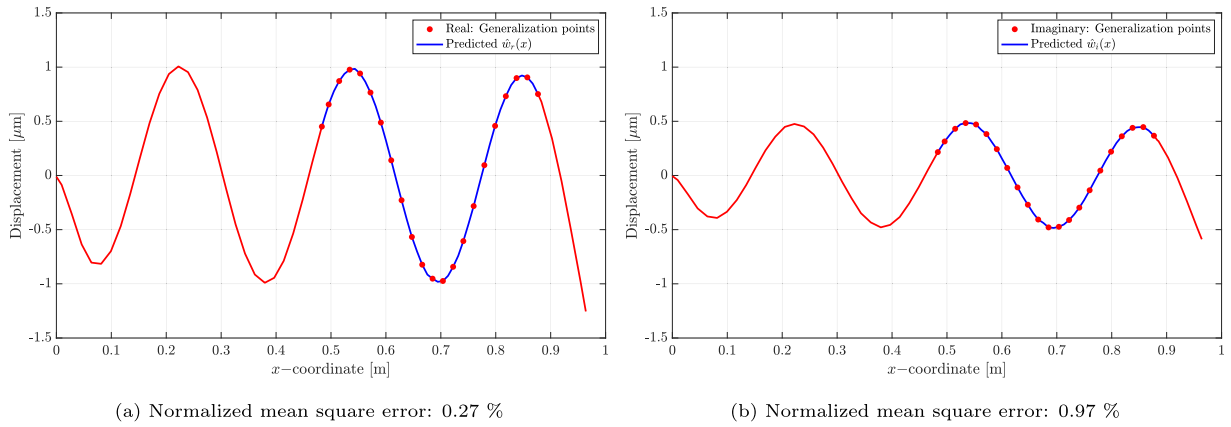
(a) Normalized mean square error: 0.27 %

(b) Normalized mean square error: 0.97 %

Fig. 18. Predictions performed by neural networks on the real $\hat{w}_r(x)$ and imaginary $\hat{w}_i(x)$ components of the beam displacement compared to experimental data around the 7th bending mode. — represents the neural network's predictions on the generalization points, whereas • represents observation points within the zone of interest; — is the experimental points across the remaining domain not utilized by the algorithm.
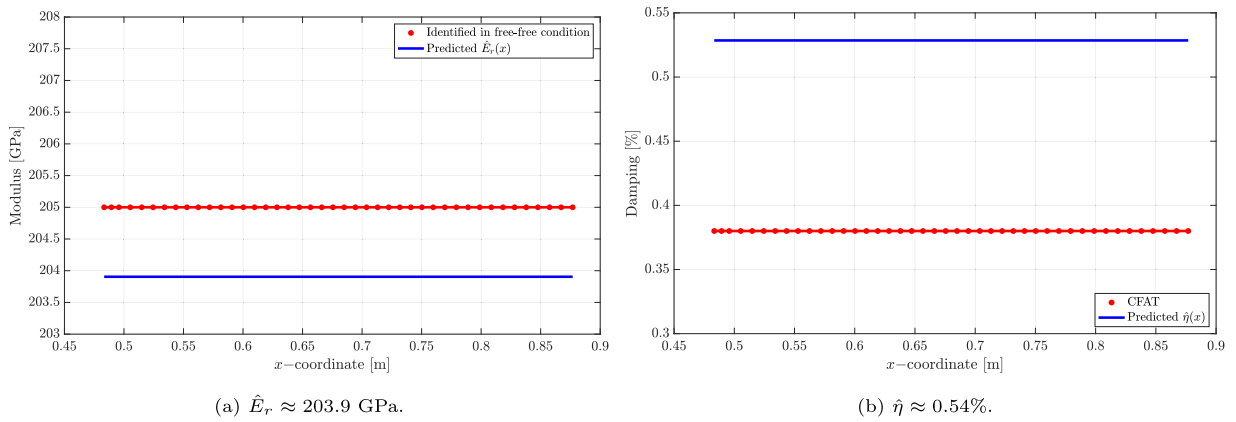

(a) $\hat{E}_r \approx 203.9$ GPa.

(b) $\hat{\eta} \approx 0.54\%$.

Fig. 19. Predictions of $\hat{E}_r$ and $\hat{\eta}$ parameters when considering the experimental data around the 7th bending mode. — represents the neural network's predictions, whereas • represents the behavior of the material properties across the beam identified by benchmark methods.

problem. To solve this inverse problem, it is necessary to use the spatial displacement of the beam within a zone of interest and its fourth-order derivative, which is estimated using a finite-difference numerical approximation. The displacement measurements are used directly from the measured experimental data. The CFAT method was implemented with the space-averaging strategy [23] to be coherent with the PINN implemented in this application, that considers the mechanical parameters as constant in space. It provides an elastic modulus of 219 GPa, which is almost 7% above the values determined as reference and the one identified by the PINN. The loss factor value determined with CFAT is 0.50%, as illustrated in Fig. 17(b), indicating a relative difference of 4% compared to the value determined by the PINN.

The η values identified by both methodologies are close to the damping values extracted in the free-free condition of the structure, as shown in Table 4, between the 9th and 11th bending modes. The damping identified in the free-free condition, if the suspension is well made, is the closest to the material damping identified by both methodologies, since the damping value in the clamped-free condition incorporates both the material damping and the dissipation resulting from the clamped boundary condition. A modal analysis of the clamped-free beam provided a modal damping ratio of the 11th bending mode (534 Hz) of approximately $\zeta = 3.30 \times 10^{-3}$ corresponding to a loss factor $\eta = 2\zeta = 0.66\%$.

To further validate the methodology in identifying the structure's properties near another vibration mode, the same analysis was performed at 190 Hz, a frequency close to the 7th vibration mode of the structure (202 Hz). The predictions for real and imaginary displacement are presented in Fig. 18, while the identified values for $\hat{E}_r(x)$ and $\hat{\eta}(x)$ are displayed in Fig. 19. The modulus estimate remains consistent with previous results, approximately 203.9 GPa, and the damping is found to be 0.54%. Fig. 19(b) compares the predictions from the PINN method with those from the CFAT method. While the difference in loss factor values is more pronounced, with CFAT yielding a loss factor of approximately 0.38%, this discrepancy is not particularly significant. These values are also consistent with the damping estimated in the free-free condition, where $\zeta = 2.20 \times 10^{-3}$ at 195.10 Hz, corresponding to $\eta = 0.44$ %.

It is important to point out that the damping presents inherent challenges in precise estimation. Structural damping is often influenced by various factors, such as material properties, environmental and boundary conditions, and measurement uncertainties.

Therefore, the marginal difference between the damping values obtained from different methods is acceptable, given the inherent complexities and sensitivity associated with accurately quantifying this parameter in structural dynamics.

6. Concluding remarks

The framework presented in this paper offers a new approach to identify structural parameters for beam-like structures, considering the so-called physics-informed neural networks. Its versatility enables the identification of non-homogeneous parameters, including damping properties, thanks to the management of real and imaginary parts of frequency-based data.

The application of the framework to different structural configurations is highlighted by considering cases with both uniform and non-uniform elastic modulus and damping. Notably, the methodology, grounded in the frequency domain, significantly reduces the necessary data for training. The presented methodology represents a pioneering result in the estimation of non-uniform elastic modulus and damping for beam-like structures, providing a novel perspective on structural parameter identification for beam-like systems. In the current approach, only the elastic modulus was identified, under the assumption that the moment of inertia is known. However, if the expression for I is a function of other geometric properties, such as thickness (e.g., $\frac{bh^3}{12}$ for a rectangular cross-section), these properties could become additional parameters to identify. In that case, different structural parameters such as thickness could be estimated independently by leveraging their distinct contributions to the governing equations. However, this would require expanding the model to include complementary tests or additional modes, and introducing separate neural networks for each parameter, which could complicate the convergence of the solution.

Nevertheless, the methodology is further validated through an experimental application involving a cantilever steel beam. Despite inherent noise in the experimental data, the PINN framework exhibits results in agreement with the experimental measurements, as evidenced by the low NMSE values when comparing real and imaginary predicted displacements. With regard to the estimation of material properties, the results obtained by the PINN method are also in agreement with those obtained by other methods traditionally used in the literature, such as the model updating procedure in the case of modulus of elasticity considering a FE model, and the model analysis and the CFAT method, both used to compare estimates of damping. A next analysis that can also be carried out is the construction of a stochastic PINN model for beam-like structures, where the material properties will be seen as random variables rather than deterministic quantities. This will allow the identified model to be more robust, especially when dealing with damping, a parameter that has inherent uncertainties in its identification process.

In this study, neither the input force nor the boundary conditions were incorporated into the observation domain for the inverse problem, since it is often difficult to measure them, especially in practical applications. Although it is feasible to measure the input force in a controlled experimental setup, translating these forces into analytical models is not straightforward due to their complex nature, including the possible presence of torques. For real-world applications, in situ force measurements may be unfeasible, which is why our approach focuses on zones outside the area of force application. Similarly, boundary conditions are even more difficult to access, if not impossible, in many situations. As such, the current methodology intentionally excludes these factors, recognizing this limitation. However, future work could explore the identification of input forces and boundary conditions as unknown or partially known variables in inverse problems, opening up new possibilities to expand the applicability and accuracy of the method.

The results obtained must now be complemented with even more complex cases to illustrate its efficiency, such as multilayer composite structures (beams, plates), frequency-dependent properties of materials embedded in structures, and highly damped materials. Another potential application of this method lies in non-destructive evaluation (NDE), where the PINN framework could be leveraged to detect localized damage in structures, providing an approximate estimate of the local variation in the physical properties of interest. In this work, the focus was on estimating the modulus of elasticity, but other parameters, such as those related to changes in geometry, could also be considered. Moreover, for cases where the number of observation points is insufficient to adequately capture the domain of interest, a multi-frequency approach (thus increasing the number of training points) could be a natural extension of this methodology. This would help tackle more complex and realistic variations, as often encountered in real NDE applications. Ongoing work aims to validate these aspects on experimental benches.

As a last perspective, future work could extend the current approach to account for assumptions suited to other structural models, such as the Timoshenko or Von Karman models, or other boundary conditions and types of solicitations. This would involve addressing coupled systems of partial differential equations governing the dynamic behavior of such structures. Incorporating these adaptations into the PINN framework could open the door to more comprehensive identification and modeling of complex mechanical systems.

CRediT authorship contribution statement

Rafael de O. Teloli: Writing – original draft, Visualization, Validation, Software, Investigation. **Roberta Tittarelli:** Writing – review & editing, Supervision, Formal analysis. **Maël Bigot:** Software, Formal analysis, Conceptualization. **Lucas Coelho:** Software, Formal analysis. **Emmanuel Ramasso:** Writing – review & editing, Software, Formal analysis. **Patrice Le Moal:** Writing – review & editing, Formal analysis. **Morvan Ouisse:** Writing – review & editing, Supervision, Project administration, Investigation, Formal analysis.

Declaration of competing interest

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

Acknowledgments

Data availability

Data will be made available on request.

References

[1] M. Ouisse, S. Cogan, Robust design of spot welds in automotive structures: A decision-making methodology, Mech. Syst. Signal Process. 24 (4) (2010) 1172–1190, http://dx.doi.org/10.1016/j.ymssp.2009.09.012, URL https://www.sciencedirect.com/science/article/pii/S0888327009003094.

[2] E.J. Cross, S.J. Gibson, M.R. Jones, D.J. Pitchforth, S. Zhang, T.J. Rogers, Physics-informed machine learning for structural health monitoring, in: A. Cury, D. Ribeiro, F. Ubertini, M.D. Todd (Eds.), Structural Health Monitoring Based on Data Science Techniques, Springer International Publishing, Cham, 2022, pp. 347–367, http://dx.doi.org/10.1007/978-3-030-81716-9_17.

[3] L.A. Aguirre, An introduction to nonlinear system identification, in: J.R. Castilho Piqueira, C.E. Nigro Mazzilli, C.P. Pesce, G.R. Franzini (Eds.), Lectures on Nonlinear Dynamics, Springer Nature Switzerland, Cham, 2024, pp. 133–154, http://dx.doi.org/10.1007/978-3-031-45101-0_5.

[4] G.E. Karniadakis, I.G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang, Physics-informed machine learning, Nat. Rev. Phys. 3 (6) (2021) 422–440, http://dx.doi.org/10.1038/s42254-021-00314-5.

[5] M. Raissi, P. Perdikaris, G. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. Comput. Phys. 378 (2019) 686–707, http://dx.doi.org/10.1016/j.jcp.2018.10.045.

[6] A. Henkes, H. Wessels, R. Mahnken, Physics informed neural networks for continuum micromechanics, Comput. Methods Appl. Mech. Engrg. 393 (2022) 114790, http://dx.doi.org/10.1016/j.cma.2022.114790.

[7] S. Amini Niaki, E. Haghighat, T. Campbell, A. Poursartip, R. Vaziri, Physics-informed neural network for modelling the thermochemical curing process of composite-tool systems during manufacture, Comput. Methods Appl. Mech. Engrg. 384 (2021) 113959, http://dx.doi.org/10.1016/j.cma.2021.113959.

[8] Z. Mao, A.D. Jagtap, G.E. Karniadakis, Physics-informed neural networks for high-speed flows, Comput. Methods Appl. Mech. Engrg. 360 (2020) 112789, http://dx.doi.org/10.1016/j.cma.2019.112789.

[9] A.M. Roy, S. Guha, A data-driven physics-constrained deep learning computational framework for solving von mises plasticity, Eng. Appl. Artif. Intell. 122 (2023) 106049, http://dx.doi.org/10.1016/j.engappai.2023.106049.

[10] F. As'ad, C. Farhat, A mechanics-informed deep learning framework for data-driven nonlinear viscoelasticity, Comput. Methods Appl. Mech. Engrg. 417 (2023) 116463, http://dx.doi.org/10.1016/j.cma.2023.116463.

[11] S. Rezaei, A. Harandi, A. Moeineddin, B.-X. Xu, S. Reese, A mixed formulation for physics-informed neural networks as a potential solver for engineering problems in heterogeneous domains: Comparison with finite element method, Comput. Methods Appl. Mech. Engrg. 401 (2022) 115616, http://dx.doi.org/10.1016/j.cma.2022.115616.

[12] W. Zhai, D. Tao, Y. Bao, Parameter estimation and modeling of nonlinear dynamical systems based on Runge–Kutta physics-informed neural network, Nonlinear Dynam. 111 (22) (2023) 21117–21130, http://dx.doi.org/10.1007/s11071-023-08933-6.

[13] Z. Lai, C. Mylonas, S. Nagarajaiah, E. Chatzi, Structural identification with physics-informed neural ordinary differential equations, J. Sound Vib. 508 (2021) 116196, http://dx.doi.org/10.1016/j.jsv.2021.116196.

[14] E. Haghighat, M. Raissi, A. Moure, H. Gomez, R. Juanes, A physics-informed deep learning framework for inversion and surrogate modeling in solid mechanics, Comput. Methods Appl. Mech. Engrg. 379 (2021) 113741, http://dx.doi.org/10.1016/j.cma.2021.113741.

[15] Y.A. Yucesan, F.A. Viana, L. Manin, J. Mahfoud, Adjusting a torsional vibration damper model with physics-informed neural networks, Mech. Syst. Signal Process. 154 (2021) 107552, http://dx.doi.org/10.1016/j.ymssp.2020.107552.

[16] X.-Y. Guo, S.-E. Fang, Structural parameter identification using physics-informed neural networks, Measurement 220 (2023) 113334, http://dx.doi.org/10.1016/j.measurement.2023.113334.

[17] D. Di Lorenzo, V. Champaney, J. Marzin, C. Farhat, F. Chinesta, Physics informed and data-based augmented learning in structural health diagnosis, Comput. Methods Appl. Mech. Engrg. 414 (2023) 116186, http://dx.doi.org/10.1016/j.cma.2023.116186.

[18] W. Zhou, Y. Xu, Damage identification for plate structures using physics-informed neural networks, Mech. Syst. Signal Process. 209 (2024) 111111, http://dx.doi.org/10.1016/j.ymssp.2024.111111.

[19] T. Kapoor, H. Wang, A. Núñez, R. Dollevoet, Physics-informed neural networks for solving forward and inverse problems in complex beam systems, IEEE Trans. Neural Netw. Learn. Syst. (2023) 1–15, http://dx.doi.org/10.1109/tnnls.2023.3310585.

[20] S. Faroughi, A. Darvishi, S. Rezaei, On the order of derivation in the training of physics-informed neural networks: case studies for non-uniform beam structures, Acta Mech. 234 (11) (2023) 5673–5695, http://dx.doi.org/10.1007/s00707-023-03676-2.

[21] W. Wu, M. Daneker, M.A. Jolley, K.T. Turner, L. Lu, Effective data sampling strategies and boundary condition constraints of physics-informed neural networks for identifying material properties in solid mechanics, Appl. Math. Mech. 44 (7) (2023) 1039–1068, http://dx.doi.org/10.1007/s10483-023-2995-8.

[22] C. Pezerat, J. Guyader, Force analysis technique: reconstruction of force distribution on plates, Acta Acust. United Acustica 86 (2) (2000) 322–332.

[23] Q. Leclère, F. Ablitzer, C. Pézerat, Practical implementation of the corrected force analysis technique to identify the structural parameter and load distributions, J. Sound Vib. 351 (2015) 106–118, http://dx.doi.org/10.1016/j.jsv.2015.04.025.

[24] A.G. Baydin, B.A. Pearlmutter, A.A. Radul, J.M. Siskind, Automatic differentiation in machine learning: a survey, J. March. Learn. Res. 18 (2018) 1–43.

[25] J.-L. Guyader, Vibration in continuous media, John Wiley & Sons, 2013.

[26] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: Y.W. Teh, M. Titterington (Eds.), Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, in: Proceedings of Machine Learning Research, vol. 9, PMLR, Chia Laguna Resort, Sardinia, Italy, 2010, pp. 249–256, URL https://proceedings.mlr.press/v9/glorot10a.html.

[27] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2017, arXiv:1412.6980, URL https://arxiv.org/abs/1412.6980.

[28] L.N. Smith, N. Topin, Super-convergence: Very fast training of neural networks using large learning rates, 2018, arXiv:1708.07120.