Advanced Steel Construction – Vol. 19 No. 4 (2023) 366–374
DOI:10.18057/IJASC.2023.19.4.5

INVERSION METHOD OF UNCERTAIN PARAMETERS FOR TRUSS STRUCTURES BASED ON GRAPH NEURAL NETWORKS

Zhang-Qi Wang [1, 2, 3], Zhe Zheng [1], Jun-Wei MengXiang [1] and Wen-Qiang Jiang [1, 2, 3, *]

[1] *Department of Mechanical Engineering, North China Electric Power University, Baoding, China*
[2] *Hebei Engineering Research Center for Advanced Manufacturing & Intelligent Operation and Maintenance of Electric Power Machinery, North China Electric Power University, Baoding, China*
[3] *Hebei Key Laboratory of Electric Machinery Health Maintenance & Failure Prevention, North China Electric Power University, Baoding 071003, China*
** (Corresponding author: E-mail: wenqiang.jiang@ncepu.edu.cn)*

ABSTRACT

Uncertainty exists widely in practical engineering. It is an important challenge in engineering structural analysis. In truss structures, the uncertainties of axial stiffness of bolted joints will significantly affect the mechanical behavior of the structure as the axial load is dominated by the member internal forces. Structural response analysis based on determined structural parameters is a common forward problem that can be solved by modeling analysis methods. However, the uncertainties parameter of axial stiffness of bolted joint cannot be determined during the design and analysis of truss structure in the direct nonlinear analysis method. Structural parameter identification based on structural response is a typical inverse problem in engineering, which is difficult to solve using traditional analysis tools. In this paper, an inverse model based on Graph Neural Network (GNN) is proposed. The feature encoding method for transforming truss structures into graph representations of GNN is defined. A parameterized acquisition method for large-scale datasets is presented, and an innovative inversion model based on GNN for the inversion of uncertain parameters of truss structures is proposed. The proposed method is shown to perform well with an inversion accuracy, and accurate results can be obtained with limited data sets. The inversion method has strong data mining capability and model interpretability, making it a promising direction for exploring engineering structural analysis.

ARTICLE HISTORY

Received: 7 April 2023
Revised: 6 June 2023
Accepted: 20 June 2023

KEYWORDS

Truss structures;
Uncertain parameters;
Inversion method;
GNN

1. Introduction

Parameter uncertainty is a common problem in practical engineering, which is a significant challenge to the current field of engineering analysis[1,2]. Engineering structures, such as steel buildings[3], bridges[4,5] and towers[6-8], are commonly simplified as ideal truss structures. Bolted connections are usually regarded as pin or rigid joints. Recent research indicates that the mechanical behavior of bolted connection has a significant impact on the overall structure as the axial load is dominated the member's internal forces. The direct nonlinear analysis method considering joint effects can greatly improve the results which is in accordance with the experimental results very well[9,10]. However, the mechanical behavior of bolted connection is greatly influenced by bolt preload force, processing and manufacturing errors and structural gaps between bolts and bolt holes. There are many uncertainties in the pore structure, interface and load transfer path of bolted connection nodes, which can cause uncertain mechanical behavior of the bolted connection, further complicating the analysis of truss structure. A large number of tower tests and finite element simulations have revealed that the test results of internal forces and nodal displacements deviate greatly from calculated values, mainly because the slip of bolted connections is not taken into account[11,12]. As illustrated in Fig. 1, conventional finite element modeling makes it relatively easy to obtain the response from determined structural parameters (i.e., the forward problem). However, it is quite difficult to obtain the uncertain parameters of the structure affected by the bolted joint slip from the structural response (i.e., the reverse problem). Since structural parameters cannot be determined in advance as the uncertainty of structure parameters, finite element calculations cannot be performed.

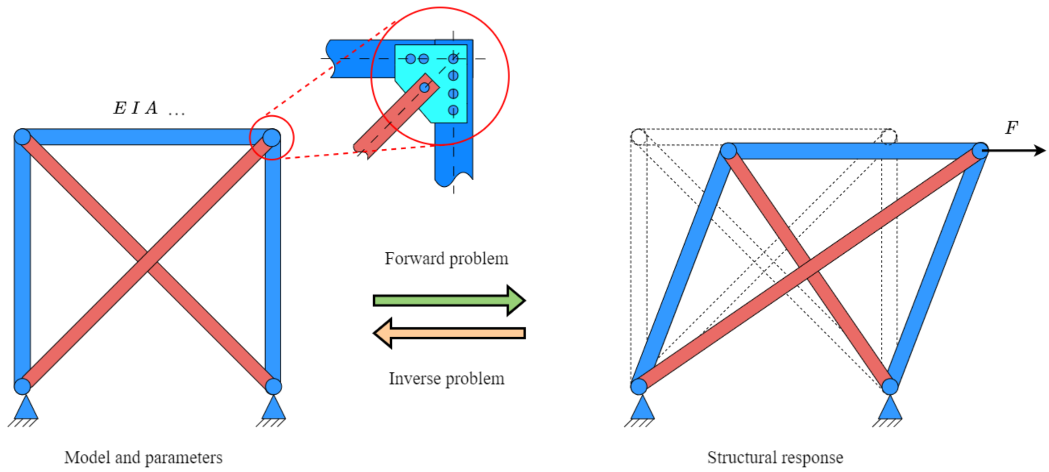


Fig. 1 The forward and inverse problems of truss structure

Structural parameter identification based on structural response is a typical inverse problem in engineering. The traditional method to solve the inverse problem is the Monte Carlo Simulation, which is widely used in probabilistic analysis and simulates the real behavioral characteristics of the actual problem. However, the uncertainty of bolted connection in truss structures cannot be expressed by a single random variable or random field, and a large number of uncertain parameters leads to difficulties in solving the description methods based on parametric probabilities, which becomes particularly challenging in inversion analysis. Therefore, It is important to put forward more effective methods to resolve the problem of uncertainty analysis and inversion of truss structures.

The inverse problem is a real challenge to solve due to the complexity of practical problems, which makes it difficult to prove the existence, uniqueness, and stability of the solution. The study of inverse problems originated in the field of mathematics, and has progressed from classical statistical/Bayesian methods[13] to cutting-edge deep neural networks[14,15]. The classical statistical/Bayesian method uses probabilistic approaches to establish a probability model based on known measurement data. However, this method is time-consuming and inefficient[16]. With the development of computer technology, neural networks have emerged as a powerful tool for solving inverse problems. The neural network methods have excellent nonlinear fitting ability and good performance in solving inverse problems in various fields[17-19]. However, for engineering problems, it is challenging to convert the actual structure into input features for the neural network model[20].

Graph Neural Network (GNN) is a method of applying deep neural networks to graph structure data, which has powerful graph data processing ability, and brings new vitality to various fields[21-23]. In recent years, GNN is integrating with the field of physical structure analysis[24], such as predicting the motion state of objects[25], and shows great potential in engineering applications. However, there are few studies related to the inversion method of engineering structures based on GNN at present. The truss structure, for example, can be represented by nodes and edges, which correspond naturally to the graph structure data. Therefore, it is easy to establish the correspondence between truss structure parameters and graph structure data, and consider whether it is possible to solve the difficult inverse problems in truss structure based on GNN.

In order to solve the inverse problems of uncertain parameters in truss structures, an innovative computational model based on GNN is studied. This approach will provide significant advances in the field of structural engineering, provide greater design optimization and enhanced safety for engineering structures.

2. Parametric uncertainty model of bolted connections

In practice, most bolted joints in truss structures such as bridges, lattice towers, etc. are suffering from transverse load. Therefore, the load direction of the bolted joint is perpendicular to the axial of the bolt rod (see Fig.2). The joint slippage is the relative displacement of bolted joints, which occurs under the action of a shear load larger than the friction force between the fastened parts[11]. Initially, the shear load in the joint is counteracted by the friction force between the fastened parts, the relative slip does not occur and the clearance between bolt and bolt hole exists. Then, the shear load eventually reaches and exceeds the maximum frictional force between the fastening parts as the external load increases, which will cause relative slip until the available clearance disappears completely. Finally, as the external load is further increased, the bolt will be extruded with the bolt-hole wall, and the bolt-hole wall will experience a complex nonlinear deformation process, which ranges from elastic deformation to plastic deformation and eventual destruction.

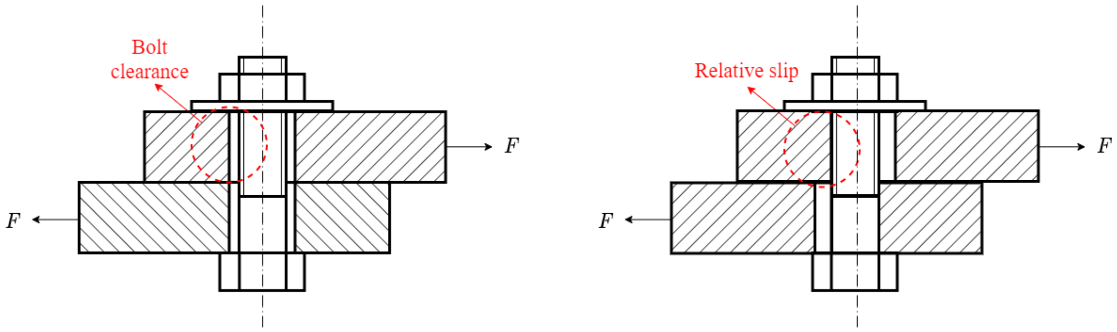


Fig. 2 Bolted joint slippage

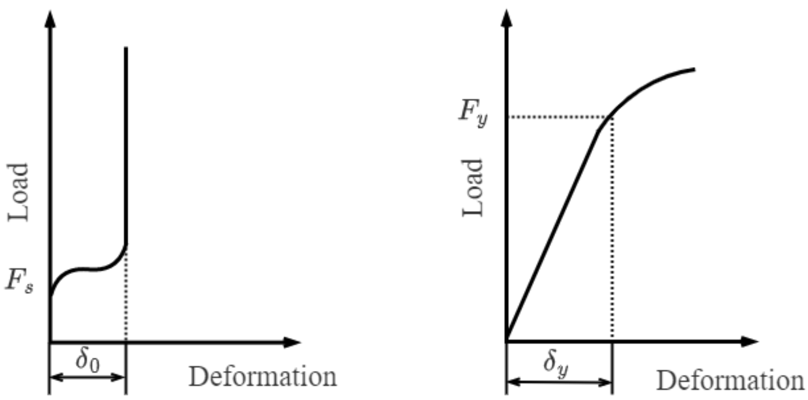


Fig. 3 Joint slippage model

In order to investigate the influence of joint slippage, various numerical models are proposed such as the numerical model[26], parametric model[27], and micro-slip model[28]. In this paper, the four parameters joint slippage model is selected, which divide joint slippage processes into clearance filled slip and extrusion deformation slip[29] (see Fig.3), the load-deformation relationship can be expressed as:

$$\delta_s = \delta_0 \left[\frac{F/F_s}{\left(1+\left(F/F_s\right)^n\right)^{1/n}} \right]^m + \alpha_s \delta_y \left(\frac{F}{F_y} \right)^{N_s} + \delta_y \frac{F}{F_y} \tag{1}$$

Where δ_s is the total deformation of the joint slip, δ_0 is the bolt clearance, δ_y is the deformation at yield; F is the external load, F_s is the critical slip load, F_y is the yield load; m, n, a_s and N_s are the shape parameters of the curve which can be determined from the tensile tests of bolted joints.

However, due to the complexity of the bolt connection slip process, the bolted joint mechanical behavior can be affected by many uncertainties, such as bolt preload force, processing and manufacturing errors and the bolt hole structure, etc. In truss structures, the bolted joint slippage is inevitable as the bolt clamping forces are relatively low, which led to the discrepancy between the test results and simulation results[7,11,29]. However, these joint slippage effects cannot be determined in advance during the analysis of truss structures because the bolted joint states are not easy to obtain. Most of the currently used truss analysis models ignore the influence of the slip of bolted joints, therefore the member internal forces cannot be accurately obtained.

In order to take joint slippage uncertainties into account, the equivalent reduction method of axial stiffness is adopted. The members in the truss structure are assumed to be in an elastic state during loading. Therefore, the total deformation of the member (δ) is the sum of the joint slip (δ_s) and the elastic deformation (δ_e), expressed as:

$$\delta = \delta_s + \delta_e \tag{2}$$

The elastic deformation of the member is determined by

$$\delta_e = \frac{Fl}{EA} \tag{3}$$

Where F is the axial force, l is the length of the member, E is the elastic modulus, and A is the cross-sectional area. The axial stiffness of the member is obtained from the formula:

$$K = \frac{F}{\delta} = \frac{EA\delta_e / l}{\delta} = \frac{\delta_e}{\delta}\frac{EA}{l} = \left(1 - \frac{\delta_s}{\delta}\right)\frac{EA}{l} \tag{4}$$

Let $\eta = 1 - \delta_s / \delta$, then the axial stiffness is expressed as:

$$K = \eta \frac{EA}{l} \tag{5}$$

Where η is the axial stiffness reduction factor. The uncertainties of the joint slip can be represented by the uncertainties of the axial stiffness reduction factor of the member.

3. The inversion method of uncertain parameters

In this paper, the uncertain parameter problem of the truss structure caused by the slip of bolted connections is considered, and an innovative method based on GNN is proposed to solve the uncertain parameters identification problem based on structural response.

3.1. Graph representation of truss structure

In practice, truss structures are commonly represented visually as a graph consisting of nodes and edges. This representation method is exactly similar to GNN, which is also composed of nodes and edges. Therefore, the graph representation can be easily defined as: graph nodes represent the nodal connections (bolted connections) of the truss, and graph edges represent the members of the truss.

In order to illustrate clearly, a basic truss structure is isolated from the whole structure of the lattice transmission tower as shown in Fig.4. The graph structure of GNN is composed of 4 nodes and 6 edges (there are 12 edges if considering the direction of the edge). The boundary conditions assume as nodes 1 and 2 are free, and nodes 3 and 4 are fixed and fully constrained. A horizontal force is loaded on Node 2. In the presented truss structure model, the uncertainty of diagonal members due to bolted joint slippage is considered, while the other members are defined as deterministic parameters. As shown in Fig.4, the red line and blue line are used to represent the difference in member parameters. The node features and edge features inputted in GNN are defined respectively according to the node parameters and edge parameters, which are listed in Table 1.

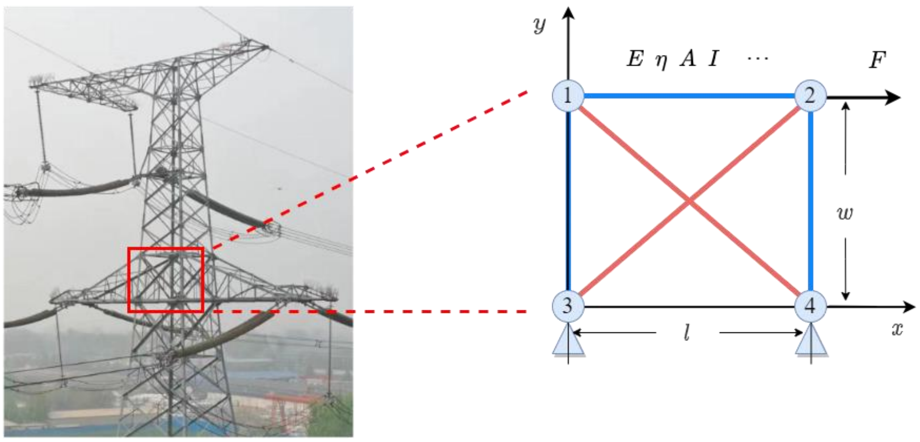


Fig. 4 Graph representation of truss structure

Table 1
Graph feature coding

Parameter	Data type	Feature description
Node features: [C_x, C_y, x, y, F_x, F_y, D_x, D_y]		
C_x	int	The constraint of the node in the x direction. 0 represents no constraints, and 1 represents constraints.
C_y	int	The constraint of the node in the y direction. 0 represents no constraints, and 1 represents constraints.
x	float	Node coordinate in the x direction (m).
y	float	Node coordinate in the y direction (m).
F_x	float	Node force in the x direction (N).
F_y	float	Node force in the y direction (N).
D_x	float	Node displacement in the x direction (m).
D_y	float	Node displacement in the y direction (m).
Edge features: [T, E, η, A, I]		
T	int	Member type. 0 represents the beam, and 1 represents the rod.
E	float	Elastic modulus of member (Pa).
η	float	Stiffness reduction factor. Randomly obtained within [0, 1].
A	float	The cross-sectional area of member (m^2).
I	float	Cross-sectional moment of inertia (m^4).

3.2. Parametrized acquisition of datasets

Considering the high cost of experiments, it is generally difficult to obtain enough experimental datasets of engineering structures. Therefore, numerical simulation methods were used to obtain datasets required for the training of GNN in this paper. The uncertainties of truss structure caused by joint slip can be represented by the uncertainties of the axial stiffness reduction factor of the member, which is assumed to be randomly varied in the range of [0, 1]. The large datasets of deterministic structural parameters can be acquired by random sampling of axial stiffness reduction factor η. The node displacements can be obtained through finite element simulation to generate the required datasets for the training of GNN. The datasets acquisition process is shown in Fig. 5.

Step 1: Define number

dataset number: N node number: m

$i=1$ edge number: n

Step 2: Establish Finite element model

material properties: geometric parameters:

$E_i=[E_i^1,E_i^2,...,E_i^n]$ $x_i=[x_i^1,x_i^2,...,x_i^m]$

$T_i=[T_i^1,T_i^2,...,T_i^n]$ $y_i=[y_i^1,y_i^2,...,y_i^m]$

$\eta_i=[\eta_i^1,\eta_i^2,...,\eta_i^n]$ $A_i=[A_i^1,A_i^2,...,A_i^n]$

$I_i=[I_i^1,I_i^2,...,I_i^n]$

Step 3: Additional conditions

constraint condition: loading:

$Cx_i=[Cx_i^1,Cx_i^2,...,Cx_i^m]$ $Fx_i=[Fx_i^1,Fx_i^2,...,Fx_i^m]$

$Cy_i=[Cy_i^1,Cy_i^2,...,Cy_i^m]$ $Fy_i=[Fy_i^1,Fy_i^2,...,Fy_i^m]$

Step 4: Finite element calculation

node displacement:

$Dx_i=[Dx_i^1,Dx_i^2,...,Dx_i^m]$,

$Dy_i=[Dy_i^1,Dy_i^2,...,Dy_i^m]$

node dataset:

$Sv_i=[Cx_i,Cy_i,x_i,y_i,Fx_i,Fy_i,Dx_i,Dy_i]^T$

edge dataset:

$Se_i=[T_i,E_i,\eta_i,A_i,I_i]^T$

$i=i+1$

Step 5: If $i<N$, Change uncertainty parameters

Stiffness reduction factor:

$\eta_i^k=\text{random}(0,1),k=1,2,...,n$

go to Step 2

Step 6: Integrating datasets

node datasets:

$Sv=[Sv_1,Sv_2,...,Sv_N]$

edge datasets:

$Se=[Se_1,Se_2,...,Se_N]$

Fig. 5 The datasets acquisition process

Commonly used parameter units, such as modulus of elasticity (GPa), area (mm2), and length (mm), need to be converted to international standard units, which leads to great differences in quantitative value. In order to avoid the effect of inconsistent parameter units on feature weights, the feature parameters are normalized as follows:

$$\bar{X}=\frac{X-X_{\min}}{X_{\max}-X_{\min}} \tag{6}$$

3.3. The inversion model based GNN

The member internal forces commonly need to be determined by the response of the truss structure in engineering. This problem can be easily conquered by the Finite Element Analysis method (FEA method) for the deterministic analysis model whether it is linear or non-linear. However, the traditional FEA method cannot be effectively solved if the structural parameters are uncertain. This parametric inversion can be challenged based on GNN.

GNN is the method designed to apply deep neural networks to graph-structured data. Compared with general neural networks, GNN can not only deal with non-Euclidean graph structure data but also capture the correlation between adjacent nodes, with strong graph feature mapping capabilities. GNN is a feature learning process on graphs, taking node features and graph structure as input and a new set of node features as output. This process can be expressed as

$$X_{out}=\phi(A,X_{in}) \tag{7}$$

where A is the adjacency matrix expressing the graph structure, X_{in} and X_{out} are the input features and output features respectively, and ϕ is the operator. Importantly, this process only changes the features, not the graph structure.

The framework of the GNN-based inversion model is illustrated in Fig. 6. The general engineering structural parameters cannot be used as the input of the GNN, so the truss structure parameters need to be preprocessed and transformed into graph structure data according to the graph representation method proposed in the previous paper. Then the adjacency matrix A and node features X representing the graph structure data are taken as inputs, and the adjacent node features are aggregated and updated through multiple message passing networks. In order to better mine relevant features and alleviate the overfitting problem, the ReLU activation function is used. The function expression is as follows:

$$\mathrm{Re}LU(x)=\max(0,x) \tag{8}$$

Finally, a new set of node features is output for parameter inversion. It should be noted that the graph structure can be any spatial structure, and both the edge features and graph features can be included in the input node features. Therefore the presented inversion model is appropriate for any other truss structure analysis.

The critical process of GNN is the design of the message passing layer. A message passing layer consists of three update operators: edge update, node update, and global update. In the following description, *u* represents the graph features, *v* represents the node features and e represents the edge features.

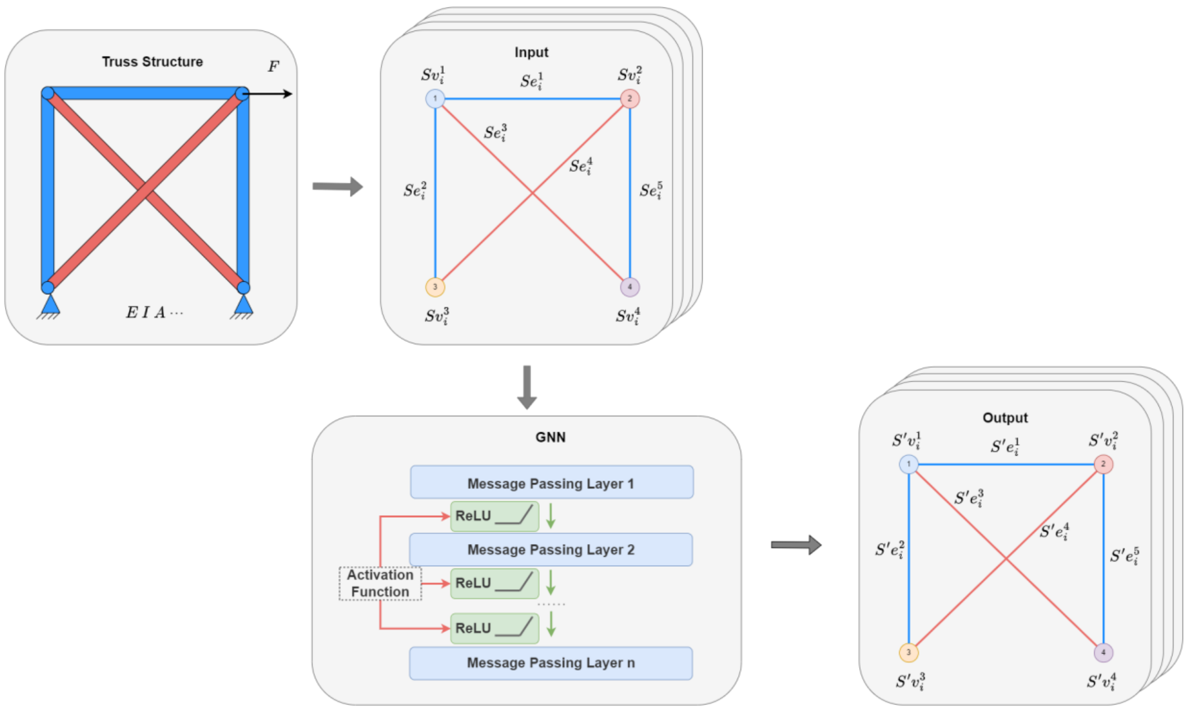


Fig. 6 The GNN-based inversion model

During the edge update step, the edge update function ϕ^e generates new edge features by computing graph features, adjacent node features and edge features. The edge update can be expressed as:

$$e' = \phi^e\left(e, v, u\right) \tag{9}$$

In the node update step, it is necessary to aggregate the adjacent edge features of each node through an aggregation function $\rho^{e\to v}$ as each node may have multiple edges. Then node update function ϕ^v generates new node features by computing graph features, own node features and aggregated edge features. The node update can be expressed as:

$$v' = \phi^v\left(\rho^{e\to v}\left(e'\right), v, u\right) \tag{10}$$

In the global update step, edge features are aggregated globally through aggregation functions $\rho^{e\to u}$, and node features are aggregated globally through aggregation functions $\rho^{v\to u}$ according to specific global problems. Then global update function ϕ^u generates new graph features by computing graph features, aggregated node features and aggregated edge features. The global update can be expressed as:

$$u' = \phi^u\left(\rho^{e\to u}\left(e'\right), \rho^{v\to u}\left(v'\right), u\right) \tag{11}$$

For the inverse problems of uncertain parameters for truss structure, it can be seen that member axial stiffness parameters belong to edge features, and node displacement parameters belong to node features. Therefore, the message passing in the truss structure only occurs between node and edge features, and the global update is unnecessary. In the edge update step, the features of each edge are updated by a multilayer perceptron (MLP) network. The adjacent node features and the original edge features are spliced as the input, and the new edge features are obtained as the output. It should be noted that the new node feature dimension is consistent with the original node feature dimension to ensure that the new node feature can be used as input again. In the node update step, the MLP network is also used to update each node. The difference is that edge feature aggregation steps need to be added in the update. Here the SUM aggregation function is used to aggregate all edge features connected to the same node. The message passing layer constructed is illustrated in Fig. 7. It should be noted that in the message passing layer, the update function ϕ needs to be trained and learned, and the aggregation function ρ is selected independently.

For simple truss inversion analysis, parameter inversion can be achieved by using a single message passing layer. It should be noted that for GNN models, the over-smoothing problem can lead to degradation of the model performance as the number of message passing layers increases.

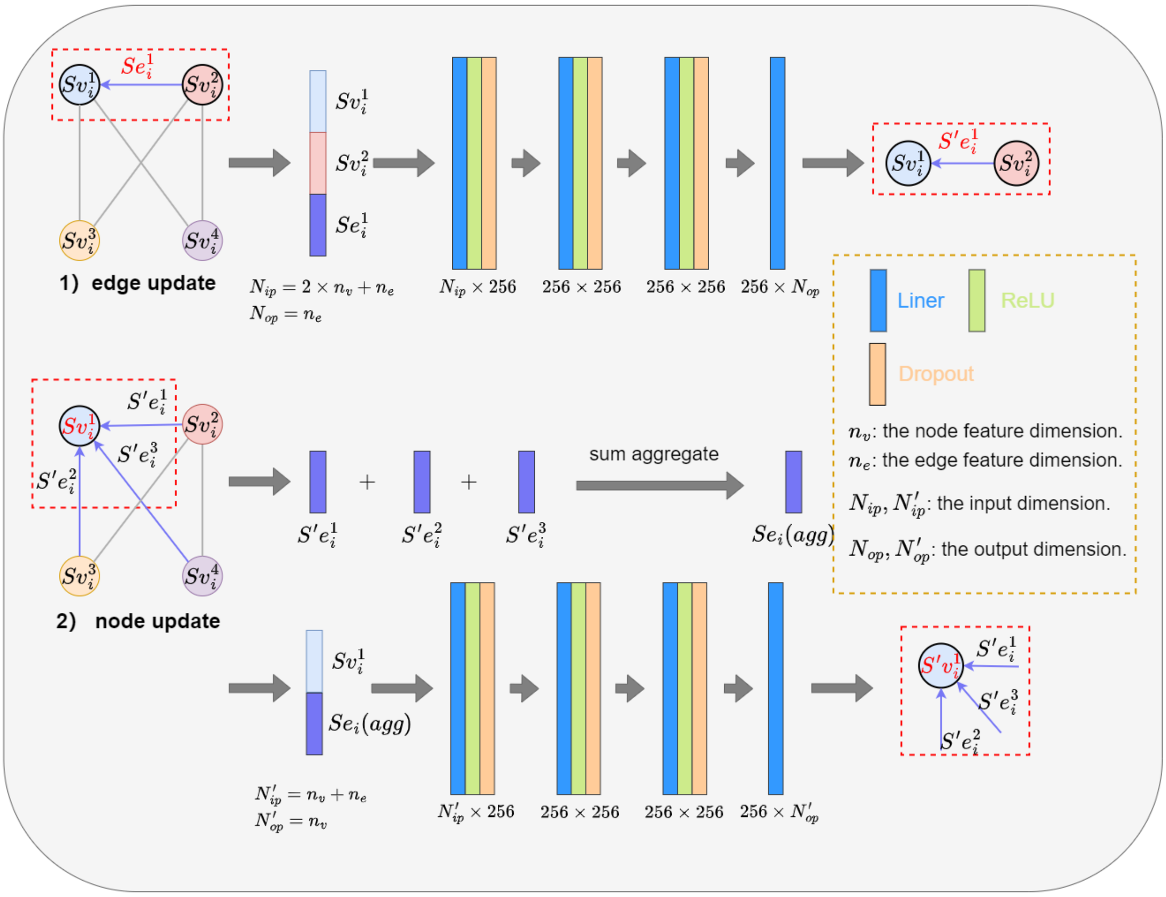


Fig. 7 The message passing mechanism

3.4. Model training and optimization

In GNNs training, the input datasets are shuffled to avoid the influence of the order of input datasets on feature learning. The Adam optimizer is selected for training. The inversion problem studied in this paper is to obtain the member stiffness and inner forces parameters from the node displacement response parameters. Therefore, the elastic modulus parameters in the edge features need to be initialized to 0 in the input features, while the actual elastic modulus parameters are used as the target labels. The inversion error is defined as follows:

$$E_{error} = \sqrt{\frac{\sum\left(E_{out} - E_{target}\right)^2}{n}} \times 100\% \tag{12}$$

Where E_{out} is the output stiffness parameters of the inversion model, E_{target} is the actual stiffness parameters, and n is the number of the output stiffness parameters.

3.5. Case study

In order to validate the computational effectiveness of the inverse model presented in this paper, the numerical tests of a basic planar truss structure are analyzed as shown in Fig. 4. The implementation steps are as follows:

(1) Graph representation of the truss structure

The graphical representation of the truss structure studied in this paper is shown in Fig. 8, the geometry and physics parameters of the truss structure are listed in Table 2, and the defined features of the corresponding graph structure are listed in Table 3.

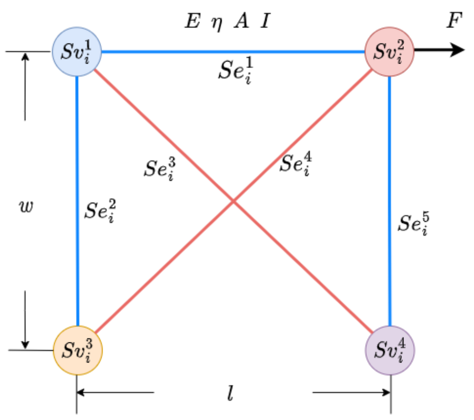


Fig. 8 Graphical representation of the studied trusses

Table 2
The truss structure parameters

L (m)	W (m)	A (m²)	I (m⁴)	E (GPa)	F (kN)	η
1	1	7.854×10^{-3}	4.9087×10^{-6}	206	10	Random(0,1)

Table 3
The defined features of graph structure

	Feature number	Feature description
Node feature	Sv^1	[0, 0, 0, l, 0, 0, Dx^1, Dy^1]
	Sv^2	[0, 0, l, w, F, 0, Dx^2, Dy^2]
[Cx, Cy, x, y, Fx, Fy, Dx, Dy]	Sv^3	[1, 1, 0, 0, 0, 0, Dx^3, Dy^3]
	Sv^4	[1, 1, l, 0, 0, 0, Dx^4, Dy^4]
	Se^1	[0, $\eta^1\times E$, A, I]
Edge feature	Se^2	[0, $\eta^2\times E$, A, I]
	Se^3	[1, $\eta^3\times E$, A, I]
[T, ηE, A, I]	Se^4	[1, $\eta^4\times E$, A, I]
	Se^5	[0, $\eta^5\times E$, A, I]

(2) Parametric acquisition of datasets

In order to meet the characteristics of difficult data acquisition in engineering practice, 1000 sets of numerical simulation datasets are obtained according to the computational flow shown in Fig. 5. To avoid the effect of inconsistent parameter units on feature weights, the features are normalized by Equation (6) and then involved in the GNN model training. To ensure the efficiency of the training process and the correctness and accuracy of the training results, the datasets are divided into training sets, test sets and validation sets in the ratio of 8:1:1.

(3) Construction of GNN-based inversion model

The inverse model is constructed according to Fig. 6, and the message passing network is constructed according to Fig. 7. The input features of the network are first preprocessed by initializing the stiffness feature in the edge features to 0, while the stiffness true values are used as the target label. The input features are the preprocessed node features and edge features:

$$\begin{aligned} Sv &= [Cx, Cy, x, y, Fx, Fy, Dx, Dy] \\ Se &= [T, 0, A, I] \end{aligned} \tag{13}$$

The inverse model contains two message passing layers, and then the edge features are output through a linear transformation layer. In the message passing layer, the message passing is realized through edge update and node update in turn. The update process is as follows:

$$\begin{aligned} Se^{(1)} &= \phi^e\left(Se^{(0)}, Sv^{(0)}\right) \\ Sv^{(1)} &= \phi^v\left(\rho^{e\to v}\left(Se^{(1)}\right), Sv^{(0)}\right) \\ Se^{(2)} &= \phi^e\left(Se^{(1)}, Sv^{(1)}\right) \\ Sv^{(2)} &= \phi^v\left(\rho^{e\to v}\left(Se^{(2)}\right), Sv^{(1)}\right) \end{aligned} \tag{14}$$

The updated edge features are then passed through a linear transformation layer to obtain the final output

$$Se_{out} = Linear\left(Sv^{(2)}\right) = \left[T^{(2)}, E_{out}, A^{(2)}, I^{(2)}\right] \tag{15}$$

(4) Model training and optimization

The model optimization uses Adam Optimizer with an initial learning rate of 1×10^{-5}, and the learning rate decreases with the training epoch. The output error is calculated by Equation (12). To reduce the hardware requirements for inversion training, this training was performed using only an i5-10400 CPU. Finally, the model parameters are optimized by the back propagation algorithm until the model results meet the requirements.

The convergence of different data sets during the training process is shown in Fig. 9. It can be seen in the figure that the model converges rapidly during the training process, and the inversion accuracy is stable at about 99%. In particular, the model training accuracy is higher than 0.978 as can be seen in the local enlargement of the first 50 epochs, and the model has a good inversion performance with less training volume. Therefore, the inversion results fully satisfy the engineering calculation requirements.

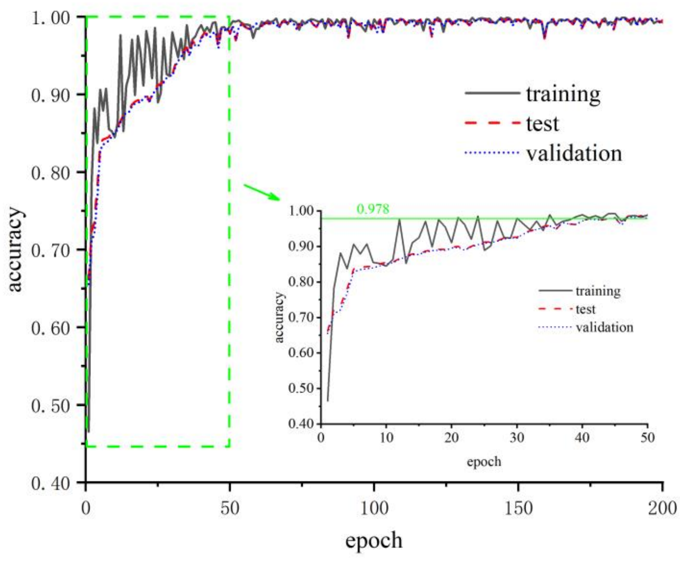


Fig. 9 The convergence curves of different datasets

4. Discussion

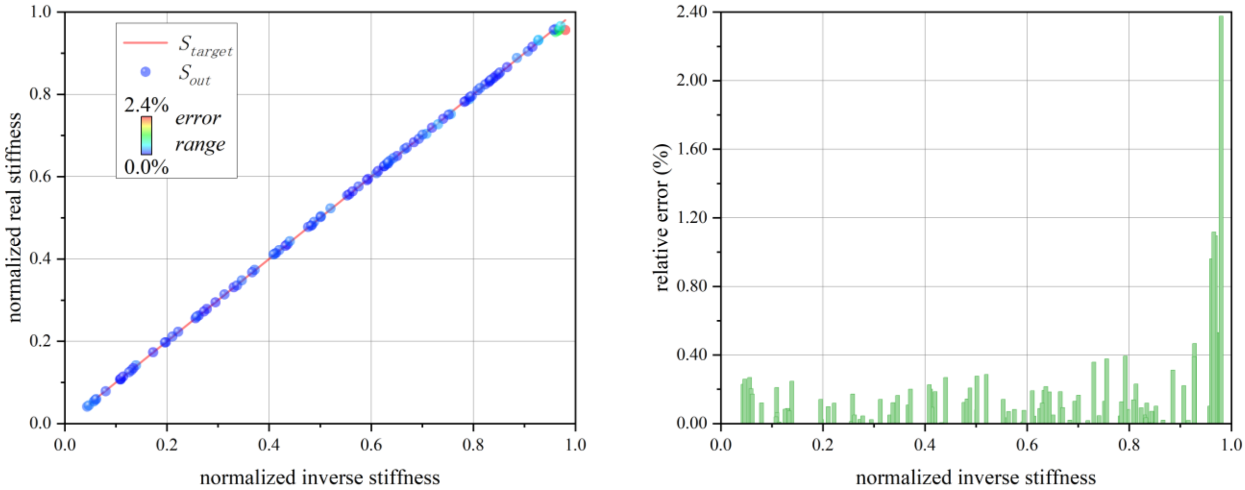
4.1. Performance of inversion model

In order to study the performance of the inversion model, the GNN model trained over 200 epochs is used to solve the inverse problem of uncertain stiffness of the truss structure. Since the graph structure in the GNN considers the direction of edges, the final result of the edge stiffness values is obtained by averaging the stiffness values of two directions on the same edge. The inversion results of member Se_i^3 (see Fig. 8) are shown in Fig.10, the inversion results of member Se_i^4 are illustrated in Fig. 11, and the inversion results of the other members are illustrated in Fig. 12. The stiffness of the member is calculated according to the following formula:

$$S = EA / l \tag{16}$$

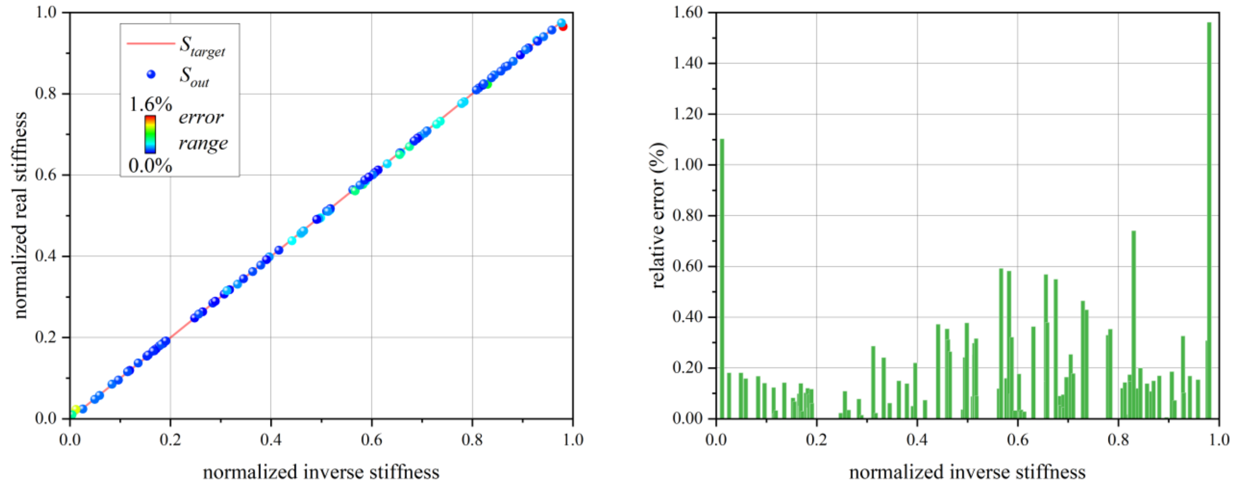
The relative error is calculated according to the formula:

$$RE = \left| S_{target} - S_{inverse} \right| \times 100\% \tag{17}$$



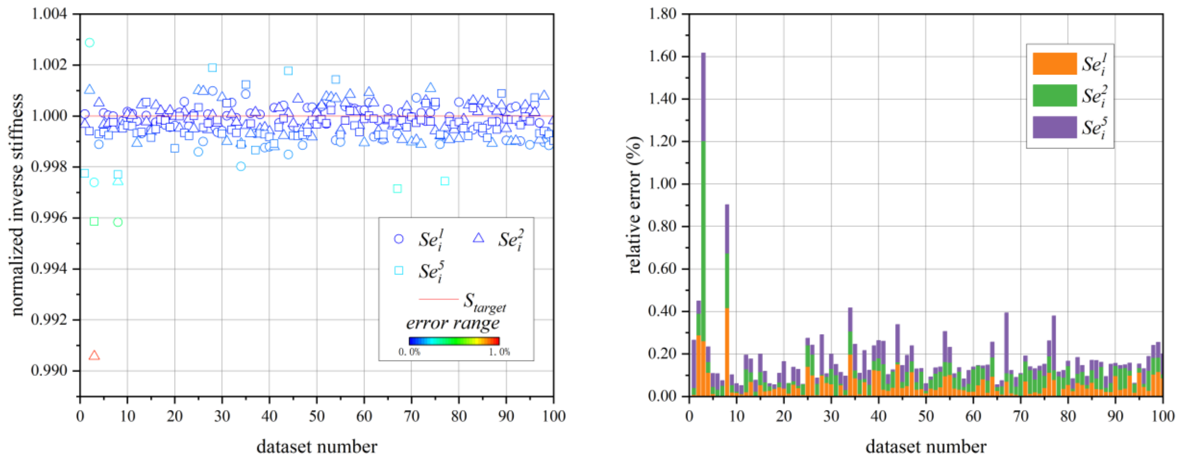
(a) Inverse results b) Inverse error

Fig. 10 Inversion results of member *Se_i^3*



(a) Inverse results b) Inverse error

Fig. 11 Inversion results of member Se_i^4



(a) Inverse results b) Inverse error

Fig. 12 Inversion results of other members

As can be seen in Fig. 10(a) and Fig. 11(a), the red line represents the normalized target value, the ball represents the normalized inversion output, and the colors of the ball represent the error range between the output and the target. The relative error in Fig. 10(b) and Fig. 11(b) is obtained from Equation (16). In Fig. 12(a), the inversion results of 100 datasets are indicated by different symbols and colors. The colors represent the error range of the inversion results. In Fig. 12(b), the relative error and error accumulation results of the inversion are illustrated.

It can be seen in Fig.10(a) and Fig.11(a) that the inverse stiffness results are almost in a straight line with the real stiffness, and the relative error is basically within 1%, which proves that the model achieves the stiffness inversion of the truss structure with good results. As can be seen in Fig. 10(b) and Fig. 11(b), the overall inversion error of the inversion model is very small, with only a few outliers with relative errors greater than 1%, which occurs when the member stiffness values close to 0 or 1. In addition, the inversion accuracy of the model is significantly higher in the moderate stiffness (between 0 and 1), which commonly represent semi-rigid in structure analysis. Since most of the member stiffness reduced due to bolt slip in the actual engineering structure is in the range of 0.3-0.5[11], this inversion model will perform better in practical engineering applications.

As can be seen in Fig.12, the inversion errors of most of the other members are within 0.2%, which is much smaller than the inversion errors of members Se_i^3 and Se_i^4. It proves that the inversion of the fixed stiffness parameter is better than that of the variable stiffness parameter. However, there are also outliers in the stiffness inversion results, which are caused by the extreme stiffness of members Se_i^3 and Se_i^4.

In conclusion, the inversion model based on GNN can complete the inversion of the uncertain stiffness parameters of the truss structure with high precision, which meets the requirement of engineering accuracy. Meanwhile, the presented model performs better when member stiffness is moderate in actual structural parameters.

4.2. Data requirements of the inversion model

In many engineering problems, there is a lack of experimental data due to technical or economic reasons. Therefore, it is necessary to reduce the number of datasets needed for training and to improve the generality of the model. For this purpose, training datasets of size N=100,200,400,800 are studied for model training. The 100 sets of displacement-stiffness datasets are randomly obtained as the validation datasets. During the training process, the inversion accuracy of the validation set with different datasets sizes is shown in Fig. 13. The inversion error of the model trained over 200 epochs for different datasets is shown in Table 4.

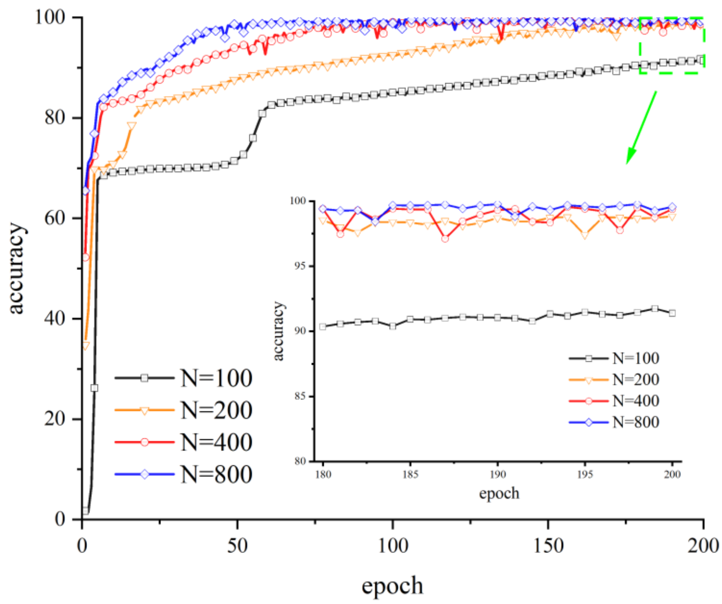


Fig. 13 The inversion accuracy of different training dataset sizes

Table 4
The inversion error for different datasets

training data sizes	training	test	validation
N=100	5.11%	8.86%	8.60%
N=200	2.06%	1.10%	1.17%
N=400	0.66%	0.63%	0.62%
N=800	0.74%	0.44%	0.43%

It can be seen in Fig. 13 that the inversion accuracy curves for different training dataset sizes increase rapidly within the first 50 epochs, which indicates that the inversion method is effective for different training dataset sizes. As the size of datasets increases, the accuracy can be improved in stages. Therefore, the more data, the faster the training speed, and the better the performance. It also can be found that the size of the training datasets has less impact on the degree of accuracy improvement when the accuracy is higher than 99%. For example, when the dataset size increases from 100 to 200, the inversion error decreases from 8.60% to 1.17%. When the size of the dataset increases from 200 to 400, the inversion error only decreases from 1.17% to 0.62%. The relationship between inversion error (IE) and datasets sizes N is approximately $IE \propto N^{-1.4}$. Therefore, a suitable dataset size can be obtained by pre-training to avoid wasting resources due to too little training data or too much data in the project.

As can be seen in Table 4, when N=100, the training data set is insufficient in this simple truss structure. The inversion error of the test set and validation set is larger than the training set, and the inversion error is around 10%. However, when N=200, the inversion error of the test and validation sets is smaller than the training set, which indicates that the model has learned the mapping relationship of truss structure inversion. Compared with N=400 and N=800, the inversion error of the validation set decreases from 0.62% to 0.43%, and the inversion performance of the model does not improve significantly, but the training time increases and resources are wasted. In summary, the inversion method of uncertain parameters for truss structure based on GNN requires a smaller dataset size and is more suitable for difficult data acquisition in engineering.

4.3. Integration of inversion method with engineering applications

There are various uncertain factors in practical engineering. In order to accurately analyze and solve engineering problems, various uncertainties need to be considered. It will become very difficult to solve the inverse problem by traditional analysis methods with the increase of uncertain factors. While these problems can be easily solved by the inversion method proposed in this paper. This is because all uncertainty factors can correspond to the features of the graph structure. For example, the uncertainty of the structure itself corresponds to the node features and edge features, while the external uncertainty factors correspond to the graph features. In the inversion method of uncertain parameter based GNN, the study of multiple uncertainties corresponds to the study of the mapping relationship of multiple features. To verify the feasibility of simultaneous inversion of multiple uncertain parameters, uncertain axial force features were added to the original graphical neural network model. It is shown that the model can realize the simultaneous inversion of nodal displacements on axial force and stiffness parameters, but the inversion accuracy will decline. It will be a further research direction to solve the inversion problem of multiple uncertain factors in specific engineering problems.

On the other hand, the actual engineering structure is more complex than the simple truss structure studied in this paper. The more complex the structure is, the more difficult it is to analyze by traditional finite element methods. However, this problem can be perfectly solved by the inversion method, due to the powerful ability of GNN to process graph data. In GNN models, complex structures only increase the number of nodes and edges, corresponding to increasing the dimension of node features and edge features. Therefore, the inversion method can be applied to all kinds of complex space structures.

5. Conclusions

The inverse problem in engineering has been extensively studied, but remains challenging to fully solve. However, with the integration of machine learning and engineering, the GNN method offers a promising solution to the inverse problem. The inversion method proposed in this paper not only allows for easy integration of the network model with the structural model but also significantly improves the accuracy and efficiency of the inversion of uncertain parameters. The main conclusions are as follows:

(1) The feature encoding method for converting truss structure to graph representation of GNN is defined. The connection relations, dimensional parameters, material properties and constraints of the truss structure are encoded as graph feature expressions, and the mapping conversion between truss structural analysis and GNN graph representation is realized.

(2) The inversion model proposed in this study effectively identified the uncertain parameters of the truss structure, which considers the effect of bolt slip on the uncertain stiffness parameters and uses the structural displacement response for inversion. The accuracy can meet the requirement of engineering.

(3) The inversion method based on GNN requires fewer data, which is suitable for the characteristics of difficult data acquisition in engineering, and it also has good inversion performance with the limited data set.

(4) The inversion method has great potential for application in engineering analysis and further research value in the combination of engineering analysis, which will become a new direction for exploring engineering structural analysis methods.

Acknowledgement

This study has been supported by the "Three Three Three Talent Project" in Hebei Province(C20231056).

Reference

[1] Gallet A., Rigby S., Tallman T. N., et.al., “Structural engineering from an inverse problems perspective”, Proceedings. Mathematical, Physical and Engineering Sciences, 2022, Vol. 478, No. 2257, pp. 1-37.
[2] Zhao Z. W., Liu H. Q., Liang B., et.al., “Influence of random geometrical imperfection on the stability of single-layer reticulated domes with semi-rigid connection”, Advanced Steel Construction, 2019, Vol. 15, No. 1, pp. 93-99.
[3] Srinivasan K., Selvakumar K., “Comparative study of pre-engineered building and space truss building with different span”, Indian Journal of Engineering and Materials Sciences, 2022, Vol. 29, No. 4, pp. 428-431.
[4] Fairclough H. E., Gilbert M., Pichugin A. V., et.al., “Theoretically optimal forms for very long-span bridges under gravity loading”, Proceedings. Mathematical, Physical and Engineering Sciences, 2018, Vol. 474, No. 2217, pp. 1-21.
[5] He G., Shao X. D., Chen Y. B., et.al., “Preliminary design of a steel-UHPFRC composite truss arch bridge and model tests of K-joints”, Journal of Bridge Engineering, 2022, Vol. 27, No. 10, 04022090.
[6] Grzywinski M., “Optimization of spatial truss towers based on Rao algorithms”, Structural Engineering and Mechanics, 2022, Vol. 81, No. 3, pp. 367-378.
[7] Jiang W. Q., Liu Y. P., Chan S. L., Wang Z. Q., “Direct analysis of an ultrahigh-voltage lattice transmission tower considering joint effects”, Journal of Structural Engineering, 2017, Vol. 143, No. 5, pp.04017009.1-04017009.14.
[8] Wang F. Y., Xu Y. L., Zhan S., “Concurrent multi-scale modeling of a transmission tower structure and its experimental verification”, Advanced Steel Construction, 2017, Vol. 13, No.3, pp. 258-272.
[9] Hussain A., Liu Y. P., Chan S. L., “Finite element modeling and design of single angle member under bi-axial bending”, Structures, 2018, Vol. 16, pp. 373-389.
[10] Chan S. L., Fong M., “Experimental and analytical investigations of steel and composite trusses”, Advanced Steel Construction, 2011, Vol. 7, No. 1, pp. 17-26.
[11] An L. Q., Wu J., Jiang W. Q., “Experimental and numerical study of the axial stiffness of bolted joints in steel lattice transmission tower legs”, Engineering Structures, 2019, Vol. 187, pp. 490-503.
[12] Li J. X., Cheng J. P., Zhang C., et.al., “Seismic response study of a steel lattice transmission tower considering the hysteresis characteristics of bolt joint slippage”, Engineering Structures, 2023, Vol. 281, 115754.1-115754.15.
[13] Habeck M., “Bayesian approach to inverse statistical mechanics”, Physical Review E, 2014, Vol. 89, No. 5, pp. 052113.1-052113.7.
[14] Adler J., Öktem O., “Solving ill-posed inverse problems using iterative deep neural networks”, Inverse Problems, 2017, Vol. 33, 124007.
[15] Kamyab S., Azimifar Z., Sabzi R., Fieguth P., “Deep learning methods for inverse problems”, PeerJ. Computer Science, 2022, Vol. 8, e951.
[16] Han X., Liu J., Chen J. L., “A manifold learning method for inverse problems with structural multi-source uncertainties”, Chinese Journal of Computational Mechanics, 2021, Vol. 38, No. 4, pp. 523-530.
[17] Keshavarzzadeh V., Kirby R.M., Narayan A., “Variational inference for nonlinear inverse problems via neural net kernels: Comparison to Bayesian neural networks, application to topology optimization”, Computer Methods in Applied Mechanics and Engineering, 2022, Vol. 400, pp.1-35.
[18] Jagtap A.D., Mao Z.P., Adams N., Karniadakis G.E., “Physics-informed neural networks for inverse problems in supersonic flows”, Journal of Computational Physics, 2022, Vol. 466, pp. 1-19.
[19] Raissi M., Perdikaris P., Karniadakis G.E., “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations”, Journal of Computational Physics, 2019, Vol. 378, pp. 686-707.
[20] Wang C., Fan J. S., “A general intelligent computation framework for structural responses in civil engineering based on deep learning”, Journal of Building Structures, 2023, Vol. 44, No. 1, pp. 259-268.
[21] Battaglia P. W., Hamrick J. B., Bapst V., et.al., “Relational inductive biases, deep learning, and graph networks”,2018, ArXiv Preprint ArXiv: 1806.01261.
[22] Fout A., Byrd J., Shariat B., et.al., “Protein interface prediction using graph convolutional networks”, In Proc. of NIPS, 2017, Vol. 30, pp. 6530-6539.
[23] Hamaguchi T., Oiwa H., Shimbo M., et.al., “Knowledge transfer for out-of-knowledge-base entities: A graph neural network approach”, In Proc. of IJCAI, 2017, pp. 1802-1808.
[24] Battaglia P., Pascanu R., Lai M., et.al., “Interaction networks for learning about objects, relations and physics”, In Proc. of NIPS, 2016, Vol. 29, pp. 4502-4510.
[25] Sanchez A., Heess N., Springenberg J. T., et.al., “Graph networks as learnable physics engines for inference and control”, In Proc. of ICLR, 2018, Vol. 80, pp. 4467-4476.
[26] Zhao Z. W., Liang B., Liu H. Q., Li Y. J., “Simplified numerical model for high-strength bolted connections. Engineering Structures”, 2018, Vol. 164, pp. 119-127.
[27] Wang D., Zhang Z. S., “A four-parameter model for nonlinear stiffness of a bolted joint with non-Gaussian surfaces”, Acta Mechanica, 2020, Vol. 231, No. 5, pp. 1963-1976.
[28] Li C. F., Jiang Y. L., Qiao R. H., Miao X. Y., “Modeling and parameters identification of the connection interface of bolted joints based on an improved micro-slip model”, Mechanical Systems and Signal Processing, 2021, Vol. 153, pp. 1-16.
[29] Jiang W. Q., Wang Z. Q., McClure G., et.al., “Accurate modeling of joint effects in lattice transmission towers”, Engineer Structure, 2011, Vol. 33, No. 5, pp. 1817–27.