Physics-informed neural network for nonlinear analysis of cable net structures

Dai D. Mai[a,1], Tri Diep Bao[b,2], Thanh-Danh Lam[b,3], Hau T. Mai[b,*]

[a] *Faculty of Mechanical Engineering, Ho Chi Minh City University of Technology and Education, Ho Chi Minh City, Viet Nam*

[b] *Faculty of Mechanical Engineering, Industrial University of Ho Chi Minh City, Ho Chi Minh City, Vietnam*

Abstract

In this study, a Physics-Informed Neural Network (PINN) framework is extended and applied to predict the geometrically nonlinear responses of pretensioned cable net structures without utilizing any incremental-iterative algorithms as well as Finite Element Analyses (FEAs). Instead of solving nonlinear equations as in existing numerical models, the core idea behind this approach is to employ a Neural Network (NN) that minimizes a loss function. This loss function is designed to guide the learning process of the network based on Total Potential Energy (TPE), pretension forces, and Boundary Conditions (BCs). The NN itself models the displacements given the corresponding coordinates of joints as input data, with trainable parameters including weights and biases that are regarded as design variables. Within this computational framework, these parameters are automatically adjusted through the training process to get the minimum loss function. Once the learning is complete, the nonlinear responses of cable net structures can be easily and quickly obtained. A series of numerical examples is investigated to demonstrate the effectiveness and applicability of the PINN for the geometrically nonlinear analysis of cable net structures. The obtained results indicate that the PINN framework is remarkably simple to use, robust, and yields higher accuracy.

Keywords: Physics-informed neural network, Nonlinear analysis, Cable net structures, Deep neural network, Geometric nonlinearity, Static analysis

[*] Corresponding author. *E-mail: maitienhau@iuh.edu.vn, maitienhaunx@gmail.com*

[1] *E-mail: daimd@hcmute.edu.vn*

[2] *E-mail: diepbaotri@iuh.edu.vn*

[3] *E-mail: lamthanhdanh@iuh.edu.vn*

1. Introduction

Owing to the significant structural advantages of cable elements, they have been widely utilized as crucial components in tension structures, particularly for designs requiring high strength, lightweight, high degree of flexibility, and cost-effective designs. Examples of such applications included bridges supported by cables, power supply lines, and large-span roof structures [1, 2]. However, cable structures face potential threats due to the high geometric nonlinearity that can lead to instability. Therefore, the effects of large deflection must be meticulously taken into account when analyzing these structures. And this poses a key challenge that has garnered the interest of numerous researchers.

Overall, various algorithms have been developed for the analysis of cable structures, broadly categorized into two main groups: the stiffness-based method and the direct energy minimization method [3]. In the first one, the stiffness matrix was iteratively updated as the structure underwent deformation due to the change of geometry and material properties. To obtain the stiffness matrix, two different approaches were employed for modeling cable elements: one based on interpolation functions, and the other based on the analytical expression of catenary elements. In the first sub-method, the finite element procedure employed interpolation polynomial functions to approximate the nonlinear behaviors of cables. And many scholars have successfully formulated two-node [4–6] and multi-node elements [7–10] based on this approach. Therein, the two-node element, being the simplest and most common, is suitable for cables with small sag and high pretension [2]. For cables with larger sag, higher-order interpolation functions are employed to model the multi-node element for improving the accuracy [1]. In the second sub-method, a two-node elastic catenary cable element is derived using exact analytical expressions to describe the realistic behavior of cables [11–14]. Compared to approximate finite element models, this method provided more accurate results and required fewer number of degrees of freedom in cable structures. Although the stiffness-based algorithms have achieved certain success, they still have limitations related to nonlinear incremental-iterative procedures [15]. To circumvent these drawbacks, the direct energy approach combined with optimization algorithms has been developed to estimate the nonlinear responses without using any incremental-iterative procedures. Several nonlinear optimization algorithms, including both gradient-based [16–20] and gradient-free methods [21, 22], have been successfully applied in the analysis of cable structures. While the gradient-free approaches have the capability to find near-optimal solutions, they often require a greater number of function evaluations and exhibit slower convergence speeds. On the other hand, the gradient-based methods converge rapidly in a few iterations but necessitate derivative information of the objective function with respect to deflections. And this is challenging to compute or may be unavailable in practical applications. As a highly effective alternative to address this challenge, the use of neural networks equipped with the capability to approximate arbitrary nonlinear functions, along with their automatic differentiation, has emerged. NN-based approaches offer the advantage of capturing complex nonlinear behavior without relying on explicit derivatives, making them well-suited for analyzing cable structures.

Unlike previous techniques, machine learning has demonstrated its power and promise as a research tool with applications across multiple fields, including language processing, voice recognition, medical diagnoses, and industrial automation, among others. Within the field of computational mechanics, NN has garnered significant interest [23–31]. Particularly, PINN has gained popularity as a robust tool for solving structural analysis problems by incorporating well-posed physical equations into the network's loss function. And the loss function plays a critical role in the success of the PINN framework. Therein, the TPE is one of the popular choices for modeling the loss function of the PINN. There are two common variants of the PINN model in an effort to minimize the TPE of the system: Deep Energy Method (DEM) and Deep Ritz Method (DRM). They have also been successfully applied to solve the mechanics problems. For instance, Nguyen-Thanh et al. [23, 32] first introduced the DEM for determining the linear and nonlinear structural responses. And then several improved versions of the DEM were developed by Huang [33], Chadha [34], and Abueidda [35]. Meanwhile, Weinan and Bing [36] first proposed the DRM which used the NN to approximate the trial solution that satisfied the boundary conditions and governing equations. Besides, an adaptive Ritz method was developed by Liu et al. [37] to solve the linear elasticity problem. Clearly, the PINN offered the potential to address several challenges faced by classical methods combined with optimization algorithms. Firstly, it provides a simulation-based approach as a mesh-free method that eliminates the necessity for discretizing the domain into a finite number of elements [35]. Hence, the PINN allows us to mitigate issues associated with the curse of dimensionality and saves much effort in generating meshes compared to conventional solvers [38]. Besides, it has the capability to capture the complex nonlinear responses of structures in any physical process [39]. Additionally, another interesting point is that all differentiable and derivative information can be easily obtained through automatic differentiation in the network's backward propagation. Therefore, this model is capable of effectively handling problems arising from the lack of gradient information [28]. In particular, PINN is capable of effectively addressing challenges posed by ill-conditioned problems [40]. Owing to the above salient advantages of the PINN model, it has proven successful in handling complex problems without discrete domain designs [35, 39, 41–43]. Until now, to the best of our knowledge, it has not yet been applied to the nonlinear analysis of cable net structures.

Motivated by previous successes, this study aims to extend and apply the PINN framework for geometrically nonlinear analysis of pretensioned cable net structures. Instead of relying on conventional incremental-iterative techniques, the NN is designed to directly estimate the cable responses by minimizing the TPE. Therein, the geometry information consists of the spatial coordinates of the joints which are regarded as the input data of the network. It is important to note that this input data constitutes the entire training dataset without labels. Therein, the unknown displacements, which serve as the network's output values, are represented by weights, biases, and the input data. Using the predicted displacement field, pretension forces, and BCs, the TPE is constructed as a derivative-free loss function to guide the training process. And the structural behaviors corresponding to the minimum loss function are obtained as soon as the learning process concludes, without utilizing any algorithms. The efficiency and simplicity of the proposed model are demonstrated through a series of examples to analyze the geometric nonlinearity of cable net structures.

The remaining article is organized as follows. Section 2 presents a robust PINN framework for the nonlinear analysis of cable net structures. Afterward, several numerical examples are investigated to demonstrate the simplicity, accuracy, and effectiveness of our approach in Section 3. Next, the efficiency of the proposed approach is discussed in Section 4. Ultimately, Section 5 draws conclusions and offers conclusions and offers prospects for further study.

2. Physics-informed neural network framework

In this section, a physics-informed NN framework is proposed to perform the geometrically nonlinear analysis of pretensioned cable net structures. The overall flow diagram is illustrated in Fig. 1, providing an overview of the proposed development approach. Unlike most existing works, the NN is designed to minimize the TPE of the structure as a loss function, instead of solving the system of nonlinear equations using conventional algorithms. This approach eliminates the need for structural analysis and incremental-iterative algorithms. Consequently, the structural analysis is transformed into an optimization problem, where the weights and biases of the NN, known as the connection parameters $\boldsymbol{\theta}$, act as the design variables. And the underlying physics is integrated apart from the NN architecture to guide the model during training, ensuring that the obtained outputs obey physics laws. Initially, a fully connected NN is established to carry out the training task. It is worth mentioning that the training data only contains the coordinates of all joints in the cable net, whilst the corresponding unknown displacements serve as the outputs. By using these predicted outputs, along with BCs, loads, and physical laws, the TPE, which comprises strain energy and external work, is formulated as the loss function. The training process utilizes feedforward (FF) and backpropagation (BP) algorithms, along with an optimizer, to adjust the connection parameters of the network. As soon as the training is complete, the displacements and other behaviors can be determined almost instantly, without requiring any additional algorithms as well as structural analyses. Further details of this procedure are provided in the following subsections to ensure a clear understanding of our approach.

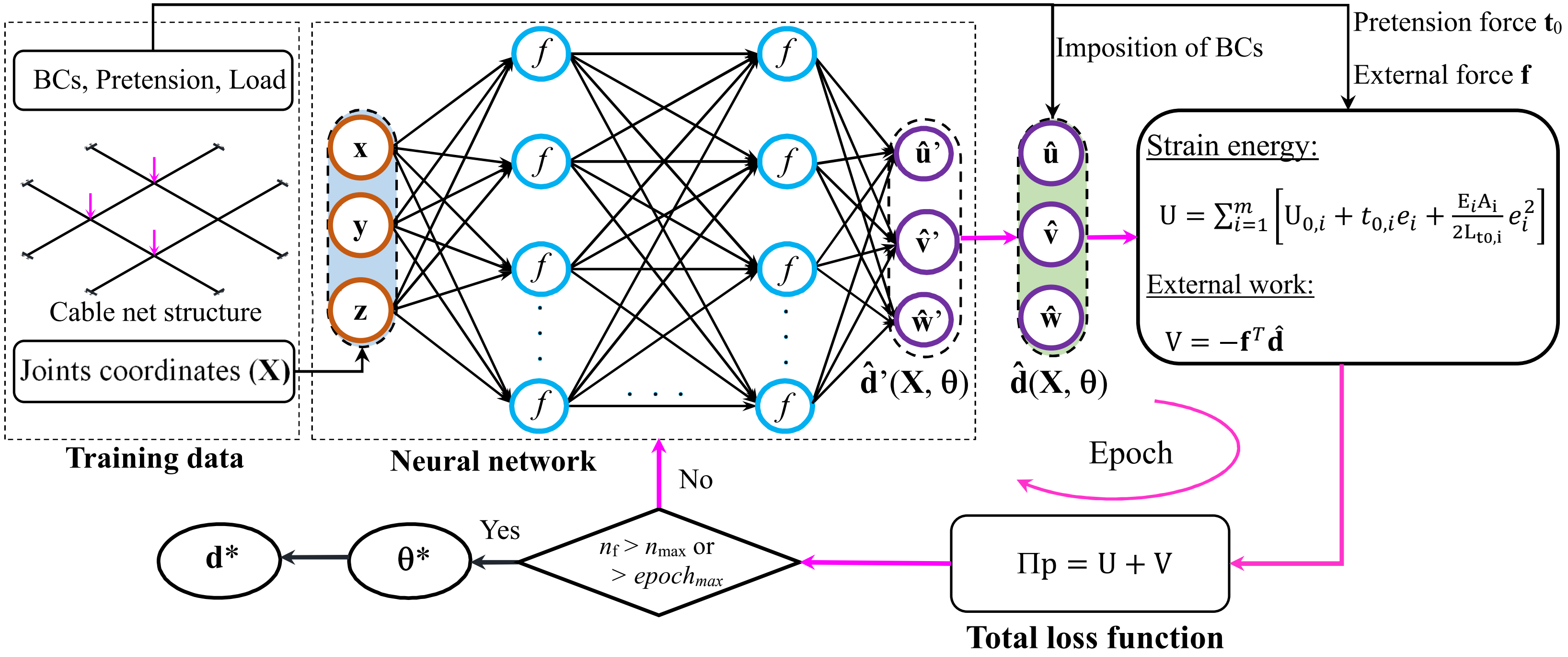
Pretension force $\mathbf{t}_0$
Imposition of BCs
External force $\mathbf{f}$
BCs, Pretension, Load
Cable net structure
Joints coordinates (**X**)
Training data
x
y
z
f
Neural network
$\hat{\mathbf{u}}$'
$\hat{\mathbf{v}}$'
$\hat{\mathbf{w}}$'
$\hat{\mathbf{d}}$'(**X**, θ)
$\hat{\mathbf{u}}$
$\hat{\mathbf{v}}$
$\hat{\mathbf{w}}$
$\hat{\mathbf{d}}$(**X**, θ)
Strain energy:
$\mathrm{U} = \sum_{i=1}^{m}\left[\mathrm{U}_{0,i} + t_{0,i}e_i + \frac{\mathrm{E}_i\mathrm{A}_\mathrm{i}}{2\mathrm{L}_{\mathrm{t0,i}}}e_i^2\right]$
External work:
$\mathrm{V} = -\mathbf{f}^T\hat{\mathbf{d}}$
Epoch
No
Yes
$n_\mathrm{f} > n_\mathrm{max}$ or $> epoch_{max}$
d*
θ*
$\Pi\mathrm{p} = \mathrm{U} + \mathrm{V}$
Total loss function

Fig. 1. Flowchart of the physics-informed NN framework for geometrically nonlinear analysis of cable net structures. The magenta line displays the training process.

2.1. *Training data*

Clearly, from the aforementioned flowchart, it can be observed that the total training data is only a set of coordinates $\mathbf{X} = [\mathbf{x}, \mathbf{y}, \mathbf{z}]$, BCs, external forces $\mathbf{f}$, and pretension forces $\mathbf{t}_0$ applied to all cable members. More concretely, the NN takes the spatial coordinates as inputs, while the displacements $\hat{\mathbf{d}}'$ as outputs, and other structural behaviors, such as strain, stress, and member forces, are not contained in the training data and unknown quantities. In other words, these outputs of the NN need to be estimated in order to minimize the TPE. Meanwhile, the other data, including the BCs, external forces, and initial tension forces, are used to support building the loss function. Obviously, it can be seen that the entire training data can be easily obtained from the geometric information and problem statement without requiring any sampling techniques or numerical methods. And this is the pivotal distinction between our model and data-driven approaches. Furthermore, its size can be easily determined as (3, n), where n is the number of joints, and 3 represents the corresponding x, y, and z coordinates of the joints.

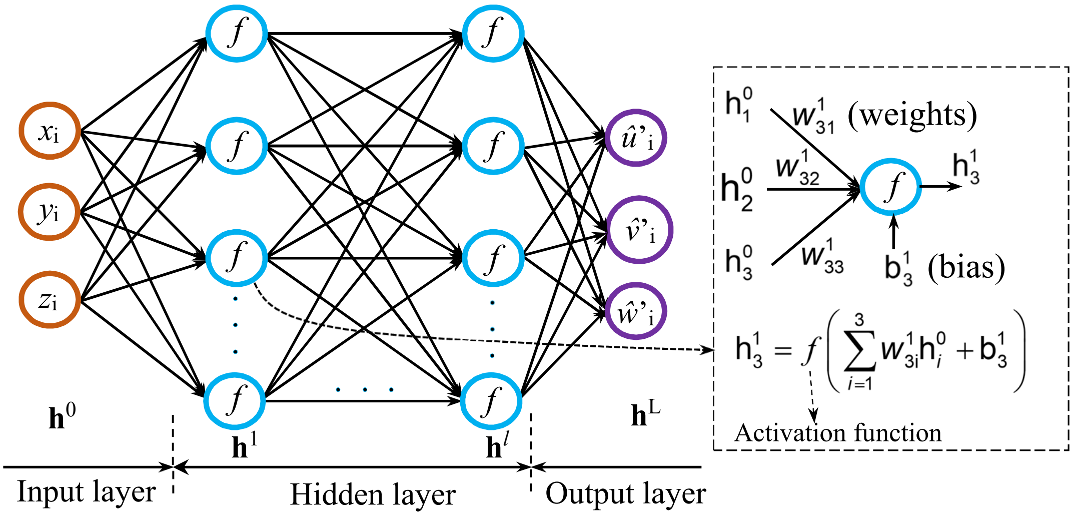

Fig. 2. A fully-connected neural network architecture.

2.2. *Physics-informed loss function*

A fully-connected NN depicted in Fig. 2 is constructed to estimate the structural responses. It consists of an input layer, two hidden layers, and an output layer. Wherein the first layer, also known as the (*0*th) layer, comprises three neurons corresponding to the coordinates (x, y, z) of the joints. The end layer, denoted as the (*L*th) layer, consists of three neurons representing the predicted displacements $(\hat{u}, \hat{v}, \hat{w})$. And the remaining layers between the input and output layers are referred to as hidden layers (*l*th). In the meantime, the complexity of the problem will affect the number of hidden neurons and hidden layers. Each neuron in the current layer is connected to all neurons in the previous layer through the connection parameters, and their initial $\boldsymbol{\theta}$ values are uniformly distributed in the range of [-1, 1].

In general, the mapping of the training data from the input to the output layers, which is known as the FF algorithm, can be expressed as $\hat{\mathbf{d}}'$: $\mathbb{R}^3 \rightarrow \mathbb{R}^3/\mathbf{X} \Rightarrow \hat{\mathbf{d}}'(\mathbf{X}, \boldsymbol{\theta})$. And the relationships between the output and input of each layer are represented as follow:

$$\begin{aligned}
&\text{in the 0th layer}: \ \mathbf{h}_0 = [x, y, z]^T \ \in \mathbb{R}^3, \\
&\text{in the } l\text{th layers: } \mathbf{h}_l = f_1\left(\mathbf{W}_l \mathbf{h}_{(l-1)} + \mathbf{b}_l\right) \in \mathbb{R}^{m_l}, \quad \text{for } 1 \leq l < L, \\
&\text{in the last layer}: \ \mathbf{h}_L = f_2\left(\mathbf{W}_L \mathbf{h}_{(L-1)} + \mathbf{b}_L\right) = [\hat{u}', \hat{v}', \hat{w}'] = \hat{\mathbf{d}}' \in \mathbb{R}^3,
\end{aligned} \tag{1}$$

where $\mathbf{h}_l$ and $\mathbf{h}_L$ are the outputs of the hidden and output layers, respectively; $\mathbf{W}(.)$ and $\mathbf{b}(.)$ denote the weight matrix and bias vector; m_l refers to the number of neurons in the lth hidden layer; f_1 and f_2 are the activation functions of the hidden and output layers. They play a central role in capturing the nonlinearity of the input-output relationship. There are several popular options, such as Sigmoid, Linear, ReLU, Softmax, LeakyReLU, and so on. In this study, the hyperbolic tangent is employed as the activation function of neurons in the hidden layers, while the output layer does not utilize the activation function. It should also be pointed out that the weights and biases of the network are the parameters that need to be trained. The total number of biases is denoted as $\sum_{i=1}^{L} p_i$, and there are $\sum_{i=1}^{L} p_{i-1} p_i$ weights ($p_0 = p_L = 3$), where p_i represents the number of neurons in the ith layer.

It is obvious that the predicted displacements $\hat{\mathbf{d}}'$ do not satisfy the BCs of the problem. In recent literature, these BCs are included as a part of the loss function, resulting in increased complexity of the objective function and inefficiency in the training process [24, 44, 45]. To overcome this challenge, several techniques for imposing BCs have been suggested to directly constrain the network outputs and ensure compliance with essential BCs [46]. In this study, the output values are imposed with the BCs before constructing the loss function. Accordingly, the displacement vector with the fulfilled BCs $\hat{\mathbf{d}}$ is expressed as follows:

$$\hat{\mathbf{d}} = \hat{\mathbf{d}}' \odot \bar{\mathbf{b}}, \tag{2}$$

with

$$\bar{\mathbf{b}} = \begin{cases} 1 \ \ \text{at the free dofs,} \\ 0 \ \ \text{otherwise,} \end{cases} \tag{3}$$

where $\bar{\mathbf{b}}$ denotes the coefficient vector corresponding to the constraints at the joints, and dofs is the degree of freedom of the cable structure.

Obviously, we strictly define the TPE with the predicted displacements $\hat{\mathbf{d}}$ that satisfy all BCs. Let us consider a pretensioned cable net system consisting of m cable members and n joints. And the TPE of the structure Π_p can be expressed as follows [16]:

$$\Pi_p = U + V, \tag{4}$$

in which

$$U = \sum_{k=1}^{m} \left[U_{0,k} + t_{0,k} e_k + \frac{E_k A_k}{2\ell_{t_0,k}} e_k^2 \right], \tag{5}$$

$$V = -\mathbf{f}^T \hat{\mathbf{d}}, \tag{6}$$

$$U_{0,k} = \frac{E_k A_k}{2\ell_{0,k}} e_{t_0,k}^2. \tag{7}$$

where U and V are the strain energy and external work; $U_{0,k}$ refers to the initial strain energy stored in the kth element as a result of pretension; $t_{0,k}$ is the initial pretension force in the kth member; E_k, A_k, $\ell_{t_0,k}$, $\ell_{0,k}$, and e_k are the Young's modulus, cross-sectional area, length caused by the pretension force, elongations caused by the pretension force and external force of the kth element, respectively; $\mathbf{f}$ denotes the external force vector; and $\hat{\mathbf{d}}$ refers to the predicted displacement vector with the fulfilled BCs.

In order to achieve the elongations of the kth element, a cable segment in space that is subjected to a pretension force with end coordinates $(x_i,\ y_i,\ z_i)$ and $(x_j,\ y_j,\ z_j)$ is considered. As a result, its length before the application of the initial tensile force is denoted as $\ell_{0,k}$, and after the application of the initial tensile force is denoted as $\ell_{t_0,k}$. They are defined as follows:

$$\ell_{t_0,k} = \sqrt{\ell_{0x,k}^2 + \ell_{0y,k}^2 + \ell_{0z,k}^2}, \tag{8}$$

$$\ell_{0,k} = \frac{\ell_{t_0,k} E_k A_k}{t_{0,k} + E_k A_k}, \tag{9}$$

in which $\ell_{0x,k} = x_j - x_i;\ \ \ell_{0y,k} = y_j - y_i;\ \ \ell_{0z,k} = z_j - z_i$; E_k, A_k, and $t_{0,k}$ denote the Young's modulus, cross-sectional area, and the initial pretension force in the kth member, respectively.

When the cable system is subjected to the external forces, the length of the kth element $\ell_{f,k}$ corresponding to the predicted displacements $(\hat{u}_i,\ \hat{v}_i,\ \hat{w}_i, \hat{u}_j,\ \hat{v}_j,\ \hat{w}_j)$ can be given by:

$$\ell_{f,k} = \sqrt{\ell_{fx,k}^2 + \ell_{fy,k}^2 + \ell_{fz,k}^2}, \tag{10}$$

where $\ell_{fx,k} = \ell_{0x,k} + \hat{u}_j - \hat{u}_i;\ \ \ell_{fy,k} = \ell_{0y,k} + \hat{v}_j - \hat{v}_i;\ \ell_{fz,k} = \ell_{0z,k} + \hat{w}_j - \hat{w}_i; \hat{u}_i,\ \hat{v}_i,\ \hat{w}_i, \hat{u}_j,\ \hat{v}_j,\ \hat{w}_j$ are the predicted displacements in the x, y,and z directions at joints i and j.

It can be seen that the elongations before $e_{t_0,k}$ and after e_k the application of the external force are easily obtained from $\ell_{0,k}$, $\ell_{t_0,k}$ and $\ell_{f,k}$, as follows:

$$e_{t_0,k} = \ell_{t_0,k} - \ell_{0,k}, \tag{11}$$

$$e_k = \ell_{f,k} - \ell_{t_0,k}. \tag{12}$$

Once the elongations are determined, the TPE is quite obtained by combining the strain energy and external work, utilizing the predicted displacement field and loads. It serves as a loss function that is minimized during the training process. And its formulation can be expressed as follows:

$$\mathcal{L}(\boldsymbol{\theta})\ = U(\boldsymbol{\theta}) + V(\boldsymbol{\theta}). \tag{13}$$

Note that unlike the classical PINN, our loss function is established only based on the predicted displacement field without including partial differential values obtained in the back-propagation process [15, 28, 47]. Precisely because of this feature, it helps save the computational time required to estimate the loss function value. Furthermore, the convergence rate is improved due to the derivative-free loss function. Instead of using the combination of incremental-iterative methods and FEA to resolve the nonlinear problem as in the conventional approaches, our model minimizes the loss function by training the network to determine the optimal parameters $\boldsymbol{\theta}^*$

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\arg\min}\ \mathcal{L}\left(\boldsymbol{\theta}\right). \tag{14}$$

To achieve this objective, Adam optimizer, also known as an extension of the gradient descent optimization algorithm, is employed for training. It is worth mentioning that the BP algorithm, integrated into the general NN, enables easy and automatic differentiation of the loss function with respect to the connection parameters $\nabla\mathcal{L}\left(\boldsymbol{\theta}\right)$. For more detailed information, interested readers are recommended to consult Ref. [48]. Note that for this study, the batch size is set equal to the training sample size and when each iteration will represent an epoch because the entire training data is processed in a single iteration. Consequently, at epoch (iteration) (t+1), the network parameters are adjusted according to the minimum lost function

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta\frac{\mathbf{m}_{t+1}\sqrt{1-\beta_2}}{\left(1-\beta_1\right)\left(\sqrt{\mathbf{v}_{t+1}}+\varepsilon\sqrt{1-\beta_2}\right)}, \tag{15}$$

in which $\mathbf{m}_{t+1}$ and $\mathbf{v}_{t+1}$ are given by

$$\begin{aligned} \mathbf{m}_{t+1} &= \beta_1\mathbf{m}_t + \left(1-\beta_1\right)\cdot\nabla\mathcal{L}\left(\boldsymbol{\theta}_t\right), \\ \mathbf{v}_{t+1} &= \beta_2\mathbf{v}_t + \left(1-\beta_2\right)\cdot\nabla\mathcal{L}^2\left(\boldsymbol{\theta}_t\right), \end{aligned} \tag{16}$$

where $\boldsymbol{\theta}_t$ represents the unknown parameter vector including the weights and biases at the tth iteration; $\beta_{1,2}$ denote the decay rate which are used to control the first $\mathbf{m}$ and second $\mathbf{v}$ raw moment vectors; η and ε are the learning rate and constant to make sure numerical stability, respectively. In this study, the model was trained using the default values as suggested by Kingma and Ba [48]. The above whole process corresponding to stages 5-12 in Algorithm 1, was repeated until a satisfactory agreement was achieved. This phase is commonly referred to as the training phase. And all of the cable responses are indicated immediately upon the completion of the training without using any structural analyses. In this study, an early stopping criterion, which is based on the disparity of gradients, is proposed to avoid unnecessary iterations efficiently [49, 50]. Accordingly, the training task concludes when the norm of the residual gradient of two consecutive epochs $\|\nabla\mathcal{L}(\boldsymbol{\theta}_t) - \nabla\mathcal{L}(\boldsymbol{\theta}_{t-1})\|$ must not be greater than 10^{-3} in the last 15 epochs ($n_{max} = 15$). A pseudo-code of our scheme, as shown in Algorithm 1, describes the main steps of the proposed framework.

Algorithm 1: Learning process of the network for nonlinear analysis of cable net

```
Input:
    - Cable net: properties of the structure, pretension forces, applied loads
    - NN: number of hidden neurons, layers, activation function, Adam optimizer
Output: optimal parameters $\boldsymbol{\theta}$, structural responses
Collect the spatial coordinates of joints
Construct a NN with initial parameters $\boldsymbol{\theta}_0$ distributed in the range [-1, 1]
Set the parameters of Adam optimizer as the default settings [48]
while $n_f < n_{max}$ or $epoch_{max}$ is not reached do
    Predict $(\hat{\mathbf{d}}'(\mathbf{X}, \boldsymbol{\theta}_t))$ using the FF
    Applied BCs on $\hat{\mathbf{d}}'$ to obtain $\hat{\mathbf{d}}$ by Eq. (2)
    Compute the elongation $e$ of cable member by Eq. (12)
    Calculate the strain energy $U$ and external work $V$ by Eqs. (5)-(6)
    Loss function $\mathcal{L}(\boldsymbol{\theta}_t)$ is estimated by Eq. (13)
    $\nabla\mathcal{L}(\boldsymbol{\theta}_t)$ is calculated automatically by the BP algorithm
    Update parameters $\boldsymbol{\theta}_{t+1}$ of the network by Eq. (15)
    If $\|\nabla\mathcal{L}(\boldsymbol{\theta}_t) - \nabla\mathcal{L}(\boldsymbol{\theta}_{t-1})\| < 10^{-3}$ then $n_f = n_f + 1$
    t=t+1
```

2.3. Detailed example

In order to easily understand the operation principle of the PINN, the sequence in which the training process during the first iteration/epoch is carried out to minimize the TPE in the simple cable net model subjected to a concentrated load P = 15 N at the center point, as depicted in Fig. 3. The pretension force, cross-sectional area, and Young's modulus of all cables are set to 200 N, 0.785 mm^2, and 124.8 kN/mm^2, respectively. Several researchers have previously investigated this benchmark problem using various algorithms, such as Lewis [20], Halvordson [51], and Toklu [21], etc. The steps in first epoch then are as follows:

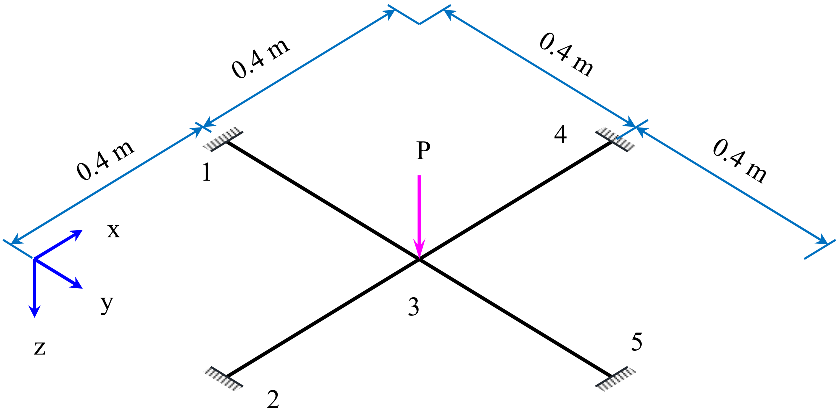

Fig. 3. Simple cable net structure.

Step 1: First, the spatial coordinates of all joints $\mathbf{X}$ are set as the entire training data in our model, which can be easily collected from the geometric information without using any numerical methods, as follows:

$$\mathbf{X} = \begin{bmatrix} 400 & 0 & 400 & 800 & 400 \\ 0 & 400 & 400 & 400 & 800 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{17}$$

And before passing into the NN, the training data is normalized to the range [-1, 1] to improve the generalization, convergence speed, numerical stability, as well as mitigate the vanishing gradient [52]. Hence, they are given by:

$$\mathbf{X} = \begin{bmatrix} 0 & -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 1 \\ -1 & -1 & -1 & -1 & -1 \end{bmatrix}. \tag{18}$$

Step 2: Next, a network architecture of (3-5-3) is employed for this example, serving as a way to easily visualize the training process of our proposed approach. Therein, tanh f_1 and linear f_2 activation functions are employed for the neurons in the hidden and output layers, respectively. They are given by:

$$\begin{aligned} &f_1\left(z\right) = \tfrac{e^z - e^{-z}}{e^z + e^{-z}}, \\ &f_2\left(z\right) = z. \end{aligned} \tag{19}$$

In addition, all the parameters of the network are initialized with Xavier normal initializer [53], and updated by using Adam optimizer with default values (η = 0.001, β_1 = 0.9, β_2 = 0.999, ε = 1e-8, $\mathbf{m}_0 = \mathbf{0}$, $\mathbf{v}_0 = \mathbf{0}$). The initial connection weights between input layer to hidden layer and hidden layer to output layer are given by:

$$\mathbf{W}_1^{(0)} = \begin{bmatrix} 0.0511 & -0.1295 & -0.0775 \\ -0.6853 & -0.0660 & 0.4424 \\ -0.1305 & 0.3052 & -0.0049 \\ -0.7237 & -0.1019 & 0.4369 \\ -0.3408 & 0.3169 & 0.0961 \end{bmatrix}, \tag{20}$$

$$\mathbf{W}_2^{(0)} = \begin{bmatrix} 0.1644 & 0.2378 & 0.3711 & 0.0901 & -0.3488 \\ -0.2199 & -0.0597 & -0.0491 & -0.0030 & 0.2563 \\ 0.1435 & -0.3307 & -0.1343 & -0.1052 & 0.2722 \end{bmatrix}. \tag{21}$$

The initial biases for the hidden neurons and for the output neuron are expressed as follows:

$$\begin{aligned} \mathbf{b}_1^{(0)} &= \begin{bmatrix} 0.2098 & -0.2498 & 0.1810 & 0.3017 & 0.2671 \end{bmatrix}^T, \\ \mathbf{b}_2^{(0)} &= \begin{bmatrix} -0.1622 & -0.1871 & -0.0719 \end{bmatrix}^T. \end{aligned} \tag{22}$$

Step 3: Calculation of the predicted displacement using the FF with Eqs. (1) and (19). At joint 1

$$\begin{aligned} \mathbf{h}_0 &= \mathbf{X}\left(:,1\right) = \begin{bmatrix} 0 & -1 & -1 \end{bmatrix}^T, \\ \mathbf{h}_1 &= \begin{bmatrix} 0.3942 & -0.5554 & -0.1187 & -0.5626 & -0.1449 \end{bmatrix}^T, \\ \mathbf{h}_2 &= \begin{bmatrix} -0.1723 & -0.2703 & 0.2040 \end{bmatrix}^T = \begin{bmatrix} \hat{u}_1 & \hat{v}_1 & \hat{w}_1 \end{bmatrix}^T. \end{aligned} \tag{23}$$

And similarly, for other joints, substituting the $\mathbf{h}_0$ vector with the corresponding columns of $\mathbf{X}$ in Eq. (18) yields the predicted displacement field $\hat{\mathbf{d}}'$ of the cable net structure. After the predicted displacement vector with the fulfilled BCs $\hat{\mathbf{d}}$ is found by using Eq. (2). These vectors are given in Table 1.

Table 1
The predicted displacements without and with the fulfilled BCs.

	$\hat{d}'_i$	$\bar{b}_i$	$\hat{d}_i$
u_1	-0.1723	0	0
v_1	-0.2703	0	0
w_1	0.2040	0	0
u_2	-0.1751	0	0
v_2	-0.1319	0	0
w_2	0.0524	0	0
u_3	-0.1930	1	-0.1930
v_3	-0.1766	1	-0.1766
w_3	0.2539	1	0.2539
u_4	-0.1578	0	0
v_4	-0.2494	0	0
w_4	0.3072	0	0
u_5	-0.2158	0	0
v_5	-0.0876	0	0
w_5	0.2964	0	0

Step 4: Substitution of the values of $\hat{\mathbf{d}}$ into Eqs. (5)-(13) yields the loss function for the first iteration

$$\mathcal{L}\left(\boldsymbol{\theta}_0\right) \; = 339.022, \tag{24}$$

where $\boldsymbol{\theta}_0$ represents the parameters, including the weights and biases of the network at the previous iteration.

Step 5: Automatic calculation of the gradients of the loss function with respect to the parameters by using BP algorithm yields

$$\frac{\partial \mathcal{L}}{\partial \mathbf{W}_1^{(0)}} = \begin{bmatrix} 0.000 & 0.000 & -1.2910 \\ 0.000 & 0.000 & 8.0488 \\ 0.000 & 0.000 & 27.9764 \\ 0.000 & 0.000 & -6.2491 \\ 0.000 & 0.000 & -6.6927 \end{bmatrix}, \tag{25}$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{W}_1^{(0)}} = \begin{bmatrix} -26.4921 & 56.7886 & -17.4123 & 59.5228 & -16.0415 \\ -24.2324 & 51.9447 & -15.9271 & 54.4457 & -14.6732 \\ -4.0523 & 8.6866 & -2.6635 & 9.1048 & -2.4538 \end{bmatrix}, \tag{26}$$

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \mathbf{b}_2^{(0)}} &= \begin{bmatrix} 1.2910 & -8.0488 & -27.9764 & 6.2491 & 6.6927 \end{bmatrix}, \\ \frac{\partial \mathcal{L}}{\partial \mathbf{b}_2^{(0)}} &= \begin{bmatrix} -94.7423 & -86.6611 & -14.4922 \end{bmatrix}. \end{aligned} \tag{27}$$

Step 6: Updating of the parameters of the network. Substituting the values of η, β_1, β_2, ε, $\mathbf{m}_0 = \mathbf{0}$, $\mathbf{v}_0 = \mathbf{0}$ and the appropriate gradients into Eqs. (15)-(16) yields the updated parameters

$$\mathbf{W}_1^{(1)} = \begin{bmatrix} 0.0511 & -0.1295 & 0.0225 \\ -0.6853 & -0.0660 & 0.3424 \\ -0.1305 & 0.3052 & -0.1049 \\ -0.7237 & -0.1019 & 0.5369 \\ -0.3408 & 0.3169 & 0.1961 \end{bmatrix}, \tag{28}$$

$$\mathbf{W}_2^{(1)} = \begin{bmatrix} 0.2644 & 0.1378 & 0.4711 & -0.1901 & -0.2488 \\ -0.1199 & -0.1597 & 0.0509 & -0.1030 & 0.3563 \\ 0.2435 & -0.4307 & -0.0343 & -0.2052 & 0.3722 \end{bmatrix}, \tag{29}$$

$$\begin{aligned} \mathbf{b}_1^{(1)} &= \begin{bmatrix} 0.1098 & -0.1498 & 0.2810 & -0.4017 & 0.1671 \end{bmatrix}^T, \\ \mathbf{b}_2^{(1)} &= \begin{bmatrix} -0.0622 & -0.0871 & 0.0281 \end{bmatrix}^T. \end{aligned} \tag{30}$$

Step 7: Check the stopping criterion ($n_f < n_{max}$ or $iteration = epoch_{max}$). If the criterion is not satisfied, then return to step 3; otherwise, stop the training process.

Table 2
Summary of methods for the nonlinear analysis of cable structures.

Reference	Summary	Method
Lewis et al. [20, 54]	Dynamic relaxation algorithm was developed to minimize the residual force vector	FEA, Experiment
Toklu et al. [21]	Harmony Search algorithm was used to find the structural response of cable net structures with minimum TPE	TPE
Halvordson [51]	Newton Raphson algorithm was applied to minimize TPE	TPE
Kwan [55]	An analytical technique, which separated the complexity of anonlinear numerical algorithm from the underlying structuralprinciples, was developed for the nonlinear static behavior of cable networks	Analytical
Andreu et al. [13]	A new deformable catenary element was derived for the analysis of cable net structures	FEA
Thai and Kim [14]	A spatial two-node catenary cable element was developed for the nonlinear analysis of cable structures subjected to static and dynamic loadings	FEA
Thornton et al. [56]	An analytical procedure was presented for determining structural responses	Analytical
Nuhoglu [57]	A point-based iterative procedure was suggeted for the geometrical nonlinear analysis of cable systems	FEA
Abad et al. [1]	A novel formulation for spatial catenary cable element was proposed for nonlinear analysis of cable structures	FEA

3. Numerical experiments

In this section, a series of experiments involving seven numerical examples is conducted to showcase the simplicity, effectiveness, and validity of the presented procedure for nonlinear analysis of cable net structures. And many researchers have employed these benchmarks to evaluate different algorithms using experimental, FEA, TPE, and analytical-based approaches, as summarized in Table 2. Accordingly, the obtained results will be compared with existing works available in the literature. Note that the number of neurons in the input and output layers is set to 2 or 3 for 2- or 3-dimensional problems, respectively. In all considered examples, a shallow network with only one hidden layer is employed, consisting of no more than 50 hidden neurons. The Tanh activation function and Adam optimizer are recommended for these networks, while the number of hidden neurons is determined by using Grid search and trial-and-error methods.

For verification purposes, the obtained results will be compared with Stiffness Matrix (SM) and the Differential Evolution (DE) algorithms for minimizing the TPE. Note that for the SM method, the Euclidean norm of the gradient vector to be less than 0.01 is set as convergence criteria [20]. In addition, the initial displacement vector and weighting factor is initialized to a zero value for the first iteration [20]. Besides, the weighting factor p of 0.9 is selected to control the size of the iterative step [20]. Therein, the parameters for the DE algorithm are chosen as follows: mutant factor F=0.8, crossover factor Cr = 0.9, population size 40, maximum number of evaluations 3000, and the convergence criteria value is set to 10^{-6} [25, 28]. Due to the stochastic nature of the DE algorithm, the optimal solution for each problem is identified through 10 independent runs. To ensure an impartial comparison between the different algorithms, all numerical examples were executed on a personal computer using the Pytorch library with the Python language. Additional experiments were conducted on a laptop PC equipped with a 2.5 GHz Intel Core i5-7200U CPU and 8 GB of RAM.

3.1. Simple net

As mentioned in Section 2.3 above, the simple net is considered as the first numerical example for the nonlinear analysis of cable structure. A 3-layer network (3-10-3) is used to perform the training task with maximum epochs of 100, as shown in Table 3. To assess the impact of different activation functions and optimizers on the network's performance, a survey is conducted to determine the optimal combination based on the aforementioned architecture. The minimum TPE values are summarized in Table 4. Upon examining all the combinations, it is evident that Softmax and Adadelta perform poorly as the activation function and optimizer, respectively. Conversely, the Tanh activation function and Adam optimizer are identified as the optimal choices, yielding the smallest TPE of (271.80499 N.mm) when combined. Consequently, they are employed throughout the present study to perform the training process. Simultaneously, the grid search method is carried out to determine the optimal number of hidden layers as well as hidden neurons. In order to demonstrate the stability of the model, 20 independent runs are performed at each grid point. And Fig. 4 illustrates the minimum values of the loss function for each respective case. The solid dash lines show the mean value of the minimum loss function, while the transparent areas represent the range between the minimum and maximum values. A comparison of the statistical results, including minimum (Min), maximum (Max), mean, standard deviation (Std), and 95% Confidence Interval Upper (95% CIU) and Lower (95% CIU) bounds found by the network with 10 neurons in each hidden layer, are reported in Table 5. Accordingly, the obtained results demonstrate that the accuracy of the network does not always improve with an increase in the number of hidden neurons or hidden layers. From the data in this figure, it should be noted that when the number of hidden neurons exceeded 10, the learning task does not show significant improvement in the results. And it is evident that the network architecture (3-10-3) is the most appropriate for this specific example, as it yields the smallest loss function (271.80499 N.mm). More concretely, the range of confidence interval (95% CI = 271.805003 N.mm to 271.805006 N.mm) changes for narrow and close to the best (271.804993 N.mm), mean (271.805005 N.mm), and worst (271.805023 N.mm) with the small relative Std (3.52E-06 N.mm). Furthermore, it can be easily seen that the obtained standard deviation values were small or close to zero. In addition, there were not significant differences between the maximum, minimum, mean, and 95% CI values. This demonstrates the stability of the proposed model.

A comparison of the analysis results gained by the NN and previous studies, including displacements, tension forces, and minimum TPE, is illustrated in Table 6. It can be observed that the optimal TPE (271.8049926 N.mm) found by the PINN is very close to DE (271.8049723 N.mm), SM (271.8049727 N.mm) with the error less than 0.00001%, and smaller than the other studies (Lewis [20]: 271.8088 N.mm, Halvordson [51]: 271.8087 N.mm, and Toklu [21]: 271.8088 N.mm). Note that although the result obtained by the DE is the smallest TPE value, the number of function evaluation is larger than those of the proposed method. More specifically, the SM and PINN only require 3 and 81 iterations for the convergence performance, while the DE and HS [21] take 1840 and 5000, respectively. This can be easily explained by the fact that the SM and PINN are based on the gradient descent method to minimize the TPE, resulting in a significant reduction in computational costs. Meanwhile, the DE and HS [21] are gradient-free algorithms, so they often require a large number of evaluation functions to achieve the optimum solution [25]. In addition, according to the data in Table 20, the SM algorithm outperforms both PINN and DE in terms of computational cost. In particular, it only takes 0.1418 s seconds for 3 iterations, whereas the PINN and DE require 1.2723 s and 3.2781 s for 81 and 1840 iterations, respectively. Results show that although both algorithms are based on the gradient descent method, there is a significant difference in computational cost between the SM and PINN due to the sensitivity analysis. More concretely, the SM algorithm used the analytical formulation, while the network utilized the approximate sensitivity technique. In general, our proposed model yields a simple and easily applied method due to automatically calculate sensitivity, without requiring any inversion of operators, as well as structural analyses. The convergence history depicted in Fig. 5 provides a detailed performance view of the present approach. As observed, the convergence speed of the SM algorithm is the fastest, followed by PINN and DE. In the case of PINN, the learning curve rapidly decreases at the beginning, tends to stabilize around 50 epochs, and reaches the minimum TPE only after 81 epochs.

Table 3
Hyperparamters of NN model for the cable structures tested in this study.

Test problems	NN architecture	Activation function	Optimizer	Epoch$_{max}$
Simple net	(3-10-3)	Tanh	Adam	100
Flat cable net 2×1	(3-10-3)	Tanh	Adam	500
Flat cable net 2×2	(3-10-3)	Tanh	Adam	500
Hyperbolic paraboloid net	(3-15-3)	Tanh	Adam	500
Spatial cable network	(3-20-3)	Tanh	Adam	500
Dual cable	(3-30-3)	Tanh	Adam	500
Saddle net	(3-30-3)	Tanh	Adam	500

Table 4
Minimum total potential energy (N.mm) with various activation functions and optimizers.

Activationfunction	Optimizers				
	Adam	Adagrad	Adamax	SGD	Adadelta
Softmax	272.80164	297.84952	280.89359	300.33197	324.97620
Tanh	**271.80499**	271.80704	271.80505	295.22107	324.90076
Softplus	271.80658	273.68015	271.93808	299.14709	410.56708
Sigmoid	271.80505	272.17883	273.57019	299.27200	365.66006
ReLU	271.80527	299.22198	279.99429	301.50919	325.41324
LeakyReLU	272.26123	279.71744	271.88046	298.14218	325.06259

Bold used to emphasize the best combination with respect to the smallest TPE

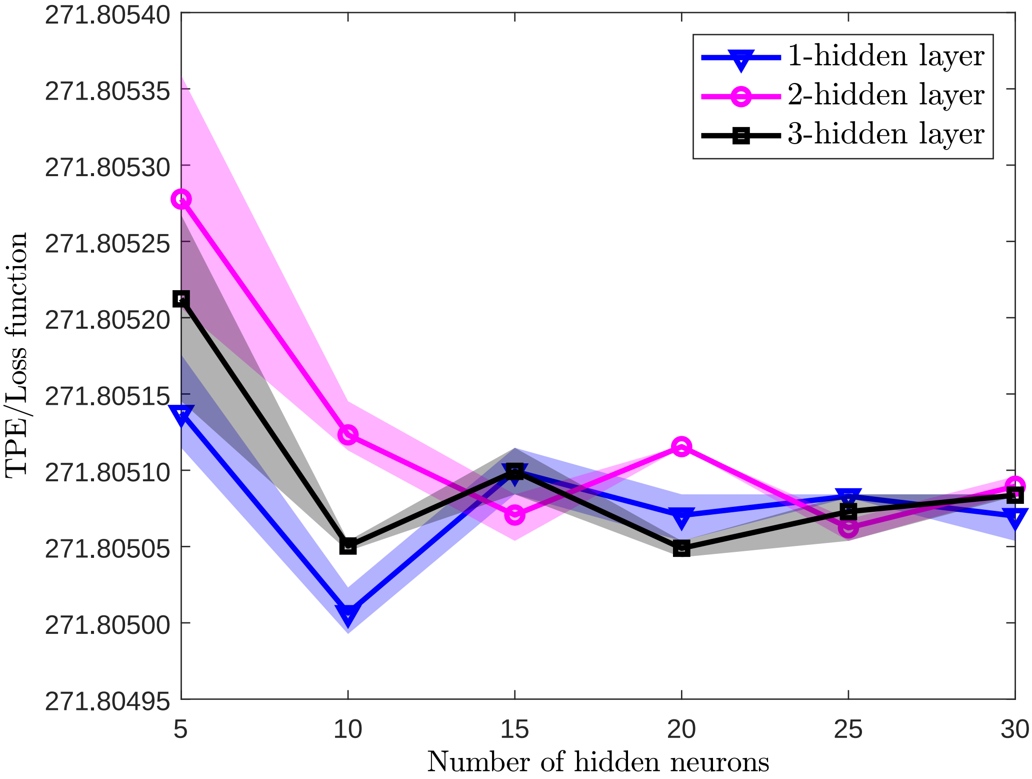

Fig. 4. Total potential energy of simple net with varying hidden layers and neurons.

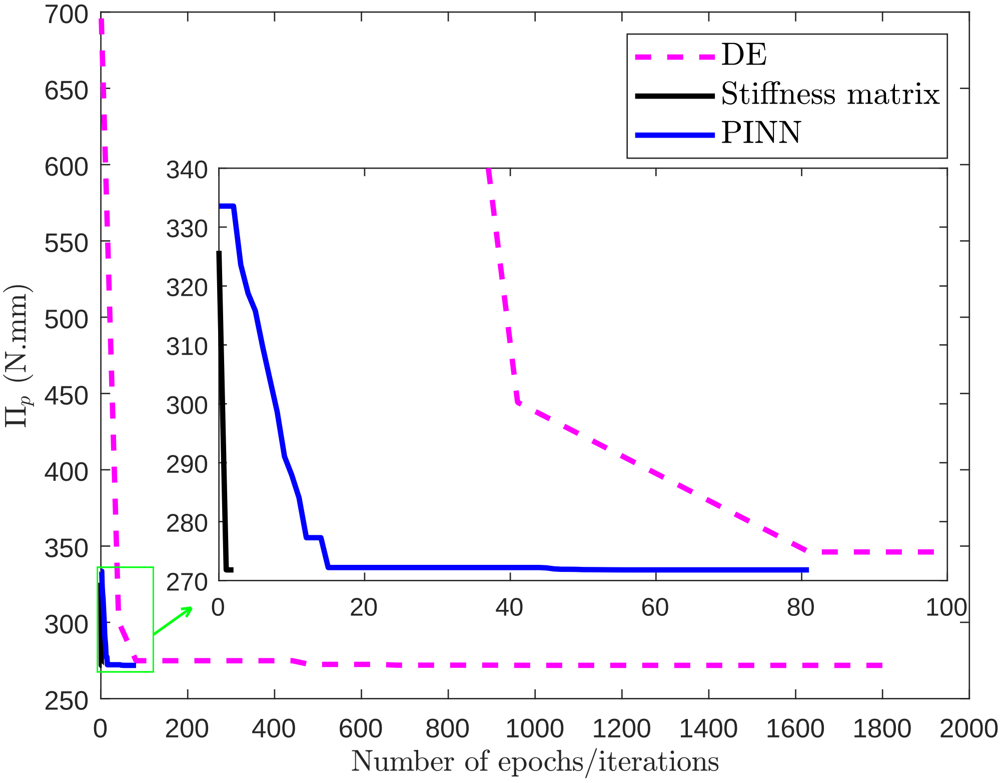

Fig. 5. The convergence history of the loss function for the simple net.

Table 5

Statistics of displacement (mm), tension (N) and total potential energy (N.mm) results for simple net with 10 neurons in each hidden layer.

Metric	Network architecture								
	(3-10-3)			(3-10-10-3)			(3-10-10-10-3)		
	w_3	$f_{1,2,3,4}$	Π_p	w_3	$f_{1,2,3,4}$	Π_p	w_3	$f_{1,2,3,4}$	Π_p
Min	6.9755	214.8952	271.804993	6.9700	214.8718	271.805113	6.9737	214.8877	271.805047
Max	6.9849	214.9356	271.805023	6.9894	214.9546	271.805145	6.9879	214.9482	271.805054
Mean	6.9799	214.9142	271.805005	6.9775	214.9040	271.805123	6.9810	214.9191	271.805050
Std	0.0007	0.0028	3.52E-06	0.0018	0.0077	3.40E-06	0.0012	0.0050	7.90E-07
95% CIL	6.9796	214.9130	271.805003	6.9767	214.9006	271.805122	6.9805	214.9169	271.805050
95% CIU	6.9802	214.9154	271.805006	6.9783	214.9074	271.805125	6.9816	214.9213	271.805051

Table 6
Comparison of displacement (mm), tension (N) and total potential energy (N.mm) results for simple net.

	Lewis	Halvordson	Toklu	Present		
	[20]	[51]	[21]	DE	SM	PINN
u_3	0	0	0	0	0	0.0000
v_3	0	0	0	0	0	0.0000
w_3	6.9700	6.9800	6.9700	6.9804	6.9795	6.9807
$f_{1,2,3,4}$	214.9000	214.9146	214.9000	214.9110	214.9127	214.9178
Π_p	271.8088	271.8087	271.8088	**271.8050**	271.8050	271.8050

Bold used to emphasize the best minimum TPE

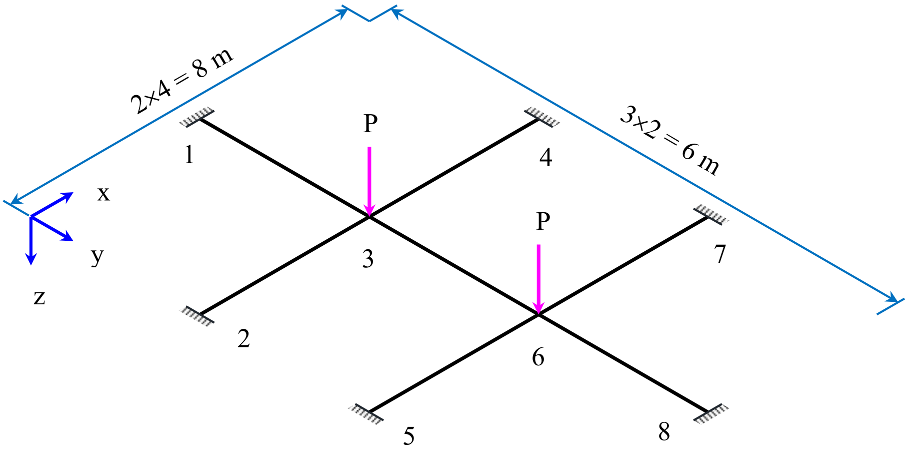

Fig. 6. Flat cable net 2 × 1.

3.2. *Flat cable net 2×1*

Next, a flat system, as shown in Fig. 6, is examined as the second example for the nonlinear analysis of cable net structures. In this case, all cable members are set the same cross-sectional areas, Young's modulus, and pretension force of 2 mm^2, 110 kN/mm^2, and 500 N, respectively. The system is subjected to a vertical load P = 200 N at nodes 3 and 6. In addition, the network configuration reported in Table 3 is used to train the network. Similar to the first example, the obtained analysis results from the proposed work are summarized in Table 7 for comparative evaluation with other algorithms. The results indicate that the PINN (-37471.21 N.mm) outperforms other approaches (DE: -37471.21 N.mm; SM: -37471.00 N.mm; Lewis [20], Halvordson [51], and Toklu [21]: -37448.49 N.mm), having the smallest energy value of TPE. And this further strengthens the evidence of the efficacy of the proposed model. Fig. 7 shows the potential energy convergence history obtained by DE, SM, and PINN. It can be easily seen that the presented model converges very quickly in the early epoch of training, requiring only 417 epochs. In contrast, the DE used 4840 iterations to achieve the solution. As an initial example, it was observed from the data in Table 20 that our framework (417) requires more iterations than that of the SM (12), so its computation cost (3.9432 s) increases more than 6 times compared with the SE (0.6337 s).

Table 7
Comparison of displacement (mm), tension (N) and total potential energy (N.mm) results for flat cable net 2 × 1.

	Lewis	Halvordson	Toklu			Present
	[20]	[51]	[21]	DE	SM	PINN
u_3	0.00	0.00	0.00	0	0.00	0.00
v_3	-3.30	-3.30	-3.30	-3.30	-3.29	-3.30
w_3	199.70	199.70	199.70	199.74	199.43	199.75
u_6	0.00	0.00	0.00	0.00	0.00	0.00
v_6	3.30	3.30	3.30	3.31	3.29	3.30
w_6	199.70	199.70	199.70	199.78	199.43	199.75
$f_{1,7}$	1232.80	1232.80	1232.80	1233.28	1230.99	1233.25
f_4	1225.40	1225.40	1225.40	1226.52	1223.68	1226.14
$f_{2,3,5,6}$	774.00	774.00	774.00	774.22	773.34	774.22
Π_p	-37448.49	-37448.49	-37448.49	-37471.20	-37471.00	**-37471.21**

Bold used to emphasize the best minimum TPE

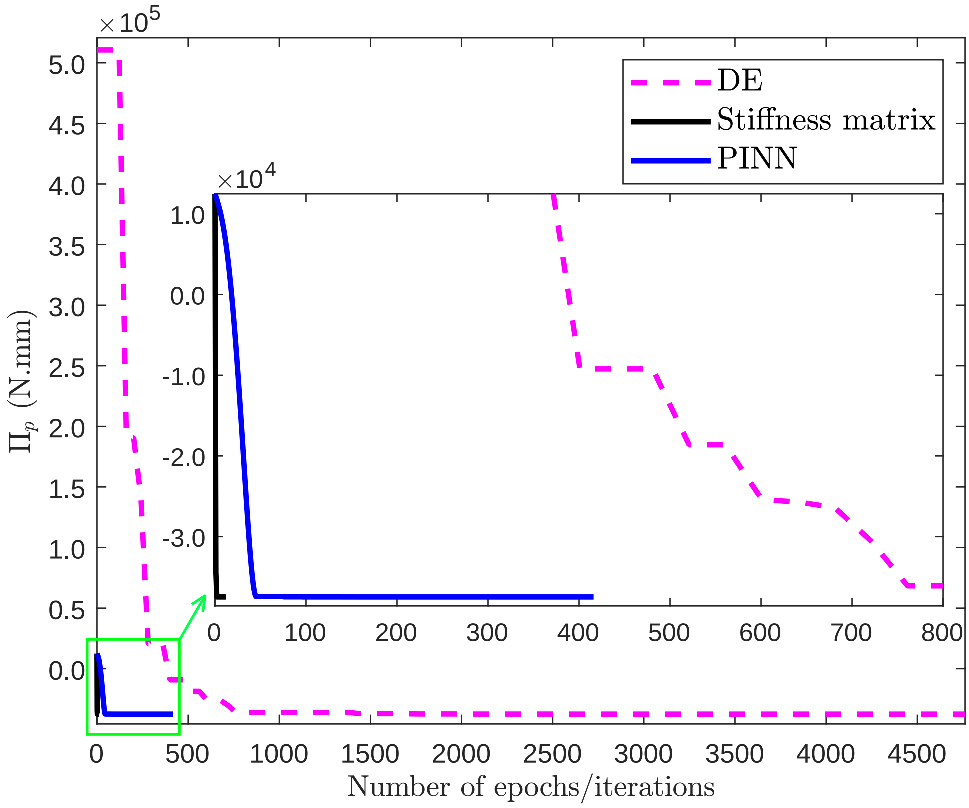

Fig. 7. The loss convergence history of the flat cable net.

3.3. Flat cable net 2×2

The next analysis problem deals with the flat cable net 2×2 shown in Fig. 8. This structure consists of 12 cables, with four joints allowing free movement and other constrained joints. All members have the same linear elastic modulus of 124.8 kN/mm^2 and cross-sectional area of 0.785 mm^2. Before applying the external load, all cable segments are pretensioned to carry a force of 200 N. A concentrated load of P = 15 N is then applied in the z-direction at joints 4, 5, and 8. This benchmark has been widely examined by researchers, including Lewis [20], Kwan [55], Halvordson [51], and Toklu [21], to evaluate the effective implementation of algorithms. For this specific application, the NN consists of one hidden layer with 10 hidden neurons, and the training phase is performed over 500 epochs.

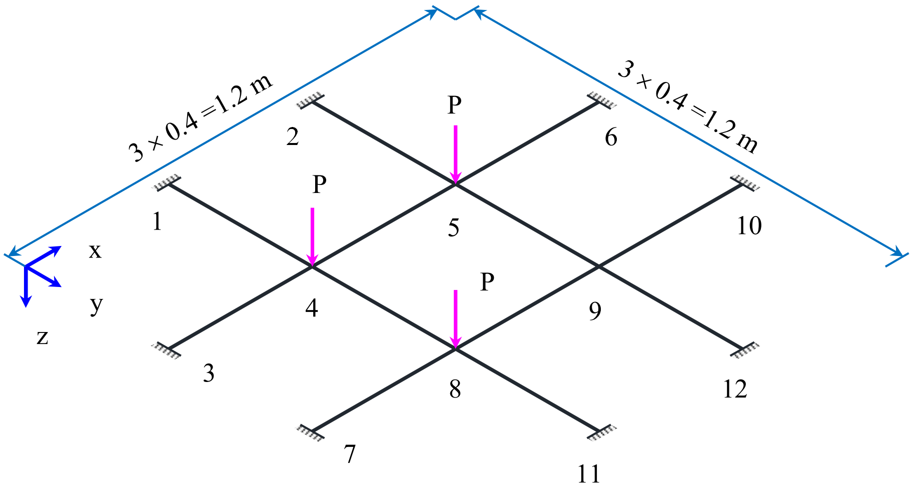

Fig. 8. Flat cable net 2 × 2.

Likewise, Table 8 provides a comparison of the solution obtained by the network and other methods. It is noticed that in this example, the PINN achieves the best minimum TPE (704.8126 N.mm). Compared to the computational cost, the PINN (303) requires more evaluations than SM (34), but our model takes 3.2566 s compared to SM's 1.972 s, as shown in Table 20. Despite utilizing the principle of minimizing the TPE function, the HS [21] and DE take 10000 and 11680 evaluations to achieve the same objective. Hence, our approach outperforms well-known existing algorithms in term of the solution quality. These experimental results have revealed that our approach not only demonstrates simplicity in performance but also ensures the quality of the solution.

3.4. *Hyperbolic paraboloid net*

In the fourth example, the analysis problem of the hyperbolic paraboloid cable net system is investigated. This structure was previously experimentally tested by Lewis [54] and addressed by researchers, such as Kwan [55], Sufian [58], Thai [14], and Andreu [13]. The geometry, dimensions, and boundary condition of the system are depicted in Fig. 9 and Table 10. The modulus of linear elasticity is 128.3 kN/mm^2 and the cross-sectional area is 0.785 mm^2 for all cable members. The initial structure geometry is established by applying the same pretension force of 200 N to all cables. Additionally, a concentrated downward load of 15.7 N is applied at several free joints, as shown in Fig. 9. A grid search similar to the first numerical example is used to estimate the network configuration.

Table 8
Comparison of displacement (mm), tension (N) and total potential energy (N.mm) results for flat cable net 2 × 2.

	Lewis	Kwan	Halvordson	Toklu			Present
	[20]	[55]	[51]	[21]	DE	SM	PINN
u_4	-0.07	-0.08	-0.07	-0.07	-0.071	-0.071	-0.071
v_4	-0.07	-0.08	-0.07	-0.07	-0.071	-0.071	-0.071
w_4	12.19	12.17	12.20	12.17	12.172	12.169	12.172
u_5	0.04	0.04	0.04	0.04	0.041	0.042	0.042
v_5	-0.08	-0.08	-0.08	-0.08	-0.078	-0.078	-0.078
w_5	11.20	11.18	11.20	11.18	11.184	11.180	11.183
u_8	-0.10	-0.08	-0.08	-0.08	-0.077	-0.078	-0.078
v_8	0.00	0.05	0.04	0.04	0.041	0.042	0.042
w_8	11.20	11.18	11.20	11.18	11.186	11.180	11.183
u_9	0.00	-0.04	-0.04	-0.04	-0.038	-0.039	-0.039
v_9	0.00	-0.04	-0.04	-0.04	-0.039	-0.039	-0.039
w_9	5.60	5.59	5.59	5.59	5.596	5.591	5.592
f_1	228.10	225.75	228.42	228.20	228.087	228.010	228.012
f_2	219.30	218.67	218.81	218.67	219.222	219.231	219.229
f_4	228.00	229.69	227.25	227.24	227.737	227.869	227.882
f_5	228.10	228.47	228.60	228.47	228.210	228.024	228.044
f_7	219.20	219.36	219.43	219.36	219.074	219.119	219.132
f_{10}	219.10	219.36	219.36	219.36	218.960	219.054	219.066
Π_p	706.9226	704.8925	704.8477	704.8458	704.8128	704.8170	**704.8126**

Bold used to emphasize the best minimum TPE

As the previously presented simple net, the minimum values of the TPE for each grid point are illustrated in Fig. 10. Once more, the increase in the number of hidden layers or hidden neurons did not always improve the accuracy of the network. From this graph, the network with 15 and 10 neurons in each hidden layer are found as the best architectures for single, two, and three hidden layers, respectively. And the comparison of the statistical displacements and TPE found by three optimal network configuration are tabulated in Table 9. Clearly, the network (3-15-3) outperforms all other models in terms of the accuracy and significantly reduces the training parameters. More concretely, the 95%CI (1074.1760 - 1074.1807 N.mm) values of the TPE found by this architecture are quite close the minimum (1074.1707 N.mm) and maximum (1074.2742 N.mm) values with very small Std (5.31E-03 N.mm). Hence, this network configuration is utilized to determine the structural responses. Furthermore, compared with the previously flat cable nets, it can be observed that when the complex cable net structure, the network requires more hidden neurons to achieve the accuracy.

As shown in Tables 10 and 20, the obtained displacements reveal that our approach closely matches the experimental result found by Lewis [54]. It is evident that the network can accurately calculate the behavior of cable with the lowest error values (min=0.11%; max=1.49%; mean=0.76%), and then Toklu [21], SM, Sufian [58], Kwan [55], Thai [14], and Andreu [13], respectively. And this approach demonstrates considerable efficiency compared to traditional numerical methods when increasing the complexity of the cable net structure. Specifically, our framework takes only 6.1750 s for 392 iterations, while the SM requires 8.4369 s for 200 iterations to achieve as accurate solutions as possible.

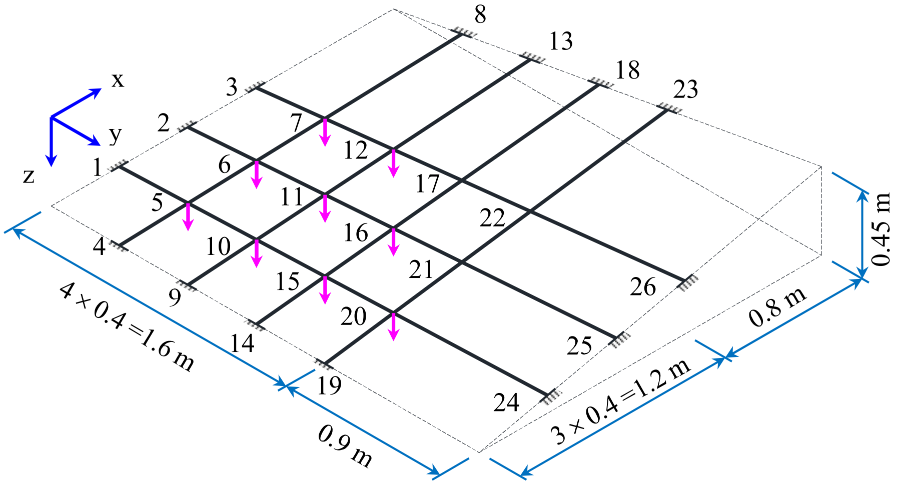

Fig. 9. Hyperbolic paraboloid net.

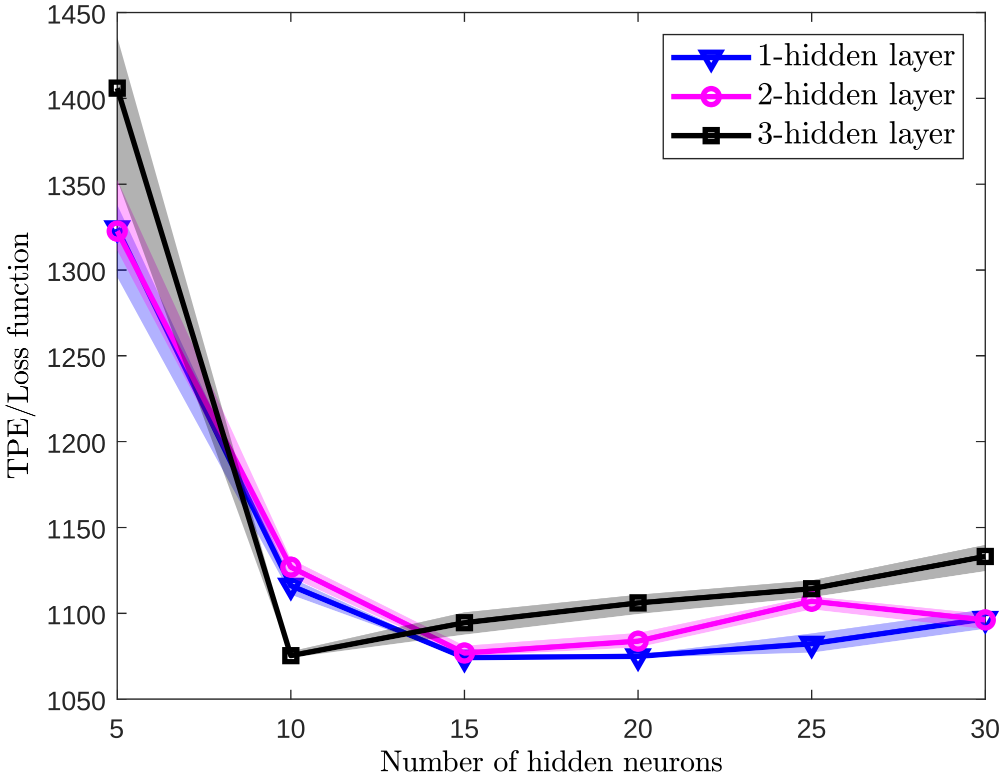

Fig. 10. Total potential energy of hyperbolic paraboloid net with varying hidden layers and neurons.

Table 9
Statistics of displacement (mm) and total potential energy (N.mm) results for the best network configuration.

Metric	Network architecture								
	(3-15-3)			(3-15-15-3)			(3-10-10-10-3)		
	w_{11}	w_{16}	Π_p	w_{11}	w_{16}	Π_p	w_{11}	w_{16}	Π_p
Min	33.6010	30.6222	1074.1707	31.9714	29.6982	1075.0552	32.4153	30.2127	1074.1785
Max	33.6517	30.6440	1074.2742	33.9121	30.8819	1081.1033	33.8578	30.7709	1077.8704
Mean	33.6340	30.6408	1074.1783	33.4885	30.4677	1076.8674	33.5806	30.5759	1075.3313
Std	0.0025	0.0011	5.31E-03	0.0959	0.0589	3.13E-01	0.0675	0.0303	2.71E-01
95% CIL	33.6329	30.6403	1074.1760	33.4465	30.4419	1076.7302	33.5509	30.5626	1075.2124
95% CIU	33.6351	30.6413	1074.1807	33.5305	30.4936	1077.0046	33.6102	30.5891	1075.4503

Table 10
Comparison of vertical displacements (mm) for hyperbolic paraboloid net system.

Node	Lewis [54]	Toklu	Kwan	Sufian	Thai	Andreu	Present	
	(Experiment)	[21]	[55]	[58]	[14]	[13]	SM	PINN
5	19.50	19.48 (0.10)[a]	19.52 (0.10)	19.3 (1.03)	19.56 (0.31)	19.51 (0.05)	19.35 (0.77)	19.36 (0.74)
6	25.30	25.59 (1.15)	25.35 (0.20)	25.5 (0.79)	25.70 (1.58)	25.65 (1.38)	25.35 (0.19)	25.36 (0.23)
7	22.80	23.17 (1.62)	23.31 (2.24)	23.1 (1.32)	23.37 (2.50)	23.37 (2.50)	23.04 (1.05)	23.05 (1.09)
10	25.40	25.75 (1.38)	25.86 (1.81)	25.8 (1.57)	25.91 (2.01)	25.87 (1.85)	25.60 (0.79)	25.61 (0.84)
11	33.60	33.86 (0.77)	34.05 (1.34)	34.0 (1.119)	34.16 (1.67)	34.14 (1.61)	33.62 (0.06)	33.64 (0.11)
12	28.80	29.27 (1.63)	29.49 (2.40)	29.4 (2.08)	29.60 (2.78)	29.65 (2.95)	29.08 (0.98)	29.09 (1.02)
15	25.20	25.65 (1.79)	25.79 (2.34)	25.7 (1.98)	25.86 (2.62)	25.86 (2.62)	25.36 (0.62)	25.37 (0.66)
16	30.60	30.96 (1.18)	31.31 (2.32)	31.2 (1.96)	31.43 (2.71))	31.47 (2.84)	30.63 (0.09)	30.64 (0.14)
17	21.00	21.03 (0.14)	21.42 (2.00)	21.1 (0.48)	21.56 (2.67)	21.57 (2.71)	20.83 (0.82)	20.84 (0.78)
20	21.00	21.33 (1.57)	21.48 (2.29)	21.1 (0.48)	21.57 (2.71)	21.62 (2.95)	20.57 (2.06)	20.77 (1.08)
21	19.80	19.67 (0.66)	20.00 (101)	19.9 (0.51)	20.14 (1.72)	20.15 (1.77)	19.79 (0.04)	19.70 (0.51)
22	14.20	14.04 (1.13)	14.40 (1.41)	14.3 (0.47)	14.55 (2.46)	14.55 (2.46)	14.41 (1.45)	14.41 (1.49)
Error (%)	Min	0.10	0.10	0.48	0.31	0.05	**0.04**	0.11
	Max	1.79	2.40	2.08	2.78	2.95	2.06	**1.49**
	Mean	1.16	1.91	1.11	2.48	2.48	0.75	**0.76**
Π_p	-	-	-	-	-	-	1074.21	**1074.17**

[a] Values in parentheses are the percentage error with respect to experiment displacements
Bold used to emphasize the smallest error

3.5. Spatial cable network

Next, the geometry nonlinear analysis of the spatial cable system is regarded as the fifth problem, which has been previously examined by Lewis [20], Thai [14], and Toklu [21]. The geometry and z-coordinates of joints are illustrated and listed in Fig. 11 and Table 11, respectively. This structure exhibits mirror symmetry along both central axes. To reach the initial geometry of the system, the pretension forces of 90 kN and 30 kN are applied to the cables in the x- and y-directions, respectively. The cross-sectional areas of the members in the x- and y-directions correspond to 350 mm^2 and 120 mm^2, and the Young's modulus of cables is 160 kN/mm^2. A vertically concentrated load of 6.8 kN is applied to all free joints of the structure. In addition, the parameters of the network, as listed in Table 3, provide the finest performance for this structure.

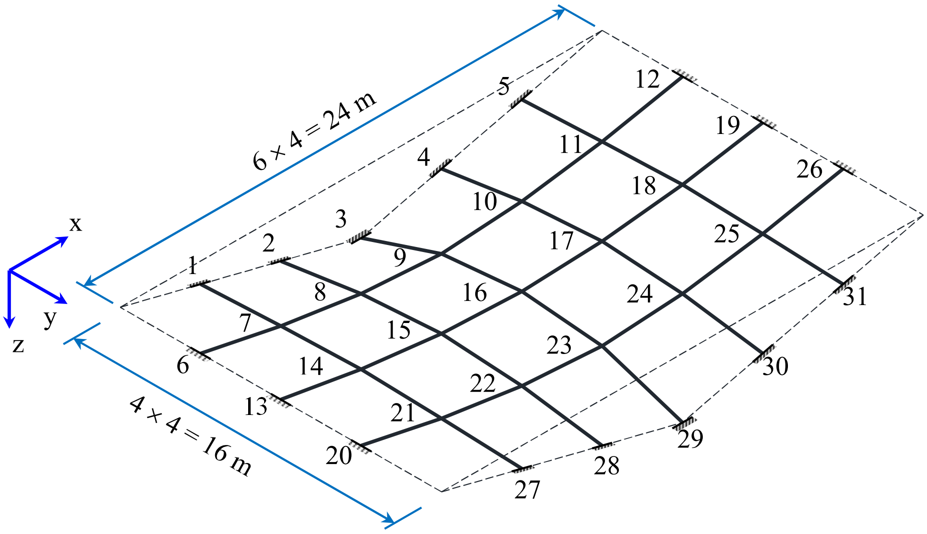

Fig. 11. Spatial cable net system.

Tables 12 and 20 present a comparison between the results obtained by the current method and alternative methods. It is evident from this data that the network achieves a smaller minimum TPE value (6504280 N.mm) compared to other algorithms (SM: 6504282 N.mm; Lewis [20]: 6505782 N.mm; Thai [14]: 6505527 N.mm; Toklu [21]: 6505019 N.mm). Thus, our model enhances the accuracy of the structural responses. In this example, the SM algorithm (6.4299 s) has a slightly lower computational cost than the proposed scheme (8.1858 s). However, it should be noted that the initial displacement vector of the SM algorithm is set to zero, while the parameters of the network are initialized randomly. This is partially due to the increased computational cost when the initial point is far from the solution.

Table 11
Z-coordinate (mm) of nodes for spatial net.

Node	1	2	3	6	7	8	9	13	14	15	16
z-coord(mm)	1000	2000	3000	0	820	1410	1677	0	687	1148	1318

Table 12
Comparison of vertical displacements (mm) and total potential energy (N.mm) results for spatial net.

	Lewis	Thai	Toklu	Present	
	[20]	[14]	[21]	SM	PINN
u_7	-5.14	-5.03	-5.03	-5.030	-5.031
v_7	0.42	0.41	0.40	0.398	0.397
w_7	30.41	29.86	29.46	29.451	29.463
u_8	-2.26	-2.23	-2.22	-2.225	-2.226
v_8	0.47	0.46	0.39	0.393	0.392
w_8	17.70	17.29	17.08	17.098	17.117
u_9	0	0	0	0.000	0.001
v_9	-2.27	-2.31	-3.12	-2.356	-2.355
w_9	-3.62	-3.61	-3.19	-3.199	-3.186
u_{14}	-4.98	-4.92	-4.92	-4.921	-4.928
v_{14}	0	0	0	0.000	0.000
w_{14}	43.49	42.85	42.84	42.828	42.894
u_{15}	-2.55	-2.55	-2.55	-2.548	-2.558
v_{15}	0	0	0	0.000	0.000
w_{15}	44.47	44.26	44.27	44.242	44.366
u_{16}	0	0	0	0.000	0.000
v_{16}	0	0	0	0.000	0.000
w_{16}	41.65	42.08	42.08	42.055	42.091
Π_p	6505782	6505527	6505019	6504282	6504280

Bold used to emphasize the best minimum TPE

3.6. Dual cable

A dual cable structure, as shown in Fig. 12 and Table 13, is studied for the next nonlinear analysis. Prior to the application of external loads, the initial shape is designed with pretension forces, which are listed in Table 13 for the parabolic-shaped tie-down and load cables. All cables are made of a material having an elasticity modulus of 165.54 GN/m^2. The hanger cables have a cross-sectional area of 64.5 mm^2, while the other cables have a cross-sectional area of 645 mm^2. In this example, the system is subjected to two distinct loading conditions as outlined below: 1) concentrated vertical forces in the negative z-direction, corresponding triangularly to the distributed loading, are applied to the joints of the tie-down cable. The values undergo linear variation, starting from 1.335 kN at joint 1 and increasing to 12.015 kN at joint 17; 2) a concentrated force of 50 kN is applied to joint 9 along the y-direction. The network configuration, as mentioned in Table 3, is employed to perform the training task.

The analysis results, including displacements, axial forces, and computational time, obtained by the present approach and other works are reported in Tables 14, 15, 16, and 20. As expected, the structural responses obtained are in agreement with those reported by Thornton [56], Nuhoglu [57], Toklu [21], and the SM algorithm. And it is obvious that the minimum TPE obtained by our model is the smallest for both loading cases. Furthermore, the computational cost of the present method is still smaller than that of the SM. Consequently, the suggested approach generally outperforms existing ones in terms of accuracy and efficiency.

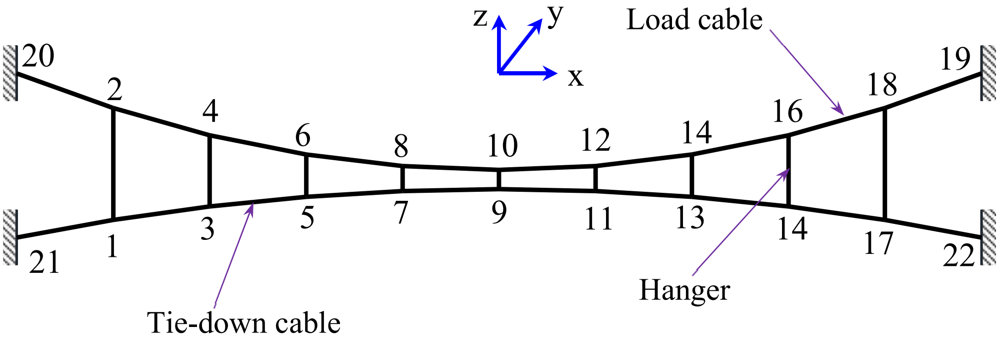

Fig. 12. Dual cable net system.

Table 13
Joint coordinates and pretension forces for dual cable net.

Coordinates (mm)						Pretension forces (kN)					
Node	x	z	Node	x	z	Member	Forces	Member	Force	Member	Force
1	-12194	-4633	7	-3048	-3719	20-2	23.65	21-1	45.22	1-2	1.78
2	-12194	-1097	8	-3048	-2926	2-4	23.1	1-3	44.94	3-4	1.78
3	-9144	-4206	9	0	-3658	4-6	22.69	3-5	44.72	5-6	1.78
4	-9144	-1951	10	0	-3048	6-8	22.41	5-7	44.58	7-8	1.78
5	-6096	-3901	20	-15240	0	8-10	22.27	7-9	44.51	9-10	1.78
6	-6096	-2560	21	-15240	-5182	Note: System is symmetric about to z axis.					

Table 14
Comparison of displacement (mm) and total potential energy (N.mm) results for dual cable net (case 1).

Node	Thornton [56]		Nuhoglu [57]		Toklu [21]		SM		PINN	
	u	w	u	w	u	w	u	w	u	w
1	-21.30	110.60	-21.90	112.80	-21.62	111.77	-21.94	113.39	-21.94	113.41
2	39.30	111.30	40.20	114.30	39.70	112.54	40.24	114.18	40.25	114.19
3	-25.90	146.60	-26.80	151.20	-26.36	148.49	-26.82	151.27	-26.83	151.29
4	50.30	146.00	51.80	150.80	50.88	148.18	51.74	150.94	51.75	150.95
5	-23.20	125.30	-23.70	130.10	-23.66	126.74	-24.10	129.69	-24.10	129.70
6	46.90	123.40	48.40	128.60	47.64	125.56	48.54	128.45	48.55	128.45
7	-19.80	66.80	-20.40	69.50	-20.12	67.55	-20.49	69.66	-20.49	69.66
8	40.50	63.10	42.00	67.60	41.18	66.67	41.97	67.69	41.98	67.69
9	-20.40	-8.80	-19.20	-9.70	-18.99	-10.39	-19.33	-9.70	-19.34	-9.71
10	37.80	-13.70	39.00	-12.20	38.35	-12.66	39.06	-12.07	39.06	-12.08
11	-20.40	-84.40	-21.30	-88.40	-20.97	-87.42	-21.34	-88.20	-21.34	-88.23
12	41.10	-87.80	42.30	-90.20	41.77	-89.29	42.51	-90.15	42.52	-90.18
13	-23.80	-140.80	-24.70	-146.00	-24.37	-144.30	-24.80	-146.25	-24.81	-146.29
14	48.50	-142.30	49.90	-146.90	49.30	-145.00	50.18	-147.19	50.19	-147.22
15	-25.30	-158.20	-25.90	-162.10	-25.53	-160.28	-25.95	-162.75	-25.96	-162.79
16	53.00	-157.90	53.00	-162.10	53.59	-159.53	54.54	-162.05	54.55	-162.08
17	-19.20	-118.00	-19.80	-120.70	-19.52	-118.81	-19.83	-120.70	-19.83	-120.73
18	42.30	-115.20	43.30	-117.60	42.50	-115.68	43.27	-117.61	43.28	-117.63
Π_p	-1994844		-2021280		-2170278		-2171202		**-2171203**	

Bold used to emphasize the best minimum TPE

Table 15
Comparison of displacement (mm) and total potential energy (N.mm) results for dual cable net (case 2).

Node	Nuhoglu [57]			Toklu [21]			SM			PINN		
	u	v	w	u	v	w	u	v	w	u	v	w
1	-4.90	150.60	46.30	-4.95	150.45	46.38	-4.92	150.45	46.16	-4.94	150.62	46.30
2	14.00	191.40	49.70	13.94	191.07	49.67	13.85	191.21	49.43	13.88	191.42	49.58
3	-7.00	299.90	85.60	-7.05	299.84	85.83	-7.01	299.76	85.47	-7.05	300.13	85.81
4	20.10	385.00	86.30	20.17	384.61	86.36	20.04	384.75	85.99	20.11	385.10	86.32
5	-6.40	444.40	116.40	-6.45	444.35	116.25	-6.43	444.25	115.93	-6.45	444.84	116.17
6	20.40	586.70	109.70	20.31	586.04	109.78	20.18	586.23	109.43	20.21	586.88	109.66
7	-3.90	576.10	144.80	-3.99	575.84	144.77	-3.98	575.61	144.59	-4.00	576.33	144.88
8	14.00	810.50	109.40	14.18	810.28	109.74	14.09	810.17	109.53	14.11	811.11	109.75
9	0.00	677.00	194.80	0.00	676.74	195.26	0.00	676.37	195.10	0.00	677.23	195.45
10	0.00	1086.60	38.70	0.00	1087.10	38.70	0.00	1086.37	38.84	0.00	1087.42	39.02
Π_p	-39504210			-39516190			-39520128			**-39520199**		

Bold used to emphasize the best minimum TPE

Table 16
Comparison of axial forces (kN) results for dual cable net.

Member	Case 1			Member	Case 2		
	Nuhoglu [57]	SM	PINN		Nuhoglu [57]	SM	PINN
10-12	115.30	115.27	115.74	20-2	177.55	177.43	177.0652
12-14	116.63	116.66	116.31	2-4	173.98	173.86	173.83
14-16	118.90	118.87	118.56	4-6	171.34	171.23	171.26
16-18	122.20	122.11	122.43	6-8	169.82	169.70	170.07
18-19	126.65	126.56	125.59	8-10	169.42	169.30	169.40
21-1	68.53	68.53	68.18	21-1	294.79	298.36	298.93
1-3	67.64	67.62	67.93	1-3	297.03	296.15	296.81
3-5	66.88	66.90	66.77	3-5	295.22	294.35	294.97
5-7	66.26	66.26	66.37	5-7	293.85	292.99	293.54
7-9	65.55	65.54	65.35	7-9	293.00	292.15	292.88
9-10	9.26	9.28	9.78	1-2	12.50	12.53	12.54
11-12	10.10	10.10	10.62	3-4	12.61	12.61	12.57
13-14	10.99	10.99	11.58	5-6	12.09	12.09	12.02
15-16	11.88	11.91	12.01	7-8	10.00	9.97	10.05
17-18	12.82	12.83	12.92	9-10	28.85	28.73	28.75

3.7. Saddle net

A saddle cable net, as depicted in Fig. 13, is considered as the last nonlinear analysis problem. Table 17 displays the z-coordinates for a quarter of the cable net, which exhibits mirror symmetry across both centerlines. A pretension force of 60 kN is applied to all cable segments to achieve the initial configuration. The elasticity modulus and cross-sectional area for all cables are taken as 147 kN/mm^2 and 306 mm^2, respectively. The cable net is subjected to forces in the x- and z-directions, with magnitudes of 1 kN at joints 11-15, 22-25, 33-37, 44-48, 55-59, 66-70, and 77-81. The network, with parameters mentioned in Table 3, is established for the training operation. Tables 18, 19, and 20 provide a comparison of solutions found by the proposed framework and previous studies. It is worthwhile to note that the SM algorithm can not converge. As pointed out by Lewis [20], this is due to numerical ill-conditioning during the inversion of the stiffness matrix. Therefore, the optimization process stability depends on a well-behaved tangent stiffness matrix. Meanwhile, our approach does not require calculating the stiffness matrix, as well as matrix inversion operations. Hence, it avoids the numerical ill-conditioning problems. Obviously, the data reveals a fairly good agreement between our model and the results reported by other authors.

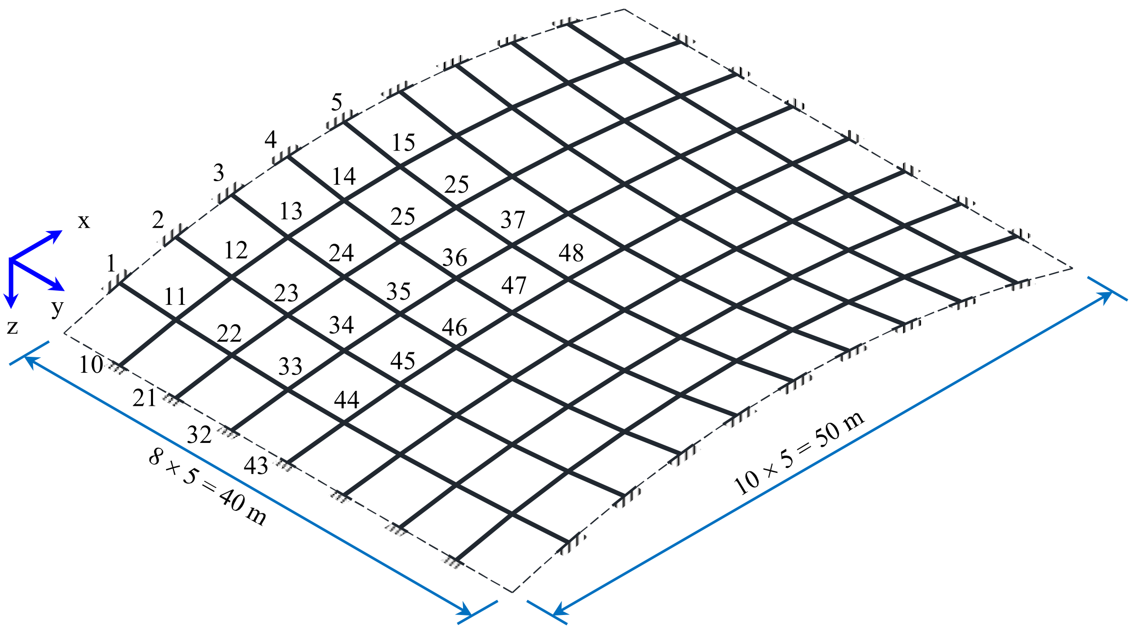

Fig. 13. Saddle net.

Table 17
Z-coordinates (mm) of nodes for saddle net.

Node	z	Node	z	Node	z
1	-1368	22	-792	44	-600
2	-2432	23	-1408	45	-1067
3	-3192	24	-1848	46	-1400
4	-3648	25	-2118	47	-1600
5	-3800	26	-2200	48	-1667
11	-1032	33	-648		
12	-1835	34	-1152		
13	-2408	35	-1512		
14	-2752	36	-1728		
15	-2867	37	-1800		

Table 18

Comparison of displacement (mm) and total potential energy (N.mm) results of saddle net.

Node	Thai [14]			Kwan [55]			Abad [1]			PINN		
	u	v	w	u	v	w	u	v	w	u	v	w
11	15.55	-4.46	81.70	15.55	-4.46	81.66	15.57	-4.46	81.79	15.55	-4.46	81.70
12	11.50	-5.55	61.22	11.50	-5.54	61.18	11.51	-5.55	61.28	11.51	-5.55	61.24
13	7.38	-4.20	33.31	7.38	-4.19	33.28	7.39	-4.20	33.36	7.39	-4.19	33.31
14	5.34	-3.11	17.88	5.34	-3.11	17.87	5.34	-3.11	17.92	5.39	-3.25	18.61
15	4.11	-2.80	11.16	4.10	-2.80	11.15	4.10	-2.80	11.21	4.13	-2.66	10.45
22	14.43	-3.53	97.14	14.42	-3.53	97.10	14.44	-3.53	97.29	14.41	-3.52	97.02
23	11.27	-4.47	72.90	11.26	-4.46	72.84	11.28	-4.47	73.03	11.26	-4.47	72.87
24	7.25	-2.97	31.98	7.25	-2.97	31.94	7.25	-2.98	32.09	7.24	-2.96	31.94
25	5.67	-2.12	10.54	5.67	-2.11	10.52	5.67	-2.12	10.64	5.67	-2.17	10.71
26	4.77	-0.60	-11.34	4.77	-0.60	-11.34	4.77	-0.60	-11.22	4.77	-0.61	-11.05
33	11.71	-1.71	92.44	11.70	-1.71	92.40	11.72	-1.71	92.63	11.72	-1.71	92.44
34	9.55	-2.11	66.94	9.54	-2.11	66.89	9.56	-2.11	67.13	9.57	-2.11	66.95
35	6.30	-1.15	20.21	6.30	-1.15	20.17	6.31	-1.16	20.37	6.34	-1.15	20.23
36	4.92	-0.23	-14.05	4.91	-0.23	-14.06	4.92	-0.23	-13.88	4.95	-0.25	-14.21
37	4.65	0.52	-35.79	4.65	0.52	-35.77	4.65	0.52	-35.59	4.70	0.51	-35.56
44	10.63	0	88.73	10.62	0	88.68	10.64	0	88.93	10.61	0.00	88.69
45	8.80	0	62.83	8.79	0	62.77	8.81	0	63.04	8.77	0.00	62.76
46	5.83	0	13.99	5.83	0	13.95	5.84	0	14.18	5.79	0.00	13.90
47	4.64	0	-22.52	4.63	0	-22.52	4.64	0	-22.32	4.58	0.00	-22.71
48	4.55	0	-45.89	4.54	0	-45.87	4.54	0	-45.66	4.49	0.00	-45.76
52	-0.92	0	5.86	-0.92	0	5.86	-0.94	0	6.10	-0.95	0.00	5.96
72	3.85	-0.78	-30.12	3.85	-0.78	-30.10	3.84	-0.77	-29.94	3.82	-0.75	-29.90
81	4.11	2.80	11.16	4.10	2.80	11.15	4.10	2.80	11.21	4.12	2.70	10.66
85	-5.40	1.87	32.17	-5.40	1.87	32.15	-5.42	1.88	32.28	-5.40	1.87	32.18
Π_p			-			-			-			27008696

Table 19
Comparison of axial forces (kN) results for saddle net.

Member	Kwan [55]	Lewis [59]	PINN
11-12	53.70	54.14	53.703
23–24	57.80	58.12	57.775
47–48	62.51	62.78	62.483
60–61	59.90	60.18	59.952
72–73	54.83	55.16	54.856
85–86	50.23	50.60	50.215
1–11	75.43	75.52	75.432
24–35	69.30	69.56	69.254
28–39	48.00	48.28	48.055
62–73	57.37	57.61	57.382
67–78	78.69	78.89	78.687
85–95	63.55	63.66	63.542

4. Discussion

From Table 20, it is easily noticeable that the SM algorithm requires considerably fewer iterations than the PINN in 5 out of 8 problems. In terms of the computational time, the efficiency of the SM method performs better than the PINN, but only for simple problems, such as those given in numerical examples 1-5. In more complex cable net structures 6-8, a substantial amount of time is spent on solving the linear equations through matrix inversion. Note that the PINN framework needed more iterations than SM, but it was a simple and easily applied method without calculating any inverse operators. Besides, the BP algorithm allows to automatically calculate the sensitivity of the TPE with respect to the parameters of the network. Furthermore, one of the most important features of the NN is its capability to approximate multivariate nonlinear function. For all these reasons, the efficiency of the PINN was demonstrated through the complex cases compared with the conventional algorithm.

Table 20
Efficiency of the different algorithms.

Example	No. of dofs	SM			PINN		
		Total no. of iterations	CPU time per iteration (sec)	Total CPU time (sec)	Total no. of iterations	CPU time per iteration (sec)	Total CPU time (sec)
Simple net	3	3	0.0473	0.1418	81	0.0157	1.2723
Flat net 2×1	6	12	0.0528	0.6337	417	0.0095	3.9432
Flat net 2×2	12	34	0.0580	1.9720	303	0.0107	3.2566
Hyperbolic net	36	200	0.0422	8.4369	392	0.0158	6.175
Spatial net	45	85	0.0756	6.4299	486	0.0158	8.1858
Dual case 1	54	900	0.0173	15.5639	459	0.0258	13.0777
Dual case 2	54	1700	0.0137	23.3420	493	0.0284	14.0182
Saddle net	189	-	0.7813	-	475	0.1034	49.1059

In terms of the accuracy, the obtained results above indicated that our model outperforms other methods. In these cable net structures, our model identified the best smallest TPE in 7 out of 8 problems, while the SM algorithm failed to converge in the case of the saddle net. It is obvious that the accuracy associated with the inversion problem of the stiffness matrix method depends on the conditioning of the original matrix. And this is one major drawback leading to the instability of the SM algorithm. In addition, as pointed out by Lewis [20], the selection of the initial point as well as the weighting factor also influences the accuracy.

For the PINN framework, despite its advantages, the implementation may face certain challenges that are yet to be resolved. Firstly, the solutions may be trapped in local minimum due to initial parameters. Secondly, Grid search and trial-and-error methods were employed to choose the best-fitted network, so it depends on user experience too much. Hence, the hyperparameters of the network need to be automatically tuned using an optimization algorithm. Another challenge is related to changes in conditions, such as loads, BCs, and so on, that occur when the network has to be iteratively re-trained.

5. Conclusions

In this study, a robust physics-informed NN framework has been successfully applied to accurately estimate the nonlinear behavior of cable net structures due to the changes of geometry. In order to achieve the objective, the network is constructed to guide the learning process by minimizing the TPE, which serves as the derivative-free loss function. And the mechanical behaviors of the cable net are found immediately at the training end. The simplicity and effectiveness of the proposed scheme are demonstrated through numerical examples for the geometric nonlinear analysis of cable net structures. The obtained results clearly demonstrate that our approach outperforms existing algorithms, achieving the smallest TPE values. One interesting aspect of this model is its ability to accurately and simply capture the nonlinear behavior of cable nets without requiring any structural analyses. Moreover, the model's learning capability solely relies on the joint coordinates provided as training data. On the other hand, the sensitivity of the TPE can be easily and quickly determined by utilizing the automatic differentiation, which is integrated into the NN model. With these remarkable characteristics, the proposed approach shows great promise in offering a novel method for tackling nonlinear structural analysis problems without relying on FEA.

However, the implementation of this study may encounter some challenges which have still left unsolved. To overcome the computing challenges, there are several promising directions for future work. First, future studies can significantly automate the tuning of hyperparameters during the training process. And surrogate-based optimization algorithms are well-suited for this task, such as Bayesian optimization technique. Next, the development of optimizers can incorporate self-adaptive momentum to avoid getting trapped in a local optimum, which is promising to circumvent these shortcomings. In addition, the combination of PINN and transfer learning is another promising strategy for generalizing to new conditions, which may help predict structural behaviors without re-training the model.

CRediT authorship contribution statement

Dai D. Mai: Conceptualization, Methodology, Software, Formal analysis, Investigation, Writing - original draft, Writing review & editing, Visualization, Funding acquisition. **Bao Tri Diep**: Data curation, Validation. **Thanh-Danh Lam**: Data curation, Validation. **Hau T. Mai**: Conceptualization, Methodology, Writing - original draft, Writing review & editing, Supervision.

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.

Acknowledgment

This work belongs to the project grant No: T2024-01 funded by Ho Chi Minh City University of Technology and Education, Vietnam.

References

[1] M. S. A. Abad, A. Shooshtari, V. Esmaeili, A. N. Riabi, Nonlinear analysis of cable structures under general loadings, Finite elements in analysis and design 73 (2013) 11–19.

[2] N.-I. Kim, S. Thai, J. Lee, Nonlinear elasto-plastic analysis of slack and taut cable structures, Engineering with Computers 32 (2016) 615–627.

[3] C. Taube, H.-G. Timmler, G. Morgenthal, Enhanced method for the nonlinear structural analysis based on direct energy principles, Engineering Structures 204 (2020) 109789.

[4] W. C. Knudson, Static and dynamic analysis of cable-net structures, University of California, Berkeley, 1971.

[5] J. H. Argyris, D. W. Scharpf, Large deflection analysis of prestressed networks, Journal of the Structural Division 98 (1972) 633–654.

[6] H. Ozdemir, A finite element approach for cable problems, International Journal of Solids and Structures 15 (1979) 427–437.

[7] Z. Chen, Y. Wu, Y. Yin, C. Shan, Formulation and application of multi-node sliding cable element for the analysis of suspen-dome structures, Finite Elements in Analysis and Design 46 (2010) 743–750.

[8] H. Ali, A. Abdel-Ghaffar, Modeling the nonlinear seismic behavior of cable-stayed bridges with passive control bearings, Computers & Structures 54 (1995) 461–492.

[9] J. Coyette, P. Guisset, Cable network analysis by a nonlinear programming technique, Engineering Structures 10 (1988) 41–46.

[10] S. Thai, N.-I. Kim, J. Lee, Isogeometric cable elements based on b-spline curves, Meccanica 52 (2017) 1219–1237.

[11] W. T. O’Brien, A. J. Francis, Cable movements under two-dimensional loads, Journal of the Structural Division 90 (1964) 89–123.

[12] H. Jayaraman, W. Knudson, A curved element for the analysis of cable structures, Computers & Structures 14 (1981) 325–333.

[13] A. Andreu, L. Gil, P. Roca, A new deformable catenary element for the analysis of cable net structures, Computers & Structures 84 (2006) 1882–1890.

[14] H.-T. Thai, S.-E. Kim, Nonlinear static and dynamic analysis of cable structures, Finite elements in analysis and design 47 (2011) 237–246.

[15] H. T. Mai, Q. X. Lieu, J. Kang, J. Lee, A robust unsupervised neural network framework for geometrically nonlinear analysis of inelastic truss structures, Applied Mathematical Modelling 107 (2022) 332–352.

[16] J. Pietrzak, Matrix formulation of static analysis of cable structures, Computers & Structures 9 (1978) 39–42.

[17] H. Bucholdt, B. McMillan, Iterative methods for the solution of pretensioned cable structures, in: IASS Symposium on Tension Structures and Space Frames Tokyo 1971, 1971.

[18] P. Sufian, A. Templeman, Analysis and design of cable net structures through optimization techniques, WIT Transactions on The Built Environment 2 (1970).

[19] F. Sufian, Analysis and design methods for pretensioned cable net structures., Ph.D. thesis, University of Liverpool, 1992.

[20] W. Lewis, The efficiency of numerical methods for the analysis of prestressed nets and pin-jointed frame structures, Computers & structures 33 (1989) 791–800.

[21] Y. C. Toklu1a, G. Bekdaş2b, R. Temür, Analysis of cable structures through energy minimization, Structural Engineering and Mechanics 62 (2017) 749–758.

[22] Y. Toklu, N. Toklu, Analysis of structures by total potential optimization using metaheuristic algorithms (tpo/ma) in siarry, P.“Heuristics: Theory and Applications”, Nova Science (2013).

[23] V. M. Nguyen-Thanh, C. Anitescu, N. Alajlan, T. Rabczuk, X. Zhuang, Parametric deep energy approach for elasticity accounting for strain gradient effects, Computer Methods in Applied Mechanics and Engineering 386 (2021) 114096.

[24] 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, Computer Methods in Applied Mechanics and Engineering 383 (2021) 113933.

[25] H. T. Mai, Q. X. Lieu, J. Kang, J. Lee, A novel deep unsupervised learning-based framework for optimization of truss structures, Engineering with Computers (2022) 1–24.

[26] H. T. Mai, S. Lee, D. Kim, J. Lee, J. Kang, J. Lee, Optimum design of nonlinear structures via deep neural network-based parameterization framework, European Journal of Mechanics-A/Solids (2022) 104869.

[27] H. T. Mai, J. Kang, J. Lee, A machine learning-based surrogate model for optimization of truss structures with geometrically nonlinear behavior, Finite Elements in Analysis and Design 196 (2021) 103572.

[28] H. T. Mai, D. D. Mai, J. Kang, J. Lee, J. Lee, Physics-informed neural energy-force network: a unified solver-free numerical simulation for structural optimization, Engineering with Computers (2023) 1–24.

[29] H. T. Mai, S. Lee, J. Kang, J. Lee, A damage-informed neural network framework for structural damage identification, Computers & Structures 292 (2024) 107232.

[30] D. T. Trinh, S. Lee, J. Kang, J. Lee, Force density-informed neural network for prestress design of tensegrity structures with multiple self-stress modes, European Journal of Mechanics-A/Solids 94 (2022) 104584.

[31] A. Chandrasekhar, K. Suresh, Tounn: Topology optimization using neural networks, Structural and Multidisciplinary Optimization 63 (2021) 1135–1149.

[32] V. M. Nguyen-Thanh, X. Zhuang, T. Rabczuk, A deep energy method for finite deformation hyperelasticity, European Journal of Mechanics-A/Solids 80 (2020) 103874.

[33] Z. Huang, L. Peng, An improved plate deep energy method for the bending, buckling and free vibration problems of irregular kirchhoff plates, Engineering Structures 301 (2024) 117235.

[34] C. Chadha, J. He, D. Abueidda, S. Koric, E. Guleryuz, I. Jasiuk, Improving the accuracy of the deep energy method, Acta Mechanica 234 (2023) 5975–5998.

[35] D. W. Abueidda, S. Koric, E. Guleryuz, N. A. Sobh, Enhanced physics-informed neural networks for hyperelasticity, International Journal for Numerical Methods in Engineering 124 (2023) 1585–1601.

[36] B. Yu, et al., The deep ritz method: a deep learning-based numerical algorithm for solving variational problems, Communications in Mathematics and Statistics 6 (2018) 1–12.

[37] M. Liu, Z. Cai, K. Ramani, Deep ritz method with adaptive quadrature for linear elasticity, Computer Methods in Applied Mechanics and Engineering 415 (2023) 116229.

[38] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, J. M. Siskind, Automatic differentiation in machine learning: a survey, Journal of Marchine Learning Research 18 (2018) 1–43.

[39] J.-H. Bastek, D. M. Kochmann, Physics-informed neural networks for shell structures, European Journal of Mechanics-A/Solids 97 (2023) 104849.

[40] T. G. Grossmann, U. J. Komorowska, J. Latz, C.-B. Schönlieb, Can physics-informed neural networks beat the finite element method?, arXiv preprint arXiv:2302.04107 (2023).

[41] A. Fallah, M. M. Aghdam, Physics-informed neural network for bending and free vibration analysis of three-dimensional functionally graded porous beam resting on elastic foundation, Engineering with Computers (2023) 1–18.

[42] M. Raissi, P. Perdikaris, G. E. Karniadakis, Physics informed deep learning (part i): data-driven solutions of nonlinear partial differential equations. arxiv e-prints, arXiv preprint arXiv:1711.10561 (2017).

[43] M. Raissi, P. Perdikaris, G. E. Karniadakis, Physics informed deep learning (part ii): Data-driven discovery of nonlinear partial differential equations. 2017, arXiv preprint arXiv:1711.10566 (2017).

[44] 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, Computational Mechanics 71 (2023) 543–562.

[45] S. Rezaei, A. Harandi, A. Moeineddin, B.-X. Xu, S. Reese, A mixed formulation for physics-informed neural networks as a potential solver for engineering problems in heterogeneous domains: comparison with finite element method, Computer Methods in Applied Mechanics and Engineering 401 (2022) 115616.

[46] K. A. Luong, T. Le-Duc, J. Lee, Automatically imposing boundary conditions for boundary value problems by unified physics-informed neural network, Engineering with Computers (2023) 1–23.

[47] H. T. Mai, T. T. Truong, J. Kang, D. D. Mai, J. Lee, A robust physics-informed neural network approach for predicting structural instability, Finite Elements in Analysis and Design 216 (2023) 103893.

[48] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).

[49] M. Mahsereci, L. Balles, C. Lassner, P. Hennig, Early stopping without a validation set, arXiv preprint arXiv:1703.09580 (2017).

[50] L. Prechelt, Early stopping-but when?, in: Neural Networks: Tricks of the trade, Springer, 2002, pp. 55–69.

[51] K. A. Halvordson, Three-Dimensional Analysis of Geogrid Reinforcement used in a Pile-Supported Embankment, Ph.D. thesis, Virginia Tech, 2007.

[52] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: International conference on machine learning, pmlr, 2015, pp. 448–456.

[53] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: Proceedings of the thirteenth international conference on artificial intelligence and statistics, JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.

[54] W. Lewis, M. Jones, K. Rushton, Dynamic relaxation analysis of the non-linear static response of pretensioned cable roofs, Computers & Structures 18 (1984) 989–997.

[55] A. Kwan, A new approach to geometric nonlinearity of cable structures, Computers & Structures 67 (1998) 243–252.

[56] C. H. Thornton, C. Birnstiel, Three-dimensional suspension structures, Journal of the Structural division 93 (1967) 247–270.

[57] A. Nuhoglu, Nonlinear analysis of cable systems with point based iterative procedure, Scientific Research and Essays 6 (2011) 1186–1199.

[58] F. Sufian, A. Templeman, On the non-linear analysis of pre-tensioned cable net structures, Struct. Eng 4 (1992) 147–158.

[59] W. Lewis, A comparative study of numerical methods for the solution of pretensioned cable networks, in: Proceedings of the International Conference on the Design and Construction of Non-Conventional Structures, Edinburgh, UK, 1987, pp. 8–10.