14th International Conference on Applications of Statistics and Probability in Civil Engineering, ICASP14
Dublin, Ireland, July 9-13, 2023

Near-real-time Identification of Seismic Damage by Graph Neural Network based on Structural Modes

Minkyu Kim
Ph.D. candidate, Dept. of Civil and Environmental Engineering, Seoul National University, Seoul, Republic of Korea

Junho Song
Professor, Dept. of Civil and Environmental Engineering, Seoul National University, Seoul, Republic of Korea

ABSTRACT: This paper proposes a near-real-time damage identification method based on the graph neural network (GNN) using the structural response data recorded during an earthquake event. The proposed method features a structural-mode-based weighted adjacency matrix to enable the GNN model to learn the spatial correlation and structural characteristics. The GNN model has an autoencoder architecture, one of the self-supervised deep neural networks that can detect anomalies in the input data by extracting important latent variables. The proposed method consists of 'encoder', 'graph structure decoder', and 'node feature decoder' that respectively learn latent variables of input data considering the spatial correlation, structural characteristics of the graph by reconstructing the adjacency matrix, and vibrational characteristics by reconstructing the response data. The GNN model is trained using the simulated structural responses and the structural-mode-based adjacency matrix of the target structure in a healthy state. The seismic damage of each member is then identified by the structural damage index calculated based on the reconstruction errors. As a numerical investigation, the proposed method is applied to two- and three-dimensional steel frame structures. Structural analyses are performed using ground motions from the PEER-NGA strong motion database to create the train, validation, and test datasets. The proposed method is verified by near-real-time simulations using the test dataset. The results demonstrate that the proposed GNN-based method can identify seismic damage accurately in near-real-time. The proposed method under various load conditions is expected to help reduce the time required for the post-disaster decision-making process by providing near-real-time damage identification.

1. INTRODUCTION

Civil infrastructure systems are subjected to external excitations and harsh environmental changes throughout their lifetime, which can introduce local or global structural damage. The failure to identify such damage potentially leads to the loss of structural integrity and catastrophic system failure, i.e., the violation of the requirements for functionality and safety. Therefore, it is essential to implement a proper post-disaster inspection process that can assess the integrity of the structural system and detect potential damage. Structural Health Monitoring (SHM), utilizing numerous sensor data and measurements, has arisen as an alternative to traditional inspection methods. SHM can handle a variety of important tasks related to the infrastructure systems, including post-disaster damage identification.

Recently, pattern recognition methods combined with vibration-based damage identification are being considered suitable particularly for the short-term SHM, which aims to rapidly identify structural condition changes and provide information on the structural integrity in near-real-time (Dawson 1976). This is because such methods can utilize the pre-trained model, such as a Deep Neural Network (DNN), using the latent features extracted from the vibration signals (Pathirage et al. 2018). Therefore, it requires low computational effort to recognize the changes in vibrational characteristics.

In practice, since it is challenging to measure the vibration signals at many points of the structure, one needs to cover the target structure by the vibration signals from multiple sensors at important points of the structure. Therefore, the vibration signals obtained from each sensor is determined based on spatial information of the sensor network, and structural information including the geometrical and physical characteristics of the target structure. However, traditional DNNs, such as a one-dimensional convolutional neural network (1DCNN), cannot fully leverage this spatial and structural information; thus, identifying damage using only the raw vibration signals is generally challenging because of its insensitivity to structural damage.

To address these issues, this paper proposes to model the sensor network as a graph and adopts a graph neural network (GNN; Scarselli et al. 2008), which is specialized in learning the features of the graph-structured data and can handle the structural information using its graph theory foundation. A graph convolutional network (GCN; Kipf and Welling 2016a), which is one of the GNN models based on the convolution theory, is utilized in this paper to learn the local features of the sensor network. In the graph-structured sensor network, each sensor and vibration signal represent a node and node feature of the graph, respectively. Each node is connected with an edge according to the geometrical characteristic of the structure, and an adjacency matrix provides information about the connectivity of the sensor network.

The traditional adjacency matrix, however, is difficult to provide accurate information about the spatial correlation between sensors since it considers every connectivity between sensors with the same weight. To handle this, this paper proposes a method to construct the weighted adjacency matrix based on the difference in mode shape between connected nodes. Thereby, the GNN model can consider the spatial correlation as well as the connectivity between sensors.

Based on these considerations, this paper proposes a near-real-time damage identification method based on the GNN with the structural-mode-based weighted adjacency matrix, using the structural response data recorded during an earthquake event. The GNN model has an autoencoder architecture, one of the self-supervised deep neural networks that can detect anomalies in the input data by extracting important latent variables (Goodfellow et al. 2016). The proposed method features the following three main parts:

1. An ‘encoder’ that consists of multiple GCN layers to learn latent variables of the graph-structured input data considering the spatial correlation
2. A ‘graph structure decoder’ that learns spatial characteristics of the sensor network by reconstructing the adjacency matrix
3. A ‘node feature decoder’ that consists of multiple GCN and 1DCNN layers to learn vibrational characteristics by reconstructing the response data

The loss function to train the GNN model is defined based on the reconstruction errors, and also utilized as a structural damage index (SDI). The GNN model is trained using the simulated structural responses and the adjacency matrix of the target structure in a healthy state. After training the GNN model, the seismic damage of each member is then identified in near-real-time by calculating the SDI at every time step. The proposed framework is demonstrated and tested by numerical examples of two- and three-dimensional steel frame structures.

2. THEORETICAL BACKGROUND

2.1. Graph convolutional network

A graph convolutional network (GCN) is one of graph neural network (GNN) models designed to handle a graph-structured data and leverage its structural information (Kipf and Welling 2016a). GCN is specialized in learning local features of the graph based on its convolution theory foundation. The input of GCN is a graph $\mathbf{G} =$

$(\mathbf{V}, \mathbf{E})$, where $\mathbf{V}$ and $\mathbf{E}$ are the sets of nodes and edges, respectively. $\mathbf{V}$ is represented as a node feature matrix $\mathbf{X} \in \mathbb{R}^{N \times F}$, composed of the node feature vectors of a length F associated with each of N nodes. The structural information of the graph is represented as an adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ in which element $A_{ij} = 1$ if node i and j are connected, otherwise $A_{ij} = 0$.

Let $\mathbf{H}^{(l)} \in \mathbb{R}^{N \times M}$ denote a node-level hidden feature matrix at layer l, where K is the number of hidden features of each node; and $\mathbf{H}^{(0)} = \mathbf{X}$. The hidden feature matrix of the next layer $l + 1$, $\mathbf{H}^{(l+1)} \in \mathbb{R}^{N \times K}$, can be defined as a function of $\mathbf{H}^{(l)}$ and $\mathbf{A}$ as

$$\mathbf{H}^{(l+1)} = \boldsymbol{f}(\mathbf{H}^{(l)}, \mathbf{A}) = \boldsymbol{\sigma}(\bar{\mathbf{A}}\mathbf{H}^{(l)}\mathbf{W}) \quad (1)$$

where $\mathbf{W} \in \mathbb{R}^{M \times K}$ denotes a trainable weight matrix of GCN model; and $\boldsymbol{\sigma}$ is an activation function, which is usually a nonlinear function such as sigmoid or rectified linear unit (ReLU). Note that a self-loop adjacency matrix $\bar{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ is used, instead of $\mathbf{A}$, to compute the hidden feature vector of each node with consideration of itself as well as its neighboring nodes, where $\mathbf{I} \in \mathbb{R}^{N \times N}$ is the identity matrix. To prevent the numerical instability and the gradient vanishing/exploding problem, the adjacency matrix needs to be normalized by computing $\bar{\mathbf{D}}^{-\frac{1}{2}}\bar{\mathbf{A}}\bar{\mathbf{D}}^{-\frac{1}{2}}$, where $\bar{\mathbf{D}}$ is a degree matrix defined as $\bar{D}_{ii} = \sum_j \bar{A}_{ij}$.

Considering the previous adjustments, the GCN layer can be rewritten as

$$\mathbf{H}^{(l+1)} = \boldsymbol{f}(\mathbf{H}^{(l)}, \mathbf{A}) = \boldsymbol{\sigma}(\bar{\mathbf{D}}^{-\frac{1}{2}}\bar{\mathbf{A}}\bar{\mathbf{D}}^{-\frac{1}{2}}\mathbf{H}^{(l)}\mathbf{W}) \quad (2)$$

It is possible to define a multi-layer GCN by feeding the output feature matrix of a layer together with the adjacency matrix $\mathbf{A}$ as the input for the next layer. GCN is trained by minimizing a loss function defined between its output and the expected output by means of the backpropagation algorithm. Since the loss function is generally difficult to minimize due to its non-linearity, gradient descent-based optimizers, such as Adam (Kingma and Ba, 2014), are commonly used.

2.2. Autoencoder

Autoencoder (AE) aims to learn patterns hidden in a set of data by finding a DNN structure that can reconstruct the input data by going through the mapping functions called encoder and decoder sequentially (Vincent et al. 2010).

Encoder: The mapping function $f(\mathbf{x})$, which transforms a d-dimensional input vector $\mathbf{x} \in \mathbb{R}^d$ into an r-dimensional latent variable $\mathbf{z} \in \mathbb{R}^r$, in which $r < d$, is called an encoder. $\boldsymbol{f}(\mathbf{x})$ is usually described as a nonlinear transformation

$$\mathbf{z} = \boldsymbol{f}(\mathbf{x}) = \boldsymbol{\sigma}(\mathbf{W}\mathbf{x} + \mathbf{b}) \quad (3)$$

where $\mathbf{W} \in \mathbb{R}^{r \times d}$ denotes the mapping weight matrix of the encoder; $\mathbf{b} \in \mathbb{R}^r$ is the bias vector; and $\boldsymbol{\sigma}$ is the activation function.

Decoder: The mapping function $\boldsymbol{g}(\mathbf{z})$, which transforms the latent variable $\mathbf{z}$ back into a reconstructed vector $\mathbf{x}' \in \mathbb{R}^d$, is called a decoder. Usually, $\boldsymbol{g}(\mathbf{z})$ is described as

$$\mathbf{x}' = \boldsymbol{g}(\mathbf{z}) = \boldsymbol{\sigma}(\widehat{\mathbf{W}}\mathbf{h} + \hat{\mathbf{b}}) \quad (4)$$

where $\widehat{\mathbf{W}} \in \mathbb{R}^{d \times r}$ denotes the mapping weight matrix of the decoder; $\hat{\mathbf{b}} \in \mathbb{R}^d$ is the bias vector; and σ is the activation function described above.

To obtain optimal estimates of the parameters of AE model, $\boldsymbol{\theta} = [\mathbf{W}, \mathbf{b}, \widehat{\mathbf{W}}, \hat{\mathbf{b}}]$ based on the training data, AE algorithms minimize the loss function $\mathcal{L}_{AE}(\mathbf{x})$, often defined as the mean squared error (MSE), i.e.,

$$\mathcal{L}_{AE}(\mathbf{x}) = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{2}\left\| x^{(i)} - \boldsymbol{g}\left(\boldsymbol{f}\left(x^{(i)}\right)\right)\right\|^2 \quad (5)$$

where m is the number of training samples; and $x^{(i)}$ is the i-th input.

3. PROPOSED GNN-BASED DAMAGE IDENTIFICATION METHOD

In this section, we propose a near-real-time GNN-based seismic damage identification method using the structural response data. The GNN model has an autoencoder architecture to detect anomalies in the input data by extracting important latent variables. The structural response data from each sensor is used as the node feature $\mathbf{X}$, while the adjacency matrix $\mathbf{A}$ of the sensor network is constructed according to the geometrical characteristic of the structure (Dang et al. 2022).

However, the adjacency matrix defined in Section 2 is difficult to provide information about the spatial correlation between sensors since it considers every connectivity between sensors with the same weight. To handle this, we first propose a method to construct the weighted adjacency matrix based on the structural mode shape.

3.1. Method to construct structural-mode-based adjacency matrix

As for the traditional adjacency matrix, one can assign values to each element according to the geometrical characteristic of the structure as stated in the last section, i.e., $A_{ij} = 1$ if node i and j are connected, otherwise $A_{ij} = 0$. As a simple example, Figure 1 shows how the traditional adjacency matrix is constructed for a structure. In the case of v_1, for instance, only A_{12} and A_{13} have a value of 1 since v_2 and v_3 are connected to v_1.

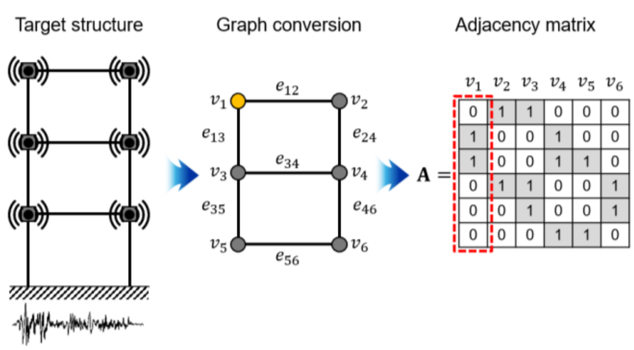

Figure 1: Example of constructing traditional adjacency matrix for a structure.

However, the adjacency matrix constructed using the traditional method cannot provide information about the spatial correlation between sensors effectively because it considers every connectivity between sensors with the same weight. In fact, vibration signals measured from the civil structure are significantly influenced by its structural characteristics such as topology and material property. For example, let us consider node 1 and its neighboring nodes in Figure 1. If it is assumed that only horizontal vibration signals are measured from each sensor, the correlation between v_1 and v_2 is greater than that between v_1 and v_3. This is because v_1 and v_2 have same vertical coordinates, and the horizontal stiffness of e_{12} is greater than that of e_{13}. Nonetheless, the traditional adjacency matrix considers them equally by the value of 1.

Every structural system can be represented by mode shapes and corresponding natural frequencies through the eigenvalue decomposition method. Namely, the structural system can be decomposed into spatial and temporal characteristics. Considering this, mode shapes of the target structure can be utilized to represent the structural characteristics, i.e., the larger the difference in mode shape between nodes, the lower the spatial correlation. In addition, it can be said that a modal participation factor quantifies the contribution of each mode to the response of the structural system. Generally, in terms of the modal participation factor, the first mode is dominant while that of a higher-order mode is lower.

Based on these considerations, the proposed modal-based weighted adjacency matrix is calculated as follows:

$$A_{ij} = \sum_m r_m \, e^{-|\bar{\phi}_{m,i} - \bar{\phi}_{m,j}|} \qquad (6)$$

where r_m denotes the modal participation factor of m-th mode; and $\bar{\phi}_{m,i}$ is the normalized mode shape of m-th mode at node i. Note that Eq. (6) is utilized if node i and j are connected, otherwise $A_{ij} = 0$. Thereby, the GNN model can consider the spatial correlation as well as the connectivity between sensors.

3.2. GNN-based damage identification method

In this section, we propose a GNN architecture based on AEs with GCN layers to detect structural damage, and a damage identification framework using the GNN model with the structural-mode-based weighted adjacency matrix.

3.2.1. GNN architecture

In general, AE models for damage detection are trained by using the vibration signals of the undamaged target structure as input data and detect damage through the outlier detection method. However, damage detection using only the raw vibration signals is challenging because they are insensitive to changes in structural characteristics, and AE models with traditional DNN layers cannot fully leverage the spatial and structural information of the sensor network.

Considering these issues, we propose a bifurcating GNN architecture, inspired by Ding et al. (2019), based on AEs with GCN layers for damage detection, as illustrated in Figure 2. The GNN model receives two type of input data: (1) the structural response data as the node feature $\mathbf{X}$; and (2) the adjacency matrix $\mathbf{A}$ of the sensor network. The GNN model then learns important latent variables and reconstructs the input data utilizing spatio-temporal information of the input data through the following three main parts:

1. **Encoder** that consists of multiple GCN layers to learn latent variable $\mathbf{Z}$ of input data considering the spatial correlation with the adjacency matrix $\mathbf{A}$ and the temporal information with the node feature matrix $\mathbf{X}$.
2. **Graph structure decoder** that learns spatial characteristics of the sensor network and reconstructs the original the adjacency matrix structure by

$$\widehat{\mathbf{A}} = sigmoid(\mathbf{Z}\mathbf{Z}^{\mathrm{T}}) \quad (7)$$

   where $\widehat{\mathbf{A}}$ denotes the reconstructed adjacency matrix. The graph structure reconstruction error $\mathbf{R}_S = \left\|\mathbf{A} - \widehat{\mathbf{A}}\right\|_2$ can be utilized to determine structural anomalies on the sensor network (Kipf and Welling 2016b).
3. **Node feature decoder** that consists of multiple GCN and 1DCNN layers to learn vibrational characteristics by reconstructing the node feature matrix $\mathbf{X}$. With the feature reconstruction error $\mathbf{R}_F = \left\|\mathbf{X} - \widehat{\mathbf{X}}\right\|_2$, we can detect anomalies on the GNN from a temporal perspective.

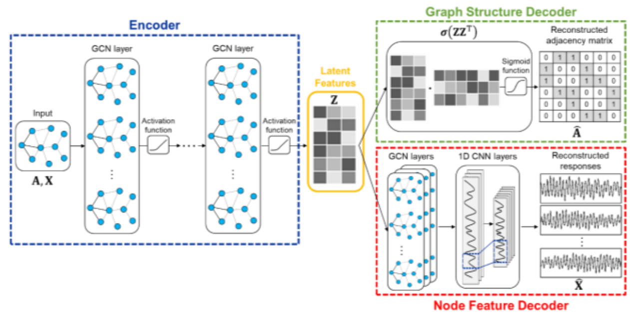

Figure 2: GNN architecture proposed for structural damage detection.

3.2.2. *Near-real-time damage identification framework*

With the methods proposed above, a GNN-based near-real-time damage identification framework is proposed. The proposed framework is divided into two parts: (1) offline; and (2) online processes.

Offline process: Before performing the near-real-time damage identification by the online process, the data processing and network training should first be performed in the offline process. The offline process has the following two steps:

1. Structural analyses are performed using the undamaged target structure to obtain the structural responses under seismic ground motions. The dataset is divided into the train, validation, and test sets. Next, the structural-mode-based adjacency matrix $\mathbf{A}$ are calculated using Eq. (6).
2. After dataset generation, the GNN model is trained using the structural responses and adjacency matrices from the undamaged structure as input data. The network is trained to minimize the following loss function $\mathcal{L}(\mathbf{X}, \mathbf{A})$ based on the reconstruction error, i.e., the difference between the input and decoded data:

$$\begin{aligned}\mathcal{L}(\mathbf{X}, \mathbf{A}) &= \alpha \mathbf{R}_S + (1-\alpha)\mathbf{R}_F \\ &= \alpha\left\|\mathbf{A} - \widehat{\mathbf{A}}\right\|_2 + (1-\alpha)\left\|\mathbf{X} - \widehat{\mathbf{X}}\right\|_2 \quad (8)\end{aligned}$$

   where α denotes a hyperparameter that controls the ratio between the graph structure and the feature reconstruction error.

Online process: The online process for the near-real-time damage identification is performed using the trained GNN model at the offline process. In this process, the structural damage indices (SDIs) of each sensor are calculated based on the reconstruction error at every timestep. The SDI of the i-th sensor, SDI_i, is defined as the value of the loss function calculated with $\mathbf{X}_i$, and $\mathbf{A}_i$, the i-th row of $\mathbf{X}$ and $\mathbf{A}$, as follows:

$$\begin{aligned}\mathrm{SDI}_i &= \mathcal{L}(\mathbf{X}_i, \mathbf{A}_i) \\ &= \alpha\left\|\mathbf{A}_i - \widehat{\mathbf{A}}_i\right\|_2 + (1-\alpha)\left\|\mathbf{X}_i - \widehat{\mathbf{X}}_i\right\|_2 \quad (9)\end{aligned}$$

4. NUMERICAL INVESTIGATIONS

4.1. Target structures

As a target structure, two- (2-D) and three-dimensional (3-D) steel frame structures in Figure 3 and Figure 4 are considered. For simplicity, the elastic section is used for each element.

As for the 2-D structure, the uniform column heights and beam lengths are set to 14 ft and 24 ft, respectively. The columns and beams have the section properties of wide flange beam W27×114 and W24×94, respectively. At the undamaged state, the frequencies of the first 5 modes are 1.11, 3.51, 6.38, 9.71, and 12.49 Hz, respectively.

As for the 3-D structure, the uniform column heights are set to 14 ft, and the columns have the section properties of wide flange beam W27×114. The uniform lengths of the beams and girders, with the section properties of wide flange beam W24×94, are set to 24 ft. At the undamaged state, the frequencies of the first 5 modes are 0.28, 0.68, 0.78, 0.81, and 1.26 Hz, respectively.

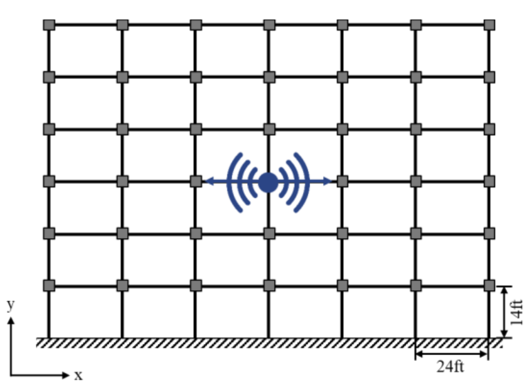

Figure 3: 2-D target structure.

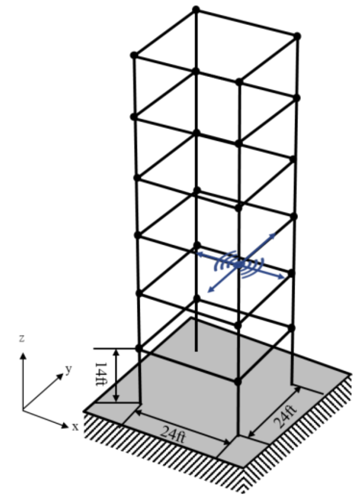

Figure 4: 3-D target structure.

4.2. Data generation and pre-processing

Structural analyses are performed using seismic ground motions from the PEER-NGA strong motion database (Chiou et al. 2008). A total of 499 ground motions are used and split into train, validation, and test datasets with a ratio of 8:1:1. It is assumed that only horizontal accelerations of all nodes are measured for the 2-D structure, while the *x*- and *y*-axis accelerations of all nodes are measured for the 3-D structure. The length of the time window is set to 5 sec for the 2-D structure and 10 sec for the 3-D structure, respectively. The time interval of the time window is set to 1 sec. The datasets are scaled to the range of $[-1, 1]$, normalized by the maximum absolute scaling.

The adjacency matrix $\mathbf{A}$ of each structure is calculated using Eq. (6) based on the mode shapes. The dimensions of $\mathbf{A}$ of 2-D and 3-D structures are 42×42 and 48×48, respectively.

4.3. Network training

The GNN model is constructed using the Python deep learning library Tensorflow and trained on a server with two NVIDIA TITAN RTX graphics cards, and 128GB RAM. The hyperparameter α of the loss function is set to 0.5. The numbers of epochs and batch size are set to 1,000 and 32, respectively. The Adam optimizer (Kingma and Ba 2014) with a learning rate of 0.001 is used for minimizing the loss function. The loss function converges fast and stably without issues of overfitting or gradient vanishing/exploding.

4.4. Near-real-time damage identification

To verify the performance of the pre-trained GNN, real-time test simulations are performed with a randomly selected test ground motion. The records of the *x*- and *y*-axis components of the test ground motion are shown in Figure 5. As for the 2-D structure, only the *x*-axis component is used. The acceleration signals are measured at every time step, and the SDIs are calculated simultaneously through the pre-trained network. In all cases, structural damage is simulated by 50% degradation in Young's modulus, occurring at peak ground acceleration (PGA).

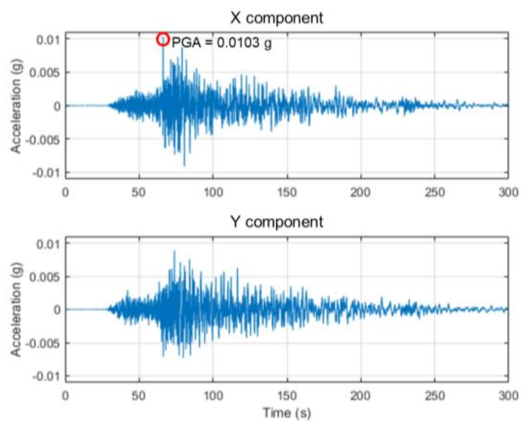

Figure 5: Records of test ground motion.

4.4.1. Case 1: Two-dimensional structure

In the figures reporting the identification results in this paper, the results of the nodes on the same floor are shown in the same subplots in order from the lower to the higher floor while the lines in each subplot indicate the SDIs of each node at every time step. As for the 2-D structure, the line colors indicate the node numbers assigned from the left to the right in Figure 3.

To verify the identification performance, two damage cases are investigated: (1) the first three columns from the left on the 2nd floor are damaged; and (2) the first three columns from the left on the 2nd and first three columns from the right on the 4th floor are damaged simultaneously. As shown in Figure 6 and Figure 7, the damaged elements are detected successfully in all cases. Note that the closer it is to the damaged elements, the higher the SDIs. Thereby, it is possible to localize damages that occurred near sensors.

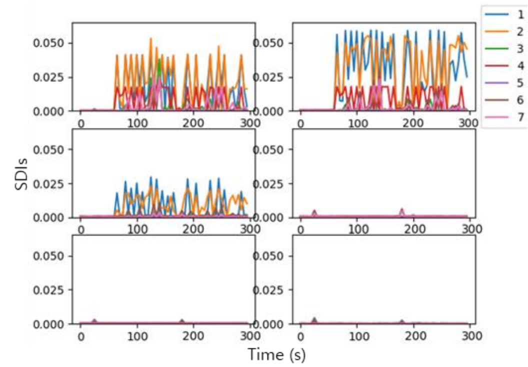

Figure 6: 2nd floor subject to damage (2-D structure).

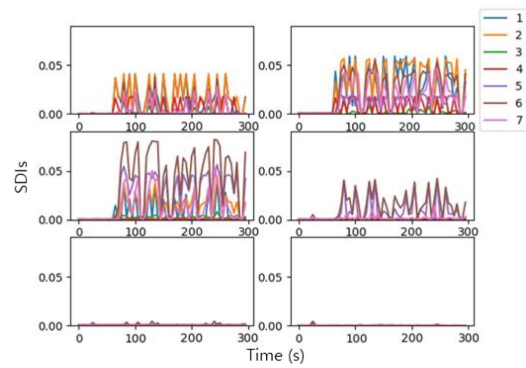

Figure 7: 2nd and 4th floor subject to damage (2-D structure).

4.4.2. Case 2: Three-dimensional structure

As for the 3-D structure, the line colors indicate the node numbers assigned counterclockwise from the lower left in Figure 4. To verify the identification performance, two damage cases are investigated: (1) two columns on the 1st floor are damaged; and (2) two columns on the 1st and 4th floors are damaged simultaneously. As shown in Figure 8 and Figure 9, the damaged elements are identified successfully in all cases.

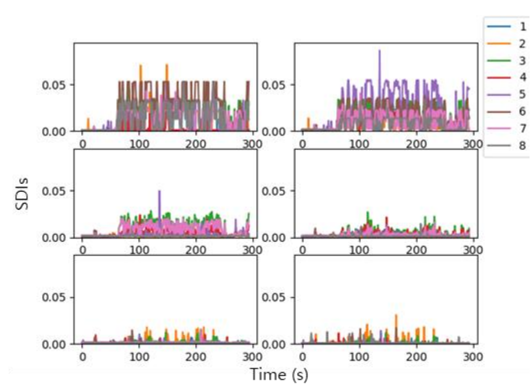

Figure 8: 1st floor subject to damage (3D structure).

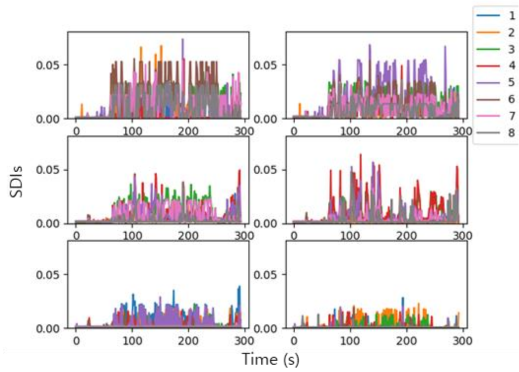
Figure 9: 1st and 4th floor subject to damage (3-D structure).

5. CONCLUSIONS

This paper proposed a new GNN-based framework for near-real-time detection of seismic damage of infrastructure systems. A novel GNN architecture was proposed to capture changes in the spatial correlation as well as the vibration characteristics. The method to construct the weighted adjacency matrix based on the structural mode was also proposed to consider accurate spatial information and the structural characteristics of the target system simultaneously. The GNN model was trained to reconstruct the vibration signals and the adjacency matrix of the target structure to leverage the spatio-temporal information of the sensor network. The SDI based on the reconstruction error of input data was introduced to quantify the damage. Numerical investigations of the 2-D and 3-D steel frames demonstrated that the proposed framework successfully identified damage under seismic load conditions in near-real-time. The robust performance of the proposed method under seismic load conditions is expected to reduce the time required for the post-disaster decision-making process. Eventually, the proposed framework will be utilized to prepare effective post-disaster operational and maintenance strategies.

6. ACKNOWLEDGEMENT

This work was supported by the National Research Foundation of Korea (NRF) Grant funded by the Korean government (MSIT) (RS-2022-00144434) and Korea Institute for Advancement of Technology (KIAT) grant funded by the Korea Government (MOTIE) (P0017304, Human Resource Development Program for Industrial Innovation).

7. REFERENCES

Chiou, B., Darragh, R., Gregor, N., and Silva, W. (2008). “NGA project strong-motion database.” *Earthquake Spectra*, 24(1), 23–44.

Dang, V. H., Vu, T. C., Nguyen, B. D., Nguyen, Q. H., and Nguyen, T. D. (2022). “Structural damage detection framework based on graph convolutional network directly using vibration data.” *Structures*, 38, 40–51.

Dawson, B. (1976). “Vibration condition monitoring techniques for rotating machinery.” *The shock and vibration digest*, 8(12), 3.

Ding, K., Li, J., Bhanushali, R., and Liu, H. (2019). “Deep anomaly detection on attributed networks.” *Proc. 2019 SIAM International Conference on Data Mining*, Society for Industrial and Applied Mathematics, Calgary, Alberta, Canada, 594–602.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT press, Cambridge.

Kingma, D.P. and Ba, J. (2014). “Adam: A method for stochastic optimization.” *arXiv preprint arXiv:1412.6980*.

Kipf, T. N., and Welling, M. (2016). “Semi-supervised classification with graph convolutional networks.” *arXiv preprint arXiv:1609.02907*.

Kipf, T. N., and Welling, M. (2016). “Variational graph auto-encoders.” *arXiv preprint arXiv: 1611.07308*.

Pathirage, C.S.N., Li, J., Li, L., Hao, H., Liu, W., and Ni, P. (2018). “Structural damage identification based on autoencoder neural networks and deep learning.” *Engineering Structures*, 172, 13–28.

Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., and Monfardini, G. (2008). “The graph neural network model.” *IEEE transactions on neural networks*, 20(1), 61–80.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P. A., and Bottou, L. (2010). “Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion.” *Journal of machine learning research*, 11(12).