Boundary Strategy for Optimization-based Structural Damage Detection Problem using Metaheuristic Algorithms

Ali Kaveh[1*], Seyed Milad Hosseini[1], Ataollah Zaerreza[1]

[1] School of Civil Engineering, Iran University of Science and Technology,16846-13114 Tehran, Iran
[*] Corresponding author, e-mail: alikaveh@iust.ac.ir

Abstract

The present paper proposes a new strategy namely Boundary Strategy (BS) in the process of optimization-based damage detection using metaheuristic algorithms. This strategy gradually neutralizes the effects of structural elements that are healthy in the optimization process. BS causes the optimization method to find the optimum solution better than conventional methods that do not use the proposed BS. This technique improves both aspects of the accuracy and convergence speed of the algorithms in identifying and quantifying the damage. To evaluate the performance of the developed strategy, a new damage-sensitive cost function, which is defined based on vibration data of the structure, is optimized utilizing the Shuffled Shepherd Optimization Algorithm (SSOA). Different examples including truss, beam, and frame are investigated numerically in order to indicate the applicability of the proposed technique. The proposed approach is also applied to other well-known optimization algorithms including TLBO, GWO, and MFO. The obtained results illustrate that the proposed method improves the performance of the utilized algorithms in identifying and quantifying of the damaged elements, even for noise-contaminated data.

Keywords

structural damage detection, Boundary Strategy, metaheuristic optimization algorithms, Shuffled Shepherd Optimization Algorithm, vibration data

1 Introduction

Damage occurrence is an inevitable event that can happen in each engineering structure and causes the performance of the structure to decrease over time. On the other hand, structural damage is prone to spread as a result of changing mechanical properties of the structures which are occurred due to crack, creep, corrosion, and so on. Considerable attention has been devoted to methods for identifying and quantifying the site and extent of damage in the elements of engineering structures. Among these methods, vibration-based damage detection methods due to obtaining easily and being independent of the external excitation can be used as an efficient measure of structural behavior before and after damage occurrence [1].

In a general view, vibration-based damage detection methods can be broadly categorized into two groups. The first group deals with non-model (data-driven) methods. Although these techniques can identify the location of damage efficiently without using the structural analytical programs, they are not capable of finding the extent

of damage with a high level of accuracy [2]. The second group is model updating methods in which the damage identification problem is defined as an inverse problem. Various techniques have been introduced for finite element model updating. They are generally categorized into two groups. The first group deals with direct methods as accurate ones. These methods are non-iterative and do not need parameter updating. The eigenstructure assignment is one of the instances of the direct model updating method [3]. The second group deals with indirect methods. These methods are generally categorized in four classes including Bayesian/Monte Carlo approaches, response surface method, sensitivity-based methods, and model updating using computational intelligence techniques (machine learning and evolutionary algorithms) [4].

The Bayesian method is a technique based on Bayes′ theorem. According to this method, a set of data with a probability distribution which leads to the probability distribution of the model is considered. Monte Carlo

approaches are frequently used to solve the Bayesian-based finite element model updating problems [5].

The response surface-based finite element model updating is a statistical technique that can obtain the best response which fits the least change between the initial finite element model and the measured response [6].

The sensitivity-based finite element model updating methods are frequently presented as optimization problems. These methods deal with linearization of the generally non-linear relationship between measurable outputs such as natural frequencies or mode shapes and model parameters that are required to be corrected. The main characteristic of these methods is that the computing derivatives of modal data or frequency responses are expensive computationally [7]. Sensitivity-based damage detection using regularization techniques has received considerable attention in recent past decades. In this technique, a regularization term is applied to the objective function to provide a physically stable solution, which is a unique one [8]. However, selecting an appropriate regularization parameter is problem-dependent and selected by the experience. An alternative technique is devoted to sparsity constrained optimization algorithms to solve the damage detection problem as an optimization one [9].

In this paper, we have focused on the finite element model updating methods using evolutionary algorithms. Evolutionary algorithm-based finite element model updating methods in comparison to other indirect ones have been utilized more extensively in the past decades [4]. On the other hand, they can determine the location and severity of damage effectively. In these methods, the cost function of the optimization problem, which is defined as a function error representing the discrepancies between vibration data identified by modal testing and those calculated by the analytical model, is minimized. Minimizing this error will provide a nonlinear optimization problem. Since damage detection by model updating is an ill-posed problem, a robust damage-sensitive cost function, as well as a powerful metaheuristic algorithm, are required to identify structural damage sites and extents [10]. To minimize the cost function of the optimization problem, metaheuristic algorithms have been employed in recent years. These optimization methods are efficient and powerful tools. Furthermore, they can not only determine the extent and location of the damage with acceptable accuracy [11] but also can apply to solve other optimization problems in the field of structural optimization [12].

Various studies have been devoted to structural damage detection utilizing finite element model updating by evolutionary algorithms. For instance, Khatir et al. [13] proposed an inverse problem method to detect and estimate damages in beam-like structures using Particle Swarm Optimization (PSO) and Bat Algorithm (BA). Kourehli [14] proposed three damage detection methods based on modal data and static response of the damaged structure utilizing Simulated Annealing (SA). Bureerat and Pholdee [15] formulated the damage detection problem as an optimization inverse problem and solved it with an improved version of the Sine Cosine Algorithm (SCA). Dinh-Cong et al. [16] proposed a hybrid objective function based on the Multiple Damage Location Assurance Criterion (MDLAC) and modal flexibility change for determining the location and extent of damage using Jaya algorithm. In the same year, Dinh-Cong et al. [17] proposed an efficient technique for optimal sensor placement and damage detection in laminated composite structures, and Jaya algorithm was again adopted to solve the inverse problem. Ghasemi et al. [18] proposed a two-stage damage detection method as an inverse method for truss structures via a modified genetic algorithm. Ghiasi et al. [19] carried out a comprehensive study on damage detection based on machine learning and inverse optimization problem utilizing Colliding Bodies Optimization (CBO). Kaveh and Dadras [20] enhanced Thermal Exchange Optimization (TEO) and applied for structural damage detection. Tiachacht et al. [21] presented a new methodology based on model updating method for damage identification and quantification of two- and three-dimensional structures using the Genetic Algorithm (GA). Jahangiri et al. [22] presented an efficient damage detection technique based on a hybrid objective function and using the only first vibration mode data utilizing meta-heuristic optimization algorithms. Mishra et al. [23] used the Ant Lion optimizer (ALO) for identifying structural damage based on the objective function, which is formulated as an inverse problem using vibration data. Hosseini et al. [24] conducted a comprehensive study on the efficiency of the objective functions developed based on Modal Strain Energy (MSE) and flexibility methods for solving structural damage detection problems as inverse optimization ones.

When the metaheuristic optimization algorithms are employed for damage detection of large-scale structures, the algorithms start to search in high-dimensional search space. This is because the number of design variables is

equal to the number of structural elements. As a result, the optimization algorithm may unable to predict the location and extent of damage accurately. For instance, the damage detection results for a 72-bar spatial investigated by Mishra et al. [25], Kaveh and Zolghadr [26], Bureerat and Pholdee [27] indicate that their techniques found different success rates (i.e. the number of successful runs / total number of runs), and only some of them have a 100 % success rate. On the other hand, there is a high probability for the metaheuristic algorithms to trap in local optima when they are employed to solve the damage detection problem. To this end, here, a new strategy namely Boundary Strategy (BS) is proposed for structural damage detection problem using metaheuristic optimization algorithms. In BS, the healthy elements are gradually neutralized from the optimization process. This strategy causes the complexity of the search space is progressively decreased and improves the performance of the employed optimization method in identifying damaged elements. In contrast, in conventional damage detection methods utilizing metaheuristic algorithms, the effects of healthy elements are not neutralized in the optimization process. In this article, these conventional techniques are considered Without Boundary Strategy (WBS).

Although many structural elements are determined as damaged elements in the optimization process due to the effects of measurement noise, most of these elements are healthy ones [28]. As a result, a penalty function can be considered in order to weight against an increasing number of damaged elements. Accordingly, many researchers have utilized this approach for their proposed objective functions [29, 30]. Similarly, in this study, a penalty-function by integrating into a damage-sensitive cost function based on vibration data is established. The Shuffled Shepherd Optimization Algorithm (SSOA) presented by Kaveh and Zaerreza [31] is employed to solve the damage detection problem. The reason for selecting SSOA is due to its low-level accuracy in detecting damaged elements when it is applied to solve the problem using WBS. However, when the SSOA utilizes the BS in the optimization process, it predicts the location and extent of damage with a high level of accuracy. In the body of the SSOA, the position of community member obtained for a given problem in each community not only moves toward the better members but also tends to move the worse members. This concept has caused SSOA to have appropriate performance in different optimization problems [32]. To evaluate the capability

of the proposed cost function using BS, four test examples including a 25-bar planar truss, a 40-element continuous beam, a 23-element asymmetrical planar frame, and a large-scale 72-bar spatial truss are examined. The obtained results are compared with the three well-known parameter-less optimizers: Teaching-learning based optimization (TLBO) [33], Grey Wolf Optimizer (GWO) [34], and Moth-flame Optimization Algorithm (MFO) [35]. Furthermore, the robustness of the BS in comparison to WBS is studied in different optimization methods and cost functions.

The rest of this paper is organized as follows: in Section 2, a general review of the SSOA is presented. Section 3 provides the formulation of the problem under consideration and introduces BS. Numerical examples are presented in Section 4 and the obtained results are discussed. Concluding remarks are finally driven in Section 5.

2 Shuffled Shepherd Optimization Algorithm (SSOA)

Shuffled Shepherd Optimization algorithms (SSOA) is a new multi-community population-based metaheuristic introduced by Kaveh and Zaerreza [31]. This optimization method mimics the behavior of shepherds in nature. In this method, firstly, members of each community are randomly generated. Next, the shuffling process is performed to enhance survivability by sharing information in the search process. This process leads to the possible improvement of the community based on the exchange of its information with other communities. In the SSOA, to calculate the new position of each member in each community, the better and worse members are randomly selected from the community of which the member under consideration belongs. Then, if the objective function value of the newly calculated position is better than the previously generated one, the newly obtained position will be replaced by the previous one. Finally, the optimization process will be terminated if the termination condition is fulfilled.

2.1 Steps of SSOA

The steps of SSOA are characterized by five main steps [36]. These steps are as follows:
1. Forming the initial community members
2. Shuffling process
3. Movement of community member
4. Updating the position of each community member
5. Termination condition of SSOA

In the following, the above-mentioned steps are described, and their mathematical interpretations are given.

2.1.1 Forming the initial community members
In the SSOA, the initial positions of members of communities (MOC) are determined with a randomly generated population in a d-dimensional search space:

$$MOC_{i,j}^0 = x_{min} + \boldsymbol{r} \times \left(x_{max} - x_{min} \right);$$
$$i = 1, 2, \ldots, c \text{ and } j = 1, 2, \ldots, nm, \tag{1}$$

where $\boldsymbol{r}$ is a random vector in which each component is generated between 0 and 1; x_{min} and x_{max} denote the minimum and maximum permissible values, respectively; c and nm refer to the number of communities and number of members that belong to each community, respectively. Since each community has nm members and total numbers of communities are equal to c, the population size is calculated as:

$$nPop = c \times nm . \tag{2}$$

2.1.2 Shuffling process
The shuffling process denotes to merging communities into one community and building new communities. Accordingly, initially, the entire population ($nPop$) is sorted based on the quality of solutions. Next, in order to form the MC matrix (see Eq. (3)), $nPop$ are grouped into c communities in which each community has the nm members. To this end, in the first step, the first c members are selected from the sorted population and are randomly assigned to c communities so that each community has one member so far. Thus, the first column of the MC matrix was formed. In the next step, the next c members are selected from the remaining population and are again assigned to the c communities randomly. Therefore, the second column of the matrix was obtained as well. This process is repeated nm times until all c communities form with nm members. Consequently, the MC matrix is achieved as follows:

$$MC = \begin{bmatrix} MOC_{1,1} & MOC_{1,2} & \cdots & MOC_{1,j} & \cdots & MOC_{1,nm} \\ MOC_{2,1} & MOC_{2,2} & \cdots & MOC_{2,j} & \cdots & MOC_{2,nm} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ MOC_{i,1} & MOC_{i,2} & \cdots & MOC_{i,j} & \cdots & MOC_{i,nm} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ MOC_{c,1} & MOC_{c,2} & \cdots & MOC_{c,j} & \cdots & MOC_{c,nm} \end{bmatrix}, \tag{3}$$

in which $MOC_{i,j}$ refers to the jth member of the ith community. According to the MC matrix (Eq. (3)), each row illustrates the members belonging to each community. Furthermore, the first column denotes the members of each community that have the best quality, and members placed in the last column are the worst members of each community.

2.1.3 Movement of community member
After forming MC matrix, the stepsize of each $MOC_{i,j}$ is calculated based on two vectors. For this purpose, for each $MOC_{i,j}$, the members that have the better and worse objective function values than $MOC_{i,j}$ are selected randomly. These members called $MOC_{i,b}$ and $MOC_{i,w}$, respectively. In SSOA, $MOC_{i,j}$ not only moves toward the $MOC_{i,b}$ but also tends toward the $MOC_{i,w}$. Moving $MOC_{i,j}$ toward the $MOC_{i,b}$ indicates its intensification tendency. In contrary, tending $MOC_{i,j}$ toward the $MOC_{i,w}$ shows its diversification tendency. This concept is schematically shown in Fig. 1 and is mathematically stated as follows:

$$stepsize_{i,j} = stepsize_{i,j}^{Worse} + stepsize_{i,j}^{Better}$$
$$i = 1, 2, \ldots, c \text{ and } j = 1, 2, \ldots, nm, \tag{4}$$

where $stepsize_{i,j}^{Worse}$ and $stepsize_{i,j}^{Better}$ are defined as follows:

$$stepsize_{i,j}^{Worse} = \alpha \times \boldsymbol{r}_1 \times \left(MOC_{i,w} - MOC_{i,j} \right), \tag{5}$$

$$stepsize_{i,j}^{Better} = \beta \times \boldsymbol{r}_2 \times \left(MOC_{i,b} - MOC_{i,j} \right), \tag{6}$$

in which $stepsize_{i,j}^{Worse}$ and $stepsize_{i,j}^{Better}$ are the stepsize vectors with d design variables. These vectors respectively illustrate diversification and intensification tendencies of the algorithm; $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ are random vectors in which each component is generated between 0 and 1. Since $MOC_{i,nm}$ is placed in the last column of the MC matrix (Eq. (3)), it does not have a member worse than itself. Thus, $stepsize_{i,j}^{Worse}$ (Eq. (5)) is equal to zero. Similarly, $stepsize_{i,j}^{Better}$ will be equal to zero because $MOC_{i,1}$ does not have member better
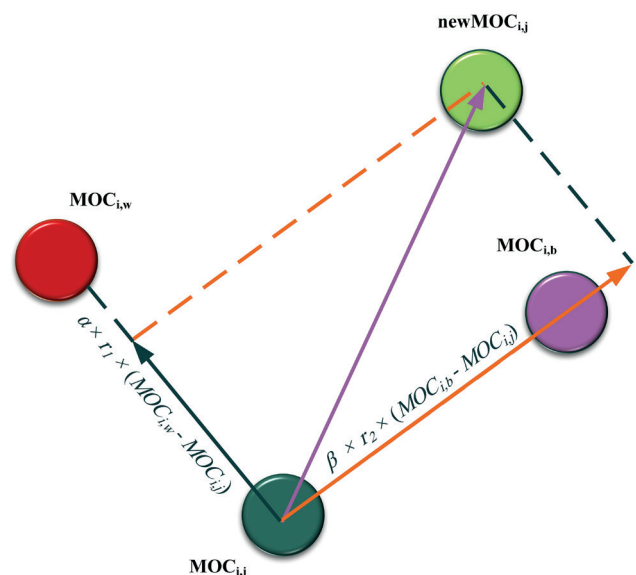

Fig. 1 A schematic of position updating in SSOA

than itself. α and β are the parameters that control exploration and exploitation, respectively. These parameters are defined as follows:

$$\alpha = \alpha_0 - \alpha_0 \times t \, , \tag{7}$$

$$\beta = \beta_0 + \left(\beta_{max} - \beta_0\right) \times t; \quad t = \frac{it}{Maxit} \, , \tag{8}$$

in which it and $Maxit$ denote the iteration and the maximum number of iterations. According to Eqs. (7) and (8), α and β are among the most important parameters of the SSOA in that they control the balance between exploration and exploitation tendencies. In this regard, here, decreasing α and increasing β respectively lead to explore the search space more efficiently in the early iterations and search around the better solutions in the last iterations.

2.1.4 Updating the position of each community member
In this step, first, the new position of each community member is determined as:

$$newMOC_{i,j} = MOC_{i,j} + stepsize_{i,j} \, , \tag{9}$$

where $newMOC_{i,j}$ is the new position of jth member of the ith community. Then, its corresponding objective function is evaluated. In order to decide which positions ($newMOC_{i,j}$ or $MOC_{i,j}$) return to the population, the replacement strategy is applied. Accordingly, the objective function value of $newMOC_{i,j}$ and $MOC_{i,j}$ are compared, and the better one is returned to the population.

2.1.5 Termination condition of SSOA
In SSOA, the maximum number of iterations ($Maxit$) is considered as a stopping criterion. Hence, if the current iteration is smaller than $Maxit$, SSOA returns to *Step 2* for a new round of iteration. Otherwise, the algorithm terminates, and the best community member is reported. The pseudo-code of SSOA is provided in Algorithm 1.

3 Structural damage detection approach
In this section, first, the theoretical background of the inverse damage detection problem including damage modeling and obtaining vibration data are presented. The second section provides the proposed damage-sensitive cost function. An efficient strategy to solve the problem is finally introduced in the last section.

3.1 Theoretical background
In the vibration-based damage detection method, the main modal parameters including natural frequency and mode

---

Algorithm 1 Framework of SSOA

The procedure of Shuffled Shepherd Optimization Algorithm (SSOA)
begin
Set the algorithm parameters; α_0, α_{max}, and β_0
Initialize number of members belong to each community (nm) and number of communities (c), and termination criterion ($Maxit$)
Generate the initial candidate solutions and evaluate them.
while (termination criterion not satisfied) do
   for j:1 to nm
      select the c members from the remaining population based on the quality of solutions.
      put c selected members randomly in the jth column of the MC matrix.
   end for
   for i:1 to c
      for j:1 to nm
         Select $MOC_{i,b}$ and $MOC_{i,w}$ randomly for $MOC_{i,j}$
         Calculate movement of $MOC_{i,j}$ using Eq. (4)
         Updating the position of each community member based on step 4.
      end for
   end for
end while
end

---

shapes vector of a vibrating structure can be calculated by the following equation:

$$\left(K - \omega_i^2\right)\varphi_i = 0; \quad i = 1, 2, \ldots, ndof \, , \tag{10}$$

where ω_i and φ_i are natural frequency and mode shape vector in ith mode, respectively; M and K are respectively the mass and stiffness matrix with the size of $ndof \times ndof$.

An approach that extensively use to model damage, considers damage as the reduction of stiffness characteristics such as modulus of elasticity (E), cross-sectional area (A), and moment of inertia (I) [30]. In this method, it is considered that mass changes before and after damage are negligible. Similarly, here, damage is modeled as a relative reduction of E in each structural element such that:

$$RF_e = \left(1 - x_e\right); \quad 0 \le x_e \le 1 \, , \tag{11}$$

$$E_e^d = RF_e \times E_e^h; \quad e = 1, 2, \ldots, nte \, , \tag{12}$$

in which x_e is the damage ratio of the eth element; nte is the total number of structural elements; E_e^h and E_e^d denote modulus of elasticity of the eth healthy and damaged elements, respectively; $x_e = 0$ indicates that the element is healthy, while $x_e = 1$ shows that the element is fully damaged. By considering this, the total stiffness matrix of structures is equal to the summation of the stiffness matrix of damaged and healthy elements:

$$K = \sum_{e=1}^{nte} RF_e \times k_e \tag{13}$$

where k_e is the stiffness matrix of the eth element.

3.2 Proposed objective function

In this section, the proposed cost function is presented in detail. Damage occurrence leads to changes in natural frequencies and corresponding mode shapes of the structure before and after the damage. Although natural frequencies changes as a result of damage occurrence can be easily obtained, its alterations are low-sensitive to damage [37, 38]. Therefore, minor structural damage cannot be solely identified by the natural frequency changes. Mode shapes in comparison to natural frequencies include local information which leads to their more sensitivity to local damages. Thus, considering mode shapes make them be used directly in multiple damage detection. Furthermore, mode shapes are less sensitive to environmental effects (e.g. temperature) than natural frequencies. However, measuring mode shapes requires the use of many sensors, which are measured with lower accuracy than natural frequencies. As a result, they are more susceptible to noise contamination than natural frequencies [28, 39]. In order to tackle the drawbacks of each main modal parameters (natural frequencies and mode shapes), in this paper, the combination of them is considered. Accordingly, a damage-sensitive cost of the optimization problems will be made up of two functions. The first function is a penalty function that weights against an increasing number of damaged elements. The second function is considered the combination of natural frequency and mode shapes. Since the effect of measurement noise leads optimization algorithms to predict many structural elements as damaged ones, a penalty function is considered against the increasing of damaged elements. The damage-sensitive cost function used for damage identification in this paper is established as follows:

$$Find \ \boldsymbol{X} = \left[x_1, x_2, \ldots, x_{nte} \right]^T ; \ 0 \le x_e \le 1, \tag{14}$$

where vector $\boldsymbol{X}$ specifies the ratio of structural damage.

$$Minimize \ F(X); \ F(X) = \left(1 + \gamma \cdot P(X)\right) \times G(X), \tag{15}$$

where $F(X)$ is the proposed cost function; γ is the penalty factor (equal to 0.5 in this study), and $P(X)$ and $G(X)$ are respectively penalty function and the cost function without penalty:

$$P(X) = \frac{m^d(X)}{nte}, \tag{16}$$

$$G(X) = \sum_{i=1}^{nmod} \left(R_i \times \left(1 - MAC(i,i)\right) \right), \tag{17}$$

where $m^d(X)$ is the number of damaged elements found by the metaheuristic algorithm in the solution X; $nmod$ number of used modes and R_i and $MAC(i,j)$ are calculated as follows:

$$R_i = \frac{\left(\omega_i^d\right)^2}{\left(\omega_i^a\right)^2}, \ MAC(i,i) = \frac{\left(\varphi_i^{dT} \times \varphi_i^a\right)^2}{\left(\varphi_i^{dT} \times \varphi_i^d\right)\left(\varphi_i^{aT} \times \varphi_i^a\right)}, \tag{18}$$

in which ω_i and φ_i represent the ith natural frequency and its corresponding mode shape, respectively. The superscript d and a stand, respectively, for the damaged model and the analytical one, and MAC represents the modal assurance criteria (MAC).

3.3 The Boundary Strategy (BS) in metaheuristic based damage detection

Structural damage identification problem using finite element model updating is a highly complex problem with lots of local optimum. Despite a lot of effort has been devoted to the damage detection methods utilizing metaheuristic algorithms, it is observed some shortcomings in damage detection results. Some of them are mentioned below: (1) when too many design variables are involved in the problem, some metaheuristic algorithms cannot find the location and extent of damage properly or may not predict it with a high level of accuracy. For instance, Mishra et al. [25] employed 10 metaheuristic algorithms for solving the problem. The obtained results revealed that most of the investigated algorithms are not capable of identifying damaged elements properly; (2) there is a high probability of being trapped in local optima when some metaheuristic algorithms are applied. As a result, the algorithms fail to find the global optimum solution as the damage detection results.

In order to alleviate these handicaps, here, a simple strategy is proposed for damage detection problems using metaheuristic algorithms. This strategy is called Boundary Strategy (BS). In this strategy, the effects of structural elements that are related to the healthy ones are gradually neutralized in the optimization process.

In BS, first of all, the lower and upper bounds of design variables are respectively set to be -1 and 1 instead of 0 and 1 in WBS. Then, the metaheuristic algorithm is executed.

Before evaluating the objective function of each agent in each step, the vibration characteristics of the analytical model (like modal data) should be calculated first according to Section 3.1. Hence, in order to calculate RF_e in Eq. (11), the value of each solution component should be changed to zero if it is negative. This is because the extent of damage in each structural element according to this equation is placed in [0,1] interval. In other words, when the vibration characteristic of the model structure is calculated, each solution component smaller than zero is changed to zero. It is worth mentioning that this change from [-1,1] to [0,1] is only performed to calculate the RF_e

in Eq. (11) and is not returned to the optimization process. This means that the values of the design variables remain unchanged so that each of them is placed in [-1,1] interval during the course of optimization.

The BS causes the design variables related to healthy elements to be in [-1,0] interval. When any design variable among all solutions of the population placed in this interval, it traps in this range. As a result, the effect of the respective design variable is neutralized from the course of the optimization process. SSOA as a population-based metaheuristic algorithm is considered to evaluate the capability of BS in comparison to WBS. The reason for choosing the SSOA
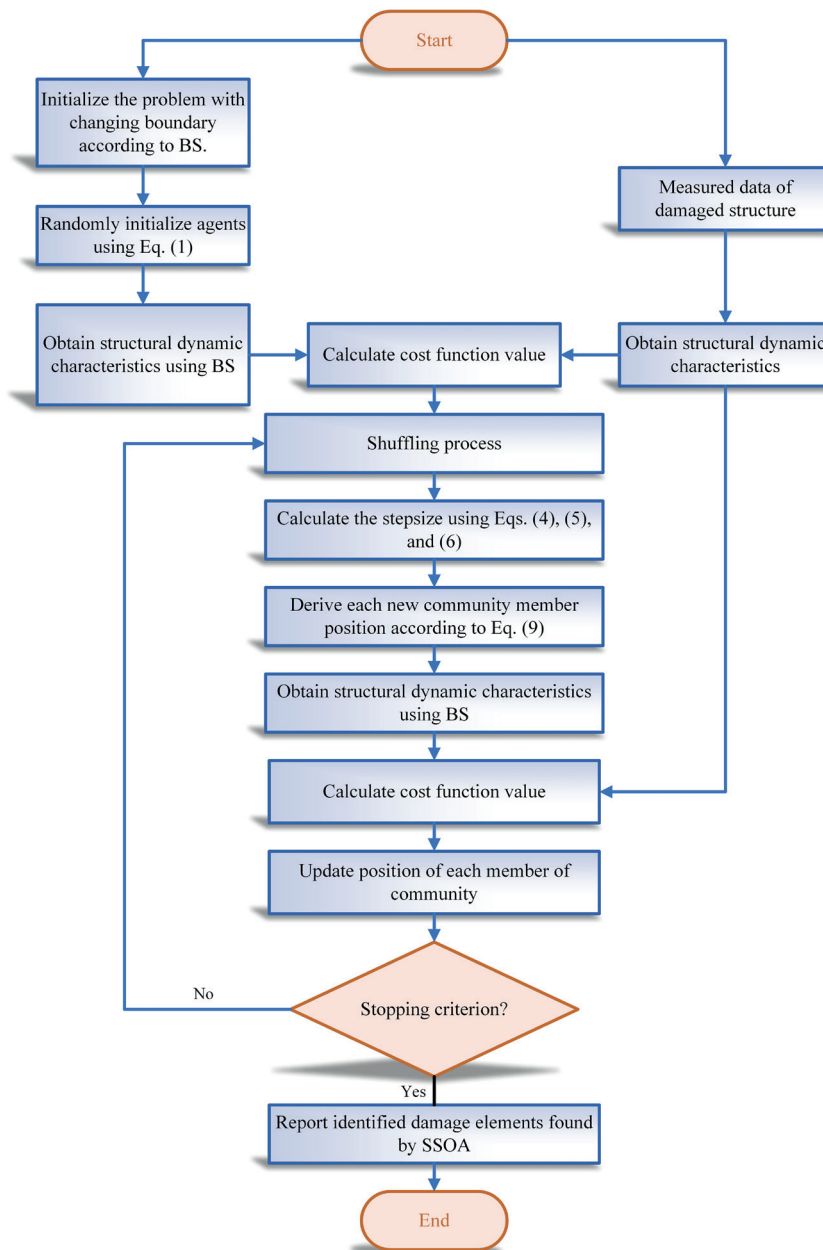

Fig. 2 Flowchart of the SSOA using BS for damage detection

is due to its many false alarms in damage detection results when WBS is applied. The flowchart of SSOA for damage detection using BS is provided in Fig. 2.

4 Numerical examples

In this section, four numerical examples are studied to show the capability and efficiency of the proposed method. These examples are as follows: a 25-bar planar truss, a 40-element continuous beam, a 23-element asymmetrical planar frame, and a large-scale 72-bar spatial truss.

All numerical case studies are investigated in two states. The first is the ideal condition in which input data are not contaminated by measurement noise. The second deals with the noisy condition in which each component of eigenvalue and eigenvector are contaminated with measurement noise as:

$$input^{noise} = input \times (1 + rand \times \sigma), \tag{19}$$

where $input^{noise}$ and $input$ are natural frequencies value or mode shape vector in the noisy and ideal states, respectively. $rand$ is a random number between -1 and 1, and σ is the intensity of the applied noise. In this paper, natural frequencies and corresponding mode shapes vectors are contaminated with 1 % and 3 % noise, respectively.

In order to compare the ratio of the identified and actual damage, an error index is defined as follows:

$$Error = \frac{100(\%)}{nte} \times \sum_{e=1}^{nte} |AD_e - ID_e|, \tag{20}$$

in which AD_e is the actual damage ratio and ID_e is the identified damage ratio.

SSOA is employed for solving the damage detection problem as an optimization one. The algorithm parameters in all test examples are assumed to be as follows: $m = 4$, $n = 5$, $Maxit = 1000$, $\alpha_0 = 0.5$, $\beta_0 = 2$, and $\beta_{max} = 2.5$. Three other well-known parameter-less metaheuristics including Teaching-learning-based optimization (TLBO), Grey Wolf Optimizer (GWO), and Moth-Flame Optimization (MFO)

are run with the same $maxNSAs$, and their obtained results are compared with those found by the SSOA. In all cases, the number required structural analyses (NSAs) is calculated. For this purpose, an iteration in which differences between its corresponding cost function value and cost function value of the $Maxit$ is smaller than 10^{-6} is found. Next, the obtained iteration is multiplied to the population size, which gives $NSAs$.

All investigated structures are modeled numerically in the MATLAB environment and are analyzed using the direct stiffness method. 10 independent runs are executed in all test examples to get statistically meaningful results. The average values of the obtained results are reported in the figures. The healthy elements which have negative values in the vector of the best solution using BS are considered equal to zero. Thus, zero values in all bar figures indicate that the respective element is healthy. It should be noted that in the first three test examples, the first five vibration modes are utilized for identifying damage, whereas in the last large-scale example this value is considered 12.

4.1 25-bar planar truss

The first example is considered a 25-bar planar truss as depicted in Fig. 3. This example has 12 nodes and 21 degrees of freedom (DOFs). For each element, the modulus of elasticity, material density, and cross-sectional area are respectively as follows: $E = 200$ GPa and $\rho = 7780$ kg/m^3, and $A = 10$ cm^2. Table 1 provides two different damage scenarios.

In this example, the capability of using BS in comparison to using WBS is investigated. For this purpose, the average values of damage detection results found by SSOA using BS and WBS in different scenarios are depicted in Figs. 4 and 5, respectively. A close examination of these figures reveals that the results obtained by using BS are much better than those found by using WBS in all cases. In other words, using BS localized and quantified damaged elements precisely even when the input data are contaminated by measurement noise. In contrast, applying WBS shows that all elements have damage even for the ideal
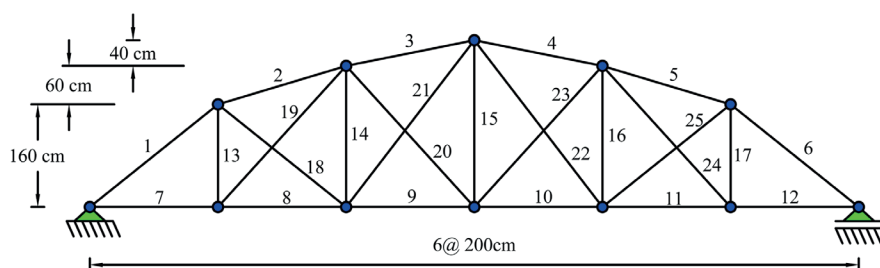

Fig. 3 Finite element model of the 25-bar planar truss

Table 1 Two different damage scenarios in the 25-bar planar truss

Scenario	I		II			
Element no.	2	21	3	7	15	20
Damage ratio (%)	25	10	20	25	20	25

state. The statistical results gained via SSOA using BS and WBS for two different damage scenarios in the case with noise and without noise are given in Table 2. From this table, in both damage scenarios, the statistical results using BS are significantly better than those obtained by using WBS. For further investigation, it can be seen that using BS decreases NSAs more than 50 % in comparison to using WBS in both damage scenarios. Moreover, the error index (calculated according to Eq. (20)) obtained from utilizing BS in comparison to WBS is considerably is low and close to zero value in all cases. It can be concluded that unlike employing WBS for damage detection, applying BS has high acceptable accuracy.
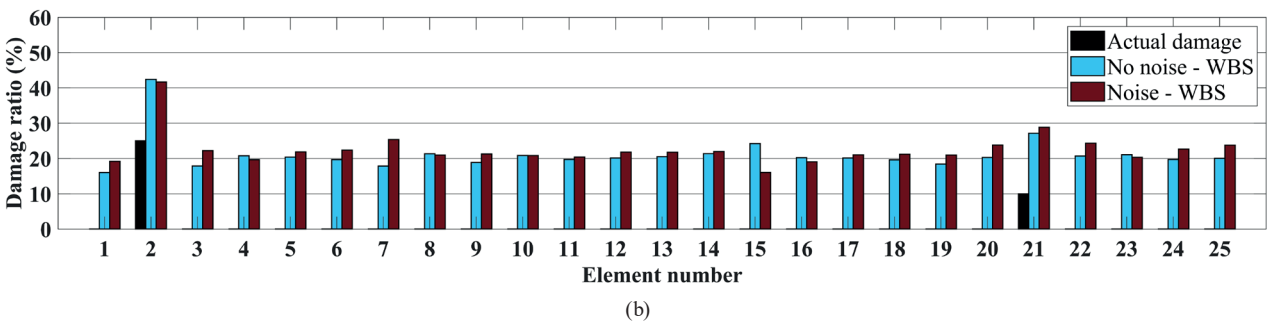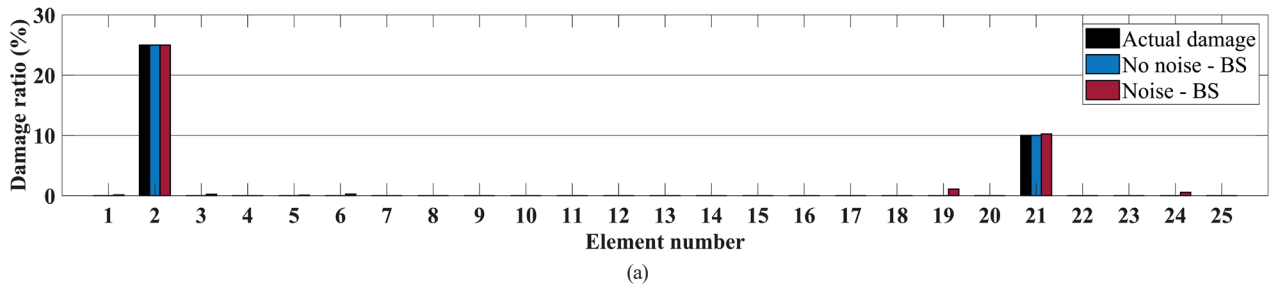

Fig. 4 Average value of damage detection results for the scenario I of the 25-bar planar truss (a) using BS, and (b) WBS
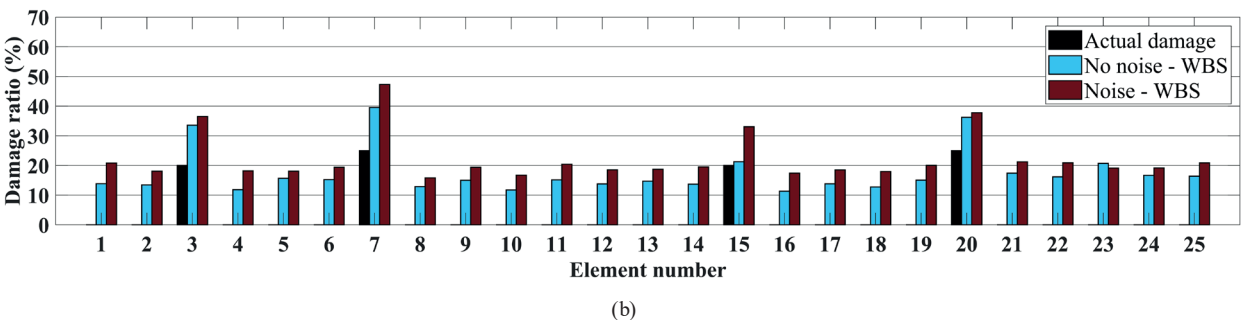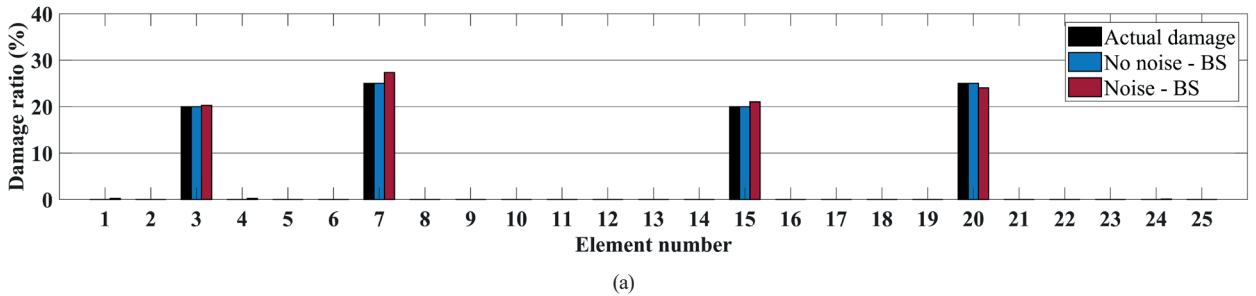

Fig. 5 Average value of damage detection results for scenario II of the 25-bar planar truss (a) using BS, and (b) WBS

Table 2 Statistical damage identification results in the 25-bar planar truss for both damage scenarios in the case with noise and without noise

| Scenario	Noise level	Actual location	Actual ratio	BS		WBS	
				Avg. value	Std. value	Avg. value	Std. value
I	Noise-free	2	25	25	9.72E-7	42.42	6.0536
		21	10	10	2.89E-6	27.16	5.6981
		Error (%)		1.34E-7	9.09E-8	19.7787	7.1536
		NSAs		7126		19800	
	Noisy	2	25	25	0.4195	41.68	5.9545
		21	10	10.24	1.7127	28.84	5.6336
		Error (%)		0.1532	0.1011	21.1405	5.9146
		NSAs		9986		19822	
II	Noise-free	3	20	20	4.72E-7	33.57	3.5304
		7	25	25	1.50E-6	39.51	13.2814
		15	20	20	5.17E-6	21.27	14.8681
		20	25	25	2.00E-6	36.21	3.1153
		Error (%)		3.33E-7	6.04E-8	14.4617	3.9270
		NSAs		4808		19802	
	Noisy	3	20	20.29	0.1928	36.51	8.0478
		7	25	27.34	0.0516	47.31	11.5700
		15	20	21.04	0.2180	33.06	17.3246
		20	25	24.07	0.0520	37.77	4.2017
		Error (%)		0.2081	0.0456	18.7160	4.9669
		NSAs		7120		19896	
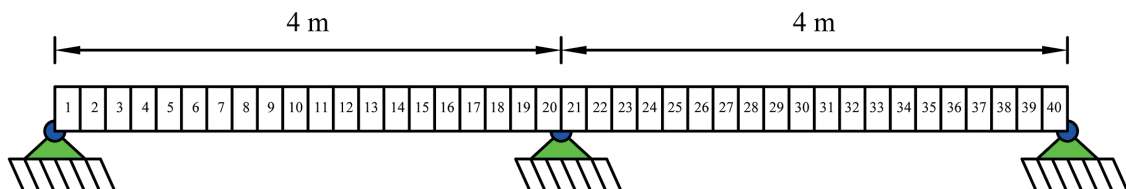
Fig. 6 Finite element model of the 40-element continuous beam
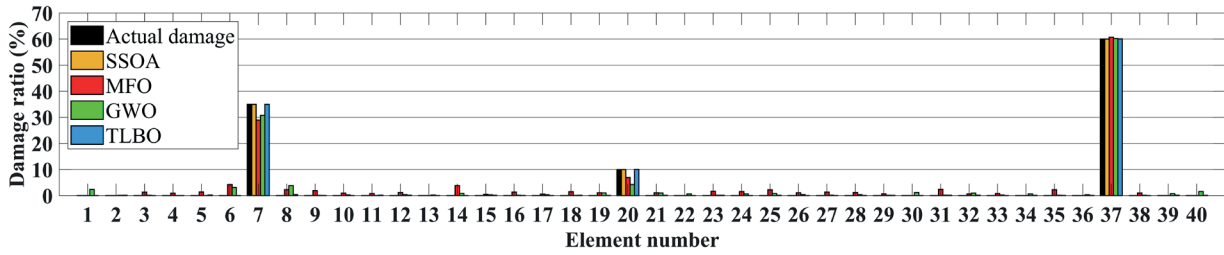
4.2 40-element continuous beam

A 40-element continuous beam is considered as the second test example to verify the capability of the proposed method. The finite element model of the beam is given in Fig. 6. Each node of this beam has 2 DOFs and only the vertical components of the supports have been limited. Therefore, the total DOFs becomes 79. In this case study, both the width and height of each element are equal to 15 cm. The modulus of elasticity and material density for all elements are $E = 210$ Gpa and $\rho = 7860$ kg/m^3, respectively. Two different damage scenarios as presented in Table 3 are considered.

In this example to evaluate the performance of SSOA, three other well-known optimization algorithms including TLBO, GWO, and MFO are considered for comparison. For this purpose, the average results obtained by these algorithms using BS are compared with the results found

Table 3 Two different damage scenarios in the 40-element continuous beam

Scenario	I				II			
Element no.	7	20	37	2	6	8	26	32
Damage ratio (%)	35	10	60	45	55	20	55	60

by SSOA. These comparisons for the damage Scenarios I and II are presented in Figs. 7 and 8, respectively. A close examination of these figures shows that by applying BS, SSOA could gain much better results than other optimization methods whether the input data are contaminated with noise or not. For further inspection to show the efficiency of the proposed method in the case when TLBO, GWO, and MFO are employed for detecting damaged elements, Fig. 9 provides the results using WBS for the second damage scenario in the noisy state. As can be shown in this
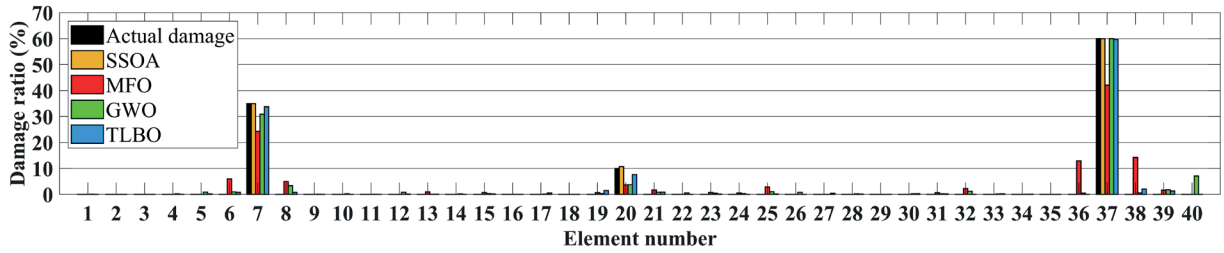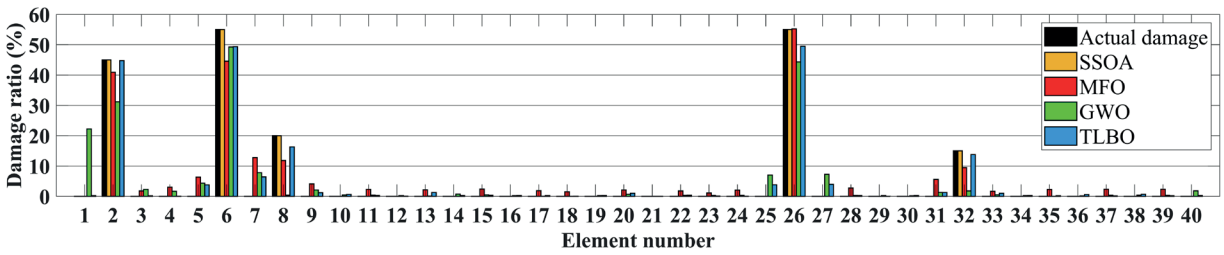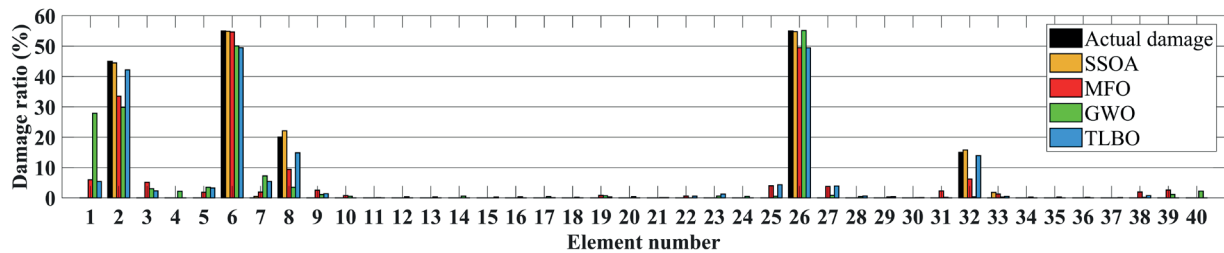
Fig. 7 Comparison of the average value of damage detection results from different metaheuristics for the scenario I of the 40-element continuous beam in: (a) noise-free state, (b) noisy state

Fig. 8 Comparison of the average value of damage detection results from different metaheuristics for the scenario II of the 40-element continuous beam in: (a) noise-free state, (b) noisy state
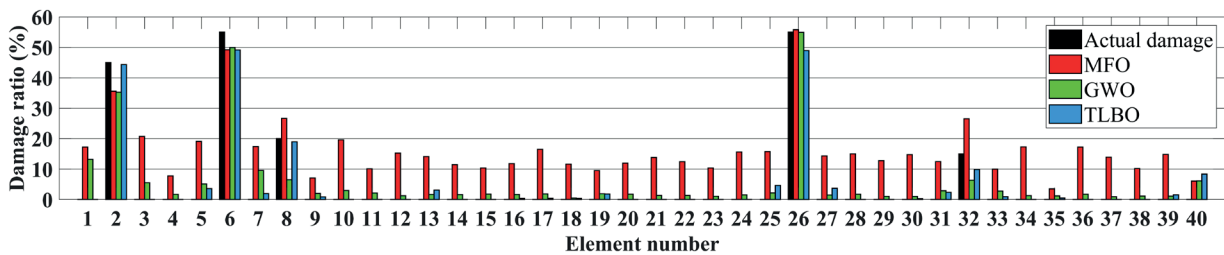

Fig. 9 Comparison of the average value of damage detection results using WBS found by MFO, GWO, and TLBO for scenario II of the 40-element continuous beam in the noisy state

Table 4 Comparison of statistical damage identification results using BS in the 40-element continuous beam for both damage scenarios in the case with noise and without noise

Scenario	State	Actual location	Actual ratio	SSOA		TLBO		MFO		GWO	
				Avg. value	Std. value	Avg. value	Std. value	Avg. value	Std. value	Avg. value	Std. value
I		7	35	35	1.66E-6	34.97	0.1922	28.96	14.6466	30.78	10.2790
		20	10	10	2.88E-6	10.05	0.1135	6.92	6.3282	4.27	5.1799
	Noise-free	37	60	60	3.98E-7	60.04	0.0393	60.70	1.0909	60.17	0.1289
		Error (%)		1.17E-7	4.79E-8	0.0997	0.0882	1.4111	1.7126	0.8636	0.6306
		NSAs		3964		13828		13714		18972	
		7	35	34.96	3.11E-5	33.81	0.6507	24.31	15.9213	30.88	10.3096
		20	10	10.70	6.87E-5	7.66	3.8591	3.67	5.6206	3.72	4.5616
	Noisy	37	60	59.97	7.53E-6	59.62	0.1730	42.11	27.5657	60	0.1774
		Error (%)		0.0194	1.77E-6	0.3624	0.2221	2.1939	1.5369	0.8746	0.7454
		NSAs		5400		14764		8066		18702	
II		2	45	45	1.36E-4	44.73	1.4099	40.91	13.9595	31.15	20.4255
		6	55	55	3.08E-5	49.34	16.4525	44.58	22.4081	49.26	16.4396
		8	20	20	9.79E-6	16.29	8.2478	11.83	9.7590	0.40	1.1870
	Noise-free	26	55	55	5.52E-5	49.47	16.5006	55.23	2.9115	44.34	22.1782
		32	15	15	1.11E-4	13.77	5.0890	9.48	7.9751	1.80	3.6374
		Error (%)		3.06E-6	8.96E-6	1.2417	1.7322	2.4310	3.4466	3.2446	2.5940
		NSAs		9920		16632		11338		19696	
		2	45	44.44	0.0843	42.15	14.1305	33.49	21.9301	29.82	19.3838
		6	55	54.88	0.0990	49.44	16.4957	54.66	1.9565	50.03	16.6915
		8	20	22.09	0.2129	14.85	7.4640	9.39	9.5739	3.50	5.3420
	Noisy	26	55	54.74	0.0687	49.39	16.4998	49.45	16.5186	55.11	1.3397
		32	15	15.77	0.7954	13.90	2.2986	6.23	6.3742	0.34	1.0284
		Error (%)		0.1597	0.0855	1.4316	1.6151	1.9666	1.5646	2.7639	2.0630
		NSAs		8422		16120		10534		19760	

figure and Fig. 8(b), it can be concluded that when BS in comparison to WBS is employed, the results obtained by BS are much better than those found by using WBS.

Table 4 gives statistical results for investigated algorithms together with the results found by SSOA when BS is employed. These statistical results are including average values of damage ratios, Errors, and NSAs. The standard deviations of damage ratios and Errors are given in this table as well. All these values reveal that SSOA reaches the best results among the other algorithms in both ideal and noisy states. The most important point is that the average NSAs obtained by SSOA is significantly smaller than the other algorithms in all cases. Likewise, from the inspecting of Tables 4 and 5 in damage Scenario II of the beam, it can be seen that when BS is applied, TLBO, MFO, and GWO algorithms gain better results than when WBS is employed.

4.3 23-element asymmetrical planar frame

The third test example is considered a 23-element asymmetrical planar frame. The finite element model of this frame is made up of 23 elements, which include 14 columns and 9 beams as illustrated in Fig. 10. The frame has 14 free nodes and each node has three DOFs, leading to 42 total DOFs. For beam elements, the cross-sectional area, mass per unit length, and moment of inertia are respectively equal to $A_{beam} = 1.62 \times 10^{-2}$ m^2, $\bar{m} = 1300$ kg/m, and $I_{beam} = 3.85 \times 10^{-4}$ m^4, whereas these values for column elements are respectively equal to $A_{column} = 1.6 \times 10^{-2}$ m^2, $\bar{m} = 125.6$ kg/m, and $I_{column} = 3.5 \times 10^{-4}$ m^4. Furthermore, modulus of elasticity and material density for all elements are the same and respectively equal to $E = 200$ GPa and $\rho = 7850$ kg/m^3. Two different damage scenarios are assumed as presented in Table 6.

Table 5 Comparison of statistical damage identification results obtained by different algorithms using WBS for scenario II of the 40-element continuous beam in the case with noise and without noise

| Noise level	Actual location	Actual ratio	TLBO		MFO		GWO	
			Avg. value	Std. value	Avg. value	Std. value	Avg. value	Std. value
Noise-free	2	45	36.12	18.0612	35.78	30.7190	35.75	17.8866
	6	55	54.72	1.1301	60.99	21.8961	49.24	15.0559
	8	20	20.76	1.0692	37.69	19.6206	8.73	9.6139
	26	55	54.93	0.6419	65.84	10.2809	49.78	16.6101
	32	15	13.07	4.7426	25.40	25.7395	7.96	6.8583
	Error (%)		0.8986	0.9816	22.3001	15.7348	2.8388	2.1340
	NSAs		16144		18596		19884	
Noisy	2	45	44.34	3.3786	35.57	23.8740	35.29	16.3648
	6	55	49.13	16.3893	49.20	25.3673	49.95	16.6521
	8	20	18.98	6.8690	26.67	20.8168	6.50	8.3067
	26	55	48.91	16.3765	55.83	19.8007	54.92	1.2156
	32	15	9.88	8.3634	26.53	17.0036	6.34	5.6541
	Error (%)		1.4825	1.7455	13.6565	12.0996	3.1814	1.8289
	NSAs		16256		16122		19890	
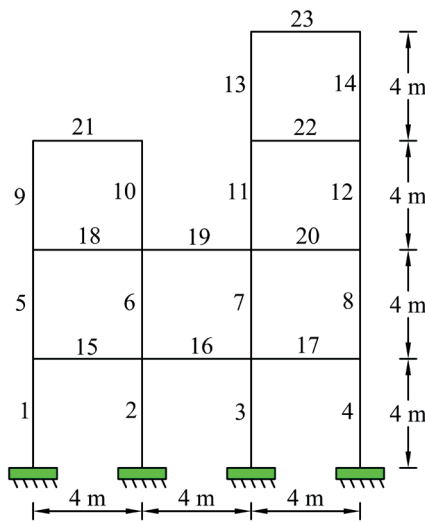
Fig. 10 Finite element model of the 23-element asymmetrical planar frame

Table 6 Two different damage scenarios in the asymmetrical 23-element planar frame

Scenario	I		II		
Element no.	4	10	4	18	21
Damage ratio (%)	15	25	15	35	20

In this example, to verify the superiority of the proposed cost function, one other cost function is investigated here for comparison. This cost function consists of three parts. The first and second parts are respectively related to the discrepancy between natural frequencies and mode shapes of the measured structure and analytical model.

The third part is a penalty against too many damage locations so that it weights against an increasing number of damage locations. This cost function is as follows [40]:

$$E(X) = \left(1 + \gamma \cdot m^d(X)\right) \times \left(\sum_{i=1}^{nmod} \left\| \frac{\phi_i^d - \phi_i^a}{\phi_i^d + \phi_i^a} \right\| + \sum_{i=1}^{nmod} \left\| \frac{\left(\omega_i^d - \omega_i^a\right)^2}{\left(\omega_i^d\right)^2} \right\| \right) \quad (21)$$

Like the previous examples, it is assumed that the first five mode's data are available for comparison. The average of damage detection results by employing the $F(X)$ and $E(X)$ and using BS for both damage scenarios are presented in Fig. 11. Although in both damage scenarios both examined cost functions can identify damaged elements, there are many false predictions for the results found by $E(X)$. Moreover, $F(X)$ using BS can precisely locate the actual location of both damage scenarios in this frame, even for noisy state. To compare the results obtained by BS with those found by WBS using $E(X)$, Scenario II of this frame in the case with noise and without noise is selected for comparison as depicted in Fig. 12. As can be seen, when WBS is employed, the results significantly get worse and the error index is increased. When $E(X)$ using WBS is incorporated by BS, the obtained results get much better even for noise-contaminated data. In this regard, it can be concluded that the BS is capable to improve the performance of $E(X)$ as well.

For further examination, Table 7 provides the statistical results consisting of the average and standard deviation values of damage ratios and errors. Furthermore, the
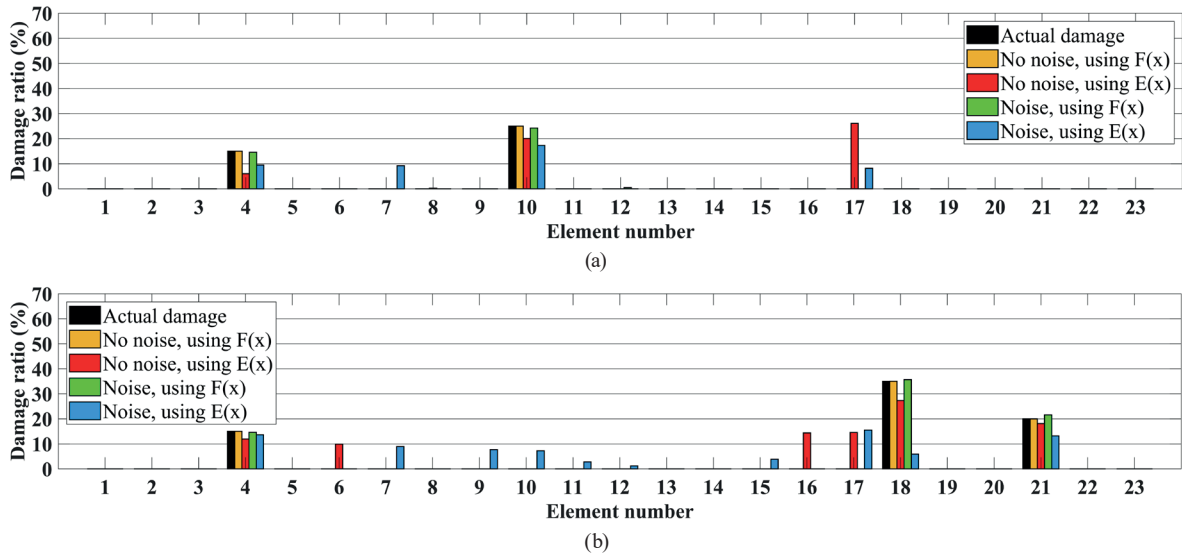
(a)

(b)

Fig. 11 Comparison of damage detection results for the 23-element asymmetrical planar frame obtained from the SSOA using two different cost functions for (a) Scenario I, and (b) Scenario II
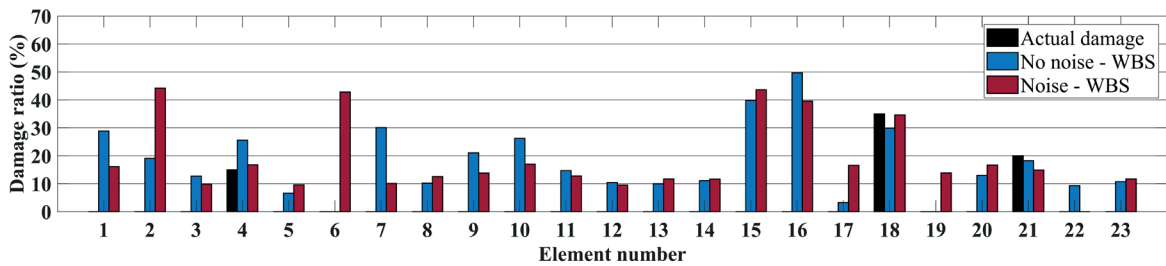

Fig. 12 Obtained damage detection results for Scenario II of the 23-element asymmetrical planar frame using $E(X)$ and WBS in the noisy condition

Table 7 Comparison of statistical damage identification results obtained by different cost functions for both scenarios of the 23-element asymmetrical planar frame in the case with noise and without noise

Scenario	Noise level	Actual location	Actual ratio	BS, using F (X)		BS, using E(X)		WBS, using E(X)	
				Avg. value	Std. value	Avg. value	Std. value	Avg. value	Std. value
I	Noise-free	4	15	15	1.26E-6	6.00	7.35	20.71	10.8528
		10	25	25	5.55E-7	20.01	10.01	30.46	15.1078
		Error (%)		9.84E-8	5.11E-8	1.7533	1.7458	11.9748	10.1250
		NSAs		3538		7004		19320	
	Noisy	4	15	14.56	0.0823	9.47	7.7532	20.67	13.6939
		10	25	24.19	0.0090	17.28	11.3157	36.37	10.8711
		Error (%)		0.0754	0.0171	1.3798	1.6445	12.3229	6.4175
		NSAs		5824		4326		19282	
II	Noise-free	4	15	15	7.39E-7	11.94	5.9714	25.58	13.4528
		18	35	35	7.34E-6	27.33	13.7262	29.90	30.4757
		21	20	20	2.17E-6	18.09	6.0329	18.24	20.0152
		Error (%)		3.88E-7	2.19E-7	2.2448	4.1413	16.8918	7.1453
		NSAs		4002		10066		19840	
	Noisy	4	15	14.58	8.94E-7	13.64	12.5562	16.78	16.1385
		18	35	35.65	5.9E-6	5.92	17.7746	34.59	37.1335
		21	20	21.60	1.49E-6	13.18	12.4078	14.91	16.9992
		Error (%)		0.1166	2.42E-7	4.3392	3.3973	18.5783	9.2930
		NSAs		3282		7544		19388	

average of NSAs in both cost functions for both damage scenarios is presented in this table. A close examination of this table reveals that: (1) the average of identified damage is much close to actual damage in all cases when $F(X)$ is employed using BS. (2) the average and standard deviation of errors determined by $F(X)$ are much better than $E(X)$. (3) the average NSA in the case $F(X)$ is employed for damage detection is much better than those obtained by $E(X)$. As a result, in a general view using $F(X)$ has better performance than using $E(X)$.

Table 8 Two different damage scenarios in the 72-bar spatial truss

Scenario	I		II	
Element no.	5	1	21	37
Damage ratio (%)	30	25	20	30

4.4 72-bar spatial truss

The last example is considered a 72-bar spatial truss as a large-scale test example. Four nonstructural masses are added to the fourth story nodes in which each mass has the weight equal to 2270 kg as shown in Fig. 13. The truss
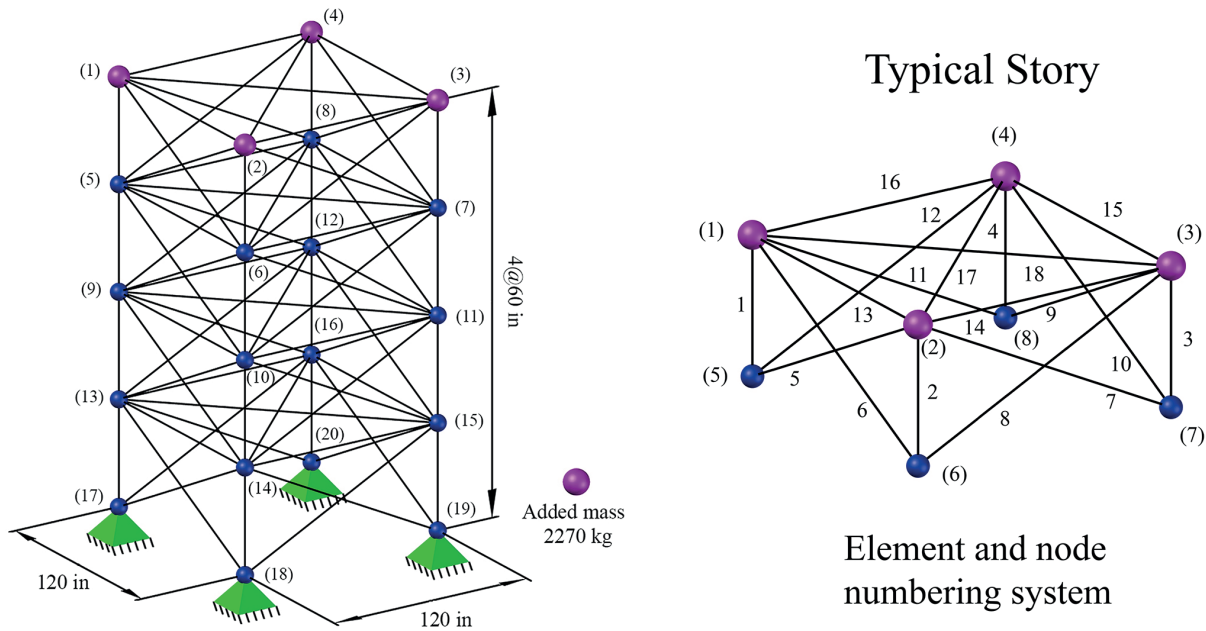

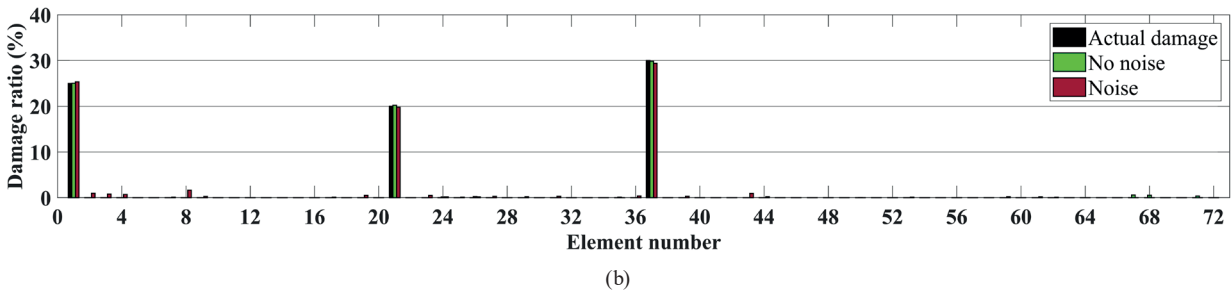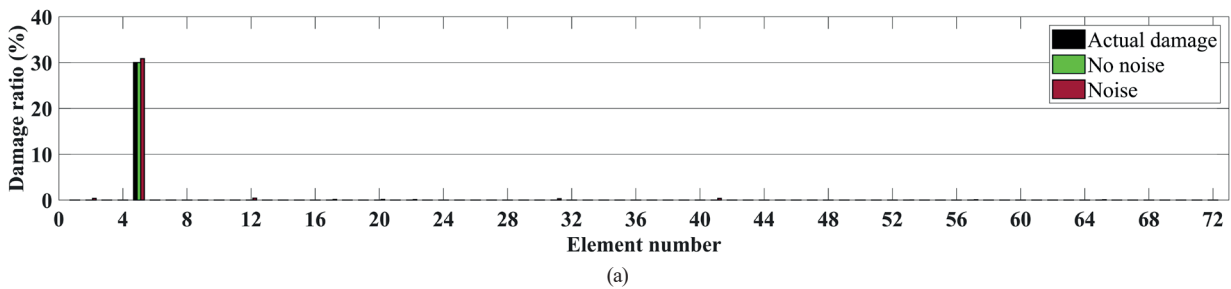**Fig. 13** Finite element model of the 72-bar spatial truss

Fig. 14 Damage detection results obtained by SSOA for the 72-bar spatial truss: (a) Scenario I, (b) Scenario II

Table 9 Comparison of statistical damage identification results for both scenarios of the 72-bar spatial truss in the case with noise and without noise

Scenario	Noise level	Actual location		Actual ratio	Avg. value	Std. value
		5		30	30	0.0088
	Noise-free		Error (%)		2.89E-4	9.14E-4
			NSAs		11704	
I		5		30	30.85	0.0606
	Noisy		Error (%)		0.0407	0.0433
			NSAs		9978	
		1		25	25.03	0.1069
		21		20	20.21	0.5871
	Noise-free	37		30	29.82	0.4998
			Error (%)		0.0323	0.1020
			NSAs		11809	
II		1		25	25.31	0.2523
		21		20	19.76	1.4704
	Noisy	37		30	29.38	0.3828
			Error (%)		0.1626	0.1673
			NSAs		16436	

has 16 free nodes, leading to 48 active DOFs. For each element, the modulus of elasticity, material density, and cross-sectional area are respectively $E = 69.8$ GPa and $\rho = 2770$ kg/m^3, and $A = 25$ cm^2. Two different damage scenarios are considered as given in Table 8.

Fig. 14 presents the average value of damage detection results in the case with noise and without noise for both damage scenarios. From this figure, although the structure has many elements, it is clear that even in the noisy condition all damaged elements are identified with a high level of accuracy. Like previous examples, the statistical results in both noise-free and noisy states for damage Scenarios I and II are provided in Table 9. A close investigation of this table reveals that the number of successful runs is equal to 100 %, and the proposed method can detect damaged elements with a maximum average error equal to 0.1626 %.

5 Conclusions

This study presents a new strategy called Boundary Strategy (BS) in the process of optimization-based damage detection problem. In this strategy, despite the conventional damage detection methods that only zero values in the vector of design variables represent healthy elements,

the range between -1 and 0 represents healthy ones. BS gradually neutralizes the effects of structural elements that are healthy in the optimization process. This strategy leads to the complexity of the search space decreases. Shuffled Shepherd Optimization Algorithm (SSOA) as a new multi-community metaheuristic is considered to solve the problem. The damage-sensitive cost function is established by utilizing vibration data with a penalty function. To evaluate the capability of the proposed method, several examples were considered. They include a 25-bar planar truss, a 40-element continuous beam, a 23-element asymmetrical planar frame, and a large-scale 72-bar spatial truss. In the first numerical examples, the performance of the BS in comparison to WBS in identifying and quantifying damage was examined. The SSOA was compared with three well-known metaheuristics namely TLBO, GWO, and MFO in the second example. In the third example, the proposed cost function was compared with one other cost function. In the last example, a large-scale truss with 72 design variables is checked with the proposed BS. All obtained results indicate that SSOA considering BS and the proposed cost function have proper functioning for both noisy state and large-scale problems.

References

[1] Kaveh, A. "Applications of Metaheuristic Optimization Algorithms in Civil Engineering", Springer, Cham, Switzerland, 2017.
https://doi.org/10.1007/978-3-319-48012-1

[2] Dinh-Cong, D., Nguyen-Thoi, T., Nguyen, D. T. "A FE model updating technique based on SAP2000-OAPI and enhanced SOS algorithm for damage assessment of full-scale structures", Applied Soft Computing, 89, Article number: 106100, 2020.
https://doi.org/10.1016/j.asoc.2020.106100

[3] Zimmerman, D. C., Kaouk, M. "Eigenstructure assignment approach for structural damage detection", AIAA Journal, 30(7), pp. 1848–1855, 1992.
https://doi.org/10.2514/3.11146

[4] Alkayem, N. F., Cao, M., Zhang, Y., Bayat, M., Su, Z. "Structural damage detection using finite element model updating with evolutionary algorithms: a survey", Neural Computing and Applications, 30, pp. 389–411, 2018.
https://doi.org/10.1007/s00521-017-3284-1

[5] Marwala, T. "Finite-element-model Updating Using Computional Intelligence Techniques", Springer, London, UK, 2010.
https://doi.org/10.1007/978-1-84996-323-7

[6] Ren, W.-X., Chen, H.-B. "Finite element model updating in structural dynamics by using the response surface method", Engineering Structures, 32(8), pp. 2455–2465, 2010.
https://doi.org/10.1016/j.engstruct.2010.04.019

[7] Mottershead, J. E., Link, M., Friswell, M. I. "The sensitivity method in finite element model updating: A tutorial", Mechanical Systems and Signal Processing, 25(7), pp. 2275–2296, 2011.
https://doi.org/10.1016/j.ymssp.2010.10.012

[8] Hou, R., Xia, Y., Bao, Y., Zhou, X. "Selection of regularization parameter for l1-regularized damage detection", Journal of Sound and Vibration, 423, pp. 141–160, 2018.
https://doi.org/10.1016/j.jsv.2018.02.064

[9] Blachowski, B. "Modal Sensitivity Based Sensor Placement for Damage Identification Under Sparsity Constraint", Periodica Polytechnica Civil Engineering, 63(2), pp. 432–445, 2019.
https://doi.org/10.3311/PPci.13888

[10] Dehcheshmeh, M. M., Hosseinzadeh, A. Z., Amiri, G. G. "Feasibility study on model-based damage detection in shear frames using pseudo modal strain energy", Smart Structures and Systems, 25(1), pp. 47–56, 2020.
https://doi.org/10.12989/sss.2020.25.1.047

[11] Seyedpoor, S. M. "A two stage method for structural damage detection using a modal strain energy based index and particle swarm optimization", International Journal of Non-Linear Mechanics, 47(1), pp. 1–8, 2012.
https://doi.org/10.1016/j.ijnonlinmec.2011.07.011

[12] Kaveh, A., Hamedani, K. B., Hosseini, S. M., Bakhshpoori, T. "Optimal design of planar steel frame structures utilizing meta-heuristic optimization algorithms", Structures, 25, pp. 335–346, 2020.
https://doi.org/10.1016/j.istruc.2020.03.032

[13] Khatir, S., Belaidi, I., Serra, R., Abdel Wahab, M., Khatir, T. "Numerical study for single and multiple damage detection and localization in beam-like structures using BAT algorithm", Journal of Vibroengineering, 18(1), pp. 202–213, 2016. [online] Available at: https://www.jvejournals.com/article/16550

[14] Kourehli, S. S. "Damage Diagnosis of Structures Using Modal Data and Static Response", Periodica Polytechnica Civil Engineering, 61(1), pp. 135–145, 2017.
https://doi.org/10.3311/PPci.7646

[15] Bureerat, S., Pholdee, N. "Adaptive Sine Cosine Algorithm Integrated with Differential Evolution for Structural Damage Detection", In: International Conference on Computational Science and Its Applications, Trieste, Italy, 2017, pp. 71–86.
https://doi.org/10.1007/978-3-319-62392-4_6

[16] Dinh-Cong, D., Ho-Huu, V., Vo-Duy, T., Ngo-Thi, H. Q., Nguyen-Thoi, T. "Efficiency of Jaya algorithm for solving the optimization-based structural damage identification problem based on a hybrid objective function", Engineering Optimization, 50(8), pp. 1233–1251, 2018.
https://doi.org/10.1080/0305215X.2017.1367392

[17] Dinh-Cong, D., Dang-Trung, H., Nguyen-Thoi, T. "An efficient approach for optimal sensor placement and damage identification in laminated composite structures", Advances in Engineering Software, 119, pp. 48–59, 2018.
https://doi.org/10.1016/j.advengsoft.2018.02.005

[18] Ghasemi, M. R., Nobahari, M., Shabakhty, N. "Enhanced optimization-based structural damage detection method using modal strain energy and modal frequencies", Engineering with Computers, 34, pp. 637–647, 2018.
https://doi.org/10.1007/s00366-017-0563-5

[19] Ghiasi, R., Ghasemi, M. R., Noori, M. "Comparative studies of metamodeling and AI-Based techniques in damage detection of structures", Advances in Engineering Software, 125, pp. 101–112, 2018.
https://doi.org/10.1016/j.advengsoft.2018.02.006

[20] Kaveh, A., Dadras, A. "Structural damage identification using an enhanced thermal exchange optimization algorithm", Engineering Optimization, 50(3), pp. 430–451, 2018.
https://doi.org/10.1080/0305215X.2017.1318872

[21] Tiachacht, S., Bouazzouni, A., Khatir, S., Wahab, M. A., Behtani, A., Capozucca, R. "Damage assessment in structures using combination of a modified Cornwell indicator and genetic algorithm", Engineering Structures, 177, pp. 421–430, 2018.
https://doi.org/10.1016/j.engstruct.2018.09.070

[22] Jahangiri, M., Najafgholipour, M. A., Dehghan, S. M., Hadianfard, M. A. "The efficiency of a novel identification method for structural damage assessment using the first vibration mode data", Journal of Sound and Vibration, 458, pp. 1–16, 2019.
https://doi.org/10.1016/j.jsv.2019.06.011

[23] Mishra, M., Barman, S. K., Maity, D., Maiti, D. K. "Ant lion optimisation algorithm for structural damage detection using vibration data", Journal of Civil Structural Health Monitoring, 9, pp. 117–136, 2019.
https://doi.org/10.1007/s13349-018-0318-z

[24] Hosseini, S. M., Ghodrati Amiri, G., Mohamadi Dehcheshmeh, M. "Efficiency evaluation of proposed objective functions in structural damage detection based on modal strain energy and flexibility approaches", International Journal of Optimization in Civil Engineering, 10(1), pp. 71–90, 2020. [online] Available at: http://ijoce.iust.ac.ir/article-1-421-en.html

[25] Mishra, M., Barman, S. K., Maity, D., Maiti, D. K. "Performance Studies of 10 Metaheuristic Techniques in Determination of Damages for Large-Scale Spatial Trusses from Changes in Vibration Responses", Journal of Computing in Civil Engineering, 34(2), Article number: 04019052, 2020.
https://doi.org/10.1061/(ASCE)CP.1943-5487.0000872

[26] Kaveh, A., Zolghadr, A. "An improved CSS for damage detection of truss structures using changes in natural frequencies and mode shapes", Advances in Engineering Software, 80, pp. 93–100, 2015.
https://doi.org/10.1016/j.advengsoft.2014.09.010

[27] Bureerat, S., Pholdee, N. "Inverse problem based differential evolution for efficient structural health monitoring of trusses", Applied Soft Computing, 66, pp. 462–472, 2018.
https://doi.org/10.1016/j.asoc.2018.02.046

[28] Friswell, M. I., Penny, J. E. T., Garvey, S. D. "A combined genetic and eigensensitivity algorithm for the location of damage in structures", Computers & Structures, 69(5), pp. 547–556, 1998.
https://doi.org/10.1016/S0045-7949(98)00125-4

[29] Kim, N.-I., Kim, S., Lee, J. "Vibration-based damage detection of planar and space trusses using differential evolution algorithm", Applied Acoustics, 148, pp. 308–321, 2019.
https://doi.org/10.1016/j.apacoust.2018.08.032

[30] Guedria, N. B. "An accelerated differential evolution algorithm with new operators for multi-damage detection in plate-like structures", Applied Mathematical Modelling, 80, pp. 366–383, 2020.
https://doi.org/10.1016/j.apm.2019.11.023

[31] Kaveh, A., Zaerreza, A. "Shuffled shepherd optimization method: a new Meta-heuristic algorithm", Engineering Computations, 37(7), pp. 2357–2389, 2020.
https://doi.org/10.1108/EC-10-2019-0481

[32] Kaveh, A., Zaerreza, A. "Size/Layout Optimization of Truss Structures Using Shuffled Shepherd Optimization Method", Periodica Polytechnica Civil Engineering, 64(2), pp. 408–421, 2020.
https://doi.org/10.3311/PPci.15726

[33] Rao, R. V., Savsani, V. J., Vakharia, D. P. "Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems", Computer-Aided Design, 43(3), pp. 303–315, 2011.
https://doi.org/10.1016/j.cad.2010.12.015

[34] Mirjalili, S., Mirjalili, S. M., Lewis, A. "Grey Wolf Optimizer", Advances in Engineering Software, 69, pp. 46–61, 2014.
https://doi.org/10.1016/j.advengsoft.2013.12.007

[35] Mirjalili, S. "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm", Knowledge-Based Systems, 89, pp. 228–249, 2015.
https://doi.org/10.1016/j.knosys.2015.07.006

[36] Kaveh, A., Zaerreza, A., Hosseini, S. M. "Shuffled Shepherd Optimization Method Simplified for Reducing the Parameter Dependency", Iranian Journal of Science and Technology, Transactions of Civil Engineering, 2020.
https://doi.org/10.1007/s40996-020-00428-3

[37] Farrar, C. R., Jauregui, D. A. "Comparative study of damage identification algorithms applied to a bridge: I. Experiment", Smart Materials and Structures, 7(5), pp. 704–719, 1998.
https://doi.org/10.1088/0964-1726/7/5/013

[38] Cao, M. S., Sha, G. G., Gao, Y. F., Ostachowicz, W. "Structural damage identification using damping: a compendium of uses and features", Smart Materials and Structures, 26(4), Article number: 043001, 2017.
https://doi.org/10.1088/1361-665X/aa550a

[39] Fan, W., Qiao, P. "Vibration-based Damage Identification Methods: A Review and Comparative Study", Structural Health Monitoring, 10(1), pp. 83–111, 2011.
https://doi.org/10.1177/1475921710365419

[40] Kaveh, A., Maniat, M. "Damage detection based on MCSS and PSO using modal data", Smart Structures and Systems, 15(5), pp. 1253–1270, 2015.
https://doi.org/10.12989/sss.2015.15.5.1253