Smart Structures and Systems, *Vol. 15, No. 1 (2015) 191-207*
DOI: http://dx.doi.org/10.12989/sss.2015.15.1.191

Sensor placement optimization in structural health monitoring using distributed monkey algorithm

Ting-Hua Yi[*1.2], Hong-Nan Li[1a] and Xu-Dong Zhang[1b]

[1]*School of Civil Engineering, Dalian University of Technology, Dalian 116023, P.R. China*
[2]*State Key Laboratory of Coastal and Offshore Engineering, Dalian University of Technology, Dalian 116023, P.R. China*

(Received January 15, 2014, Revised May 20, 2014, Accepted May 25, 2014)

Abstract. Proper placement of sensors plays a key role in construction and implementation of an effective structural health monitoring (SHM) system. This paper proposes a novel methodology called the distributed monkey algorithm (DMA) for the optimum design of SHM system sensor arrays. Different from the existing algorithms, the dual-structure coding method is adopted for the representation of design variables and the single large population is partitioned into subsets and each subpopulation searches the space in different directions separately, leading to quicker convergence and higher searching capability. After the personal areas of all subpopulations have been finished, the initial optimal solutions in every subpopulation are extracted and reordered into a new subpopulation, and the harmony search algorithm (HSA) is incorporated to find the final optimal solution. A computational case of a high-rise building has been implemented to demonstrate the effectiveness of the proposed method. Investigations have clearly suggested that the proposed DMA is simple in concept, few in parameters, easy in implementation, and could generate sensor configurations superior to other conventional algorithms both in terms of generating optimal solutions as well as faster convergence.

Keywords: structural health monitoring; optimal sensor placement; distributed monkey algorithm; harmony search algorithm; modal assurance criterion

1. Introduction

Large and complex civil infrastructures are being placed in new and extreme conditions for extended periods of time. As a result, concerns on the structural integrity, durability and reliability, i.e., the health state of the civil infrastructures continues to grow (Yi *et al*. 2012a, Lei *et al*. 2012a). Structural health monitoring (SHM) provides a vital manner for the safe operation of key civil infrastructures and enables operational cost reduction by performing prognostic and preventative maintenance (Lei *et al*. 2013a, b). In general, a typical SHM system includes three major components: a sensor system, data processing system (including data acquisition, transmission and storage), and health evaluation system (including diagnostic algorithms and information

∗Corresponding author, Professor, E-mail: yth@dlut.edu.cn
[a] Professor, E-mail: hnli@dlut.edu.cn
[b] MSc Student, E-mail: zxd21006040@126.com

http://www.techno-press.org/?journal=sss&subpage=8 ISSN: 1738-1584 (Print), 1738-1991 (Online)

management) (Wenzel 2009). Obviously, the efficiency of the SHM system relies on the acquired data to structural changes that may be obtained by an extended sensor network which ideally has sensor information at all the degrees of freedom (DOFs) of a finite element (FE) model used for monitoring the integrity of the structure. Practically, however, due to cost, weight and accessibility issues a limited number of locations can be instrumented. In such cases, a question that naturally arises is how to select a set with a minimum number of sensor locations from all possibilities, such that the data collected may provide the greatest opportunities for identification of the structural behavior. Otherwise, incomplete dynamic behaviors will be measured and an accurate structural safety assessment will be impossible.

For a structure that has simple geometry, or smaller number of DOF, engineer's experience or a trial-and-error approach may suffice to solve the problem. However, for those structures that were relatively less studied in the past, it is very difficult to determine the sensor locations according to past experience or empirical methods. In addition, for the large-scale complicated structure, candidate sensor positions are usually large (counts in order of thousands to tens of thousands) (Lei *et al*. 2013b). The amount of work for the sensor placement is thus not insignificant, and computation efficiency is of great concern. In order to detect sensitive changes within the structure, a more systematic and efficient approach is needed. Numerous techniques have been advanced for solving the optimal sensor placement (OSP) problem and are widely reported in the literature (Maul *et al*. 2007). These have been developed using a number of approaches, some based on gradient optimization approaches, others employing systematic optimization methods. However, gradient-based local optimization methods are unable to handle efficiently the multiple local optima and may present difficulties in estimating the global minimum. In the recent years, computational intelligence approaches have been extensively used for the optimization of OSP problems due to its many advantages over the classical optimization techniques such as global optimization, blind search and highly parallel properties. Among the emerged algorithms, the Genetic algorithm (GA), which is a global probabilistic search algorithm inspired by the Darwin's survival-of-the-fittest theory (Holland 1975), has been proven as an effective and powerful tool to the OSP problem (Yi *et al*. 2011a). The mapping between the physical minimization variables and the chromosomes is the big difficulty in the application of GAs to the OSP problem. The traditional coding method is the simple one dimension binary coding method which is intuitionistic (Liu *et al*. 2006). However, the number of sensors will be changed in the crossover and mutation, which is impractical and should avoided (Yi *et al*. 2011a). In addition, the binary coding method requires increased string length and computational time especially in the large-scale structures where possible sensor combinations are large. Roy *et al*. (2009) and Chow *et al*. (2010) suggested using the integer coding method to get over the problem. Huang *et al*. (2010) proposed a kind of dual-structure coding approach to overcome the problem. Different genetic operators, such as force mutation (Yao *et al*. 1993), filter operator (Abdullah *et al*. 2001), and partially matched crossover (PMX) (Huang *et al*. 2005), have also been used in order to overcome these faults although the process of these particular genetic operators are complex and the computational efficiency is relatively low. Hwang and He (2006) used simulated annealing and adaptive mechanism to insure the solution quality and to improve the convergence speed. In addition, the GAs and other intelligent algorithms suffer from the curse of dimensionality, when the dimension of the parameter space increases, the computational time required to solve the problem tends to increase, while the quality of the solution tends to decrease (Ngatchou *et al*. 2005). In order to improve the convergence speed and avoid premature convergence, some improved GAs have also been adopted to sensor placement problems, such as the elite genetic algorithms (EGA) (Brooks *et al*. 1996), the virus evolutionary genetic algorithm (VEGA) (Yang *et al*. 2007), the coevolutionary genetic algorithm (CGA) (Lin *et al*. 2009), the multi-objective genetic algorithm (MGA) (Jia *et al*. 2009), and the generalized genetic algorithms (GGA) (Yi *et al*. 2011b). The successful application of GAs in the sensor network design for the SHM system led to the development of several other intelligent approaches, such as the simulated annealing (SA) algorithm (Gou and Cui 2008), particle swarm optimization (PSO) algorithm (Kukunuru *et al*. 2010), and Ant colony optimization (ACO) algorithm (Fidanova *et al*. 2012). However, in most of these approaches, either very limited network characteristics were considered, or several requirements of the application cases were not incorporated into the performance measure of the algorithm.

The above discussion renders solving the OSP problems for complex structures difficult. Keeping these things in view, a novel intelligent algorithm called distributed monkey algorithm (DMA) that could tackle the practical civil engineering OSP problem with a large number of possible candidate sensor locations is proposed in this paper. Different from the existing intelligent algorithms, the DMA uses several small subpopulations in place of a single large population and simple monkey algorithm (SMA) is executed on each subpopulation separately, leading to quicker convergence and higher searching capability. The remainder of this paper is organized as follows: Section 2 presents the basic features and detailed implementation steps of the proposed algorithm. Section 3 introduces the performance index used to optimize sensor placement. In the following Section 4, the effectiveness of the proposed algorithm is demonstrated via a numerical simulation study for the OSP in a high-rise structure. Finally, in Section 5, some overall conclusions are drawn.

2. Distributed monkey algorithm for sensor placement

2.1 Basic concepts and ideas

The monkey algorithm (MA) was developed originally by Zhao and Tang (2008), is a population-based algorithm, which is inspired by the mountain-climbing process of monkeys. The algorithm mainly consists of climb process, watch-jump process, and somersault process in which the climb process is employed to search the local optimal solution, the watch-jump process to look for other points whose objective values exceed those of the current solutions so as to accelerate the monkeys' search courses, and the somersault process to make the monkeys transfer to new search domains rapidly. After much repetitious iteration of the three processes, the mountaintop could be found by the monkeys (i.e., find the optimal value).

For configurations with a small number of sensors, the MA could converge to close to global optimal solutions. But for larger configurations the run-time grows prohibitive. In fact, the climb process is so computationally expensive that it is impossible to run, in a reasonable time, enough MA iterations to adequately explore the parameter search space. An intuitive idea to overcome this defect is that the problem can be decomposed into subproblems of smaller size. This is the core idea of our proposed DMA, i.e., the entire population of monkeys is divided into a number of parallel subpopulations that are permitted to evolve independently to search the space in different directions. After the personal area deep-searching of all subpopulations have been finished, the best monkeys (initial optimal solutions) in every subpopulation are extracted and reordered into a new subpopulation. Then the deep-searching are processed again by another advanced intelligent algorithm (harmony search algorithm, HSA) and the final optimal solution can be found. Fig.1 displays a schematic drawing of the DMA.

2.2 Coding method and initialization

The OSP problem is a kind of single-objective optimization problem involving discrete-valued variables, which could be either the spatial coordinate or the number of DOFs on the FE model mesh excluding the constrained DOFs. Its mathematic model is a 0-1 programming problem, if the value of the j-th code is 1, in which it denotes that a sensor is located on the j-th DOF. In contrast, if the value of the j-th code is 0, it denotes that no sensor is placed on the j-th DOF. The total number of 1 is equal to the sensor number. However, the MA was originally designed to solve optimization problems with continuous variables (Zhao and Tang 2008, Lan *et al*. 2011). The mapping between the physical minimization variables and the monkey's position is thus to become one of a big difficulty in the application of MA in addressing the OSP problem. To implement the MA in the OSP problem, it is necessary to devise a general coding system for the representation of the design variables first.

Considering the characteristics of the OSP problem, a kind of dual-structure coding method is designed and adopted for the representation of the design variables in the DMA. Let ordered pair (x,c) stand for the possible solutions of each monkey, where x denotes the position vector in the DMA and c means the binary vector which represents the sensor's location, thus, an outline of solution representation using dual-structure coding method and initialization process are given as follows:

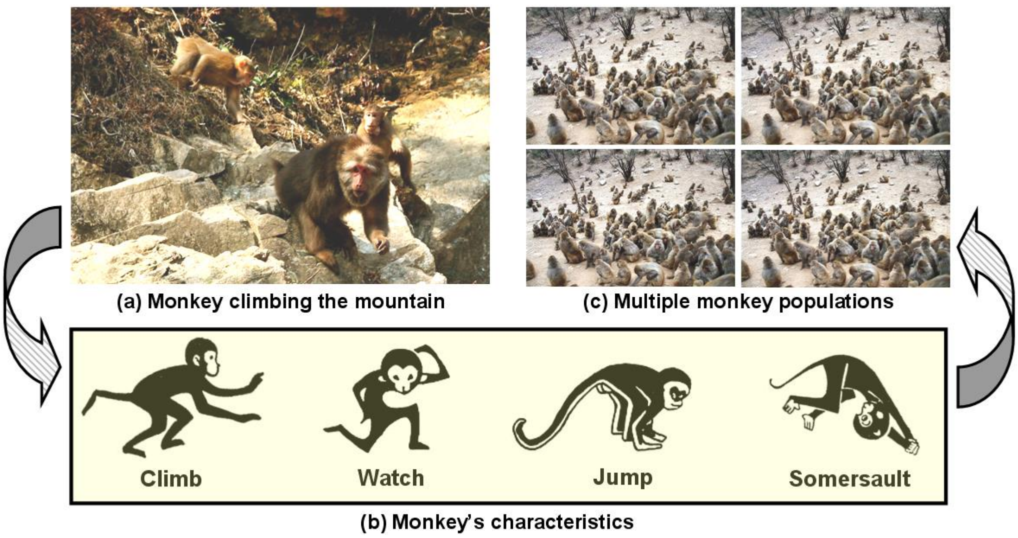


Fig. 1 Schematic drawing of the DMA

Step (1): Suppose there are $fnum$ candidate sensor positions (i.e., the total DOF of the FE model), thus the $fnum$ integers from $1\sim fnum$ can be obtained.

Step (2): For the monkey i in the monkey population, its solution of the proposed optimization problem is denoted as $p_i=(x_i,c_i)=\{(x_{i,1},c_{i,1}),(x_{i,2},c_{i,2}),....,(x_{i,fnum},c_{i,fnum})\}$, in which the component of the position vector x_i is the real number selected randomly from the interval $[down,up]$, where $down=-5$, $up=5$, and c_i is the binary vector which can be obtained by the follow equation (Yi *et al*. 2012b):

$$sig(x_i)=\frac{1}{1+e^{-x_i}} \tag{1}$$

When using equation (1), a judgment threshold ε should be defined first. That is, if $sig(x)\le\varepsilon$, then c_i=0 (i.e., no sensor is located on this DOF); if $sig(x)>\varepsilon$, then c_i=1 (i.e., a sensor is located on this DOF). In this paper, the ε is defined as 0.5, thus when select real number x_i randomly from the interval $[-5,5]$, it can be found that $0.0067\le sig(x_i)\le 0.9933$ and $sig(0)=0.5$ which proves that the judgment threshold given here is reasonable.

Step (3): It should be noted that the total number of 1 may be not equal to the sensor number sp after initialization process. In order to guarantee all the possible solutions in the monkey population satisfy the requirement, the initial monkey population is generated by regeneration method when this issue is encountered.

In the iterative process of the proposed DMA, the position vector x, which satisfies the requirement of the MA, is used firstly; and then, the Eq. (1) is adopted to obtain the binary vector c which subsequently is utilized to calculate the optimal objective value; as a consequence, each monkey will arrive at its own best position representing the personal optimal objective value $f(x_i,c_i)$ when the iteration accuracy has been achieved or a relative large number of iterations has been reached.

2.3 Distribution mode of multiple monkey populations

As aforementioned, our proposed DMA uses several small subpopulations in place of a single large population and simple monkey algorithm (SMA) is executed on each subpopulation separately, which has simple concept, fast calculation speed, and good global search capability. After the initial monkey population is randomly generated, the monkeys are arranged from good to bad according to their optimal objective values to divide the whole population into subpopulations. The outline of the proposed distribution mode is as follows:

Step (1): Suppose that there are M monkey subpopulations needing to be defined and each subpopulation has N monkeys. Thus, $P=M\times N$ monkeys should be randomly generated to compose initial single large monkey population $P=\{(x_1,c_1),(x_2,c_2),.....,(x_p,c_p)\}$.

Step (2): Calculate all monkeys' fitness values $f(x_i,c_i)$ and sort them from good to bad.

Step (3): Divide $P=M\times N$ monkeys into M subpopulations. Among them, the monkey ranking 1st is assigned into 1st subpopulation, one ranking 2nd into 2nd sub-population, one ranking M into Mth subpopulation, one ranking $M+1$ into 1st subpopulation, one ranking $M+2$ into 2nd subpopulation, analogizing in sequence until all monkeys have been assigned.

2.4 Obtaining initial optimal solution using SMA

Then, every subpopulation is used for personal area deep searching and to find the M initial optimal solution using the SMA. The process of the SMA is summarized as follows:

(1) *Climb process*

Climb process is a step-by-step procedure to change the monkeys' positions from the initial positions to new ones that can make an improvement in the objective function. In this paper, the thought of large-step climb in reference (Wang *et al*. 2010) is taken during the climb process. The large-step climb process makes the monkeys' positions greatly change before/after updating, which can expand the search extent of the potential solution.

For the monkey i in a subpopulation with the position $x_i = (x_{i,1}, x_{i,2}, ..., x_{i,fnum})$, $i = 1, 2, \cdots, N$, an outline of the improved climb process is given as follows:

Step (1): Randomly generate a integer vector $\Delta x_i = (\Delta x_{i,1}, \Delta x_{i,2}, ..., \Delta x_{i,fnum})^T$ in the interval $[-as, as]$, $j \in \{1, 2, ..., fnum\}$, respectively, where the parameter as $(as > 0)$ is called the step length of the climb process.

The step length as plays a crucial role in the precision of the approximation of the local solution in the climb process. Usually, the smaller the parameter as is, the more precise the solutions are. Considering the characteristics of the OSP problem, the as should be defined as 1, 2, or other positive integer.

Step (2): Calculate $f((x_i + \Delta x_i), c_{new})$, update the monkeys' position x_i with $x_i + \Delta x_i$ (update c_i with c_{new} accordingly) only if $f((x_i + \Delta x_i), c_{new}) < f(x_i, c_i)$, otherwise keep x_i unchanged;

Step (3): Repeat steps (1) to (2) until there is little change on the values of objective function in the neighborhood iterations or the maximum allowable number of iterations (called the climb number, denoted by Ns) has been reached.

It has to be noted that the "spillover" phenomenon sometimes occurs in step (2) and other steps (i.e., the new components in $x_i + \Delta x_i$ may exceed the interval $[down, up]$). Thus, here if a new component exceeds the upper limit up, then take the component to up; if a new component below the lower limit $down$, then take the component to $down$.

(2) *Watch-jump process*

For each monkey, when it gets on the top of the mountain in the local area, it is natural to have a look and to find out whether there are other mountains around it higher than its present positions. If yes, it will jump to some place of the mountain watched by it from the current position (this action is called "watch-jump process") and then repeat the climb process until it reaches the highest top of the mountain.

For the monkey i in a subpopulation, the outline of the proposed watch-jump process is as follows:

Step (1): Randomly generate integer numbers x''_{ij} from $[x_{ij} - bs, x_{ij} + bs]$, $j \in \{1, 2, ..., fnum\}$, where the parameter bs is a positive integer which represents the eyesight of the monkey (i.e. the maximal distance that the monkey can watch), thus the new position $x''_i = (x_{i,1}, x_{i,2} .... x_{i,fnum})^T$ can be obtained.

Usually, the bigger the feasible space of optimal problem is, the bigger the value of the parameter bs should be taken. The eyesight bs can be determined by specific situations, like the step length as, the eyesight bs should also be defined as 1, 2, or other positive integer in sensor location problem.

Step (2): Calculate $f(x_i'', c_{new})$, update the monkeys' position x_i with x_i'' provided that $f(x_i'', c_{new}) < f(x_i, c_i)$, otherwise go back to the *Step* (1).

(3) *Somersault process*

After repetitions of the climb process and the watch-jump process, each monkey will find a locally highest mountaintop around its initial point. In order to find a much higher mountaintop, it is natural for each monkey to somersault to a new search domain (this action is called "somersault process").

For the monkey i in a subpopulation, the outline of the proposed somersault process is as follows:

Step (1): Generate integer numbers θ randomly from the interval $[cs, ds]$ (called the somersault interval which governs the maximum distance that monkeys can somersault).

Step (2): Obtain the monkeys' pivot $ps = (ps_1, ps_2, ..., ps_{fnum})^T$ by calculating all the monkeys' barycentre $ps_j = \sum_i^N x_{ij} \Big/ N$, $j \in \{1, 2, ..., fnum\}$.

Step (3): Calculate $x_{ij}''' = x_{ij} + round(\theta \mid p_j - x_{ij} \mid)$, update the monkeys' position with $x_i''' = (x_{i,1}''', x_{i,2}''', ... x_{i,fnum}''')$ provided that the new objective values of x_i''' are better, and then return to the large-step climb process; otherwise go back to the *Step* (1).

The condition for terminating the SMA iteration could either be when the iteration accuracy has been achieved, or when a relative large number of iterations N has been reached. For the problem considered in this paper, the ending condition of the SMA is chosen to be the later one to avoid redundant iteration. After all subpopulations finish the iteration operation, the initial solution $P1_i$ can be obtained, where $i \in \{1, 2, ..., M\}$.

2.5 Obtaining final optimal solution using HAS

Although the SMA has proven its ability of finding near global regions within a reasonable time, it is comparatively poor at finding the precise optimum solution in the region to which the algorithm converges. Thus, here the HSA is employed to improve the precision of the initial solutions obtained by the SMA. After the personal area deep-searching of all subpopulations have been finished, the best M monkeys in every subpopulation are extracted and reordered into a new subpopulation. Then the deep-searching are processed again by the HSA to obtain the final optimal solution. The HSA is a new meta-heuristic optimization method imitating the music improvisation process where musicians improvise their instruments' pitches searching for a perfect state of harmony (Geem *et al*. 2001). The HSA is simple in concept, few in parameters, and easy in implementation, and it has good ability of global search although it is relatively inefficient in performing local search. This kind of combination is ideal to compensate deficiencies of the individual algorithms. Since the HSA is designed to solve engineering optimization problems with continuous design variables, the dual-structure coding method is adopted too. Correspondingly, the main steps of the improved HSA are as follows:

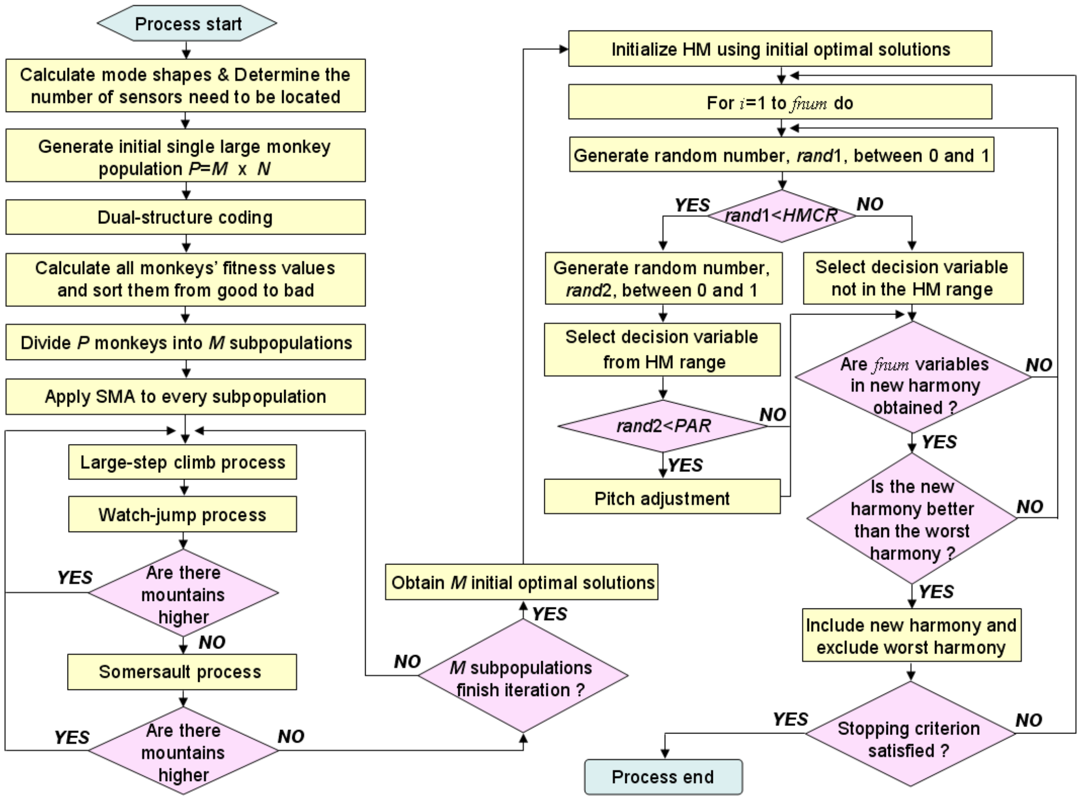


Fig. 2 Flowchart of the DMA for OSP

Step (1): Initialize the harmony memory (HM). For initialization, the HM is a memory location where the M initial optimal solutions are stored, i.e., $P1_i = ((x_{i,1}, c_{i,1}), (x_{i,2}, c_{i,2}), ...(x_{i,fnum}, c_{i,fnum}))$, where $i \in \{1, 2, ..., M\}$, and then, the M harmony vector can be obtained and the worst one is denoted as $P1_m$.

Step (2): Improvise a new harmony. A new harmony, $P1_{new} = (x_{new,1}, x_{new,2}, ...x_{new,fnum})$, is generated based on three rules: 1) the memory consideration, 2) the pitch adjustment and 3) the random selection. Generating a new harmony is called "improvisation". In the memory consideration, the harmony memory considering rate (HMCR), which varies between 0 and 1, is the rate of choosing one value from historic values stored in the HM, while (1-HMCR) is the rate of randomly selecting one value from other possible range of values. Generate a uniform random number, $rand1$, between 0 and 1, if $rand1 < \text{HMCR}$, the value of the first decision variable $x_{new,1}$ for the new vector is chosen from any of the values in the specified HM range $(x_{1,1}, x_{2,1}, ..., x_{M,1})$; Otherwise, the values of the first decision variable is chosen from other possible range of values not in the specified HM range but within the range $[down, up]$. Values of the other decision variables are chosen in the same manner.

Every component $x_{new,1}$ obtained by the memory consideration is examined to determine whether it should be pitch-adjusted. Generate a random number, $rand2$, between 0 and 1, if $rand2 \le \mathrm{PAR}$, it should be pitch-adjusted, otherwise doing nothing. The pitch adjustment operation needs to use pitch adjusting rate (PAR) parameters as follows

$$P1_{new,1} = round(x_{new,1} + 2 * v * rand - v) \tag{2}$$

where, $round$ stands for rounding, v means the arbitrary distance bandwidth, $rand$ denotes a random number between 0 and 1.

Step (3): Update the harmony memory. If the new harmony vector $P1_{new}$ has better fitness function than the worst harmony $P1_m$ in the HM, the new harmony is included in the HM and the existing worst harmony is excluded from the HM.

Step (4): Check the stopping criterion. If the stopping criterion (i.e., maximum number of improvisations Nh) is satisfied, computation is terminated. Otherwise, Steps (2) and (3) are repeated.

To sum up, the whole flowchart of the proposed DMA to find the optimal sensor locations presented herein is shown in Fig. 2, that can be fully implemented easily with the commercial software MATLAB (MathWorks, Natick, MA, USA).

3. Objective function

In the case under investigation the objective function is a weighting function that measures the quality and the performance of a specific sensor network design. In order to keep the original properties of the structure, the larger space angles among the measured modal vectors should be guaranteed. Carne and Dohmann (1995) thought that the modal assurance criterion (MAC) was an ideal scalar constant relating to the relationship between two modal vectors

$$\mathrm{MAC}_{ij} = \frac{(\Phi_i^T \Phi_j)^2}{(\Phi_i^T \Phi_i)(\Phi_j^T \Phi_j)} \tag{3}$$

where, Φ_i and Φ_j represent the ith and jth column vectors in matrix Φ, respectively, and the superscript T denotes the transpose of the vector.

In Eq. (3), the element values of the MAC matrix range between 0 and 1, where zero indicates that there is little or no correlation between the off-diagonal element $\mathrm{MAC}_{ij} (i \ne j)$ (i.e., the modal vector easily distinguishable) and one denotes that there is a high degree of similarity between the modal vectors (i.e., the modal vector fairly indistinguishable). For an optimal (orthogonal) set the MAC matrix would be diagonal, thus the size of the off-diagonal elements could be an indication of optimal result.

In this paper, the biggest value in all the off-diagonal elements in the MAC matrix is defined as the objective function, i.e.

$$\min \quad f_1(x) \tag{4}$$

where, $f_1(x) = \max\limits_{i \ne j} \left\{ \mathrm{MAC}_{ij} \right\}$ and x means the current position of the monkey (i.e., the scheme of the sensor placement).

4. Numerical case study

To demonstrate the effectiveness of the proposed method, a numerical case study to determine the senor configuration on a high-rise building is considered.

4.1 Description of building

The Dalian International Trade Mansion (DITM), currently being constructed in the centre of Dalian city, when completed in the near future, will be the tallest building in the northeast of China. It has 5 stories under the ground level and 79 stories above. The main structure is about 330.25 m high from ground level. The plan of a standard floor is 77.70 m long in the east-west direction and 44.00 m wide in south-north direction, and the floor-to-floor height is 3.8 m. The 10th, 23rd, 37th, 50th, 62nd and 79th floors are the refuge floors, with the height of 5.10 m. Fig. 2(a) shows the bird view of the DITM. (Li *et al*. 2011).

4.2 Analytical model

In order to provide input data for the OSP method, a three-dimensional FE model of the mansion is built using the ETABS software (CSI, Berkeley, CA, USA), as shown in Fig. 3(b). The FE model is built considering the bending and shearing deformation of the beam and column, and also the axial deformation of the column. The rigid floor assumption is used. For the strengthened story, the axial deformation of the column needs to be considered, and so the corresponding floors are computed as flexible floors. The overall model has 34,308 nodes, 34,791 frame elements and 29,071 shell elements, considering 36 section types and 11 materials' properties. The vibration properties were calculated by performing a modal analysis using the FE analytical code and pre/post-processor system ETABS.

For the problem at hand, the size of the searching space is the number of nodes on the FE model excluding the constrained nodes and the vibration nodes of the selected modes. Although the structure has a large number of DOFs, only translational DOFs are considered for possible sensor installation in this case study, as rotational DOFs are usually difficult to measure. Since the structural stiffness of the DITM in two translational directions is obviously different, it should mainly take into account the structural vibration monitoring in the direction of weaker stiffness. Consequently, a total of 79 DOFs are available for sensor installation (i.e., $fnum = 79$), as shown in Fig. 3(c), and the first 8 modes of the DITM are selected for calculation.

4.3 Optimization process, results and discussion

Stephan (2012) suggested that the number of sensors should be 2~3 times the number of modes. Here, it's assumed that the number of sensors needing to be placed on the building is 20 (i.e., $sp = 20$) which have been made and thus optimal locations for the given number of sensors is the target of this paper.

(a) Bird view (b) FE model (c) Simplified model

Fig. 3 Dalian international trade mansion

Table 1 Experiment results for different values of parameters

Scenario	Number of monkeys in each subpopulation (N)	Number of iterations in climb process (Ns)	Somersault interval (cs)	Objective values
1	2	500	2	0.0058
2	2	1000	3	0.0037
3	2	2000	4	0.0095
4	4	500	2	0.0125
5	4	1000	3	0.0076
6	4	2000	4	0.0116
7	8	500	2	0.0064
8	8	1000	3	0.0089
9	8	2000	4	0.0054

(1) *Comparison study*

In order to demonstrate the superiority and also the computational performance of the proposed DMA, three cases are carried out and their performances are compared:

Case (1): The conventional SMA using a single large monkey population;

Case (2): The modified HSA using random generated HM;

Case (3): The proposed DMA using multiple monkey populations.

(2) *Parametric analysis*

The DMA has several parameters that need to be explored and tuned so that the best algorithm performance can be achieved. The aim of this section is to study the evolution of the algorithm solution over generations under different settings of three important parameters: the number of monkeys in each subpopulation (N), the number of iterations in climb process (Ns), and the somersault interval (cs). Here, the number of monkey subpopulations M is set to 5 and the somersault interval is defined as $[-cs, cs]$. Each scenario is tested for 10 times, and the best results are selected. Particularly, the following nine different scenarios are selected and shown in Table 1.

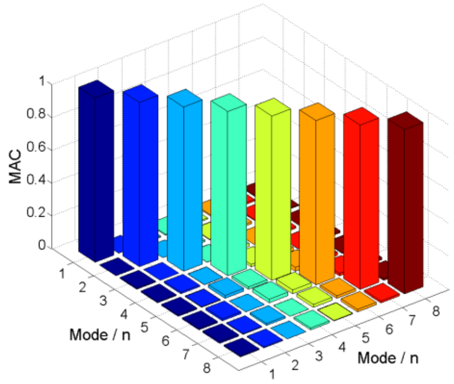
From the empirical study on the impact of different DMA parameters, it is remarked that: 1) the number of monkeys in each subpopulation is larger, much more time (or iterations) is needed for algorithm to find the optimal solution, but usually the higher quality is achieved. Note that when the time or number of iterations is finite, increasing N may deteriorate the quality of the solution. Thus, the population size should be decided according to the specific problems and hardware configuration of computers used. 2) large number of iterations in climb process could cause the improvement of local optimum. However, the local optimum could be obtained without too much iteration. In order to increase the algorithm efficiency, a rational number of iterations Ns in the climb process should be determined. 3) the somersault interval cs in the somersault process governs the maximum distance that monkeys can somersault. However, the disadvantages of this process are that the larger interval of the somersault may skip the global optimal solution while the smaller will lead to a decrease in the solution quality.

Based on this study, it seems that the typical values N, Ns and cs for the DMA can here be set as 4, 2000 and 3, respectively. Considering the characteristics of the OSP problem and the dual-structure coding method, the as and bs are defined as 1 and 2, respectively. According to the reference (Yong *et al*. 2011), the basic parameters of HSA are selected as follows: $\mathrm{HMCR} = 0.9$, $\mathrm{PAR} = 0.3$, $v = 1$, and $Nh = 20000$.

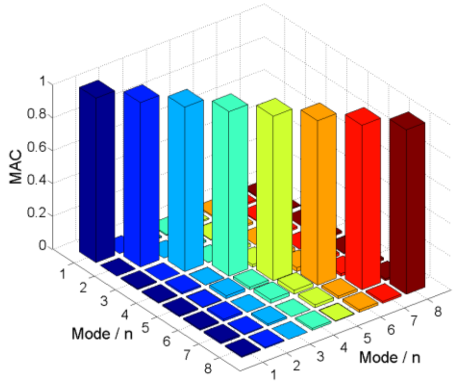
(3) *Optimization results and analysis*

The MAC values obtained by different methods are shown and compared in Fig. 4. Among them, Fig. 4(a) demonstrates the MAC values of all of the 79 DOFs and Fig. 4(b) indicates the MAC values from the initial optimal solution obtained by the DMA in the SMA stage (here, we call it BSMA). It can be seen intuitively that the trend and the values of the MAC values of the different method are very close, while the off-diagonal values in the MAC are very small. The reason for such increasing phenomenon is that the selected first 8 mode shapes in the weak axis may be distinguishable easily.

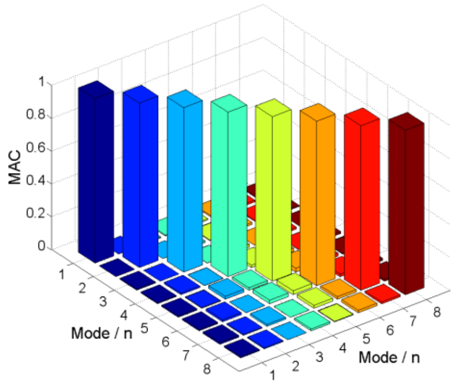
Since the MAC values obtained by different methods are almost similar from the intuitive, therefore, in order to highlight the effectiveness of the proposed algorithm, another figure is plotted in each of the modes (Fig. 5). A close look at the results presented in Fig. 5 indicates that the proposed DMA is far superior to other algorithm implementations in finding the optimal sensor locations. Most of the maximum MAC off-diagonal values obtained by the DMA in each of the modes are much smaller than other algorithms. Further to demonstrate the effectiveness of the proposed algorithm, the maximum MAC off-diagonal values in all of the modes obtained by different method are compared each other in Fig. 6 and Table 2. While compared with each other, the performance of the DMA is found to be of the best cost performance among all of the algorithms as expected. Also, it is evident from Fig.6 that all DOFs yield the worst performance due to the fact that some sensors will conflict with other sensors. Mathematically speaking, the row vectors determined by this way have strong linear relationships with each other. It is shown that the BSMA can improve the performance by 30.6% when compared with the Case 1, which verified the effectiveness of the distribution mode of multiple monkey populations given in this paper.



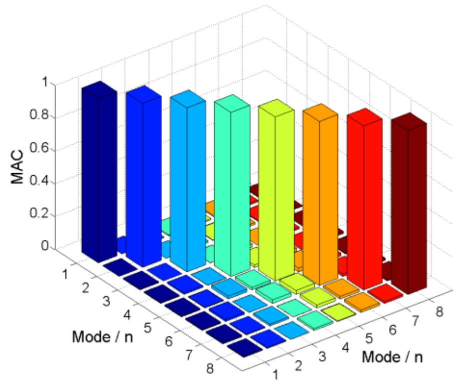
(a) MAC values of all of 79 DOFs



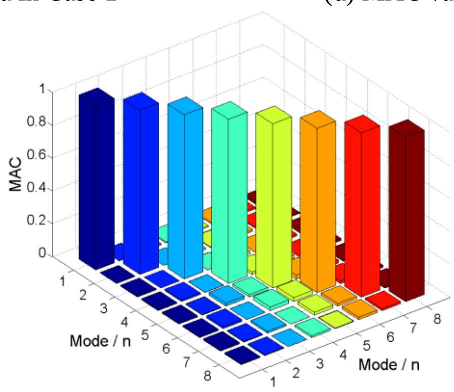
(b) MAC values obtained by BSMA method



(c) MAC values obtained in Case 1



(d) MAC values obtained in Case 2



(e) MAC values obtained in Case 3

Fig. 4 MAC values obtained by different methods

This kind of distribution mode increases the search space which grants high search efficiency to the DMA and consequently the convergence rate of algorithm is improved. It is also remarked from Table 2 that the largest off-diagonal MAC term is 0.0086 for the BSMA, whereas 0.0033 for Case 3, that means the search capabilities of the SMA have been significantly improved by employing the HSA and 61.6% reduction is gained to reach a satisfactory solution. This study clearly indicates that it is desirable to employ a sophisticated search technique like the proposed DMA in order to optimally locate a small number of sensor locations from a large possible candidate sensor set. In addition, the running time of DMA also decreases quickly as the population is partitioned into subsets and each subpopulation search the space in different directions in parallel. Although this kind of comparisons may be imperfect since implementation details will impact execution time, it is reasonable in theory. The final sensor placement result of the DITM obtained by the proposed DMA is given in Table 3.

Table 2 Maximum off-diagonal element of MAC of each kind of sensor placement

Scheme selection of the sensor placement	All of the 79 DOFs	BSMA	Case 1	Case 2	Case 3
Maximum MAC off-diagonal value	0.0239	0.0086	0.0124	0.0143	0.0033

Table 3 Sensor placements of the DITM

Sensor number	1	2	3	4	5	6	7	8	9	10
Story	3	4	6	9	10	14	20	21	29	31
Sensor number	11	12	13	14	15	16	17	18	19	20
Story	35	39	44	48	54	57	62	67	72	78

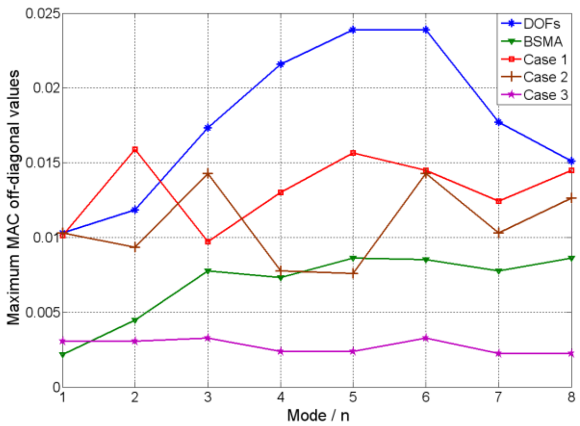


Fig. 5 Maximum MAC off-diagonal value in each of the modes

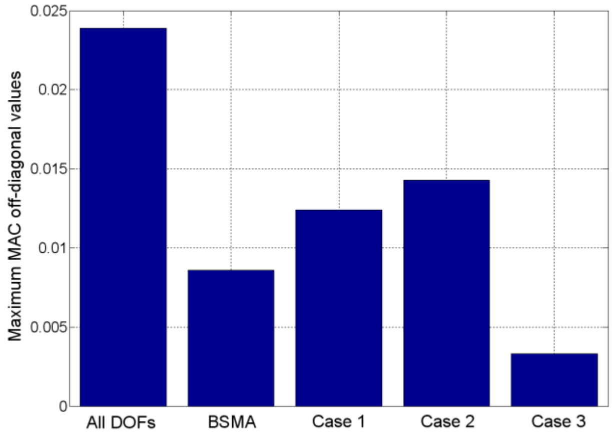


Fig. 6 Maximum MAC off-diagonal value in all of the modes

5. Conclusions

Considering the characteristics of the OSP techniques in the large-scale civil structure, this paper outlines a novel and efficient methodology called the DMA for the optimal design of SHM system sensor arrays. With the case analysis, some conclusions and recommendations are summarized as follows:

(1) The OSP is a discrete combinatory integer problem that means the original MA cannot be implemented directly. In order to overcome this difficulty, the dual-structure coding method, which uses a ordered pair to stand for the possible solutions of each monkey, is designed and adopted skillfully in the DMA, and the corresponding implementation steps for coding method is also presented in details.

(2) In the proposed DMA, the single large population is partitioned into subsets and each subpopulation searches the space in different directions in parallel. The proposed method is easy in understanding as it is based on simple and logical principles. Advantages of the method include its computationally non-intensive nature compared with exhaustive search techniques found in the literature and the benefit of physical insight into the ranking and ultimate selection of sensor locations. This kind of distribution mode effectively solves the combinatorial optimization problem such as the OSP problem when the performance tradeoffs are not unbearable and when the number of combinations is too large to preclude enumeration.

(3) Although the SMA has proven its ability of finding near global regions within a reasonable time, it is comparatively poor at finding the precise optimum solution in the region which the algorithm converges to. To obtain a more robust optimization result, it is common to combine different search strategies trying to compensate deficiencies of the individual algorithms. In this study the efficiency of the SMA is improved by incorporation of another intelligent method, i.e., the HSA. This can be considered as an innovative type of hybridization of the MA that has not yet been explored in the literature.

(4) The proposed DMA is simple in concept, few in parameters, easy in implementation, good global search capability, imposes fewer mathematical requirements, and does not require initial value settings of the decision variables. Since there are not any precise recommendations for

tuning the DMA parameters in the literature, an empirical study to determine the impact of three important parameters of the algorithm on the solution quality and convergence behavior is performed. The parameter analysis indicates that the number of monkeys in each subpopulation is larger, much more time is needed for algorithm to find the optimal solution, but usually the higher quality is achieved; the large number of iterations in the climb process could cause the improvement of local optimum although the local optimum could be obtained without too much iteration; the larger interval of the somersault may skip the global optimal solution while the smaller will lead to a decrease in the solution quality.

(5) Numerical investigations presented herein clearly suggest that the proposed DMA outperforms the other conventional algorithms both in terms of generating optimal solutions as well as faster convergence, which is expected to be even more pronounced should it be used for other high dimensional optimization problems. This study clearly indicates that it is desirable to employ a sophisticated search technique like the proposed DMA in order to optimally locate a small number of sensor locations from a large possible candidate sensor set.

Acknowledgements

This research work was jointly supported by the National Natural Science Foundation of China (Grant No. 51478081, 51421064, 51222806), the Science Fund for Distinguished Young Scholars of Dalian (2014J11JH125), and the Research Fund of State Key Laboratory of Coastal and Offshore Engineering (Grant No. SL2012-6).

References

Abdullah, M.M., Richardson, A. and Hanif, J. (2001), "Placement of sensors / actuators on civil structures using genetic algorithms", *Earthq. Eng. Struct. D.*., **30**(8), 1167-1184.

Brooks, R.R., Iyengar, S.S. and Chen, J. (1996), "Automatic correlation and calibration of noisy sensor readings using elite genetic algorithms", *Artifi. Intell.*, **84**(1-2), 339-354.

Carne, T. G. and Dohmann, C.R. (1995), "A modal test design strategy for modal correlation", *Proceedings of the 13th International Modal Analysis Conference*, Schenectady, New York, USA, February.

Chow, H.M., Lam, H.F., Yin, T. and Au, S.K. (2010), "Optimal sensor configuration of a typical transmission tower for the purpose of structural model updating", *Struct. Control Health.*, **18**(3), 305-320.

ETABS, *Computer & Structure*s, Inc. Berkeley, CA (USA), http://www.csiberkeley.com.

Fidanova, S., Marinov, P. and Alba, E. (2012), "Ant algorithm for optimal sensor deployment", *Comput. Intell.*, **399**, 21-29.

Geem, Z., Kim, J. and Loganathan, G. (2001), "A new heuristic optimization algorithm: harmony search", *Simulation*, **76**(2), 60-68.

Gou, X. K. and Cui, M.Y. (2008), "Application of genetic and simulated annealing algorithms in placement optimization of sensor / actuator", *Machinery Electronics*, **11**, 39-41.

Holland, J.H. (1975), *Adaption in natural and artificial systems*, Ann Arbor, University of Michigan Press.

Huang, M.S., Li, J. and Zhu, H.P. (2010), "Optimal sensor layout for bridge health monitoring based on dual-structure coding genetic algorithm", *Proceedings of the 2nd International Conference on Computational Intelligence and Software Engineering*, Wuhan, China, December.

Huang, W.P., Liu, J. and Li, H.J. (2005), "Optimal sensor placement based on genetic algorithms", *Eng. Mech.*, **22**(1), 113-117.

Hwang, S.F. and He, R.S. (2006), "A hybrid real-parameter genetic algorithm for function optimization",

Adv. Eng.Inform., **20**(1), 7-21.
Jia, J., Chen, J., Chang, G.R. and Tan, Z.H. (2009), "Energy efficient coverage control in wireless sensor networks based on multi-objective genetic algorithm", *Comput. Math. Appl.*, **57**(11-12), 1756-1766.
Kukunuru, N., Thella, B.R. and Davuluri, R.L. (2010), "Sensor deployment using particle swarm optimization", *Int. J. Eng. Sci. Tech*, **2**(10), 5395-5401.
Lan, Y.F., Zhao, R.Q. and Tang, W.S. (2011), "Minimum risk criterion for uncertain production planning problems", *Comput. Ind. Eng.*, **61**(3), 591-599.
Lei, Y., He, M.Y., Liu, C. and Lin, S.Z. (2013a), "Identification of tall shear buildings under unknown seismic excitation with limited output measurements", *Adv. Struct. Eng.*, **16**(11), 1839-1849.
Lei, Y., Wang, H.F. and Shen, W.A. (2012b), "Update the finite element model of Canton Tower based on direct matrix updating with incomplete modal data", *Smart Struct. Syst.*, **10**(4-5), 471-483.
Lei, Y., Wu, D.T. and Lin, S.Z. (2013b), "Integration of decentralized structural control and the identification of unknown inputs for tall shear building models under unknown earthquake excitation", *Eng. Struct.*, **52**, 306-316
Lei, Y., Wu, D.T. and Lin, Y. (2012a), "A decentralized control algorithm for large-scale building structures", *Comput.- Aided Civil. In.*, **27**(1), 2-13.
Li, H.N., Yi, T.H., Jing, Q.Y., Huo, L.S. and Wang, G.X. (2012), "Wind-induced vibration control of Dalian international trade mansion by tuned liquid dampers", *Math. Probl. Eng.*, Article ID 848031, 1-20.
Lin, X.K., Zhang, L.M., Guo, Q.T. and Zhao, X.P. (2009), "Application of coevolutionary genetic algorithm in optimal sensor placement", *J.Vib. Shock*, **28**(3), 190-194.
Liu, W., Hou, Z.K., and Demetriou, M.A. (2006), "A computational scheme for the optimal sensor / actuator placement of flexible structures using spatial H_2 measures", *Mech. Syst. Signal Pr.*, **20**(4), 881-895.
MATLAB, *The MathWorks,* Inc. Natwick, MA (USA), http://www.mathworks.com.
Maul, W.A., Kopasakis, G., Santi, L.M., Sowers, T.S. and Chicatelli, M. (2007), "Sensor selection and optimization for health assessment of aerospace systems", *Proceedings of the AIAA Infotech Conference and Exhibit*, Rohnert Park, CA, USA, May.
Ngatchou, P.N., Fox, W.L.J. and El-Sharkawi, M.A. (2005), "Distributed sensor placement with sequential particle swarm optimization", *Proceedings of the IEEE Swarm Intelligence Symposium*, CA, USA, June.
Roy, T. and Chakraborty, D. (2009), "Optimal vibration control of smart fiber reinforced composite shell structures using improved genetic algorithm", *J. Sound Vib.*, **319**(1-2), 15-40.
Stephan, C. (2012), "Sensor placement for modal identification", *Mech. Syst. Signal Pr.*, **27**, 461-470.
Wang, J.R, Yu, Y.X. and Zeng, Y. (2010), "Discrete monkey algorithm and its application in transmission network expansion planning", *J. Tianjin Univ.*, **43**(9), 798-803.
Wenzel, H. (2009), *Health monitoring of bridges*, USA, John Wiley and Sons Ltd.
Yang, Y., Gu, Z.Q. and Hu, L. (2007), "The application of virus evolutionary genetic algorithm to dynamic path planning", *Auto. Eng.*, **29**(1), 67-70.
Yao, L., Sethares, W.A. and Kammer, D.C. (1993), "Sensor placement for on orbit modal identification via a genetic algorithm", *AIAA J.*, **31**(10), 1167-1169.
Yong, L.Q. (2011), "Advances in harmony search algorithm", *Comput. Syst. Appl.*, **20**(7), 244-248.
Yi, T.H., Li, H.N. and Gu, M. (2011a), "Optimal sensor placement for health monitoring of high-rise structure based on genetic algorithm", *Math. Probl. Eng.*, Article ID 395101, 1-11.
Yi, T.H., Li, H.N. and Gu, M. (2011b), "Optimal sensor placement for structural health monitoring based on multiple optimization strategies", *Struct. Des. Tall Spec.*, **20**(7), 881-900.
Yi, T.H., Li, H.N. and Zhang, X.D. (2012a), "A modified monkey algorithm for optimal sensor placement in structural health monitoring", *Smart Mater. Struct.*, **21**(10), 1-9.
Yi, T.H., Li, H. N., and Zhang, X. D. (2012b), "Sensor placement on Canton Tower for health monitoring using asynchronous-climbing monkey algorithm", *Smart Mater. Struct.*, **21**(12), 1-13.
Zhao, R. Q., and Tang, W. S. (2008), "Monkey algorithm for global numerical optimization", *J. Uncer. Syst.*, **2**(3), 165-176.