



Physics-Informed Neural Networks for Solving Free Vibration Response of Cables Considering Bending Stiffness

Danhui Dan^{1,2} · Xia Liao¹ · Liangfu Ge³ · Fei Han⁴

Received: 1 August 2025 / Revised: 8 October 2025 / Accepted: 15 October 2025
© The Author(s) 2025

Abstract

Purpose Physics-informed neural networks (PINNs), leveraging their exceptional capacity for nonlinear feature learning, offer a novel approach to solving partial differential equations (PDEs) in structure dynamics. While PINNs have demonstrated feasibility in analyzing the dynamic response of idealized one-dimensional structures, such as tensioned strings and beams, their applicability is limited when addressing the vibration PDEs of real-world cables, particularly those with significant bending stiffness. To overcome this challenge, this paper presents an enhanced PINN methodology designed for the accurate and robust solution of free vibration responses in cables incorporating bending stiffness.

Methods Firstly, a preferred hard-soft boundary constraints strategy is introduced to enhance the prediction accuracy of boundary values. Secondly, a sine activation function is adopted to accelerate network training, replacing conventional alternatives. Thirdly, a hierarchical gradient loss function, coupled with adaptive weights, is introduced to eliminate manual parameter tuning. Finally, a coordinate transformation technique is employed to balance the order-of-magnitude of parameters in the vibration PDEs of the actual suspension cable.

Results This paper systematically explores training strategies for improved PINNs and verifies their effectiveness in solving vibration PDEs for cables considering bending stiffness. The proposed approach delivers accurate solutions for the free vibration of arbitrary cables, providing valuable insights for future research on PINN-based cable vibration analysis.

Conclusion Furthermore, a sensitivity analysis of PDE parameters and network hyperparameters is conducted to examine the time-accumulative effect of PINN solution errors. Some research should focus on solving cable vibration at any time.

Highlights

- The improved PINN with three improvements achieves solutions of any cables.
- A hard-soft constraint strategy is proposed to improve training accuracies.
- Comparing study from trigonometric-activate function is conducted in detail.
- Hierarchical gradient loss function and adaptive weights are proposed.

Keywords Partial differential equations · Cables considering bending stiffness · Physics-informed neural networks · Hard-soft boundary constraints · Adaptive loss function · Coordinate transformation

✉ Liangfu Ge
liangfu.ge@polyu.edu.hk

Danhui Dan
dandanhui@tongji.edu.cn

Xia Liao
lxer@tongji.edu.cn

Fei Han
hanfei@nwpu.edu.cn

¹ School of Civil Engineering, Tongji University, 1239 Siping Road, Shanghai 200092, P. R. China

² Key Laboratory of Performance Evolution and Control for Engineering Structures of Ministry of Education, Tongji University, 1239 Siping Road, Shanghai, China

³ Department of Civil and Environmental Engineering, Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong SAR

⁴ School of Mechanics Civil Engineering and Architecture, Northwestern Polytechnical University, Xian 710129, P. R. China

Introduction

The solution of cable dynamic responses, which essentially solving the vibration partial differential equations (PDEs) of cables, is of great significance in the design, vibration control, and operation and maintenance of cable structures [1]. However, due to high-order partial derivatives associated with bending stiffness and mixed partial derivatives terms with respect to time stemming from axial force, solving the vibration PDEs of cables is more challenging than that of general cable-like structures, such as bending beams and tensioned strings.

Current methodologies for determining cable dynamic responses are predominantly categorized as either analytical or numerical approaches [2]. Analytical approaches primarily formulate vibration partial differential equations (PDEs) based on either tensioned string theory or beam theory, subsequently solving these PDEs analytically through mathematical techniques [3]. However, the inherent bending stiffness of real-world cables introduces discrepancies when employing dynamic analysis based solely on tensioned string theory. Therefore, it is necessary to establish vibration PDEs based on beam theory that additionally considers the effect of bending stiffness. Moreover, analytical approaches often falter when applied to complex cable configurations, such as those incorporating dampers or interconnected multiple cables [4, 5], due to the intractability of these sophisticated PDEs. To address these limitations, numerical methods have been developed to approximate cable dynamic responses. Compared to analytical methods, numerical techniques, including finite element methods, finite difference methods, and Galerkin methods [6–8] can effectively handle the dynamic response of arbitrary complex cable systems. For instance, Abad et al. [9] proposed novel 3D finite elements tailored for cable analysis to examine nonlinear behavior under general loading conditions. To decrease the computational burden, Song et al. utilized a new surrogate model-assisted differential evolution method to solve cables dynamics [10]. Although these methods have achieved some success in practice, solving high-dimensional and complexly configured cable systems or achieving higher accuracy often requires additional interpolation functions or significantly increased mesh density to handle high-order derivatives and complex boundary conditions. This not only substantially increases computational complexity but also potentially leads to numerical error accumulation.

In recent years, with the widespread application of neural networks across diverse domains, including mathematics [11], solid mechanics [12] and thermodynamics [13], the research on solving PDEs by meshless neural networks has garnered significant interest. Berg et al. [14] utilized deep neural networks to approximate PDE solutions in complex

geometries where classical methods based on mesh are impractical. Subsequently, Long et al. [15, 16] proposed new deep learning networks PDE-Net and PDE-Net 2.0 to solve PDEs based on observed dynamic data. However, the neural networks, mostly applied in a supervised manner [17], require a large amount of data to effectively learn underlying mappings. In practical engineering, acquiring extensive datasets is often costly and challenging, with data acquisition hindered by inaccessible measurement points and installation difficulties. Furthermore, data is inevitably subject to noise contamination from sensors and environmental noise. To reduce dependence on real-world data, PINNs were proposed to solve forward and inverse problems involving nonlinear PDEs by incorporating underlying physical information [18]. PINNs obviate the requirement for mesh discretization and extensive datasets, demonstrating efficacy in resolving complex and high-dimensional PDEs. This capability holds the potential to disrupt conventional numerical methodologies for PDE solutions, thereby instigating a paradigm shift in numerical simulation technologies [19].

Recently, Physics-informed neural networks (PINNs) have been well applied in solving problems involving PDEs in various fields, such as solid mechanics, fluid mechanics, etc [20–22]. Some researchers have also begun apply PINNs to solve vibration PDEs of cable-like structures in engineering, such as beams and tensioned strings. Yuan et al. [23] first demonstrated the potential of PINNs in structural dynamics by successfully solving the free vibration equations of Euler-Bernoulli beams. However, these equations contain only the fourth-order spatial derivatives and second-order time derivatives, with all coefficients artificially set to 1. Kapoor et al. [24] extended this approach to solve the complex partial differential equations (sets) of single and double beam systems based on the Euler-Bernoulli and Timoshenko beam, which successfully solved the dynamic response of a single beam under a moving load by using the same network [25]. More recently, Kapoor et al. [26] introduced transfer learning into causal PINNs to improve long-term solution accuracy for these beams. Cem Söyleyici et al. [27] proposed a PINN framework to solve the vibrations of traverse beams under different boundary conditions, tackling high-frequency equations by utilizing the Fourier Feature combined with the Neural Tangent Kernel method. Notably, these studies omitted the second-order spatial derivatives of the deflection function in the beam vibration equations, thus neglecting the influence of axial force on beam dynamic response. Subsequently, Chen et al. [28] employed AT-PINN to solve the vibration PDEs of tensioned strings, which omits the fourth-order spatial derivatives of the deflection function, thereby neglecting the effects of bending stiffness. Based on tensioned string

theory, PINNs have also been applied to the multi-objective prestress optimization of suspension dome structures [29]. Additionally, an improved gradient-enhanced physics-informed neural network (gPINN) has been employed for the shape-finding analysis of tensile membrane structures (TMS) with different forms and boundary conditions [30]. However, in contrast to existing PINN solutions for cable-like structure responses, the cable vibration PDE considering bending stiffness incorporates both bending stiffness and axial force term. To achieve accurate solutions for the dynamic response of actual cables by PINNs, it is necessary to incorporate bending stiffness terms for string vibration equations or introduce axial force terms for beam vibration equations in the PINN framework.

Given this, this paper investigates the free vibration response of suspension cables with bending stiffness by formulating the standard vibration PDE based on Euler-Bernoulli beam theory. Thus, this paper innovates a synergistic approach to successfully achieve accurate solutions for the free vibration response of standard cables (cables considering bending stiffness and beams considering axial force). To achieve robust training of boundary and initial conditions, this paper implements a comprehensive preferred strategy of hard-soft boundary constraints, ensuring intensive learning. The PINN, activated by the sine function similar to the vibration characteristics of cables, significantly accelerates model convergence. Furthermore, the reconstructed loss function, equipped with adaptive weights, intelligently differentiates and penalizes loss terms, eliminating the need for complex parameter tuning and markedly improving network generalization.

The remainder of this paper is arranged as follows. Firstly, this paper presents the general framework of PINN for solving PDEs and applies a classic PINN for cable-like structures to solve the vibration response of cables. Then, an improved PINN with three improvements is introduced: hard-soft boundary constraints, a selected activation function and an adaptive loss function, to better solve the cable vibration response. Subsequently, the cable response with actual parameters can be solved by the improved PINN, associated with the coordinate transformation. Finally, the paper discusses the PDE parameter and the network hyperparameter and performs error analysis of the response solutions.

Limitations of Classical PINNs Solving PDEs

General Framework

A neural network is a hierarchical structure composed of multiple neurons, each of which is interconnected

through weighted connections, and performs a task by learning the mappings between inputs and outputs. Generally, traditional supervised learning requires a large amount of data to achieve accurate predictions. Given this, PINNs, a category of methods integrating physical knowledge into neural networks, are developed to reduce the dependence on data and enhance the modeling and simulation of complex physical systems. PINNs are commonly used in problems described by PDEs of the following form and are essentially applied to approaching their solutions.

$$K_k(u(x); \gamma) = F(x), x \in \Omega_k \quad (1)$$

Where, $k = i, b, p$ represents the initial conditions, boundary conditions, and PDEs, respectively. $\Omega \subseteq R$ is the definition domain. $x = [x_1, \dots, x_n]$ denotes the space and time variables, and u is the unknown function to be solved. $\mathcal{K}$ is the differential operator, while γ is the equation coefficient.

The classical PINN architecture typically consists of a neural network, a physics-informed layer, and a feedback mechanism. Common neural network types used in PINNs include Fully Connected Neural Networks (FCNNs), Convolutional Neural Networks (CNNs), and Recurrent Neural Networks (RNNs). FCNNs are preferred in recent studies for solving PDEs due to generality, flexibility, and user-friendliness [24, 25]. Figure 1 illustrates the general framework of a classical PINN for PDEs, featuring an FCNN as its core.

The input of the whole network in Fig. 1 is the vector $\mathbf{a}^0 = \mathbf{X} = (x_1, x_2, \dots, x_n)^T$. The hidden layers process this input through network weighting and activation, ultimately producing the fitted function for the displacement $\mathbf{U} = (u_1, u_2, \dots, u_n)^T$ as the output. The network logic is expressed with the formula as shown in Eq. (2).

$$z^l = \mathbf{w}^l \cdot \mathbf{a}^{l-1} + b^l, \mathbf{a}^l = f(z^l) \quad (2)$$

Where, $l = 1, 2, \dots, L$ is the number of hidden layers, with the input layer of the entire network denoted as layer 0. z^l is the effective input of layer l . $\mathbf{w}^l$ and b^l are the weight and bias vectors from layer $l-1$ to layer l , respectively. $\mathbf{a}^{l-1}$ is the activation value vector of layer $l-1$, and $f(\bullet)$ is the activation function of the network.

The PINNs involved in this paper are not only a mesh-free function approximator, but also an unsupervised learning method, as they do not rely on a data-driven training process. Instead, they directly incorporate prior physical knowledge into the loss function, so the total loss function of the network is represented as,

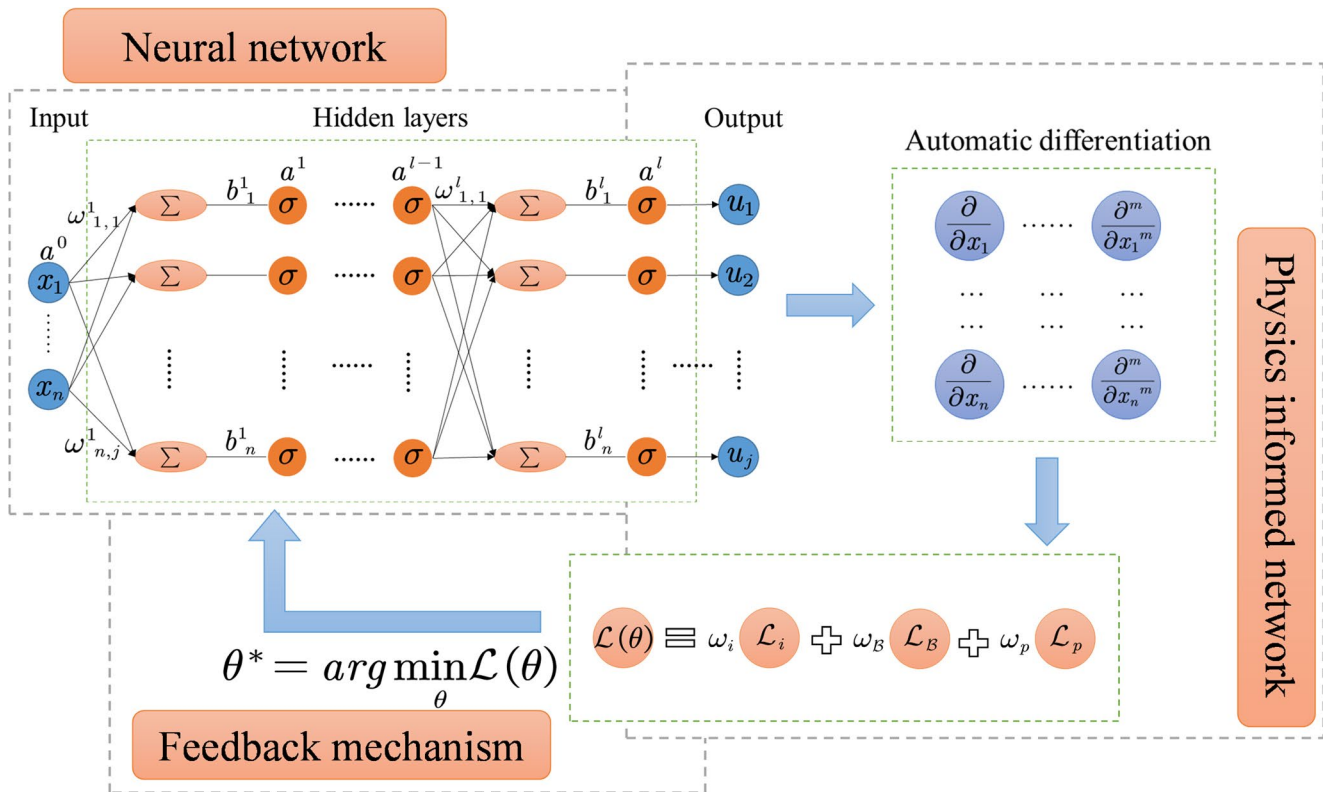


Fig. 1 Classical PINN architecture for solving PDEs

$$\mathcal{L}(\theta) = \omega_i \mathcal{L}_i(\theta) + \omega_b \mathcal{L}_b(\theta) + \omega_p \mathcal{L}_p = \sum_{k=i,b,p} \omega_k \mathcal{L}_k(\theta) \quad (3)$$

Where θ represents the parameters in the network (including $\omega_1, \omega_2 \dots \omega_n$ and $b_1, b_2 \dots b_n$); $\omega_i, \omega_b, \omega_p$ represent the weights corresponding to each loss; $\mathcal{L}_i(\theta), \mathcal{L}_b(\theta), \mathcal{L}_p(\theta)$ denote the loss functions for initial conditions, boundary conditions, and the PDE, respectively. These loss terms are typically measured by the mean squared error (MSE), as expressed below,

$$\mathcal{L}_k(\theta) = \frac{1}{N_k} \sum_{i=1}^{N_k} \|\mathcal{K}_k(\hat{u}_\theta(x_k); \gamma) - \mathcal{F}_k(x_k)\|^2, (k = i, b, p) \quad (4)$$

Where, N is the number of sampling points for initial, boundary conditions and interior of the PDE during training, x stands for the spatial and temporal variables at each sample point, $\hat{u}$ represents the function predicted by the network.

The essence of PINNs lies in transforming the problem of directly solving PDEs into an optimization problem of minimizing the loss function [31]. Currently, the L-BFGS and Adam optimizers are commonly employed to optimize the model parameter $\theta (\omega_1, \omega_2 \dots \omega_n; b_1, b_2 \dots b_n)$. When the loss function given by Eq. (3) is minimized, parameter

θ reaches its optimal value θ^* , leading to optimal-precision prediction of the model.

PDE Solution for Cable Vibration

This paper extends the existing PINNs framework, originally used for solving vibration PDEs of cable-like structures, to the case of standard cables, specifically a simply supported suspension cable, as shown in Fig. 2.

Since the suspension cable is oriented vertically, its static configuration and sag can be neglected in the analysis of transverse vibration. Additionally, given the relatively small vibration displacement, the influence of the additional cable force induced by the vibration on the dynamic response can be disregarded. According to the Euler-Bernoulli beam theory, the free vibration PDE of the cable without damping and considering bending stiffness can be obtained as follows [32].

$$EI \frac{\partial^4 u(x, t)}{\partial x^4} + m \frac{\partial^2 u(x, t)}{\partial t^2} - H \frac{\partial^2 u(x, t)}{\partial x^2} = 0 \quad (5)$$

Where l is the calculated length, EI the bending stiffness, $u(x, t)$ the transverse displacement, H the cable force, x the distance to the lower end and m the cable mass per unit length.

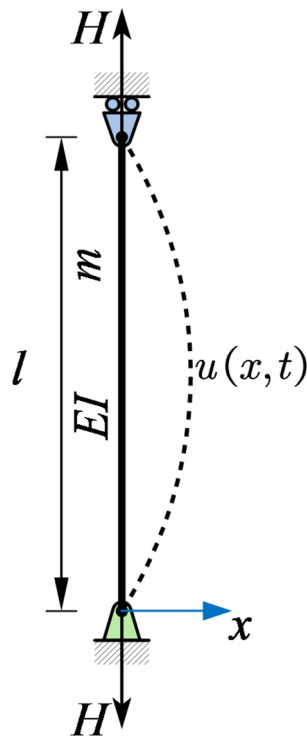


Fig. 2 Analytical model of single suspension cable

To analyze dynamic responses of an actual cable, a suspension cable from an actual engineering is considered [33], as shown in Table 1. Where, initial conditions and boundary conditions of the simply supported cable are as follows.

$$\begin{cases} \Gamma : u(x, 0) = \sin(\pi x/l), u_t(x, 0) = 0 \\ B : u(0, t) = u(l, t) = u_{xx}(0, t) = u_{xx}(l, t) = 0 \end{cases} \quad (6)$$

The analytical solution for the vibration response of the real suspension cable is shown in Eq. (7) and Fig. 3.

$$u(x, t) = \sin \pi x \cos \frac{n\pi}{l} \sqrt{\left(\frac{H}{m} + \frac{EI}{m} \left(\frac{n\pi}{l}\right)^2\right)} t \quad (7)$$

Recent studies have revealed that unnormalized coefficients can adverse to network convergence [24, 25]. Therefore, to calculate the vibration response of cables over the time interval $t \in [0, 1]$ by the classical PINN, firstly nondimensionalizing Eq. (5) should be required, resulting in,

$$u_{xxxx} + u_{tt} - \frac{Hl^2}{EI} u_{xx} = 0, x \in [0, 1], t \in \left[0, \sqrt{\frac{EI}{ml^4}}\right] \quad (8)$$

Where $\tilde{x} = x/l$ and $\tilde{t} = t\sqrt{\frac{EI}{ml^4}}$ represent dimensionless spatial and temporal coordinates respectively.

To train the neural network, 16,000 training points are randomly generated from initial conditions, boundary conditions and the interior of the PDE, and distributed as $\aleph_i = 2000$, $\aleph_b = 4000$, $\aleph_p = 10000$, the distribution of training points shown in Fig. 4. The neural network consists of 4 hidden layers, each containing 20 neurons, and employs the hyperbolic tangent (Tanh) function as the activation

Table 1 Parameters of the actual cable used for validation

Mass m (kg/m)	Length l (m)	Elastic Modulus E (N/m ²)	Moment of inertia I (m ⁴)	Cable force H (N)
14.3	13.5	2×10^{11}	2.45×10^{-7}	2.63×10^5

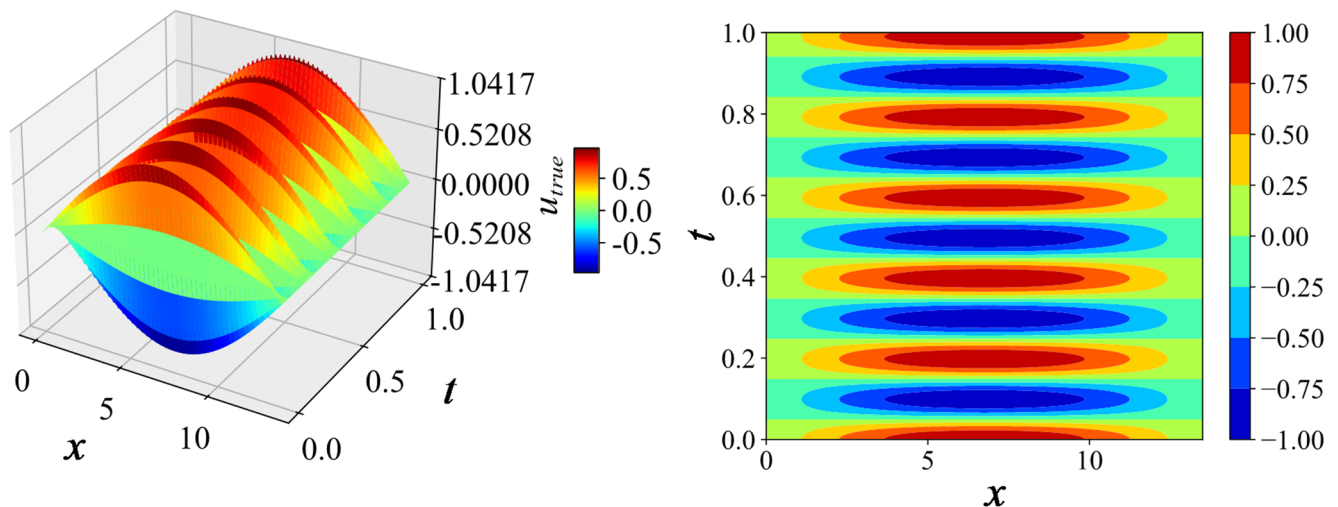


Fig. 3 Analytical solution of the real suspension cable

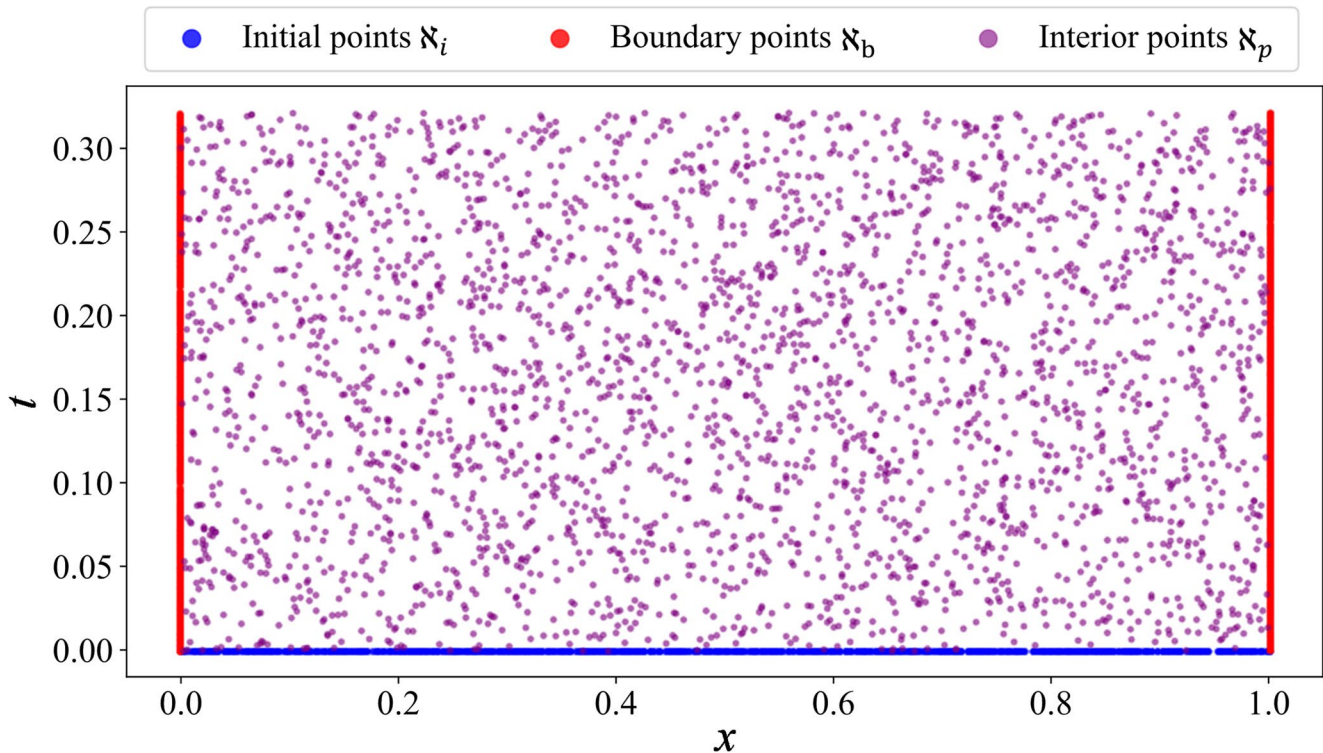


Fig. 4 Distribution of training points

function. The loss function is composed of loss terms from the initial conditions, boundary conditions, and the PDE, regularized by adjusting the weight of the PDE loss term [34]. The loss function is shown in Eq. (3) and minimized by the most common L-BFGS optimizer [13] with the learning rate $l_r = 0.1$. After 15,000 training epochs, the results are shown in Fig. 5.

Figure 5 shows the loss function changes and predicted vibration responses of the cable when using the classical

PINN designed for dynamic responses of cable-like structures to solve that of cables. Although the loss function decreases significantly and converges, the predictions are evidently inaccurate. Specifically, the predicted vibration responses fail to meet the boundary and initial conditions, and there is no apparent vibration trend over time. Compared to the true solution shown in Fig. 3, there are significant errors in solving the vibration response of the cable by the classical PINN. This indicates that the PINN for elementary

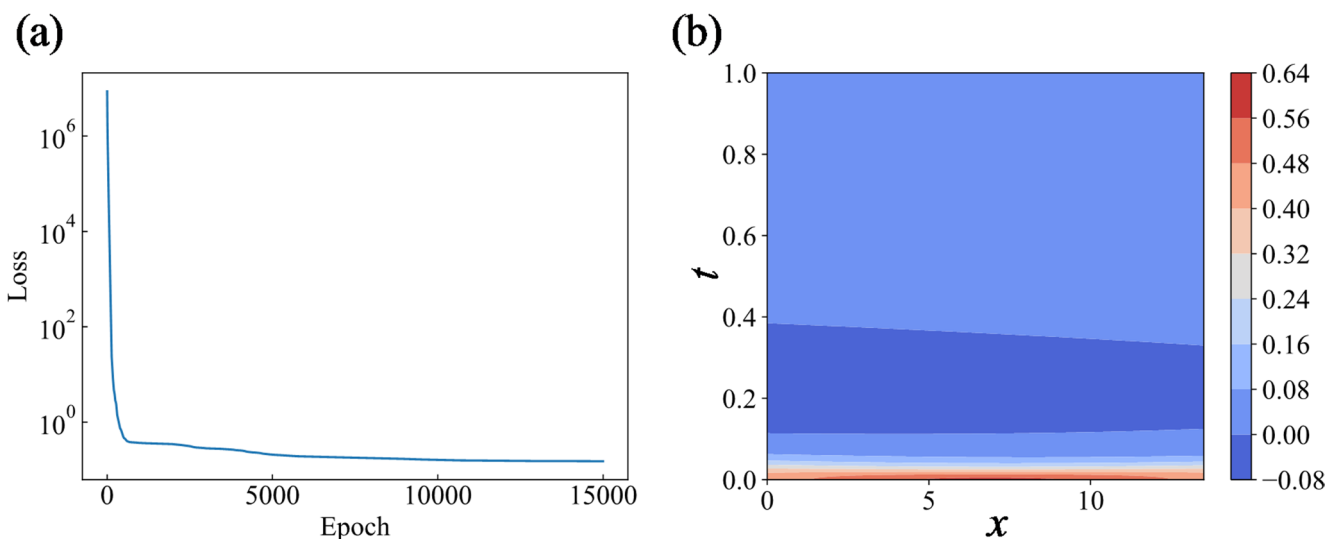


Fig. 5 Cable vibration response by the classic PINN: (a) Loss function; (b) Predicted vibration response

beams is inadequate for capturing the vibration response of cables, and its network lacks generalization. Therefore, it is necessary to improve and innovate the existing network in order to solve the cable vibration PDE.

The inapplicability of the classical PINN mainly stems from the fact that the vibration PDEs of cables simultaneously involve both second-order and fourth-order partial derivatives with respect to coordinates, compared with that of beams, i.e., considering the influence of axial force and bending stiffness on the vibration response. Thus, it is difficult to simultaneously normalize the equation coefficients through dimensionless equations.

In addition, even though some methods consider tension forces in PINN models, such as AT-PINN [28], they ignore the influence of bending stiffness on vibration. It is still challenging to directly apply them to solve vibration PDEs of the aforementioned standard cable structures.

In view of this, this paper addresses the limitations of two existing PINNs mentioned above and proposes an improved PINN tailored for solving the vibration of cables considering bending stiffness and beams considering axial force.

The Improved PINN

This paper improves the classical PINN frameworks of cable-like structures and develops an architecture suitable for solving the dynamic response of cables. To improve the training process for arbitrary boundary conditions, this paper

establishes the preferred strategy of hard-soft boundary constraints. Simultaneously, complementary improvements are applied to the existing PINNs, including modifying the activation function, refining the loss function, and adopting adaptive weight to collectively address the challenges posed by vibration PDEs of cables considering bending stiffness.

This paper follows the basic network parameters of the PINN framework for cable-like structures in Sect. [General framework](#). However, the Adam optimization algorithm is employed as the optimizer for the network parameters, with the Cosine Annealing LR algorithm employed to dynamically adjust the learning rate. To train the neural network efficiently, mini-batch sampling is utilized to accelerate the training process, with resampling every 100 epochs. Each sampling generates 600 random training points, distributed as $N_i = 100$, $N_b = 200$ and $N_p = 300$, where 100,000 training epochs are performed and the distribution of sampling points for a single epoch is shown in Fig. 6. Based on this network configuration, this paper improves the PINN-solving process as follows.

Preferred Strategy of Hard-Soft Boundary Constraints

In the existing PINN framework, boundary conditions are typically enforced through optimization with the soft boundary constraint, where penalties are imposed by setting the weights of the boundary loss term. The soft boundary constraints emphasize three contributing components in

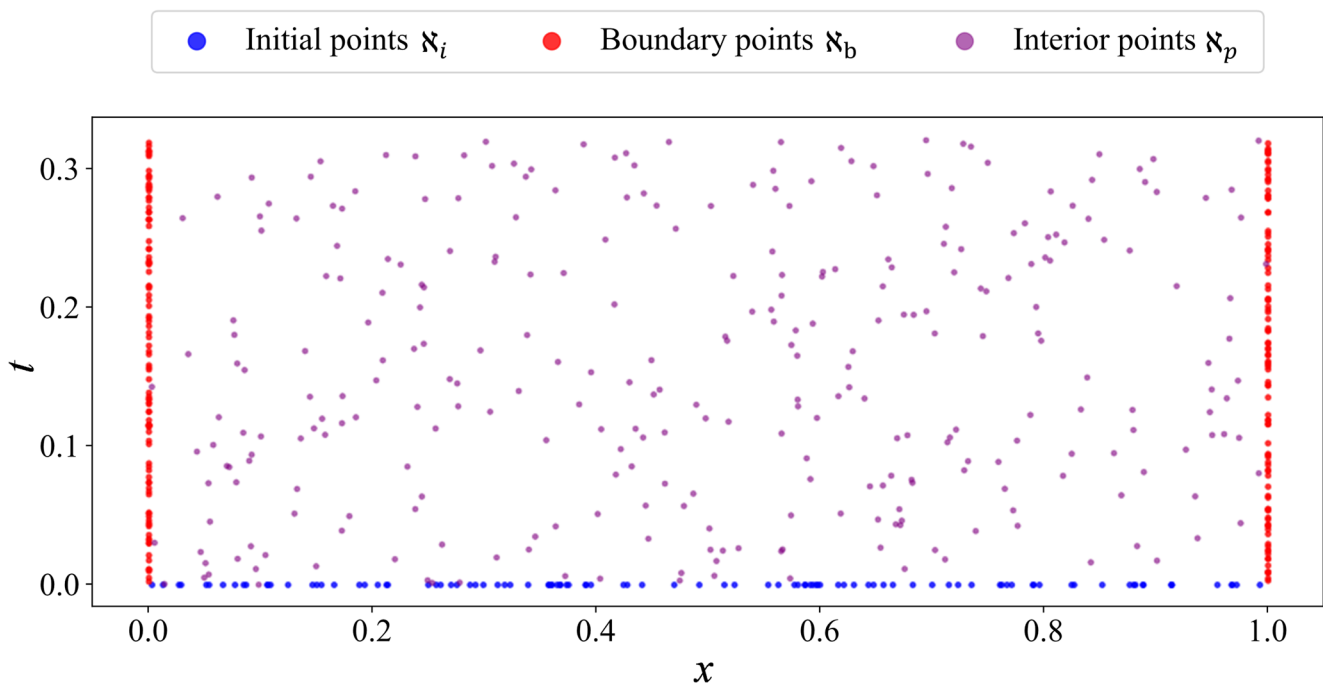


Fig. 6 Distribution of sampling points

constructing the loss function, where the initial and boundary condition loss guarantees training points to satisfy the initial and boundary conditions, and the PDE loss guarantees training points of the entire definition domain to satisfy the PED. However, due to the PDE loss trained on random sampling throughout the entire domain, the probability that sampling points accurately fall on the initial/boundary conditions is very small, so it cannot be fully guaranteed that the initial/boundary points meet the PDE while satisfying the initial/boundary conditions.

In contrast, PINNs with hard boundary constraints (hPINNs) can improve the learning effect of the boundary points, which involve boundary conditions in neural networks. Compared to the soft boundary constraint PINN, hPINNs can ensure that the network output automatically meets the boundary conditions [35]. However, for cases where the boundary conditions cannot be expressed in analytic form, and where the analytic boundary conditions contain derivative terms, hPINNs fails due to the inability to construct a hard boundary function.

To address the limitations of the two cases described above, this paper proposes a comprehensive preferred strategy of hard-soft boundary constraints to reinforce boundary conditions, which establishes the selection criteria by identifying whether the hard boundary function can be expressed analytically. If an analytical expression is not feasible, an improved soft boundary constraint PINN should be employed, which incorporates additional terms into the loss function to ensure that boundary sampling points meet the PDEs. Conversely, when analytical boundary functions are available, a hard boundary constraint PINN is prior to be selected to ensure that the network solution satisfies boundary conditions, thereby reducing the difficulty of optimizing the loss function. The PINN with this optimization strategy is shown in Fig. 7.

This paper proposes the loss function of the improved soft boundary constraint PINN, which additionally adds the contribution of the boundary/initial points to the PDE loss compared to the loss function of the existing PINN, to strengthen the PDE learning. Thus, $\mathcal{L}_p(\theta)$ in Eq. (4) is rewritten $\mathcal{L}_{pall}(\theta)$ as follows.

$$\begin{cases} \mathcal{L}(\theta) = \omega_i \mathcal{L}_i(\theta) + \omega_b \mathcal{L}_b(\theta) + \omega_{pall} \mathcal{L}_{pall}(\theta) \\ \mathcal{L}_{pall}(\theta) = \frac{1}{N_p} \sum_{i=1}^{N_p} \|\mathcal{K}_p(\hat{u}_\theta(x_p); \gamma) - \mathcal{F}_p(x_p)\|^2 \\ + \frac{1}{N_b} \sum_{i=1}^{N_b} \|\mathcal{K}_p(\hat{u}_\theta(x_b); \gamma) - \mathcal{F}_p(x_b)\|^2 \\ + \frac{1}{N_i} \sum_{i=1}^{N_i} \|\mathcal{K}_p(\hat{u}_\theta(x_i); \gamma) - \mathcal{F}_p(x_i)\|^2 \end{cases} \quad (9)$$

Where the second and third terms of the right-hand side of the $\mathcal{L}_{pall}(\theta)$ represent the contributions of boundary and initial conditions points to the PDE loss term, as previously mentioned.

For the Dirichlet boundary conditions of cables, this paper constructs the hard boundary function as follows to ensure that the network output directly satisfies boundary conditions and initial conditions.

$$\hat{u}_b(x, t, \theta) = t \cdot \hat{u}(x, t, \theta) (x - x_{\min})(x_{\max} - x) + u(x, t_{\min}) \quad (10)$$

Where, $\hat{u}(x, t, \theta)$ is the original output of the network training, $\hat{u}_b(x, t, \theta)$ is the function that satisfies the hard boundary constraint, $u(x, t_{\min})$ is initial displacement function, $x_{\min}$ and $x_{\max}$ are the left and right boundaries of the spatial variables, and $t_{\min}$ is the initial time. When the network performs Eq. (10), $\hat{u}_b$ is forced to align with the boundary and initial displacement function, i.e., when $x = x_{\min}$ or $x = x_{\max}$, $\hat{u}_b(x, t, \theta) = 0$; when $t = t_{\min}$, $\hat{u}_b(x, t, \theta) = u(x, t_{\min})$. Hence, the loss function can be simplified as follows,

$$\mathcal{L}(\theta) = w_p \mathcal{L}_p \quad (11)$$

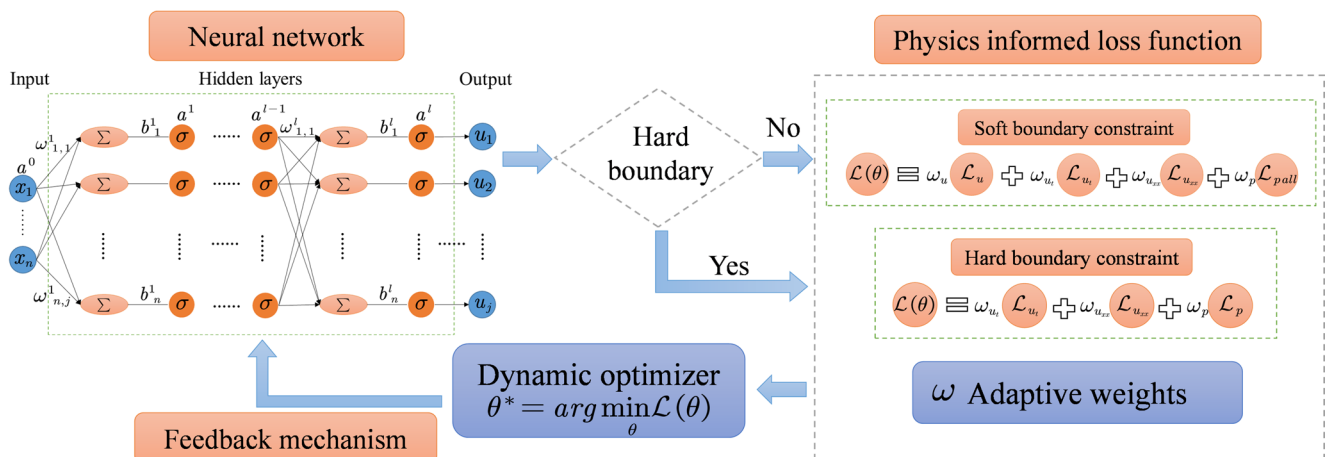


Fig. 7 PINNs for solving cable dynamic response

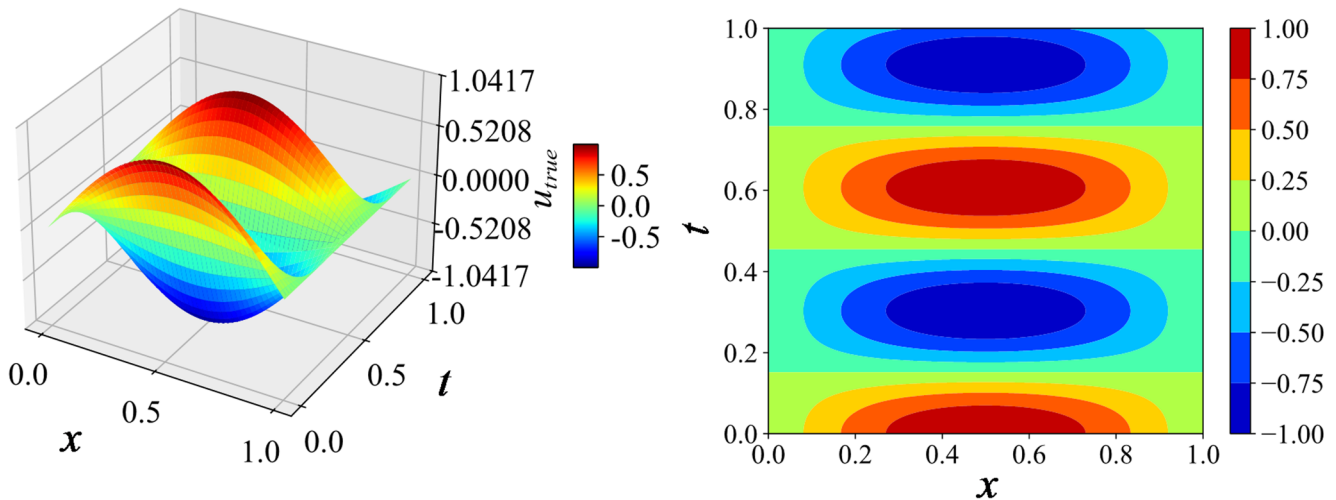


Fig. 8 True vibration response of the simplified cable

When solving the PDEs of the general cables, either Eq. (9) or Eq. (11) should be selected as the loss function to enhance the training of the boundary conditions, according to the selection criteria of hard-soft boundary constraints established above.

In order to illustrate the validity of the optimization strategy, this paper specifically designs the PDE case shown in Eq. (12), which models the free vibration of a simplified parameter cable that can be suitable for the two-types constraints PINNs. In this case, all cable parameters are set to 1. The training has been carried out by the aforementioned PINN parameters in Sect. The improved PINN, and the calculations are shown in Figs. 9 and 10, respectively. For ease of comparison, the analytical solution of Eq. (12) is provided as shown in Fig. 8.

$$u_{xxxx} + u_{tt} - u_{xx} = 0, x \in [0, 1], t \in [0, 1] \quad (12)$$

Where the initial condition is $u(x, 0) = \sin(\pi x)$, $u_t(x, 0) = 0$, the boundary condition is $u(0, t) = u(1, t) = u_{xx}(0, t) = u_{xx}(1, t) = 0$, and the true solution is $u(x, t) = \sin \pi x \cos n \pi \sqrt{1 + (n\pi)^2} t$, as shown in Fig. 8.

From Fig. 9, the training results of the improved soft boundary constraint PINN can be observed. Figure 9(a) shows that the loss function basically converges after training 100,000 epochs, although there is a trend of further decreasing and converging. From Fig. 9(b), predicted responses of the network are largely consistent with the true solution of the PDE, with its absolute errors remaining below 0.0042. Figure 9 illustrates that the improved soft boundary constraint PINN can accurately catch dynamic response of the cable by effectively solving the PDE.

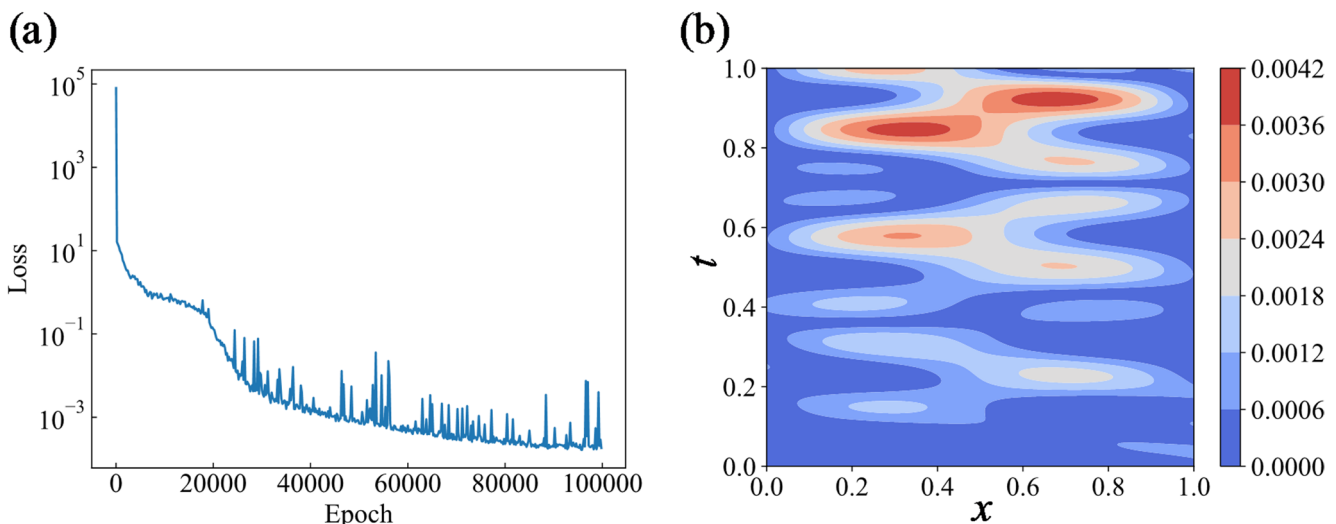


Fig. 9 Solution of the improved soft boundary constraint PINN: (a) Loss function; (b) Absolute errors of predicted responses

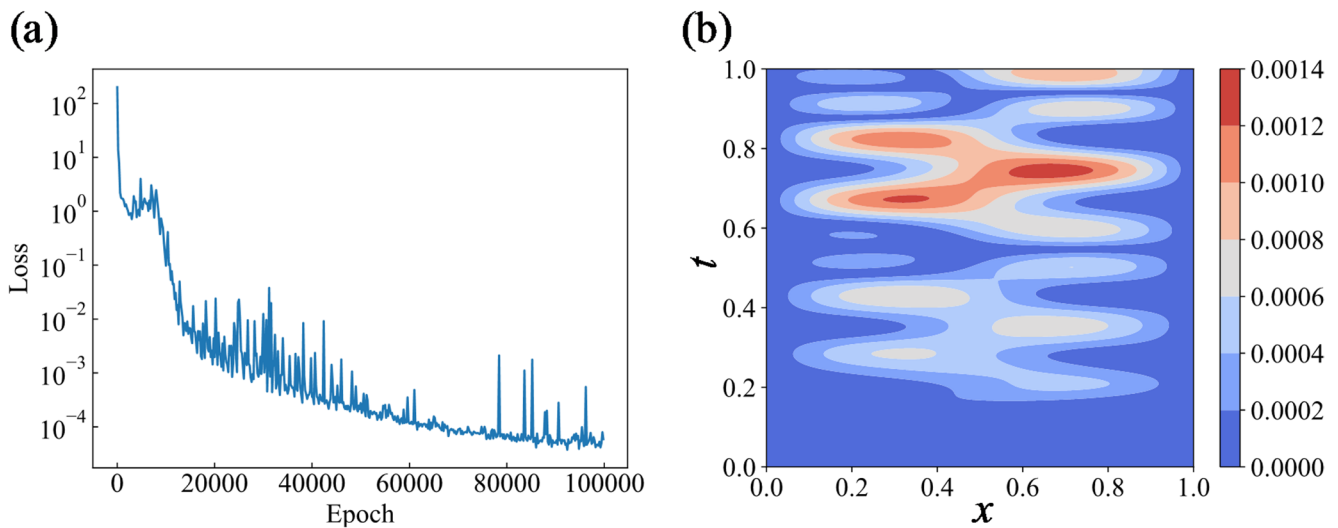


Fig. 10 Solution of the hard boundary constraint PINN: (a) Loss function; (b) Absolute errors of predicted responses

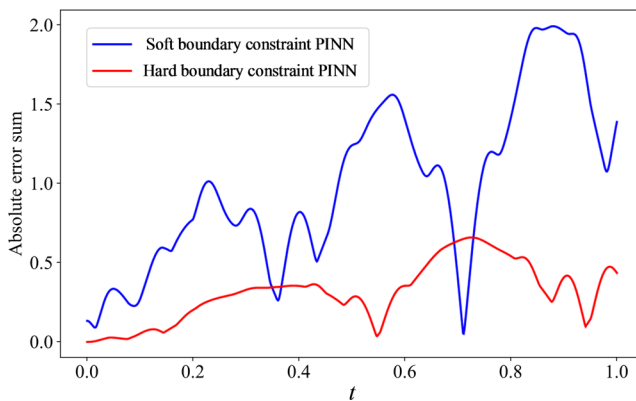


Fig. 11 Absolute error sum with respect to time

The training results of hard boundary constraint PINN in Fig. 10 show that predicted values highly match true values. It can be seen from these figures that the hard boundary constraint PINN can effectively solve vibration PDEs of the standard cable. In addition, the presence of the hard boundary constraint makes the boundary loss zero at floating-point precision, which reduces the workload of network optimization to a certain extent. As seen in Fig. 10 (b), the absolute errors of the hard boundary constraint PINN are less than 0.0014, with its relative errors below 0.14%, whereas the improved soft boundary constraint PINN needs more training epochs to achieve comparable error levels.

To compare the calculation results of soft boundary PINN and hard boundary PINN, the absolute errors from the calculations of both networks are spatially aggregated to evaluate the temporal variation of error, as illustrated in Fig. 11.

From Fig. 11, it can be seen that the absolute error sum increases over time for both PINNs. Comparing absolute

error sum of the improved soft boundary constraint PINN with that of the hard boundary constraint PINN, it is revealed that the latter exhibits errors approximately half those of the former within the same training epochs. Although both PINNs can accurately solve the cable vibration with similar approximation, the hard boundary constraint PINN is more efficient in training and should be prioritized. However, if the hard boundary function is difficult to be expressed analytically, the improved soft boundary constraint PINN remains an effective alternative for solving such PDEs. Therefore, the comprehensive preferred strategy of hard-soft boundary constraint PINNs proposed in this paper can enhance boundary training in solving vibration responses of arbitrary cables, and improve the generality of PINNs while conserving computational resources.

Activation Function

The activation function can introduce nonlinear factors into neurons, enabling the neural network to arbitrarily approximate any nonlinear function, so that the neural network expression ability is more powerful. Different activation functions employ distinct approximation methods and operate within specific activation domains. Currently, the common activation functions include the Sigmoid function, Tanh function, and ReLU function, where Tanh function is one of the most commonly used activation functions in PINNs for solving PDEs due to its continuous differentiability.

Chen et al. [28] mentioned that choosing activation functions that have the same form as the solution of the PDE can improve network training efficiency. In this paper, the vibration responses of the cables generally consist of trigonometric and hyperbolic functions. Theoretically, employing these two functions as activation functions can more

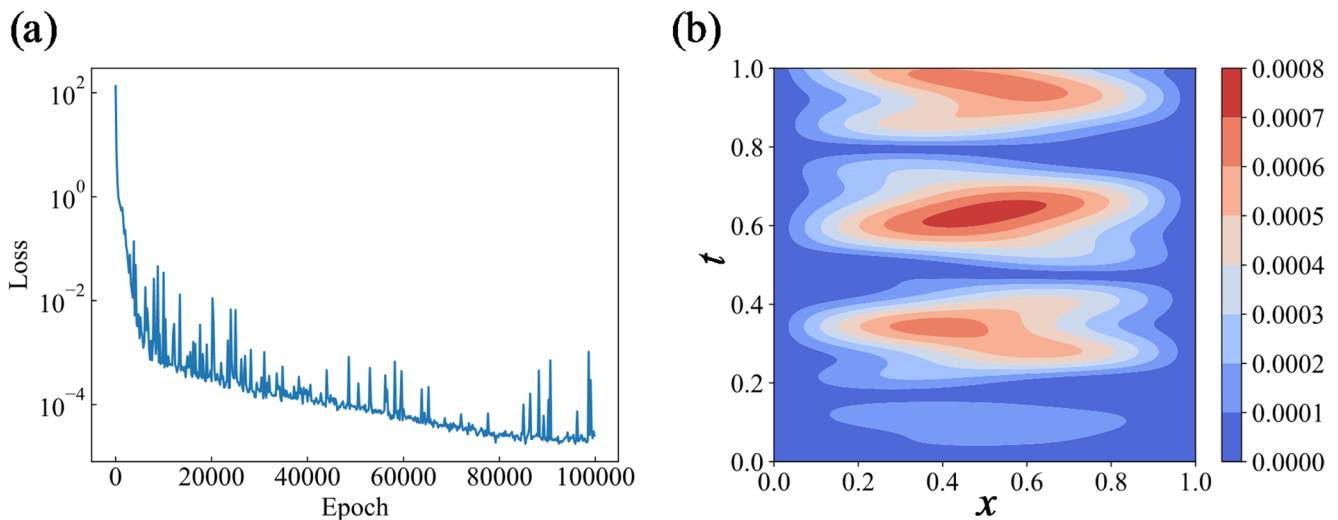


Fig. 12 Solution of Sin-activated PINN: (a) Loss function; (b) Absolute errors of predicted responses

quickly approximate the PDE solution. However, the function selected as the activation function and its derivatives should be within a suitable interval, neither excessively large nor too small, as this affects the training efficiency and stability. Obviously, hyperbolic functions are not suitable, which often yield extreme values. Therefore, this paper selects the trigonometric function sine (Sin) as the activation function for the network.

To illustrate the effectiveness of the activation function, this paper compares the solutions using a hard boundary constraint PINN activated by different functions. The previous solution in Fig. 10 has been realized by the Tanh-activated PINN, and under the same conditions as that case, the activation function Tanh is replaced by the Sin function in hard boundary constraint PINN. The results solved by Sin-activated PINN are shown in Fig. 12.

From Fig. 12, it can be seen that the Sin-activated PINN demonstrates greater accuracy in solving the vibration PDEs of the standard cable structure, with the absolute errors between the predicted and the true solutions being less than 0.0008. Comparing with the solutions of the Tanh-activated PINN in Fig. 10, it can be seen that, although they can accurately solve such PDEs, the loss of the Sin-activated PINN is significantly smaller within the same training epochs, with errors approximately half those of the Tanh-activated PINN. As can be seen in Fig. 13, the absolute error sum in the Sin-activated PINN is smaller and exhibits a distinct periodicity compared with that of the Tan-activated PINN. Moreover, compared to the divergence of the absolute errors in the Tanh-activated PINN, the absolute errors in the Sin-activated PINN are mostly concentrated at points where the true values are larger, further reducing the relative errors, the variation of which has the same periodicity as cable vibration. Therefore, when solving vibration PDEs of the

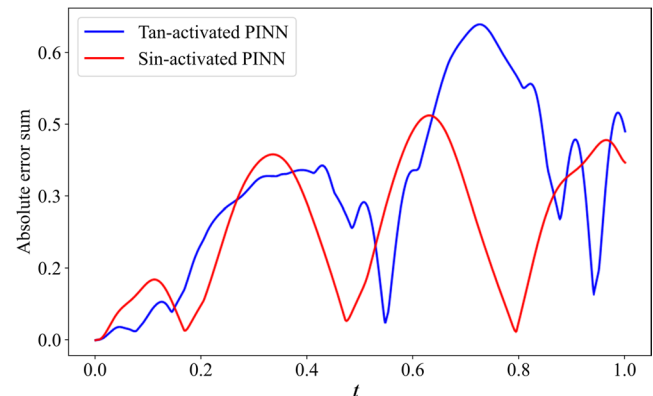


Fig. 13 Absolute error sum with respect to time

standard cable structure in PINN, choosing the Sin function as the activation function can lead to higher training efficiency and more accurate solutions than the Tanh function. Thus, the Sin function is more suitable to serve as the activation function and is recommended to be chosen for solving vibration responses of cables in PINNs.

Adaptive Loss Function

Hierarchical Gradient Loss Function

The essence of solving PDEs by PINNs is the optimization problem of the loss function, and the loss function in the classical PINN is mainly the weighted sum of the loss terms of initial conditions, boundary conditions, and the PDE, as shown in Eq. (3). By setting the weights of each loss to adjust the influence of these components on the model training, their interactions can be balanced for more stable convergence. For the vibration PDEs of the cable structure,

the boundary and initial conditions contain displacements and their various derivatives, leading to significant gradient disparities. This gradient disparity makes it challenging to adequately penalize the internal components of the loss through weight adjustments alone, thereby hindering the optimization process. Therefore, to address this challenge and accurately penalize each component of the loss in Eq. (3), this paper proposes the hierarchical gradient loss function according to the partial derivatives order.

$$\mathcal{L}(\theta) = \omega_u \mathcal{L}_u(\theta) + \omega_{u_t} \mathcal{L}_{u_t}(\theta) + \omega_{u_{xx}} \mathcal{L}_{u_{xx}}(\theta) + \omega_p \mathcal{L}_{pall}(\theta) \quad (13)$$

Where, $\mathcal{L}_{pall}(\theta)$ as shown in Eq. (8) is the PDE loss, with training points derived from the initial conditions, boundary conditions, and the interior of the PDE. $\mathcal{L}_u(\theta)$ and ω_u are the loss term and weight corresponding to the function in the initial and boundary conditions. $\mathcal{L}_{u_t}(\theta)$ and ω_{u_t} are the loss term and weight corresponding to the first-order time gradient of the function in the initial conditions. $\mathcal{L}_{u_{xx}}(\theta)$ and $\omega_{u_{xx}}$ are the loss term and weight corresponding to the second-order spatial gradient of the function in the boundary conditions. The specific expressions for each loss term are as follows.

$$\left\{ \begin{array}{l} \mathcal{L}_u(\theta) = \frac{1}{N_b} \sum_{i=1}^{N_b} \|\hat{u}(0, t, \theta) - u(0, t)\|^2 + \\ \quad \frac{1}{N_b} \sum_{i=1}^{N_b} \|\hat{u}(l, t, \theta) - u(l, t)\|^2 + \\ \quad \frac{1}{N_i} \sum_{i=1}^{N_i} \|\hat{u}(x_i, 0, \theta) - u(x, 0)\|^2 \\ \mathcal{L}_{u_t} = \frac{1}{N_i} \sum_{i=1}^{N_i} \|\hat{u}_t(x_i, 0, \theta) - u_t(x, 0)\|^2 \\ \mathcal{L}_{u_{xx}} = \frac{1}{N_i} \sum_{i=1}^{N_i} \|\hat{u}_{xx}(0, t, \theta) - u_{xx}(0, t)\|^2 + \\ \quad \frac{1}{N_b} \sum_{i=1}^{N_b} \|\hat{u}_{xx}(l, t, \theta) - u_{xx}(l, t)\|^2 \end{array} \right. \quad (14)$$

When a hard boundary function is used in the network, $\mathcal{L}_u(\theta)$ remains consistently zero. Therefore, in the hard boundary constraint PINN, Eq. (12) can be simplified according to the hard boundary function as follows,

$$\mathcal{L}(\theta) = \omega_{u_t} \mathcal{L}_{u_t}(\theta) + \omega_{u_{xx}} \mathcal{L}_{u_{xx}}(\theta) + \omega_p \mathcal{L}_p(\theta) \quad (15)$$

Adaptive Weights

Adjusting the weight of each loss term can further activate the optimizer and accelerate the optimization of the loss function. Concerning the selection of appropriate weight hyperparameters, Kapoor et al. [24] solved vibration responses of beams by directly assigning a weight of 0.1 to the PDE loss. However, such weight settings are often problem-specific, which significantly limits the generalization of the network. Therefore, Wang et al. [36] theoretically provided the principles and derivations of weighting from the tangent kernel perspective. However, this approach primarily addresses the issue that the convergence of the PDE

loss is much faster than the fitting of the boundary and initial conditions, and calculating the Jacobi matrix in the tangent kernel idea greatly increases the computational cost. However, for solving the vibration PDEs of cables, the fitting of the boundary and initial conditions is far outpacing the convergence of the PDE. If a hard boundary function is applied, it can further accelerate the convergence in the loss of the boundary and initial conditions. From this, it can be seen that setting weights from the tangent kernel idea is not suitable for the vibration PDE of cables.

Therefore, this paper proposes an adaptive weight criterion for each loss from its values and decrease rates. Under this approach, loss terms with relatively larger values and slower decline rates should be assigned greater penalty weights. This adaptive balancing approach can accelerate the optimization of the loss function, which is conducive to the network to find the global optimum and improve the generalization. The adaptive weights for each loss term are as follows,

$$\omega_i = \frac{\mathcal{L}_i'}{\min(\mathcal{L}_i')} \cdot \frac{\max\left(\left|\frac{\mathcal{L}_i'' - \mathcal{L}_i'}{\mathcal{L}_i'}\right|\right)}{\left|\frac{\mathcal{L}_i'' - \mathcal{L}_i'}{\mathcal{L}_i'}\right|} \quad (16)$$

Where i denotes the various components of the loss function, $\min(\cdot)$ denotes the minimum value among all loss terms from the previous training epoch, $\max(\cdot)$ denotes the maximum decline rate across all loss terms, $\mathcal{L}_i'$ and $\mathcal{L}_i''$ denote the loss terms from the previous two training epochs, respectively.

To verify the effectiveness of the reconstructed loss function, adaptive weights are employed to adaptively balance the loss function during solving PDEs by improved soft boundary constraint PINNs. Figure 14 compares the PINN solutions obtained with the classical loss function and the reconstructed loss function.

As can be seen from Fig. 14, the adaptive weights adjust in real-time in response to changes in each loss term. Both PINNs guided by the classical loss function and the hierarchical gradient loss function with adaptive weights can accurately solve vibration responses of the cable, achieving maximum absolute errors on the order of 10^{-3} . However, comparing the absolute errors computed by the two loss function networks in Fig. 14 (e) and (f), it is evident that the errors under the hierarchical gradient loss function are significantly smaller. The main reason is that the reconstructed loss function can make the weights penalize the inside of each loss, preventing mutual interference and overshadowing among the components, thereby promoting overall optimization of the loss function and improving the solution accuracy. Comparing the convergence of the loss

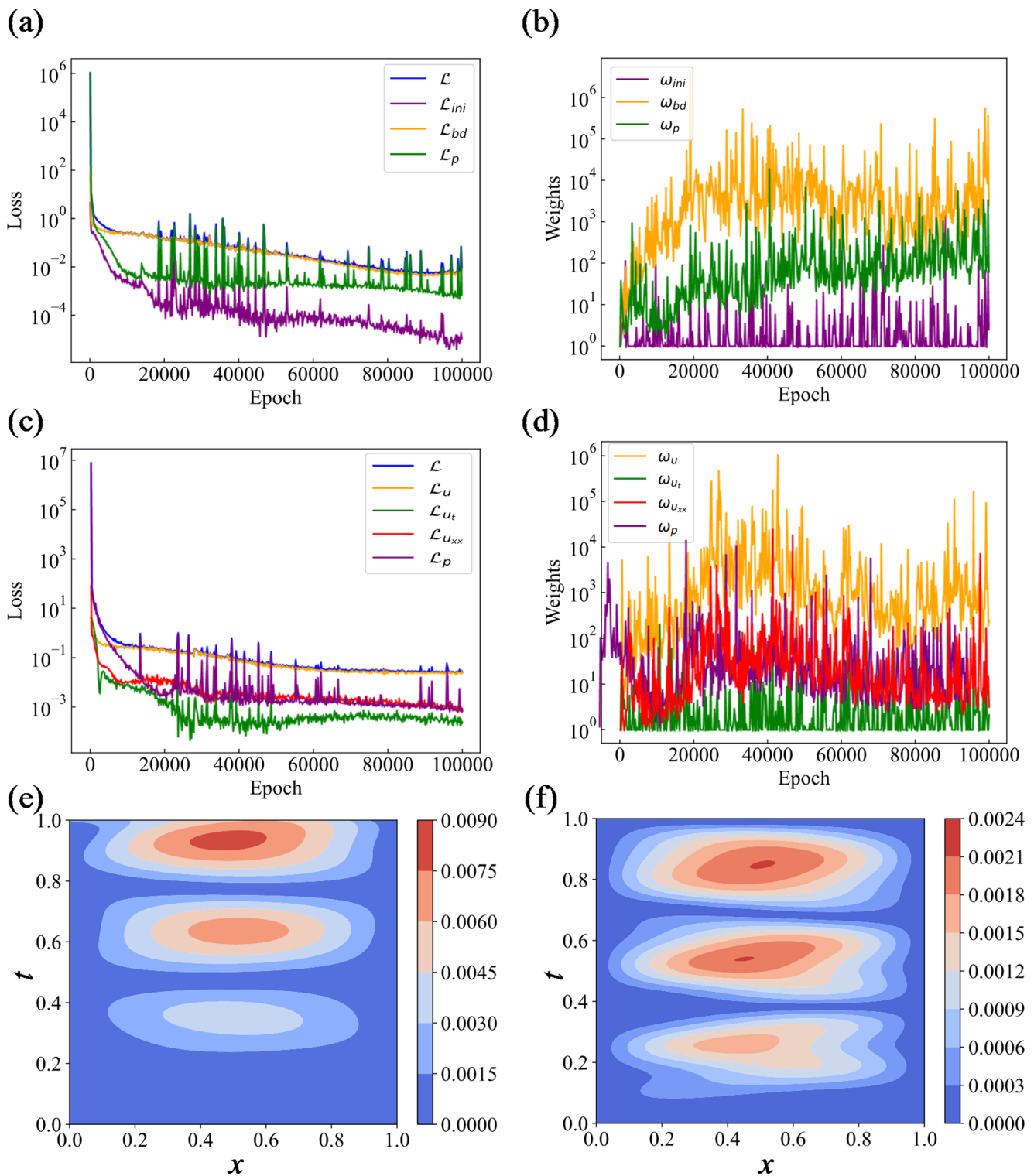


Fig. 14 Solution of adaptive weight PINN: (a) Each loss in the classic loss function; (b) Adaptive weights in the classic loss function; (c) Each loss in the reconstructed loss function; (d) Adaptive weights in

the reconstructed loss function; (e) Absolute errors guided by classical loss function; (f) Absolute errors guided by the reconstructed loss function

function adaptively weighted in Fig. 14 (a) and the loss function manually weighted in Fig. 8 (a) under the same training conditions, Fig. 14 (a) shows a much faster convergence than Fig. 9 (a). The main reason is that the adaptive weights can penalize and balance each loss component in real-time, facilitating the network to efficiently search for the global optimum. At the same time, it can also avoid the work of manually adjusting parameters and improve the generalization and usability of the network, as shown in Fig. 14 (b) and (d), where the weights are able to change in real-time with training. From the weight variation, it can be observed that the weights corresponding to larger loss or smaller loss reduction rates are also larger. For example, the adaptive weight corresponding to the loss is greater than those of others in Fig. 14(a). As the reduction rate of loss decreases in the later stages, the corresponding weight adaptively increases.

Additionally, it is necessary to further compare the additional computational cost before and after the network incorporates adaptive weights. In this paper, the network training was conducted 100,000 epochs using an Intel(R) Core (TM) i7-8700 CPU @ 3.20 GHz processor, keeping network parameters consistent both before and after introducing adaptive weights. The training times for the network with manually adjusted weights and adaptive weights were 5856.45 s and 5840.49 s, respectively, indicating a negligible time difference. However, the time consumed in manually selecting weight parameters far exceeds that of the adaptive weight network. The primary reason is that manual weight parameter selection requires continuous adjustment of weights during the training process to select the most suitable parameters, and the parameter selection varies depending on the problem, invisibly increasing the time required for successful training. In contrast, the introduction of adaptive weights reduces this time cost. Furthermore, the adaptive weight calculation in this paper introduces no additional training parameters or selection time to the network structure. The values required for adaptive weight computation are derived from intermediate values generated during the network training process, requiring only storage and simple dynamic calculations to obtain, without incurring significant additional time costs.

Response Solution for Cables with Actual Parameters

Coordinate Transformation

As mentioned above, PINNs designed for cable-like structures failed to solve the vibration responses of cables,

mainly due to the significant differences from the order-of-magnitude of physical parameters in the vibration PDE of actual cables. To address the challenges during training, this paper balances the unevenness of coefficients through coordinate transformation, which can convert each coefficient to 1.

In the new coordinate system, $\tilde{x} = x/L$, $\tilde{t} = t/T$, then $u_{xx} = u_{\tilde{x}\tilde{x}}/L^2$, $u_{xxxx} = u_{\tilde{x}\tilde{x}\tilde{x}\tilde{x}}/L^4$, $u_{tt} = u_{\tilde{t}\tilde{t}}/T^2$, substituting them into Eq. (5) yields,

$$u_{\tilde{x}\tilde{x}\tilde{x}\tilde{x}} + \frac{mL^4}{EIT^2}u_{\tilde{t}\tilde{t}} - \frac{HL^2}{ET}u_{\tilde{x}\tilde{x}} = 0, \tilde{x} \in [0, \frac{L}{L}], \tilde{t} \in [0, \frac{1}{T}] \quad (17)$$

When the coefficients in the aforementioned equation are normalized to 1, we can obtain $L = \sqrt{\frac{EI}{H}}$, $T = \sqrt{\frac{mEI}{H^2}}$. Therefore, the relationship between the new coordinates and the original coordinates can be expressed as follows,

$$\begin{cases} \tilde{x} = x\sqrt{\frac{H}{EI}} \\ \tilde{t} = t\sqrt{\frac{H^2}{mEI}} \end{cases} \quad (18)$$

Through utilizing Eq. (17), the vibration PDE of the actual cables will become Eq. (18) as follows.

$$u_{\tilde{x}\tilde{x}\tilde{x}\tilde{x}} + u_{\tilde{t}\tilde{t}} - u_{\tilde{x}\tilde{x}} = 0, \tilde{x} \in [0, l\sqrt{\frac{H}{EI}}], \tilde{t} \in [0, \sqrt{\frac{H^2}{mEI}}] \quad (19)$$

Similarly, the initial conditions under the new coordinate system can be transformed as,

$$u(\tilde{x}, 0) = \sin\left(\pi\tilde{x}\sqrt{\frac{EI}{H}}l\right), u_{\tilde{t}}(\tilde{x}, 0) = 0 \quad (20)$$

The boundary conditions are expressed in the new coordinate system as,

$$u(0, \tilde{t}) = u(l\sqrt{\frac{H}{EI}}, \tilde{t}) = u_{\tilde{x}\tilde{x}}(0, \tilde{t}) = u_{\tilde{x}\tilde{x}}(l\sqrt{\frac{H}{EI}}, \tilde{t}) = 0 \quad (21)$$

Substituting Eq. (19) and Eq. (20) into Eq. (9) yields the hard boundary function under the new coordinate system as follows.

$$\hat{u}_b(\tilde{x}, \tilde{t}, \theta) = \tilde{t} \cdot \hat{u}(\tilde{x}, \tilde{t}, \theta) (\tilde{x} - 0) \left(l\sqrt{\frac{H}{EI}} - \tilde{x}\right) + \sin\left(\pi\tilde{x}\sqrt{\frac{EI}{H}}l\right) \quad (22)$$

For comparative analysis, the analytical solution of Eq. (18) in the new coordinate system is given as follows, shown in Fig. 3 above in the original coordinate system.

$$u(\tilde{x}, \tilde{t}) = \sin\left(\pi\tilde{x}\sqrt{\frac{EI}{H}}l\right) \cos\left(\pi\sqrt{\frac{EI}{H^2}}\sqrt{(1 + \pi^2\frac{EI}{H^2})}\tilde{t}\right) \quad (23)$$

Response Solution

After the coordinate transformation, the definition domains of the vibration PDE are scaled in the new coordinate system. Therefore, it is necessary to normalize the inputs uniformly within the network, while the subsequent network can continue to track the normalization process without additional transformations during differentiation, improving the training efficiency and stability of the network. By adopting the improved PINN in Sect. [The improved PINN](#), Eq. (16) is solved in the new coordinate system. Since the hard boundary function can be expressed as Eq. (21), the hard boundary constraint PINN is preferred to solve the vibration response of cables with actual parameters. The loss function is determined by Eq. (12), while the weights are adjusted by the adaptive weight in Eq. (15). Taking the actual suspension cable in Sect. [PDE solution for cable vibration](#) as an example validates the accuracy and stability of the improved PINN, whose vibration response is firstly predicted under the new coordinate system, then the predicted vibration response in the original coordinate system can be obtained through the inverse coordinate transformation. The results are shown in Figs. 15 and 16.

Figure 15 shows changes of the loss function and adaptive weight during solving the vibration response of the actual cable by the improved PINN, where each loss rapidly converges and weights change in real-time to adjust the network. From the figures, it can be seen that due to the hard boundary constraint, the loss term corresponding to the hard boundary function remains zero at floating-point precision. Additionally, the adaptive weights proposed in this paper can be suitable for solving the vibration response of the actual cable, which can stimulate the network to further

optimize the loss function, leading to rapid convergence of the loss function and avoiding parameter adjustment.

From Fig. 16, it can be observed that the improved PINN accurately solves the free vibration response of the actual cable, and the solution is essentially consistent with the analytical solution in Fig. 3. Figure 16 (b) represents that the absolute errors between the predictions and true values are small, with the maximum absolute error being less than 0.0048. To observe the local fitting ability of the PINN, the variations of vibration response with respect to spatial and temporal variables are presented as shown in Fig. 16(c) and (d). When $t=0.5$, the variation of the predicted responses in space coincides almost with the true solution. Similarly, when $x=0.5$, the change of the predicted responses in time is completely close to the true solution. These results indicate that the improved PINN can accurately capture the vibration response of the actual cable. Therefore, the improved PINN in this paper can accurately solve the vibration response of actual cables and has a certain degree of generalization.

Discussion

Discussion of Parameters

Tension-bending Ratio

While exploring the PINN solution for the vibration response of actual cables, it is found that the PINNs designed for solving the response of cable-like structures are not suitable for solving that of cable structures. Therefore, comparing the vibration PDEs of cable-like structures and cable structures, it can be seen from the dimensionless process that the

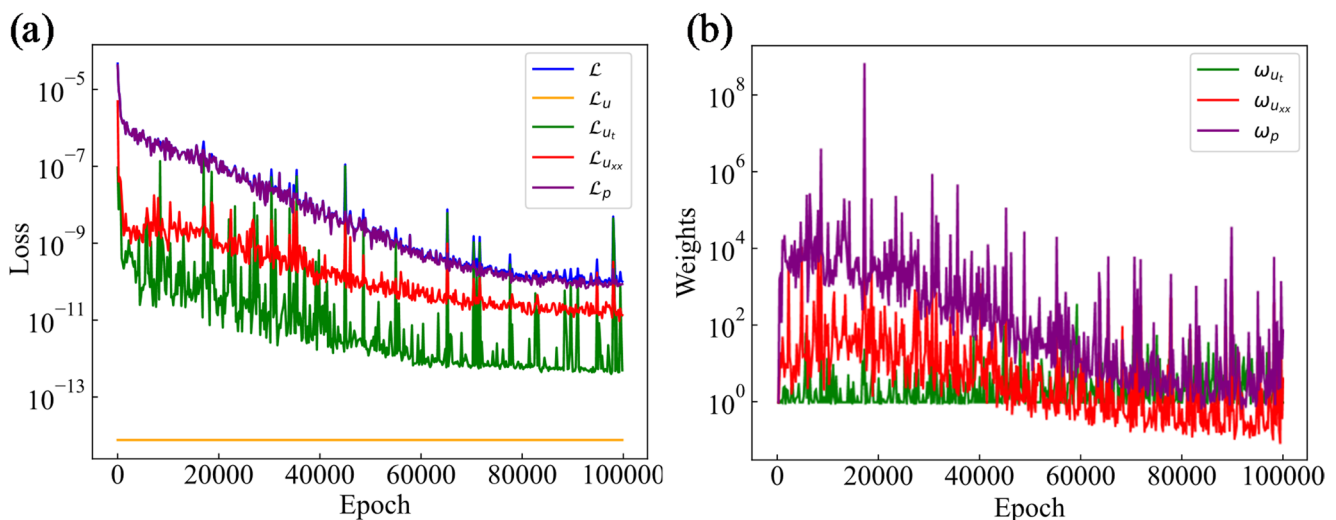


Fig. 15 Changes of loss function and weights during training: (a) Changes of each loss; (b) Weight changes

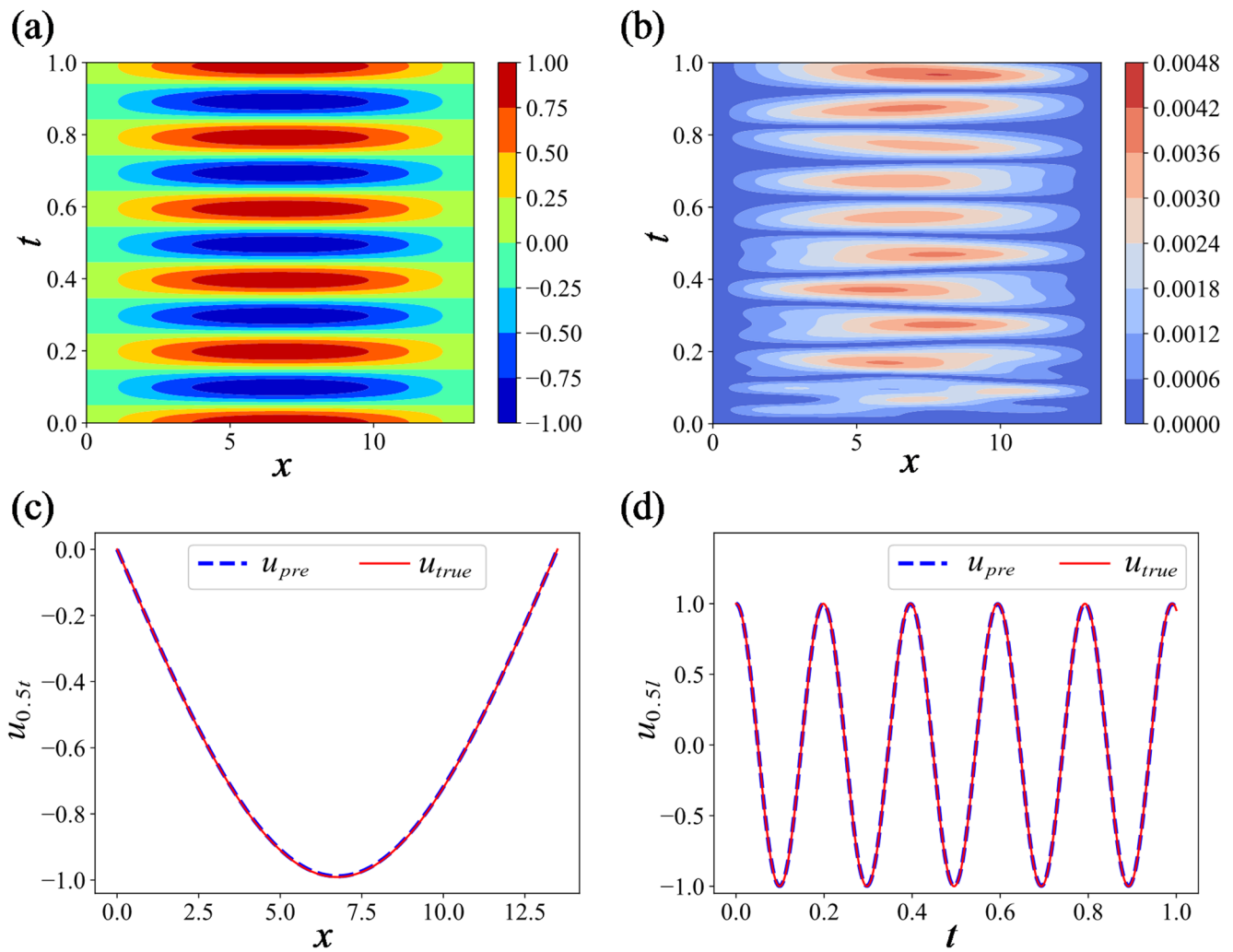


Fig. 16 Vibration responses of the actual cable: (a) Predicted vibration responses; (b) Absolute errors of predicted responses; (c) Response at $t=0.5$; (d) Responses at $x=0.5l$

primary distinction lies in the tension-bending ratio coefficient $\xi = \sqrt{Hl^2/EI}$, which is a dimensionless parameter proposed by Irvine [37] to describe the characteristics of cables. When $\xi \rightarrow 0$, it represents the single beam model of the cable-like structure. When $\xi \rightarrow \infty$, it represents the cable solution based on tensioned string theory. However, actual cables considering bending stiffness lies between these two. Therefore, it is necessary to further discuss the impact of ξ on network training.

For the same cable, the length l and bending stiffness EI remain constant, so the parameter ξ can be changed by varying the cable force, as the larger cable force produces the greater ξ . Therefore, this paper changes parameter ξ by varying the cable force of the actual suspension cable in Sect. [PDE solution for cable vibration](#). According to the literature [38], when $\xi > 210$, the response solution derived from the Euler-Bernoulli beam theory is similar to

that based on the tensioned string theory, indicating that the bending stiffness of the cable can be disregarded. For this reason, this article only needs to consider the case $\xi < 210$ when analyzing cables considering bending stiffness. Hence, this paper adjusts the cable force to make $\xi \in (0, 300)$, and the effect of ξ on the PINN solution can be observed from the error changes with respect to ξ . To intuitively understand the results, this paper introduces the relative L_2 -norm error $\mathfrak{R}$ and calculates its variation, as shown in Fig. 17.

$$\mathfrak{R} = \frac{\|u_{pre} - u\|_2}{\|u\|_2} = \frac{\sqrt{\sum (u_{pre} - u)^2}}{\sqrt{\sum u^2}} \quad (24)$$

Where, u_{pre} is the response displacement predicted by the PINN, u is the true solution while $\sum$ is the sum of the entire solution domains.

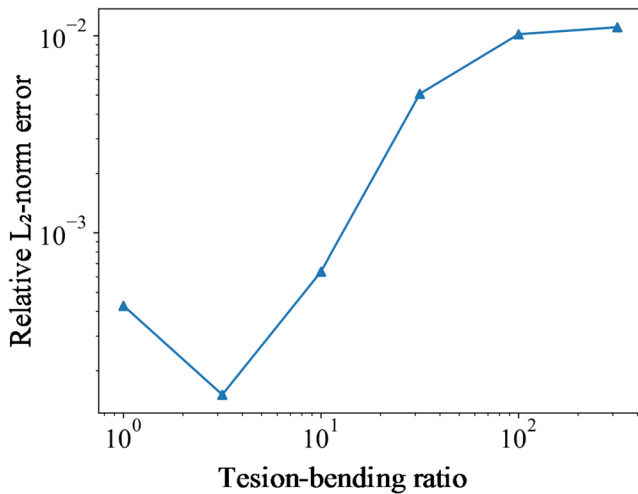


Fig. 17 Change of the relative L_2 -norm error with the tension-bending ratio

Figure 17 illustrates the variation of the relative L_2 -norm error between the PINN solution and the true solution under different tension-bending ratios. As the parameter increases, the relative L_2 -norm error also increases. Although the error in solving cable vibration has increased, it remains within an acceptable range, with the maximum of the relative L_2 -norm error less than 10^{-2} , which demonstrates high accuracy in engineering applications. From Fig. 17, it is seen that the improved PINN can obtain highly accurate solutions for the free vibration response of any cables with the tension-bending ratio ranging from 0 to 300. This also proves that the improved PINN in this paper can achieve the robust solution of actual cables with high generalization.

Moreover, the decrease in the tension-bending ratio ξ is equivalent to an increase in the stiffness of the cable itself, corresponding to a structure with high-frequency vibrations. Even for such high-frequency structures, the improved PINN can accurately solve the vibration response. Conversely, the increase in the tension-bending ratio corresponds to the increase of the cable force for the same cable, which means the relative L_2 -norm error increases with the increase of cable force. However, fundamentally, all coefficients of PDEs are normalized to 1 after the coordinate transformation, and the mathematical forms of PDEs are the same. Therefore, the parameter changes reflect the range changes of definition domains in new coordinates. With the increase of the parameter, the definition domains increases in the new coordinates accordingly. Therefore, the calculation errors of the vibration response predicted by the PINN gradually accumulate with the expansion of the definition domains.

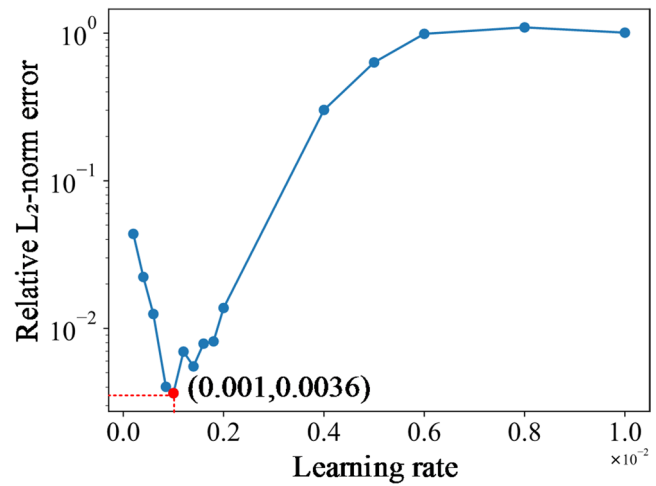


Fig. 18 Change of the relative L_2 -norm error with initial learning rate

Learning Rate

The learning rate is a crucial hyperparameter in neural networks, primarily responsible for controlling the step size of weight updates, determining the motion speed of the model through the parameter space. An excessively large learning rate may cause the model to diverge or oscillate around a local minimum, while an overly small learning rate may lead to slow convergence or even overfitting. Therefore, selecting an appropriate learning rate is essential for the successful training and generalization of neural networks [39]. This article adopts the Adam adaptive learning rate algorithm, which dynamically adjusts the learning rate according to the gradient and second-order moment estimation of each parameter. However, the initial learning rate needs to be adjusted to determine the optimal one. Therefore, this paper employs the Cosine Annealing LR algorithm to specifically adjust the initial learning rate of the optimizer, which contributes to enhancing the efficiency and generalization of the model. For PINNs solving the dynamic response of cables, this paper has found that the initial learning rate directly affects the solution precision. Therefore, the paper investigates the effects of different initial learning rates on PINN solutions, as shown in Fig. 18.

From Fig. 18, it can be observed that using different initial learning rates in the network yields varying computational effects. However, as the initial learning rate decreases, the relative L_2 -norm error of solutions cannot consistently decrease. When the initial learning rate is greater than 0.001, the relative error decreases as the initial learning rate decreases. This indicates that as the step size of parameter updates decreases, it is beneficial for the model to move towards a more optimal gradient descent path during

training, avoiding rapid jumps over local optima or oscillations, thereby enabling the model to converge faster to the global optimum. However, when the initial learning rate is less than 0.001, the relative error will increase instead as the initial learning rate decreases. This is mainly because excessively reducing the step size of parameter updates can lead to the model getting stuck in local optima while finely adjusting the network parameters. When the initial learning rate is 0.001, the relative error is minimized and the solution accuracy is optimal, which is adopted in this paper. Therefore, for PINNs solving the cable vibration responses, the initial learning rate between 0.0005 and 0.002 can ensure the network optimal fitting performance and stability. At present, setting initial learning rates relies heavily on manually adjusting parameters, so it is necessary to further research for setting the initial learning rate of the network based on specific problem characteristics in an informed and systematic manner.

Analysis of the Accuracy

With the introduction of the improved PINN, it can be seen from its error diagrams of solving the vibration PDEs that the absolute errors accumulate with the increase of time. As shown in the aforementioned Figs. 11 and 13, the absolute error sum in the spatial domain increases as time marches. To understand the variation of the errors with time, this paper solves the free vibration response of the actual cable in Sect. [PDE solution for cable vibration](#) during different times by the hard boundary constraint PINN. The changes of the error over time are shown in Fig. 19.

Figure 19 represents that the improved PINN solves the vibration response of the cable at different time thresholds, where Fig. 19 (a)–(e) show the varying curves of predicted responses at $x=0.5l$ within 1 s ~ 5 s, and Fig. 19 (f) illustrates the variation of the relative L_2 -norm error between network predictions and true solutions with time. From Fig. 19 (a)–(c), it can be observed that the deviation between predicted and true values grows with time increasing, although the predictions still maintain a certain accuracy. However, as time marches for $t>4$ s, as shown in Fig. 19 (d) and (e), the deviation between predicted and true values becomes significant and cannot be ignored. Moreover, it is clear from Fig. 19 (e) that the fitting performance of the network deteriorates over time, even at the training beginning. In addition, it is observed from Fig. 19 (f) that the relative L_2 -norm error increases with time and sharply escalates after 4 s, where the network will fail. Therefore, computational errors in the network accumulate and even lead to network failure, as time progresses.

In this paper, when PINNs solve the vibration response of actual cables, the pre-processing of coordinate

transformation is required, which normalizes the PDE coefficients at the cost of expanding the time domain. Therefore, it is necessary to divide the time in the new coordinate system to ensure training accuracy and avoid error accumulation over time. Some previous studies have implemented certain improvements for the error accumulation effect in PINNs. Chen et al. [28] divided the entire time domain and employed transfer learning, which solved structural vibration over a long time by AT-PINN. The latest study [40] proposed CEENs that divide the time domain into non-overlapping subintervals and assign neural networks for them to obtain long-time solutions. However, these time divisions mainly rely on empirical judgment. Comparing the vibration phenomenon of the absolute error sum and the actual cable vibration mentioned above, it can be obtained that the vibration period of the error is consistent with that of the cable. Therefore, while segmenting time in PINNs, the vibration period of the cable should also be considered. Certainly, the underlying principles governing this relationship warrant further investigation.

Conclusion

Due to the differences in dynamic analysis between cable-like structures and standard cable, as well as the limitations of networks, existing PINNs designed for cable-like structures are not directly applicable to general cable vibration problems. This paper systematically analyzes the key challenges in extending traditional PINNs to real cables through case studies. An improved PINN framework, incorporating optimized training strategies, is proposed to accurately capture the free vibration response of arbitrary cables with enhanced generalization. The key findings are presented below.

- (1) This paper proposes a comprehensive preferred strategy of hard-soft boundary constraints for reinforcing boundary conditions, ensuring accurate vibration response solutions for cables using both improved soft boundary constraint PINNs and hard boundary PINNs. When the hard boundary function can be constructed, hard boundary constraint PINNs are preferred for solving the PDEs. In cases where such a function is infeasible, the improved soft boundary constraint PINNs serve as a robust alternative.
- (2) The study shows that the Sin function is more suitable as the activation function for PINNs in solving the vibration PDEs of cables. Compared with the Tanh-activated PINN, the periodicity of the Sin function enables the

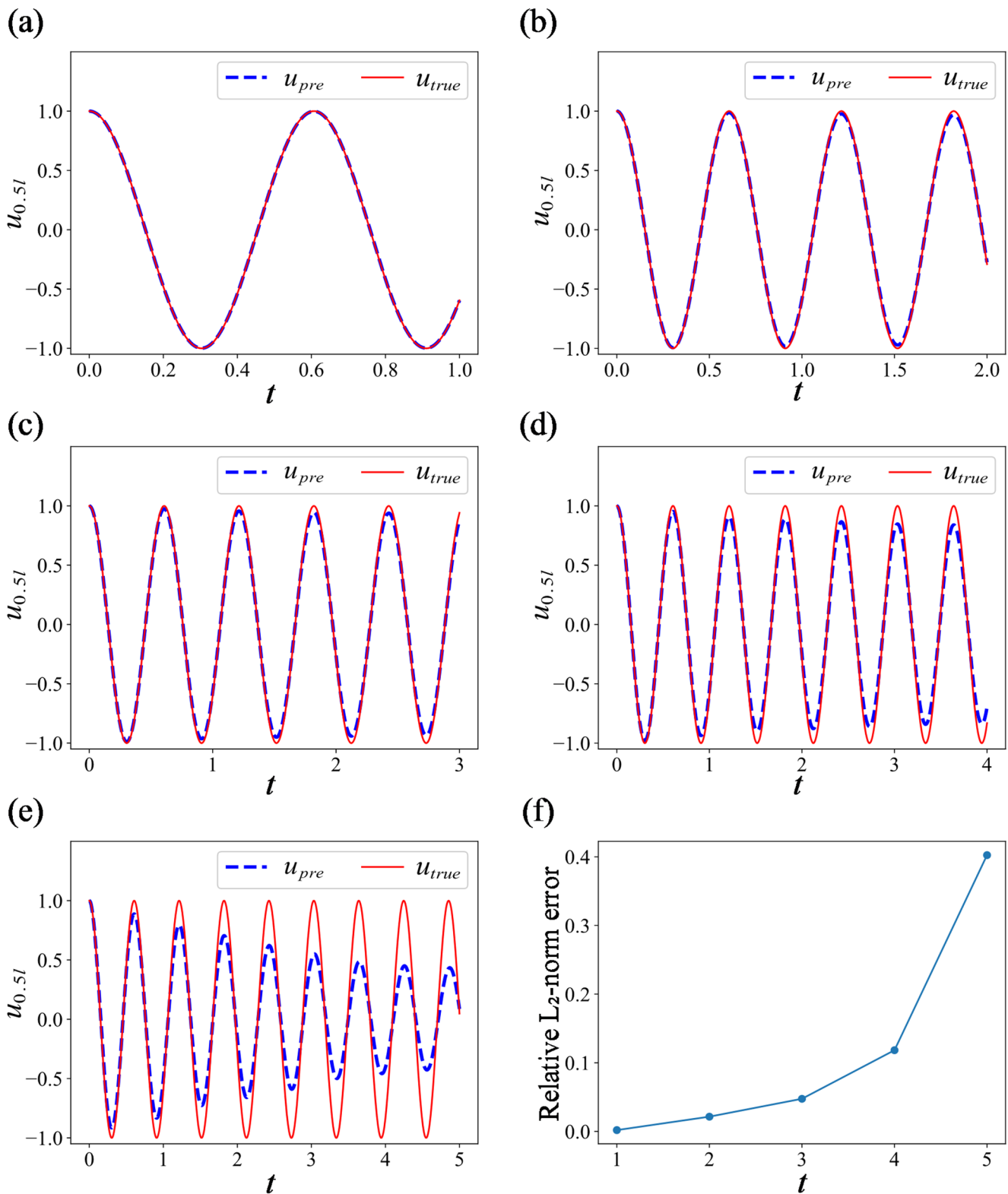


Fig. 19 Vibration responses at different times: (a) Vibration responses within 1 s; (b) Vibration responses within 2 s; (c) Vibration responses within 3 s; (d) Vibration responses within 4 s; (e) Vibration responses within 5 s; (f) Change of the relative L_2 -norm error with time

network to more efficiently and accurately approximate the vibration responses of cables.

- (3) This paper proposes hierarchical gradient loss function based on the partial derivatives order, precisely penalizing the internal component of each loss. In addition, the proposed adaptive weight avoids the repeated and invalid manual adjustments to the loss function weights, thereby improving the generalization and usability of the network.

Through parameter analysis and error evaluation in solving the vibration responses of actual cables by PINNs, it is found that the errors increase with the tension-bending ratio of cables, essentially reflecting that the errors gradually accumulate with the expansion of the PDEs definition domains. Additionally, analysis of network parameters further reveals that the learning rate significantly affects solution accuracy, with an optimal initial range recommended between 0.0005 and 0.002. Furthermore, PINN solution errors grow over longer prediction times, with their periodic variation aligning with the vibration period of cables. Therefore, further improvements in the network architecture and training strategies should be needed in future research to achieve accurate solutions for cable vibration at any time.

Acknowledgements This work was supported by the National Nature Science Foundation of China (grant number 51878490); the Fundamental Research Funds for the Central Universities (grant number 20210205); the National key R&D Program of China (grant number: 2017YFF0205605); and China Railway Engineering Design and Consulting Group Co., Ltd. technology development project “research and development of bridge management and maintenance system based on perceptual neural network”.

Funding Open access funding provided by The Hong Kong Polytechnic University.

Declarations

Conflict of Interest The authors declare that they have no conflict of interest.

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. Dan D, Sun L, Guo Y, Cheng W (2015) Study on the mechanical properties of stay cable HDPE sheathing fatigue in dynamic bridge environments. *Polymers* 7(8):1564–1576
2. Fei H, Danhui D, Wei C, Jubao Z (2020) A novel analysis method for damping characteristic of a type of double-beam systems with viscoelastic layer. *Appl Math Model* 80:911–928
3. Han F, Dan D, Cheng W (2018) An exact solution for dynamic analysis of a complex double-beam system. *Compos Struct* 193:295–305
4. Han F, Dan D, Cheng W (2018) Extension of dynamic stiffness method to complicated damped structures. *Comput Struct* 208:143–150
5. Fei H, Zichen D, Danhui D (2021) Exact dynamic analysis of multi-segment cable systems. *Mech Syst Signal Process* 146:107053
6. Ozdemir H (1979) A finite element approach for cable problems. *Int J Solids Struct* 15(5):427–437
7. Tabatabai H, Mehrabi AB (1999), February Vibration suppression measures for stay cables. In *Society for Experimental Mechanics, Inc., 17th International Modal Analysis Conference*. (Vol. 2, pp. 1237–1243)
8. Su X, Liu G, Kang H, Zhang W, Liu T, Wang Y (2025) Study on dynamic behaviors of the cable-cross-tie system considering the vibration of the bridge deck. *J Vib Eng Technol* 13(1):126
9. Abad MSA, Shooshtari A, Esmaili V, Riabi AN (2013) Non-linear analysis of cable structures under general loadings. *Finite Elem Anal Des* 73:11–19
10. Song C, Xiao R, Sun B, Wang Z, Zhang C (2023) Cable force optimization of cable-stayed bridges: a surrogate model-assisted differential evolution method combined with B-spline interpolation curves. *Eng Struct* 283:115856
11. Kutyniok G (2022) The mathematics of artificial intelligence. In *Proceedings of the International Congress of Mathematicians*
12. Saini R (2025) A review on artificial neural networks for structural analysis. *J Vib Eng Technol* 13(2):142
13. Cuomo S, Di Cola VS, Giampaolo F, Rozza G, Raissi M, Piccialli F (2022) Scientific machine learning through physics-informed neural networks: where we are and what's next. *J Sci Comput* 92(3):88
14. Berg J, Nyström K (2018) A unified deep artificial neural network approach to partial differential equations in complex geometries. *Neurocomputing* 317:28–41
15. Long Z, Lu Y, Ma X, Dong B (2018), July Pde-net: Learning pdes from data. In *International conference on machine learning* (pp. 3208–3216). PMLR
16. Long Z, Lu Y, Dong B (2019) Pde-net 2.0: learning pdes from data with a numeric-symbolic hybrid deep network. *J Comput Phys* 399:108925
17. Faysal A, Keng NW, Lim MH (2021) Ensemble Augmentation for Deep Neural Networks Using 1-D Time Series Vibration Data. *arXiv arXiv preprint arXiv:2108.03288*.
18. Raissi M, Perdikaris P, Karniadakis GE (2019) Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J Comput Phys* 378:686–707
19. Zha WS, Li DL, Shen LH, Zhang W, Liu XL (2022) Review of neural Network-based methods or solving partial differential equations. *Chin J Theoretical Appl Mech* 54(3):543–556 (in Chinese)
20. Bai J, Rabczuk T, Gupta A, Alzubaidi L, Gu Y (2023) A physics-informed neural network technique based on a modified loss function for computational 2D and 3D solid mechanics. *Comput Mech* 71(3):543–562

21. He G, Zhao Y, Yan C (2023) A physics-informed generative adversarial network framework for multiaxial fatigue life prediction. *Fatigue Fract Eng Mater Struct* 46(10):4036–4052
22. Cai S, Mao Z, Wang Z, Yin M, Karniadakis GE (2021) Physics-informed neural networks (PINNs) for fluid mechanics: A review. *Acta Mechanica Sinica* 37(12):1727–1738.1
23. Yuan FG, Zargar SA, Chen Q, Wang S (2020) Machine learning for structural health monitoring: challenges and opportunities. *Sensors and smart structures technologies for civil, mechanical, and aerospace systems 2020* 11379:1137903
24. Kapoor T, Wang H, Núñez A, Dollevoet R (2023) Physics-informed neural networks for solving forward and inverse problems in complex beam systems. *IEEE Transactions on Neural Networks and Learning Systems*
25. Kapoor T, Wang H, Núñez A, Dollevoet R (2024) June Physics-informed machine learning for moving load problems. In *Journal of Physics: Conference Series*, Vol. 2647 (No. 15). IOP Publishing, p. 152003
26. Kapoor T, Wang H, Núñez A, Dollevoet R (2024) Transfer learning for improved generalizability in causal physics-informed neural networks for beam simulations. *Eng Appl Artif Intell* 133:108085
27. Söyleyici C, Ünver HÖ (2025) A physics-informed deep neural network based beam vibration framework for simulation and parameter identification. *Eng Appl Artif Intell* 141:109804
28. Chen Z, Lai SK, Yang Z (2024) AT-PINN: advanced time-marching physics-informed neural network for structural vibration analysis. *Thin-Walled Struct* 196:111423
29. Zhu M, Wang J, Hu X, Dong S (2025) Machine learning-based multi-objective prestress optimization framework of suspend dome structure and case study. *Eng Struct* 322:118987
30. Kabasi S, Marbaniang AL, Ghosh S (2024) Form-finding of tensile membrane structures with strut and anchorage supports using physics-informed machine learning. *Eng Struct* 299:117093
31. Huang Y, Tang J, Jiang J, Li Y, Liu M, Chen H, Cao D (2025) Physics-informed neural networks identification and reinforcement learning control for a nonlinear vibration system. *J Vib Eng Technol* 13(5):301
32. Humar J (2012) *Dynamics of structures*. CRC press
33. Xu X (2008) Estimation and testing of suspension cable forces considering the influence of boundary conditions (Doctoral dissertation, Central South University). (in Chinese)
34. Mishra S, Molinaro R (2023) Estimates on the generalization error of physics-informed neural networks for approximating PDEs. *IMA J Numer Anal* 43(1):1–43
35. Lu L, Pestourie R, Yao W, Wang Z, Verdugo F, Johnson SG (2021) Physics-informed neural networks with hard constraints for inverse design. *SIAM J Sci Comput* 43(6):B1105–B1132
36. Wang S, Yu X, Perdikaris P (2022) When and why PINNs fail to train: a neural tangent kernel perspective. *J Comput Phys* 449:110768
37. Irvine HM (1981) *Cable structures (Structural Mechanics)*. MIT Press
38. Ren WX, Chen G (2005) Practical formulas to determine cable tension by using cable fundamental frequency. *China Civil Eng J* 38(11):26–31 (in Chinese)
39. Lobacheva E, Pockonechnyy E, Kodryan M, Vetrov D (2023) Large learning rates improve generalization: but how large are we talking about? *arxiv preprint arxiv:2311.11303*
40. Jung J, Kim H, Shin H, Choi M (2024) CEENs: causality-enforced evolutionary networks for solving time-dependent partial differential equations. *Comput Methods Appl Mech Eng* 427:117036

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