Smart Structures and Systems, *Vol. 28, No. 1 (2021) 1-12*
https://doi.org/10.12989/sss.2021.28.1.001

Damage detection in structures using Particle Swarm Optimization combined with Artificial Neural Network

L. Nguyen-Ngoc [1b], H. Tran-Ngoc [1,2a], T. Bui-Tien*[1b], A. Mai-Duc [1a],
M. Abdel Wahab [3,4b], Huan X. Nguyen [5b] and G. De Roeck [6b]

[1] *Department of Bridge and Tunnel Engineering, Faculty of Civil Engineering, University of Transport and Communications, Hanoi, Vietnam*
[2] *Department of Electrical Energy, Metals, Mechanical Constructions, and Systems,*
Faculty of Engineering and Architecture, Ghent University, 9000 Gent, Belgium
[3] *Institute of Research and Development, Duy Tan University, 03 Quang Trung, Da Nang, Vietnam*
[4] *Soete Laboratory, Faculty of Engineering and Architecture, Ghent University, Technologiepark Zwijnaarde 903, B-9052 Zwijnaarde, Belgium*
[5] *London Digital Twin Research Centre, Faculty of Science and Technology, Middlesex University, London, UK*
[6] *Department of Civil Engineering, KU Leuven, B-3001 Leuven, Belgium*

(*Received November 18, 2019, Revised September 23, 2020, Accepted March 24, 2021*)

Abstract. In this paper, a novel approach to damage identification in structures using Particle Swarm Optimization (PSO) combined with Artificial neural network (ANN) is proposed. With recent substantial advances, ANN has been extensively utilized in a wide variety of fields. However, because of the application of backpropagation algorithms based on gradient descent techniques, ANN may be trapped in local minima when seeking the best solution. This may reduce the accuracy of ANN. Therefore, we propose employing an evolutionary algorithm, namely PSO to deal with the local minimum problems of ANN. PSO is employed to improve the training parameters of ANN consisting of weight and bias ratios by reducing the deviation between calculated and desired results. These training parameters are then used to train the network. Since PSO applies global search techniques to look for the best solution, it can assist the network in avoiding local minima by looking for a beneficial starting point. In order to assess the effectiveness of the proposed approach, both numerical and experimental models with different damage scenarios are employed. The results show that ANN -PSO not only significantly reduces computational time compared to PSO but also possibly identifies damages in the considered structures more accurately than ANN and PSO separately.

Keywords: Artificial Neural Network (ANN); damage identification; local minima; Particle Swarm Optimization (PSO); training parameters

1. Introduction

During exploitation, due to the effect of environment, service loads, and accidental actions, structures may suffer from serious defects that are extremely difficult to detect through visualization. Therefore, damage detection methods based on non-destructive techniques have brought enormous benefits to Structural Health Monitoring (SHM). These methods can find defects that are embedded in the structure without destroying or adversely affecting the structure. During the past decades, there has been a sharp increase in the number of successful applications of damage identification employing non-destructive methods (Kourehli 2017a, Ho *et al.* 2018, Tran-Ngoc *et al.* 2019a, Tran and Bui 2019, Ghannadi and Kourehli 2019a, b, Khatir and Wahab 2019a, Mai *et al.* 2020, Cuong and Quang 2020, Hien 2020). Hwang and Kim (2004) applied methods derived from frequency response function data to damage identification and quantification in a helicopter rotor blade and a cantilever beam. Maity and Tripathy (2005) proposed a method based on a hybrid Genetic Algorithm (GA) to identify damages in a plane frame and a cantilever beam with several damage scenarios. Miguel *et al.* (2013) identified damage cases of cantilever beams by applying a hybrid optimization algorithm based on stochastic and deterministic principles. To evaluate the effectiveness and reliability of the proposed approach, its results were compared with other algorithms including PSO, GA, and Harmony Search algorithm. Kaved and Maniat (2015) used PSO algorithm and Magnetically Charged System Search algorithm for damage identification in trusses and beams. Na *et al.* (2011) presented a fresh approach to damage identification in a 20-storey building using GA combined with structural dynamic responses. Mares and Surace (1996) combined the residual force method with GAs to detect damages in a truss structure and a cantilever beam. Chou and Ghaboussi (2001) identified damage location and severity in a truss structure by solving an inverse problem based on GA. Xu *et al.* (2016) employed an Improved Artificial Bee Colony algorithm based on a chaotic search mechanism and tournament selection strategy to detect

---

*Corresponding author, Ph.D., Professor,
E-mail: btthanh@utc.edu.vn
[a] Ph.D. Student
[b] Professor

http://www.techno-press.org/?journal=sss&subpage=7
ISSN: 1738-1584 (Print), 1738-1991 (Online)

damages in a double-beam system. Hou *et al.* (2019) used Improved PSO by adjusting inertia weight coefficients to deal with optimization problems of 12 benchmark functions. However, since such aforementioned algorithms have to adjust too many parameters in the process of searching for the best solution through iterations, it is very time-consuming when using them for optimization issues of structures with many Degrees of Freedom (DoFs). This may restrict their capacity to handling the optimization problems of large-scale structures (Khatir *et al.* 2017, Samir *et al.* 2018, Tiachacht *et al.* 2018, Zenzen *et al.* 2018, Ghannadi and Kourehli 2019c, Tran-Ngoc *et al.* 2018, 2020a, b, 2021a, Hoa *et al.* 2020).

ANN is a model inspired by biological neural systems. In recent years, there has been an increasing tendency towards successful applications of ANN for different fields, e.g., disease diagnosis, target recognition as well as sales forecasting (Kourehli 2017b, Ghadimi and Kourehli 2017, Kourehli 2018, Khatir *et al.* 2019, 2020a). Zhou and Wahab (2017) combined transmissibility with an auto-associative neural network to detect damages in a simulated ten-floor structure. The impacts of noise were fully evaluated in the model. Abdeljaber *et al.* (2017) used Convolutional Neural Networks (CNNs) to identify damage locations and levels in a large-scale steel frame. Janssens *et al.* (2016) identified defects in bearings of rotating machines using a learning model derived from CNNs. Hakim and Razak (2013) employed ANNs to identify damages in a steel bridge, in which the natural frequencies of the first five modes were selected and used as the objective function to minimize the difference between calculated and desired outputs. It is acknowledged that in the process of training the network, the particles are transferred from the input layers to the output layer based on training parameters in the forward process. If calculated outputs are still significantly different from the desired ones, a backward process is conducted using backpropagation algorithms to recalculate training parameters. This process repeats until the difference between real and desired outputs is the smallest. However, because backpropagation algorithms are based on gradient descent techniques, the network seems to be got stuck in local minima.

To overcome the aforementioned limitations of ANN, some researchers proposed combining ANN with evolutionary algorithms to handle optimization problems (Khatir *et al.* 2020b and Yazdanmehr *et al.* 2009). Those algorithms are based on global search techniques to look for the best solution, which plays a crucial role in avoiding local minima. Inspired by the aforementioned approaches, we proposed a hybrid algorithm combining PSO with ANN and used to detect damages in considered structures. This hybrid algorithm enhances the effectiveness of ANN by improving the parameters used to train the network. PSO is employed to determine the training parameters of the ANN, consisting of weight ratio and bias ratio. Each value of training parameters (weight and bias) is considered as a solution that the PSO algorithm is looking for. The objective function is the difference between calculated and desired results. Training parameters are determined when the value of the objective function is the smallest (Tran-Ngoc *et al.* 2019b). After that, these training parameters are used to train the network. To demonstrate the effectiveness of the proposed approach, PSO and ANN are also used for damage identification in the considered structures.

2. PSO

As natural law, some animals e.g., birds or fishes live in groups to gain considerable advantages during evolution. In the process of seeking food, they share information with other members, which not only assists in the reduction in the search time for food but also helps to increase the opportunity for obtaining the most food. Drawing creative inspiration from this strategy, in 1995, Eberhart and Kennedy (1995) developed an evolutionary algorithm called PSO. Particles of PSO imitate the behaviors of fishes and birds when seeking the optimal solution. The PSO algorithm is derived from Eqs. (1)-(9).

The first equation determines the velocity of each particle, which is represented by elements in matrix $\boldsymbol{V}^i$ of size $n \times m$

$$V^{(i+1)} = w_0 V^i + C_1 r_1 (P^i - X^i) + C_2 r_2 (\overline{P}^i - X^i) \tag{1}$$

Where

$$V^i = [v_1^i, v_2^i, \ldots, v_m^i] \tag{2}$$

$$P^i = [p_1^i, p_2^i, \ldots, p_m^i] \tag{3}$$

$$X^i = [x_1^i, x_2^i, \ldots, x_m^i] \tag{4}$$

$$\overline{P}^i = [\overline{P}^i]_{1*m} \tag{5}$$

And

$$v_j^i = [{v_{1j}}^i, {v_{2j}}^i, \ldots, {v_{nj}}^i]^T, j = 1,2,\ldots,m. \tag{6}$$

$$p_j^i = [{p_{1j}}^i, {p_{2j}}^i, \ldots, {p_{nj}}^i]^T, j = 1,2,\ldots,m. \tag{7}$$

$$x_j^i = [{x_{1j}}^i, {x_{2j}}^i, \ldots, {x_{nj}}^i]^T, j = 1,2,\ldots,m. \tag{8}$$

$$\overline{P}^i = {[\overline{p}^i]_{n*1}}^T \tag{9}$$

C_1 and C_2 are the social learning factor and the cognition learning factor, respectively; r_1 and r_2 represent random values in the range of (0,1); w_0 denotes the inertia weight parameter; *n* is the number of populations, whereas *m* is the number of uncertain parameters that need to find. x_q^i and v_q^i represent the position and velocity of particle *q* at the i^{th} iteration, respectively. The position of particles (*x*) is also the training parameters used to train the network. p_q^i presents the best local position of particle *q* at the i^{th} iteration, whereas $\bar{p}^i$ is the best global position of all particles at the i^{th} iteration. $(.)^T$ denotes transpose operation.

The second equation is to update the position of each particle

$$X^{(i+1)} = X^{i} + V^{(i+1)} \tag{10}$$

PSO identifies the best solution based on the objective function consisting of structural damage location and level. When the discrepancy between calculated and desired outputs (objective function) is minimal, the best solution is achieved. Each element is featured by its physical position and velocity vector in the space. In the moving process, each element can remember the best local solutions and compare them with other members. Thanks to this strategy, the best solution is obtained when iterations finish. Since only the best optimal solution is given after each iteration, PSO is superior to existing optimization algorithms e.g., genetic algorithm (GA) in terms of convergence speed and accuracy when handling optimization issues (Tran-Ngoc *et al*. 2018).

3. ANN

ANN is an algorithm that can automatically learn identification from experience to improve its performance. The trained network can be used to identify new data sets that have the same characteristics as trained ones. ANN has three major learning methods consisting of unsupervised learning, semi-supervised learning together with supervised learning. The unsupervised learning model solely contains input variables without the corresponding output ones (Tran-Ngoc *et al*. 2019b, 2020c, 2021b, Nguyen *et al*. 2020, Dang *et al*. 2020, 2021). The semi-supervised learning model is used to deal with incomplete training data in which a part of the input variables does not possess corresponding output ones. The supervised learning model owns both input variables and corresponding output ones. The supervised learning model can be split into regression and classification functions. Regression functions are employed to predict outputs, whereas classification functions utilized to categorize outputs. Although ANN is a subset of machine learning (ML), it still has several distinctive functions. While the ML algorithm needs guidance when learning identification from the data, ANN itself can tackle problems without interventions. In this paper, since we used both input and output variables for training the network described in more detail in Section 3.1, the supervised learning model of ANN is applied and combined with PSO to detect damages in structures

3.1 Input and output variables

ANN is a computational model used to deal with multi-parametric issues. This function provides ANN with the flexibility to choose input and output variables based on the problems that need to be solved. Selected input data must produce significant effects on the output ones. Therefore, in this paper, structural dynamic characteristics consisting of natural frequencies are utilized for input variables, and damage characteristics (damage locations and levels) are used as outputs. The input and output variables are organized into a tabular form for the process of training the network.

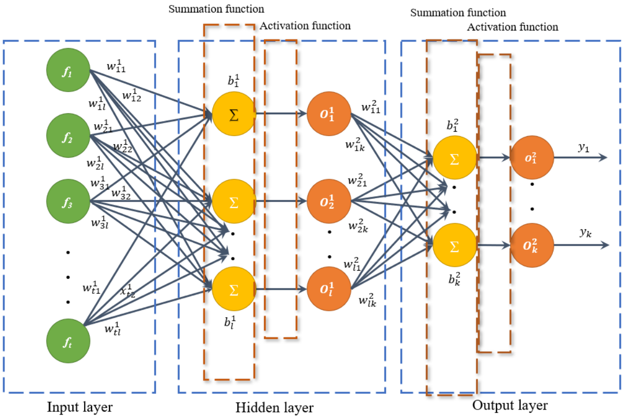


Fig. 1 Three-layer neural network architecture

3.2 ANN network architecture

The network of ANN consists of three major parts, i.e., an input layer, an output layer, and hidden layers located between the input and output layer. Each layer includes processing elements linked to each other using weight and bias ratios. In the group of the various network architectures used in ANN, Multilayer Feedforward Perceptron (MFP) networks are the most popular. This architecture comprises one input layer, one output layer, and one or more hidden layers. The selection of the number of hidden layers is based on the specific problems that need to be tackled. No convincing proofs demonstrate that increasing the number of hidden layers can create a better network. On the other hand, utilizing too many hidden layers may lead to an increase in computational cost. Therefore, in this paper, we only use one hidden layer for the network as shown in Fig. 1.

3.3 Training, validation and test

The process of training the network is conducted as follows: Firstly, the input layer receives input signals (f_1,., f_t) and then transfers them to hidden layers as shown in Fig. 1. This process is formed using two equations. The first one is a summation function (φ) depending on training parameters (weight and bias) and output values of previous layers, i.e.

$$Input_z = \varphi_z = \sum_{y=1}^{t} w_{yz} f_y + b_z \, ; z = 1 \div l \tag{11}$$

Where f_y indicates the output of the y^{th} neuron of the input layer, $Input_z$ is the input data of the z^{th} neuron of the hidden layer. b_z and w_{yz} are training parameters, namely bias and weight ratios; whereas t is the number of the neurons of the input layer connecting with the z^{th}neuron of the hidden layer; l is the number of the neuron of the hidden layer. The second one is an activation function used to map the relationship between input and output values. Four main types of activation function exist, namely step function, ramp function, sigmoid function, and Gaussian function. The sigmoid function is commonly used because it can deal with numerous complex issues relating to nonlinear functions. In this paper, we use the sigmoid function to calculate output values. Output values (O_z^1) are obtained using the sigmoid function as shown in Eq. (12).

$$O_z^1 = \frac{1}{1 + e^{-input_z}} \tag{12}$$

The process of transferring neurons from the hidden layer to the output layer is the same one mentioned in Eqs. (11)-(12). Training a network indeed is to change weight and bias ratios (training parameters) to minimize the difference between the calculated and desired outputs. The particles of the network of ANN are transferred from input layers to output layers based on training parameters in the forward process. If calculated outputs are still significantly different from the desired ones, the backward process is conducted using backpropagation algorithms to recalculate training parameters. This process repeats until the difference between real and desired outputs is the smallest. However, since backpropagation algorithms are based on gradient descent techniques, the network tends to be trapped in local minima when the network creates too many locally optimal solutions. To solve this issue, we employ PSO to identify the training parameters (weight and bias) used to train the network. PSO is an evolutionary algorithm based on global search techniques to look for the best solution. Each value of training parameters (weight and bias) is considered as a solution that the PSO algorithm is looking for. For the process of seeking the global best of PSO, the reader can refer to section 2 and the work of Tran-Ngoc *et al.* (2019b). The objective function is the difference between calculated and desired results (damage locations and levels). Training parameters are determined when the value of the objective function is the smallest. After that, training parameters are used to train the network. To consider the effect of noise on the input data, white Gaussian noise is applied for input data (natural frequencies) as shown in Eq. (13) (DjuriC 1996).

$$Noise = \sqrt{10^{10 * log_{10}\frac{\sum_{v=1}^{N}(f_v)^2}{N} - snr}} * rand \tag{13}$$

Where N is the number of input data, f_v is input data; *snr* is noise ratio used for the vector signal of input data; rand represents random numbers ($0 < rand < 1$). The application of ANN-PSO to damage detection of structures is illustrated in Fig. 2.

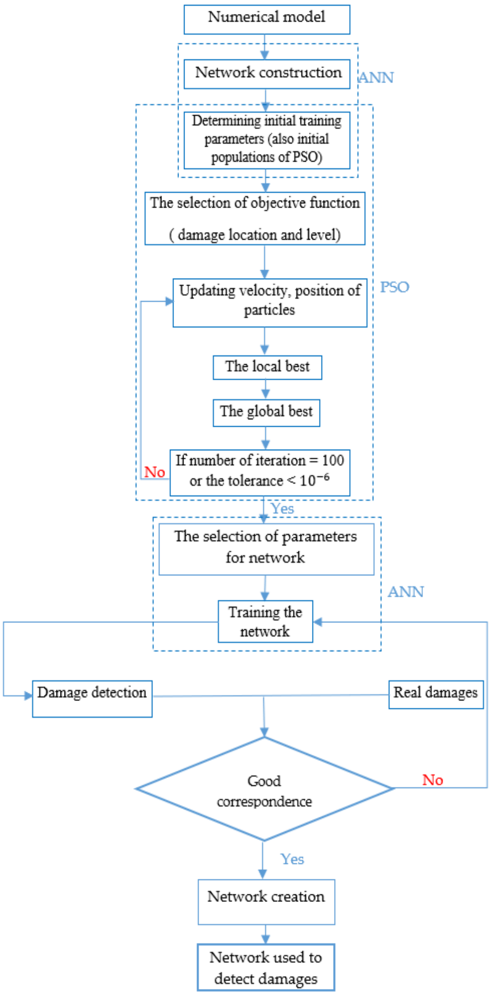


Fig. 2 Methodological approach for damage identification in the structures employing ANN-PSO

4. Numerical examples

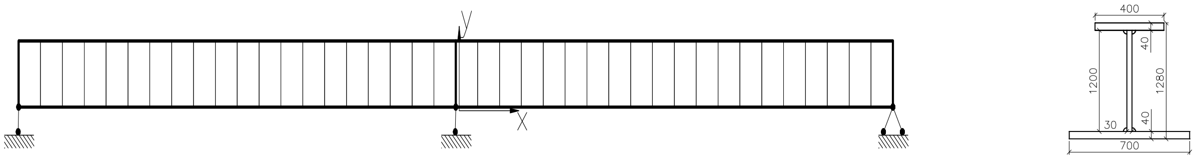
In this section, the effectiveness of the proposed approach is demonstrated through a continuous beam, with a wide variety of damage scenarios. The effect of the noise on the input data is evaluated by adding white Gaussian noise that is evaluated as 3% for natural frequencies. A computer with an Intel I7- 45000 CPU, and 8 GB RAM is used for data analysis. To compare with the proposed method, ANN and PSO separately are also employed to detect damages in considered structures. The steel beam includes two continuous spans of equal length (20 m) shown in Fig. 3(a). The cross-section of the beam is an I-Beam (Fig. 3(b)). The beam is put on three supports. While supports 1 and 2 are expansion ones, support 3 is a fixed one. The material properties of the beam are listed in Table 1.

The finite element model (FEM) of the beam is generated using StaBIL, a MATLAB Toolbox was developed by François *et al.* (2021). The beam comprises 41 nodes and 40 elements modeled with two-dimensional (2D) beam elements. This element has 3 DoFs at each node including two translational displacements in the x, y - axes, and rotational displacement around the z-axis.

Modal analysis is performed utilizing a baseline model to create the input and output data for the network. The network with three layers consisting of one input layer, one

Table 1 Material properties of the beam

Parameters		
Young's modulus	Volumetric mass density	Poisson's ratio
2×10^{11} N/m^2	7850 Kg/m^3	0.3



(a) The layout of the beam (b) The cross-section of the beam

Fig. 3 The continuous beam. The unit is in mm.

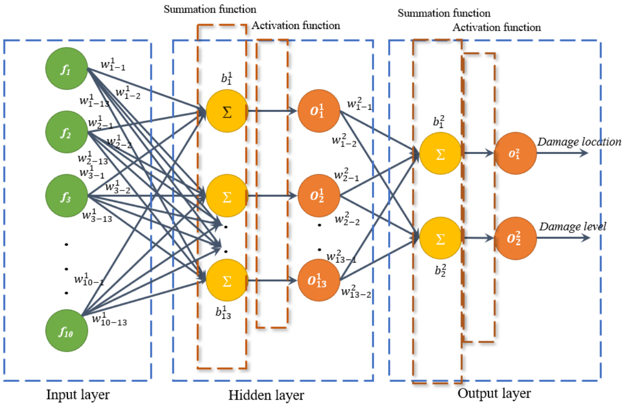


Fig. 4 The network architecture for the continuous beam

Table 2 Natural frequencies of the first ten modes for the continuous beam

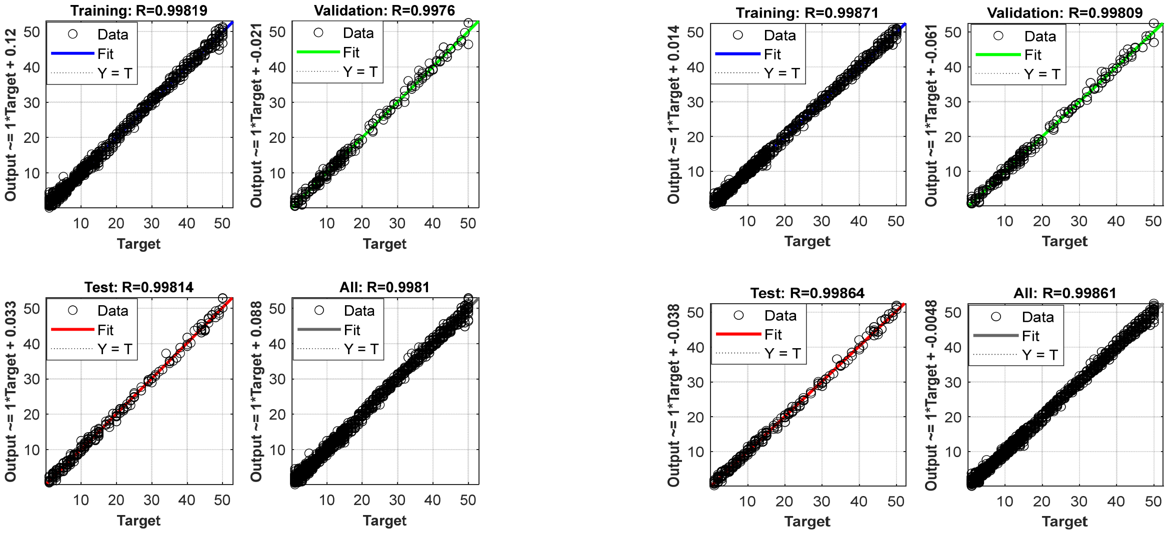
Modes	Natural frequencies (Hz)	Modes	Natural frequencies (Hz)
1	9.12	6	81.15
2	14.25	7	95.36
3	32.36	8	97.88
4	36.27	9	144.00
5	45.92	10	162.66

output layer and one hidden layer shown in Fig. 4. While input data consists of the first ten natural frequencies (Table 2), output data (objective function) relates to damage locations and levels. A hidden layer with 13 neurons is employed.

The samples used for training, validation, and test are chosen randomly from the generated datasets with a division of 70%, 15%, and 15%, respectively. The trained network is used for damage identification and quantification in the considered structure. In order to compare with ANN-PSO, ANN, and PSO are also used for damage detection in the beam. For PSO, the number of population is 150, the values of the social learning factor and the cognition learning factor are $C_1 = 2$ and $C_2 = 2$, respectively, whereas the inertia weight parameter (w_0) is 0.3. The ANN employs the Levenberg-Marquardt (LM) algorithm to train the network.

4.1 Single damages

Damage scenarios in the beam are generated by decreasing the stiffness of elements. The stiffness parameters range between 0 and 1. While 1 indicates the intact case and 0 represents the totally damaged case. For single damage cases, the stiffness of the damaged elements reduces from 1 to 0.5 with a step of 0.01, whereas the remaining elements are intact. Because the beam is symmetric, only $\frac{1}{2}$ of the beam is used for structural



(a) ANN (b) ANN-PSO

Fig. 5 The process of training the network - Single damage scenario

Table 3 The indexes assess the performance of algorithms - Single damage scenario

Algorithms	MSE-values	*R*-Values	Training time (sec)
PSO	0.42		2216
ANN	0.71	0.9981	245
ANN-PSO	0.053	0.9986	267

damage identification in both single and multiple damages. A total number of 1000 data samples are created from the baseline model and used to train the network. Output data consists of damage locations and levels.

Fig. 5 indicates that the regression line coincides with the 45-degree one, and regression values (*R*) are higher than 0.9. This demonstrates that a close correspondence between real and target results is achieved. Table 3 shows the indexes assessing the performance of algorithms.

Table 3 shows that MSE value calculated by ANN-PSO is the lowest, at 0.053, whereas MSE values determined by PSO and the ANN are 0.42 and 0.71, respectively. Additionally, ANN-PSO outperforms ANN in terms of *R*-value. This means that analysis outputs (damage locations and levels in the beam) determined by ANN-PSO are closer to real outputs than PSO and ANN. MSE determined by ANN is the highest because this algorithm applies backpropagation algorithms, which may trap in the local minima. ANN-PSO demonstrates its ability to the search for the optimal solution since ANN-PSO applies global search techniques of PSO to look for the best training parameters

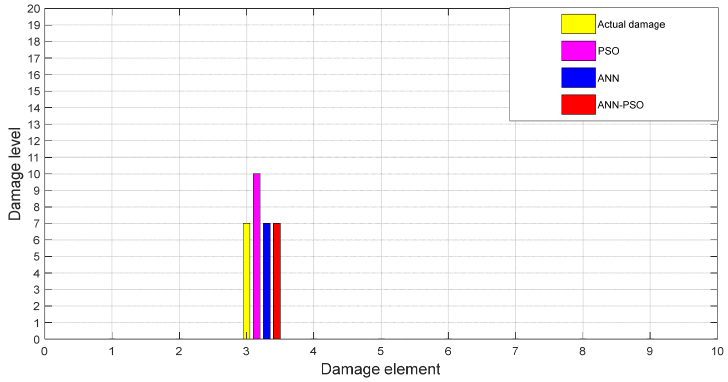


Fig. 6 Damage detection results of PSO, ANN, and ANN-PSO of element 3: 7% of damage

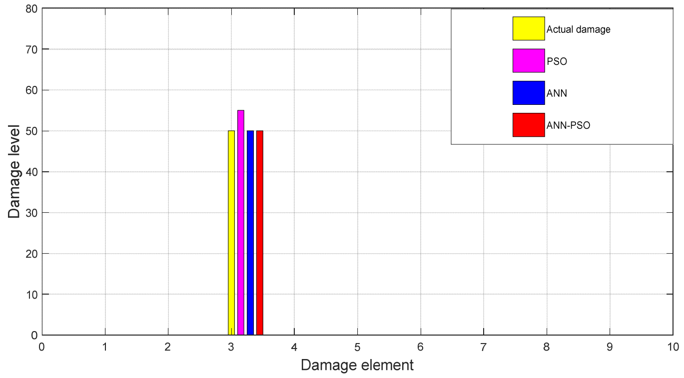


Fig. 7 Damage detection results of PSO, ANN, and ANN-PSO of element 3: 50% of damage

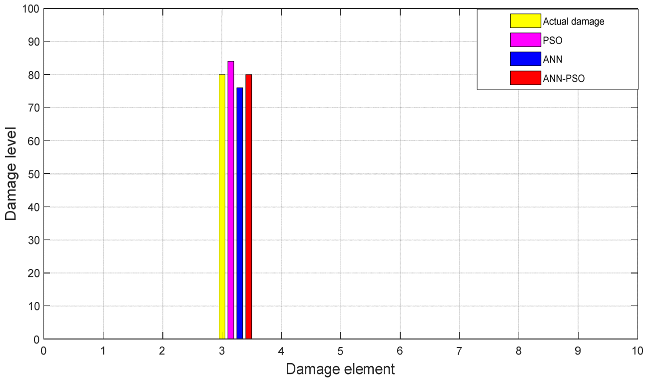


Fig. 8 Damage detection results of PSO, ANN, and ANN-PSO of element 3: 80% of damage

used to train the network. In terms of the time to look for the best solution, ANN-PSO requires more time than ANN but this deviation is insignificant, whereas ANN-PSO spends less time on training the network than PSO. Specifically, ANN-PSO spends 267 seconds on training the network, whereas ANN and PSO devote approximately 245 seconds and 2216 seconds, respectively to train the network. PSO expends the most time in looking for the best solution because this algorithm has to adjust too many parameters in the search process.

From Figs. 6-8, it can be seen that PSO, ANN, and ANN-PSO can detect damage location of element 3 accurately. ANN only identifies exactly damage levels that belong to the trained network, whereas the results of damage level detection of PSO still contain minor errors compared to real results. ANN-PSO demonstrates its effectiveness when determining damage levels of element 3 exactly.

Table 4 The indexes assess the performance of the proposed algorithms – Multiple damages

Algorithms	MSE-Values	*R*-Values	Training time (sec)
PSO	0.58		2278
ANN	0.93	0.998	289
ANN-PSO	0.062	0.999	323

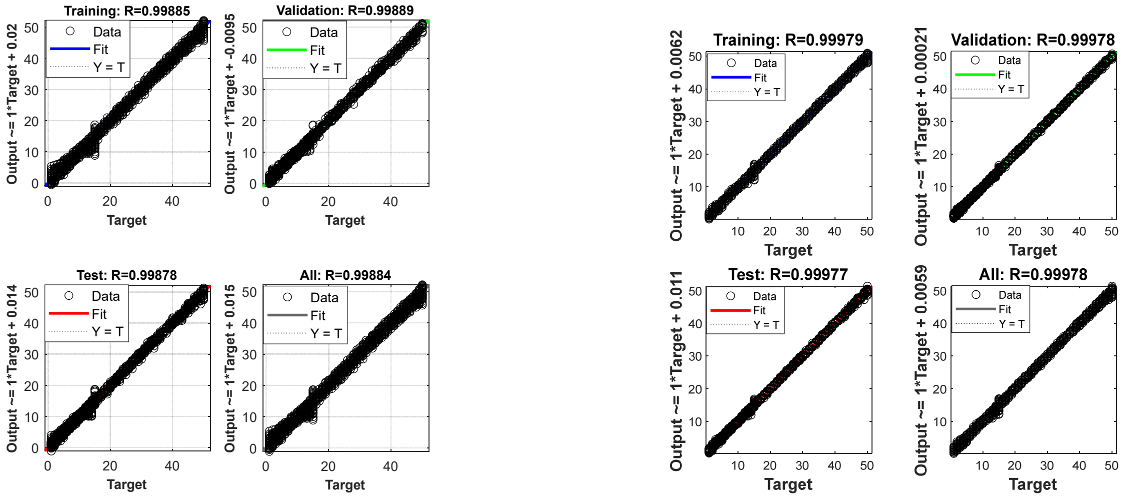
4.2 Multiple damages

The stiffness of elements reduces from 0% to 50% with an interval of 1%, which is randomly assigned to two elements at the same time, whereas other elements are intact. A total number of damage cases (9500) are created from the baseline model used to train the network.

Table 4 shows the indexes assessing the performance of the proposed algorithms.

Table 4 and Fig. 9 show that ANN-PSO provides a smaller discrepancy between calculated and desired outputs than both PSO and ANN. For the time to find the best solution, while ANN and ANN-PSO spend only 289 seconds and 323 seconds on looking for the best solution, respectively, PSO devotes 2278 seconds to this process.

Fig. 10 shows that PSO, ANN, and ANN-PSO can identify damage location of elements accurately. Some minor errors happen when using PSO and ANN to detect the damage level of elements. The predicted results of damage severity applying ANN-PSO correspond to the real ones.



(a) ANN (b) ANN-PSO

Fig. 9 The process of training the network – Multiple damages

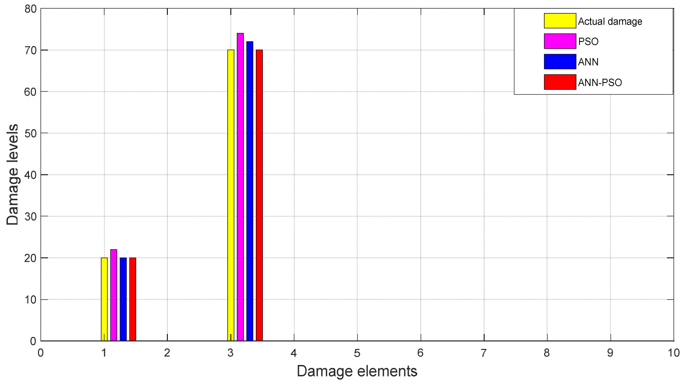


Fig. 10 Damage detection results of PSO, ANN, and ANN-PSO in the steel beam: (a) 20% of the damage of element 1 and 70% of the damage of element 3

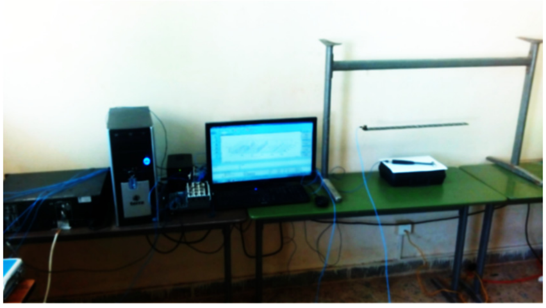


Fig. 11 Experimental setup (Tran-Ngoc *et al*. 2019b)

Table 5 The dimension of the beam

The length of the beam (m)	The height of the beam (m)	The width of the beam (m)
0.6	0.038	0.006

Table 6 The material properties of the beam

Young's modulus (E)	Poisson's ratio	Volumetric mass density (G)
190 ~ 200 Mpa	0.3	7850Kg/m^3

Table 7 Natural frequencies of the first 3 modes

	f_1 [Hz]	f_2 [Hz]	f_3 [Hz]
Intact case	526.5	1410.9	2751.6
damaged case – 3 mm	523.4	1409.0	2707.4
damaged case – 6 mm	514.8	1405.9	2673.4
damaged case – 9 mm	500.8	1403.9	2623.8

5. A laboratory beam

To evaluate the robustness and effectiveness of ANN-PSO, a steel beam with a free-free boundary condition was employed (Fig. 11). The dimension and material properties of the beam are described in detail in Tables 5-6.

A laboratory measurement (Fig. 11) was conducted to obtain structural dynamic behaviors. The excitation source was generated by the force of the hammer. Accelerometers (PCB 356A15) were used to get the signal.

The intact beam was first considered. After that damages generated in the middle of the beam with crack lengths of 3 mm, 6 mm, 9 mm. Peak picking method was applied to obtain natural frequencies of all cases shown in Figs. 12-15 and Table 7 (Tran-Ngoc *et al*. 2019b).

A FEM of the beam is constructed using StaBIL. This model consists of 12 nodes, 11 elements. Two-dimensional beam elements are used that have 4 DoFs at each node

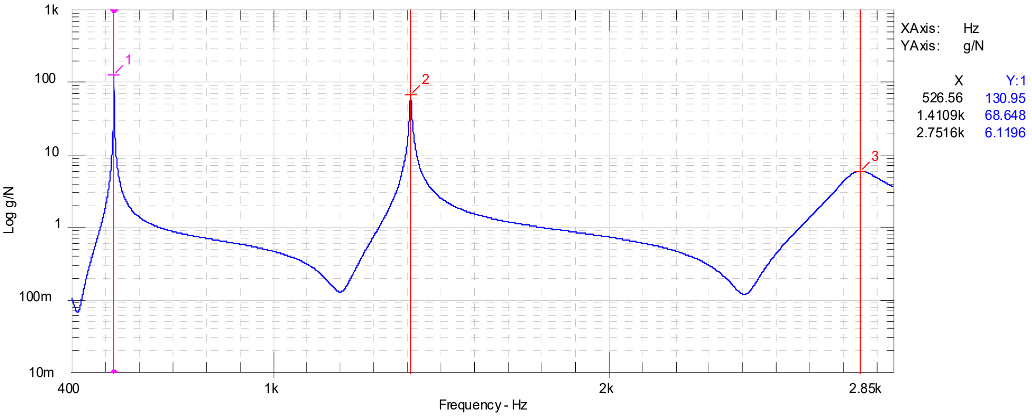


Fig. 12 Undamaged case

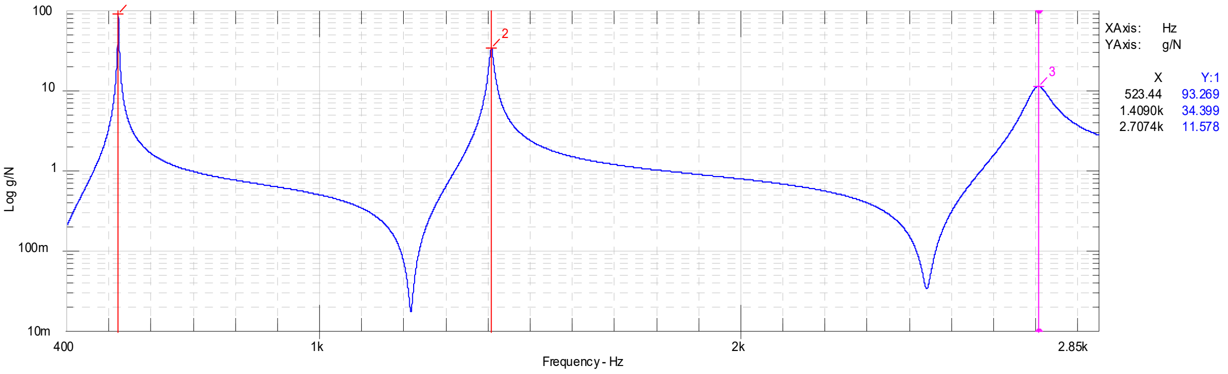


Fig. 13 Damaged case 1 – 3 mm

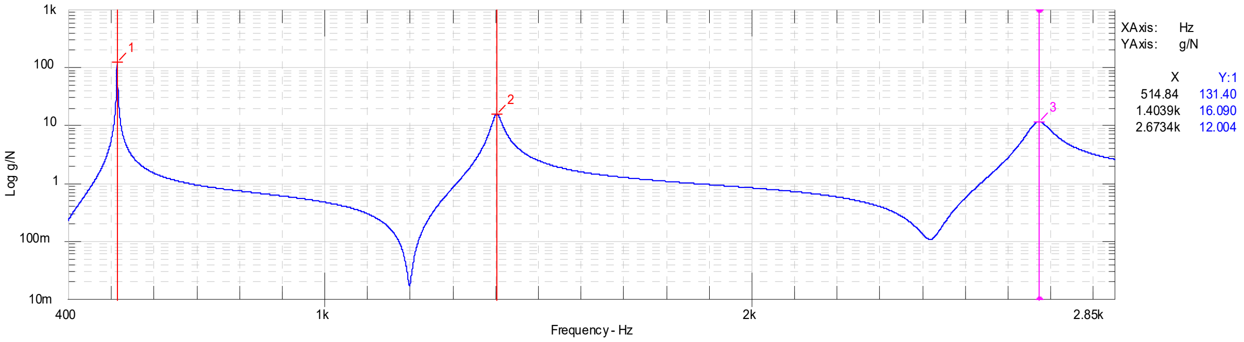


Fig. 14 damaged case 2 – 6 mm

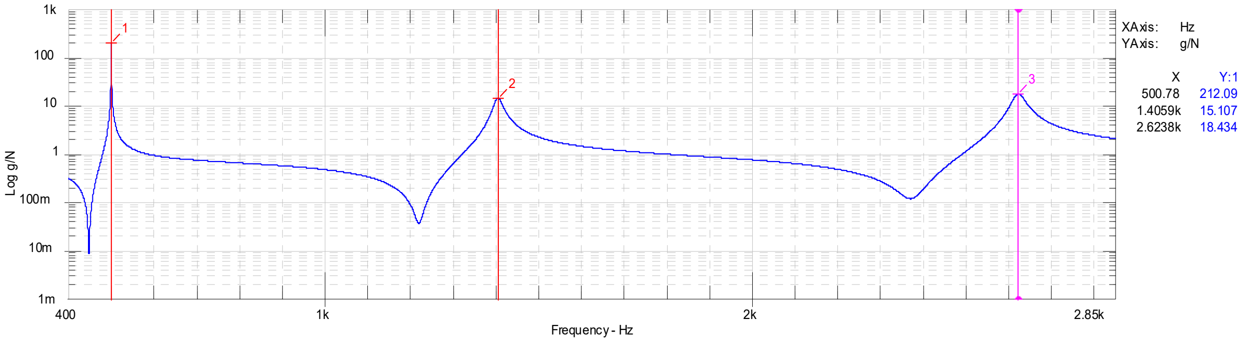


Fig. 15 Damaged case 3 – 9 mm

Table 8 Natural frequencies before and after model updating (Tran-Ngoc *et al.* 2019b)

Modes	1	2	3
Before (Hz)	525 (0.19%)	1406 (0.28%)	2743 (0.29%)
After (Hz)	526 (0%)	1411 (0%)	2752 (0.05%)
Measurement (Hz)	526	1411	2751

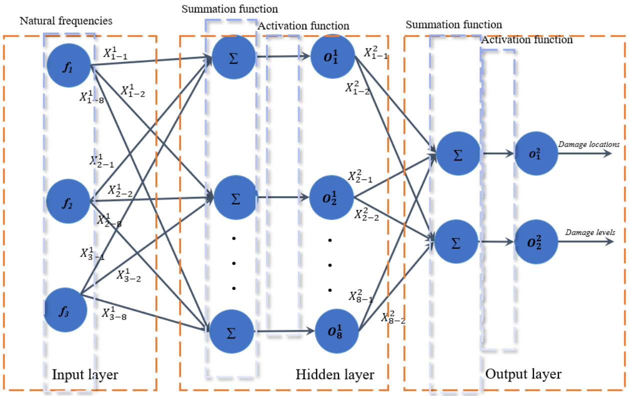


Fig. 16 Network architecture

comprising translational displacements in the *x*, *y*-directions, and rotational displacements around the *x*, *y* -directions. Model updating is applied to minimize the difference between analytic and measured results.

Uncertain parameters include Young's modulus of elements. Table 8 shows natural frequencies of the beam before and after model updating.

Table 8 shows that after model updating a close correspondence between analytic and measured results is achieved.

The updated model is used to generate data for the network. Damages are created by reducing the stiffness at each element with an interval of 1%. A total of 550 samples are used to train the network. These samples are split into 70%, 15%, and 15% for training, evaluation, and test, respectively. The network consists of 3 layers (one input layer, one output layer, and one hidden layer) shown in Fig. 16. The input data includes natural frequencies of 3 modes and the output data is damage location and level. The hidden layer consists of 8 neurons.

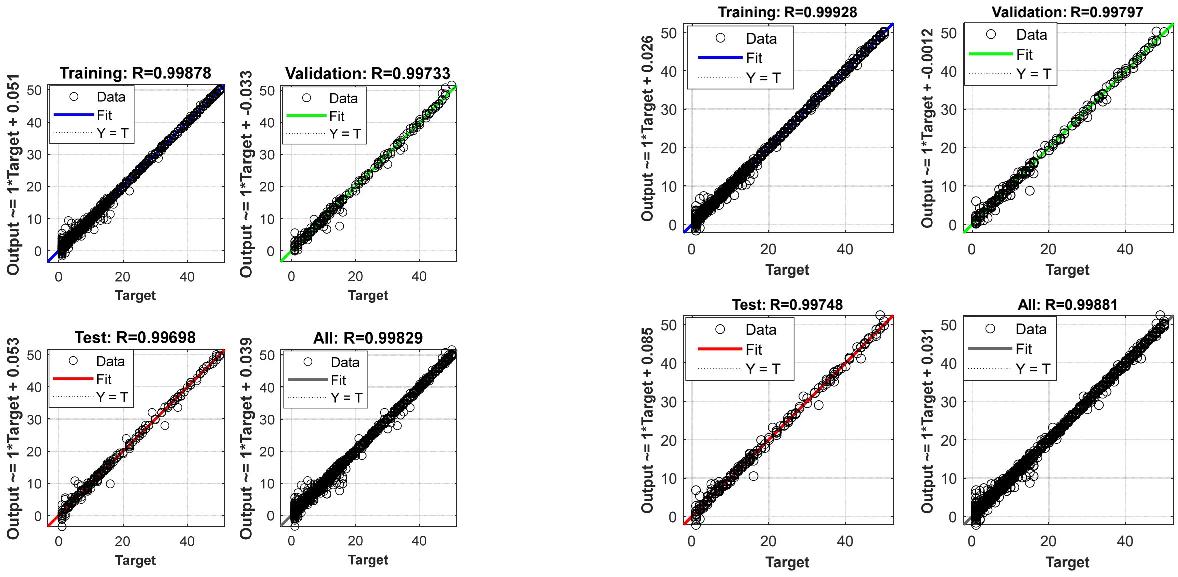
PSO alone and ANN alone are also used to compare with ANN-PSO. Apart from the number of population of PSO (in this section, 50 populations of PSO are employed), other parameters of PSO and ANN are chosen the same ones as section 4.

Table 9 shows the performance evaluation indices of the proposed algorithms.

Table 9 and Fig. 17 show that R-Values calculated by ANN-PSO are higher than those identified by ANN, whereas MSE values calculated by ANN-PSO are smaller than those provided by ANN and PSO. In terms of

Table 9 Performance indices of algorithms

Algorithms	MSE-Values	R-Values	Training time (sec)
PSO	0.051		545
ANN	0.82	0.9982	289
ANN-PSO	0.289	0.9989	392



(a) ANN (b) ANN-PSO

Fig. 17 The process of training the network

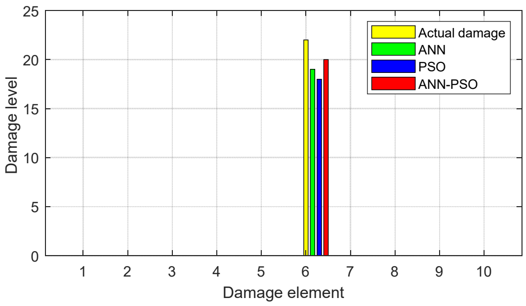


Fig. 18 Damage detection results of PSO, ANN, and ANN-PSO in the steel beam: (22) % damage of element 6

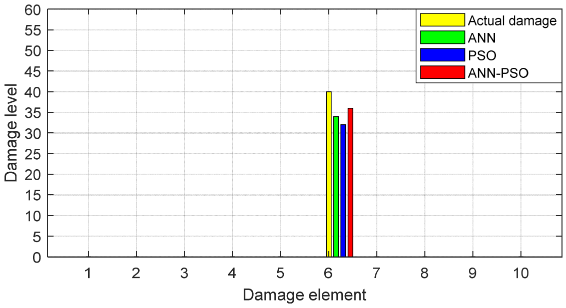


Fig. 19 Damage detection results of PSO, ANN, and ANN-PSO in the steel beam: (40) % damage of element 6

computational time, ANN-PSO and ANN spend less computational time than PSO. Specifically, While ANN, and ANN-PSO devote 289, and 392 seconds, respectively to finding the best solution, PSO devotes the most time (545 seconds) to this process.

As shown in Figs. 18-20, the damage location of element 6 is exactly determined by PSO, ANN, and ANN-PSO. In terms of detecting damage level, the predicted result calculated by ANN-PSO is closest to actual damage.

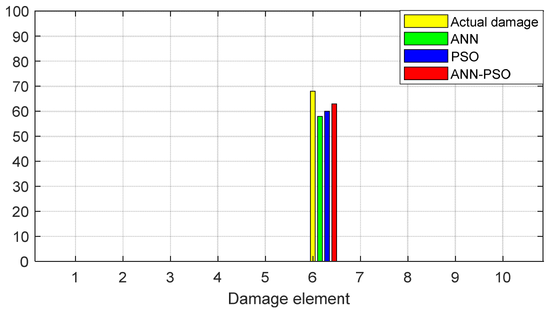


Fig. 20 Damage detection results of PSO, ANN, and ANN-PSO in the steel beam: (68) % damage of element 6

6. Conclusions

In this paper, a flexible combination of ANN and PSO is used for damage identification in structures consisting of a continuous beam, a laboratory beam. This combination is employed to overcome the limitation of ANN applying gradient descent techniques. PSO is employed to identify training parameters (weight and bias) used to train the network. To compare with ANN-PSO, ANN, and PSO also are employed to determine damages in the considered structures. Based on the results achieved, some main conclusions can be drawn as follows:

- PSO, ANN, and ANN-PSO can detect damage location in the considered structures exactly. A close correspondence between analysis and measured results is achieved (MSE lower than 0.08, and R-Values over 0.99).
- PSO devotes too much CPU time to looking for the best solution. Due to this big drawback, it is extremely difficult to employ PSO to handle the problems of damage identification in complex structures with numerous DoFs.
- ANN spends the least time on finding the best

solution. However, ANN may trap in the local minima when the network generates too much the local optimal solution.
- ANN-PSO not only considerably reduces computational time than PSO but also outperforms than ANN and PSO in terms of accuracy when seeking the optimal solution.
- Further investigation into damage identification should be conducted in more large-scale structures to assess the robustness of the proposed method.

Acknowledgments

The authors acknowledge the financial support of VLIR-UOS TEAM Project, VN2018TEA479A103, “Damage assessment tools for Structural Health Monitoring of Vietnamese infrastructures”, funded by the Flemish Government. This work is also supported by the Newton Fund Institutional Links through the U.K. Department of Business, Energy, and Industrial Strategy and managed by the British Council under Grant 429715093 and partly supported by the UK-India Education and Research Initiative (UKIERI) grant ID ‘DST-UKIERI 2018-19-11’. Moreover, the author needs to acknowledge the financial supports from Ministry of Education and Training (MOET) under the project research B2018-GHA-04SP, University of Transport and Communications (UTC) under the project code T2019-CT-04TD, and Bijzonder Onderzoeksfonds (BOF) of Ghent University.

vliruos SHARING MINDS, CHANGING LIVES

With the support of THE BELGIAN DEVELOPMENT COOPERATION .be

References

Abdeljaber, O., Avci, O., Kiranyaz, S., Gabbouj, M. and Inman, D.J. (2017), “Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks”, *J. Sound Vib.*, **388**, 154-170. https://doi.org/10.1016/j.jsv.2016.10.043

Chou, J.H. and Ghaboussi, J. (2001), “Genetic algorithm in structural damage detection”, *Comput. Struct.*, **79**(14), 1335-1353. https://doi.org/10.1016/S0045-7949(01)00027-X

Cuong, N.H. and Quang, N.D. (2020), “Experimental study on flexural behavior of prestressed and non-prestressed textile reinforced concrete plates”. https://doi.org/10.25073/tcsj.71.1.5

Dang, H.V., Tran-Ngoc, H., Nguyen, T.V., Bui-Tien, T., De Roeck, G. and Nguyen, H.X. (2020), “Data-driven structural health monitoring using feature fusion and hybrid deep learning”, *IEEE Transact. Automat. Sci. Eng.* https://doi.org/10.1109/TASE.2020.3034401

Dang, H.V., Raza, M., Tran-Ngoc, H., Bui-Tien, T. and Nguyen, H.X. (2021), “Connection stiffness reduction analysis in steel bridge via deep CNN and modal experimental data”, *Struct. Eng. Mech., Int. J.*, **77**(4), 495-508. https://doi.org/10.12989/sem.2021.77.4.495

DjuriC, P.M. (1996), “A model selection rule for sinusoids in white Gaussian noise”, *IEEE Transact. Signal Process.*, **44**(7), 1744-1751. https://doi.org/10.1109/78.510621

François, S., Schevenels, M., Dooms, D., Jansen, M., Wambacq, J., Lombaert, G., Degrande, G. and De Roeck, G. (2021), “Stabil: An educational Matlab toolbox for static and dynamic structural analysis”, *Comput. Applicat. Eng. Edu.* https://doi.org/10.1002/cae.22391

Eberhart, R. and Kennedy, J. (1995), “A new optimizer using particle swarm theory”, *Proceedings of the 6th International Symposium on Micro Machine and Human Science*, Nagoya, Japan. October. https://doi.org/10.1109/MHS.1995.494215

Ghadimi, S. and Kourehli, S.S. (2017), “Multiple crack identification in Euler beams using extreme learning machine”, *KSCE J. Civil Eng.*, **21**(1), 389-396. https://doi.org/10.1007/s12205-016-1078-0

Ghannadi, P. and Kourehli, S.S. (2019a), “Data-driven method of damage detection using sparse sensors installation by SEREPa”, *J. Civil Struct. Health Monitor.*, **9**(4), 459-475. https://doi.org/10.1007/s13349-019-00345-8

Ghannadi, P. and Kourehli, S.S. (2019b), “Model updating and damage detection in multi-story shear frames using Salp Swarm Algorithm”, *Earthq. Struct., Int. J.*, **17**(1), 63-73. https://doi.org/10.12989/eas.2019.17.1.063

Ghannadi, P. and Kourehli, S.S. (2019c), “Structural damage detection based on MAC flexibility and frequency using moth-flame algorithm”, *Struct. Eng. Mech., Int. J.*, **70**(6), 649-659. https://doi.org/10.12989/sem.2019.70.6.649

Hakim, S.J.S. and Razak, H.A. (2013), “Structural damage detection of steel bridge girder using artificial neural networks and finite element models”, *Steel Compos. Struct., Int. J.*, **14**(4), 367-377. https://doi.org/10.12989/scs.2013.14.4.367

Hien, T.D. (2020), “A static analysis of nonuniform column by stochastic finite-element method using weighted integration approach”, *Transp. Commun. Sci. J.*, 70.

Ho, V.L., Tran, N.H., De Roeck, G., Bui, T.T. and Wahab, M.A., (2018), “System identification based on vibration testing of a steel I-beam”, *Proceedings of International Conference on Numerical Modelling in Engineering* (pp. 254-268), Springer, Singapore. https://doi.org/10.1007/978-981-13-2405-5_21

Hoa, T.N., Khatir, S., De Roeck, G., Long, N.N., Thanh, B.T. and Wahab, M.A. (2020), “An efficient approach for model updating of a large-scale cable-stayed bridge using ambient vibration measurements combined with a hybrid metaheuristic search algorithm”, *Smart Struct. Syst., Int. J.*, **25**(4), 487-499. https://doi.org/10.12989/sss.2020.25.4.487

Hou, G., Xu, Z., Liu, X. and Jin, C. (2019), “Improved particle swarm optimization for selection of shield tunneling parameter values”, *Comput. Model. Eng. Sci.*, **118**(2), 317-337. https://doi.org/10.31614/cmes.2019.04693

Hwang, H. and Kim, C. (2004), “Damage detection in structures using a few frequency response measurements”, *J. Sound Vib.*, **270**(1-2), 1-14. https://doi.org/10.1016/S0022-460X(03)00190-1

Janssens, O., Slavkovikj, V., Vervisch, B., Stockman, K., Loccufier, M., Verstockt, S., Van de Walle, R. and Van Hoecke, S. (2016), “Convolutional neural network based fault detection for rotating machinery”, *J. Sound Vib.*, **377**, 331-345. https://doi.org/10.1016/j.jsv.2016.05.027

Kaveh, A. and Maniat, M. (2015), “Damage detection based on MCSS and PSO using modal data”, *Smart Struct. Syst., Int. J.*, **15**(5), 1253-1270. http://doi.org/10.12989/sss.2015.15.5.1253

Khatir, S. and Wahab, M.A. (2019a), “Fast simulations for solving fracture mechanics inverse problems using POD-RBF XIGA and Jaya algorithm”, *Eng. Fract. Mech.*, **205**, 285-300. https://doi.org/10.1016/j.engfracmech.2018.09.032

Khatir, S. and Wahab, M.A. (2019b), “A computational approach for crack identification in plate structures using XFEM, XIGA, PSO and Jaya algorithm”, *Theor. Appl. Fract. Mech.*, **103**, 102240. https://doi.org/10.1016/j.tafmec.2019.102240

Khatir, S., Belaidi, I., Khatir, T., Hamrani, A., Zhou, Y.L. and Wahab, M.A. (2017), “Multiple damage detection in composite beams using Particle Swarm Optimization and Genetic Algorithm", *Mechanics*, **23**(4), 514-521. https://doi.org/10.5755/j01.mech.23.4.15254

Khatir, S., Tiachacht, S., Thanh, C.L., Bui, T.Q. and Wahab, M.A. (2019), "Damage assessment in composite laminates using ANN-PSO-IGA and Cornwell indicator", *Compos. Struct.*, **230**, 111509. https://doi.org/10.1016/j.tafmec.2019.102240

Khatir, S., Boutchicha, D., Le Thanh, C., Tran-Ngoc, H., Nguyen, T.N. and Abdel-Wahab, M. (2020a), "Improved ANN technique combined with Jaya algorithm for crack identification in plates using XIGA and experimental analysis", *Theor. Appl. Fract. Mech.*, **107**, 102554. https://doi.org/10.1016/j.tafmec.2020.102554

Khatir, S., Khatir, T., Boutchicha, D., Le Thanh, C., Tran-Ngoc, H., Bui, T.Q., Capozucca, R. and Abdel-Wahab, M. (2020b), "An efficient hybrid TLBO-PSO-ANN for fast damage identification in steel beam structures using IGA", *Smart Struct. Syst.*, *Int. J.*, **25**(5), 605-617. https://doi.org/10.12989/sss.2020.25.5.605

Kourehli, S.S. (2017a), "Application of extreme learning machine to damage detection of plate-like structures", *Int. J. Struct. Stabil. Dyn.*, **17**(07), 1750068. https://doi.org/10.1142/S0219455417500687

Kourehli, S.S. (2017b), "Structural damage diagnosis using incomplete static responses and LS-SVM", *Inverse Problems Sci. Eng.*, **25**(3), 418-433. https://doi.org/10.1080/17415977.2016.1169277

Kourehli, S.S. (2018), "Prediction of unmeasured mode shapes and structural damage detection using least squares support vector machine", *Struct. Monitor. Maint.*, *Int. J.*, **5**(3), 379-390. https://doi.org/10.12989/smm.2018.5.3.379

Mai, B.T.T., Cuong, N.H., Quang, N.D. and Tai, D.H. (2020), "Experimental study on flexural and shear behaviour of sandwich panels using glass textile reinforced concrete and autoclaved aerated concrete", *Transp. Commun. Sci. J.*, **71**(1), 18-26.

Maity, D. and Tripathy, R.R. (2005), "Damage assessment of structures from changes in natural frequencies using genetic algorithm", *Struct. Eng. Mech.*, *Int. J.*, **19**(1), 21-42. https://doi.org/10.12989/sem.2005.19.1.021

Mares, C. and Surace, C. (1996), "An application of genetic algorithms to identify damage in elastic structures", *Journal of Sound and Vibration*, **195**(2), 195-215. https://doi.org/10.1006/jsvi.1996.0416

Miguel, L.F.F., Lopez, R.H. and Miguel, L.F.F. (2013), "A hybrid approach for damage detection of structures under operational conditions", *J. Sound Vib.*, **332**(18), 4241-4260. https://doi.org/10.1016/j.jsv.2013.03.017

Na, C., Kim, S.P. and Kwak, H.G. (2011), "Structural damage evaluation using genetic algorithm", *J. Sound Vib.*, **330**(12), 2772-2783. https://doi.org/10.1016/j.jsv.2011.01.007

Nguyen, D.H., Tran-Ngoc, H., Bui-Tien, T., De Roeck, G. and Wahab, M.A. (2020), "Damage detection in truss bridges using transmissibility and machine learning algorithm: Application to Nam O bridge", *Smart Struct. Syst.*, *Int. J.*, **26**(1), 35-47. https://doi.org/10.12989/sss.2020.26.1.035

Samir, K., Brahim, B., Capozucca, R. and Wahab, M.A. (2018), "Damage detection in CFRP composite beams based on vibration analysis using proper orthogonal decomposition method with radial basis functions and cuckoo search algorithm", *Compos. Struct.*, **187**, 344-353. https://doi.org/10.1016/j.compstruct.2017.12.058

Tran-Ngoc, H., Khatir, S., De Roeck, G., Bui-Tien, T., Nguyen-Ngoc, L. and Abdel Wahab, M. (2018), "Model updating for Nam O bridge using particle swarm optimization algorithm and genetic algorithm", *Sensors*, **18**(12), 4131. https://doi.org/10.3390/s18124131

Tran-Ngoc, H., Khatir, S., De Roeck, G., Bui-Tien, T. and Wahab, M.A. (2019a), "An efficient artificial neural network for damage detection in bridges and beam-like structures by improving training parameters using cuckoo search algorithm", *Eng. Struct.*, **199**, 109637. https://doi.org/10.1016/j.engstruct.2019.109637

Tran-Ngoc, H., Khatir, S., De Roeck, G., Bui-Tien, T., Nguyen-Ngoc, L. and Wahab, M.A. (2019b), "Stiffness identification of truss joints of the nam o bridge based on vibration measurements and model updating", *Proceedings of International Conference on Arch Bridges*, pp. 264-272.

Tran-Ngoc, H., He, L., Reynders, E., Khatir, S., Le-Xuan, T., De Roeck, G., Bui-Tien, T. and Wahab, M.A. (2020a), "An efficient approach to model updating for a multispan railway bridge using orthogonal diagonalization combined with improved particle swarm optimization", *J. Sound Vib.*, **476**, 115315. https://doi.org/10.1016/j.jsv.2020.115315

Tran-Ngoc, H., Khatir, S., De Roeck, G., Bui-Tien, T. and Wahab, M.A. (2020b), "Damage assessment in beam-like structures using cuckoo search algorithm and experimentally measured data", *Proceedings of the 13th International Conference on Damage Assessment of Structures*, pp. 380-385. https://doi.org/10.1007/978-981-13-8331-1_27

Tran-Ngoc, H., Khatir, S., Le-Xuan, T., De Roeck, G., Bui-Tien, T. and Wahab, M.A. (2020c), "A novel machine-learning based on the global search techniques using vectorized data for damage detection in structures", *Int. J. Eng. Sci.*, **157**, 103376. https://doi.org/10.1016/j.ijengsci.2020.103376

Tran-Ngoc, H., Khatir, S., Ho-Khac, H., De Roeck, G., Bui-Tien, T. and Wahab, M.A. (2021a), "Efficient Artificial neural networks based on a hybrid metaheuristic optimization algorithm for damage detection in laminated composite structures", *Compos. Struct.*, **262**, 113339. https://doi.org/10.1016/j.compstruct.2020.113339

Tran-Ngoc, H., Khatir, S., Le-Xuan, T., De Roeck, G., Bui-Tien, T. and Wahab, M.A. (2021b), "Finite element model updating of a multispan bridge with a hybrid metaheuristic search algorithm using experimental data from wireless triaxial sensors", *Eng. Comput.*, 1-19. https://doi.org/10.1007/s00366-021-01307-9

Tiachacht, S., Bouazzouni, A., Khatir, S., Wahab, M.A., Behtani, A. and Capozucca, R. (2018), "Damage assessment in structures using combination of a modified Cornwell indicator and genetic algorithm", *Eng. Struct.*, **177**, 421-430. https://doi.org/10.1016/j.engstruct.2018.09.070

Tran, N.H. and Bui, T.T. (2019), "Damage detection in a steel beam structure using particle swarm optimization and experimentally measured results", *Sci. J. Transport.*, (9), 3-9.

Xu, H.J., Ding, Z.H., Lu, Z.R. and Liu, J.K. (2016), "Structural damage detection using a modified artificial bee colony algorithm", *CMES-Comput. Model. Eng. Sci.*, **111**, 335-355.

Yazdanmehr, M., Anijdan, S.M. and Bahrami, A. (2009), "Using GA–ANN algorithm to optimize soft magnetic properties of nanocrystalline mechanically alloyed Fe–Si powders", *Computat. Mater. Sci.*, **44**(4), 1218-1221. https://doi.org/10.1016/j.commatsci.2008.08.003

Zenzen, R., Belaidi, I., Khatir, S. and Wahab, M.A. (2018), "A damage identification technique for beam-like and truss structures based on FRF and Bat Algorithm", *Comptes Rendus Mécanique*, **346**(12), 1253-1266. https://doi.org/10.1016/j.crme.2018.09.003

Zhou, Y.L. and Wahab, M.A. (2017), "Damage detection using vibration data and dynamic transmissibility ensemble with auto-associative neural network", *Mechanics*, **23**(5), 688-695. http://dx.doi.org/10.5755/j01.mech.23.5.15339

CC