

## Research Article

# A Novel Artificial Bee Colony Algorithm for Structural Damage Detection

Yinghao Zhao , Quansheng Yan , Zheng Yang, Xiaolin Yu , and Buyu Jia 

*School of Civil Engineering and Transportation, South China University of Technology, Guangzhou, China*

Correspondence should be addressed to Buyu Jia; ctjby@scut.edu.cn

Received 3 July 2019; Revised 13 December 2019; Accepted 27 December 2019; Published 17 January 2020

Academic Editor: Hossein Moayedi

Copyright © 2020 Yinghao Zhao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A novel artificial bee colony (ABC) algorithm to detect structural damage via modal and frequency analyses is proposed (named as TCABC algorithm). Compared to the standard ABC algorithm, tabu search method and chaotic search method are adopted in the proposed algorithm to enhance the exploration and exploitation ability. The tabu search method uses a memory function to avoid the solution being trapped in a local minimum, which increases the exploitation ability. Chaotic search method generates more searching points for finding the global minimum, which increases the exploration ability. Additionally, the first roulette wheel selection is replaced by the tournament selection to enhance the global searching ability of the TCABC algorithm. Several explicit test functions and an implicit damage detection function are employed to check the numerical results obtained from ABC and TCABC algorithms. Afterward, the damage detection accuracy of the TCABC algorithm is verified under different circumstances, and several recommendations are given for using the TCABC algorithm to detect structural damages under actual conditions. Finally, an experimental study is applied to examine the performance of TCABC algorithm for damage detection. The results show the following: (1) compared to traditional ABC algorithm, TCABC algorithm performs better; (2) fewer groups lead to faster convergence as demonstrated by both algorithms used in the same damage situation; (3) TCABC algorithm can infer the locations and extents of the damage when the groupings are inaccurate; (4) the accuracy of the field test data profoundly affects the precision of the damage detection results. In other words, stronger noises result in worse identification results; (5) whether or not the noises exist, the more data are measured, the more accurate the results can be achieved; (6) the TCABC algorithm can efficiently detect structural damage in the experimental study.

## 1. Introduction

Early-stage damage detection is one of the main tasks in structural health monitoring [1], and this task is accomplished by fundamental damage detection method and parameter identification. Once the structural parameters are identified or the structural uncertainties are reduced, the corresponding structural numerical model can be updated and used to predict relevant structural responses under possible catastrophic events, such as earthquake and hurricane [2, 3]. Besides, with the detected damages, structural repair cost and maintenance parameters can be dynamically evaluated [4]. However, traditional local damage detection methods, such as the visual detection method and x-ray method, require specific prior knowledge, which is difficult to obtain. Hence, global damage detection methods based on

physical quantities, such as natural frequencies [5, 6], modal shapes [7], accelerations [8, 9], static displacements, and strain responses [10, 11], become popular.

For these global damage detection methods, an idea to obtain structural damages is to transfer the damage detection problem to the finite element (FE) model updating problem; the theoretical basis of the updating is that, in numerical calculation, a real structure with infinite degrees of freedom can be discretized into the FE model. The structural vibration properties (i.e., frequencies and modal shapes) vary with the changes in physical parameters (i.e., stiffness and mass) of the FE model. Thus, obtaining physical parameters via vibration properties is an inverse problem of modal and frequency analyses or a pattern-recognition problem [12]. One main difficulty of this pattern-recognition problem lies in a large number of unknowns. To simplify the

problem, Moslem and Nafaspour [13] assumed that any change in the properties of each element due to the damage is equivalent to a change in its elastic modulus. By this simplification, the FE model updating-based damage detection problem can be formulated as an optimization problem with a simple-structured implicit objective function. For example, the norm of the residuals between the numerical results of the FE model and the test data of the actual structure is defined as the objective function. Once the objective function reaches the global minimum, the input parameters of the FE model will be treated as the physical parameters of the actual structure.

The structural damage detection problem is transferred to find the extreme values of the corresponding objective function. However, these extrema cannot be seen by the traditional method because the objective function is expressed by implicit functions. To overcome this problem, several metaheuristic algorithms and artificial intelligence (AI) are used. Perera and Torres [14] identified the damage of a simply supported beam by genetic algorithm (GA), and these detection results are better than those calculated with the method that Hu et al. [15] proposed. Park et al. [16] updated bridge boundary conditions using artificial neural networks (ANN) and verified the results by both laboratory and field tests; it was concluded that the ANN reduces the uncertainties of the boundary conditions in FE model updating. Pathirage et al. [12] introduced dimensionality reduction and relationship learning in the ANN to optimize the weights in neural networks; the laboratory test indicated that the detection results of this method are better than the results of the traditional ANN. Qiao and Yang [17] introduced a quantum search dolphin swarm algorithm (QDSA) in finding the optimal solution of large-scale functions; they confirmed that the QDSA has outstanding advantages than dolphin swarm algorithm (DSA). Kang et al. [18] proposed an improved particle swarm optimization (PSO) algorithm to identify structural damages using vibration data, and the results showed that compared to differential evolution (DE) algorithm, the improved PSO algorithm is more efficient in determining the sites and the extents of structural damages. Qiao et al. [19] combined the Volterra adaptive filter and an improved whale optimization algorithm (WOA) to predict the short-term natural gas consumption; the study showed that the proposed prediction model is better than some advanced prediction models.

Besides the abovementioned algorithms, the artificial bee colony (ABC) algorithm also attracted the attention of many scholars in the optimization area since it was proposed. ABC algorithm was suggested by Karaboga [20], and this algorithm is motivated by the intelligent behavior of honey bees when seeking a high-quality food source. In 2009, Karaboga and Akay [21] first conducted a detailed and comprehensive analysis of ABC algorithm and tested it with 50 numerical benchmark functions and other well-known evolutionary algorithms such as GA, PSO, DE, and ant colony algorithm (ACO); the results indicate ABC algorithm has several advantages in getting the extrema of the function than other methods. In the same year, the influences of changing parameters in the ABC algorithm were analyzed by Akay and

Karaboga [22]. With the benefits such as simple structure, easy implementation, and outstanding performance, the ABC algorithm is widely used in many fields. Awan et al. [23] applied the ABC algorithm to optimize the connection weights of neurons in the ANN for electric load forecasting. Uzlu et al. [24] combined ABC and ANN to predict the annual output of hydroelectric power in Turkey. Menon and Ramakrishnan [25] used the ABC algorithm to process brain tumors segmentation in magnetic resonance images. Wen et al. [26] applied the ABC algorithm to achieve global optimization of controlled-source audio-frequency magnetotelluric data. However, there are limited researchers utilizing the ABC algorithm in structural damage detection. Combining the ABC algorithm with the Nelder–Mead simplex method, Sun and Betti [27] identified the structural parameters and the input forces of buildings even when the structural responses are noise contaminated. By applying the hybrid search strategy in the ABC algorithm for structural damage detection, Ding et al. [28] got better identification results compared with those from GA and traditional ABC algorithms. Nevertheless, there are some flaws in Sun and Betti [27] and Ding et al. [28]. The calculations of Sun and Betti [27] are purely numerical-based and lack of experimental verification. The data from the experiment are different from those artificially assumed and used in the numerical models. So, an experimental study should be used as a verification after the numerical study; Ding et al. [28] did not consider several factors that influence the target function and corresponding identification results. However, the implicit target function highly relies on the observed frequencies and modal shapes, and these observed data are easily affected by the test conditions such as grouping, noise effect, mode shape order, and sensor location, which should be taken into account.

Based on the fact that there are limited research studies in ABC-based structural damage detection, and the assumed identification condition is relatively simple. In this study, ABC algorithm combined with the tabu search method [29] and chaotic search method [30] (named as TCABC algorithm) is used to detect structural damages under different identify conditions such as grouping, noise effect, mode shape order, sensor location and is also evaluated by several explicit test functions. Recommendations for realistically using the TCABC algorithm in damage detection are also put forward. Furthermore, an experimental study is used to examine the damage detection results of TCABC algorithm.

The rest of the paper is organized as follows. Section 2 describes how the structural damage detection problem is transferred to an optimization problem; traditional ABC algorithm and proposed TCABC algorithm are presented in Section 3, and their numerical results are compared in Section 4; also, in Section 4, the accuracy of TCABC algorithm for damage detection is verified under different circumstances (i.e., influences of grouping size, noise effect, mode shape order, and sensor location), and several recommendations are given for using TCABC algorithm to detect structural damages under actual conditions; in Section 5, an experimental study is used to examine the performance of TCABC algorithm; Section 6 is the conclusions.

## 2. Theoretical Background

**2.1. Parameterization of Damages.** When a real structure is discretized into an N-DOF system, the modal characteristics can be obtained by

$$[\mathbf{K} - \omega_j^2 \cdot \mathbf{M}] \cdot \{\Phi_j\} = 0, \quad j = 1, 2, \dots, N, \quad (1)$$

where  $\mathbf{M}$  and  $\mathbf{K}$  are the  $N \times N$  global mass and stiffness matrices of the structure, respectively,  $\omega_j$  is the  $j$ th eigenvalue, and  $\Phi_j$  is the corresponding modal vector.

For actual structures, the  $\mathbf{M}$  (or  $\mathbf{K}$ ) for the damaged and the undamaged ones are different. However, in damage detection problems, the mass matrix is assumed constant because the internal structural damage usually does not result in losses of materials, and the stiffness of the  $i$ th element is expressed as follows [13]:

$$\mathbf{K}_i = f(A_i, I_i, J_i, t_i, L_i, E_i, G_i), \quad (2)$$

in which  $A_i$ ,  $I_i$ , and  $J_i$  are cross-sectional area, bending, and polar moments of the inertia of the  $i$ th element;  $t_i$  and  $L_i$  are the element dimensions; and  $E_i$  and  $G_i$  are, respectively, the elastic and shear modulus. To reduce the number of input parameters in the FE model, Moslem and Nafaspour [13] mentioned that any change in the properties of each element due to the damage is equivalent to the change in its elastic modulus ( $E_i$ ). According to the abovementioned statements, the stiffness of the  $i$ th element can be expressed as follows:

$$\mathbf{K}_i = (1 - a_i) \cdot E_i, \quad (3)$$

where  $a_i$  is defined as the damage index;  $a_i = 0$  represents no damage and  $a_i = 1$  represents absolute damage of the  $i$ th element.

**2.2. Implicit Objective Function.** Theoretically, when the input parameters of the FE model are exactly the same as the physical parameters of the damaged structure, for both the FE model and real structure, their natural frequencies should be the same and the value of modal assurance criterion (MAC) [31] should be equal to one. So, the objective function can be set as a combination of the natural frequencies and the values of MAC. Perera and Torres [14] used GA with two objective functions to detect the damage of a beam, and the objective function with better performance is used in this paper. The used MAC is defined as follows:

$$\text{MAC}([\Phi_j^T], [\Phi_j^F]) = \frac{\left([\Phi_j^T]^t [W] [\Phi_j^F]\right)^2}{\left([\Phi_j^T]^t [W] [\Phi_j^T]\right) \left([\Phi_j^F]^t [W] [\Phi_j^F]\right)}, \quad (4)$$

where  $\Phi_j^T$  and  $\Phi_j^F$  are, respectively, the  $j$ th test and FE modal vectors,  $t$  represents transpose, and  $[W]$  is a diagonal square weight matrix to ensure all test data are effectively used. The  $k$ th diagonal element of  $[W]$  is defined as follows:

$$W_{kk} = \left| \left( \Phi_{jk}^T \right)^{-1} \right|, \quad (5)$$

where  $\Phi_{jk}^T$  is the  $k$ th coordinate of the  $j$ th test modal vector. The objective function is defined as follows:

$$F = 1 - \prod_{j=1}^m \frac{\text{MAC}([\Phi_j^T], [\Phi_j^F])}{(1 + a_j)}, \quad (6)$$

in which  $F$  and  $m$ , respectively, represent the objective function and the number of the total measured orders and  $a_j$  is a penalty function to account for differences between the test and the FE model frequencies:

$$a_j = \left| \frac{(f_j^F)^2 - (f_j^T)^2}{(f_j^F)^2 + (f_j^T)^2} \right|. \quad (7)$$

$f_j^F$  and  $f_j^T$  are the  $j$ th frequencies of the FE model and the test; the factor  $a_j$  accelerates the convergence of the objective function as well.

Theoretically, when the global minimum of this implicit objective function equals to zero, the input parameters of the FE model equally represent the physical parameters of the actual structure.

## 3. Algorithms for Damage Detection

**3.1. Standard ABC Algorithm.** ABC algorithm is proposed by Karaboga [20], and the colony of artificial bees is categorized into three main groups: employed bees, onlooker bees, and scout bees. The number of the employed bees and the onlooker bees is the same as the number of the food sources. In some cases, an employed bee should be transformed into a scout bee. Each bee can only exploit one food source each time. The employed bees first explore the food sources according to their memories and explore possible food sources in corresponding neighborhoods, and then they fly back to the hive and share the information of the food sources by dancing. After digesting the information, the onlooker bees select the food sources one by one and become employed bees (the food sources with more nectar have a higher probability of being chosen). When a food source is exploited several times and no other better food sources are found in the neighborhood, the employed bee of this food source transforms into a scout bee. The scout bee randomly selects a new food source and continues its work as an employed bee. The main parts of ABC algorithm are as follows [32]:

**Initialization:** the location of an initial food source, or a candidate solution, is a multidimensional parameter vector. The parameters of this vector are generated by the following equation:

$$x_{ij} = x_j^{\min} + \text{rand}(0, 1) \times (x_j^{\max} - x_j^{\min}), \quad (8)$$

where  $i$  and  $j$  are individually selected from  $[1, \text{SN}]$  and  $[1, D]$ ; SN is the number of the food sources;  $D$  is the number of the dimensions;  $x_{ij}$  is the  $j$ th dimension of the  $i$ th food source;  $x_j^{\min}$  and  $x_j^{\max}$  are, respectively, the lower and upper bound of the  $j$ th dimension; and  $\text{rand}(0, 1)$  is a random real number in the range  $[0, 1]$ .

Employed bee phase: first, each employed bee is associated with a food source ( $x_i$ ), and then a new food source ( $v_i$ ) is produced. The parameters of the new food source are as follows:

$$v_{ij} = x_{ij} + \text{rand}(-1, 1) \times (x_{ij} - x_{kj}), \quad (9)$$

where  $x_{kj}$  represents the  $j$ th dimension of another food source ( $x_k$ ) selected randomly from the remaining population. The employed bee compares the results of the two food sources (i.e.,  $x_i$  and  $v_i$ ) and brings back the information of the better one; for the optimization problem of finding the global minimum, the better means a smaller objective value (the greedy selection strategy).

Onlooker bee phase: before the onlooker bees start working, the fitness values of the food sources brought back by the employed bees are calculated as follows:

$$\text{fit}_i = \begin{cases} \frac{1}{1 + f_i}, & f_i \geq 0, \\ 1 + \text{abs}(f_i), & f_i < 0, \end{cases} \quad (10)$$

where  $\text{fit}_i$  and  $f_i$  are, respectively, the fitness value and the objective value of the  $i$ th food source. The selection probability of the  $i$ th food source equals to  $p_i$  (roulette wheel selection):

$$p_i = \frac{\text{fit}_i}{\sum_{m=1}^{\text{SN}} \text{fit}_m}, \quad (11)$$

After the selection, the onlooker bees fly to the selected food sources one by one and generate corresponding new food sources in a similar way as in equation (9). Again, each generated food source is compared with the corresponding previous food source via the greedy selection strategy.

The updating process of the employed bees and the onlooker bees is the same, except that the employed bees update all food sources while the onlooker bees only update the selected food sources.

Scout bee phase: a food source may be visited several times without a better food source founded nearby. This kind of food source is abandoned after a pre-defined number of visits. Then, the corresponding employed bee transforms into a scout bee to randomly generate a food source via equation (8) and continues its work as an employed bee. The schematic of the ABC algorithm is shown in Figure 1.

**3.2. TCABC Algorithm.** In this paper, two searching methods, the tabu search method [29] and chaotic search method [30], are adopted to enhance the exploration and exploitation ability of the traditional ABC algorithm (named as TCABC algorithm). Additionally, the original roulette wheel selection is replaced by the tournament selection to enhance the global searching ability of TCABC algorithm.

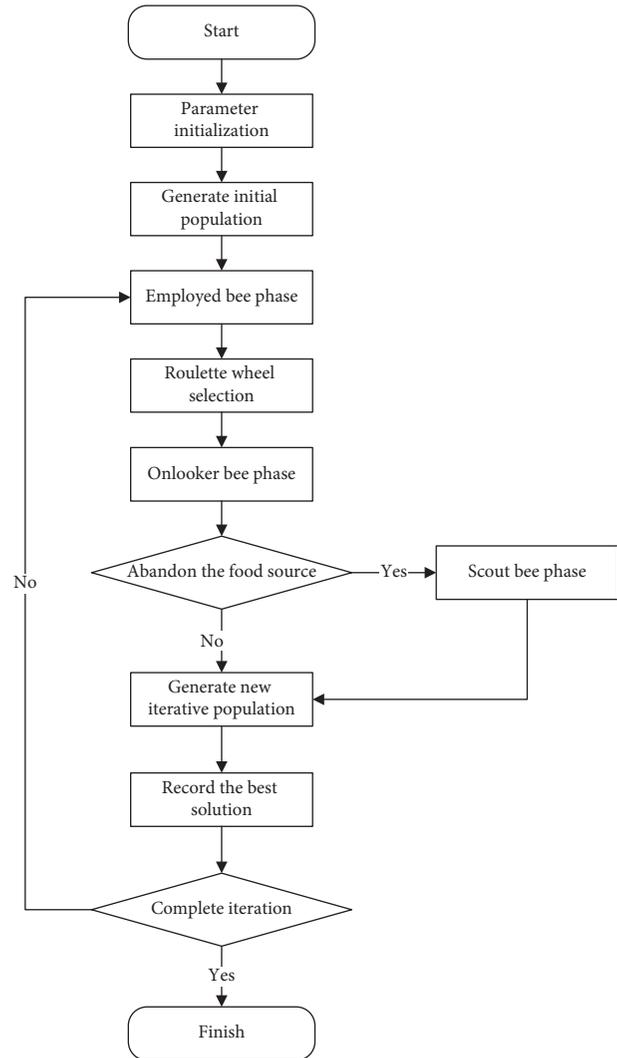


FIGURE 1: Schematic of the ABC algorithm.

**3.2.1. Tabu Search Method.** The tabu search, a simulation of the human intelligence process, is first proposed by Glover [29] as an extension of local neighborhood search. As a global stepwise optimization algorithm, it effectively avoids the solution falling into the local optima and prevents the algorithm from entering an infinite loop. The so-called tabu is to prevent exploiting the previous visited and bad food sources. By using a tabu list to record these visited food sources, the repetition of the previous work can be prohibited, and the total exploitation ability is improved. The tabu search method avoids roundabout search by introducing a flexible storage structure and corresponding tabu criteria. Simultaneously, some prohibited good food sources are “unlocked” by the aspiration criterion.

In this paper, the tabu search is applied throughout the whole process of the TCABC algorithm, and two tabu lists (i.e.,  $T1$  and  $T2$ ) are introduced into the algorithm.  $T1$  is a tabu list with a limited length, which stores the recently visited food sources. By applying the  $T1$  list, other bees cannot search recently visited food sources. In detail, the

length of  $T1$  is assumed as  $n$ . When the  $n + 1$  food source is recorded in  $T1$ , the first food source in  $T1$  is taken out.  $T2$  saves the deserted food sources (i.e., the food sources, where the scout bees are generated); those food sources are exploited several times, and no improvement is found, which means their qualities are not good. Unlike  $T1$ , which only ban finite food sources within a limited time, those abandoned food sources in  $T2$  are banned eternally and the length of the  $T2$  list is infinite.

**3.2.2. The Tournament Selection Strategy.** In the standard ABC algorithm, the roulette wheel selection is used to decide the probability of each food source selected by the onlooker bees. However, some food sources may have immense fitness values. According to the roulette wheel selection in equation (11), many onlooker bees may fly to these food sources, which reduces the global searching ability of the algorithm. To avoid this flaw, the tournament selection is employed to replace the original roulette wheel selection in TCABC algorithm.

In the tournament selection strategy, the fitness values of every two food sources are compared. The one with a bigger fitness value gains 1 point and the other gains 0 point, and each food source is also compared to itself (to guarantee the worst food source gains 1 point). After the comparison, each food source ( $x_i$ ) has a corresponding score ( $s_i$ ), and the possibility of food source  $x_i$  being selected equals

$$p_i = \frac{s_i}{\sum_{m=1}^{SN} s_m}, \quad (12)$$

where SN means the number of food sources. This strategy guarantees that those food sources with significant fitness values do not overly affect the selection tendency. Therefore, the onlooker bees will not fly to the same food source in droves.

**3.2.3. Chaotic Search Method.** Mathematically, logistic mapping is a straightforward chaotic mapping with several characteristics, such as uncertainty, ergodicity, and regularity. The logistic mapping is expressed as follows [30]:

$$\beta_{n+1} = \mu\beta_n(1 - \beta_n). \quad (13)$$

Equation (13) is chaotic when  $\mu = 4$  and  $0 \leq \beta_0 \leq 1$ . Figure 2 displays the calculation results when  $\mu = 4$  and  $\beta_0 = 0.27$ . As can be seen, the initial information is lost after several times of iterations, and the output is like a stochastic output.

In this paper, a chaotic search method is applied during the onlooker bees' phase and the scout bees' phase. The operation steps are as follows: (1) capturing the new food source generated by the onlooker bee (or capture the abandoned food source); (2) mapping this food source into a uniform space; (3) applying the chaotic search method to the uniform space-based food source several times, record all results; (4) transferring all these results back to the normal

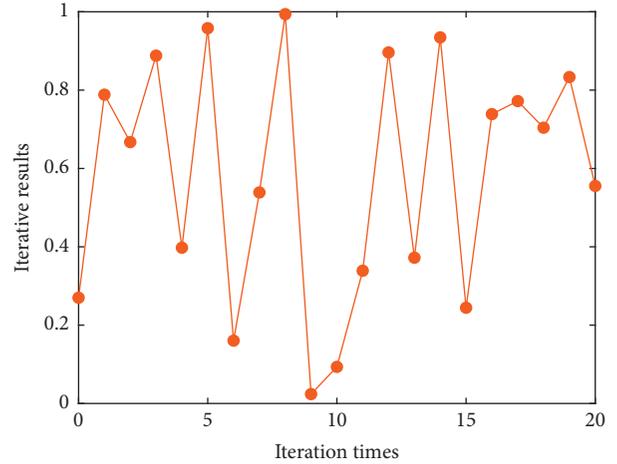


FIGURE 2: The output of equation (13) when  $\mu = 4$  and  $\beta_0 = 0.27$ .

space; (5) calculating the fitness values of all these food sources and bringing back the information of the best one.

Compared to the standard ABC algorithm, the chaotic search method allows each bee fly to several food sources simultaneously; by doing so, more food sources are visited at the same time and the population diversity of the algorithm is improved. OpenSees platform [33], a widely used software in civil engineering, is also used to establish the numerical model of the studied structure in this paper. The schematic of the TCABC algorithm is shown in Figure 3.

## 4. Numerical Study

**4.1. Description of the Benchmark Beam.** With MATLAB [34] and Opensees [33], the algorithm's main procedure and corresponding FE model are established, and the accuracy of the TCABC algorithm in structural damage detection is validated and verified.

In this paper, a twenty-four-meter steel beam with two fixed ends shown in Figure 4 is modeled as the benchmark case. The density and the elastic modulus of steel are set as  $7850 \text{ kg/m}^3$  and  $206 \text{ GPa}$ , respectively. This beam is discretized into a twenty-four-element structure; every two connected elements are assigned as a group and share the same elastic modulus.

### 4.2. Comparison of Standard ABC and TCABC Algorithms

**4.2.1. Comparison by Explicit Test Functions.** Before directly applying the TCABC algorithm to the damage detection problem, several explicit test functions are used to compare the performance between ABC and TCABC algorithms. By comparing the calculation results, differences between the two algorithms can be obviously observed, and the generality and accuracy of the TCABC algorithm can also be verified. The used explicit test functions are listed in Table 1.

To purely compare the differences between ABC and TCABC algorithms and to simplify the calculation

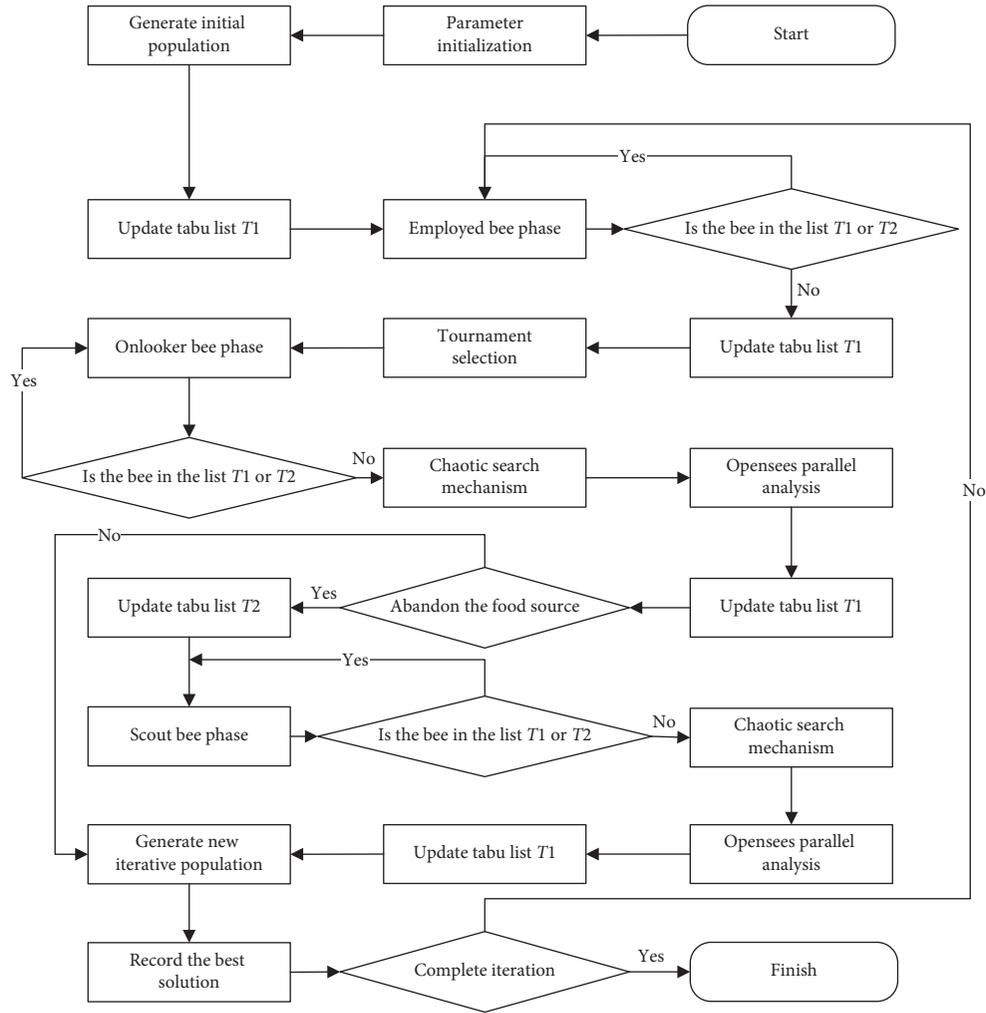


FIGURE 3: Schematic of the TCABC algorithm.

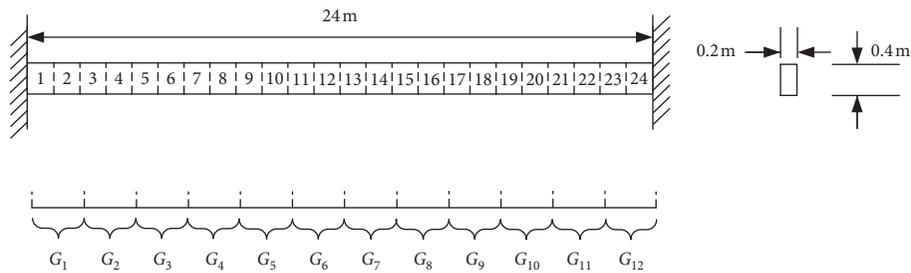


FIGURE 4: The 24 m benchmark beam.

TABLE 1: Used explicit test functions.

No.	Test functions	Searching range	Global minimum
F1	$f_1(x) = \sum_{i=1}^D x_i^2$	$(-10, 10)^D$	0
F2	$f_2(x) = \sum_{i=1}^D (10^6)^{(i-1)/(D-1)} x_i^2$	$(-10, 10)^D$	0
F3	$f_3(x) = \sum_{i=1}^D i x_i^2$	$(-10, 10)^D$	0
F4	$f_4(x) = \sum_{i=1}^D  x_i  + \prod_{i=1}^D  x_i $	$(-10, 10)^D$	0
F5	$f_5(x) = (1/4000) \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos(x_i/\sqrt{i}) + 1$	$(-10, 10)^D$	0
F6	$f_6(x) = \sum_{i=1}^D i x_i^4$	$(-10, 10)^D$	0

procedure, corresponding input parameters of the two algorithms in the six test functions are set as the same. The calculation generations are set as 500 and each generation contains 100 food sources; the abandoned limitation of the food sources is set as 1800, which is the same as  $0.6 \times SN \times D$  in [21]. Besides, the dimensions ( $D$ ) of all six test functions are set to 30. After trial calculations (which is introduced later), two more parameters (i.e., the length of T1 and the times for chaotic search) of the TCABC algorithm are separately set to 100 and 30. Each test function is calculated 20 times, and the corresponding results are displayed in Table 2. It can be observed that except for the worst solution and the standard deviation of F5, the TCABC algorithm performs better than the ABC algorithm.

To study how the generation number, the length of T1, and the times of chaotic search influence the performance of TCABC algorithm, a single variable method is applied and corresponding results are compared. The benchmark parameters and the calculation times are set the same as previously defined. Different generation numbers are analyzed (i.e., conduct the analysis every 50 from 200 to 700 generations), and the results are shown in Table 3 and Figure 5. The growth rate in Figure 5 is defined as follows:

$$\text{Growth rate} = \frac{V_n - V_p}{V_p}, \quad (14)$$

where  $V_n$  and  $V_p$ , respectively, represent the value of present and previous studied generations.

In Figure 5, the growth rates of both mean and standard deviation are always negative, which means by increasing the generations, the results are closer to the global minimum.

Similar to the generation number, different chaotic search times and different lengths of T1 are studied; Figures 6 and 7 display the relevant mean results of the target functions. All results are normalized by the identified value when the times of chaotic search (or the length of T1) equals zero (i.e., the dashed line). As can be observed, except for 2 results in Figure 7, all other results are below the dashed line, which means by introducing the tabu search method and chaotic search method, the algorithm performs better. However, the two parameters show differently to the result of each target function; trial analyses are recommended to find appropriate T1 and chaotic search times when using this method.

**4.2.2. Comparison by the Implicit Objective Function.** The comparison between ABC and TCABC algorithms for the implicit damage detection function (i.e., equation (6)) is studied in this section. Similar as in Section 4.2.1, the input parameters of the two algorithms are set as the same, and after trial analyses, the used parameters are listed in Table 4. It is noteworthy that to damage detection problems, the recognition accuracy is not required as high as in previously studied test functions. The reason is that due to the limitation of the existed experimental devices, those errors less than one percent are unrecognizable. So, the “extremely precise” to the identified parameters are

meaningless (e.g., for the benchmark beam, the elastic modulus of group one that equals from  $1.0003 E_0$  to  $1.0007 E_0$  may result in very close vibration properties, and these values cannot be distinguished under test conditions due to the instrument accuracy). Nevertheless, evolution-based algorithms intend to achieve the global minimum and find input parameters as accurately as possible. To achieve a balance between them, the identified parameters are rounded to five decimal places in the damage detection problem.

For the benchmark case in Section 4.1, the first ten orders of modal vectors and frequencies are used for the objective function (i.e., equation (6)); the modal vectors and corresponding MACs are captured by all nodal displacements (i.e., 23 nodes here). The assumed damage level of this beam is listed in Table 5, and the meaning of damage index is described in equation (3). Negative values here for the assumed damages are meaningful because for actual tests, before detecting the damages, the structure parameters should be modified first, and the range of the modification is beyond the boundary (i.e., 0 and 1) of just considering the damage. To verify the applicability of the proposed algorithm under the implicit damage detection function, the searching space is changed, and the negative damage indexes are used. By combining the damage indexes (Table 5), the lower bound and the upper bound of the indexes' searching space are set as  $-0.5$  and  $0.5$ , respectively.

The convergence performance of the implicit objective function is shown in Figure 8(a). In general, the results of the TCABC algorithm are much closer to zero compared to the results of the standard ABC algorithm, which means the TCABC algorithm has a better performance. Figure 8(b) displays the identified parameters; the  $y$ -axis represents the corresponding damage index  $a_i$ . The results indicate that to damage detection problems here, both algorithms work well, but the TCABC algorithm is more accurate.

**4.3. Consideration of the Influences of Different Grouping Techniques.** For actual damage detection problems, how to set the benchmark groups may influence the accuracy of the identification results. This section investigates the precision of the TCABC algorithm through two cases with different grouping techniques.

**4.3.1. The Influences of Different Grouping Sizes.** Grouping sizes may alter even in the same damage situation. For example, as shown in Figure 9, the elastic modulus of element 9 to 12 are reduced to 0.65 of the undamaged ones (the benchmark beam in Section 4.1 is used here, and  $E_0$  is the elastic modulus of the intact beam), and three types of grouping are used to detect the damage in this section; the beam is, respectively, divided into 6, 12, and 24 groups in Figures 9(a)–9(c). Same as in Section 4.1, the elements in each group share the same elastic modulus; the input parameters and the searching space are set the same as in Section 4.2.2.

TABLE 2: Performance comparison between ABC and TCABC algorithms.

No.	Best solution		Worst solution		Mean		Standard deviation	
	ABC	TCABC	ABC	TCABC	ABC	TCABC	ABC	TCABC
F1	0.017068	<b>0.005334</b>	0.037956	<b>0.024644</b>	0.024267	<b>0.013655</b>	0.007486	<b>0.002975</b>
F2	3.554567	<b>1.468162</b>	7.301145	<b>3.746693</b>	5.35974	<b>2.534457</b>	1.228749	<b>0.774593</b>
F3	0.283409	<b>0.124682</b>	0.382502	<b>0.201409</b>	0.325852	<b>0.147296</b>	0.030473	<b>0.022702</b>
F4	0	0	0	0	0	0	0	0
F5	0.006508	<b>0.005878</b>	<b>0.067564</b>	0.136364	0.042717	<b>0.035809</b>	<b>0.019248</b>	0.044153
F6	8.134684	<b>2.758961</b>	40.50607	<b>12.24841</b>	18.82364	<b>6.891574</b>	10.49996	<b>2.536104</b>

TABLE 3: Performance of TCABC with different generation numbers.

Calculation generation	Results based on 20 calculations	F1	F2	F3	F4	F5	F6
200	Mean	3.659166	844.5289	36.46164	0.087539	0.511423	875.2596
	Standard deviation	0.520355	152.3502	7.336005	0.001679	0.088312	367.7924
700	Mean	0.000443	0.077873	0.004132	0	0.001442	0.344597
	Standard deviation	0.000093	0.014836	0.001234	0	0.001476	0.174365

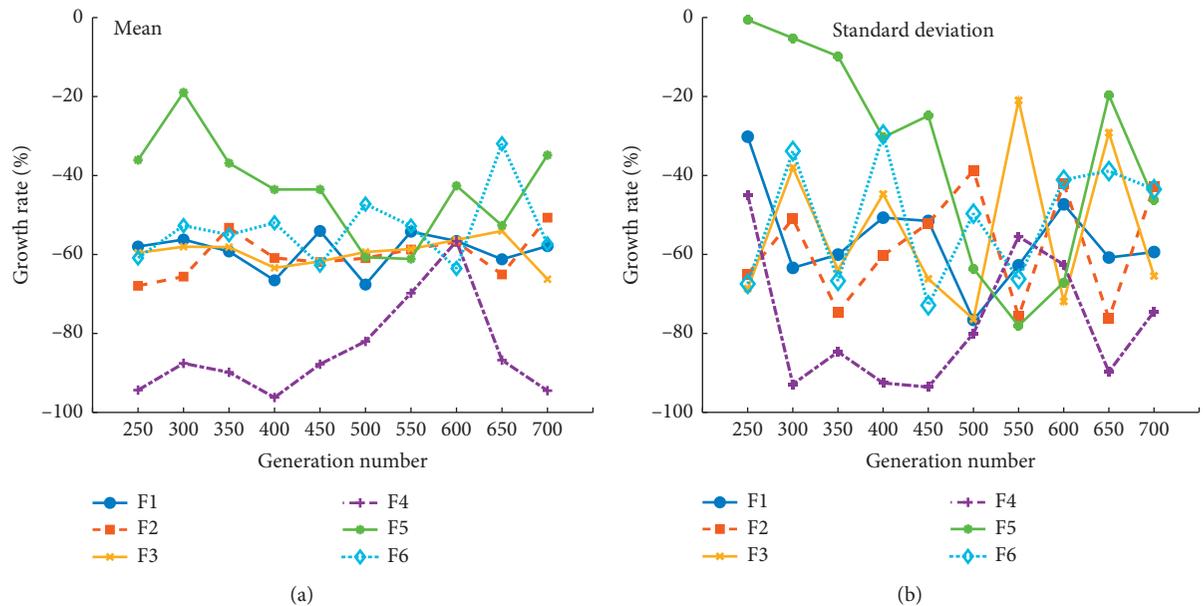


FIGURE 5: The growth rate of the TCABC algorithm with different generations for (a) mean and (b) standard deviation.

The numerical results are shown in Figure 10. Figure 10(a) indicates that (1) in the same damage situation, both ABC and TCABC algorithms converge faster with fewer groupings; (2) it is demonstrated again that the TCABC algorithm performs better than the ABC algorithm for the implicit damage detection function. Because the focus here is the influences of the grouping sizes, Figure 10(b) only displays the results of the TCABC algorithm under different scales of grouping; it can be observed that even though there are some noises, the damages are detected by all three dimensions of grouping.

**4.3.2. The Influences of Inappropriate Groupings.** In previous studies, the assumed damage groups are set exactly the same

as the actual damage groups. However, it is complicated to locate the damage position in actual tests precisely. In this section, three scenarios are studied to demonstrate whether the TCABC algorithm is able to detect structural damages in inappropriate groupings. As shown in Figure 11, the benchmark beam is divided into 12 groups. Figures 11(a)–11(c) represent damage scenario 1 to 3, individually.

In Figure 11(a), the stiffness damage index of element 9 is 0.35 and the position of this damaged element is in group 5 ( $G_5$ ). This sort of damage is named as “in-group” damage. Scenario 1 is used to check the detection results of the TCABC algorithm in this “in-group” damage situation. In Figure 11(b), both elements 8 and 9 are damaged, and they are exactly across the boundary of  $G_4$  and  $G_5$ . Damage like this is named as “cross-group” damage. The performance of

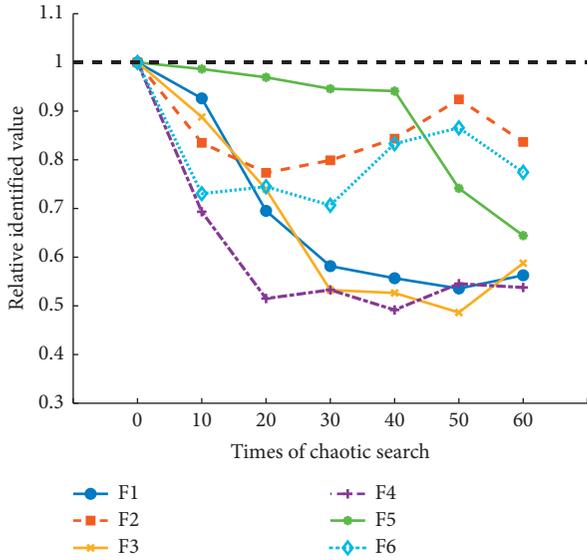


FIGURE 6: Influences of the times of chaotic search for TCABC algorithm.

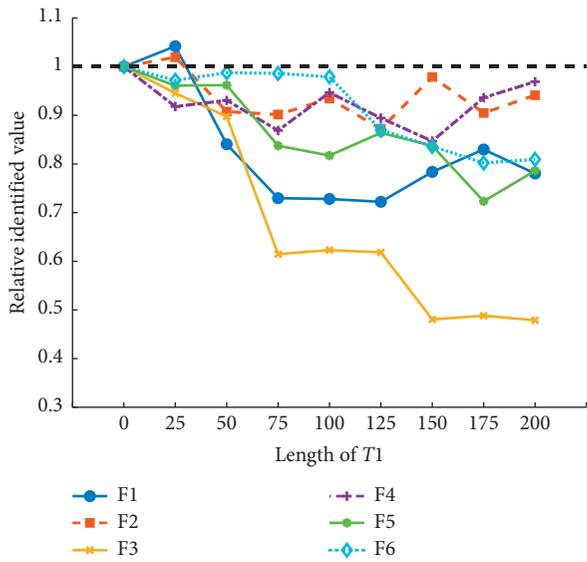


FIGURE 7: Influences of the length of  $T_1$  for TCABC algorithm.

TABLE 4: Input parameters of the damage detection.

Input parameters	ABC	TCABC
Calculation generations	200	200
Number of food sources	60	60
Abandoned limitations	360	360
Length of $T_1$	—	100
Times of chaotic search	—	15

the TCABC algorithm in this “cross-group” damage situation is studied in scenario 2. Multidamage condition (i.e., shown in Figure 11(c)), which combines “in-group” damage and “cross-group” damage, is reviewed in scenario 3.

After trial calculations and setting reasonable input parameters, the detection results of scenario 1 to 3 are separately shown in Figures 12(a)–12(c). Due to the inappropriate groupings, the accuracy of the proposed algorithm cannot be guaranteed. However, by checking the detection results, the detected damaged positions (enclosed in those black dash boxed) contain or near the real damage positions, which means even though the groupings are inaccurate, TCABC algorithm still contributes to locating the damage under all three circumstances as well as under actual test conditions. Compared to Figures 8(b) and 10(b), the damage detection results in Figure 12 have a lower damage index. The reason is that due to the inappropriate groupings, the theoretical global minimum (i.e., zero) of the objective function cannot be achieved. So, a local minimum is found instead. The local minimum is obtained by adjusting the damage indexes of all elements to let the value of equation (6) be as small as possible, which causes the damage indexes of the undamaged and damaged elements, respectively, increased and reduced.

4.4. Consideration of the Influences of the Field Test Data. Besides grouping size, the accuracy of the damage detection results is affected by the field test data as well. It is easy to obtain all the vibration properties by numerical analyses. However, there are several obstacles to gain modal and frequencies data in the field test. Firstly, the values of nodal displacements and MACs are highly influenced by the sensors, and usually, only limited numbers of sensors are used; secondly, unlike numerical analyses, only first few orders of frequencies and modal vectors can be obtained in the field test, which controls the  $m$  value in equation (6); thirdly, environmental, electromagnetic, and instrumental noises are everywhere during the measurement and the noise level is highly uncertain.

Subjectively, the more data are measured, the more accurate the results will be; however, this is not confident when these noises are introduced, by increasing the amount of measured data, the amount of obtained noises also increases. So, a quantitative analysis is needed. This section focuses on how the measured data influence the detection results.

4.4.1. Measuring Noises. In an experimental study, measured data are all contaminated by noises. Therefore, when different kinds of measured data are used to account for the influences on the damage detection results, the artificial noises should simultaneously be included. The random noises are applied to the theoretical frequencies and modal vectors by equations (15) and (16), separately:

$$f_i^N = f_i(1 + \eta_1 \varphi_i^N), \quad (15)$$

where  $f_i^N$  and  $f_i$  separately represent the noise-contaminated and noise-free  $i$ th frequency;  $\eta_1$  is a random real value in the interval  $[-1, 1]$ ; and  $\varphi_i^N$  represents the noise level of  $i$ th frequency.

TABLE 5: Assumed damage level of the elastic modulus of the benchmark beam.

Group number	G1 (%)	G2 (%)	G3 (%)	G4 (%)	G5 (%)	G6 (%)	G7 (%)	G8 (%)	G9 (%)	G10 (%)	G11 (%)	G12 (%)
Damage index	-30	0	10	-14	-42	0	25	-26	30	0	0	34

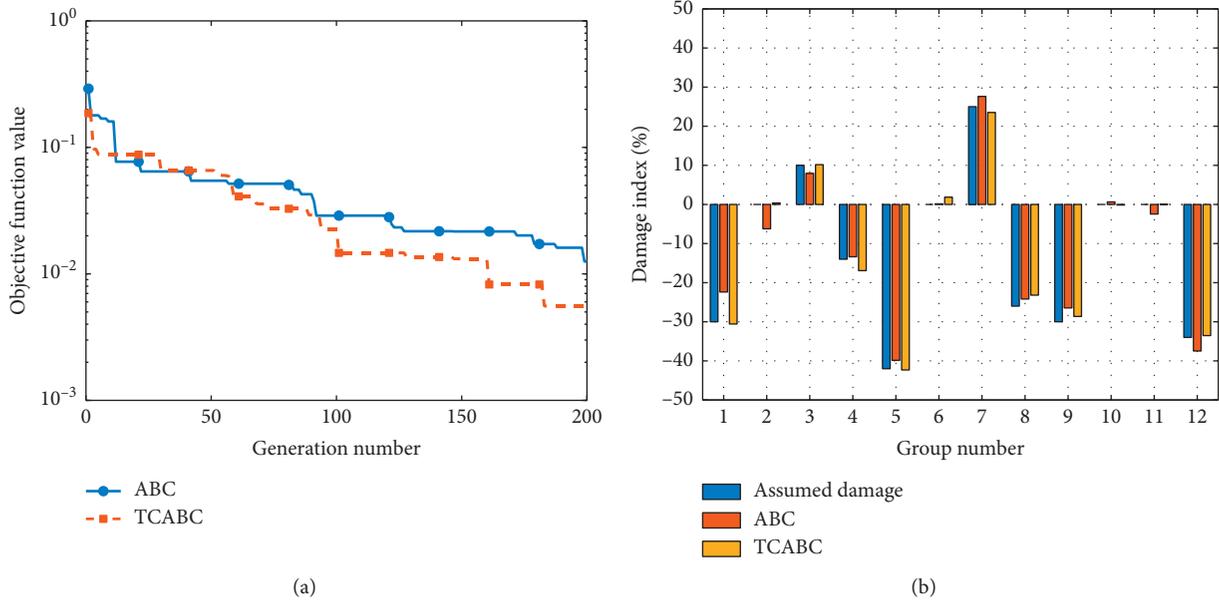


FIGURE 8: (a) The convergence performances of the objective function. (b) The identified parameters of the damage detection.

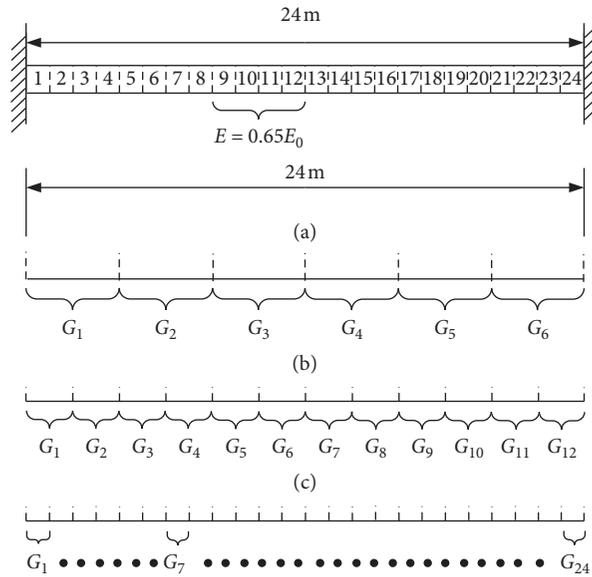


FIGURE 9: The same damage situation with different sizes of grouping.

$$\Phi_{ij}^N = \Phi_{ij}(1 + \eta_2 \xi_{ij}^N), \quad (16)$$

where  $\Phi_{ij}^N$  and  $\Phi_{ij}$  separately represent the modal displacement of the noise-contaminated and noise-free  $j$ th position of  $i$ th modal vector;  $\eta_2$  is a different random real value in the same interval as  $\eta_1$ ; and  $\xi_{ij}^N$  represents the corresponding noise level.

During the field test, the noise level may be influenced by a lot of factors (e.g., test condition, the precision of the instrument, and testers' skill). To verify the robustness of the proposed algorithm, three levels of noise listed in Table 6 are studied; because the frequency measurements are more stable, its noise levels are always smaller than those obtained by modal displacements.

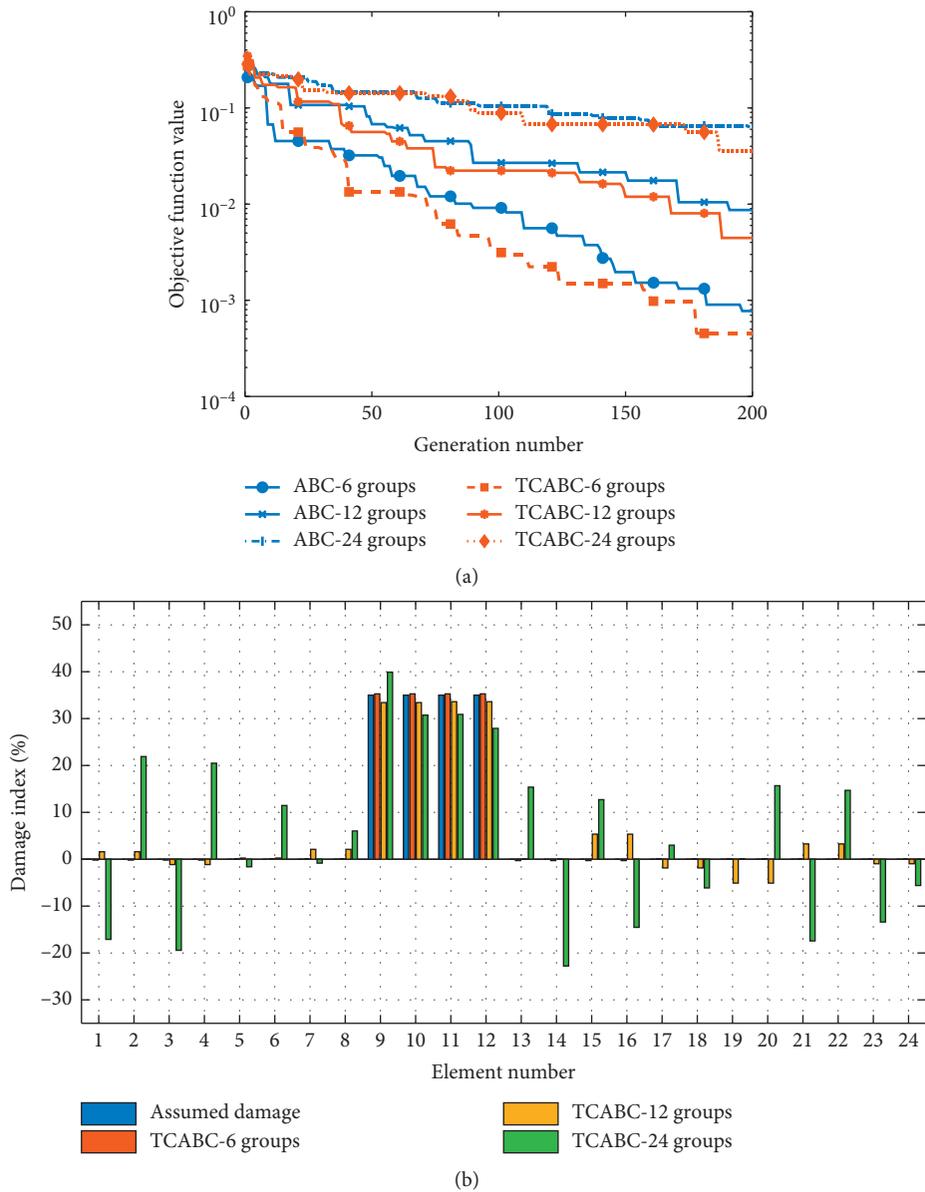


FIGURE 10: (a) The convergence performance under different sizes of grouping. (b) The identified parameters under different sizes of grouping.

4.4.2. *The Total Orders of the Measured Modal Shapes and Frequencies.* The total orders of the measured frequencies and modal shapes influence the format of the objective function, and the accuracy of the proposed algorithm under different objective functions may be different. In order to compare the effects of different numbers of measured orders, the first three, six, and ten orders of modal shapes and frequencies are used separately to check the damage detection results of the benchmark case in Section 4.2.2 (i.e., damage described in Table 5); the noises mentioned in Section 4.4.1 are simultaneously added. It is noteworthy that the modal displacements here are captured by all existed nodes (i.e., 25 nodes in this case), and the influences of the number of the measured nodes are introduced later. Figures 13(a)–13(c) show the damage detection results of

only using the first three, six, and ten orders of modal shapes and frequencies.

It can be found that (1) when there is no noise, the detection results are not influenced by the number of the measured orders. This can be seen from the similar height of the assumed damage bars and the non-noise bars in all three cases, as shown in Figure 13; (2) when noises are included, the damage detection results become better as the orders of modal vectors and frequencies increase. This can be obtained by comparing the LV3-noise bars of Figures 13(a) and 13(c). In Figure 13(a), the LV3-noise bars are far different from the assumed-damage bars to some groups, such as group 2, 3, 10, and 11; but, in Figure 13(c), their differences are not that much; (3) when noises are included, the stronger the noises are, the worse the identification results become. This

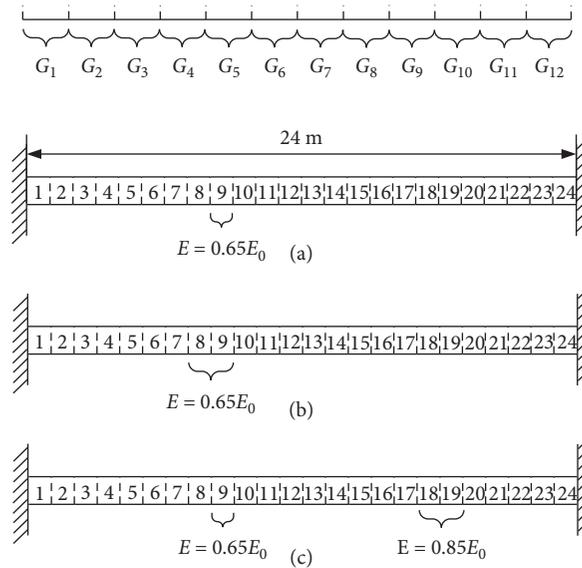


FIGURE 11: The inappropriate groupings.

phenomenon is easily observed by checking the LV1-noise bars (which represent the lowest noise level) and the LV3-noise bars (which serve the highest noise level). In general, the damage indexes of the LV1-noise bars are closer to the assumed-damage bars than the LV3-noise bars.

**4.4.3. The Number of the Measured Nodes.** Unlike in numerical analyses whose information can be abstracted out quickly, the number of sensors for field tests is usually limited. In this section, twenty five, thirteen, seven, and five sensors are assumed to be used in the field test for the benchmark case in Section 4.2.2 (i.e., damage described in Table 5). These sensors are applied at the nodes of the benchmark beam, as displayed in Figure 14. Figures 11(a)–11(d) correspond to the cases with twenty five, thirteen, seven, and five sensors, individually. Since this section focuses on the influences of the number of the measured modal displacements, other parameters are set the same as in Section 4.2.2, including that the first ten orders of modal shapes and frequencies are used. Figures 15(a)–15(c), respectively, represent the numerical results of Figures 14(c)–14(d). The scenario given in Figure 14(a) is not displayed again because it is the same as that in Figure 13(c).

The numerical results show (1) the more the points are measured, the more accurate the damage detection results will be; combining this with conclusion (2) in Section 4.4.2, it is concluded that regardless of noises, the results will be more accurate when more data are measured; (2) it also verifies conclusion (3) in Section 4.4.2 that stronger noises result in worse identification results. Thus, in the field test, the measurement error should be reduced as much as possible to guarantee detection accuracy.

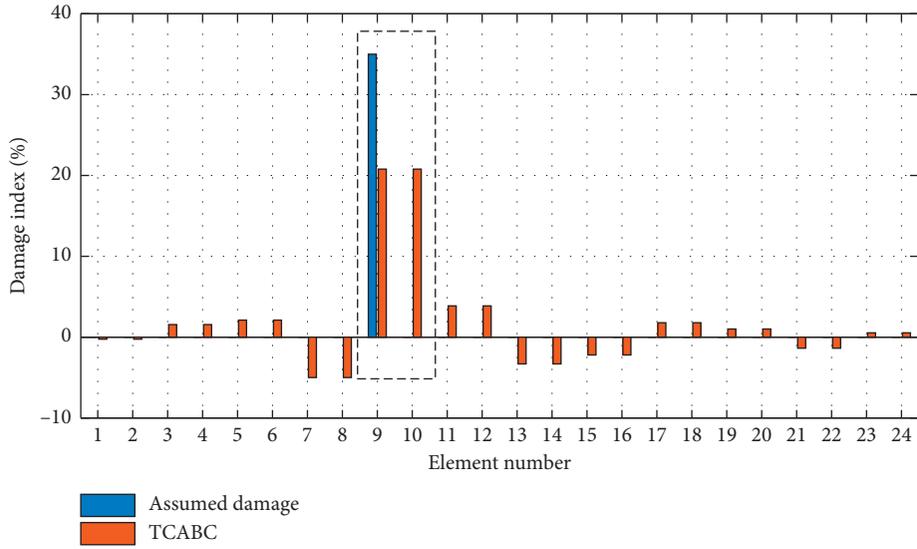
It is noteworthy that, in Section 4.4.2, when noises are not included, the accuracy of the damage detection results is not influenced by the measured orders; however, in this section, no matter whether the noises are included or not,

as more points are measured, the results will be better. This indicates that the damage detection problem with the proposed objective function is more sensitive to the number of the measured points than the number of the determined orders. Therefore, ensuring sufficient sensors has a higher priority to guarantee the accuracy of the damage detection.

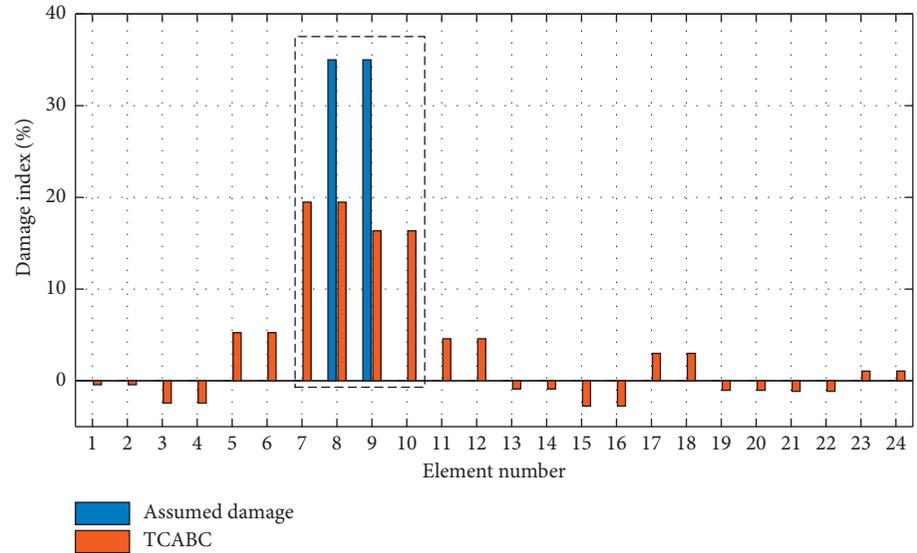
## 5. Experimental Study

All the case studies in Section 4 are numerically based. Though artificial noises are applied in them, these noises cannot correctly represent the modeling errors between the real structure and FE model. Thus, an experimental study is necessary for checking the accuracy and validity of TCABC algorithm. As an example, the experimental study conducted by Hu et al. [15] is used to verify the accuracy of TCABC algorithm. The experimental beam is shown in Figure 16. The used material constants are  $E = 70$  Gpa,  $\nu = 0.3$ , and  $\rho = 2700$  kg/m<sup>3</sup>. As shown in Figure 16, saw-cut damage at the middle of the ninth element is applied and the depth of the crack is one-quarter of the total height of the beam. According to the section properties and the assumption of Moslem and Nafaspour [13], the elastic modulus of element 9 is between  $0.125E_0$  and  $1E_0$ , where  $E_0$  is the undamaged elastic modulus of the beam. This experimental study considered the errors of inappropriate groupings, noises, and lack of measured data (only the first three orders of vibration data are obtained), which can influence the detection results.

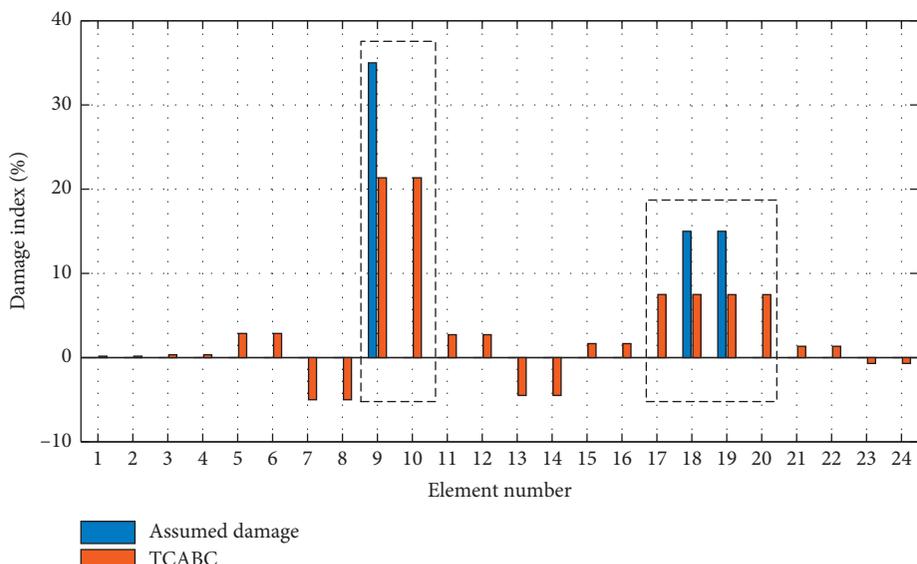
Hu et al. [15] abstracted the first three orders of vibration data of both the damaged and the undamaged models and used two identification algorithms to detect the damage; in the first algorithm, the global analytical stiffness and mass matrices are not needed, and the algorithm is cited as DDNKM; for the second one, only the analytical mass matrix is employed, which is named as DDNK. Perera and



(a)



(b)



(c)

FIGURE 12: (a) The identified parameters of scenario 1. (b) The identified parameters of scenario 2. (c) The identified parameters of scenario 3.

TABLE 6: Different noise levels.

Noise level	$\varphi_i^N$ (%)	$\xi_{ij}^N$ (%)
LV1	3	8
LV2	5	15
LV3	8	20

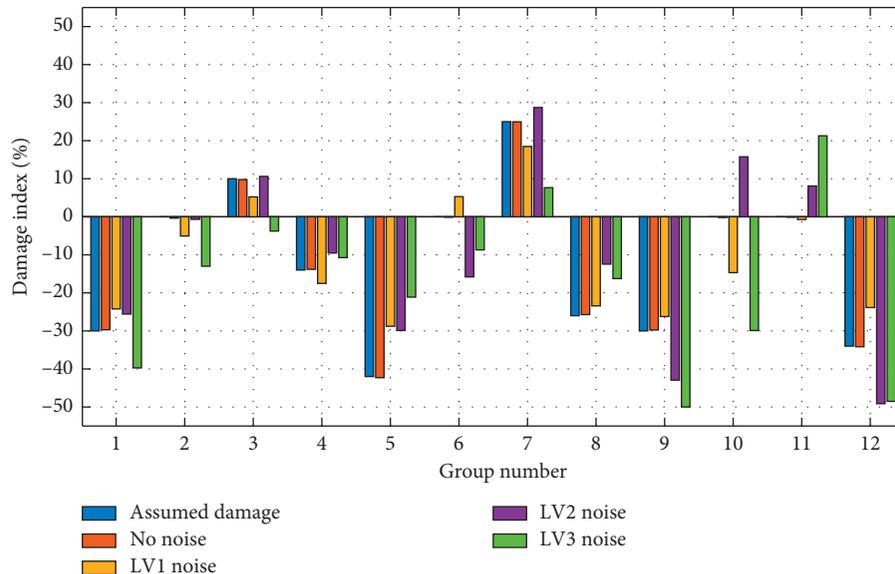
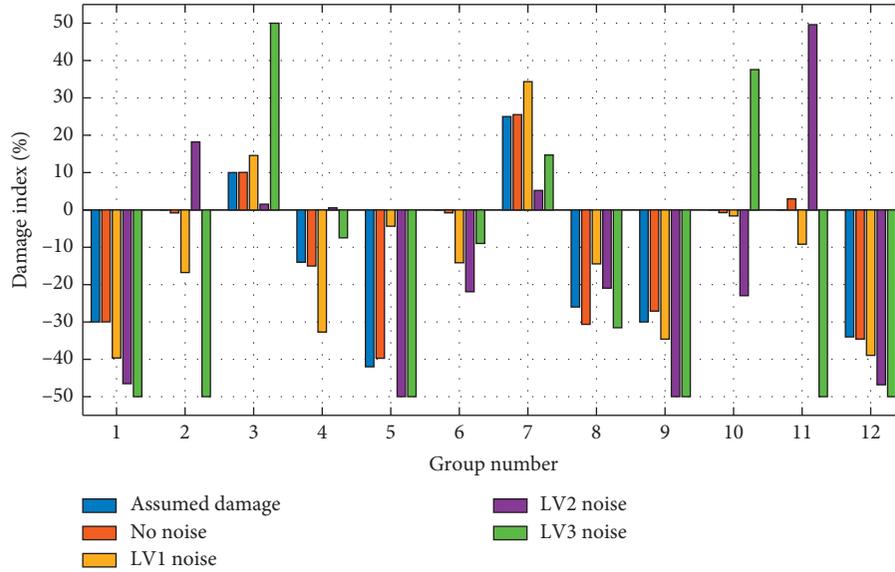


FIGURE 13: Continued.

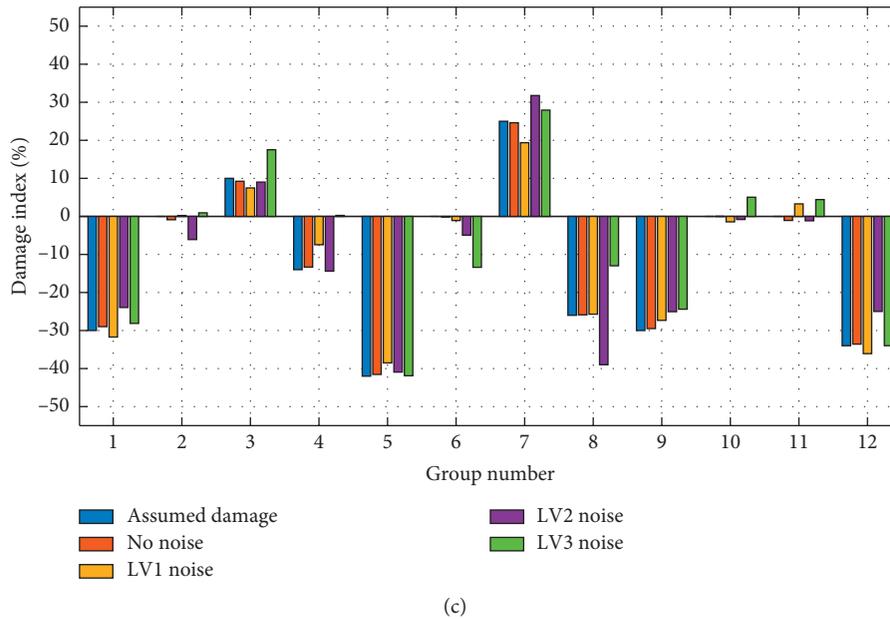


FIGURE 13: (a) The identified parameters with the first 3 orders of modal shapes and frequencies. (b) The identified parameters with the first 6 orders of modal shapes and frequencies (c) The identified parameters with the first 10 orders of modal shapes and frequencies.

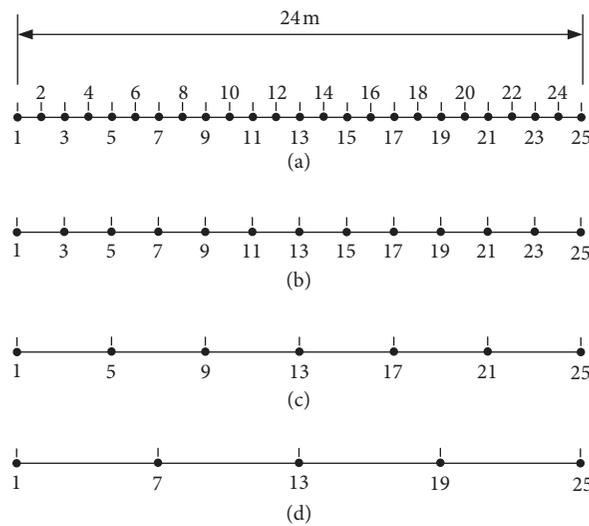
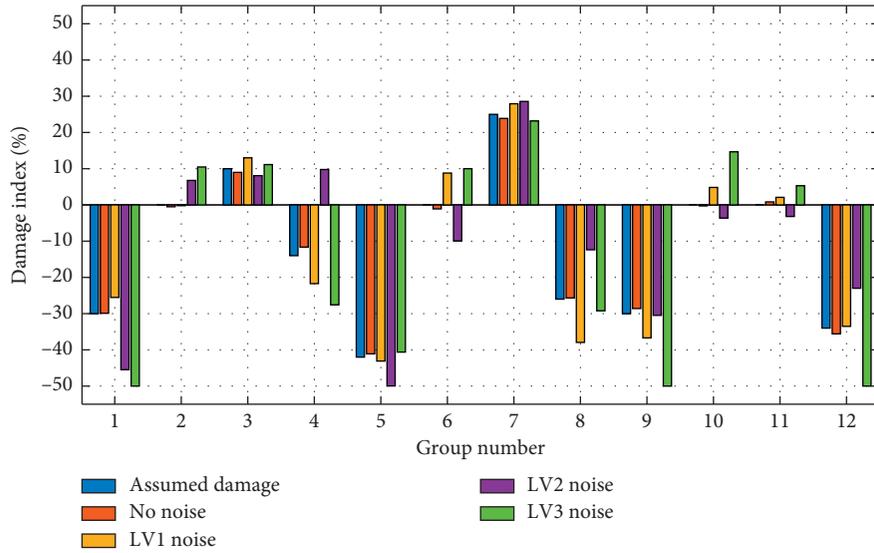


FIGURE 14: The applied sensors and corresponding positions.

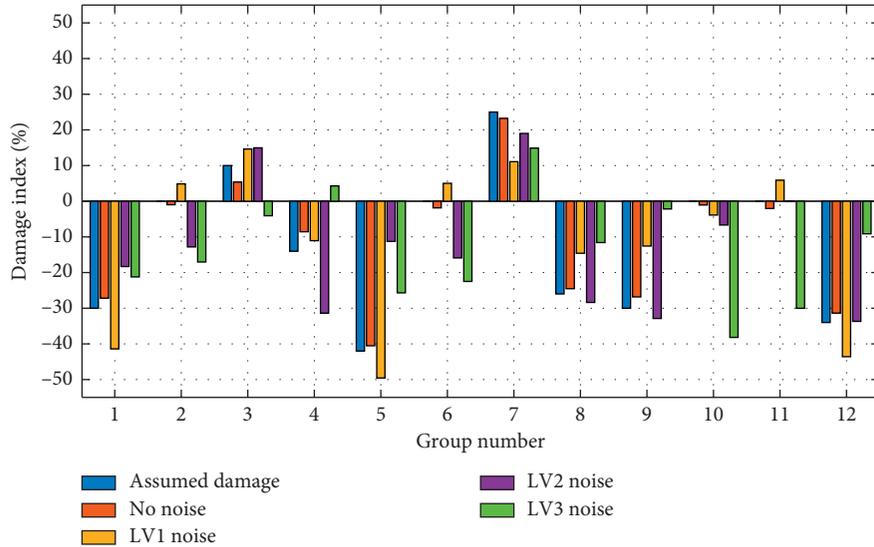
Torres [14] used the same case but GA to identify the damage; GA shows a significant improvement compared with DDNK. However, both Hu et al. [15] and Perera and Torres [14] did not establish a baseline model before the damage detection; the baseline model, which reduces the influences of lacking the prior knowledge of the structure, is always developed first in actual tests. So, to compare the detection results with Hu et al. [15] and Perera and Torres [14], the unmodified numerical model is used first and then the baseline model is adopted, as well to simulate a more general situation. The input parameters are the same, as shown in Table 4. The indexes' searching spaces are individually set as  $[-0.5, 0.5]$  and  $[0, 1]$  in different trial

calculations, and their final identification results are similar. To maintain consistency with Hu et al. [15] and Perera and Torres [14], the identification results based on indexes' searching space of  $[0, 1]$  are listed. By applying the same domain, the accuracy of the proposed algorithm can be fairly compared.

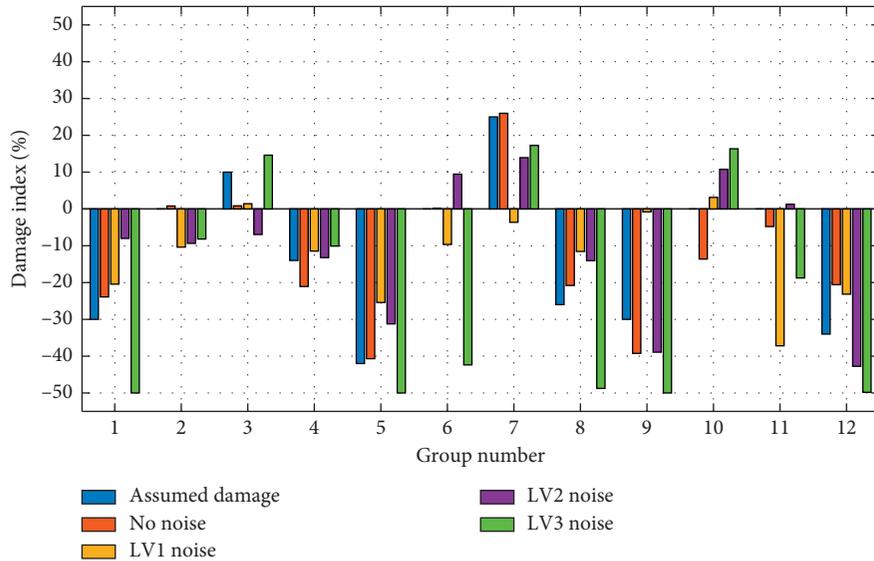
Figures 17 and 18 display the first three orders of modal shapes of the undamaged and damaged beams; Table 7 shows the corresponding MAC values; Tables 8 and 9 contain the related frequencies information. When comparing the data obtained from the baseline model and the original model with the test data, as shown in Figures 17 and 18 and Table 7, the modal data based on the baseline model



(a)



(b)



(c)

FIGURE 15: (a) The identified parameters with 13 sensors. (b) The identified parameters with 7 sensors. (c) The identified parameters with 5 sensors.

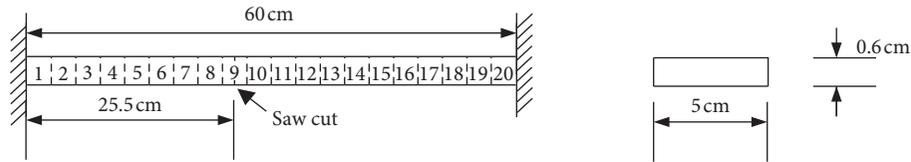


FIGURE 16: The structure for the experimental study.

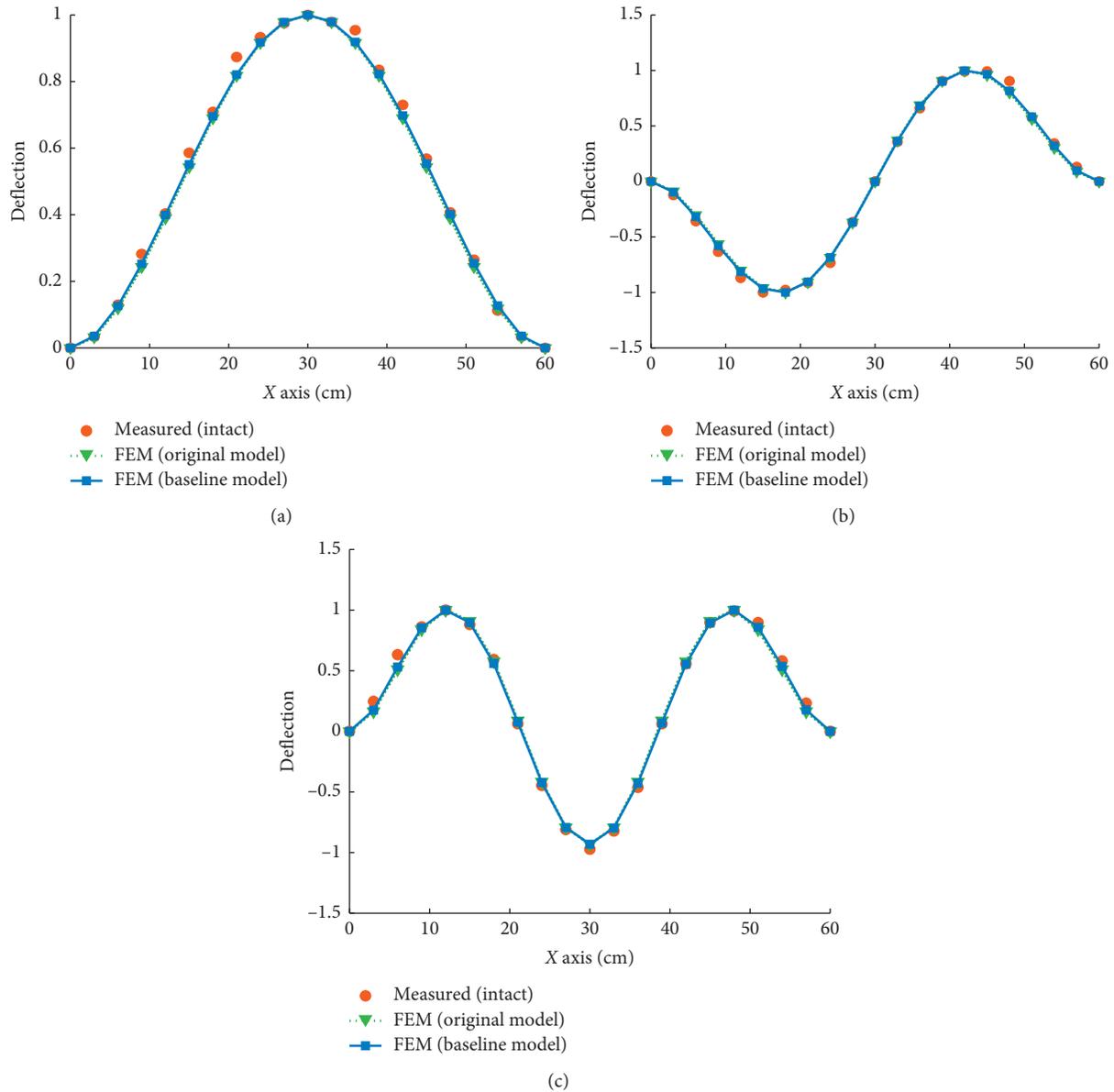


FIGURE 17: Modal shapes of the undamaged beam of (a) the first modal, (b) the second modal, and (c) the third modal.

are closer to the test data, which is consistent with the expectation that the adoption of the baseline model improves the performance of the FE model. However, as a similar reason is shown in Section 4.3.2, the global minimum is replaced by a local minimum due to test noises and modeling errors, which leads to the frequencies error of the baseline model more significant than the error of the original model in this specific case.

The final identification results obtained from Hu et al. [15] (DDNK), Perera and Torres [14] (GA), original model, and baseline model are listed simultaneously in Figure 19. As expected, due to the inappropriate grouping and noises, several elements besides the damaged one (i.e., element 9) are identified as damaged, which is similar to the results in Section 4.3.2. In this case, a smaller number of damaged elements means better performance (because there is only

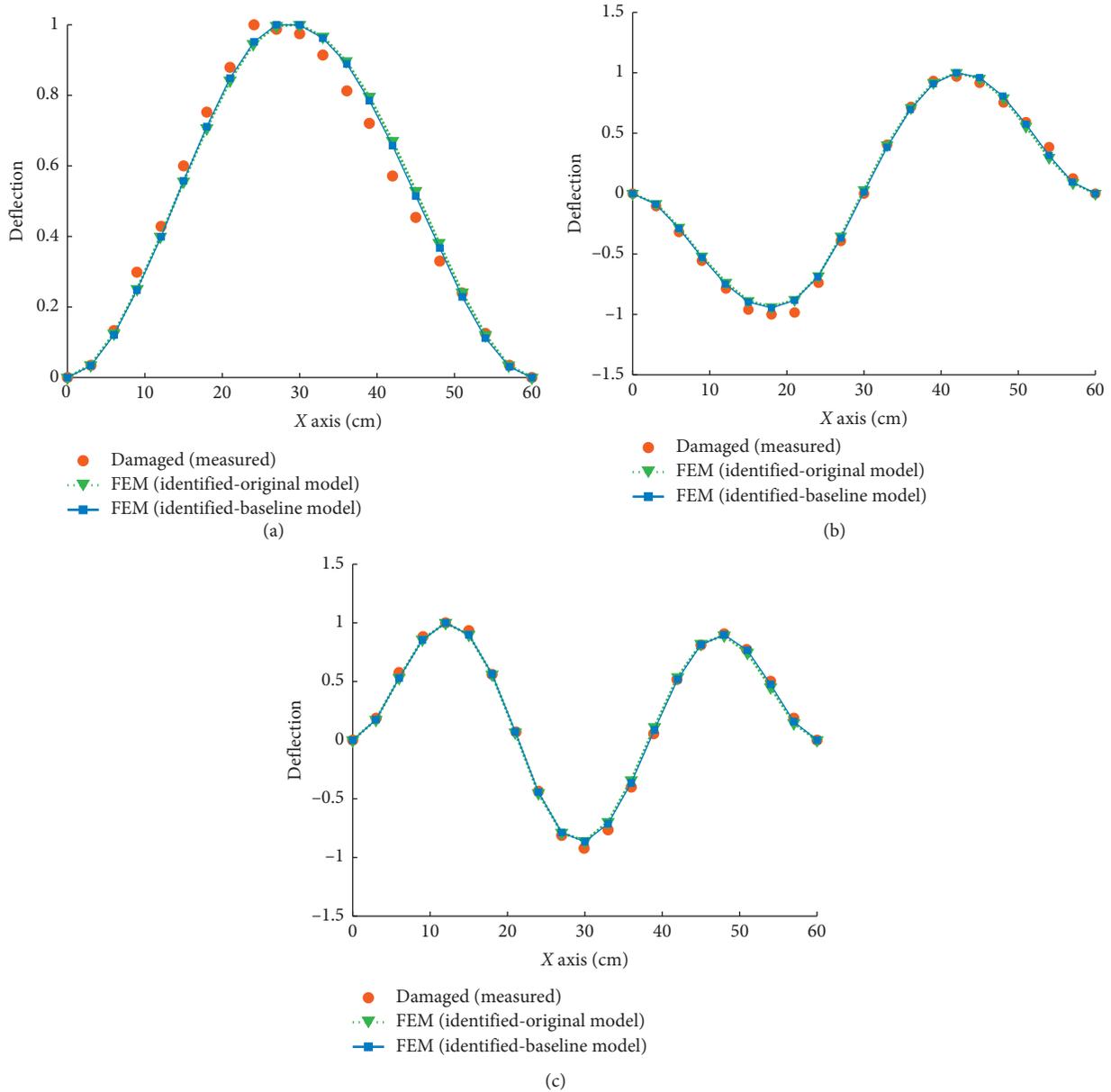


FIGURE 18: Modal shapes of the damaged beam of (a) the first modal, (b) the second modal, and (c) the third modal.

TABLE 7: MAC of the experimental beam.

MAC of		The first model	The second model	The third model
Undamaged beam	Original model	0.999225688	0.997261687	0.996034642
	Baseline model	0.999423431	0.998165522	0.997671728
Damaged beam	Original model	0.994524097	0.996194531	0.950126061
	Baseline model	0.995630316	0.997770042	0.960814259

TABLE 8: Frequencies of the undamaged beam.

	First-order frequency (Hz)			Second-order frequency (Hz)			Third-order frequency (Hz)		
	Numerical	Test	Error (%)	Numerical	Test	Error (%)	Numerical	Test	Error (%)
Original model	87.232	85.726	1.76	240.454	238.572	0.79	471.359	467.694	0.78
Baseline model	84.638		1.23	234.937		1.52	461.950		1.17

\*Error = |numerical - test|/test.

TABLE 9: Frequencies of the damaged beam.

Identified by	First-order frequency (Hz)			Second-order frequency (Hz)			Third-order frequency (Hz)		
	Numerical	Test	Error (%)	Numerical	Test	Error (%)	Numerical	Test	Error (%)
Original model	82.978		1.20	232.007		1.61	458.107		1.00
Baseline model	81.876	83.984	2.51	229.305	235.792	2.75	453.064	462.712	2.09

\*Error = |numerical – test|/test.

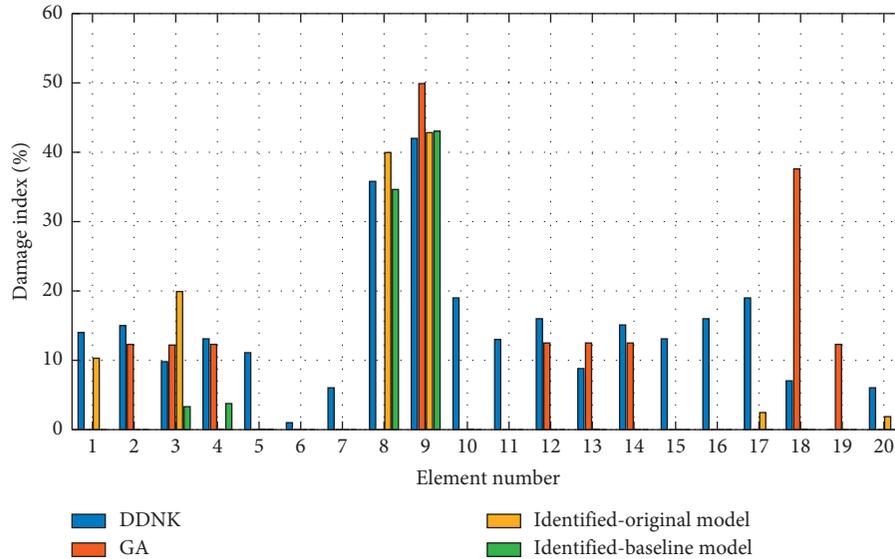


FIGURE 19: The identified parameters of the experimental study.

one damaged element), and a more extensive damage index of the element suggests a higher damage probability at this location. In other words, the two identified models using the TCABC algorithm work better in this case, and elements 8 and 9 have a more significant probability of being the real damage part. Thus, it can be conducted that the TCABC algorithm, as a better method, can precisely locate damage position in a smaller approximate range. And the identified baseline model has the best identification ability.

## 6. Conclusions

A novel ABC algorithm (i.e., TCABC algorithm) for structural damage detection via modal and frequency analyses is presented; two searching methods, tabu search method and chaotic search method, are adopted to enhance the exploration and exploitation ability. The tabu search method uses a memory function to avoid the solution being trapped in a local minimum, which increases the exploitation ability. The chaotic search method generates more searching points for finding the global minimum, which increases the exploration ability. Additionally, the original roulette wheel selection is replaced by the tournament selection to enhance the global searching ability of TCABC algorithm. Several explicit test functions and an implicit damage detection function are employed for comparing the numerical results obtained from ABC and TCABC

algorithms. The numerical results show that compared to the standard ABC algorithm, the performance of TCABC algorithm is improved. The applicability of the TCABC algorithm for damage detection is verified under different numerical circumstances. The results show the following: (1) fewer groups lead to faster convergence as demonstrated by both ABC and TCABC algorithms in the same damage situation; (2) even though the precision of TCABC algorithm cannot be guaranteed when inappropriate groupings are assigned to the structure, the approximate damage position can still be identified; (3) the environmental, electromagnetic, and instrumental noises in the field test influence the damage detection results. Stronger noises result in worse identification results; (4) regardless of whether or not the noises exist, the more data are measured, the more accurate the results can be. Based on the above statements, to obtain better-damaged detection results in actual tests, a large amount of measured data, good test and equipment conditions, and reasonable groupings are all required. A detailed and complete analysis of setting reasonable groupings will be considered in subsequent research. An experimental study is used for checking the accuracy and validity of the TCABC algorithm for structural damage detection. Compared with DDNK and GA, the proposed TCABC algorithm performs much better, and the detection results based on the baseline model are better than the original model.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This study was financially supported by the National Natural Science Foundation of China (nos. 51608207 and 51478193).

## References

- [1] J. Li and H. Hao, "A review of recent research advances on structural health monitoring in western Australia," *Structural Monitoring and Maintenance*, vol. 3, no. 1, pp. 33–49, 2016.
- [2] K. Liu, J. Yan, M. S. Alam, and C. Zou, "Seismic fragility analysis of deteriorating recycled aggregate concrete bridge columns subjected to freeze-thaw cycles," *Engineering Structures*, vol. 187, pp. 1–15, 2019.
- [3] T. Choudhury and H. B. Kaushik, "Treatment of uncertainties in seismic fragility assessment of RC frames with masonry infill walls," *Soil Dynamics and Earthquake Engineering*, vol. 126, Article ID 105771, 2019.
- [4] B. Jia, X. Yu, Q. Yan, and Z. Yang, "Analysis of bridge time-dependent performance based on dynamic Bayesian networks," *International Journal of Earth Sciences and Engineering*, vol. 9, no. 6, pp. 2427–2436, 2016.
- [5] H.-P. Zhu, B. He, and X.-Q. Chen, "Detection of structural damage through changes in frequency," *Wuhan University Journal of Natural Sciences*, vol. 10, no. 6, pp. 1069–1073, 2005.
- [6] N. Bicanic and H.-P. Chen, "Damage identification in framed structures using natural frequencies," *International Journal for Numerical Methods in Engineering*, vol. 40, no. 23, pp. 4451–4468, 1997.
- [7] C. P. Ratcliffe, "Damage detection using a modified Laplacian operator on mode shape data," *Journal of Sound and Vibration*, vol. 204, no. 3, pp. 505–517, 1997.
- [8] S. Sehgal and H. Kumar, "Structural dynamic model updating techniques: a state of the art review," *Archives of Computational Methods in Engineering*, vol. 23, no. 3, pp. 515–533, 2016.
- [9] S.-S. Jin and H.-J. Jung, "Sequential surrogate modeling for efficient finite element model updating," *Computers & Structures*, vol. 168, pp. 30–45, 2016.
- [10] W.-X. Ren, S.-E. Fang, and M.-Y. Deng, "Response surface-based finite-element-model updating using structural static responses," *Journal of Engineering Mechanics*, vol. 137, no. 4, pp. 248–257, 2011.
- [11] M. Sanayei, J. E. Phelps, J. D. Sipple, E. S. Bell, and B. R. Brenner, "Instrumentation, nondestructive testing, and finite-element model updating for bridge evaluation using strain measurements," *Journal of Bridge Engineering*, vol. 17, no. 1, pp. 130–138, 2012.
- [12] C. S. N. Pathirage, J. Li, L. Li, H. Hao, W. Liu, and P. Ni, "Structural damage identification based on autoencoder neural networks and deep learning," *Engineering Structures*, vol. 172, pp. 13–28, 2018.
- [13] K. Moslem and R. Nafaspour, "Structural damage detection by genetic algorithms," *AIAA Journal*, vol. 40, no. 7, pp. 1395–1401, 2002.
- [14] R. Perera and R. Torres, "Structural damage detection via modal data with genetic algorithms," *Journal of Structural Engineering*, vol. 132, no. 9, pp. 1491–1501, 2006.
- [15] N. Hu, X. Wang, H. Fukunaga, Z. H. Yao, H. X. Zhang, and Z. S. Wu, "Damage assessment of structures using modal test data," *International Journal of Solids and Structures*, vol. 38, no. 18, pp. 3111–3126, 2001.
- [16] Y.-S. Park, S. Kim, N. Kim, and J.-J. Lee, "Finite element model updating considering boundary conditions using neural networks," *Engineering Structures*, vol. 150, pp. 511–519, 2017.
- [17] W. Qiao and Z. Yang, "Solving large-scale function optimization problem by using a new metaheuristic algorithm based on quantum dolphin swarm algorithm," *IEEE Access*, vol. 7, pp. 138972–138989, 2019.
- [18] F. Kang, J.-J. Li, and Q. Xu, "Damage detection based on improved particle swarm optimization using vibration data," *Applied Soft Computing*, vol. 12, no. 8, pp. 2329–2335, 2012.
- [19] W. Qiao, Z. Yang, Z. Kang, and Z. Pan, "Short-term natural gas consumption prediction based on Volterra adaptive filter and improved whale optimization algorithm," *Engineering Applications of Artificial Intelligence*, vol. 87, Article ID 103323, 2020.
- [20] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Technical Report-TR06, Department of Computer Engineering, Engineering Faculty, Erciyes University, Kayseri, Turkey, 2005.
- [21] D. Karaboga and B. Akay, "A comparative study of artificial bee colony algorithm," *Applied Mathematics and Computation*, vol. 214, no. 1, pp. 108–132, 2009.
- [22] B. Akay and D. Karaboga, "Parameter tuning for the artificial bee colony algorithm," *Computational Collective Intelligence. Semantic Web, Social Networks and Multiagent Systems*, pp. 608–619, 2009.
- [23] S. M. Awan, M. Aslam, Z. A. Khan, and H. Saeed, "An efficient model based on artificial bee colony optimization algorithm with neural networks for electric load forecasting," *Neural Computing and Applications*, vol. 25, no. 7-8, pp. 1967–1978, 2014.
- [24] E. Uzlu, A. Akpınar, H. T. Öztürk, S. Nacar, and M. Kankal, "Estimates of hydroelectric generation using neural networks with the artificial bee colony algorithm for Turkey," *Energy*, vol. 69, pp. 638–647, 2014.
- [25] N. Menon and R. Ramakrishnan, "Brain tumor segmentation in MRI images using unsupervised artificial bee colony algorithm and FCM clustering," in *Proceedings of the 2015 International Conference on Communications and Signal Processing (ICCSP)*, pp. 6–9, Chengdu, China, October 2015.
- [26] L. Wen, J. Cheng, F. Li, J. Zhao, Z. Shi, and H. Zhang, "Global optimization of controlled source audio-frequency magnetotelluric data with an improved artificial bee colony algorithm," *Journal of Applied Geophysics*, vol. 170, Article ID 103845, 2019.
- [27] H. Sun and R. Betti, "Simultaneous identification of structural parameters and dynamic input with incomplete output-only measurements," *Structural Control and Health Monitoring*, vol. 21, no. 6, pp. 868–889, 2013.
- [28] Z. H. Ding, M. Huang, and Z. R. Lu, "Structural damage detection using artificial bee colony algorithm with hybrid search strategy," *Swarm and Evolutionary Computation*, vol. 28, pp. 1–13, 2016.

- [29] F. Glover, "Future paths for integer programming and links to artificial intelligence," *Computers & Operations Research*, vol. 13, no. 5, pp. 533–549, 1986.
- [30] B. Li and W. Jiang, "Optimizing complex functions by chaos search," *Cybernetics and Systems*, vol. 29, no. 4, pp. 409–419, 1998.
- [31] D. Ewins, *Modal Testing*, Research Studies Press, Baldock, UK, 2000.
- [32] S. Ghambari and A. Rahati, "An improved artificial bee colony algorithm and its application to reliability optimization problems," *Applied Soft Computing*, vol. 62, pp. 736–767, 2018.
- [33] F. McKenna, M. H. Scott, and G. L. Fenves, "Nonlinear finite-element analysis software architecture using object composition," *Journal of Computing in Civil Engineering*, vol. 24, no. 1, pp. 95–107, 2010.
- [34] A. Gilat, *MATLAB*, Wiley, New York, NY, USA, 2016.



**Hindawi**

Submit your manuscripts at  
[www.hindawi.com](http://www.hindawi.com)

