**Advances in Computational Design**, *Vol. 1, No. 4 (2016) 315-327*
DOI: http://dx.doi.org/10.12989/acd.2016.1.4.315


# Structural health monitoring through meta-heuristics - comparative performance study

Nantiwat Pholdee[a] and Sujin Bureerat[*]

*Sustainable and Infrastructure Research and Development Center, Department of Mechanical Engineering, Faculty of Engineering, Khon Kaen University, Thailand*

*(Received June 9, 2016, Revised July 29, 2016, Accepted August 18, 2016)*

**Abstract.** Damage detection and localisation in structures is essential since it can be a means for preventive maintenance of those structures under service conditions. The use of structural modal data for detecting the damage is one of the most efficient methods. This paper presents comparative performance of various state-of-the-art meta-heuristics for use in structural damage detection based on changes in modal data. The meta-heuristics include differential evolution (DE), artificial bee colony algorithm (ABC), real-code ant colony optimisation (ACOR), charged system search (ChSS), league championship algorithm (LCA), simulated annealing (SA), particle swarm optimisation (PSO), evolution strategies (ES), teaching-learning-based optimisation (TLBO), adaptive differential evolution (JADE), evolution strategy with covariance matrix adaptation (CMAES), success-history based adaptive differential evolution (SHADE) and SHADE with linear population size reduction (L-SHADE). Three truss structures are used to pose several test problems for structural damage detection. The meta-heuristics are then used to solve the test problems treated as optimisation problems. Comparative performance is carried out where the statistically best algorithms are identified.

**Keywords:** structural health monitoring; meta-heuristics; modal data; damage detection

## 1. Introduction

Damage in structures can be caused by several reasons e.g. defects in structural elements, cracks from fatigue, and wears. Such a phenomenon always shortens structural service life. Over the years, many techniques have been developed to prevent this undesirable occurrence. Some causes may be able to detect by using visual inspection. Cracks inside the structures may be found by using X-ray scan. However, assessment of damage with one effective type of measurement is more wanted since it reduces time and cost while increasing reliability. This is often called structural health monitoring, usually referring to the use of a non-destructive testing technique for predicting the damage of structures under service loadings. Structural health monitoring can have three steps as identifying the presence of structural damage, localising the damage, and predicting its severity (Sinou 2009).

*Corresponding author, Professor, E-mail: Sujbur@kku.ac.th
[a]Ph.D., E-mail: nantiwat@kku.ac.th


http://www.techno-press.com/journals/acd&subpage=7 ISSN: 2383-8477 (Print), 2466-0523 (Online)

One of the most popular and efficient ways to perform damage detection in structures is using changes in structural model information e.g., natural frequencies, mode shapes, and damping ratios. The direct use of the modal data has been presented by many researchers (Lifshitz and Rotem 1969, Hearn and Testa 1991, Messina *et al*. 1998, Koh and Dyke 2007). The use of soft computing techniques has also been invented such as fuzzy logic systems (Buchholz *et al*. 2007, Agarwalla *et al*. 2015, Jiao *et al*. 2015) and neural networks (Abdeljaber and Avci 2016, Alavi *et al*. 2016, Sidibe *et al*. 2016). In recent years, meta-heuristics (MHs) have been introduced to solve a structural damage detection inverse problem. The problem is treated to be an optimisation problem where structural natural frequencies and mode shapes play a vital role. Several problems were formulated and solved using a genetic algorithm (Chou and Ghaboussi 2001), particle swarm optimisation (Pal and Banerjee 2015), an artificial bee colony algorithm (Xu *et al*. 2015, Ding *et al*. 2016), a differential evolution algorithm (Fu and Yu 2014), an evolutionary strategy (Jafarkhani and Masri 2011), an artificial immune algorithm (Chen and Zang 2009), an ant colony optimization algorithm (Majumdar *et al*. 2012) and a charged system search algorithm (Kaveh and Zolghadr 2015). Nevertheless, in the field of meta-heuristics, there have been recently many state-of-the-art algorithms that have not been tested with this problem.

Therefore, this paper is an attempt to bridge the gap between two research fields, one who employs MHs for structural damage detection and another who mainly focuses on developing MH algorithms. Three truss structures are used as numerical test problems to assess the performance of a number of MHs for solving structural damage detection and localisation. The results obtained are compared and discussed.

## 2. Formulation of damage detection based on natural frequency

In this study, a truss structure is modelled as a linear dynamic system. Damage of the structure can be identified by detecting variation of its natural frequency in any mode. The natural frequency can be calculated from an eigenvalue problem

$$[\mathbf{K}]\{\phi_j\} - \lambda_j[\mathbf{M}]\{\phi_j\} = 0 \tag{1}$$

where [**K**] is a structural stiffness matrix which is expressed as the summation or assembly of element stiffness matrices [$\mathbf{k}_e$]

$$[\mathbf{K}] = \sum_{i=1}^{n_e}[\mathbf{k}_e] \tag{2}$$

where $i$ and $n_e$ are the $i^{th}$ element and the total number of elements. [**M**] is a structural mass matrix obtained in a similar fashion as with the stiffness matrix. $\{\phi_j\}$ and $\lambda_j$ are the mode normalized eigenvector and eigenvalue respectively. The natural frequency of any mode shape ($\omega_j$) can be calculated by

$$\omega_j = \sqrt{\lambda_j} \quad \text{j=1,2,3,…, n} \tag{3}$$

where $j$ is the $j^{th}$ mode shape.

The damage on structural elements is assumed to affect the structural stiffness matrix and consequently alter structural natural frequencies. The stiffness matrix of the damaged structure, denoted by $[\mathbf{K}_d]$, can be calculated as

$$[\mathbf{K}_d] = \sum_{i=1}^{n_e} \frac{100 - p_i}{100} [\mathbf{k}_e] \tag{4}$$

where $p_i$ is percentage of damage in the $i^{th}$ element. The mode shapes and natural frequencies of the damaged structure can be calculated by solving Eqs. (1) and (3) after replacing $[\mathbf{K}]$ by $[\mathbf{K}_d]$. It should be note that only reduction of the element stiffness is considered in this study. The mass matrix of the structure is not considered to change.

To identify the damage of the structural elements, an optimisation problem is formulated to find the optimum solution for the percentages of damage on each element ($p_i$) whilst minimising the root mean square error (RMSE) between natural frequencies measured from the damaged structure and those from solving (1) - (4) with the given $p_i$. The objective function is then defined as follows

$$\text{Min}: f(\mathbf{x}) = \sqrt{\frac{\sum_{j=1}^{n_{mode}} \left(\omega_{j,damage} - \omega_{j,computed}\right)^2}{n_{mode}}} \tag{5}$$

where $\omega_{j,damage}$ and $\omega_{j,computed}$ are the structural natural frequency of mode $j$ obtained from a damaged structure and that from solving (1) - (4) (with the independent variables $\mathbf{x}=\{p_1, \ldots, p_{\mathrm{nele}}\}^T$) respectively. The values of $p_i$ determine the location of damaged elements. Note that only 6 vibration modes are used for calculation. In practice, the first $n_{mode}$ mode shapes and natural frequencies of the normal structure should be measured and used to dynamically update the finite element model in Eq. (1) (provided that the measured data is reliable). Once damage on the structure takes place, the natural frequencies are altered which can be detected by measuring the structure in a scheduled period or online. The changed natural frequencies are used for calculation in (5).

## 3. Test problems with trusses

To investigate the performance of various MHs on solving the optimisation problem of damage detection of truss structures, three truss structures are used in this study. For simplicity, structural damage is simulated while the natural frequencies of the undamaged and damaged structures are obtained from finite element analysis rather than performing experimental modal analysis of real structures.

### *3.1 Nine-bar truss*

The structure is shown in Fig. 1 (Majumdar *et al*. 2012). The cross sections of all bar elements are set to be 0.0025 $m^2$. Material density and modulus of elasticity are 7,850 kg/$m^3$ and 200 GPa, respectively. Two case studies are simulated for the 9 bar truss: Case I 50% damage at element number 2, and Case II 50% damage at element number 2 and 25% damage at element number 9. The data of natural frequencies of the undamaged and damaged 9-bar structures are shown in Table 1.

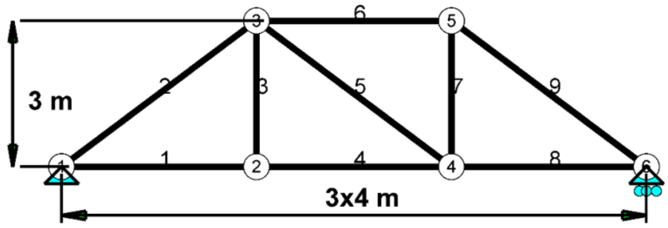


Fig. 1 Nine bar truss

Table 1 Natural frequencies (Hz) of damaged and undamaged of 9-bar structure

| Mode | Undamaged | 50 %damage at element number 2 | 50 %damage at element number 2 and 25 %damage at element number 9 |
|---|---|---|---|
| 1 | 38.3606 | 36.0103 | 35.0257 |
| 2 | 74.5226 | 66.3895 | 66.2781 |
| 3 | 117.8257 | 104.8556 | 101.6319 |
| 4 | 198.0133 | 194.2126 | 188.2125 |
| 5 | 260.1367 | 256.4372 | 255.9724 |
| 6 | 334.7825 | 334.7771 | 334.7585 |

### *3.2 Twenty-five-bar truss*

The structure is shown in Fig. 2 (Majumdar *et al*. 2012). The cross sections of all bar elements are set to be 6.4165 mm$^2$ .Material density and modulus of elasticity are 7,850 kg/m$^3$ and 200 GPa, respectively. Two case studies are simulated for the 25 bar truss :Case I 35% damage at element number 7, and Case II 35% damage at element number 7 and 40% damage at element number 9 . The data of natural frequencies of the undamaged and damaged 25-bar truss structures are shown in Table 2.

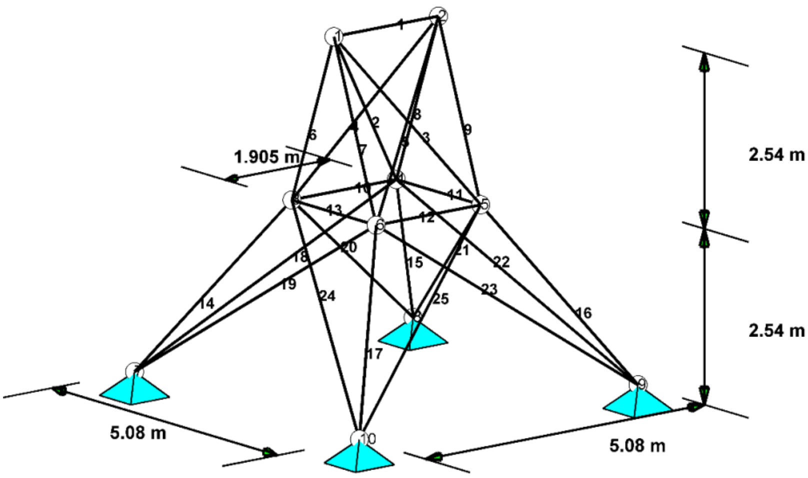


Fig. 2 Twenty-five bar truss

Table 2 Natural frequencies (Hz) of damaged and undamaged of 25 bar structure

| Mode | Undamaged | 35% damage at element number 7 | 35% damage at element number 7 and 40% damage at element number 9 |
|---|---|---|---|
| 1 | 69.7818 | 69.1393 | 68.5203 |
| 2 | 72.8217 | 72.2006 | 71.3167 |
| 3 | 95.8756 | 95.3372 | 94.5625 |
| 4 | 120.1437 | 119.8852 | 119.6514 |
| 5 | 121.5017 | 121.4774 | 121.4253 |
| 6 | 125.0132 | 125.0130 | 125.0129 |

### *3.3 Seventy-two-bar truss*

The structure is shown in Fig. 3 (Kaveh and Zolghadr 2015) .Four non-structural masses of 2270 kg are attached to the top nodes. The cross sections of all bar elements are set to be 0.0025 $m^2$. Material density and modulus of elasticity are 2,770 kg/$m^3$ and $6.98 \times 10^{10}$ Pa, respectively. Two case studies are simulated for the 72-bar truss :Case I, 15% damage at element number 55 (15% damage in element number 56, 57, or 58 results in the same set of natural frequencies), and Case II, 10% damage at element number 4 and 15% damage at element number 58 (90, 180, and 270 degrees rotation along the *z* axis lead to the same set of natural frequencies). The data of natural frequencies of the undamaged and damaged 72-bar truss structure are shown in Table 3.

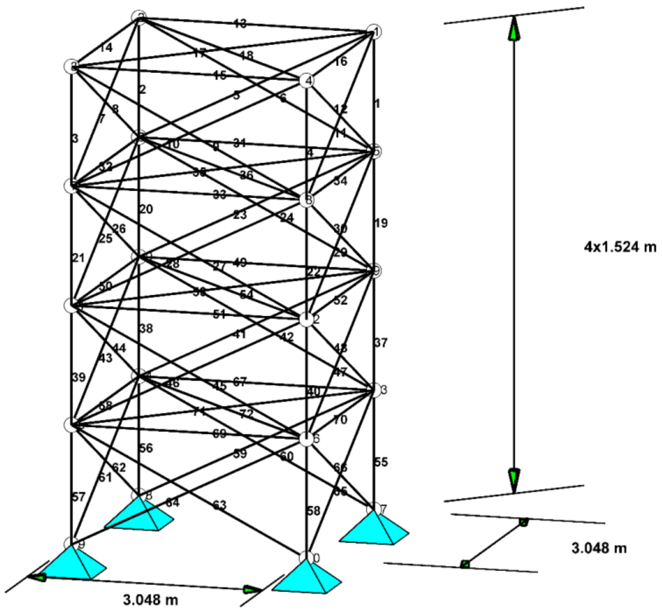


Fig. 3 Seventy-two bar truss

Table 3 Natural frequencies (Hz) of damaged and undamaged of 72 bar structure

| Mode | Undamaged | 15 %damage at element number 55 | 15 %damage at element number 58 and 10 %damage at element number 4 |
|---|---|---|---|
| 1 | 6.0455 | 5.9553 | 5.9530 |
| 2 | 6.0455 | 6.0455 | 6.0455 |
| 3 | 10.4764 | 10.4764 | 10.4764 |
| 4 | 18.2297 | 18.1448 | 18.0921 |
| 5 | 25.4939 | 25.4903 | 25.2437 |
| 6 | 25.4939 | 25.4939 | 25.4939 |

## 4. Numerical experiment

In this work, a comparative study of thirteen MHs search performance on solving the truss damage detection problems is conducted. Those methods are said to be established and some of them are considered the currently best optimisers. Given that $n_P$ is a population size, the MHs and their optimisation parameter settings used in this study (details of notations can be found in the corresponding references of each method) are detailed as:

• Differential evolution (DE) (Storn and Price 1997): a DE/best/2/bin strategy was used. A scaling factor, crossover rate and probability of choosing elements of mutant vectors are 0.5, 0.7, and 0.8 respectively.

• Artificial bee colony algorithm (ABC) (Karaboga and Basturk 2007): The number of food sources for employed bees is set to be $n_P/2$. A trial counter to discard a food source is 100.

• Real-code ant colony optimisation (ACOR) (Socha and Dorigo 2008): The parameter settings are $q$=0.2, and $\xi$=1.

• Charged system search (ChSS) (Kaveh and Talatahari 2010): The number of solutions in the charge memory is $0.2\times n_P$. The charged moving considering rate and the parameter *PAR* are set to be 0.75 and 0.5 respectively.

• League championship algorithm (LCA) (Husseinzadeh Kashan 2011): The probability of success $P_c$ and the decreasing rate to decrease $P_c$ are set to be 0.9999 and 0.9995, respectively.

• Simulated annealing (SA) (Bureerat and Limtragool 2008): Starting and ending temperatures are 10 and 0.001 respectively. For each loop, *nele* candidates are created by mutating on the current best solution while other *nele* candidates are created from mutating the current parent. The best of those 2nmode solutions are set as an offspring to be compared with the parent.

• Particle swarm optimisation (PSO) (Pal and Banerjee 2015): The starting inertia weight, ending inertia weight, cognitive learning factor, and social learning factor are assigned as 0.5, 0.01, 0.5 and 0.5 respectively.

• Evolution strategies (ES) (Back 1996): The algorithm uses a binary tournament selection operator and a simple mutation without the effect of rotation angles.

• Teaching-learning-based optimisation (TLBO) (Rao *et al*. 2011): Parameter settings are not required.

• Adaptive differential evolution (JADE) (Zhang and Sanderson 2009): The parameters are self-adapted during an optimisation process.

• Evolution strategy with covariance matrix adaptation (CMAES) (Hansen *et al*. 2003): The parameters are self-adapted during an optimisation process.

• Success-History Based Adaptive Differential Evolution (SHADE) (Tanabe and Fukunaga 2013): The parameters are self-adapted during an optimisation process.

• SHADE with Linear Population Size Reduction (L-SHADE) (Tanabe and Fukunaga 2014): The parameters are self-adapted during an optimisation process.

Each optimiser is used to solve each problem for 30 optimisation runs. The population size is set to be 20, 30 and 50 for the 9-bar, 25-bar and 72-bar trusses, respectively, whereas the number of iterations is 100 for all case studies. For the optimisers using different population sizes, their search processes are terminated with the total number of functions evaluations (FEs) equal to 20×100, 30×100 and 50×100 for the 9-bar, 25-bar and 72-bar trusses, respectively. Another termination criterion is when one of the members in the current population has an objective function value of $1\times10^{-3}$ or lower. It should be noted that the total FEs used in this study can be considered insufficient for some meta-heuristics according to the literature; nevertheless, this value is set so as to look for really powerful algorithms.

## 5. Results and discussions

After solving the truss damage detection problems for 30 optimisation runs, the results are illustrated in Tables 4-9. The mean and standard deviation (STD) values of the objective function are used to measure the search convergence and consistency of the algorithms.

Table 4 Results for 9 bar truss with 50% damage at element 2

| Optimisers | Objective function Values | | | | No .of successful runs from 30 runs | Mean of FEs |
|---|---|---|---|---|---|---|
| | Mean | STD | Min | Max | | |
| DE | 0.0008 | 0.0002 | 0.0004 | 0.0013 | 29 | 1366 |
| ABC | 0.0712 | 0.0540 | 0.0045 | 0.2113 | 0 | 2000 |
| ACOR | 0.0616 | 0.0296 | 0.0132 | 0.1250 | 0 | 2000 |
| ChSS | 0.1547 | 0.0780 | 0.0020 | 0.3110 | 0 | 2000 |
| LCA | 0.3197 | 0.1260 | 0.0590 | 0.6193 | 0 | 2000 |
| SA | 0.1069 | 0.0759 | 0.0023 | 0.2369 | 0 | 2000 |
| TLBO | 0.0205 | 0.0341 | 0.0001 | 0.1267 | 17 | 1516 |
| CMAES | 0.0020 | 0.0050 | 0.0005 | 0.0250 | 28 | 1479 |
| ES | 0.1178 | 0.1122 | 0.0004 | 0.4380 | 6 | 1782 |
| PSO | 0.9249 | 0.3183 | 0.0142 | 1.4527 | 0 | 2000 |
| JADE | 0.0094 | 0.0117 | 0.0019 | 0.0604 | 0 | 2000 |
| SHADE | 0.0030 | 0.0044 | 0.0003 | 0.0194 | 14 | 1890 |
| LSHADE | 0.0008 | 0.0002 | 0.0001 | 0.0016 | 29 | 903 |

Table 5 Results for 9 bar truss with 50 %damage at element 2 and 25 %damage at element number 9

| Optimisers | Objective function Values | | | | No. of successful runs from 30 runs | Mean of FEs |
|---|---|---|---|---|---|---|
| | Mean | STD | Min | Max | | |
| DE | 0.0008 | 0.0003 | 0.0003 | 0.0017 | 28 | 1435 |
| ABC | 0.0493 | 0.0300 | 0.0020 | 0.1214 | 0 | 2000 |
| ACOR | 0.1036 | 0.0476 | 0.0246 | 0.1916 | 0 | 2000 |
| ChSS | 0.0197 | 0.0217 | 0.0009 | 0.0914 | 2 | 1989 |
| LCA | 0.1347 | 0.0902 | 0.0156 | 0.3823 | 0 | 2000 |
| SA | 0.1159 | 0.1097 | 0.0026 | 0.4351 | 0 | 2000 |
| TLBO | 0.0224 | 0.1190 | 0.0003 | 0.6526 | 29 | 1267 |
| CMAES | 0.0008 | 0.0002 | 0.0003 | 0.0010 | 30 | 1282 |
| ES | 0.0055 | 0.0176 | 0.0002 | 0.0956 | 23 | 1167 |
| PSO | 0.4017 | 0.3391 | 0.0008 | 1.1933 | 2 | 1894 |
| JADE | 0.0114 | 0.0074 | 0.0010 | 0.0353 | 1 | 1974 |
| SHADE | 0.0037 | 0.0039 | 0.0005 | 0.0125 | 10 | 1941 |
| LSHADE | 0.0008 | 0.0002 | 0.0004 | 0.0011 | 29 | 1056 |

### *5.1 Nine-bar truss*

For the 9 bar truss with 5% damage at element 2, the results are illustrated in Table 4. The best performer based on the mean value of the objective function is DE and L-SHADE while the second best and the third best are CMAES and SHADE, respectably .Based on the STD of the objective function, the most consistent optimisers are DE and L-SHADE while the second best is SHADE .When looking at the number of successful runs (obtained objective function value lower than $1\times10^{-3}$), only six from thirteen optimisers including DE, TLBO, CMAES, ES, SHADE and L-SHADE can detect the true damage of the structure for, in that order, 29, 17, 28, 6, 14 and 29 times from totally 30 optimisation runs. The average numbers of function evaluations for convergence results of the five algorithms, DE, TLBO, CMAES, ES, SHADE and L-SHADE are 1366, 1516, 1479, 1782, 1890 and 903, respectively.

For the results of the 9-bar truss with 5% damage at element 2 and 25% damage at element number 9 in Table 5, the best performers based on the objective mean values are DE, L-SHADE and CMAES which obtained the same mean objective function values while the fourth best is SHADE. Based on STD, the best performers is CMAES and L-SHADE which obtained the same STD values while the third best is DE. For this case, four optimisers including ABC, ACOR, LCA and SA cannot detect the damage of the structure. The most efficient optimisers which can detect the damage of structure for 28, 29, 30 and 29 times from 30 runs with average numbers of FEs 1435, 1267, 1282 and 1056 are DE, TLBO, CMAES and L-SHADE respectively.

### *5.2 Twenty-five-bar truss*

For the 25-bar truss with 35 %damage at element 7, the results are given in Table 6 .The best performers based on the mean objective function values are TLBO and ES which obtained the same results while the third best methods are DE and L-SHADE which also obtained the same results. Based on the STD value. There are four optimisers which obtained the same best values including: DE, TLBO, ES and L-SHADE. When considering the number of successful runs, five optimisers including ABC, ACOR, ChSS, LCA and PSO cannot detect the damage of the structure. The most efficient optimisers which can detect the damage of the structure for 30 runs with average numbers of function evaluations of 1733, 1400, 1592 and 937 are DE, TLBO, ES and L-SHADE, respectively.

For the 25 bar truss with 35% damage at element 7 and 40 %damage at element number 9, the

Table 6 Results for 25 bar truss with 35% damage at element number 7

| Optimisers | Objective function Values | | | | No. of successful runs from 30 runs | Mean of FEs |
|---|---|---|---|---|---|---|
| | Mean | STD | Min | Max | | |
| DE | 0.0008 | 0.0002 | 0.0003 | 0.0010 | 30 | 1733 |
| ABC | 0.4780 | 0.5089 | 0.0096 | 2.1914 | 0 | 3000 |
| ACOR | 0.0078 | 0.0027 | 0.0041 | 0.0145 | 0 | 3000 |
| ChSS | 0.3992 | 0.1379 | 0.1766 | 0.7438 | 0 | 3000 |
| LCA | 4.2594 | 0.6234 | 2.8640 | 5.3094 | 0 | 3000 |
| SA | 0.0047 | 0.0058 | 0.0004 | 0.0208 | 10 | 2611 |
| TLBO | 0.0007 | 0.0002 | 0.0004 | 0.0010 | 30 | 1400 |
| CMAES | 0.0024 | 0.0024 | 0.0003 | 0.0080 | 18 | 2680 |
| ES | 0.0007 | 0.0002 | 0.0001 | 0.0010 | 30 | 1592 |
| PSO | 7.3405 | 1.1242 | 5.0301 | 9.5734 | 0 | 3000 |
| JADE | 0.0021 | 0.0009 | 0.0005 | 0.0039 | 2 | 2957 |
| SHADE | 0.0012 | 0.0005 | 0.0006 | 0.0025 | 14 | 2875 |
| LSHADE | 0.0008 | 0.0002 | 0.0003 | 0.0010 | 30 | 937 |

Table 7 Results for 25 bar truss with 35 %damage at element number 7 and 40% damage at element number 9

| Optimisers | Objective function Values | | | | No. of successful runs from 30 runs | Mean of FEs |
|---|---|---|---|---|---|---|
| | Mean | STD | Min | Max | | |
| DE | 0.0007 | 0.0002 | 0.0002 | 0.0010 | 30 | 2018 |
| ABC | 0.3272 | 0.2745 | 0.0368 | 1.0296 | 0 | 3000 |
| ACOR | 0.0176 | 0.0069 | 0.0019 | 0.0313 | 0 | 3000 |
| ChSS | 0.1111 | 0.0534 | 0.0414 | 0.2951 | 0 | 3000 |
| LCA | 3.5560 | 0.7799 | 1.5968 | 4.8099 | 0 | 3000 |
| SA | 0.0069 | 0.0051 | 0.0012 | 0.0230 | 0 | 3000 |
| TLBO | 0.0019 | 0.0044 | 0.0003 | 0.0236 | 26 | 1910 |
| CMAES | 0.0060 | 0.0056 | 0.0006 | 0.0273 | 5 | 2938 |
| ES | 0.0029 | 0.0120 | 0.0003 | 0.0667 | 29 | 1742 |
| PSO | 6.5441 | 1.1249 | 3.2461 | 8.3184 | 0 | 3000 |
| JADE | 0.0039 | 0.0021 | 0.0005 | 0.0094 | 1 | 2993 |
| SHADE | 0.0024 | 0.0009 | 0.0008 | 0.0050 | 2 | 2959 |
| LSHADE | 0.0008 | 0.0002 | 0.0003 | 0.0010 | 30 | 948 |

results are illustrated in Table 7. The best performer based on mean values is DE while the second best and the third best are L-SHADE and TLBO, respectably. Based on the STD values, the best performer is DE and L-SHADE which obtained the same results while the third best is SHADE. When examining the number of successful runs, seven from thirteen optimisers including DE, TLBO, CMAES, ES, JADE, SHADE and L-SHADE can detect the damage of the structure for 30, 26, 5, 29, 1, 2 and 30 runs, respectively. The average numbers of function evaluations for convergence results of the seven algorithms, DE, TLBO, CMAES, ES, JADE, SHADE and L-SHADE are 2018, 1910, 2938, 1742, 2993, 2959, and 948 in that order.

Overall, it was found that the most efficient optimiser for damage detection of the 25-bar truss for both simulation cases is L-SHADE.

### *5.3 Seventy-two-bar truss*

For the 72-bar truss with 15% damage at element 5, the results are given in Table 8 .The best and the second best performers based on both mean and STD of objective function values are TLBO and L-SHADE, respectively. When looking at the number of successful runs (*f* reaching $1\times10^{-3}$ or lower), only three method including DE, TLBO and L-SHADE can detect the damage of the structure for 3, 29, and 6 times from totally 30 optimisation runs .The average numbers of function evaluations for convergence results of the three algorithms, DE, TLBO, and L-SHADE are 4893,2722, and 4868, in that order.

For the 72 bar truss with 15% damage at element number 58 and 10% damage at element number 4, the results are given in Table 9 .The best and second best performer based on both mean and STD of objective function values are TLBO and L-SHADE, respectively. Only three optimisers including DE, TLBO and L-SHADE can detect the damage of the structure for 5, 27 and 5 times from totally 30 optimisation runs .The average numbers of function evaluations for convergence results of the three algorithms, DE, TLBO, and L-SHADE are 4934, 3036, and 4856,

Table 8 Results for 72 bar truss with 15% damage at element number 55

| Optimisers | Objective function Values | | | | No. of successful runs from 30 runs | Mean of FEs |
|---|---|---|---|---|---|---|
| | Mean | STD | Min | Max | | |
| DE | 0.0061 | 0.0151 | 0.0009 | 0.0837 | 3 | 4893 |
| ABC | 0.6413 | 0.0842 | 0.4711 | 0.7926 | 0 | 5000 |
| ACOR | 0.0111 | 0.0022 | 0.0077 | 0.0161 | 0 | 5000 |
| ChSS | 0.2317 | 0.0237 | 0.1783 | 0.2710 | 0 | 5000 |
| LCA | 0.9111 | 0.0344 | 0.8592 | 0.9695 | 0 | 5000 |
| SA | 0.1094 | 0.0218 | 0.0740 | 0.1466 | 0 | 5000 |
| TLBO | 0.0008 | 0.0002 | 0.0002 | 0.0011 | 29 | 2722 |
| CMAES | 0.0047 | 0.0013 | 0.0025 | 0.0087 | 0 | 5000 |
| ES | 0.0043 | 0.0013 | 0.0016 | 0.0066 | 0 | 5000 |
| PSO | 0.8870 | 0.0631 | 0.7702 | 0.9959 | 0 | 5000 |
| JADE | 0.0200 | 0.0026 | 0.0155 | 0.0247 | 0 | 5000 |
| SHADE | 0.0076 | 0.0014 | 0.0051 | 0.0111 | 0 | 5000 |
| LSHADE | 0.0018 | 0.0012 | 0.0010 | 0.0059 | 6 | 4868 |

Table 9 Results for 72 bar truss with 15% damage at element number 58 and 10% damage at element number 4

| Optimisers | Objective function Values | | | | No. of successful runs from 30 runs | Mean of FEs |
|---|---|---|---|---|---|---|
| | Mean | STD | Min | Max | | |
| DE | 0.0063 | 0.0152 | 0.0009 | 0.0857 | 5 | 4934 |
| ABC | 0.6506 | 0.0780 | 0.4437 | 0.8062 | 0 | 5000 |
| ACOR | 0.0104 | 0.0019 | 0.0074 | 0.0156 | 0 | 5000 |
| ChSS | 0.2318 | 0.0172 | 0.2042 | 0.2691 | 0 | 5000 |
| LCA | 0.8691 | 0.0483 | 0.7418 | 0.9447 | 0 | 5000 |
| SA | 0.1094 | 0.0226 | 0.0659 | 0.1491 | 0 | 5000 |
| TLBO | 0.0009 | 0.0001 | 0.0006 | 0.0014 | 27 | 3036 |
| CMAES | 0.0044 | 0.0014 | 0.0022 | 0.0085 | 0 | 5000 |
| ES | 0.0044 | 0.0013 | 0.0024 | 0.0084 | 0 | 5000 |
| PSO | 0.9015 | 0.0621 | 0.7368 | 1.0140 | 0 | 5000 |
| JADE | 0.0207 | 0.0031 | 0.0131 | 0.0262 | 0 | 5000 |
| SHADE | 0.0079 | 0.0015 | 0.0058 | 0.0119 | 0 | 5000 |
| LSHADE | 0.0021 | 0.0011 | 0.0009 | 0.0051 | 5 | 4856 |

respectively.

Overall, TLBO is the best performer for the 72-bar truss problem which is a large scale problem.

The comparative results show that the overall best optimizer are DE, TLBO and L-SHADE. For the small-scale 9-bar truss and the medium-scale 25 bar truss, the overall best optimisers are L-SHADE and DE. For the large-scale 72-bar truss, TLBO is the best performer. The maximum number of function evaluations assigned for the 72-bar truss problems is not sufficient but it is enough to show the best performers. TLBO is good at solving a large-scale problem as demonstrated in (Pholdee *et al*. 2015).

## 6. Conclusions

The various meta-heuristic optimisers were tested for the problems of damage detection in trusses. The damage detection problems are based on vibration measurement, which can be treated as an inverse optimisation problem. The comparative results reveal that the currently top meta-heuristics DE, TLBO and L-SHADE are the overall best method where TLBO is outstanding for a large-scale problem. The results obtained from TLBO can be used as the baseline for future investigation of structural damage detection using meta-heuristics.

## Acknowledgments

The authors are grateful for the support from the Thailand Research Fund (TRF).

## References


Abdeljaber, O. and Avci, O. (2016), "Nonparametric structural damage detection algorithm for ambient vibration response: utilizing artificial neural networks and self-organizing maps", *J. Architec. Eng.*, **22**(2), 04016004.

Agarwalla, D.K., Dash, A.K., Bhuyan, S.K. and Nayak, P.S.K. (2015), "Damage detection of fixed-fixed beam: a fuzzy neuro hybrid system based approach", *Proceedings of the 5th International Conference on Swarm, Evolutionary, and Memetic Computing*, SEMCCO 2014, Bhubaneswar, India, December 18-20.

Alavi, A.H., Hasni, H., Lajnef, N., Chatti, K. and Faridazar, F. (2016), "An intelligent structural damage detection approach based on self-powered wireless sensor data", *Auto. Constr.*, **62**, 24-44.

Back, T. (1996), *Evolutionary Algorithms in Theory and Practice*, Oxford: Oxford University Press.

Buchholz, M., Pecheur, G., Niemeyer, J. and Krebs, V. (2007), "Fault detection and isolation for PEM fuel cell stacks using fuzzy cluster", *Proceedings of the Control Conference (ECC)*, 2007 European.

Bureerat, S. and Limtragool, J. (2008), "Structural topology optimisation using simulated annealing with multiresolution design variables", *Finite Elements Anal. Des.*, **44**(12-13), 738-747.

Chen, B. and Zang, C. (2009), "Artificial immune pattern recognition for structure damage classification", *Comput. Struct.*, **87**(21-22), 1394-1407.

Chou, J.-H. and Ghaboussi, J. (2001), "Genetic algorithm in structural damage detection", *Comput. Struct.*, **79**(14), 1335-1353.

Ding, Z.H., Huang, M. and Lu, Z.R. (2016), "Structural damage detection using artificial bee colony algorithm with hybrid search strategy", *Swarm Evolut. Comput.*, **28**, 1-13.

Fu, Y.M. and Yu, L. (2014), "A DE-based algorithm for structural damage detection", *Adv. Mater. Res.*, **919**, 303-307.

Hansen, N., Muller, S.D. and Koumoutsakos, P. (2003), "Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES)", *Evol. Comput.*, **11**(1), 1-18.

Hearn, G. and Testa, R.B. (1991), "Modal analysis for damage detection in structures", *J. Struct. Eng.*, **117**(10), 3042-3063.

Husseinzadeh Kashan, A. (2011), "An efficient algorithm for constrained global optimization and application to mechanical engineering design: League Championship Algorithm (LCA)", *Comput. Aid. Des.*, **43**(12), 1769-1792.

Jafarkhani, R. and Masri, S.F. (2011), "Finite element model updating using evolutionary strategy for damage detection", *Comput. Aid. Civ. Infrastruct. Eng.*, **26**(3), 207-224.

Jiao, Y.-B., Liu, H.-B., Cheng, Y.-C. and Gong, Y.-F. (2015), "Damage identification of bridge based on chebyshev polynomial fitting and fuzzy logic without considering baseline model parameters", *Shock Vib.*, **2015**, 187956.

Karaboga, D. and Basturk, B. (2007), "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm", *J. Global Optim.*, **39**(3), 459-471.

Kaveh, A. and Talatahari, S. (2010), "A novel heuristic optimization method: charged system search", *Acta Mech.*, **213**(3), 267-289.

Kaveh, A. and Zolghadr, A. (2015), "An improved CSS for damage detection of truss structures using changes in natural frequencies and mode shapes", *Adv. Eng. Softw.*, **80**, 93-100.

Koh, B.H. and Dyke, S.J. (2007), "Structural health monitoring for flexible bridge structures using correlation and sensitivity of modal data", *Comput. Struct.*, **85**(3-4), 117-130.

Lifshitz, J.M. and Rotem, A. (1969), "Determination of Reinforcement Unbonding of Composites by a Vibration Technique", *J. Compos. Mater.*, **3**(3), 412-423.

Majumdar, A., Maiti, D.K. and Maity, D. (2012), "Damage assessment of truss structures from changes in natural frequencies using ant colony optimization", *Appl. Math. Comput.*, **218**(19), 9759-9772.

Messina, A., Williams, E.J. and Contursi, T. (1998), "Structural damage detection by a sensitivity and statistical-based method", *J. Sound Vib.*, **216**(5), 791-808.

Pal, J. and Banerjee, S. (2015), "A combined modal strain energy and particle swarm optimization for health monitoring of structures", *J. Civ. Struct. Hlth. Monit.*, **5**(4), 353-363.

Pholdee, N., Park, W.-W., Kim, D.-K., Im, Y.-T., Bureerat, S., Kwon, H.-C. and Chun M.-S. (2015), "Efficient hybrid evolutionary algorithm for optimization of a strip coiling process", *Eng. Optimiz.*, **47**(4), 521-532.

Rao, R.V., Savsani, V.J. and Vakharia, D.P. (2011), "Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems", *Comput. Aid. Des.*, **43**(3), 303-315.

Sidibe, Y., Druaux, F., Lefebvre, D., Maze, G. and Léon, F. (2016), "Signal processing and Gaussian neural networks for the edge and damage detection in immersed metal plate-like structures", *Artif. Intel. Rev.*, in press.

Sinou, J.-J. (2009), "A review of damage detection and health monitoring of mechanical systems from changes in the measurement of linear and non-linear vibrations", Ed., R.C. Sapri, *Mechanical Vibrations: Measurement, Effects and Control*, Nova Science Publishers, Inc.

Socha, K. and Dorigo, M. (2008), "Ant colony optimization for continuous domains", *Eur. J. Oper. Res.*, **185**(3), 1155-1173.

Storn, R. and Price, K. (1997), "Differential Evolution - A Simple and Efficient Heuristic for global Optimization over Continuous Spaces", *J. Global Optim.*, **11**(4), 341-359.

Tanabe, R. and Fukunaga, A. (2013), "Evaluating the performance of SHADE on CEC 2013 benchmark problems", *IEEE Congress on Evolutionary Computation (CEC)*, Cancún, México, 20-23 June.

Tanabe, R. and Fukunaga, A.S. (2014), "Improving the search performance of SHADE using linear population size reduction", *IEEE Congress on Evolutionary Computation (CEC)*, Beijing, China, 6-11 July.

Xu, H., Ding, Z., Lu, Z. and Liu, J. (2015), "Structural damage detection based on Chaotic Artificial Bee Colony algorithm", *Struct. Eng. Mech.*, **55**(6), 1223-1239.

Zhang, J. and Sanderson, A.C. (2009), "JADE: adaptive differential evolution with optional external archive", *IEEE T. Evolut. Comput.*, **13**(5), 945-958.

*TK*