

# A physics-informed neural network framework for laminated composite plates under bending

Weixi Wang, Huu-Tai Thai <sup>\*</sup>

*Department of Infrastructure Engineering, The University of Melbourne, Parkville, VIC 3010, Australia*

## ARTICLE INFO

**Keywords:**  
physics-informed machine learning  
artificial neural network  
classical laminated plate theory  
bending

## ABSTRACT

The use of machine learning in the field of structural engineering is becoming more common. However, the high dependence of traditional purely data-driven models on the size and quality of the database has posed challenges to the practical application of machine learning. Applying physics-informed machine learning can achieve accurate predictions while reducing the need for extensive input data. This study develops a Physics-Informed Neural Network (PINN) framework to predict the bending behaviors of laminated composite plates. In this framework, the Classical Laminated Plates Theory (CLPT) is incorporated as the physical constraint, and the loss function is formulated based on the energy method. The machine learning prediction results were validated with the CLPT analytical solutions and Finite Element Method (FEM) results, which were sourced from existing literature. These validations demonstrate that the PINN framework achieves satisfactory bending behavior predictions, potentially serving as a promising alternative.

## 1. Introduction

As the most vigorous branch of artificial intelligence, machine learning has been integrated into the research trends across various academic fields [1]. It achieves great success in addressing complex natural systems characterized by multiple variables, including but not limited to information science, life science, materials science, and social science [2–5]. In the field of structural engineering, machine learning also holds considerable promise. Benefiting from the low-cost data accumulation and collection in this big data era, data-driven approaches based on machine learning can present an alternative to the traditional physics-driven approaches by fitting extensive data [6,7]. These approaches provide an opportunity to extract complex relationships directly from data without requiring a deep comprehension of the system being studied [8,9]. Since Adeli and Yeh [10] used Artificial Neural Networks (ANNs) in 1989 to design steel beams, there has been growing research exploring the application of machine learning in structural behavior modeling [11,12], mechanical property analysis [13,14], structural health monitoring [15–17], fire resistance predictions [18], etc. These innovative machine learning applications demonstrate the efficacy of data-driven approaches in predicting structural systems, even in cases where the fundamental physics laws are not fully understood [19]. Compared with mechanics-based models, machine learning tends

to be more accurate and resilient, especially when abundant data is available. Moreover, it can significantly save time and effort in addressing complex nonlinear structural systems [20,21].

However, machine learning comes with inherent limitations and challenges. Like other statistical approaches, such as regression analysis and curve fitting, it relies on high-quality datasets to ensure the efficiency and accuracy of predictions [22,23]. When the training data is limited or noisy, machine learning algorithms, especially deep learning networks, often lack extrapolation ability and may overfit [24]. Additionally, the decision-making process in purely data-driven models lacks transparency. Unlike physics-driven models, which describe system behavior through governing equations derived from first principles, the "black box" nature of machine learning complicates interpretation and may inconsistently align with established physical laws [25,26]. Therefore, in cases where traditional machine learning models encounter difficulties in offering reliable predictions from available data, integrating the complexity of the physical system into the modeling strategy is a promising approach [27]. Fig. 1 schematically illustrates the relationship between known physical laws and available training data. By informing machine learning models of the physical information, such as governing rules and domain knowledge, coupled models can acquire strong theoretical constraints and inductive biases, thereby reducing reliance on data and enhancing their interpretability [28,29].

\* Corresponding author at: Department of Infrastructure Engineering, The University of Melbourne, Rm 413, Parkville, VIC 3010, Australia.  
E-mail address: [tai.thai@unimelb.edu.au](mailto:tai.thai@unimelb.edu.au) (H.-T. Thai).

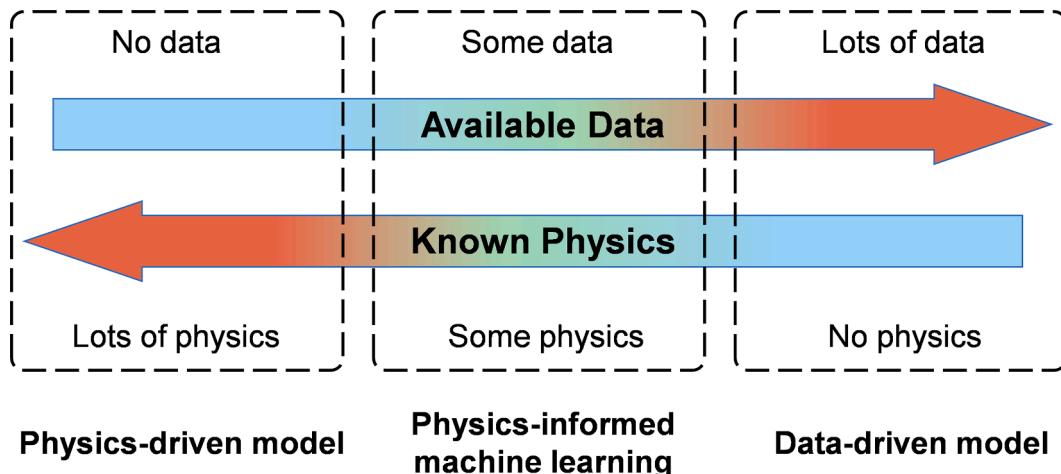


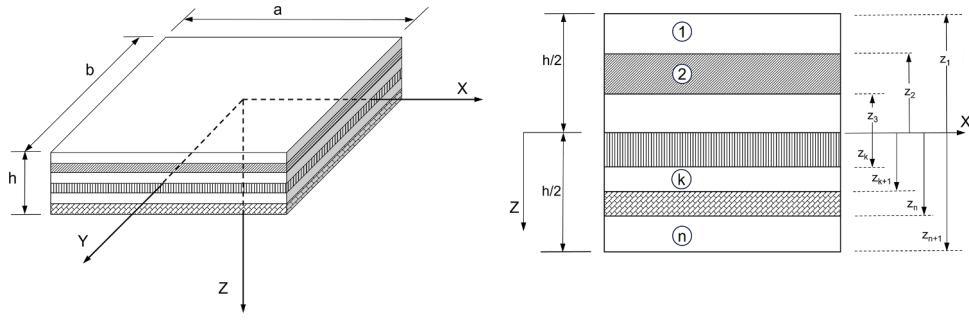
Fig. 1.. The relationship between known physical laws and available training data.

In this context, there is growing interest in combining flexible data-driven approaches with rigorous physical laws to leverage both strengths. For example, Hao et al. [30] embedded physical information parameters into the machine learning model to achieve probabilistic prediction and uncertainty assessment of notch fatigue life in aerospace materials. Wang et al. [31] incorporated physical information to design the architecture of neural networks and proposed a Turbulent-Flow Net (TF-Net) framework to enhance the accuracy and interpretability of deep learning models for complex turbulent flows. Among a range of physics-informed machine learning frameworks, Physics-Informed Neural Networks (PINNs) stand out for their widespread application and significant attention they have received.[32]. Unlike the frameworks mentioned above, PINNs incorporate physical information directly into the model training process rather than into the model inputs/outputs or architecture. This class of frameworks typically embed the Partial Differential Equations (PDEs) and Boundary Conditions (BCs) into the loss function through automatic differentiation, employing neural networks as universal approximators for a desired solution [33–35]. Since Raissi et al. [36] used PINNs to solve nonlinear PDEs like one-dimensional Burgers' equation and Navier-Stokes equations in two dimensions, many extensions and variants based on PINN frameworks have been proposed and applied [37]. For example, Jin et al. [38] successfully used PINNs to solve the incompressible Navier-Stokes equations under turbulent conditions, highlighting the capability of PINNs in modeling complex fluid dynamics. Chiu et al. [39] developed the CAN-PINN framework, combining the advantages of automatic and numerical differentiation. The proposed coupled differentiation approach optimizes the balance between computational efficiency and differentiation accuracy.

Building on these successes in fluid dynamics [40], researchers like Haghighe et al. [41] explored the application of PINNs in solid mechanics, expanding their utility to a broader range of physical problems. Recently, Li et al. [42] and Vahab et al. [43] successfully applied the PINNs to elasticity and elastic plate theory for mechanical responses and parameter identification. Their studies demonstrated satisfactory accuracy and computational efficiency in characterizing the behavior of elastic plates using PINN based on the classical plate theory. Zhang et al. [44] combined the domain continuity of PINN solutions with its Automatic Differentiation (AD) capability to propose a general PINN framework based on mesh-free method, addressing geometric identification problems in continuum solid mechanics and demonstrating its efficacy across 2D linear elasticity, hyperelasticity, and plasticity models. These PINN applications rely on strong-form governing equations to directly solve high-order PDEs, ensuring strict satisfaction of the known mathematical and physical principles at every point within the solution domain [45–47]. Another class of PINN frameworks transforms

the PDEs in the loss function into their integral weak forms, enabling the neural network to approximate the weak-form solutions while reducing regularity requirements and computational cost [48–50]. For example, Kharazmi et al. [51] incorporated the variational principle into PINNs to handle complex boundary conditions and nonlinear behavior in solid mechanics, reduce the order of involved differentiation operators and enhance their ability to solve complex PDEs. Similarly, Liu and Wu [52] employed domain decomposition to evaluate the variational form of governing equation residuals, efficiently identifying homogeneous and heterogeneous material distributions from limited observational data. Their PINN model demonstrated higher accuracy than the FEM when solving forward problems in solid mechanics. Bai et al. [53] proposed a modified loss function using the Least Squares Weighted Residual (LSWR) method and validated its effectiveness in addressing elasticity and plasticity problems through numerical experiments. Inspired by the nonlocal integral operator-based peridynamic theory, Ning et al. [54] developed a PINN framework that minimizes peridynamic potential energy to accurately simulate crack initiation and propagation in quasi-brittle plates without relying on any labeled datasets.

For laminated composite plates, which are widely used in civil, aerospace, automotive, shipbuilding, and other industries [55], there is also a growing trend of using machine learning techniques to analyze the mechanical behavior of orthotropic laminates [56]. For example, Furtado et al. [57] utilized common machine learning algorithms, including Extreme Gradient Boosting and Random Forest algorithms, to predict the notched strength of composite laminates, while Mishra et al. [58] combined the FEM with machine learning regression algorithms to discuss the buckling behavior of composite laminates. Reiner. et al. [59] proposed a data-rich framework combined with machine learning algorithms and validated the effectiveness of Feedforward Neural Networks (FNN) and Recurrent Neural Networks (RNN) in accurately characterizing and simulating compressive damage in laminated composite plates. Most current machine learning applications on laminates rely on purely data-driven approaches, which typically require extensive experimental or simulation datasets to train the models [60,61]. However, the analysis of laminate deformation behavior is supported by well-established mechanical theories. The governing PDEs can be straightforwardly derived from widely used theories such as CLPT or First-Order Shear Deformation Theory (FSDT), enabling the PINN framework, proven effective for isotropic materials, to be extended to orthotropic composite laminates. By embedding governing equations, BCs, and physical constraints such as conservation of energy directly into the neural network training process, the data-efficient PINN framework can accurately capture the complex mechanical behavior of laminates. Furthermore, it effectively overcomes the inherent limitations of analytical methods for solving PDEs, presenting a promising



**Fig. 2.** The global coordinate system and layer numbering scheme used for a rectangular laminated plate.

mesh-free alternative to traditional numerical methods.

In this study, we incorporated the CLPT into the model training process and developed a PINN framework capable of predicting the bending behavior of laminated composite plates. Our framework employs an energy-based approach to formulate the loss function, characterizing out-of-plane deformation by minimizing total potential energy. By comparing the PINN-predicted results with the analytical solutions and FEM results provided in the studies of Reddy [62], Bhaskar et al. [63,64], and Altunsaray et al. [65], it is evident that our computational framework provides accurate predictions for the deflection and stress distribution of laminated composite plates while demonstrating excellent applicability and scalability.

## 2. Classical laminated plate theory

The CLPT is incorporated into the training process as the physical constraint in this PINN framework. As an extension of the classical plate theory on composite laminates, the CLPT is also required to satisfy the Kirchhoff's Hypothesis [66]. Fig. 2 shows the global coordinate system and the layer numbering scheme used for a rectangular laminated plate. Assume the three displacement components along  $x$ ,  $y$  and  $z$  directions are  $u$ ,  $v$  and  $w$ , respectively, while the corresponding displacements of the mid-plane are denoted as  $u_0$ ,  $v_0$  and  $w_0$ . Based on the Kirchhoff's Hypothesis, the transverse normal and shear strains in the plate are all assumed to be zero, that is,  $\varepsilon_{xz} = \varepsilon_{yz} = \varepsilon_{zz} = 0$ .

The strain-displacement relationship for a laminated plate can be expressed as:

$$\begin{aligned} \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{xy} \end{Bmatrix} &= \begin{Bmatrix} \varepsilon_{xx}^0 \\ \varepsilon_{yy}^0 \\ \varepsilon_{xy}^0 \end{Bmatrix} + z \begin{Bmatrix} \varepsilon_{xx}^1 \\ \varepsilon_{yy}^1 \\ \varepsilon_{xy}^1 \end{Bmatrix} \\ &= \begin{Bmatrix} \frac{\partial u_0}{\partial x} + \frac{1}{2} \left( \frac{\partial w_0}{\partial x} \right)^2 \\ \frac{\partial v_0}{\partial y} + \frac{1}{2} \left( \frac{\partial w_0}{\partial y} \right)^2 \\ \frac{\partial u_0}{\partial x} + \frac{\partial v_0}{\partial y} + \left( \frac{\partial w_0}{\partial x} \right) \left( \frac{\partial w_0}{\partial y} \right) \end{Bmatrix} + z \begin{Bmatrix} -\frac{\partial^2 w_0}{\partial x^2} \\ -\frac{\partial^2 w_0}{\partial y^2} \\ -2 \frac{\partial^2 w_0}{\partial x \partial y} \end{Bmatrix} \end{aligned} \quad (1)$$

where  $(\varepsilon_{xx}^0, \varepsilon_{yy}^0, \varepsilon_{xy}^0)$  represent the membrane strains, and  $(\varepsilon_{xx}^1, \varepsilon_{yy}^1, \varepsilon_{xy}^1)$  represent the flexural strains.

Assume each layer in laminated plates is orthotropic and obeys Hooke's law. The force and moment resultants are obtained by integrating the stress components  $(\sigma_{xx}, \sigma_{yy}, \sigma_{xy})$  of each layer throughout the laminate thickness. Thus, the constitutive equation for the laminated plates is expressed as:

$$\begin{aligned} \begin{Bmatrix} \{N\} \\ \{M\} \end{Bmatrix} &= \begin{Bmatrix} N_{xx} \\ N_{yy} \\ M_{xy} \\ M_{yy} \\ M_{xy} \end{Bmatrix} \\ &= \begin{Bmatrix} A_{11} & A_{12} & A_{16} & B_{11} & B_{12} & B_{16} \\ A_{12} & A_{22} & A_{26} & B_{12} & B_{22} & B_{26} \\ A_{16} & A_{26} & A_{66} & B_{16} & B_{26} & B_{66} \\ B_{11} & B_{12} & B_{16} & D_{11} & D_{12} & D_{16} \\ B_{12} & B_{22} & B_{26} & D_{12} & D_{22} & D_{26} \\ B_{16} & B_{26} & B_{66} & D_{16} & D_{26} & D_{66} \end{Bmatrix} \begin{Bmatrix} \frac{\partial u_0}{\partial x} + \frac{1}{2} \left( \frac{\partial w_0}{\partial x} \right)^2 \\ \frac{\partial v_0}{\partial y} + \frac{1}{2} \left( \frac{\partial w_0}{\partial y} \right)^2 \\ \frac{\partial u_0}{\partial x} + \frac{\partial v_0}{\partial y} + \left( \frac{\partial w_0}{\partial x} \right) \left( \frac{\partial w_0}{\partial y} \right) \\ -\frac{\partial^2 w_0}{\partial x^2} \\ -\frac{\partial^2 w_0}{\partial y^2} \\ -2 \frac{\partial^2 w_0}{\partial x \partial y} \end{Bmatrix} \\ &= \begin{bmatrix} [A][B] \\ [B][D] \end{bmatrix} \begin{Bmatrix} \{\varepsilon^0\} \\ \{\varepsilon^1\} \end{Bmatrix} \end{aligned} \quad (2)$$

where  $A_{ij}$  are extensional stiffness coefficients,  $B_{ij}$  are bending stiffness coefficients, and  $D_{ij}$  are bending-extensional coupling stiffness coefficients.

Whether using analytical methods or numerical simulations, the fundamental step in solving the bending equilibrium of laminated plates lies in obtaining the governing equations under specified BCs and load cases. A common derivation approach is based on the principle of virtual work. Assume there is a virtual displacement  $(\delta u_0, \delta v_0, \delta w_0)$  in the in-plane displacement field  $(u_0, v_0, w_0)$ , the system virtual work can be written as:

$$\delta\Pi \equiv \delta(U + V) = \delta U + \delta V = 0 \quad (3)$$

where  $\delta U$  is the internal virtual work of the laminate, while  $\delta V$  is the external virtual work. These are defined as:

$$\begin{aligned} \delta U &= \int_{\Omega} \int_{-h/2}^{h/2} (\sigma_{xx} \delta \varepsilon_{xx} + \sigma_{yy} \delta \varepsilon_{yy} + 2\sigma_{xy} \delta \varepsilon_{xy}) dz dx dy \\ \delta V &= - \int_{\Omega} q \delta w dx dy - \int_{\Gamma_0} \int_{-h/2}^{h/2} (\sigma_{nn} \delta u_{0n} + \sigma_{ns} \delta u_{0s} + \sigma_{nz} \delta w_0) dz ds \end{aligned} \quad (4)$$

where  $q$  is the external load on the laminate, and  $(\sigma_{nn}, \sigma_{ns}, \sigma_{nz})$  are the normal stress, tangential stress and transverse stress components on the portion  $\Gamma_0$  of the boundary  $\Gamma$ . Similarly,  $(\delta u_{0n}, \delta u_{0s}, \delta w_0)$  are the corresponding virtual displacements along the directions of the normal, tangential, and transverse axes, respectively. By substituting Eq. (2) into Eq. (3), the equilibrium equation for the laminate can be derived as:

### Construct Neural Network

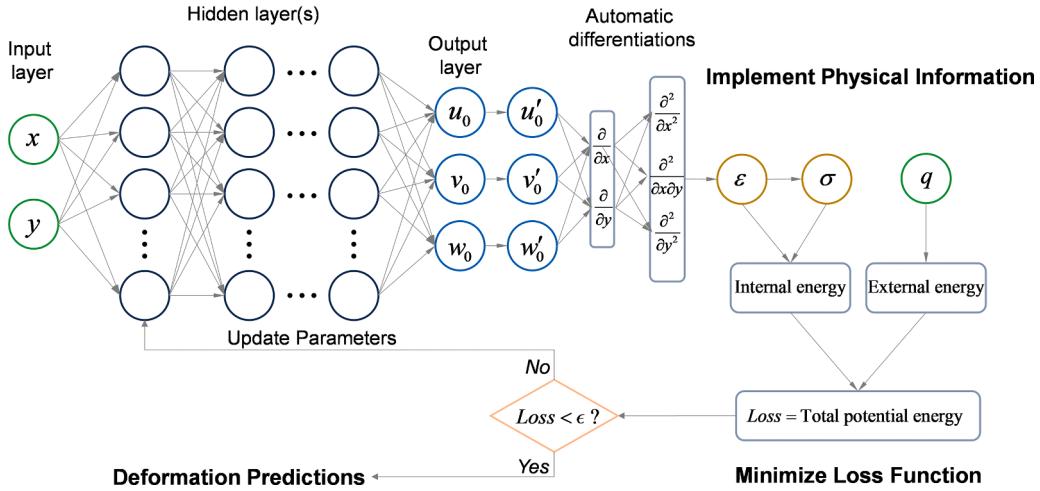


Fig. 3. The flow chart of the PINN framework.

of the virtual displacement components ( $\delta u_0, \delta v_0, \delta w_0$ ) to zero, the governing equations for membrane forces and bending moments can be

$$0 = \int_{\Omega} \left( N_{xx} \delta \epsilon_{xx}^0 + M_{xx} \delta \epsilon_{xx}^1 + N_{yy} \delta \epsilon_{yy}^0 + M_{yy} \delta \epsilon_{yy}^1 + N_{xy} \delta \epsilon_{xy}^0 + M_{xy} \delta \epsilon_{xy}^1 - q \delta w_0 \right) dx dy - \int_{\Gamma_0} \left( N_{nn} \delta u_{0n} + N_{ns} \delta u_{0s} - M_{nn} \frac{\partial \delta w_0}{\partial n} - M_{ns} \frac{\partial \delta w_0}{\partial s} - Q_n \delta w_0 \right) ds \quad (5)$$

where  $\begin{Bmatrix} N_{nn} \\ N_{ns} \end{Bmatrix} = \int_{-h/2}^{h/2} \begin{Bmatrix} \sigma_{nn} \\ \sigma_{ns} \end{Bmatrix} dz$ ,  $\begin{Bmatrix} M_{nn} \\ M_{ns} \end{Bmatrix} = \int_{-h/2}^{h/2} \begin{Bmatrix} \sigma_{nn} \\ \sigma_{ns} \end{Bmatrix} dz$ ,  $\{Q_n\} = \int_{-h/2}^{h/2} \sigma_{nz} dz$ .

The relationship between virtual displacement and virtual strain can be derived from Eq. (1):

$$\begin{aligned} \delta \epsilon_{xx}^0 &= \frac{\partial \delta u_0}{\partial x} + \frac{\partial w_0}{\partial x} \frac{\partial \delta w_0}{\partial x}, \quad \delta \epsilon_{xx}^1 = -\frac{\partial^2 \delta w_0}{\partial x^2} \\ \delta \epsilon_{yy}^0 &= \frac{\partial \delta v_0}{\partial y} + \frac{\partial w_0}{\partial y} \frac{\partial \delta w_0}{\partial y}, \quad \delta \epsilon_{xx}^1 = -\frac{\partial^2 \delta w_0}{\partial y^2} \\ \delta \epsilon_{xy}^0 &= \frac{\partial \delta u_0}{\partial y} + \frac{\partial v_0}{\partial y} \frac{\partial \delta w_0}{\partial x} + \frac{\partial w_0}{\partial x} \frac{\partial \delta w_0}{\partial y}, \quad \delta \epsilon_{xy}^1 = -2 \frac{\partial^2 \delta w_0}{\partial x \partial y} \end{aligned} \quad (6)$$

By substituting the virtual strains into Eq. (5), and performing multiple integrations by parts, the virtual work equation can ultimately be expressed in terms of the virtual displacements as follows:

$$0 = \int_{\Omega} \left[ \left( \frac{\partial N_{xx}}{\partial x} + \frac{\partial N_{xy}}{\partial y} \right) \delta u_0 + \left( \frac{\partial N_{yy}}{\partial y} + \frac{\partial N_{xy}}{\partial x} \right) \delta v_0 + \left( \frac{\partial}{\partial x} \left( N_{xx} \frac{\partial w_0}{\partial x} + N_{xy} \frac{\partial w_0}{\partial y} \right) \delta w_0 + \frac{\partial}{\partial y} \left( N_{yy} \frac{\partial w_0}{\partial y} + N_{xy} \frac{\partial w_0}{\partial x} \right) \delta w_0 + \left( \frac{\partial^2 M_{xx}}{\partial x^2} + 2 \frac{\partial^2 M_{xy}}{\partial x \partial y} + \frac{\partial^2 M_{yy}}{\partial y^2} - q \right) \delta w_0 \right] \right] + \int_{\Gamma_0} \left( N_{nn} \delta u_{0n} + N_{ns} \delta u_{0s} - M_{nn} \frac{\partial \delta w_0}{\partial n} - M_{ns} \frac{\partial \delta w_0}{\partial s} - Q_n \delta w_0 \right) ds \quad (7)$$

According to the principle of virtual work, both the domain integral terms and the boundary integral terms in Eq. (7) must vanish to ensure the total potential energy remains unchanged. By setting the coefficients

derived as :

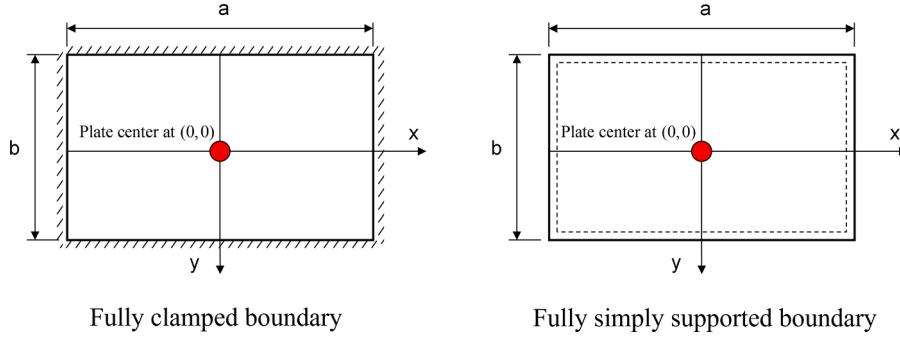
$$\begin{aligned} \frac{\partial N_{xx}}{\partial x} + \frac{\partial N_{xy}}{\partial y} &= 0, \quad \frac{\partial N_{yy}}{\partial y} + \frac{\partial N_{xy}}{\partial x} = 0 \\ \frac{\partial}{\partial x} \left( N_{xx} \frac{\partial w_0}{\partial x} + N_{xy} \frac{\partial w_0}{\partial y} \right) + \frac{\partial}{\partial y} \left( N_{yy} \frac{\partial w_0}{\partial y} + N_{xy} \frac{\partial w_0}{\partial x} \right) + \frac{\partial^2 M_{xx}}{\partial x^2} + 2 \frac{\partial^2 M_{xy}}{\partial x \partial y} + \frac{\partial^2 M_{yy}}{\partial y^2} &= q \end{aligned} \quad (8)$$

### 3. PINN computational framework

For the laminated plate bending behavior, analytical methods, such as the Navier's and Levy's methods, provide classical solutions based on series expansion or closed-form expressions by introducing reasonable assumptions. However, these methods typically require strict BCs and regular load distributions. In practical engineering applications, numerical methods based on weak-form solutions, such as the FEM, are more common to approximate exact solutions effectively. This paper develops a PINN framework as Fig. 3 shown, which combines the powerful fitting capabilities of the ANN with CLPT, offering an alternative solution for laminated plate bending analysis.

#### 3.1. Loss function

In the PINN framework, the physical information is implemented through the loss function. A commonly used approach to construct the PINN framework is the PDE-based approach. The loss function of this approach is a weighted sum of the PDE residual loss and the boundary/initial condition losses [36,67]. By minimizing these residuals with gradient descent optimization algorithms, neural networks can approximate the solution to satisfy model governing equations and BCs [68]. For example, Cheng et al. [69] designed loss functions based on the Reynolds-Average Navier-Stokes (RANS) equation coupled with the



**Fig. 4.** The schematic diagram for rectangular laminated plates with fully clamped and fully simply supported boundaries.

structure dynamic motion equation to address the vortex-induced vibration. Similarly, Bischof [70] proposed a PINN framework to predict elastic plate deformations using the Kirchhoff plate bending equation, which is a fourth-order PDE with BCs on both zeroth and second-order derivatives.

The governing equations of the CLPT also include fourth-order PDEs for deflections and can be derived from Eq. (8). If the loss function is formulated using the PDE-based approaches, it would involve incorpo-

The energy integrals can be approximated using the discrete form of numerical integration. Assume the integration region  $\Omega$  is divided into  $P_1$  small subregions with each subregion having an area of  $\Delta A_i$ . Similarly, the boundary integration region  $\Gamma_0$  is divided into  $P_2$  small boundary segments, with each segment having a length of  $\Delta s_i$ . So, the energy items can be rewritten as a summation of the energy at discrete points:

$$\begin{aligned} U = U_m + U_b &\approx \sum_{i=1}^{P_1} \frac{1}{2} \left( N_{xx}^i \varepsilon_{xx}^{0,i} + N_{yy}^i \varepsilon_{yy}^{0,i} + 2N_{xy}^i \varepsilon_{xy}^{0,i} - M_{xx}^i \varepsilon_{xx}^{1,i} - M_{yy}^i \varepsilon_{yy}^{1,i} - 2M_{xy}^i \varepsilon_{xy}^{1,i} \right) \Delta A_i \\ V &\approx \sum_{i=1}^{P_1} -q w_0 \Delta A_i + \sum_{j=1}^{P_2} \left( -N_{mn}^j u_{0n}^j - N_{ns}^j u_{0s}^j + M_{mn}^j \frac{\partial w_0^j}{\partial n} + M_{ns}^j \frac{\partial w_0^j}{\partial s} + Q_n^j w_0^j \right) \Delta s_i \end{aligned} \quad (11)$$

rating high-order PDEs calculation and different-scale residual terms into the neural network training, which can significantly increase computational cost of the framework [53]. Additionally, the need to balance and interpret the weights of the residual terms also poses a significant challenge [71]. An alternative approach is to formulate the loss function based on the principle of minimum potential energy and use variational methods to solve PDEs [72,73]. These energy-based PINNs, inspired by the Deep Ritz Method proposed by Yu [74], solve the PDEs using a weak form rather than directly handling differential operators and BCs.

Our framework also uses the energy-based approach to formulate the loss function. The total potential energy in the laminate system can be expressed as:

$$\Pi = U + V = U_m + U_b + V \quad (9)$$

where  $U$  represents the internal energy of the laminate, including internal membrane energy  $U_m$  and bending energy  $U_b$ , while  $V$  denotes the external energy done by the applied loads. These energy items are defined as:

When the quantities of  $P_1$  and  $P_2$  are sufficiently large and uniformly distributed, the  $\Delta A_i$  and  $\Delta s_i$  can be approximated as  $A/P_1$  and  $s/P_2$ , where  $A$  and  $s$  are the total area and total boundary length of the laminate.

Take the total potential energy equation as the basis of the model loss. The loss function can be formulated as:

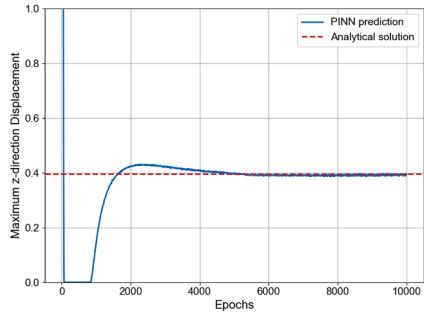
$$\text{Loss} = U + V + K \quad (12)$$

where  $K$  is the loss penalty term for the essential BCs. This is because Eq. (12) accounts only for the natural BCs and cannot explicitly incorporate the essential BCs. In this study, the essential BCs of the laminate system are strictly satisfied using the hard constraints discussed in Section 3.3. Consequently, the penalty term can be eliminated, and the total loss is considered as the sum of internal and external potential energies. Compared with the PDE-based loss function, our framework requires fewer residual terms and avoids high-order differential calculations.

### 3.2. Artificial neural network

This computational framework utilizes ANNs to approximate the

$$\begin{aligned} U = U_m + U_b &= \int_{\Omega} \frac{1}{2} \left( N_{xx} \varepsilon_{xx}^0 + N_{yy} \varepsilon_{yy}^0 + 2N_{xy} \varepsilon_{xy}^0 \right) dx dy + \int_{\Omega} \frac{1}{2} \left( -M_{xx} \varepsilon_{xx}^1 - M_{yy} \varepsilon_{yy}^1 - 2M_{xy} \varepsilon_{xy}^1 \right) dx dy \\ V &= \int_{\Omega} -q w_0 dx dy + \int_{\Gamma_0} \left( -N_{mn} u_{0n} - N_{ns} u_{0s} + M_{mn} \frac{\partial w_0}{\partial n} + M_{ns} \frac{\partial w_0}{\partial s} + Q_n w_0 \right) ds \end{aligned} \quad (10)$$



**Fig. 5.** The predicted maximum z-direction displacement and training loss of the clamped ( $0^\circ/90^\circ$ ) laminated plate.

mid-plane displacement solutions of laminated plates while ensuring compliance with the physical constraints of the energy-based loss function described above. The structure of the neural network employed is shown in Fig. 3, which consists of one input layer, one output layer, and multiple hidden layers. Assume the neural network has  $n$  layers, and there are  $k$  neurons in each layer. There are no connections between neurons in the same layer, and each neuron in the  $n_{th}$  layer is connected to all neurons in the  $n_{th-1}$  layer, which defines a fully connected layer. The outputs of neurons in the  $n_{th-1}$  layer serve as inputs to neurons in the  $n_{th}$  layer. For each neuron, the input from the previous layer is first multiplied by the corresponding weights  $W$  and then summed with a bias term  $b$  to enhance the predictive performance. This result is then passed through an activation function  $f$  to compute the output of each current neuron. Eq. (13) presents the calculation process of the output  $A_j^n$  of the  $j_{th}$  neuron in the  $n_{th}$  layer.

$$A_j^n = f \left( \sum_{i=1}^k W_{ij}^n A_i^{n-1} + b_j^n \right) \quad (13)$$

where  $W_{ij}$  and  $b_{ij}$  are the corresponding weight and biases of  $A_i^{n-1}$  to  $A_j^n$ .

The established neural network takes the plane coordinates of sampling points ( $x, y$ ) within the solution domain as inputs to the input layer, while the output layer generates the predicted mid-plane displacements ( $u_0, v_0, w_0$ ) as the outputs. Through the built-in AD algorithm of the deep learning framework, the strain components at any point ( $x, y, z$ ) within the plate can be determined using Eq. (1) once the mid-surface displacement components are known. Subsequently, stress components at each point in the training dataset can also be obtained using the laminate constitutive equations. Based on the derived stress and strain fields, the neural network is trained using a gradient descent algorithm to minimize the loss function. During each epoch, backpropagation is applied to update the network weights and biases by descending along the gradient. Repeat this process until the loss function converges to an acceptable error  $\epsilon$  or the maximum number of epochs is reached. The trained neural network is then able to approximate the laminate displacement field to the exact solution based on the provided physical information.

### 3.3. Boundary conditions

According to the CLPT, the BCs can be derived from the boundary integral term of Eq. (7). There are eight BCs based on the boundary local coordinate system, categorized into four essential BCs and four natural BCs as follows:

$$u_0, v_0, w_0, \frac{\partial w_0}{\partial n} \text{ (essential); } N_{nn}, N_{ss}, V_n, M_{nn} \text{ (natural)}$$

where  $V_n = Q_n + \frac{\partial M_{nn}}{\partial s}$  is known as the Kirchhoff free-edge condition [62]. The essential BCs describe the input conditions at the boundary, explicitly specifying the kinematics constraints, while natural BCs are from governing equations derivation and reflect the mechanical

equilibrium on the boundary.

Fig. 4 depicts the schematic diagrams for clamped and simply supported rectangular laminated plates under the global coordinate system. Take the center of the plate as the global coordinate origin, and the BCs are defined as follows:

$$\begin{aligned} \text{Clamped: } u_0 &= v_0 = w_0 = \frac{\partial w_0}{\partial x} = 0 \left( \text{at } x = -\frac{a}{2} \text{ and } \frac{a}{2} \right) \\ u_0 &= v_0 = w_0 = \frac{\partial w_0}{\partial y} = 0 \left( \text{at } y = -\frac{b}{2} \text{ and } \frac{b}{2} \right) \end{aligned} \quad (14)$$

$$\begin{aligned} \text{Simply supported: } v_0 &= w_0 = N_{xx} = M_{xx} = 0 \left( \text{at } x = -\frac{a}{2} \text{ and } \frac{a}{2} \right) \\ u_0 &= w_0 = N_{yy} = M_{yy} = 0 \left( \text{at } y = -\frac{b}{2} \text{ and } \frac{b}{2} \right) \end{aligned} \quad (15)$$

Since the energy-based approach does not directly incorporate the loss of BCs, our PINN framework imposes a hard constraint on the neural network outputs to eliminate the penalty term in the loss function. Assume the original output of the neural networks is  $Net(x, y)$ , and the modified output  $Net'(x, y)$  is expressed as:

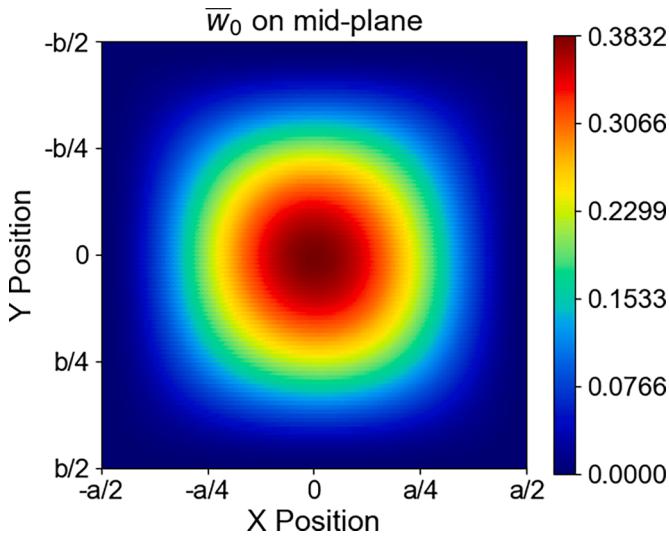
$$Net'(x, y) = D(x, y) \cdot Net(x, y) + G(x, y) \quad (16)$$

where  $D(x, y)$  and  $G(x, y)$  are two boundary correction functions specifically designed to modify the neural network outputs. These functions ensure that BCs are consistently satisfied when the input coordinates ( $x, y$ ) lie on the boundaries. Further details on their design are discussed in Section 4 to address various boundary combinations.

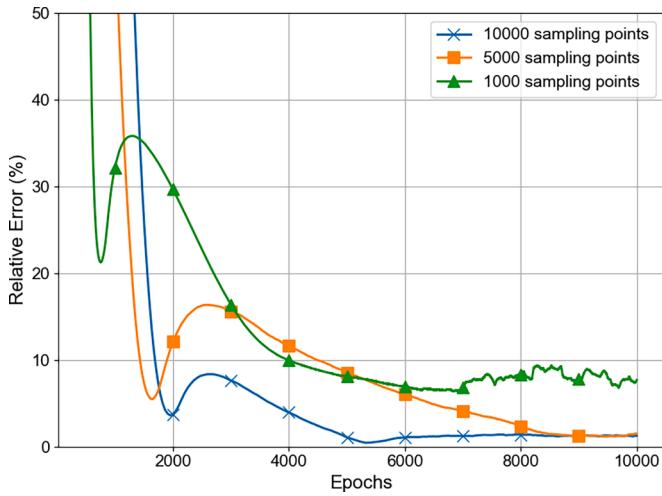
## 4. Applications and validations

As previously mentioned, a significant advantage of the PINN framework is its independence from the actual solution dataset for the problem to be solved. Instead, it samples points within the solution domain to construct the training dataset. To ensure comprehensive coverage of the solution domain, the initial training data is uniformly sampled. Random resampling is then employed during each training epoch to capture complex behavior, enhancing model generalization and improving the accuracy of numerical integration. Additionally, the coordinates of the points input to the neural network need to be normalized in the range of [-1, 1].

For the mid-plane displacement components ( $u_0, v_0, w_0$ ) along three directions, three independent fully connected neural networks are constructed for predictions. Each neural network consists of 5 layers, with each layer containing 5 neurons, designed to balance computational efficiency and prediction accuracy. The hyperbolic tangent function  $tanh$  is selected as the activation function, and the *Adam* optimizer is adopted to simultaneously update the parameters of the three networks based on the gradient of the loss function. The trained PINN framework is capable of predicting the displacement components at any



**Fig. 6.** The out-of-plane deformation distribution of the clamped ( $0^\circ/90^\circ$ ) laminated plate under UDL.



**Fig. 7.** The convergence process of the deformation predictions for the clamped ( $0^\circ/90^\circ$ ) laminated plate under UDL.

point within the mid-plane of the laminated plate and computing the corresponding stress and strain fields through AD. To validate the accuracy and generalizability of the developed computational framework in predicting laminate bending behavior, this study conducts the following applications, examining the predictive performance from three perspectives: load cases, laminate lay-ups, and BCs. The source code for the developed PINN framework is available at <https://github.com/VissyWANG/laminate-bending-PINN>.

#### 4.1. Clamped cross-ply laminated plate under uniformly distributed load

Bhaskar and Kaushik [63] provided the analytical solutions for the bending behavior of the clamped cross-ply square laminated plates subject to a transverse uniformly distributed load (UDL). The square laminated plates discussed in their study had lay-ups with  $(0^\circ)$ ,  $(0^\circ/90^\circ)$ ,  $(0^\circ/90^\circ/0^\circ)$  and  $(0^\circ/90^\circ)_2$ . The material for each layer in laminated plates is defined as orthotropic and has the following properties:

$$E_1/E_2 = 25, \quad G_{12} = 0.5E_2, \quad \nu_{12} = 0.25$$

For the clamped boundaries, this study formulates the following hard constraints to meet the corresponding BCs according to Eqs. (14) and

**Table 1**

Comparison of predicted and analytical solutions [63] for the deflection and moment resultants of the clamped laminated plates.

(0°) laminated plate	Predicted results	Analytical solutions	Error
$\bar{w}_0$ at the center (0, 0)	0.1324	0.1308	1.22%
$\bar{M}_{xx}$ at the center (0, 0)	0.0430	0.0439	2.03%
$\bar{M}_{yy}$ at the center (0, 0)	0.0007	0.0007	1.48%
$\bar{M}_{xx}$ at mid-edge (-a/2, 0)	-0.0885	-0.0870	1.69%
$\bar{M}_{yy}$ at mid-edge (0, -b/2)	-0.0111	-0.0112	0.98%
(0°/90°) laminated plate			
$\bar{w}_0$ at the center (0, 0)	0.3832	0.3954	3.09%
$\bar{M}_{xx}$ at the center (0, 0)	0.0250	0.0239	4.73%
$\bar{M}_{yy}$ at the center (0, 0)	0.0232	0.0239	2.81%
$\bar{M}_{xx}$ at mid-edge (-a/2, 0)	-0.0580	-0.0553	4.81%
$\bar{M}_{yy}$ at mid-edge (0, -b/2)	-0.0531	-0.0553	4.05 %
(0°/90°/0°) laminated plate			
$\bar{w}_0$ at the center (0, 0)	0.1382	0.1371	0.80%
$\bar{M}_{xx}$ at the center (0, 0)	0.0447	0.0444	0.63%
$\bar{M}_{yy}$ at the center (0, 0)	0.0019	0.0018	3.37%
$\bar{M}_{xx}$ at mid-edge (-a/2, 0)	-0.0893	-0.0881	1.37%
$\bar{M}_{yy}$ at mid-edge (0, -b/2)	-0.0157	-0.0157	0.06%
(0°/90°) <sub>2</sub> laminated plate			
$\bar{w}_0$ at the center (0, 0)	0.1706	0.1761	3.12%
$\bar{M}_{xx}$ at the center (0, 0)	0.0238	0.0249	4.42%
$\bar{M}_{yy}$ at the center (0, 0)	0.0254	0.0249	2.01 %
$\bar{M}_{xx}$ at mid-edge (-a/2, 0)	-0.0552	-0.0565	2.28 %
$\bar{M}_{yy}$ at mid-edge (0, -b/2)	-0.0558	-0.0565	1.22 %

(16).

$$\begin{aligned} Net_w(x, y) &= Net_w(x, y)((x+1)(x-1)(y+1)(y-1))^2 \\ Net_u(x, y) &= Net_u(x, y)((x+1)(x-1)(y+1)(y-1)) \\ Net_v(x, y) &= Net_v(x, y)((x+1)(x-1)(y+1)(y-1)) \end{aligned} \quad (17)$$

Obviously, when the input coordinates are on the plate boundaries (i.e.,  $x$  or  $y = -1$  or 1), the corrected neural network outputs of in-plane displacement  $u$  and  $v$  consistently remain zero. For out-of-plane displacements  $w$ , the hard constraints ensure that both  $w$  and its first-order derivatives on the boundaries are equal to zero.

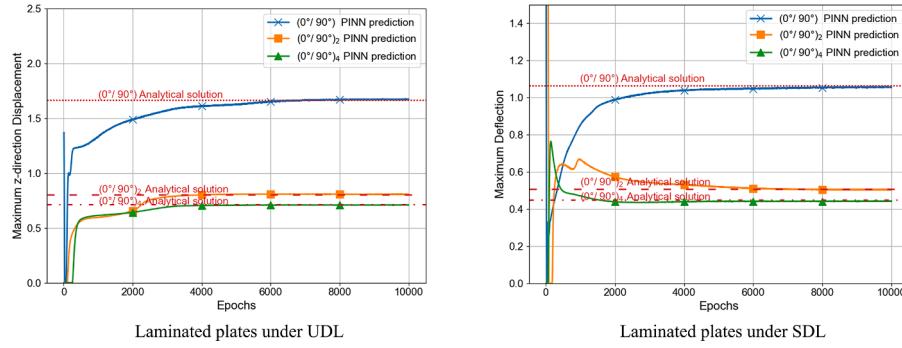
By incorporating the specified physical information into our PINN framework, the mid-lane displacement components of the laminated plate can be predicted, and the bending moment resultants in the solution domain can also be derived using Eq. (2). The following non-dimensionalized quantities are used to convert the predicted results to numerical results:

$$\bar{w}_0 = \frac{100E_2h^3}{qa^4}w_0, \quad \bar{M}_{xx} = \frac{M_{xx}}{qa^2}, \quad \bar{M}_{yy} = \frac{M_{yy}}{qa^2} \quad (18)$$

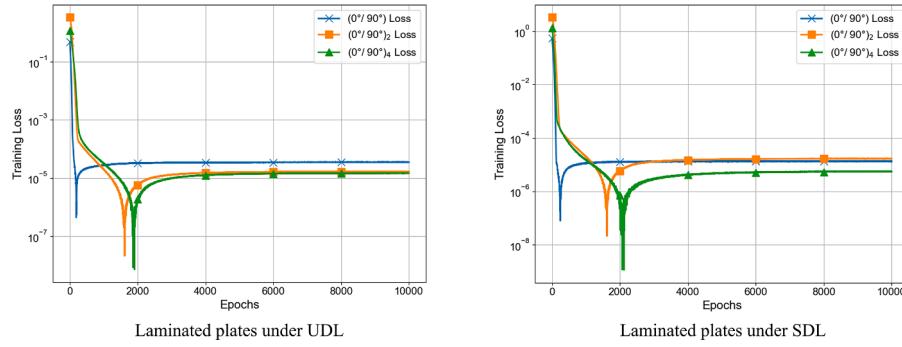
where  $\bar{w}_0$  and  $\bar{M}$  represent numerical results for  $z$ -direction displacement and moment resultant.

As an example, this study considers a simply supported ( $0^\circ/90^\circ$ ) square laminated plate subjected to a UDL. Within the solution domain, 10,000 points are uniformly sampled, and their coordinates are input into the neural network. The framework is then used to conduct 10,000 epochs of training with a learning rate of  $10^{-3}$ . Fig. 5 shows the change in the training loss and predicted maximum  $z$ -direction displacement during the entire training process. The training loss tends to converge and stabilize after 6000 epochs. Furthermore, the maximum  $z$ -direction displacement predictions remain stable around the analytical solutions (the red dashed line shown in Fig. 5) after 5000 epochs. Fig. 6 shows the out-of-plane deformation distribution of the ( $0^\circ/90^\circ$ ) laminate. The deformation follows a symmetric sine half-wave pattern, with the maximum displacement located at the center of the laminate and gradually decreasing toward the edges.

In order to assess the impact of sampling dataset size on the convergence and accuracy of the energy-based PINN framework, a validation is performed using three different sets of sampling points: 1000, 5000, and 10,000 points. The model performance is evaluated



**Fig. 8.** The predicted maximum z-direction displacement of simply supported laminated plates under different load cases.



**Fig. 9.** The training loss of simply supported laminated plates under different load cases.

**Table 2**

Comparison of predicted and analytical solutions [62] for the deflection and stress components of the simply supported laminated plates.

(0°/90°)	Laminated plates under UDL			Laminated plates under SDL		
	Predicted results	Analytical solutions	Error	Predicted results	Analytical solutions	Error
$\bar{w}_0(0, 0)$	1.6826	1.6955	0.76 %	1.0562	1.0636	0.70 %
$\bar{\sigma}_{xx}(0, 0, h/2)$	0.1255	0.1268	1.01 %	0.0850	0.0843	0.88 %
$\bar{\sigma}_{yy}(0, 0, h/2)$	1.0894	1.0761	1.24 %	0.7083	0.7157	1.03 %
$\bar{\sigma}_{xy}(a/2, -b/2, h/2)$	0.0953	0.0933	2.12 %	0.0533	0.0525	1.52 %
$(0^\circ/90^\circ)_2$						
$\bar{w}_0(0, 0)$	0.8049	0.8085	0.45 %	0.5028	0.5065	0.73 %
$\bar{\sigma}_{xx}(0, 0, h/2)$	0.0534	0.0541	0.92 %	0.0348	0.0357	2.52 %
$\bar{\sigma}_{yy}(0, 0, h/2)$	0.7284	0.7367	0.19 %	0.4775	0.4868	1.91 %
$\bar{\sigma}_{xy}(a/2, -b/2, h/2)$	0.0448	0.0442	1.36 %	0.0247	0.0250	1.20 %
$(0^\circ/90^\circ)_4$						
$\bar{w}_0(0, 0)$	0.7146	0.7150	0.06 %	0.4457	0.4479	0.49 %
$\bar{\sigma}_{xx}(0, 0, h/2)$	0.0449	0.0449	0.11 %	0.0291	0.0296	2.03 %
$\bar{\sigma}_{yy}(0, 0, h/2)$	0.7486	0.7496	0.13 %	0.4987	0.4950	0.75 %
$\bar{\sigma}_{xy}(a/2, -b/2, h/2)$	0.0387	0.0391	1.02 %	0.0227	0.0221	2.71 %

based on the relative error between the PINN prediction and the analytical solution, which is calculated using the formula:

$$\text{Relative Error} = \frac{| \text{Predicted Value} - \text{Analytical Solution} |}{| \text{Analytical Solution} |} \times 100\% \quad (19)$$

Based on the corresponding convergence processes of out-of-plane deformation predictions shown in Fig. 7, it is evident that the number of sampling points significantly impacts both the convergence speed and accuracy of the energy-based PINN framework. The models trained with 5000 and 1000 sampling points demonstrate slower convergence and higher final errors, with the latter showing considerable fluctuations throughout the training process.

Extend the framework to the laminated plates with other lay-up configurations used in Bhaskar and Kaushik's study [63]. Table 1 presents a comparison between the predicted results and the analytical solutions. For all the lay-ups considered, our PINN framework achieves

accurate predictions for z-direction displacement at the plate center (also the maximum deflection of the plate). The predicted moments also fit well with the analytical solution at the plate center and mid-edges, with errors within 5 %. Compared with other lay-ups,  $(0^\circ/90^\circ)$  laminate has the maximum out-of-plane displacement due to its high extension-bending coupling. In contrast, symmetric laminates, which lack extension-bending coupling, show better predictive performance for the bending behavior of  $(0^\circ)$  and  $(0^\circ/90^\circ/0^\circ)$  compared to  $(0^\circ/90^\circ)$  and  $(0^\circ/90^\circ)_2$  lay-ups.

#### 4.2. Simply supported cross-ply laminated plate under sinusoidally distributed load

Reddy [62] provided the analytical solutions for the bending behavior of the supported cross-ply square laminated plates subject to the UDL and sinusoidally distributed load (SDL). The square laminated

plates discussed in his study had lay-ups with  $(0^\circ/90^\circ)$ ,  $(0^\circ/90^\circ)_2$  and  $(0^\circ/90^\circ)_4$ . The material properties and geometric parameters are defined identically to those used in the previous validation.

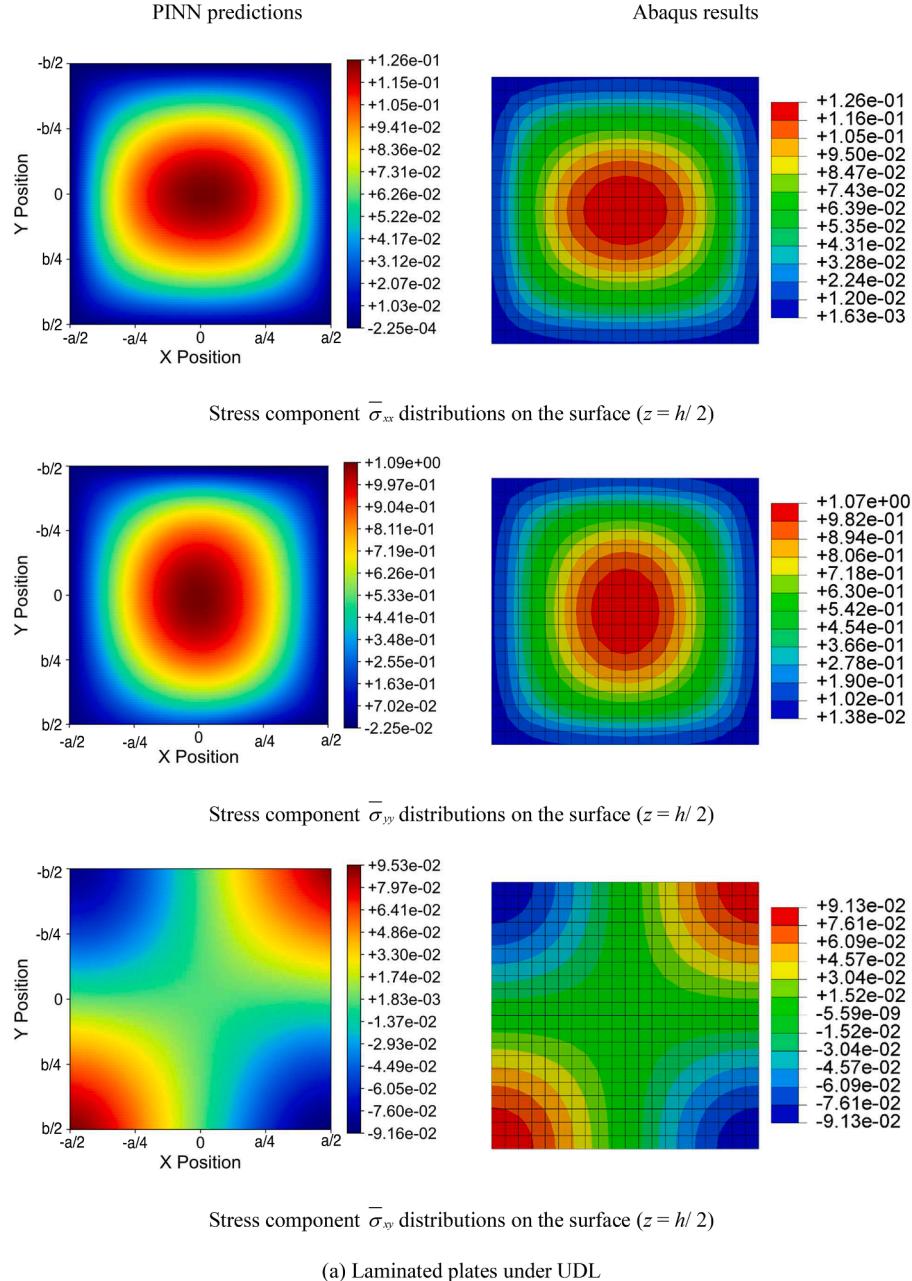
For simply supported boundaries, the hard constraints need to be modified as follows:

$$\begin{aligned} \text{Net}_w(x,y) &= \text{Net}_w(x,y)((x+1)(x-1)(y+1)(y-1)) \\ \text{Net}_u(x,y) &= \text{Net}_u(x,y)((y+1)(y-1)) \\ \text{Net}_v(x,y) &= \text{Net}_v(x,y)((x+1)(x-1)) \end{aligned} \quad (20)$$

When the input coordinates are on the plate boundary, the corrected output can automatically satisfy the corresponding BCs specified in Eq. (15). To align the load function with the laminate coordinate system and normalized neural network inputs, the SDL planar distribution function for this model is defined as:

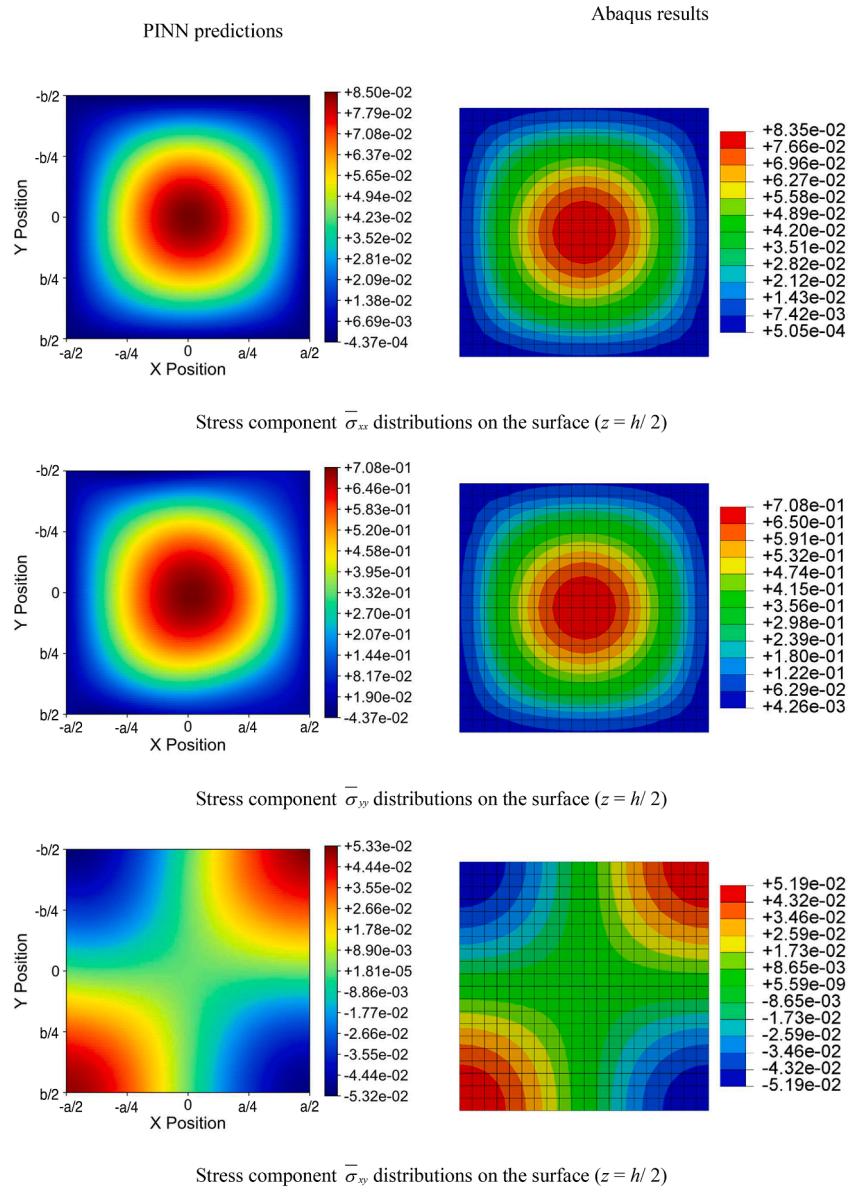
$$q(x,y) = q_0 \sin \frac{\pi(x+1)}{a} \sin \frac{\pi(y+1)}{b} \quad (21)$$

Similarly, the number of sampling points was set as 10,000 points. The framework was used to conduct 10,000 epochs of training with a learning rate of  $10^{-3}$ . Fig. 8 demonstrates the convergence process for the predicted maximum z-direction displacement of the laminated plates under different load cases, while Fig. 9 presents the corresponding training losses. For simply supported square laminated plates under the UDL and SDL, the training loss stabilized after approximately 4000 epochs, indicating that the model gradually converged. It is evident that the PINN framework consistently provides accurate out-of-plane deformation predictions after around 6000 epochs of training. Using Eq. (18) to convert the z-direction displacement into numerical results, the maximum deflection predictions at the plate center are presented in Table 2. For all antisymmetric laminated plates, the final relative errors of the maximum deflections after 10,000 epochs are within 2 %. As the increased number of layers in the  $(0^\circ/90^\circ)_2$  and  $(0^\circ/90^\circ)_4$  plates, the influence of the bending-extensional coupling coefficient  $B_{ij}$  is progressively diminished, also resulting in an enhanced accuracy of the PINN

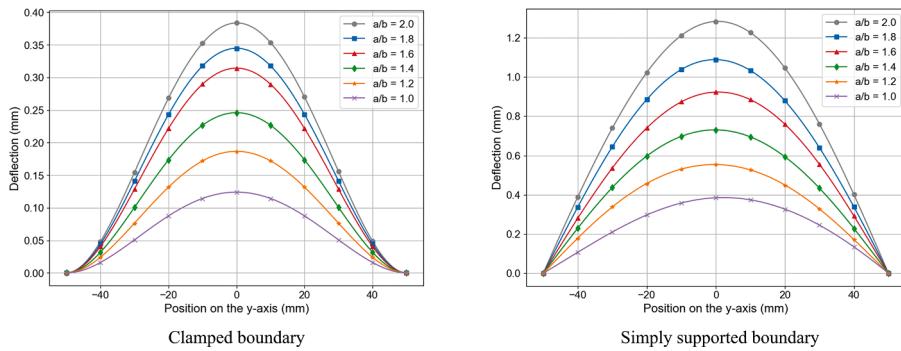


(a) Laminated plates under UDL

**Fig. 10.** Comparison of stress component of simply supported  $(0^\circ/90^\circ)$  laminated plates predicted by PINN and Abaqus.



(b) Laminated plates under SDL

**Fig. 10. (continued).****Fig. 11.** The deflections along the y-axis of the  $(-45_2/0_2/45_2/90_2)_s$  laminated plate under UDL.

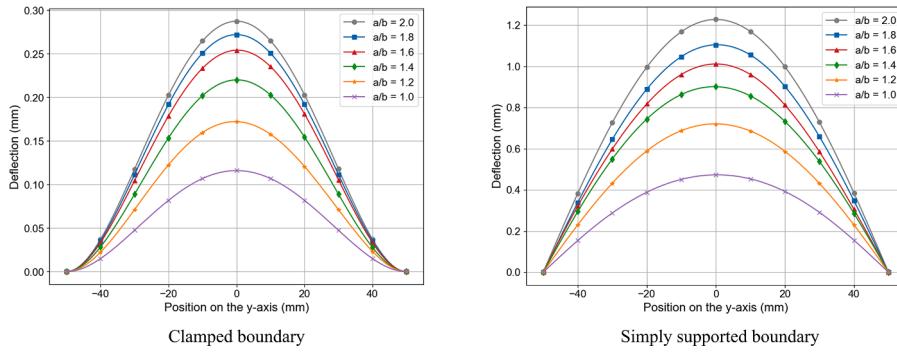


Fig. 12. The deflections along the y-axis of the  $(0^\circ/90^\circ/-45^\circ/45^\circ)_s$  laminated plate under UDL.

Table 3

Comparison of predicted and FEM results [65] for the maximum deflection of the angle-ply rectangular laminated plates.

Aspect ratio	Maximum deflection (mm) for $(-45^\circ_2/0^\circ_2/45^\circ_2/90^\circ_2)_s$ plate				Maximum deflection (mm) for $(0^\circ_2/90^\circ_2/-45^\circ_2/45^\circ_2)_s$ plate			
	Clamped	Simply supported		Clamped	Simply supported			
$a/b = 1.0$	PINN FEM Error	0.127 0.130 2.15 %	PINN FEM Error	0.385 0.392 1.79 %	PINN FEM Error	0.118 0.120 1.83 %	PINN FEM Error	0.473 0.488 3.03 %
$a/b = 1.2$	PINN FEM Error	0.188 0.196 3.93 %	PINN FEM Error	0.554 0.583 4.97 %	PINN FEM Error	0.172 0.178 3.31 %	PINN FEM Error	0.712 0.718 2.23 %
$a/b = 1.4$	PINN FEM Error	0.247 0.257 3.74 %	PINN FEM Error	0.745 0.778 4.22 %	PINN FEM Error	0.220 0.224 1.79 %	PINN FEM Error	0.904 0.920 1.74 %
$a/b = 1.6$	PINN FEM Error	0.313 0.321 2.49 %	PINN FEM Error	0.924 0.965 4.25 %	PINN FEM Error	0.254 0.255 0.35 %	PINN FEM Error	1.057 1.079 2.04 %
$a/b = 1.8$	PINN FEM Error	0.342 0.350 2.29 %	PINN FEM Error	1.089 1.136 4.14 %	PINN FEM Error	0.272 0.275 1.09 %	PINN FEM Error	1.147 1.197 4.18 %
$a/b = 2.0$	PINN FEM Error	0.384 0.370 3.68 %	PINN FEM Error	1.286 1.287 0.09 %	PINN FEM Error	0.287 0.285 0.84 %	PINN FEM Error	1.235 1.281 3.60 %

predictions.

The stress component distribution at thickness position  $z$  within the  $k_{th}$  layer can be calculated using the following equation:

$$\begin{Bmatrix} \sigma_{xx}(z) \\ \sigma_{yy}(z) \\ \sigma_{xy}(z) \end{Bmatrix}^{(k)} = \begin{Bmatrix} \bar{Q}_{11}\bar{Q}_{12}\bar{Q}_{16} \\ \bar{Q}_{12}\bar{Q}_{22}\bar{Q}_{26} \\ \bar{Q}_{16}\bar{Q}_{26}\bar{Q}_{66} \end{Bmatrix}^{(k)} \begin{Bmatrix} \varepsilon_{xx}^0 \\ \varepsilon_{yy}^0 \\ \varepsilon_{xy}^0 \end{Bmatrix} + z \begin{Bmatrix} \bar{Q}_{11}\bar{Q}_{12}\bar{Q}_{16} \\ \bar{Q}_{12}\bar{Q}_{22}\bar{Q}_{26} \\ \bar{Q}_{16}\bar{Q}_{26}\bar{Q}_{66} \end{Bmatrix}^{(k)} \begin{Bmatrix} \varepsilon_{xx}^1 \\ \varepsilon_{yy}^1 \\ \varepsilon_{xy}^1 \end{Bmatrix} \quad (22)$$

where  $[\bar{Q}_{ij}]^{(k)}$  represents the stiffness matrix components of the  $k_{th}$  layer lamina in the global coordinate system of the laminated plate. The stress components of the laminated plate can be nondimensionalized to numerical results as follows:

$$\bar{\sigma}_{xx} = \frac{h^2}{q_0 a^2} \sigma_{xx}, \quad \bar{\sigma}_{yy} = \frac{h^2}{q_0 a^2} \sigma_{yy}, \quad \bar{\sigma}_{xy} = \frac{h^2}{q_0 a^2} \sigma_{xy} \quad (23)$$

As an example, this study considers a simply supported  $(0^\circ/90^\circ)$

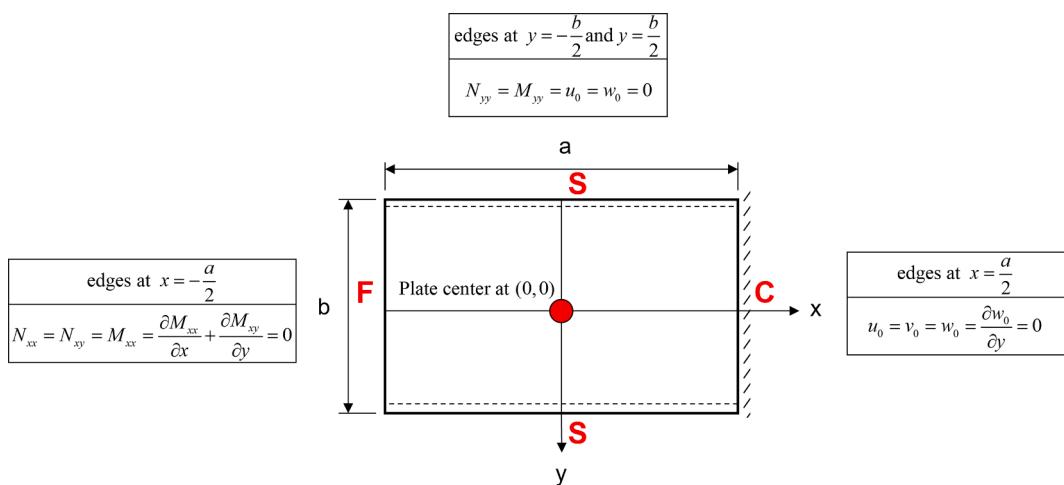


Fig. 13. The schematic diagram for the "FSCS" boundary.

**Table 4**

Comparison of predicted and analytical solutions [64] for the central deflection of the symmetric square laminated plates under the UDL.

Boundary combination	Central deflection for (0°) laminated plate under UDL			Central deflection for (0°/90°/0°) laminated plate under UDL		
	PINN results	Analytical solution	Error	PINN results	Analytical solution	Error
SCSS	0.6199	0.6157	0.68 %	0.5897	0.5999	2.36 %
CSSS	0.2717	0.2644	2.76 %	0.2765	0.2799	1.21 %
SCSC	0.5792	0.5812	3.44 %	0.5436	0.5340	1.79 %
CSCS	0.1312	0.1293	1.46 %	0.1421	0.1385	2.59 %
CCSS	0.2655	0.2615	3.28 %	0.2667	0.2598	2.65 %
CCSC	0.2671	0.2586	1.52 %	0.2721	0.2696	0.93 %

### Out-of-plane deformation

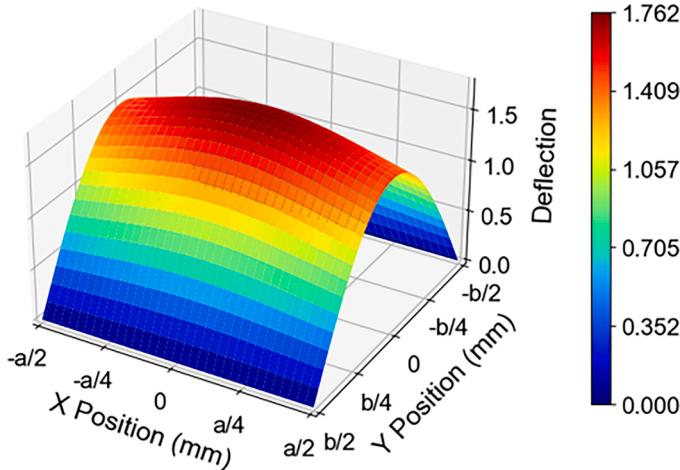
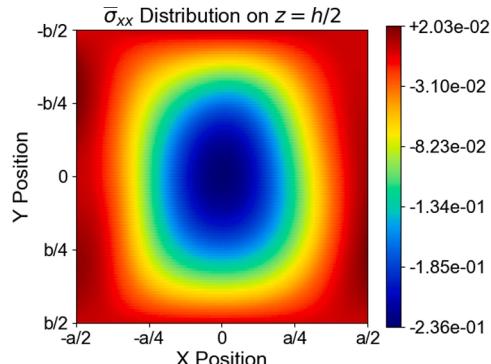


Fig. 14. The out-of-plane deformation of the (0°/90°) laminated plate with FSFS boundary under SDL.

square laminated plate subjected to an SDL. We employed the FEM in Abaqus to model the plate, and Fig. 10 compares the stress distributions on the laminate surface at  $z=h/2$  predicted by Abaqus and the proposed PINN framework. It can be observed that the stress components predicted by PINN show excellent agreement with the FEM results. Additionally, Table 2 compares the stress components at the center and corners of the laminated plates for all lay-ups with the analytical solutions provided by Reddy [62]. The comparisons demonstrate that our PINN framework achieves ideal predictive performance for the stress fields under different loading cases.

#### 4.3. Angle-ply rectangular laminated plate under uniformly distributed load

Following the previous two validation cases on cross-ply square



laminated plates, we extended our framework to the angle-ply rectangular laminated plates. Altunsaray and Bayer [65] provided their FEM results for the maximum deflection of the angle-ply rectangular laminated plates under the UDL. Their study considered 16-layer symmetric laminated composite plates with different lay-up configurations. Take symmetric laminated plates with the lay-ups of  $(-45_2/0_2/45_2/90_2)_s$  and  $(0_2/90_2/-45_2/45_2)_s$  used in their study to conduct the validation. The following material properties for each layer in laminated plates are adopted in our computational framework:

$$E_1 = 148 \text{ GPa}, E_2 = 9.65 \text{ GPa}, G_{12} = 4.45 \text{ GPa}, \nu_{12} = 0.3$$

For this validation, the plate short edge length  $b$  was set to 100mm, with the plate thickness  $h$  of 3.2mm and each layer thickness  $t$  of 0.2mm. A UDL  $q$  of 100kPa was applied to the laminated plate. After configuring the geometric parameters, BCs, and load cases in our PINN framework, we conducted 10,000 epochs training with a learning rate of  $10^{-3}$  to predict the maximum  $z$ -direction displacement of the plate with six different aspect ratios ( $a/b$ ). After around 6000 epochs, the maximum deflection predictions of PINN frameworks can approximate the FEM results and tend to stabilize. Figs. 11 and 12 show the final predicted deflections along the  $y$ -axis ( $x=0$ ) for the plates with different aspect ratios under clamped and simply supported BCs.

All comparisons between the predicted results and the FEM results are presented in Table 3. For the angle-ply laminated plates with different aspect ratios, the predictions for maximum  $z$ -direction displacement closely align with the FEM results, with relative errors within 5 % for all cases. Combined with the previous validation, it is evident that our PINN framework demonstrates reliable predictive performance even when the lay-ups and aspect ratio of the laminated plates vary.

#### 4.4. Laminated plate with arbitrary boundary combinations

In the final validation, we aim to extend our PINN computational framework to laminated plates with arbitrary boundary conditions. For a more convenient representation of different boundary combinations, we use a four-letter notation system in this study, where "C" represents clamped boundaries, "S" represents simply supported boundaries and "F" represents free boundaries. The sequence of the letters corresponds to

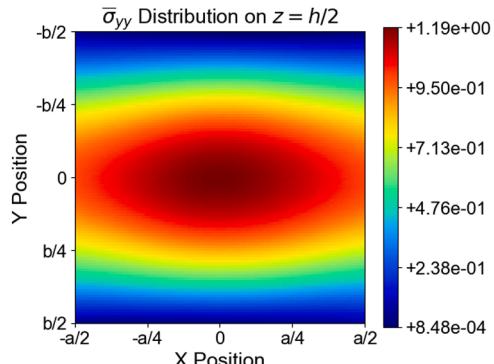


Fig. 15. The normal stress distribution of the (0°/90°) laminated plate with FSFS boundary under SDL.

**Table 5**

Comparison of predicted and analytical solutions [62] for the central deflection and in-plane normal stress components of the antisymmetric square laminated plates under SDL.

FSFS	2-layer ( $0^\circ/90^\circ$ ) laminated plate under SDL			10-layer ( $0^\circ/90^\circ$ ) <sub>5</sub> laminated plate under SDL		
	Predicted results	Analytical solutions	Error	Predicted results	Analytical solutions	Error
$\bar{w}_0(0, 0)$	1.773	1.777	0.84 %	0.647	0.665	2.71 %
$\bar{\sigma}_{xx}(0, 0, -h/2)$	-0.236	-0.240	1.69 %	-0.167	-0.173	3.47 %
$\bar{\sigma}_{yy}(0, 0, h/2)$	1.188	1.186	0.13 %	0.729	0.748	2.54 %
FSSS	Predicted results	Analytical solutions	Error	Predicted results	Analytical solutions	Error
$\bar{w}_0(0, 0)$	1.415	1.471	3.81 %	0.567	0.579	2.07 %
$\bar{\sigma}_{xx}(0, 0, -h/2)$	-0.442	-0.444	0.45 %	-0.308	-0.299	3.01 %
$\bar{\sigma}_{yy}(0, 0, h/2)$	0.955	0.984	2.91 %	0.656	0.653	0.46 %
FSCS	Predicted results	Analytical solutions	Error	Predicted results	Analytical solutions	Error
$\bar{w}_0(0, 0)$	0.963	0.980	1.73 %	0.373	0.380	1.84 %
$\bar{\sigma}_{xx}(0, 0, -h/2)$	-0.292	-0.304	3.95 %	-0.198	-0.187	3.21 %
$\bar{\sigma}_{yy}(0, 0, h/2)$	0.679	0.656	3.51 %	0.438	0.428	2.34 %
CSCS	Predicted results	Analytical solutions	Error	Predicted results	Analytical solutions	Error
$\bar{w}_0(0, 0)$	0.423	0.429	1.40 %	0.166	0.167	0.72 %
$\bar{\sigma}_{xx}(0, 0, h/2)$	-0.467	-0.480	2.71 %	-0.327	-0.317	3.15 %
$\bar{\sigma}_{yy}(0, 0, h/2)$	0.281	0.291	3.44 %	0.186	0.191	2.62 %
SSCS	Predicted results	Analytical solutions	Error	Predicted results	Analytical solutions	Error
$\bar{w}_0(0, 0)$	0.678	0.664	2.11 %	0.273	0.266	2.63 %
$\bar{\sigma}_{xx}(0, 0, -h/2)$	-0.579	-0.566	2.30 %	-0.374	-0.383	2.35 %
$\bar{\sigma}_{yy}(0, 0, h/2)$	0.441	0.448	1.56 %	0.304	0.303	0.33 %

the boundaries at  $x=-a/2$ ,  $y=-b/2$ ,  $x=a/2$ , and  $y=b/2$ . For example, the notation "FSCS" denotes the boundary combination shown in Fig. 13.

Previous validations have demonstrated the effectiveness of imposing hard constraints on the neural network outputs for fully clamped or fully simply supported boundaries. By adjusting and combining corresponding boundary correct functions for each plate edge, the bending behavior of the laminated plates with arbitrary boundary combinations can be predicted. Specifically, for CFCs boundary combination, the hard constraints need to be modified as follows:

$$\begin{aligned} Net_w(x, y) &= Net_w(x, y)((x+1)^2(x-1)^2(y+1)) \\ Net_u(x, y) &= Net_u(x, y)((x+1)(x-1)(y+1)) \\ Net_v(x, y) &= Net_v(x, y)((x+1)(x-1)) \end{aligned} \quad (24)$$

Bhaskar and Kaushik [64] provided analytical solutions for the cross-ply square laminated plates with any combination of clamped and simply supported boundaries under the UDL. The square laminated plates discussed in their study have symmetric lay-ups of ( $0^\circ$ ) and ( $0^\circ/90^\circ/0^\circ$ ). Using the same framework configurations to conduct 10,000 epochs of training with a learning rate of  $10^{-3}$ , the comparison between nondimensionalized central deflection predictions and analytical solutions is presented in Table 4.

Further incorporating free boundaries into consideration, we apply our PINN framework to 2-layer and 10-layer antisymmetric plates with lay-ups of ( $0^\circ/90^\circ$ ) and ( $0^\circ/90^\circ$ )<sub>5</sub> under the SDL. Covert the PINN-predicted z-direction displacement and stress components using Eqs. (18) and (23). Figs. 14 and 15 show the out-of-plane deformation and normal stress distribution of the ( $0^\circ/90^\circ$ ) laminated plate with FSFS boundary. For the laminated plates with other boundary combinations, we use the analytical solutions provided by Reddy [62] to assess the predictive performance of our PINN framework. As shown in Table 5, the predicted central deflection and central normal stress components on plate surfaces align closely with the analytical solutions. This series of validations demonstrates that the energy-based PINN framework can accurately predict the bending behavior of laminated plates with arbitrary boundary combinations by imposing appropriate hard constraints on the neural network outputs.

## 5. Conclusions

This study develops a data-free PINN framework to predict the bending deformation behavior of laminated composite plates, exploring the potential of PINNs in the application to orthotropic materials. By

incorporating the CLPT as the physical constraint into the neural network training process, the trained PINN accurately predicts the mid-plane displacement field of the laminates. Additionally, its capability to predict stress and moment distributions within the solution domain is validated through comparisons with existing analytical solutions and FEM results. The proposed framework constructs an energy-based loss function based on the principle of minimum potential energy. Instead of directly solving high-order PDEs, it approximates weak-form solutions by numerically integrating the total potential energy, thereby reducing computational complexity. To address the inherent limitations of the energy-based approach in handling essential BCs, we design a series of boundary correction functions to enforce hard constraints on the neural network outputs, effectively eliminating boundary residuals and simplifying hyperparameter tuning.

The most apparent advantage of our PINN framework is its independence from the training dataset, requiring only the coordinates of sampling points within the solution domain. Compared to traditional data-driven methods that rely on experimental data or generated datasets for fitting, the PINN framework shows great promise for analyzing composite structures in data-scarce scenarios. However, this framework also has certain application limitations. The energy-based approach exhibits high sensitivity to sampling point distribution and density, while enforcing hard constraints on boundaries with complex stress conditions or dynamic BCs poses additional implementation challenges.

Future research will focus on leveraging the mesh-free nature of PINN to extend the framework for analyzing geometric and material nonlinearities. Moreover, the flexibility of PINN holds significant potential for solving inverse problems such as material parameter identification, as well as incorporating transfer learning to enhance computational efficiency.

## CRediT authorship contribution statement

Weixi Wang: Writing – original draft, Visualization, Validation, Investigation. Huu-Tai Thai: Writing – review & editing, Supervision, Methodology, Conceptualization.

## 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

This work was supported by the Australian Research Council (ARC) under Future Fellowship Grant (FT200100024). The financial support is gratefully acknowledged.

## Data availability

No data was used for the research described in the article.

## References

- [1] Y. Xu, X. Liu, X. Cao, C. Huang, E. Liu, S. Qian, et al., Artificial intelligence: a powerful paradigm for scientific research, *The Innov.* (2021) 100179.
- [2] J. Betty Jane, E.N Ganesh, Big data and internet of things for smart data analytics using machine learning techniques, *Lect. Notes Data Eng. Commun. Technol.* (2020) 213–223.
- [3] J. Vamathevan, D. Clark, P. Czodrowski, I. Dunham, E. Ferran, G. Lee, et al., Applications of machine learning in drug discovery and development, *Nat. Rev. Drug Discov.* (2019) 463–477.
- [4] K. Choudhary, B. DeCost, C. Chen, A. Jain, F. Tavazza, R. Cohn, et al., Recent advances and applications of deep learning methods in materials science, *NPJ Comput. Mater.* (2022) 59.
- [5] Y. Chen, X. Wu, A. Hu, G. He, G. Ju, Social prediction: a new research paradigm based on machine learning, *J. Chin. Sociol.* (2021) 1–21.
- [6] R.H. Jhaveri, A. Revathi, K. Ramana, R. Raut, R.K. Dhanaraj, A review on machine learning strategies for real-world engineering applications, *Mob. Inf. Syst.* (2022) 1833507.
- [7] C. Málaga-Chuquitaype, Machine learning in structural design: an opinionated review, *Front Built Environ.* (2022) 815717.
- [8] A. Famili, W.M. Shen, R. Weber, E. Simoudis, Data preprocessing and intelligent data analysis, *Intell. Data Anal.* (1997) 3–23.
- [9] M. Crippa, A. Cardellini, M. Cioni, G. Csányi, G.M. Pavan, Machine learning of microscopic structure-dynamics relationships in complex molecular systems, *Mach. Learn. Sci. Technol.* (2023) 045044.
- [10] H. Adeli, C. Yeh, Perceptron learning in engineering design, *Comput. Aided Civil Infras. Eng.* (1989) 247–256.
- [11] G. Cerè, Y. Rezgui, W. Zhao, I. Petri, A machine learning approach to appraise and enhance the structural resilience of buildings to seismic hazards, *Structures* (2022) 1516–1529.
- [12] H. Sun, H.V. Burton, H. Huang, Machine learning applications for building structural design and performance assessment: state-of-the-art review, *J. Build. Eng.* (2021).
- [13] B.P. Koya, S. Aneja, R. Gupta, C. Valeo, Comparative analysis of different machine learning algorithms to predict mechanical properties of concrete, *Mech. Adv. Mater. Struct.* (2022) 4032–4043.
- [14] M.A. Shaheen, R. Presswood, S. Afshan, Application of machine Learning to predict the mechanical properties of high strength steel at elevated temperatures based on the chemical composition, *Structures* (2023) 17–29.
- [15] M. Flah, I. Nunez, W. Ben Chaabene, M.L. Nehdi, Machine learning algorithms in civil structural health monitoring: A systematic review, *Arch. Comput. Methods Eng.* (2021) 2621–2643.
- [16] T. Fleet, K. Kamei, F. He, M.A. Khan, K.A. Khan, A. Starr, A machine learning approach to model interdependencies between dynamic response and crack propagation, *Sensors* (2020) 6847.
- [17] A. Gomez-Cabrera, P.J. Escamilla-Ambrosio, Review of machine-learning techniques applied to structural health monitoring systems for building and bridge structures, *Appl. Sci.* (2022) 10754.
- [18] M.Z. Naser, V. Kodur, H.T. Thai, R. Hawileh, J. Abdalla, V.V. Degtyarev, StructuresNet and FireNet: benchmarking databases and machine learning algorithms in structural and fire engineering domains, *J. Build. Eng.* (2021) 102977.
- [19] S.R. Vadyala, S.N. Betgeri, J.C. Matthews, E. Matthews, A review of physics-based machine learning in civil engineering, *Res. Eng.* (2022) 100316.
- [20] H.T. Thai, Machine learning for structural engineering: a state-of-the-art review, *Structures* (2022) 448–491.
- [21] D.C. Feng, W.J. Wang, S. Mangalathu, G. Hu, T. Wu, Implementing ensemble learning methods to predict the shear strength of RC deep beams with/without web reinforcements, *Eng. Struct.* (2021) 111979.
- [22] Y. Gong, G. Liu, Y. Xue, R. Li, L. Meng, A survey on dataset quality in machine learning, *Inf. Softw. Technol.* (2023) 107268.
- [23] I.H. Sarker, Machine learning: algorithms, real-world applications and research directions, *SN Comput.* (2021) 160.
- [24] E.S. Muckley, J.E. Saal, B. Meredig, C.S. Roper, J.H. Martin, Interpretable models for extrapolation in scientific machine learning, *Digit. Discov.* (2023) 1425–1435.
- [25] Z. Chen, Y. Liu, H. Sun, Physics-informed learning of governing equations from scarce data, *Nat. Commun.* (2021) 6136.
- [26] A. McGovern, R. Lagerquist, D.J. Gagne, G.E. Jergensen, K.L. Elmore, C. R. Homeyer, et al., Making the black box more transparent: understanding the physical implications of machine learning, *Bull. Am. Meteorol. Soc.* (2019) 2175–2199.
- [27] A.M. Tartakovsky, C.O. Marrero, P. Perdikaris, G.D. Tartakovsky, D. Barajas-Solano, Physics-informed deep neural networks for learning parameters and constitutive relationships in subsurface flow problems, *Water Resour. Res.* (2020) 101029.
- [28] G.E. Karniadakis, I.G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang, Physics-informed machine learning, *Nat. Rev. Phys.* (2021) 422–440.
- [29] J. Chen, Y. Chen, X. Xu, W. Zhou, G. Huang, A physics-guided machine learning for multifunctional wave control in active metabeams, *Extreme Mech. Lett.* (2022) 101827.
- [30] W.Q. Hao, L. Tan, X.G. Yang, D.Q. Shi, M.L. Wang, G.L. Miao, et al., A physics-informed machine learning approach for notch fatigue evaluation of alloys used in aerospace, *Int. J. Fatigue* (2023) 107536.
- [31] R. Wang, K. Kashinath, M. Mustafa, A. Albert, R. Yu, Towards physics-informed deep learning for turbulent flow prediction, in: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2020, pp. 1457–1466.
- [32] Z.K. Lawal, H. Yassin, D.T.C. Lai, A. Che Idris, Physics-Informed neural Network (PINN) Evolution and beyond: a systematic literature review and bibliometric analysis, *Big Data Cognitive Comput.* (2022) 140.
- [33] L. Lu, X. Meng, Z. Mao, G.E. Karniadakis, DeepXDE: a deep learning library for solving differential equations, *SIAM Review* (2021) 208–228.
- [34] A. Güneş Baydin, B.A. Pearlmutter, A. Andreyevich Radul, J. Mark Siskind, Automatic differentiation in machine learning: a survey, *J. Mach. Learn. Res.* (2018) 1–43.
- [35] T. Zhou, E.L. Drogue, A. Mosleh, Physics-informed deep learning: a promising technique for system reliability assessment, *Appl. Soft. Comput.* (2022) 109217.
- [36] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, *J. Comput. Phys.* (2019) 686–707.
- [37] S. Cuomo, V.S. Di Cola, F. Giampaolo, G. Rozza, M. Raissi, F. Piccialli, Scientific machine learning through physics-Informed neural networks: where we are and what's next, *J. Sci. Comput.* (2022) 88.
- [38] X. Jin, S. Cai, H. Li, G.E. Karniadakis, NSFnets (Navier-Stokes flow nets): physics-informed neural networks for the incompressible Navier-Stokes equations, *J. Comput. Phys.* (2021) 109951.
- [39] P.H. Chiu, J.C. Wong, C. Ooi, M.H. Dao, Y.S. Ong, CAN-PINN: a fast physics-informed neural network based on coupled-automatic-numerical differentiation method, *Comput. Methods Appl. Mech. Eng.* (2022) 114909.
- [40] P. Sharma, W.T. Chung, B. Akoush, M. Ihme, A review of physics-informed machine learning in fluid mechanics, *Energies* (2023) 2343.
- [41] 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. Eng.* (2021) 113741.
- [42] W. Li, M.Z. Bazant, J. Zhu, A physics-guided neural network framework for elastic plates: comparison of governing equations-based and energy-based approaches, *Comput. Methods Appl. Mech. Eng.* (2021) 113933.
- [43] M. Vahab, E. Haghighat, M. Khaleghi, N. Khalil, A physics-informed neural network approach to solution and identification of biharmonic equations of elasticity, *J. Eng. Mech.* (2022) 04021154.
- [44] E. Zhang, M. Dao, G.E. Karniadakis, S. Suresh, Analyses of internal structures and defects in materials using physics-informed neural networks, *Sci. Adv.* (2022) eabk0644.
- [45] Y. Gu, C. Zhang, M.V. Golub, Physics-informed neural networks for analysis of 2D thin-walled structures, *Eng. Anal. Bound Elem.* (2022) 161–172.
- [46] S. Niu, E. Zhang, Y. Bazilevs, V. Srivastava, Modeling finite-strain plasticity using physics-informed neural network and assessment of the network performance, *J. Mech. Phys. Solids* (2023) 105177.
- [47] T. Kapoor, H. Wang, A. Nunez, R. Dollevoet, Physics-informed neural networks for solving forward and inverse problems in complex beam systems, *IEEE Trans. Neural Netw. Learn Syst.* (2024) 35.
- [48] N. Radin, S. Klinkel, O. Altay, Effects of variational formulations on physics-informed neural network performance in solid mechanics, *PAMM* (2023) e202300222.
- [49] S. Berrone, C. Canuto, M. Pintore, Variational physics informed neural networks: the role of quadratures and test functions, *J. Sci. Comput.* (2022) 100.
- [50] J. He, D. Abueidda, R. Abu Al-Rub, S. Koric, I. Jasius, A deep learning energy-based method for classical elastoplasticity, *Int. J. Plast.* (2023) 103531.
- [51] E. Kharazmi, Z. Zhang, G.E.M. Karniadakis, Hp-VPINNs: variational physics-informed neural networks with domain decomposition, *Comput. Methods Appl. Mech. Eng.* (2021) 113547.
- [52] C. Liu, H.A. Wu, A variational formulation of physics-informed neural network for the applications of homogeneous and heterogeneous material properties identification, *Int. J. Appl. Mech.* (2023) 2350065.
- [53] J. Bai, T. Rabczuk, A. Gupta, L. Alzubaidi, Y. Gu, A physics-informed neural network technique based on a modified loss function for computational 2D and 3D solid mechanics, *Comput. Mech.* (2023) 543–562.
- [54] L. Ning, Z. Cai, H. Dong, Y. Liu, W. Wang, Physics-informed neural network frameworks for crack simulation based on minimized peridynamic potential energy, *Comput. Methods Appl. Mech. Eng.* (2023) 116430.
- [55] A.P. Mouritz, E. Gellert, P. Burchill, K. Challis, Review of advanced composite structures for naval ships and submarines, *Compos. Struct.* (2001) 21–42.
- [56] A. Garg, T. Mukhopadhyay, M.O. Belarbi, L. Li, Random forest-based surrogates for transforming the behavioral predictions of laminated composite plates and shells from FSDT to Elasticity solutions, *Compos. Struct.* (2023) 116756.

- [57] C. Furtado, L.F. Pereira, R.P. Tavares, M. Salgado, F. Otero, G. Catalanotti, et al., A methodology to generate design allowables of composite laminates using machine learning, *Int. J. Solids Struct.* (2021) 111095.
- [58] B.B. Mishra, A. Kumar, P. Samui, T. Roshni, Buckling of laminated composite skew plate using FEM and machine learning methods, *Eng. Comput.* (2021) 501–528 (Swansea, Wales).
- [59] J. Reiner, R. Vaziri, N. Zobeiry, Machine learning assisted characterisation and simulation of compressive damage in composite laminates, *Compos. Struct.* (2021) 114290.
- [60] S.S. Sorour, C.A. Saleh, M. Shazly, A review on machine learning implementation for predicting and optimizing the mechanical behaviour of laminated fiber-reinforced polymer composites, *Heliyon* (2024) e33681.
- [61] N. Mentges, B. Dashtbozorg, S.M. Mirkhalaf, A micromechanics-based artificial neural networks model for elastic properties of short fiber composites, *Compos. B Eng.* (2021) 108736.
- [62] J.N. Reddy, Mechanics of laminated composite plates and shells, *Mech. Laminat. Compos. Plates Shells* (2003) 101201.
- [63] K. Bhaskar, B. Kaushik, Analysis of clamped unsymmetric cross-ply rectangular plates by superposition of simple exact double fourier series solutions, *Compos. Struct.* (2005) 303–307.
- [64] K. Bhaskar, B. Kaushik, Simple and exact series solutions for flexure of orthotropic rectangular plates with any combination of clamped and simply supported edges, *Compos. Struct.* (2004) 63–68.
- [65] E. Altunsaray, I. Bayer, Deflection and free vibration of symmetrically laminated quasi-isotropic thin rectangular plates for different boundary conditions, *Ocean Eng.* (2013) 197–222.
- [66] R. Khandan, S. Noroozi, P. Sewell, J. Vinney, The development of laminated composite plate theories: a review, *J. Mater. Sci.* (2012) 5901–5910.
- [67] S. Mowlavi, S. Nabi, Optimal control of PDEs using physics-informed neural networks, *J. Comput. Phys.* (2023) 111731.
- [68] C.A. Yan, R. Vescovinci, L. Dozio, A framework based on physics-informed neural networks and extreme learning for the analysis of composite structures, *Comput. Struct.* (2022) 106761.
- [69] C. Cheng, H. Meng, Y.Z. Li, G.T. Zhang, Deep learning based on PINN for solving 2 DOF vortex induced vibration of cylinder, *Ocean Eng.* (2021) 109932.
- [70] Bischof R., Kraus M. Multi-objective loss balancing for physics-informed deep learning. ArXiv preprint. 2021;2110.09813.
- [71] J. Wang, Y.L. Mo, B. Izzuddin, C.W. Kim, Exact dirichlet boundary physics-informed Neural network EPINN for solid mechanics, *Comput. Methods Appl. Mech. Eng.* 414 (2023) 116184.
- [72] J.H. Bastek, D.M. Kochmann, Physics-informed neural networks for shell structures, *Eur. J. Mech. A Solids.* (2023) 104849.
- [73] E. Samaniego, C. Anitescu, S. Goswami, V.M. Nguyen-Thanh, H. Guo, K. Hamdia, et al., An energy approach to the solution of partial differential equations in computational mechanics via machine learning: concepts, implementation and applications, *Comput. Methods Appl. Mech. Eng.* (2020) 112790.
- [74] E. Weinan, B. Yu, The deep ritz method: a deep learning-based numerical algorithm for solving variational problems, *Commun. Math. Stat.* (2018) 1–12.