ISSN 0554-6397
Izvorni znanstveni članak
(*Original scientific paper*)

Yuecheng Cai
E-mail: ycai05@mail.ubc.ca
Jasmin Jelovica
E-mail: jasmin.jelovica@ubc.ca
The University of British Columbia, 6250 Applied Science Ln #2054, Vancouver, BC V6T 1Z4

Graph Neural Network for Stress Predictions in Stiffened Panels

Abstract

Graph neural network (GNN) is a particular type of neural network which processes data that can be represented as graphs. This allows for efficient representation of complex geometries that can change during conceptual design of a structure or a product, such as ship structures, replacing computationally expensive finite element analysis (FEA) in optimization. In this study, we demonstrate how GNN can be used to predict stress distributions in stiffened panels with varying geometries under patch loading, for which we use Graph Sampling and Aggregation (GraphSAGE) network. Parametric study is performed to examine the effect of structural geometry on the prediction performance. Our results demonstrate the immense potential of graph neural networks with the proposed graph embedding method as robust reduced-order models for 3D structures.

Keywords: Machine learning, Deep learning, Graph neural networks, Stiffened panels, Structural analysis

1. Introduction

The progress in developing efficient modern structures is significantly driven by advancements in structural analysis methods such as Finite Element Analysis (FEA) [1]. However, FEA for large, complex structures such as ships, bridges and aircraft is computationally costly. This is especially troublesome in optimization, where numerous iterations need to be performed for both sizing [2] and topology optimization [3]. In order to overcome this problem, traditional reduced-order models (ROMs) have been used, such as Multivariate Adaptive Regression Splines (MARS), Kriging (KRG), Radial Basis Functions (RBF), and Response Surface Method (RSM) [4]. These methods aim to maintain the precision of high-fidelity models while having a relatively low computational cost.

However, applying the traditional ROMs to complicated engineering problems is limited due to their inherent assumptions. Moreover, they may lose fidelity when the structure changes geometrically. More recently, with the advancements in machine learning (ML) and in particular deep learning (DL) methodologies, there is a tendency to adopt ML/DL models as ROMs. Particularly in mechanical engineering, ML/DL techniques are effective modelling tools and approximators that often exceed conventional ROM techniques in accuracy and capacity to represent even nonlinearities of engineering problems [5].

Typical ROMs require a structure to be represented parametrically, where the structural variables are identified as inputs to the ML model. The typical technique for reduced-order modelling is artificial neural networks (ANNs). One of the most commonly used is multi-layer perceptron, also known as MLP, which has been widely implemented in various fields, as evidenced by numerous scholarly works. However, MLPs are inefficient at accurately describing complicated structural behaviour since they demand a large amount of training data and computational resources. In addition, MLPs tend to overfit, and they are less interpretable than other types of neural network (NN) approaches, therefore its capabilities are limited and not suited for advanced problems.

To capture more complex features, some researchers have used more advanced NNs such as convolutional neural networks (CNNs) as ROMs for the structures that can be represented as images (2D matrices) or composition of images (3D matrices). Significant amount of work has been done demonstrating benefits and extending the CNN approaches in fluid dynamics, e.g. [6, 7]. In addition, researchers utilized CNN for stress predictions in different structures, for instance, the maximum stress of brittle materials [8], and stress contour in components of civil engineering structures [9]. However, modelling engineering structures using fixed-size vectors or matrices, such as images, proves to be challenging, considering that one of the variables is structural geometry. It introduces a dimensional change in the design input, which is primarily handled by re-training for NN approaches such as MLP and CNN. Furthermore, stiffened panels are discrete structures, which consist of repeated structural units, e.g., stiffeners and plates, whose connection can not be neglected as they affect the mechanical response of the structure. Motivated by these two reasons, this paper is based on an approach that transforms structural models into graphs. This transformation permits flexibility in varying the dimension of design inputs, a characteristic that aligns well with the capabilities of graph neural networks (GNNs).

Since GNN just gained significant attention in recent years, not many advancements have been made in the structural engineering field. In a recent study by Zheng et al. [10], a graph embedding approach was employed to represent 2D and 3D trusses as graphs, with vertices denoting the joints and edges representing the bars. Similarly, other recent investigations applying GNNs to truss-related problems can be found in references such as [11] and [12]. These studies employed GNNs in conjunction with various techniques, including Q-learning and transfer learning, to address distinct objectives. Nevertheless, there has been a notable absence of research dedicated to the application of GNNs to structures beyond the scope of truss problems.

In this paper, we demonstrate applicability of graph neural networks for structural analysis of 3D stiffened panels. These panels are used to construct bridges, ships, offshore platforms and other important structures, and as such need to be optimized during design, which comes with a high computational cost if FEA is used for structural analysis. Once GNN model is trained, it could be used in optimization instead of FEA, similarly to other reduced-order models [13-16]. In this article we use Graph Sampling and Aggregation network (GraphSAGE). We conduct a comprehensive parametric study for various structural geometry parameters under patch loading.

2. Methodology

2.1. Basis of graph neural networks

Graph Neural Networks (GNNs) represent a specialized subset of neural networks (NNs) renowned for their capacity to handle data with graph embeddings. Unlike the CNN, which is typically used for tasks involving data that is defined on a regular grid, such as images, GNNs are designed to process and analyze data represented in the form of graphs, such as data with complex relationships between entities or data that has a natural representation as a network.

In general, a graph can be defined as $G = (V,E,A)$, where V represents the set of vertices, E indicates the set of edges between these vertices, and A is the adjacency matrix. We denote the edge going from vertex v_i to vertex v_j as $e_{ij} = (v_i, v_j) \in E$. If a graph is undirected, every two vertices contain two edges $e_{ij} = (v_i, v_j) \in E$, and $e_{ji} = (v_j, v_i) \in E$. The adjacency matrix $A \in \mathbb{R}^{N \times N}$ is a convenient way to represent the graph structure, where $N = |A|$ is the number of vertices, $A_{ij} = 1$ if $e_{ij} \in E$. For an undirected graph, $A_{ij} = A_{ji}$. Therefore, a graph is associated with graph attributes $X \in \mathbb{R}^{N \times D}$ and $A \in \mathbb{R}^{N \times N}$, where D is the number of input features of each vertex. It is worth mentioning that in this study, all graphs are defaulted as undirected graphs.

To conduct convolution on a graph, researchers developed several techniques. In this study we employ the GraphSAGE network [17], which is a prominent and benchmark method for many graph-related tasks. To preserve the maximum information of each vertex, the 'sum' operator is determined as the aggregation function [18].

A general structure of the employed GraphSAGE network can be found in Figure 1. Structural geometric data and external loading are initially transformed into the proposed graph representation, as detailed in Section 2.2. At each layer, the GraphSAGE operator is employed to process and learn the features of the graph. Batch normalization has been utilized after each GraphSAGE convolution layer to stabilize the training process of the GNN. Mean square error (MSE) is adopted as the loss function in this study. The hyperparameters for the utilized model have been fine-tuned and are detailed in Table 1.

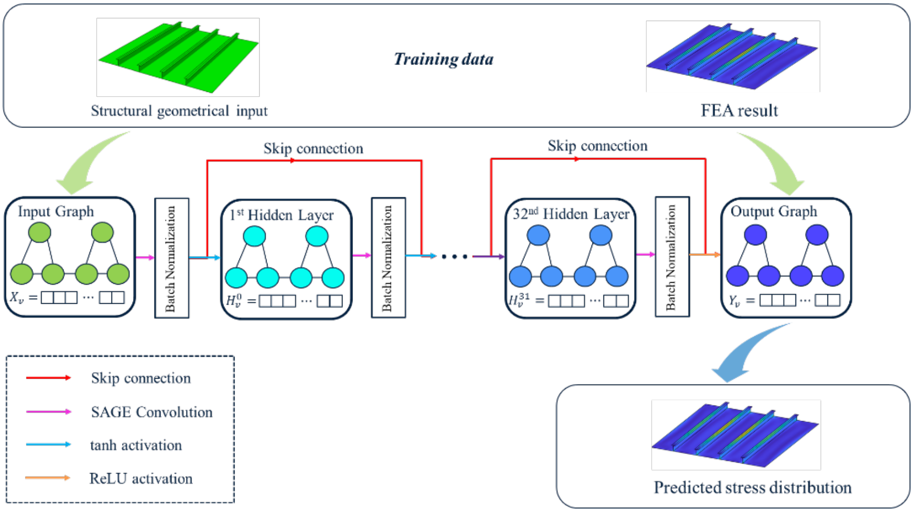


Figure 1. Architecture of the GraphSAGE network. Each vertex represents a finite element of the stiffened panel. Input graph contains information about structural geometry, boundary conditions and loading. Vertices in output graph contain the stress information at a particular location.

Table 1: Hyperparameter settings of the employed GraphSAGE network.

Category	Value
Number of layers	32
Number of hidden neurons for each layer	64
Activation function	tanh
Optimizer	Adam
Learning rate	0.02
Batch size	512
L2 regularization factor	1e-4

2.2. Graph embedding for stiffened panels

As indicated in the previous section, graph embedding is the prerequisite for a structure being handled by a GNN. In this study, we use a simple approach where each finite element is represented as a vertex (node) in graph neural network. Thus, the number of nodes in a graph is equal to the number of finite elements in the model. This is easy to define and implement. Each vertex incorporates geometrical information of the element and its boundary conditions. For each vertex, we employ ten variables, including structural element width, length, thickness, and boundary conditions for each edge, together with the position and value of the applied pressure. The connectivity between each vertex is encoded by the adjacency matrix and is not reflected in the vertex input embedding.

The objective of this study is to predict the von Mises stress distribution across stiffened panels, which is crucial for the structural design of ships and other complex structures made of such structural units. Thus, the output of each vertex is the von Mises stress of the corresponding finite element, which is a scalar. For plate and flange, stress at the surface with lower z-value is considered. For the web, stress at the surface with higher y-value is considered. Coordinate system is illustrated in Figure 2. For a balance between training time and FE model accuracy, we have allocated 15 elements between each stiffener, using square elements. 15 elements were used along the stiffener web, and 8 elements for the flange width.

3. Data preparation

The case study is a stiffened panel because it represents the basic repeating unit of many large-scale structures. To approximate the stiffened panel of real-life structures, the span of the panel has been set as 3m 3m. Main plate thickness ranges from 10 mm to 20 mm. Each panel contains 2 to 8 stiffeners, with a random height from 0.1 m to 0.4 m. All stiffeners have a T-shaped cross-section, with a rectangular flange whose width ranges from 0.05 m to 0.15 m. The thickness for both stiffener web and flange can change from 5 mm to 20 mm. We allow the stiffener/flange height and thickness to continuously vary in this range, which allows a wider design space. We assume that the plate of the panel is subjected to a patch loading, which ranges from 0.2 MPa to 0.3 MPa. All edges are clamped. The summary of the upper and lower limits for the geometrical details can be found in Table 2.

Table 2: Lower and upper limits of stiffened panel geometrical variables.

Category	Lower limit	Upper limit	Unit
Plate thickness	10	20	mm
Stiffener thickness	5	20	mm
Stiffener height	100	400	mm
Flange thickness	5	20	mm
Flange width	50	150	mm
Number of stiffeners	2	8	-

We prepare a total of 2,000 randomly generated design configurations, allocating 80% of them for training and 20% for validation. The dataset is obtained using parametric models prepared in MATLAB and executed through ABAQUS FEM software. The static analysis is performed on a model discretized using the 'SR4' element. The training procedure of GNN is executed using Pytorch Geometric and carried out on a computing device with a GTX 3090 GPU.

4. Results and discussion

In practical engineering scenarios, the distribution of external loads might not be uniform. For instance, cargo loading on a ship's deck can be imposed on a relatively small area. Thus, in this study, we employ a randomly positioned patch loading. The details of the geometrical variables can be found in Table 2.

In Figure 2, we present a comparison between the predicted stress distribution across the stiffened panel with the GraphSAGE network and the ground truth distribution obtained with finite element analysis (FEA). We have selected two test examples at random to illustrate the prediction accuracy of our model. Details of the two examples are given in Table 3. The 3D view of the structure is shown, where the contour shows the stress distribution and stress intensity across the panel. For each test example, the same color for both GraphSAGE prediction and FEA ground truth represents the same stress value. To better visualize the stress contour, the orientation of the stiffener webs and flanges is reversed for the second test example.

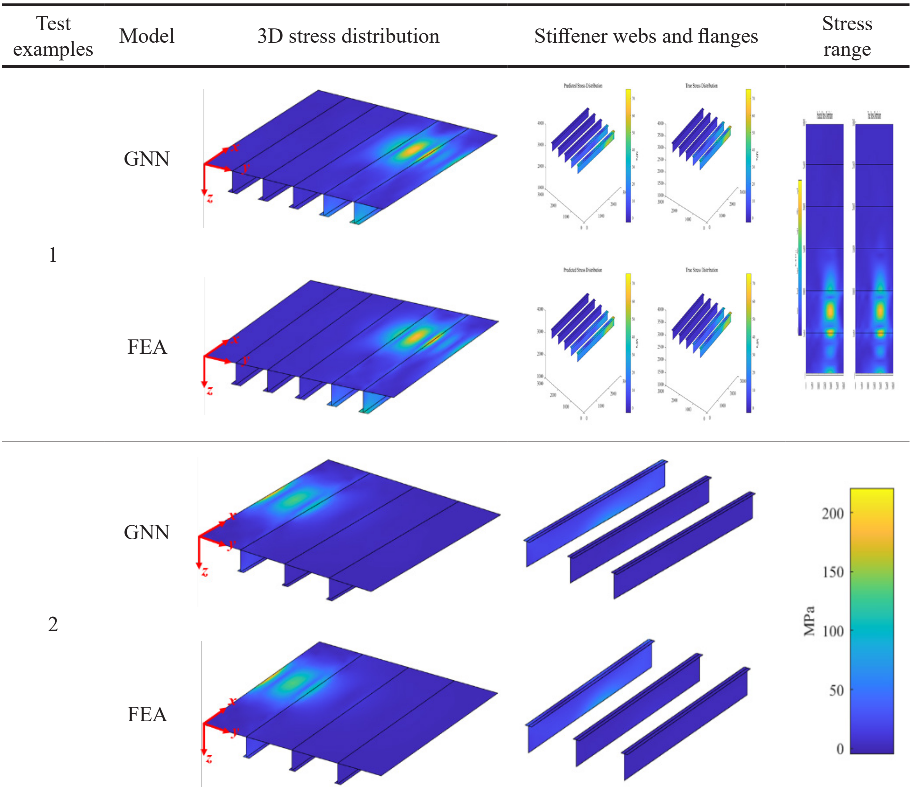
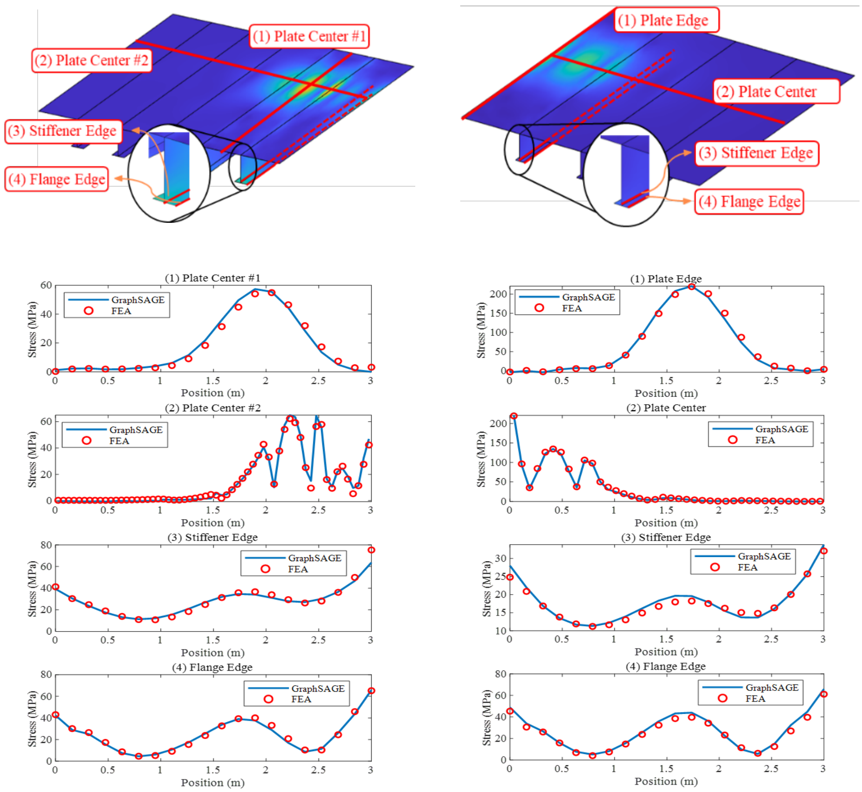


Figure 2. Comparison of GraphSAGE predictions and FEA results.

Table 3. Structural geometry details for test examples.

Category	Test Example 1	Test Example 2	Unit
Plate thickness	17.22	13.95	mm
Stiffener web thickness	5.88	8.58	mm
Stiffener web height	371.4	382.3	mm
Flange thickness	5.18	16.64	mm
Flange width	149.5	99.57	mm
Number of stiffeners	5	3	-
Patch loading amplitude	0.240	0.207	MPa
Patch loading position (x,y,z)	(1995,2450,0)	(1758,525,0)	mm



(a) Test example 1 (b) Test example 2

Figure 3. Comparison of stress distribution along specified paths for both test examples. For plate and flange, stress at the surface with lower z-value is considered. For the web, stress at the surface with higher y-value is considered.

Given the patch loading, stress primarily accumulates in the loading area and the neighboring stiffeners, specifically at the intersection of the plate and stiffener web for both test examples. In any case, the GraphSAGE model consistently captures the stress distribution of the structure, as demonstrated in Figure 2. To gain deeper insights into the discrepancies between GraphSAGE predictions with the FEA simulations, detailed stress comparisons for both test examples along specified paths are shown in Figure 3. In test example 1, the stress predictions for the stiffener web and flanges are very accurate, with the maximum error occurring at the center of the plate, which is 8.2 MPa. In test example 2, which features a thinner plate and fewer stiffeners, the maximum stress is concentrated at the plate edges. As depicted in Figure 3 (b), stress accumulates primarily at the edges of the plate, with an average prediction deviation of 4.4 MPa. The maximum stress has been accurately captured by the GraphSAGE model in this test case and exhibits an accuracy of 99.4%. For both test examples, the GraphSAGE model could effectively approximate the general trend of stress distribution within the center of the plate but not all the finer stress details, resulting in a prediction error of approximately 7.67% at the stiffener web and flanges.

5. Conclusion and future work

In this study, we demonstrate the potential of graph neural networks (GNN) as a promising avenue for developing a reduced-order model for stress prediction in stiffened panels. Parametric analysis is conducted to demonstrate the versatility and robustness of the employed model, by handling various geometric variations and loading intensity. The results indicate that the GNN is a viable reduced-order model for structural modelling and analysis in structural design, offering both accuracy and efficiency. Although significant amount of data is required to train the GNN, once training is completed, the model can be used in optimization to replace FEA, as was done previously with other reduced-order models, see e.g. [13, 15, 16]. The stiffened panel serves as a fundamental structure that aids in cultivating insights, thereby paving the way for the development of more sophisticated approaches utilizing GNNs to efficiently model more complex structures in the future.

6. Acknowledgements

This research was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC). This research was supported in part through computational resources and services provided by Advanced Research Computing at The University of British Columbia.

7. References

1. CLOUGH, R.W.: „Original formulation of the finite element method“. Finite elements in analysis and design 7(2), 89–101 (1990)
2. JELOVICA, J., CAI, Y.: „Improved multi-objective structural optimization with adaptive repair-based constraint handling“. Engineering Optimization, 1–20 (2022)
3. CHU, S., FEATHERSTON, C., KIM, H.A.: „Design of stiffened panels for stress and buck- ling via topology optimization“. Structural and Multidisciplinary Optimization 64, 3123–3146 (2021)
4. CHEN, V.C., TSUI, K.-L., BARTON, R.R., MECKESHEIMER, M.: „A review on design, modeling and applications of computer experiments“. IIE transactions 38(4), 273– 291 (2006)
5. MAI, H.T., LIEU, Q.X., KANG, J., LEE, J.: „A robust unsupervised neural net- work framework for geometrically nonlinear analysis of inelastic truss structures“. Applied Mathematical Modelling 107, 332–352 (2022)
6. MALLIK, W., JAIMAN, R.K., JELOVICA, J.: „Predicting transmission loss in underwater acoustics using convolutional recurrent autoencoder network“, The Journal of the Acoustical

Society of America, Vol. 152, 1627, 2022
7. MALLIK, W., JELOVICA, J., JAIMAN, R.K.: „Deep neural network for learning wave scattering and interference of underwater acoustics“, 36, 017137, 2024.
8. WANG, Y., OYEN, D., GUO, W., MEHTA, A., SCOTT, C.B., PANDA, N., FERNANDEZ-GODINO, M.G., SRINIVASAN, G., YUE, X.: „Stressnet-deep learning to predict stress with fracture propagation in brittle materials“. npj Materials Degradation 5(1), 6 (2021)
9. BOLANDI, H., LI, X., SALEM, T., BODDETI, V.N., LAJNEF, N.: „Deep learning paradigm for prediction of stress distribution in damaged structural components with stress concentrations“. Advances in Engineering Software 173, 103240 (2022)
10. ZHENG, S., QIU, L., LAN, F.: Tso-gcn: „A graph convolutional network approach for real-time and generalizable truss structural optimization“. Applied Soft Computing 134, 110015 (2023)
11. BACCIU, D., ERRICA, F., MICHELI, A., PODDA, M.: „A gentle introduction to deep learning for graphs“. Neural Networks 129, 203–221 (2020)
12. WHALEN, E., MUELLER, C.: „Toward reusable surrogate models: Graph-based transfer learning on trusses“. Journal of Mechanical Design 144(2), 021704 (2022)
13. MALLIK, W., FARVOLDEN, N., JELOVICA, J., JAIMAN, R.K.: „Deep convolutional neural network for shape optimization using level-set approach“. arXiv:2201.06210, January 2022.
14. KLANAC, A., EHLERS, S., JELOVICA, J.: „Optimization of crashworthy marine structures“. Marine Structures, Vol. 22(4), p. 670-90, 2009, https://doi.org/10.1016/j.marstruc.2009.06.002
15. CAI, Y., JELOVICA, J.: „Neural network-enabled discovery of mapping between variables and constraints for improved repair-based constraint handling in multi-objective structural optimization“. Knowledge-Based Systems, Vol. 280, 111032, 2023
16. CAI, Y., JELOVICA, J.: „Structural optimization of ships: Benchmark study of metaheuristic algorithms and constraint handling approaches“. Proc. 41th International Conference on Ocean, Offshore and Arctic Engineering (OMAE), Hamburg, Germany, June 2022
17. HAMILTON, W., YING, Z., LESKOVEC, J.: „Inductive representation learning on large graphs“. Advances in neural information processing systems 30 (2017)
18. XU, K., HU, W., LESKOVEC, J., JEGELKA, S.: „How powerful are graph neural networks?“ arXiv preprint arXiv:1810.00826 (2018)