Discover Civil Engineering

Research

Automated damage assessment of prestressed concrete bridge beam based on sagging

Khatereh Dakhili[1] · Sebastian Schommer[1] · Stefan Maas[1]

Received: 4 March 2024 / Accepted: 21 August 2024
Published online: 28 August 2024

Abstract
The appropriate bridge maintenance strategy cannot be determined unless the damage is identified, localized, and quantified correctly. Damage assessment can be performed based on model updating, where material properties of a numerical model are modified to represent the damaged state as accurately as possible. However, this approach may become tedious for complex structures such as bridges due to the high number of unknown variables. This study replaces the time-consuming Finite Element (FE) simulations with Artificial Neural Network (ANN) as a surrogate model to reduce the required computational time. The implementation of ANN enables automating the existing manual damage assessment of a prestressed concrete bridge beam. In this paper, the objective is to minimize the difference between the simulated and measured sagging, which is the irreversible downward movement of the bridge due to its weight. The minimization is performed with the Simulated Annealing (SA) algorithm, and the optimization process is repeated with 100 different starting points to ensure robustness. The results indicate that the automated approach performs similarly to the manual approach while being faster and enabling wider exploration in the search space without compromising accuracy. The proposed approach serves as a practical tool for real-world problems by offering an efficient damage assessment.

Keywords Bridge damage detection · Model updating · Sagging · Artificial neural networks · Optimization

1 Introduction

Damage assessment of bridges has become increasingly important due to the growing transportation and economic development. Various kinds of damages are expected to occur during the bridge service life because of aging and material deterioration as well as natural hazards and environmental influences. The goal is to detect damages right in time to prevent catastrophic failures by implementing the most appropriate maintenance strategy [1–3].

The best bridge maintenance strategy cannot be determined unless the damage is identified, localized, and quantified with the highest accuracy. In this regard, one major challenge is temperature effects that change dynamic and static properties in the same order of magnitude as damage [4–6]. These influences should be either integrated into the utilized damage assessment method, or they should be removed from the measurements to ensure successful damage detection.

An example of integrating thermal responses into the damage assessment workflow can be found in [7], where an analytical model was developed to investigate the strain and deflection distributions within a small-scale prestressed concrete beam. While including temperature changes in numerical models allows for predicting thermal responses, employing time series analysis can be highly complicated for real-scale case studies. Besides, large datasets are

✉ Khatereh Dakhili, khatereh.dakhili@uni.lu; Sebastian Schommer, sebastian.schommer@uni.lu; Stefan Maas, stefan.maas@uni.lu | [1]Faculty of Science, Technology and Medicine, University of Luxembourg, 6, Rue Coudenhove-Kalergi, L-1359 Luxembourg, Luxembourg.

essentially required to represent the possible temperature ranges [8]. More details about the performance of various temperature-based damage detection techniques can be found in [9].

In order to deal with the complexities associated with including temperature influences, various techniques are developed to remove the temperature influences from the measurements. These techniques are based on the correlation between the temperatures and the structural responses. Then, regression analysis [10, 11] or statistical tools such as Principal Component Analysis (PCA) [12, 13] are utilized to eliminate thermal effects.

The state-of-the-art bridge damage assessment technique is the visual inspection. Nevertheless, this widely applied method has several shortcomings. Surface-level assessment without capturing the hidden damages and variable interpretation due to the subjective judgment of the inspector are two of the most important reasons why visual inspection should be amended with more advanced techniques [14].

Vibration-based damage assessment can assist visual inspection [15–17]. However, damage indicators based on vibrational characteristics are not always sensitive enough to detect damage at early stages [15]. Therefore, the current paper focuses on static measurements, particularly sagging, which is a powerful damage indicator suitable for real-world applications. Sagging is defined as the progressive and irreversible downward movement of the bridge due to its own weight, traffic loads, and environmental effects.

Damage assessment can be treated as a model updating problem. First, a Finite Element (FE) model is prepared and validated with the experiments in the undamaged state. Later, if any deviation occurs between the experimental and numerical characteristics, the material properties are adjusted to minimize this deviation. After updating the material properties correctly, the required information is obtained about the damage location and severity [18].

Although this approach involves the computational effort of developing an FE model, it is suitable for real-world applications, and it can be coupled with numerous optimization algorithms [18]. For instance, the damage assessment of a prestressed concrete beam taken from a real bridge was performed with this approach in [19, 20]. Despite introducing progressive artificial damage to the beam, no cracks were visible at the early stages. Nevertheless, the damage was successfully localized and quantified with FE model updating.

Finding the best values of material properties to minimize the deviation between the measurements and simulations is a trial-and-error process. This process soon becomes tedious for complex Finite Element (FE) models with numerous unknown parameters and vast search spaces. Thus, this paper focuses on improving the model updating technique proposed by [19], where the authors minimized the differences manually. This paper aims to perform an automatic minimization to reduce the required computational efforts.

The automation of the model updating process requires an objective function. This function corresponds to the difference between the numerical and experimental results. Instead of using the high-fidelity FE model with a significant number of unknown variables in every evaluation of the objective function, a surrogate model can be utilized to replace time-consuming simulations while maintaining the intended level of accuracy. An overview of the recent advances in surrogate modeling is provided in [21–23].

This paper implements Artificial Neural Networks (ANNs) as a surrogate model to replace the FE model. The structure of ANNs can be easily adjusted to the complexity of the investigated problem to achieve low approximation errors [21, 24, 25]. ANNs are also suggested as the best option when the final goal is to decrease the required computational time [21]. Moreover, the generated function with the ANNs can be easily integrated with advanced optimization algorithms to perform the subsequent minimization in the model updating process.

In this study, the Simulated Annealing (SA) algorithm is utilized for optimization since it is suitable for complex objective functions with wide search spaces by offering a balance between the exploration and exploitation phases. SA allows for the exploration of less favorable solutions in the early stages. Consequently, it avoids being stuck at local optima and searches for the global optimum. As the exploration continues, fewer worse solutions are allowed, focusing on exploitation and finding the best solution. Further details about the SA terminology and workflow are presented in [26].

As stated, the goal of this paper is to automate the damage detection of the bridge beam presented by [19]. The strength of the proposed approach lies in its simplicity, making it appropriate for real-world complex problems. First, the experimental setup and the numerical model are briefly described. Then, an Artificial Neural Network (ANN) is trained with the available numerical dataset. It is explained how ANN replaces the FE model, and the objective function is minimized with the Simulated Annealing (SA) algorithm. The minimization is repeated with 100 different starting points to ensure the robustness of the optimization. Lastly, the automated damage detection is compared with the manual approach.

2 The investigated beam

The experimental setup is introduced in detail by [11, 19]. Here, a brief description is provided.

2.1 Experimental setup

In 2013, the prestressed concrete bridge connecting Grevenmacher (Luxembourg) and Wellen (Germany) was demolished for safety concerns. A beam was taken under study for damage assessment. This beam was about 46m long with an approximate weight of 120t. The beam had 19 longitudinal tendons that remained intact after the bridge's demolition. Nevertheless, the transverse tendons were cut and lost their prestress.

In order to replicate the original conditions of the bridge beam, it was supported by a fixed and sliding bearing. Although the asphalt layer was removed, a dead load of 30t was put on the bridge to represent the asphalt layer and the demolished superstructures such as the sideway. The dead load was shown to be a satisfactory approximation of the removed parts. Moreover, two concrete blocks of 13t each were used to represent the live loads caused by traffic. These blocks were put on the bridge and removed after at least 24 h before repeating the procedure. The investigated beam together with the loads is shown in Fig. 1.

The displacements were measured with 8 transducers. Seven sensors (SV1-SV6 & SV8) measured the vertical displacements, and one sensor (SH7) measured the horizontal displacement at the sliding bearing. Seven temperature sensors (T1-T7) recorded temperatures at different points. The air temperature was measured with the T_{amb} sensor. The measurements were performed during 1 month in winter. More details on the conducted experiments can be found in [27]. Figure 2 presents a schematic diagram of the experimental setup.

2.2 Temperature compensation

This paper works with the available temperature-compensated measurements from the previous study [11]. Temperature effects were removed before damage assessment using a projection-based temperature compensation technique. Since the correlations between the concrete temperature (T4) and the displacements were strongly linear, the mentioned compensation technique calculated the linear regression slope. Using this information, the displacements were mapped to the reference temperature, $T_{ref} = 5°C$, which represented the average ambient temperature of the testing period.

2.3 Damage Scenarios (DS)

Five Damage Scenarios (DS) were investigated, which are shown in Fig. 3. The initial scenario (DS0) corresponded to a healthy state with no damage to the bridge beam. The progressive artificial damage was introduced by cutting the tendons 5m off the middle of the mean towards the sliding bearing, i.e., the cutting line shown in Fig. 2. As can be seen in Fig. 3, two, four, and six tendons were cut in DS1-DS3, respectively. In DS4, three tendon pairs were cut in half. Then, the experiments of the healthy state were repeated in each DS followed by temperature compensation. Finally, the temperature-compensated values were used for model updating.

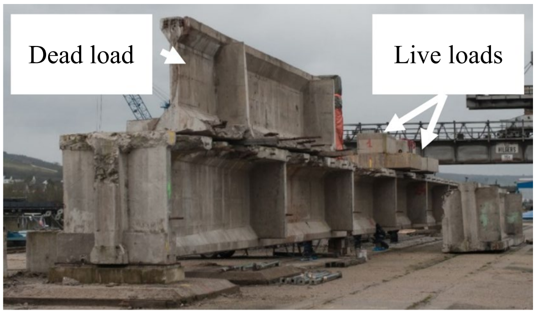


Fig. 1 The experimental setup [27]

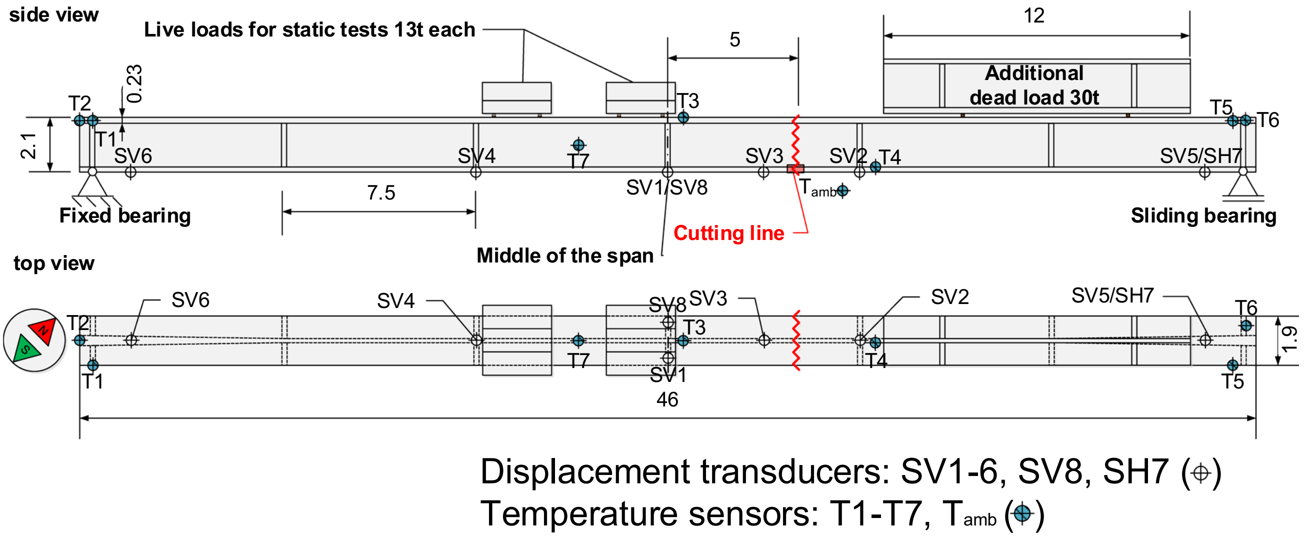
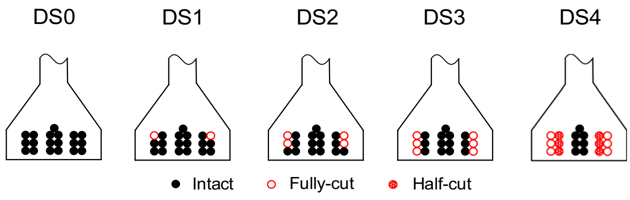


Fig. 2 Schematic diagram of the sensor locations

Fig. 3 Investigated Damage Scenarios (DS)



3 Numerical model

It was assumed that damage reduces the stiffness locally. The idea was to identify, locate, and quantify damage by modeling the stiffness reduction with a Finite Element (FE) model. Hence, a linear FE model was developed using ANSYS Parametric Design Language (APDL). The model consisted mainly of SOLID185, SOLID186, and BEAM188 elements. Since the model was intended to be linear, contact elements CONTA174 and TARGET170 were utilized with the MPC algorithm and always-bonded condition. These elements were also utilized to model the fixed bearing. In addition to the mentioned contact elements, MPC184 revolute and planer joints were used for modeling the sliding bearing. The tendons were modeled without including prestressing. The tendons (BEAM188) were connected to the concrete (SOLID186) using dummy beams with the same properties as the tendons except for having zero density. The material properties and elements are listed in Tables 1 and 2, respectively.

For further information about FE model, refer to [27, 28]. In the following, two key features of the FE are explained. These features are relevant to comparing the automated and manual model updating approaches.

3.1 FE model with mapped mesh

Damage detection based on model updating requires hundreds of repeated simulations with different material properties to find the one that matches the experimental results the best. Complex structures, such as the bridge beam under study, have millions of elements. Hence, damage detection soon becomes a demanding task.

The first measure to decrease the number of variables is the implementation of mapped mesh. The elements are divided into slices in the longitudinal direction, such that each slice covers the entire cross-section of the beam. Then, it is assumed that the stiffness/Young's modulus is constant within each slice; so that the damage location along the beam length can be identified. As a result, it suffices to only update Young's modulus of the slice instead of modifying its value for all the elements one by one. Subsequently, the unknown variables reduce from millions to the number of slices created in the model, i.e., 227.

Table 1 Material properties of the FE model

	Youngs Modulus $E[MPa]$	Density $\rho[kg/m^3]\}$	Poisson ratio	Description
Concrete	3×10^4	2500	0.13	Concrete parts of the model that remain unchanged after model updating, such as the additional and dead load
	3×10^4 (initial value)	2500	0.13	Concrete beam, Young's modulus of the slices changes after model updating to identify, localize, and quantify the damage
	2×10^4	2500	0.13	Fresh concrete, properties used for model parts representing bearings and foundation of the test setup
Steel	2×10^5	7850	0.3	Tendons
Dummy beams	2×10^5	0	0.3	Beams used for connecting tendons to concrete
Wood	1×10^4	470	0.2	Wood parts under the loads
Soil	3×10^3	2500	0.13	Simulated soil under the test setup

Table 2 Summary of element types used in the FE model

Element type	Usage	Node number
SOLID186	Hexahedral elements for mapped meshing	20
SOLID185	Similar usage as SOLID186 for less important parts	8
SHELL181	2D elements for mapped meshing without mid-side nodes	4
SHEL281	2D elements for mapped meshing and subsequent extrusion to 3D mesh	8
BEAM188	Beam elements for tendons and dummy beams, Timoshenko beam theory	2
CONTA174	Contact definition in different areas, MPC algorithm (always bonded)	8
TARGE170	Target elements for CONTA174 elements	Up to 8
MPC184	Multi-point constraint element, revolute and planer joint to model sliding bearing	2

Figure 4 shows a section of the FE model. The changing colors indicate the elements grouped as slices. There are 227 slices in the model. The colors are repeated every eleven slices, yet each can have unique Young's modulus values at the end of the model updating process. The width of each slice is about 20cm. As discussed by [27, 28], this mesh size is sufficient for a coarse localization of the damage in the longitudinal direction.

3.2 Damage function

Although the number of variables decreases significantly using the mapped mesh, it is still too high for a model updating problem. Therefore, a damage function is implemented in the FE model to describe the stiffness/Young's modulus distribution.

In [19, 27], it is explained how selecting the Gaussian bell curve as the damage function allows defining Young's modulus distribution with only three parameters, namely, p, μ, and σ. These parameters can be easily associated with the magnitude, location, and width of the damage, respectively. The damage function is formulated as in Eq. (1).

$$a(x_{norm}, p, \mu, \sigma) = p \cdot e^{-0.5\left(\frac{x_{norm}-\mu}{\sigma}\right)^2} \tag{1}$$

where a : Young's modulus reduction ratio; p: maximal reduction, $p \in [0\ 1]$; μ: position of the damage (the peak of the bell curve), $\mu \in [-1+1]$; σ: width of the Gaussian bell curve.

$$x_{norm} = \frac{2x - x_s - x_e}{x_e - x_s}, x \in \left[x_s x_e\right],$$

In Eq. (1), x_{norm} is the normalized longitudinal axis of the beam. x can vary between 0 and 46 m. Thus, $x_s = 0$ and $x_e = 46$. Since the tendons are cut 5m to the right of the center of the beam as shown in Fig. 2, the normalized damage location is $\mu = 0.2$. Similar to [19, 27], the whole bridge beam is considered the damage element for coarse damage localization. An example of damage function is provided in Fig. 5.

Fig. 4 FE model with mapped mesh and the elements divided into slices

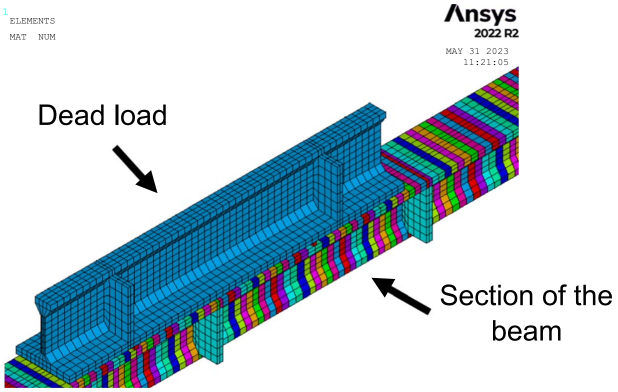
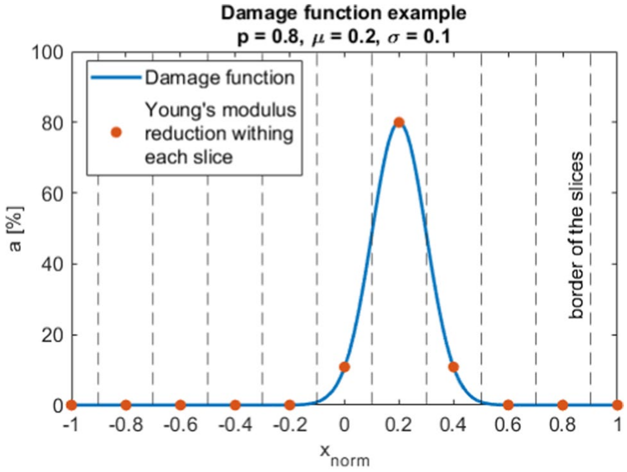


Fig. 5 Example of the damage function



Consequently, instead of updating Young's modulus of each slice separately (i.e., 227 variables in this case), only 3 values of p, μ, and σ should be adjusted. This solution can be also extended to other bridges, e.g., box girder bridges, if their numerical model has a mapped mesh, in which the elements can be divided into slices. Note that the damage function calculates the stiffness reduction at the center of each slice. Therefore, within each slice, the damage localization can have an offset of 10 cm, i.e., half of the slice width.

4 Manual damage assessment

The objective function and the optimization process of the manual approach are briefly explained in the following. Refer to [19, 27] for more details.

4.1 Objective function

The objective function was defined based on the Root Mean Squared Error (RMSE) between the numerical and experimental values. As stated, the unknown variables are the 3 damage function parameters (p, μ, σ), describing Young's modulus reduction over the longitudinal axis of the bridge beam. The outputs are the temperature-compensated vertical (SV1-SV6) and horizontal (SH7) displacements due to sagging. As explained before, sagging is the constant downward movement of the bridge due to its own weight. Therefore, the reference values (e.g., the measurements at the beginning of the test period) are subtracted from the measurements in the damaged states to calculate the displacements due to sagging. Next, the objective function is formulated as:

$$obj\ F(p,\mu,\sigma)=\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(sagg_{exp,i}-sagg_{sim,i}\right)^2} \tag{2}$$

In Eq. (2), $N = 7$, which is the number of the outputs/sensors. As can be seen, all the displacement sensors have an equal weight. If one sensor measurement was faulty, or if a certain location was more prone to damage, the weights could have been decreased or increased accordingly. Nevertheless, the equal weights led to satisfactory results in the investigated problem.

4.2 Manual optimization

Next, the best set of input variables (p, μ, σ) is found such that the difference between the experimental and numerical values is minimized. Previously, this task was completed manually; meaning that the objective values were plotted versus one or two of the inputs while assuming the remaining one(s) are constant. The trends were investigated in each plot, and more simulations were performed until the minimum of the objective function was found. This task was completed after 2280 simulations with the described coarse FE model [19, 27].

5 Automated damage assessment

Despite being straightforward, the manual approach has several drawbacks. For instance, it is not possible to investigate all the possible combinations of p, μ, and σ in the search space. Moreover, if damage assessment is to be extended to multi-span bridges with several beams, the manual approach soon becomes cumbersome due to the increase of required simulations.

This study aims to implement Artificial Intelligence (AI) into model updating to formularize the objective function and automate the optimization process. The AI model replaces the FE model, and it has the same inputs (p, μ, σ) and outputs (displacements). This replacement reduces the simulation time from a few minutes to only a few seconds. Since the number of the required simulations in a model updating problem usually reaches up to thousands, the benefit of employing an AI model is pronounced. As a result, not only the optimization can be completed quicker, but it can also become more effective since the modified objective function can be coupled with advanced optimization algorithms.

5.1 Workflow

Initially, the RegressionLearner App in MATLAB is used to investigate different multi-input single output models such as linear regression models, regression trees, the ensemble of trees, support vector machines, Gaussian Regression Process (GPR) models, and Artificial Neural Networks (ANNs). ANNs are chosen due to their promising performance regarding the estimation errors and required training time. Additionally, they can be easily converted to multi-input multi-output models. In the following, the training and testing of the ANN are explained.

5.1.1 Dividing dataset

Ideally, the training dataset roughly covers the whole search space of the input variables (p, μ, σ). This implies performing the simulations within reasonable ranges of variables with preselected steps.

The manual approach employed a different search strategy as explained in Sect. 4.2. The input–output dataset is collected based on intermediate checks. Therefore, if the location of the damage was different, another dataset should have been gathered. This is regarded as a shortcoming of the manual approach since the search space is only partially investigated by relying on time-consuming FE simulations.

While it is suggested to utilize a comprehensive training dataset, this paper employs the available numerical dataset from the manual approach for training the ANN to demonstrate the strength of the automated approach. Note that using the available dataset does not impose a dependency on the manual approach. If the ANN is successfully trained even with a partial representation of the search space, it can calculate the displacements given any input values (p, μ, σ). Nevertheless, if the results are not satisfactory, a different training dataset should be utilized.

Here, out of 2280 input–output pairs from the numerical model, 15% of the data (342 pairs) is set aside for the final testing of the model. The remaining data (1938 pairs) is considered for training with the fivefold cross-validation to avoid overfitting.

5.1.2 Pre-processing

Pre-processing should be performed carefully to facilitate training the best possible model. The most important steps include data cleaning, normalization, and feature selection. These steps ensure that the inputs contribute equally to the model. Since the final model will be used for damage assessment, no outliers are removed from the data as they can represent a damaged state. In this study, inputs are normalized such that they all vary from 0 to 1.

With feature selection techniques, only the most important features are selected, and the ones that do not improve the model performance are discarded [29]. Since there are only three features (p, μ, σ) in this study, no feature selection is performed.

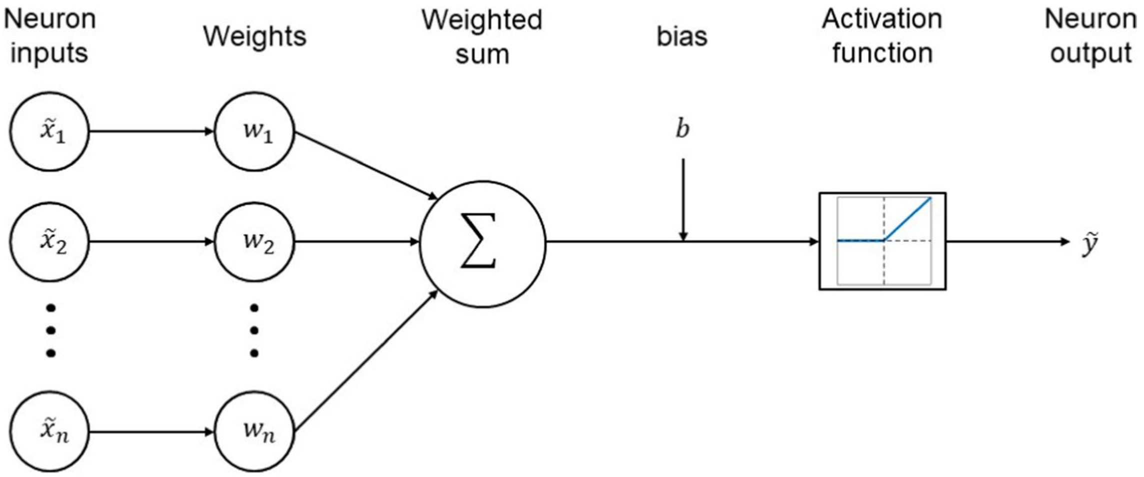
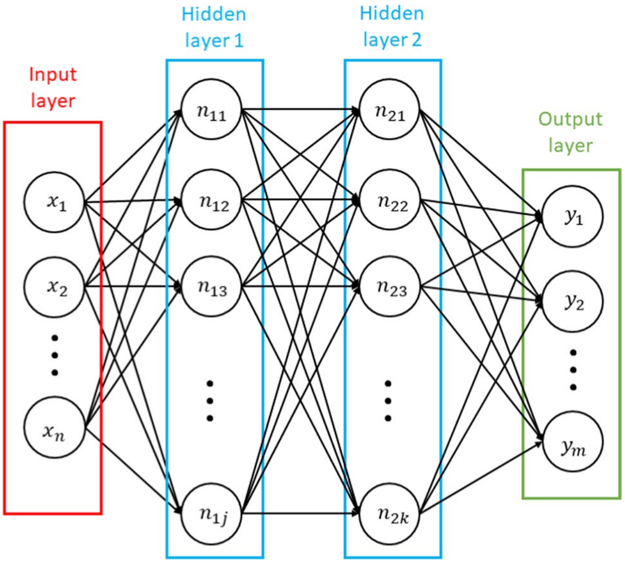


Fig. 6 Configuration of an artificial neuron

Fig. 7 Configuration of an Artificial Neural Network (ANN) with 2 hidden layers



5.1.3 Model configuration

As mentioned, Artificial Neural Network (ANN) is selected for this study. The fundamental component of ANN is a single artificial neuron, also referred to as a perceptron [30, 31]. For better understanding, the configuration of a single neuron is demonstrated in Fig. 6.

The demonstrated neuron in Fig. 6 is a mathematical model that tries to estimate the function between the given inputs ($\widetilde{x}_1, \widetilde{x}_2, \ldots, \widetilde{x}_n$) and output ($\widetilde{y}$). As can be seen, the summation of the weighted inputs and bias (b) are fed into an activation function to calculate the output value. Bias serves as a threshold to determine whether the activation output of the neuron is propagated forward. Finally, the activation function restrains the neuron output within a practical range [30, 31]. Here, the Piecewise Linear activation function (ReLU) is shown as an example. Other transfer function examples are step function, Sigmoid function, and hyperbolic tangent (tanh) function.

When several neurons are combined, they form a Multi-Layered Perceptron (MLP) or an Artificial Neural Network (ANN). A common structure of a fully connected feedforward ANN is illustrated in Fig. 7. Each ANN consists of the input layer, hidden layer(s), and output layer. ANN estimates the relation between the inputs and outputs by tuning weights and bias values during the training phase. The number of the hidden layers, the number of neurons of each layer, and the activation function are among the hyperparameters that are selected before the training phase considering the problem to be solved. A hyperparameter optimization is essential to find the characteristics that lead to the best performance.

In this study, ANN is implemented to approximate the function between the damage function parameters (p, μ, σ) and displacements due to sagging. As explained in Sect. 3.2, the damage function describes Young's modulus reduction along the beam. Therefore, once an approximate function between the displacements and Young's modulus reduction is established, damage identification, localization, and quantification can be achieved.

The built-in ANN configurations of MATLAB RegressionLearner App are networks with 1, 2, and 3 layers, each layer with 10 neurons. Among these configurations, the 2-layered ANN provides a satisfactory balance between the required training time and approximation accuracy. Similarly, hyperparameter optimization leads to achieving higher accuracies with more complex configurations. In this case, one must be also cautious about overfitting and testing the model with more data. Here, the goal is to train the ANN with only the available dataset to make a reasonable comparison between the automated and the manual approach. Moreover, the ANN will be coupled with an optimization algorithm, and numerous function evaluations are required. Thus, the 2-layered ANN ($j = k = 10$ in Fig. 7) is chosen to avoid time-consuming optimizations and overfitting.

The other ANN characteristics are as follows. Piecewise Linear function (ReLU) is selected as the activation function. The output of this function is equal to 0 if the input is negative. Otherwise, its output is equal to the input value. Additionally, the Levenberg–Marquardt training function is chosen. The iteration limit and the regularization strength are 1000 and 0, respectively.

5.1.4 Model evaluation

Various metrics are used to evaluate the performance of a model. In this study, the Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE) are investigated to select the best model. The model hyperparameters are selected such that the error metrics are acceptable. These metrics are calculated according to the Eqs. (3)–(5):

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}_i| \tag{3}$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N} \left(y_i - \hat{y}_i\right)^2 \tag{4}$$

$$RMSE = \sqrt{\left(\frac{1}{N}\sum_{i=1}^{N} \left(y_i - \hat{y}_i\right)^2\right)} \tag{5}$$

where: y_i: the actual output; $\hat{y}_i$: the predicted output; N: number of the outputs.

5.1.5 Testing

The trained model is validated with the fivefold cross-validation technique to prevent overfitting. Additionally, the model is tested with data that it has not seen before. The same pre-processing steps applied to the training data are applied to the test data. The model's ability to make predictions with the new data is assessed in this phase. If its performance is not satisfactory, the model configuration is adjusted accordingly, and the training is repeated. The final model is implemented in the model updating procedure.

5.2 Replacement of FE model by ANN

The ANN replaces the Finite Element (FE) model by providing a function between the inputs (p, μ,σ) and outputs (displacements due to sagging). Thus, each evaluation of the objective function (Eqs. (2)) does not require performing an FE analysis anymore. Instead, the displacements are quickly obtained for any combination of p, μ, and σ with the available function from the trained ANN. This modification, which enables the automated model updating, is the main contribution of the current paper.

5.3 Automized optimization

The minimum of the generated objective function can now be found with an appropriate optimization algorithm. Five optimization algorithms are investigated in terms of damage localization accuracy. To do so, the optimization result for μ is compared with the location of the introduced damage, $\mu = 0.2$. Note that this comparison is only possible because the damage was artificially introduced at a preselected location (Fig. 2). The investigated algorithms being as follows: fmincon, Genetic Algorithm (GA), pattern search, particle swarm, and Simulated Annealing (SA). Except for the fmincon method, the rest of the algorithms are non-deterministic; meaning that they lead to a different solution with a different starting point. Thus, the optimization is repeated with 100 randomly generated starting points to increase the robustness.

In this step, the Simulated Annealing (SA) algorithm was selected as it led to the lowest damage localization error. As explained, SA explores worse solutions more freely in the beginning to find the global optimum. As the optimization continues, it then focuses on improving the current solution instead. Inspired by the physical annealing process, SA considers an initial temperature, T_0 to calculate the probability of accepting a less favorable solution according to Eq. (6) [26].

$$\frac{1}{1+\exp\left(\frac{\textit{new objective value} - \textit{old objective value}}{\textit{current temperature}}\right)} \tag{6}$$

where: Current temperature: $T = T_0 \times 0.95^k$

Annealing parameter:

$$k = \log\left(\frac{T_0}{T}\frac{\textit{maximum gradient of objective function accross all variables}}{\textit{gradient of objective function of variable i}}\right)$$

Several default properties of the utilized SA algorithm are listed in Table 3, where the number of the variables is equal to 3 (p, μ, σ).

6 Results and discussion

In the manual approach, the objective function based on sagging is plotted versus the input variables (p, μ, σ) to find the minimum. Here, the normalized location of the introduced artificial damage is $\mu = 0.2$. Thus, the objective function is initially plotted versus p and σ with fixed $\mu = 0.2$. The corresponding plots in all damage scenarios (DS1-DS4) are provided in Fig. 8. Then, the minimum of the objective function is identified in each DS. Once the corresponding p and σ are determined, the influence of μ on the objective function is investigated while p and σ are fixed to the values determined before. This step is presented in Fig. 9.

According to Fig. 9, the minimum of the objective function deviates from $\mu = 0.2$, which is the actual location of the introduced damage. Before adjusting the μ values and repeating the step explained with Fig. 8, the influences of μ and p on the objective function are simultaneously studied while σ fixed to values indicated in Fig. 9. As an example, this investigation is only demonstrated for DS4 in Fig. 10.

Figure 10 provides a comprehensive overview of the objective function. Considering the results, $\boldsymbol{\mu}$ is adjusted, and the mentioned steps are repeated (Figs. 8, 9, 10) until p, μ, and σ values associated with the minimum of the objective function are manually identified. The final results of the manual approach are presented in Table 5.

With the automated approach, a 2-layered ANN, each layer with 10 neurons, estimates the displacements based on p, μ, and σ. The error metrics of the ANN are presented in Table 4. The displacement sensors are listed in the same

Table 3 Properties of the implemented SA algorithm

Property	Value
Minimum function tolerance	$1e-6$
Number of stall iterations (the change in the objective function is smaller than the minimum function tolerance)	500 × *number of variables*
Maximum number of function evaluations	3000 × *number of variables*
Initial Temperature	100

Fig. 8 Objective function due to sagging versus ***p*** and ***σ*** - ***μ*** = 0.2

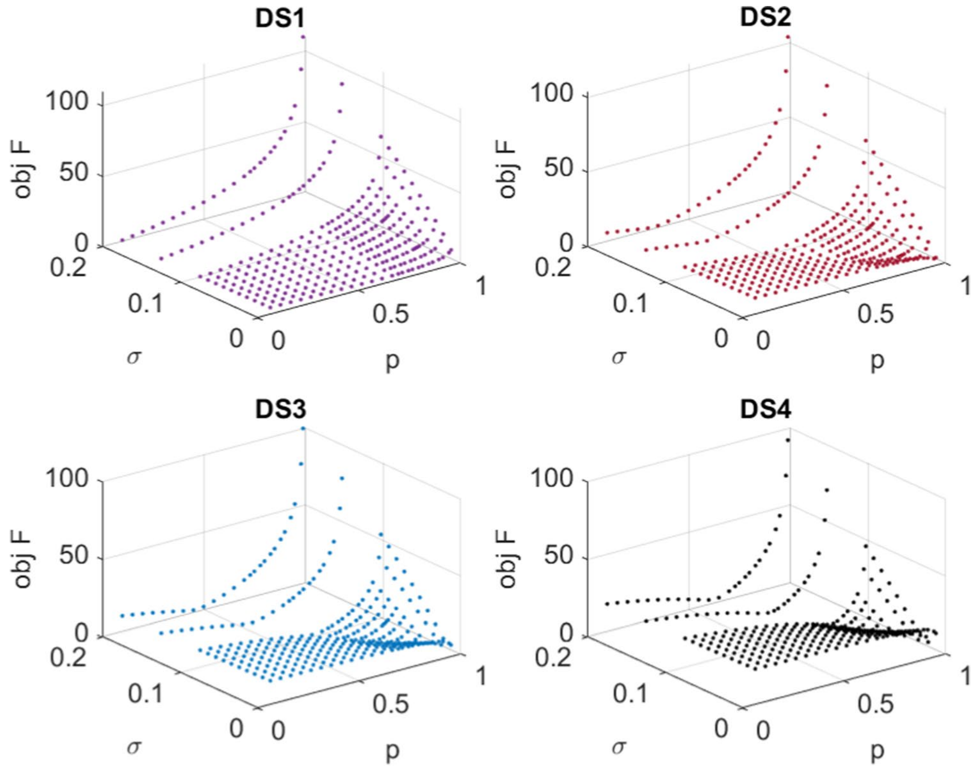


Fig. 9 Objective function due to sagging versus ***μ*** - ***p*** and ***σ*** fixed

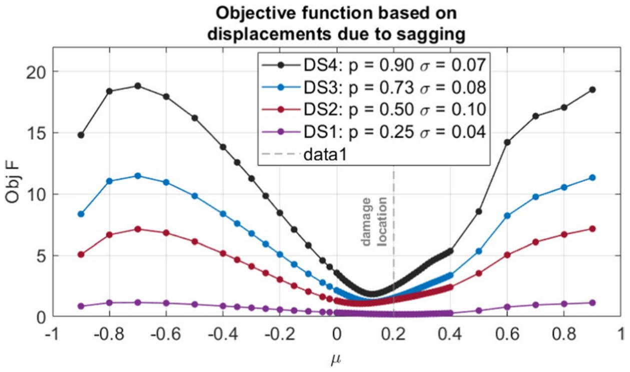


Fig. 10 Objective function due to sagging versus ***p*** and ***μ*** - fixed ***σ***

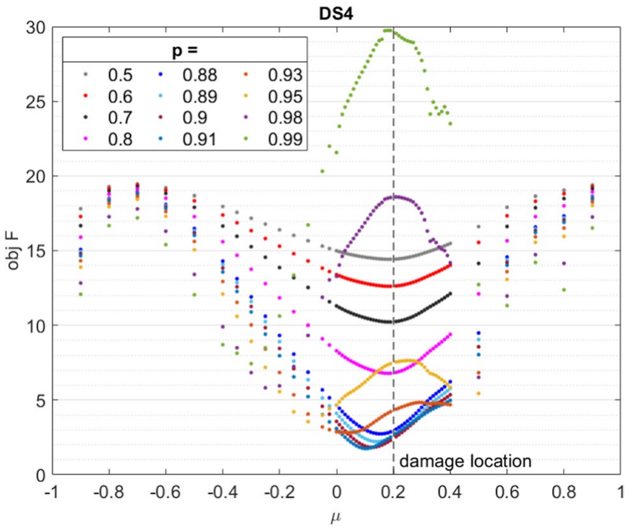


Table 4 Error metrics of the ANN estimating the displacement values

Sensors		SV6	SV4	SV1	SV3	SV2	SV5	SH7
MAE	Train	3.20E−05	1.75E−04	2.39E−04	2.53E−04	2.56E−04	8.63E−05	1.24E−04
	Test	4.08E−05	2.73E−04	4.02E−04	4.44E−04	4.21E−04	1.02E−04	1.32E−04
MSE	Train	6.76E−09	1.47E−07	3.18E−07	3.30E−07	2.79E−07	2.49E−08	2.62E−08
	Test	1.07E−08	5.87E−07	1.55E−06	1.89E−06	1.58E−06	2.96E−08	2.80E−08
RMSE	Train	8.22E−05	3.84E−04	5.64E−04	5.74E−04	5.28E−04	1.58E−04	1.62E−04
	Test	1.03E−04	7.66E−04	1.24E−03	1.38E−03	1.26E−03	1.72E−04	1.67E−04

Table 5 Comparison of the manual and automated damage assessment of the prestressed concrete beam

	Manual approach			Automated approach		
DS#	p	μ	σ	p	μ	σ
1	0.25	0.20	0.04	0.27	0.21	0.03
2	0.50	0.09	0.10	0.61	0.00	0.08
3	0.73	0.13	0.08	0.79	0.11	0.07
4	0.90	0.13	0.07	0.90	0.08	0.07

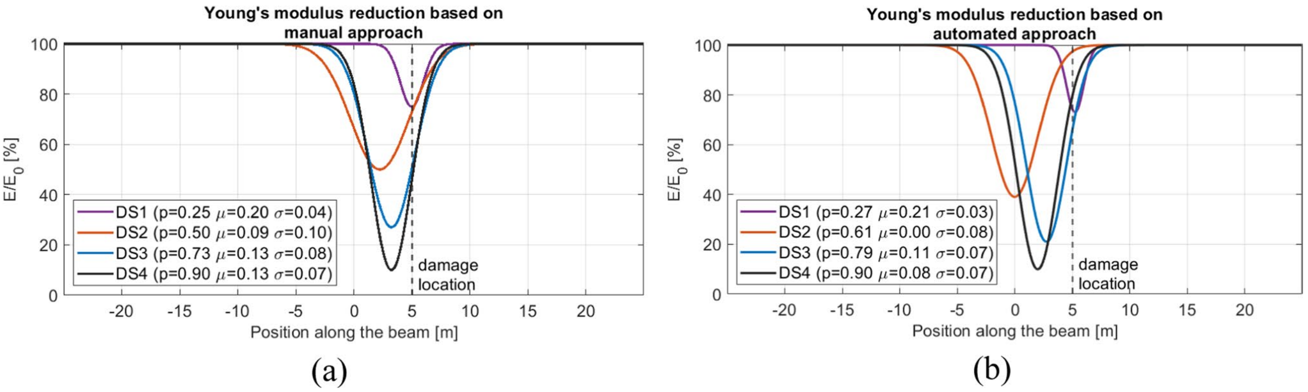


Fig. 11 Young's modulus reduction along the beam **a** manual **b** automated approach

order as in the experimental setup. Here, the highest MAE and RMSE values are one order of magnitude smaller than the displacement values. Therefore, this ANN is expected to be an acceptable substitute for the FE model.

After defining the objective function, the minimization is performed using the Simulated Annealing (SA) algorithm as explained before. Table 5 provides a comparison between the automated and manual damage assessment results. The identified values of damage magnitude (p) and damage width (σ) are almost equal in all Damage Scenarios (DS). However, the damage localization accuracy is only preserved in DS1 and DS3. The error regarding the damage localization increases slightly in DS2 and DS4.

Next, Young's modulus reduction along the beam is plotted in Fig. 11 based on the Eq. (1). The increasing values of p and σ from DS1–DS4 indicate the growing damage magnitude and width, respectively. The artificial damage was at $\mu = 0.2$. The goal is to detect the damage in early scenarios (DS1 & DS2) as accurately as possible. Although the cracks were not visible in these scenarios, both the automated and manual approaches precisely localized the damage in DS1. Since the damage localization accuracy is the lowest in DS2, it might be possible that the measurements in this scenario were erroneous.

Similarly, only high success rates are acceptable in DS3 and DS4 because any mistakes in detecting severe damages lead to catastrophic failures. Therefore, not only the location of the damage should be determined precisely, but higher values should be observed for damage magnitude (p) and damage width (σ). According to Table 5 and Fig. 11, both approaches detect the damage successfully in these scenarios.

7 Conclusion

Aging and exposure to environmental factors have made timely bridge damage detection increasingly important. Despite ongoing research in this area, visual inspection remains the state-of-the-art damage assessment method. However, visual inspection alone cannot detect hidden damage at early stages. To address this issue, this study suggests utilizing static measurements as a practical complement to visual inspections.

The displacements due to sagging have been shown as beneficial damage indicators combined with the model updating approach. Here, the same experimental and numerical dataset provided in [19, 27] is utilized. However, the model updating process was previously performed manually, which was less efficient. Therefore, the model updating process was automated in this study.

To do so, the Finite Element (FE) model was replaced with an Artificial Neural Network (ANN) as a surrogate model. The implementation of the ANN significantly reduced the required computational time while maintaining a low approximation error. The ANN was trained and tested with the available numerical data. Then, the objective function was defined and minimized using the Simulated Annealing (SA) method to find the material properties best representing the damaged state. As a result, the required information was obtained about damage location and severity.

The ANN combined with Simulated Annealing (SA) performed similarly to the manual damage assessment. Although damage localization error increased slightly in two damage Scenarios (DS2 & DS4), the automated approach is still considered advantageous. Damage assessment was performed more quickly with the automated approach as each evaluation of the ANN took much less time compared to the FE simulation. Moreover, this automation enabled wider exploration in the search space. If the artificial damage was introduced in a different location, the automated approach could be able to identify the damage using the trained ANN. Whereas, hundreds of new simulations would be required for the manual approach.

The automated damage assessment approach was easily and quickly adapted to the investigated prestressed concrete bridge beam, demonstrating its suitability for real-world problems. Nevertheless, it is recommended to further study this approach for other bridge types, such as multi-span bridges with several beams. These types of bridges require the damage function to be defined not only in the longitudinal direction but also in the transverse direction.

Finally, the authors declare that there are no direct or indirect competing interests regarding the presented study.

Author contributions K.D. wrote the main manuscript text and worked on the automated damage assessment technique. S.S. developed the FE model, performed temperature compensation, and performed manual damage assessment. S. M. monitored the experimental and numerical experiments as well as temperature compensation, and revised the text.

Data availability The experimental and the numerical datasets generated during the current study are available from the corresponding author on reasonable request.

Declarations

Competing interests The authors declare no competing interests.

References

1. Pooya SMH, Massumi A. A novel and efficient method for damage detection in beam-like structures solely based on damaged structure data and using mode shape curvature estimation. Appl Math Model. 2021;91:670–94. https://doi.org/10.1016/j.apm.2020.09.012.
2. Sarmadi H, Yuen KV. Structural health monitoring by a novel probabilistic machine learning method based on extreme value theory and mixture quantile modeling. Mech Syst Signal Process. 2022;173: 109049. https://doi.org/10.1016/j.ymssp.2022.109049.

3. Hasija Y. A machine learning approach to bioinformatics. Sensors. 2019, 4035. https://doi.org/10.1016/b978-0-443-15250-4.00010-1.
4. Cross EJ, Koo KY, Brownjohn JMW, Worden K. Long-term monitoring and data analysis of the Tamar Bridge. Mech Syst Signal Process. 2013;35(1–2):16–34. https://doi.org/10.1016/j.ymssp.2012.08.026.
5. Xia Y, Lei X. Long-term performance monitoring and assessment of concrete beam bridges using neutral axis indicator. Struct Control Heal Monit. 2020. https://doi.org/10.1002/stc.2637.
6. Design L, Kromanis R, Kripakaran P, Harvey B. Long-term structural health monitoring of the Cleddau bridge: evaluation of quasi-static temperature effects on bearing movements. Struct Infrastruct Eng. 2015. https://doi.org/10.1080/15732479.2015.1117113.
7. Kulprapha N, Warnitchai P. Structural health monitoring of continuous prestressed concrete bridges using ambient thermal responses. Eng Struct. 2012;40:20–38. https://doi.org/10.1016/j.engstruct.2012.02.001.
8. Yarnold MT, Moon FL. Temperature-based structural health monitoring baseline for long-span bridges. Eng Struct. 2015;86:157–67. https://doi.org/10.1016/j.engstruct.2014.12.042.
9. Kromanis R, Kripakaran P. Performance of signal processing techniques for anomaly detection using a temperature-based measurement interpretation approach. J Civ Struct Heal Monit. 2021;11(1):15–34. https://doi.org/10.1007/s13349-020-00435-y.
10. Hu WH, Tang DH, Teng J, Said S, Rohrmann RG. Structural health monitoring of a prestressed concrete bridge based on statistical pattern recognition of continuous dynamic measurements over 14 years. Sensors. 2018;18(12):4117. https://doi.org/10.3390/S18124117.
11. Nguyen VH, Schommer S, Maas S, Zürbes A. Static load testing with temperature compensation for structural health monitoring of bridges. Eng Struct. 2016;127:700–18. https://doi.org/10.1016/j.engstruct.2016.09.018.
12. Huang H, et al. Strain-based performance warning method for bridge main girders under variable operating conditions. J Bridge Eng. 2020. https://doi.org/10.1061/(ASCE)BE.1943-5592.0001538.
13. Nguyen VH, Kebig T, Golinval JC, Maas S. Reduction of temperature effects for bridge health monitoring. Proc Int Conf Struct Dyn. 2020;1:1195–204. https://doi.org/10.47964/1120.9096.19343.
14. Agdas D, Rice J, Martinez J, Lasa I. Comparison of visual inspection and structural-health monitoring as bridge condition assessment methods. J Perform Constr Facil. 2016;30(3):1–10. https://doi.org/10.1061/(ASCE)CF.
15. Yang Y, Zhang Y. Review on vibration-based structural health monitoring. Symmetry. 2021;13:1998. https://doi.org/10.3390/sym13111998.
16. Casas JR, Moughty JJ. Bridge damage detection based on vibration data: Past and new developments. Front Built Environ. 2017. https://doi.org/10.3389/fbuil.2017.00004.
17. Singh T, Sehgal S. Damage identification using vibration monitoring techniques. Mater Today Proc. 2022;69:133–41. https://doi.org/10.1016/j.matpr.2022.08.204.
18. Marwala T. Finite-element-model updating using computional intelligence techniques. London: Springer; 2010. https://doi.org/10.1007/978-1-84996-323-7.
19. Schommer S, Dakhili K, Nguyen VH, Kebig T, Zürbes A, Maas S. A Gaussian damage function combined with sliced finite-element meshing for damage detection. J Civ Struct Heal Monit. 2022. https://doi.org/10.1007/s13349-022-00602-3.
20. Schommer S, Nguyen VH, Maas S, Zürbes A. Model updating for structural health monitoring using static and dynamic measurements. Procedia Eng. 2017;199:2146–53. https://doi.org/10.1016/j.proeng.2017.09.156.
21. Alizadeh R, Allen JK, Mistree F. Managing computational complexity using surrogate models: a critical review. Res Eng Des. 2020;31(3):275–98. https://doi.org/10.1007/s00163-020-00336-7.
22. Kudela J, Matousek R. Recent advances and applications of surrogate models for finite element method computations: a review. Soft Comput. 2022;26(24):13709–33. https://doi.org/10.1007/s00500-022-07362-8.
23. Bhosekar A, Ierapetritou M. Advances in surrogate based modeling, feasibility analysis, and optimization: a review. Comput Chem Eng. 2018;108:250–67. https://doi.org/10.1016/j.compchemeng.2017.09.017.
24. Alkayem NF, Cao M, Zhang Y, Bayat M, Su Z. Structural damage detection using finite element model updating with evolutionary algorithms: a survey. Neural Comput Appl. 2018;30(2):389–411. https://doi.org/10.1007/s00521-017-3284-1.
25. Jahjouh M. An experience based artificial neural network in the design optimization of steel frames. Eng Res Express. 2022. https://doi.org/10.1088/2631-8695/aca6ce.
26. Jepsen L, Trawick BB Eds. Handbook of Meta-Heuristics, vol. 57. Kluwer Academic Publishers; 2006. https://doi.org/10.2307/3198208.
27. Schommer S. Damage detection in prestressed concrete bridges based on static load testing, sagging, and modal parameters, using measurements and model updating. University of Luxembourg, 2017. https://orbilu.uni.lu/handle/10993/32971
28. Schommer S, Kebig T, Nguyen VH, Zürbes A, Maas S. Modeling of a prestressed concrete bridge with 3D finite elements for structural health monitoring using model updating techniques. Proc. ISMA 2018 - Int. Conf. Noise Vib. Eng. USD 2018 - Int Conf Uncertain Struct Dyn. pp. 1607–1620, 2018.
29. Sammut C, Webb GI, editors. Encyclopedia of Machine Learning. Boston: Springer; 2011. https://doi.org/10.1007/978-0-387-30164-8.
30. Joshi AV. Machine Learning and Artificial Intelligence. 1st ed. Cham: Springer; 2020. https://doi.org/10.1007/978-3-030-26622-6.
31. Suzuki K. Artificial neural networks - methodological advances and biomedical applications. 2011.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.