KSCE Journal of Civil Engineering (2018) 22(8):2982-2993

DOI 10.1007/s12205-017-0095-y

Structural Engineering

pISSN 1226-7988, eISSN 1976-3808
www.springer.com/12205

A Comparative Study of Eighteen Self-adaptive Metaheuristic Algorithms for Truss Sizing Optimisation

Nantiwat Pholdee* and Sujin Bureerat**

Received June 14, 2016/Accepted March 18, 2017/Published Online November 15, 2017

Abstract

Performance comparison of meta-heuristics (MHs) is conducted for truss sizing design. Six traditional truss sizing design problems with mass objective function subject to displacement and stress constraints were employed for performance test. The test problems have two types with and without including buckling constraints. Eighteen self-adaptive MHs from literature are employed to tackle the truss sizing problems. The results from implementing the self-adaptive MHs are compared in terms of convergence rate and consistency. It is found that for the test problem without buckling constraints, the top two optimisers according to the statistical Wilcoxon rank sum tests are Success-History Based Adaptive Differential Evolution with Linear Population Size Reduction (L-SHADE) and Success-History Based Adaptive Differential Evolution (SHADE) while the top two optimiser for the test problems with buckling constraints is L-SHADE and L-SHADE with Eigenvector-Based Crossover and Successful-Parent-Selecting Framework (SPS-L-SHADE-EIG). The buckling constraints are significantly important and should be included to truss design subjected to static loads.

Keywords: *meta-heuristics, self-adaptive algorithms, truss sizing optimisation, buckling constraints, performance comparison*

1. Introduction

Over the last few decades, needs for optimum design of truss structures tend to increase since a light weight structure which is safe under given loading conditions is desired. The truss optimisation problem is usually posed to find a set of design variables such as topology (Noilublao and Bureerat, 2011; Richardson, Adriaenssens *et al.*, 2012; Ahrari, Atai *et al.*, 2015; Li, 2014; Richardson, Coelho *et al*., 2016; Yang, Zhang *et al*., 2015; Yang, Zhang *et al*., 2015), shape (Gholizadeh and Barzegar, 2012; Miguel and Fadel Miguel, 2012; Kaveh and Javadi, 2014), sizes (Degertekin and Hayalioglu, 2013; Kaveh and Khayatazad, 2013; Camp and Farshchin, 2014; Flager, Adya *et al*., 2014; Kaveh *et al*., 2014; Bekdaş *et al*., 2015; Bureerat and Pholdee, 2015; Mortazavi and Toğan, 2017), combination of shape and sizes (Miguel and Fadel Miguel, 2012; Ahrari and Atai, 2013; Kaveh and Khayatazad, 2013; Pholdee and Bureerat, 2013; Pholdee and Bureerat, 2014a; Flager *et al*., 2014; Kaveh and Javadi, 2014), or simultaneously combining all three type of variables (Noilublao and Bureerat, 2011; Noilublao and Bureerat, 2013; Ahrari *et al*., 2015) of a structure in order to minimise structural mass while fulfilling strength (Camp and Farshchin, 2014; Flager *et al.*, 2014; Kaveh, Bakhshpoori *et al*., 2014; Kaveh *et al*., 2014; Bekdaş *et al*., 2015; Bureerat and Pholdee, 2015; Dede and Ayvaz, 2015) and/or vibration (Kaveh and Javadi, 2014; Kaveh and Zolghadr, 2014; Pholdee and Bureerat, 2014b; Kaveh and Ilchi Ghazaan, 2016; Kaveh and Ilchi Ghazaan, 2015; Kaveh and Mahdavi, 2015) constraints. Recently, research in sizing optimisation of truss structures can be classified into two groups based on types of structural analyses i.e. static or dynamic design. From literature, truss optimisation with static analysis is posed to minimise structural mass while nodal displacements and stresses on truss members are usually set as constraints (Ahrari and Atai, 2013; Degertekin and Hayalioglu, 2013; Camp and Farshchin, 2014; Flager *et al*., 2014; Kaveh *et al*., 2014; Kaveh *et al*., 2014; Bekdaş *et al.*, 2015; Bureerat and Pholdee, 2015; Dede and Ayvaz, 2015). For truss optimisation based on dynamic structural analysis, the objective function is to minimise structural mass while the constraints are natural frequencies of various modes (Gholizadeh and Barzegar, 2012; Miguel and Fadel Miguel, 2012; Kaveh and Javadi, 2014; Kaveh and Zolghadr, 2014; Pholdee and Bureerat, 2014b; Kaveh and Ilchi Ghazaan, 2016; Kaveh and Ilchi Ghazaan, 2015; Kaveh and Mahdavi, 2015). Both types of truss sizing design are known to have non-convex feasible regions (note that only statically indeterminate trusses (Hajela, 1990; Stolpe and Svanberg, 2001; Beck *et al*., 2010) may lead to non-convex feasible regions in cases of static loading). This means that some traditional gradient-based optimisers may

*Assistant Professor, Sustainable Infrastructure Research and Development Center, Dept. of Mechanical Engineering, Faculty of Engineering, Khon Kaen University, 40002, Thailand (E-mail: nantiwat@kku.ac.th)
**Professor, Sustainable Infrastructure Research and Development Center, Dept. of Mechanical Engineering, Faculty of Engineering, Khon Kaen University, 40002, Thailand (Corresponding Author, E-mail: sujbur@kku.ac.th)

struggle to converge. As a consequence, meta-heuristics (MHs) have become an alternative optimiser for this task (Kaveh and Talatahari, 2009; Gholizadeh and Barzegar, 2012; Kaveh and Khayatazad, 2013; Camp and Farshchin, 2014; Flager *et al*., 2014; Kaveh *et al*., 2014; Kaveh, Sheikholeslami *et al.*, 2014; Kaveh and Zolghadr, 2014; Bekdaş *et al*., 2015; Bureerat and Pholdee, 2015; Dede and Ayvaz, 2015; Kaveh and Mahdavi, 2015).

For the design based on static analysis, there are numerous MHs having been proposed and reported worldwide (Ahrari and Atai, 2013; Degertekin and Hayalioglu, 2013; Kaveh and Khayatazad, 2013; Camp and Farshchin, 2014; Flager *et al*., 2014; Kaveh *et al*., 2014; Kaveh *et al*., 2014; Bekdaş *et al*., 2015; Bureerat and Pholdee, 2015; Dede and Ayvaz, 2015; Richardson *et al.*, 2016; Yang *et al.*, 2015; Yang *et al*., 2015). From the literature, most of them only considered stress and displacement constraints in the test problems as can be seen in (Camp and Farshchin, 2014; Flager *et al*., 2014; Kaveh *et al.*, 2014; Bekdaş *et al*., 2015; Bureerat and Pholdee, 2015; Dede and Ayvaz, 2015). Essentially, stress and displacement are said to be inevitable mechanical phenomena for structural design particularly for those truss members under tension loading. However, it is very important to check for buckling instability of members under compression because, normally, there is no guarantee that the inclusion of only stress failure constraints is adequate for structural safety. Therefore, truss sizing optimisation with the inclusion of buckling constraints should be examined.

The field of meta-heuristics which is one fast growing soft computing area has caught considerable attention from a great many researchers over the last few decades. Their use for structural optimisation has been investigated worldwide (Camp and Farshchin, 2014; Flager *et al.*, 2014; Kaveh *et al.*, 2014; Kaveh and Javadi, 2014; Kaveh *et al.*, 2014; Kaveh and Zolghadr, 2014; Pholdee and Bureerat, 2014b; Bekdaş *et al.*, 2015; Bureerat and Pholdee, 2015; Dede and Ayvaz, 2015; Kaveh and Ilchi Ghazaan, 2015; Kaveh and Ilchi Ghazaan, 2016; Kaveh and Mahdavi, 2015; Hosseini *et al.*, 2016; Nariman, 2016; Sheikholeslami *et al.*, 2016). For truss optimisation, researchers tend to propose new or improved MHs and then compared their results with those optimisers that only appeared in the truss optimisation articles (Kaveh and Talatahari, 2009; Gholizadeh and Barzegar, 2012; Miguel and Fadel Miguel, 2012; Degertekin and Hayalioglu, 2013; Kaveh and Khayatazad, 2013; Kaveh and Javadi, 2014; Dede and Ayvaz, 2015). Nevertheless, there have been a great number of MHs proposed based on new search concepts (Baluja, 1994; Teh and Rangaiah, 2003; Rashedi *et al.*, 2009; Kaveh and Talatahari, 2010; Tan and Zhu, 2010; Yang and Deb, 2010; Husseinzadeh Kashan, 2011; Rao *et al*., 2011; Yang and Hossein Gandomi, 2012; Kaveh and Khayatazad, 2013; Kaveh *et al*., 2014; Bekdaş *et al*., 2015; Dede and Ayvaz, 2015; Muthiah-Nakarajan and Noel, 2016; Zhao *et al*., 2016), improvement of existing algorithms (Bureerat, 2011; Jia *et al.*, 2013; Baykasoğlu and Ozsoydan, 2014; Kaveh and Zolghadr, 2014; Talaei *et al.*, 2016; Meng and Pan, 2017), hybridisation (Kaveh and Talatahari, 2009; Kaveh *et al*., 2014; Kaveh and Javadi, 2014; Jung *et al.*, 2015; Kaveh and Ilchi Ghazaan, 2015; Kaveh and Mahdavi, 2015; Sheikholeslami *et al*., 2016; Zhao *et al*., 2016), and self-adaptive versions (Hansen *et al.*, 2003; Brest *et al*., 2006; Liang *et al*., 2006; García-Martínez *et al.*, 2008; Zhang and Sanderson, 2009; Mallipeddi *et al*., 2011; Yong *et al*., 2011; Wang *et al.*, 2012; Yong and Zixing, 2012; Tanabe and Fukunaga, 2013; Elsayed *et al.*, 2014; Lei *et al.*, 2014; Tanabe and Fukunaga, 2014; Lei *et al.*, 2015; Sallam *et al.*, 2015; Shu-Mei *et al*., 2015). Self-adaptive meta-heuristics (SAMHs) are optimisers that can automatically tune their optimisation parameters during an optimisation run. Compared with other group, they are superior since users will not worry much about the parameter setting. It is unfortunate that most of the aforementioned algorithms were developed independently and have yet to be tested for truss optimisation. Therefore, comparative performance of such meta-heuristic algorithms for truss optimisation is important since the real best algorithm could be identified and it could provide refreshed information for the researchers in this field.

As a result, this paper has an intention to somewhat bridge the gap from the truss optimisation and general meta-heuristics fields. Comparative performance of many self-adaptive MHs (mostly well-established) for truss sizing design optimisation is conducted. The objective function is set to minimise structural mass while stress, displacement and buckling are set to be constraints. The MHs are employed to find the optima of six traditional truss sizing optimisation design problems, which can be grouped into the design problems with and without buckling constraints. Each method is employed to solve a particular problem 30 times whereas the results received from implementing those methods are compared statistically. Some top performance meta-heuristics are then discovered.

The rest of this paper is organised as follows. Section 2 gives details of the truss design problems used in this study. Details of numerical test and optimisation parameter settings of the optimisers are given in Section 3. Section 4 displays the performance comparison of the 18 self-adaptive meta-heuristics for solving the six truss sizing design problems. Finally, conclusions are drawn in Section 5.

2. Test Problems for Truss Sizing

Six traditional truss mass minimisation problems with sizing design variables will be used in this study (Bureerat and Pholdee, 2015). In this work, buckling constraints were considered for all design problems. Therefore, the optimisation problems are set to minimise truss mass subjected to displacement, stress and buckling constraints detailed and expressed as follows:

2.1 Ten Bar Truss

The 10-bar truss is shonw in Fig. 1 (Degertekin, 2012; Degertekin and Hayalioglu, 2013; Camp and Farshchin, 2014; Bureerat and Pholdee, 2015). The structure is subjected to vertical (*y*-direction) applied forces P_1 and P_2 where the force P_1 is applied on nodes 2 and 4 and P_2 is acted on nodes 1 and 3. Two applied load

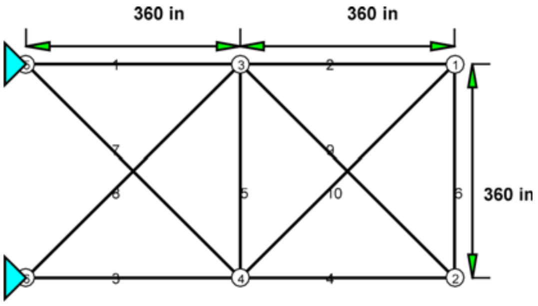

Fig. 1. Ten-bar Truss (Bureerat and Pholdee, 2015)

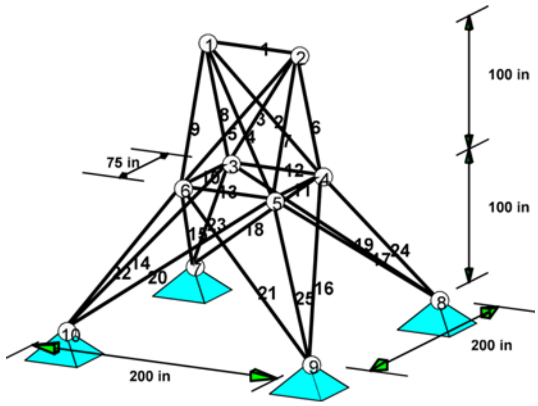

Fig. 2. Twenty-five Bar Space Truss (Bureerat and Pholdee, 2015)

conditions are used i.e. $P_1 = -100$ kips and $P_2 = 0$ kips for Case I, and $P_1 = -150$ kips and $P_2 = 50$ kips for Case II. The design variables comprise all cross-sectional areas of the truss members. Young modulus and density are respectively 10^4 ksi and 0.1 lb/in^3. The design problem is posed to minimise truss mass subjected to displacement, stress and buckling constrains which is mathematically defined as:

$$\begin{aligned}
&\min_{\mathbf{x}} f(\mathbf{x}) = \text{structural mass}\\
&\text{subject to}\\
&\frac{|d_{\max}|}{d_{all}} \le 1\\
&\frac{\sigma_{\max}}{\sigma_{all}} \le 1\\
&\frac{P_i}{P_{cr,i}} \le 1\\
&0.1 \le A_i \le 70 \quad \text{in}^2
\end{aligned} \tag{1}$$

where $\mathbf{x}^T=\{x_1, \ldots, x_{10}\} = \{A_1, \ldots, A_{10}\}$ is a vector of design variables, $f(\mathbf{x})$ is mass of the truss. d_{all} and σ_{all} are the allowance stress and displacement which was set to be 2 *in* and 25 *ksi*, respectively. The variables $d_{\max}$ and $\sigma_{\max}$ imply the maximum displacement and stress due to the applied loads respectively. P_i and $P_{cr,i}$ are respectively applied compressive and critical buckling loads at the bar element number *i*. A_i is cross-sectional areas of all truss members.

In cases of bar element under compression, $P_{cr,i}$ can be calculated as:

$$P_{cr,i} = \begin{cases} A_i \dfrac{\pi^2 E}{(l_i/r_i)^2} & ;if \;\; \dfrac{l_i}{r_i} < \sqrt{\dfrac{2\pi^2 E}{\sigma_{all}}} \\ A_i \sigma_{all}\left\{1 - \dfrac{\sigma_{all}}{4\pi^2 E}\left(\dfrac{l_i}{r_i}\right)^2\right\} & ;otherwise \end{cases} \tag{2}$$

where E, l_i and r_i are modulus of elasticity, effective length and cross-section radius of gyration of the *i*-th element respectively.

2.2 Twenty Five Bar Space Truss

The truss is displayed in Fig. 2 (Degertekin, 2012; Degertekin and Hayalioglu, 2013; Camp and Farshchin, 2014; Bureerat and Pholdee, 2015; Bekdas *et al*., 2017a). It is subjected to two cases of applied forces given in Table 1. All cross-sectional areas of the truss members are assigned as design variables which are divided into 8 groups as given in Table 2. Truss member density and Young modulus are 0.1 Ib/in^3 and 10^4 ksi. The mass minimisation problem is set to minimise structural mass subjected to displacement, stress and buckling constraints, which is written as:

Table 1. Loading Conditions for the Twenty Five Bar Space Truss (Bureerat and Pholdee, 2015)

Node	Condition 1			Condition 2		
	F_x	F_y	F_y	F_x	F_y	F_z
1	0.0	20.0	-5.0	1.0	10.0	-5.0
2	0.0	-20.0	-5.0	0.0	10.0	-5.0
3	0.0	0.0	0.0	0.5	0.0	0.0
6	0.0	0.0	0.0	0.5	0.0	0.0

Table 2. Groups of Cross-sectional Areas of the Twenty Five Bar Space Truss and Their Allowable Stress Values (Bureerat and Pholdee, 2015)

Element groups	Element numbers in the groups	Allowable compressive stresses (ksi)	Allowable tension stresses (ksi)
1	1	35.092	40.0
2	2-5	11.590	40.0
3	6-9	17.305	40.0
4	10-11	35.092	40.0
5	12-13	35.092	40.0
6	14-17	6.7590	40.0
7	18-21	6.9590	40.0
8	22-25	11.082	40.0

$$\begin{aligned}
&\min_{\mathbf{x}} f(\mathbf{x}) = \text{structural mass}\\
&\text{subject to}\\
&\frac{|d_{\max,j}|}{d_{all}} \le 1\\
&\frac{\sigma_{\max,j}}{\sigma_{all}} \le 1\\
&\frac{P_{i,j}}{P_{cr,i,j}} \le 1\\
&0.01 \le A_i \le 10 \quad \text{in}^2
\end{aligned} \tag{3}$$

where $\mathbf{x}^T = \{x_1, \ldots, x_{10}\}$ is a vector of design variables, $f(\mathbf{x})$ is mass of the truss. d_{all} is displacement which was set to be 0.35 while σ_{all} is an allowable stress reported in Table 2. The variables $d_{max,j}$ and $\sigma_{max,j}$ are respectively the maximum displacement and stress due to the j^{th} loading case. $P_{i,j}$ and $P_{cr,i,j}$ are respectively applied compressive and critical buckling loads at the bar element number i of the j^{th} load case. A_i are the cross-sectional areas of all truss members.

2.3 Seventy Two Bar Space truss

The 72-bar truss is displayed in Fig. 3 (Degertekin, 2012; Degertekin and Hayalioglu, 2013; Camp and Farshchin, 2014; Bureerat and Pholdee, 2015; Bekdas *et al.*, 2017a). It is acted on by two loading cases as detailed in Table 3. The design variables consist of the cross-sectional areas of all members grouped into 16 set as reported in Table 4. Young modulus and element density are 10^4 ksi and 0.1 lb/in^3 respectively. The optimal truss sizing problem is posed to minimise structural mass subjected to stress, displacement and buckling constraints, which is written as:

$$
\begin{aligned}
&\min_{\mathbf{x}} f(\mathbf{x}) = \text{structural mass} \\
&\text{subject to} \\
&\frac{|d_{\max,j}|}{d_{all}} \le 1 \\
&\frac{\sigma_{\max,j}}{\sigma_{all}} \le 1 \\
&\frac{P_{i,j}}{P_{cr,i,j}} \le 1 \\
&A_{\min} \le A_i \le A_{\max} \quad \text{in}^2
\end{aligned}
\tag{4}
$$

where $\mathbf{x}^T = \{x_1, \ldots, x_{16}\}$ is a design vector. d_{all} and σ_{all} are the allowance stress and displacement which was set to be 0.25 *in* and 25 *ksi*, respectively. The variables $d_{max,j}$ and $\sigma_{max,j}$ are the maximum deflection and stress due to the j^{th} loading condition respectively. $P_{i,j}$ and $P_{cr,I,j}$ are respectively applied compressive and critical buckling loads at the bar element number i of the j^{th} load case respectively. A_i are the cross-sectional areas of all truss members. The parameters A_{min} are assigned as 0.1 in^2 for design Case I and 0.01 in^2 for Case II. The value of A_{max} is set to be 35 in^2 for both design cases.

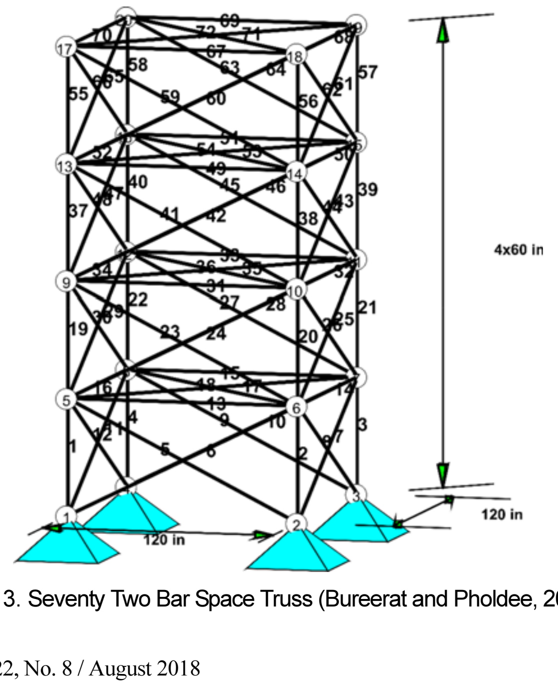

Fig. 3. Seventy Two Bar Space Truss (Bureerat and Pholdee, 2015)

Table 3. Loading Condition for 72- bar Space Truss (Bureerat and Pholdee, 2015)

Node	Condition 1			Condition 2		
	F_x	F_y	F_y	F_x	F_y	F_z
17	5.0	5.0	-5.0	0.0	0.0	-5.0
18	0.0	0.0	0.0	0.0	0.0	-5.0
19	0.0	0.0	0.0	0.0	0.0	-5.0
20	0.0	0.0	0.0	0.0	0.0	-5.0

Table 4. Element Groups of the 72-bar Space Truss (Pholdee and Bureerat, 2014b)

Element groups	Element numbers in the groups
1	1-4
2	5-12
3	13-16
4	17-18
5	19-22
6	23-30
7	31-34
8	35-36
9	37-40
10	41-48
11	49-52
12	53-54
13	55-58
14	59-66
15	67-70
16	71-72

2.4 Two Hundred Bar Plane Truss

The 200-bar plane truss is depicted in Fig. 4 (Degertekin, 2012; Degertekin and Hayalioglu, 2013; Camp and Farshchin, 2014; Bureerat and Pholdee, 2015; Bekdas *et al.*, 2017a). It is applied by three load cases as (1) x-direction 1.0 kip acting on nodes 1, 6, 15 20, 29, 34, 43, 48, 57, 62 and 71 (2) y-direction 10.0 kips acting on nodes 1, 2, 3, 4, 5, 6, 8, 10, 12, 14, 16, 17, 18, 19, 20, 22, 24, 26, 28, 29, 30, 31, 32, 33, 34, 36, 38, 40 ,42, 43, 44, 45, 46, 47, 48, 50, 52, 54, 56, 57, 58, 59, 60, 61, 62, 64, 66, 68, 70, 71, 72, 73, 74 and 75 (3) condition (1) and (2) combined. The cross-sectional areas of all truss members are set as design variables where they are grouped into 29 groups as shown in Table 5. Modulus of elasticity and truss element density are 30,000 ksi and 0.283 lb/in^3 respectively. The optimal truss sizing problem is posed to minimise structural mass subjected to stress

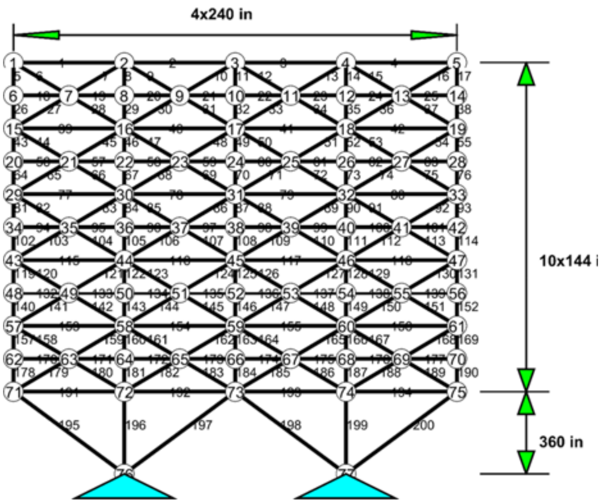

Fig. 4. Two Hundred Bar Plane Truss (Bureerat and Pholdee, 2015)

Table 5. Element Groups of the 200-bar Plane Truss (Bureerat and Pholdee, 2015)

Element groups	Element numbers in the groups
1	1, 2, 3, 4
2	5, 8, 11, 14, 17
3	19, 20, 21, 22, 23, 24
4	18, 25, 56, 63, 94, 101, 132, 139, 170, 177
5	26, 29, 32, 35, 38
6	6, 7, 9, 10, 12, 13, 15, 16, 27, 28, 30, 31, 33, 34, 36, 37
7	39, 40, 41, 42
8	43, 46, 49, 52, 55
9	57, 58, 59, 60, 61, 62
10	64, 67, 70, 73, 76
11	44, 45, 47, 48, 50, 51, 53, 54, 65, 66, 68, 69, 71, 72, 74, 75
12	77, 78, 79, 80
13	81, 84, 87, 90, 93
14	95, 96, 97, 98, 99, 100
15	102, 105, 108, 111, 114
16	82, 83, 85, 86, 88, 89, 91, 92, 103, 104, 106, 107, 109, 110, 112, 113
17	115, 116, 117, 118
18	119, 122, 125, 128, 131
19	133, 134, 135, 136, 137, 138
20	140, 143, 146, 149, 152
21	120, 121, 123, 124, 126, 127, 129, 130, 141, 142, 144, 145, 147, 148, 150, 151
22	153, 154, 155, 156
23	157, 160, 163, 166, 169
24	171, 172, 173,174, 175, 176
25	178, 181, 184, 187, 190
26	158, 159, 161, 162, 164, 165, 167, 168, 179, 180, 182, 183, 185, 186, 188,189
27	191, 192, 193, 194
28	195, 197, 198, 200
29	196, 199

and buckling constraints, which is written as:

$$\min_{\mathbf{x}} f(\mathbf{x}) = \text{structural mass}$$
$$\text{subject to}$$
$$\frac{\sigma_{\max,j}}{\sigma_{all}} \le 1$$
$$\frac{P_{i,j}}{P_{cr,i,j}} \le 1 \tag{5}$$
$$0.1 \le A_i \le 30 \quad \text{in}^2$$

where $\mathbf{x}^T = \{x_1, \ldots, x_{29}\}$ is a vector of design variables. σ_{all} is the allowance stress which was set to be 10 *ksi*, respectively. The variables $\sigma_{\max,\ j}$ are the maximum stress of the j^{th} loading condition. $P_{i,j}$ and $P_{cr,I,j}$ are respectively applied compressive and critical buckling loads at the bar element number i of the j^{th} load case respectively. A_i are the cross-sectional areas of all truss elements.

In conclusion, there are six test problems of truss sizing. Two problems are from the 10-bar truss, one from the 25-bar truss, other two from the 72-bar truss, and another one from the 200-bar truss. The numbers of design variables for the test problems are 10, 10, 8, 16, 16, and 29 respectively.

Apart from the six test problems mentioned above, 2 test problems of the 25-bar and 72-bar trusses with all truss elements being assigned as design variables without grouping them are also used for performance test. For the 25-bar truss, the design problem with ungrouped element members as design variables (Bekdas *et al*., 2017b) is considered with loading conditions as shown in Table 6. The lower and upper bounds for all element cross-sections are set to be 0.01 in^2 and 3.4 in^2, respectively.

For the 72-bar truss, the design problem with ungrouped element members as design variables (Bekdas *et al.*, 2017b) is considered with loading conditions as show in Table 7. The lower and upper bound for all element cross-sections are set to be 0.1 in^2 and 3 in^2, respectively.

Table 6. Load Cases for the Twenty-five Bar Space Truss Without Grouping Element Member (Bekdas *et al*., 2017b)

Node	Condition 1			Condition 2		
	F_x	F_y	F_y	F_x	F_y	F_z
1	1.0	10.0	-5.0	0.0	20.0	-5.0
2	0.0	10.0	-5.0	0.0	-20.0	-5.0
3	0.5	0.0	0.0	0.0	0.0	0.0

Table 7. Loading Conditions for the 72 bar Space Truss Without Grouping Element Member (Bekdas *et al*., 2017b)

Node	Condition 1			Condition 2		
	F_x	F_y	F_y	F_x	F_y	F_z
17	-5.0	-5.0	-5.0	5.0	5.0	-5.0
18	-5.0	-5.0	-5.0	0.0	0.0	0.0
19	-5.0	-5.0	-5.0	0.0	0.0	0.0
20	-5.0	-5.0	-5.0	0.0	0.0	0.0

3. Self-adaptive Meta-heuristics Used in This Study

In this work, eight teen self-adaptive MHs were compared by implementing them to solve the six optimal truss sizing test problems. The reason only self-adaptive meta-heuristics are employed is because this can avoid controversy in optimisation parameter settings (such as crossover and mutation rates in genetic algorithms), which is always an issue when implementing such optimisers. Those SAMHs (details of notations are given in the corresponding references of those algorithms) are:

1. Adaptive Differential Evolution (JADE) (Zhang and Sanderson, 2009).
2. Evolution Strategy with Covariance Matrix Adaptation (CMAES) (Hansen *et al.*, 2003).
3. Composite Differential Evolution (CoDE) (Yong *et al.*, 2011).
4. Comprehensive Learning Particle Swarm Optimizer (CLPSO) (Liang *et al.*, 2006).
5. Ensemble of Mutation Strategies and Control Parameters with the Differential Evolution (EPSDE) (Mallipeddi *et al.*, 2011).
6. Hybrid Global and Local Real-Coded Genetic Algorithms (GL-25) (García-Martínez *et al.*, 2008).
7. Self-Adaptive Control Parameter in Differential Evolution (jDE) (Brest, Greiner *et al.*, 2006).
8. Self-Adaptive Differential Evolution (SaDE) (Qin *et al.*, 2009).
9. Combining Multiobjective Optimization with Differential Evolution (CMODE) (Yong and Zixing, 2012).
10. Improved $(\mu+\lambda)$-Constrained Differential Evolution (ICDE) (Jia *et al.*, 2013).
11. Orthogonal Crossover Based Differential Evolution (OXDE) (Wang *et al.*, 2012).
12. Success-History Based Adaptive Differential Evolution (SHADE) (Tanabe and Fukunaga, 2013).
13. United Multi-Operator Evolutionary Algorithms (UMOEAs) (Elsayed *et al.*, 2014)
14. SHADE with Linear Population Size Reduction (L-SHADE) (Tanabe and Fukunaga, 2014)
15. Covariance Matrix Learning and Searching Preference (CMLSP) (Lei *et al.*, 2014).
16. L-Shade with Eigenvector-Based Crossover and Successful-Parent-Selecting Framework (SPS-L-SHADE-EIG) (Shu-Mei *et al.*, 2015).
17. Improved Covariance Matrix Learning and Searching Preference (ICMLSP) (Lei *et al.*, 2015).
18. Neurodynamic Differential Evolution Algorithm (L-SHADE-ND) (Sallam *et al.*, 2015).

Each optimiser is run to tackle each optimisation problem for 30 independent runs (Pholdee and Bureerat, 2014b). The comparative results are performed with two experiments: optimisation design with and without considering buckling constraints. With the exception of the test problem VI, the population size, for all test problems, is set to be $n_P = 50$ while the number of generations is 200. For the test problem VI, the number of generations and the population size are 250 and 80 respectively. Any optimiser that uses different population size and number of loops will be terminated with the total number of functions evaluations (FEs) equal to 50 × 200 for the first five test cases and 80 × 250 for Case VI. It should be noted that 50 × 200 = 10,000 FEs for Case I-V and 80 × 250 = 20,000 FEs for Case VI can be considered insufficient for some meta-heuristics according to the literature; nevertheless, this value is set so as to look for only really powerful algorithms. The penalty function employed in this paper is the Kaveh-Zolghadr technique, which can be expressed as:

$$f_p(\mathbf{x}) = (1+\varepsilon_1 \cdot v)^{\varepsilon_2} \tag{6}$$

where f_p is a penalised objective function used for converting a constrained problem to an equivalent unconstrained problem. The variables ε_1 and ε_2 are selected considering the exploration and exploitation rates of the search space.

$$v = \sum_{\text{i-1}}^{\text{ng}} v_i \tag{7}$$

where ng and v_i are the number of constrain the constraints violation. The v_i for displacement, stress and buckling constraints can be calculate as follow:

$$v_i = \begin{cases} \max(\frac{d_{\max}}{d_{all}} - 1,0) \\ \max(\frac{\sigma_{\max}}{\sigma_{all}} - 1,0) \\ \max(\max(\frac{P_i}{P_{cr,i}} - 1),0) \end{cases} \tag{8}$$

The technique was proven to be effective in cases of truss sizing subjected to natural frequency constraints (Kaveh and Zolghadr, 2012).

4. Results and Discussion

4.1 Comparatives Results of the Trusses Optimisation Without Buckling Constraints

Having performed 30 optimisation runs of 18 SAMHs for solving the six truss sizing design problems, the comparative results of the six design cases are shown in Table 8. It should be note that only top two best performers are shown in the tables. The mean values of objective functions are used to measure the search convergence of the MHs. The lower mean objective function value satisfying all constraints the better search convergence.

For the measure of search convergence based on the mean objective function values of the design problems without considering buckling constraints, SHADE is the best for the all design cases except for the Case III while the best performer for the Case III is L-SHADE. The second best for the Case II, Case IV, Case V and Case V\I is L-SHADE while the second best for

Table 8. Comparative Performance for Case I-VI

Design Problem	Objective function (Weight, Ib)			
	Without buckling constraints		With buckling constraints	
	Best (mean ± std)	Second best (mean ± std)	Best (mean ± std)	Second best (mean ± std)
Case I	SHADE (5060.961 ± 0.061)	SPS-L-SHADE-EIG (5061.332 ± 0.4711)	L-SHADE (8741.060 ± 163.89)	SPS-L-SHADE-EIG (8753.553 ± 164.19)
Case II	SHADE (4677.412 ± 0.3657)	L-SHADE (4677.768 ± 1.2787)	SPS-L-SHADE-EIG (8220.654 ± 5.3910)	SHADE (8246.900 ± 37.331)
Case III	L-SHADE (545.163 ± 0.0006)	SHADE (545.166 ± 0.0032)	L-SHADE SPS-L-SHADE-EIG (1623.212 ± 0.0025,0.0033)	-
Case IV	SHADE (379.985 ± 0.2285)	L-SHADE (380.036 ± 0.3802)	SPS-L-SHADE-EIG (1274.927 ± 6.4514)	L-SHADE (1279.792 ± 12.1403)
Case V	SHADE (364.261 ± 0.2280)	L-SHADE (364.399 ± 0.2854)	SPS-L-SHADE-EIG (1269.523 ± 8.1007)	L-SHADE (1270.900 ± 14.2772)
Case VI	SHADE (26109.67 ± 187.34)	L-SHADE (26197.27 ± 398.28)	SPS-L-SHADE-EIG (58662.63 ± 244.24)	SHADE (58961.50 ± 476.66)

Table 9. Top Five Optimisers Based on Wilcoxon Rank Sum Test for All Problems

Problem	MHs	Ranking Case I	Ranking Case II	Ranking Case III	Ranking Case IV	Ranking Case V	Ranking Case VI	Sum	Ranking All
Without buckling constraints	CMAES	3	3	2	4	8	6	26	4
	L-SHADE	2	1	1	1	1	2	8	1
	SPS-L-SHADE-EIG	3	3	3	4	4	2	19	3
	SHADE	1	1	3	1	1	1	8	1
	ICMLSP	3	3	5	3	1	11	26	4
With buckling constraints	JADE	5	4	4	4	4	4	25	4
	jDE	6	8	6	5	5	6	36	5
	L-SHADE	1	1	1	1	1	2	7	1
	SPS-L-SHADE-EIG	2	1	2	1	1	1	8	2
	SHADE	3	3	3	3	3	2	17	3

the design Case I and Case III are SPS-L-SHADE-EIG and SHADE, respectively.

For solving the six truss sizing design problems with buckling constraints, the best optimiser for the design Case I is L-SHADE while the best performer for the rest is SPS-L-SHADE-EIG. For the design Case III, the SPS-L-SHADE-EIG and L-SHADE obtained the same best mean objective function value. The second best for Case I, Case IV and Case V are SPS-L-SHADE-EIG, L-SHADE, and L-SHADE, respectively, while the second best for the rest is SHADE.

Additionally, the ranking of MHs is carried out using the statistical Wilcoxon rank sum test (Pholdee and Bureerat, 2013). The null hypothesis for the Wilcoxon rank sum test is that, for a particular test problem, the median of 30 objective function values (from 30 runs of each optimiser) obtained from method *I* is not different from the median of 30 objective function values obtained from using method *J* at the 5% significance level. If the null hypothesis is rejected and the median from method *J* is lower, the element *IJ* of the matrix is set to be '1'. After summing up all values in the matrix columns, the best optimiser is the one that has the highest score. More details of this performance assessment are given in (Pholdee and Bureerat, 2014b). Note that this comparison is made if and only if both optimisers *I* and *J* give feasible solutions for all optimisation runs. If such a condition does not hold, between methods *I* or *J*, the method that give more feasible solutions for all runs is the winner. In case that both of them fail to find feasible solutions for all runs, one having lower constraint violation will win; otherwise, both of them are equal (Pholdee and Bureerat, 2014b).

Table 9 shows top 5 optimisers from the 18 implemented optimisers when solving all truss sizing design problems with and without buckling constraint based on the Wilcoxon rank sum test. For the design problems without considering bucking constraints, after summing up the ranking numbers of all design cases, it is found that SHADE and LSHADE are equally good as the best overall optimiser while the SPS-L-SHADE-EIG and ICMLSP are the third and fourth best, respectively. For the design problems with bucking constraints, L-SHADE is the best optimiser while the second best is SPS-L-SHADE-EIG. SHADE and JADE are the third and fourth best respectively.

In this work, the self-adaptive meta-heuristics are coded in MATLAB computing language (Pholdee and Bureerat, 2014b). Most of them are downloaded from their respective authors while some of them are created by the authors of this article following the details in the references. In addition, the number of function evaluations for searching optima is limited to 10,000 for

Table 10. Best Results Obtained from This Study for Case I-III

Problems	Case I		Case II		Case III	
Buckling Constraint Inclusion	No	Yes	No	Yes	No	Yes
Optimiser	L-SHADE	L-SHADE	L-SHADE	L-SHADE	L-SHADE	L-SHADE, SPS-L-SHADE-EIG CMASE
Set of optimum design variable (Detailed in section 2)	30.5138 0.1000 23.2211 15.2332 0.1000 0.5508 7.4540 21.0268 21.5250 0.1000	17.6822 0.1000 57.0877 40.6217 0.1000 0.1000 7.4164 69.1659 12.5494 0.1000	23.5348 0.1000 25.2745 14.3773 0.1000 1.9697 12.3938 12.8314 20.3244 0.1000	13.6549 3.3098 63.8208 40.7563 0.1000 2.4409 11.7427 49.8125 11.9802 0.1000	0.0100 1.9871 2.9935 0.0100 0.0100 0.6840 1.6769 2.6621	0.0100 5.5265 5.3385 0.0100 0.9159 3.9525 6.9541 5.6652
Weight (Ib)	5,060.86	8,707.83	4,676.92	8,215.89	545.163	1,623.21
Max stress constraint (load 1)	0	0	0	0	0	-0.7608
Max stress constraint (load 2)	None	None	None	None	-0.1958	-0.7746
Max displacement constraint (load 1)	0	0	0	0	0	-0.6209
Max displacement constraint (load 2)	None	None	None	None	0	-0.6015
Max buckling constraint (load 1)	215.8330	0	1,247.7	0	133.1585	0
Max buckling constraint (load 2)	None	None	None	None	277.7786	0
No. of analyses	10,000	10,000	10,000	10,000	10,000	10,000

Case I-V and 20,000 for Case VI. Such numbers of function evaluations may be inadequate for some implemented optimisers.

Tables 10-11 show the comparison between the best results found in this study for all design cases based on the analyses with and without buckling constraints. The minimum objective function values and percentage of constraints violation of stress, displacement and buckling are present in these tables. In cases of the problems without buckling, the best results found in this study for all design cases except for the Case IV and Case V is L-SHADE while ICMLSP produces the best solutions for Case IV and Case V. In cases of the problems with buckling constraints, the best results found in this study for the design Case I-V is obtained by using L-SHADE while SPS-L-SHADE-EIG gives the best results for the Case VI. For the design Case III, L-SHADE and SPS-L-SHADE-EIG give the best results.

From literature, the best results obtained from the recently published work by Khatibinia and Yazdanib, 2017 for Cases I-II and Cases IV-VI without considering bucking constraints are respectively 5060.9 Ib (5060.9 ± 0), 4677.0 Ib (4677.2 ± 0.1), 379.6 Ib (379.6 ± 0), 363.8 Ib (363.9 ± 0), and 25461.0 Ib (25547.4 ± 110.1 lb), with 10,500 function evaluations. For Case II, the best result is obtained from Degertekin et al., 2017 as 545.13 Ib (545.47 ± 0.476) within 7,653 function evaluations. It should be noted that those previously reported numbers are of the form *Best* (*Mean* ± *Standard deviation*). By comparing to Tables 8, 10 and 11, it was found that the results obtained from the self-adaptive algorithms used in this study are comparable with so far best results in the literature. New best results for Cases I and II are reported in the present study for the optimisation problem without buckling constraints. Also, all the optimum results reported in this paper when taking into account buckling constraints can be used as the new baseline results for future investigation.

In cases of having buckling constraints, the minimum mass obtained are significantly higher. According to the constraint violation check in the tables, it is shown that the results which ignored buckling in the design are not safe from buckling instability. This implies that, in reality, the buckling constraints should be added to a truss design problem ensuring that the designed structure fulfils all safety requirements. Moreover, the results presented in this paper could be considered as the initial benchmark results for truss optimisation with buckling constraints in which following researchers could use for testing their developed algorithms.

The comparative results for the test problems of 25-bar truss and 72-bar truss with ungrouped design variables are shown in Table 12. Based on the mean objective function value, CMAES and L-SHADE are the best performer for both 25-bar truss and 72-bar truss, respectively. The second best for the 25-bar truss is SHADE while SPS-L-SHADE-EIG is the second best for the 72-bar truss. The optimum results obtained from Bekdas *et al.*, 2017 are 543.20 Ib and 360.518 Ib, for the 25-bar truss and the 72-bar truss, respectively, with 2,000,000 function evaluations. The minimum weight for the 25-bar truss and the 72-bar truss are obtained by SHADE and ICMLSP which are 544.301 Ib and 393.785 Ib, respectively with 10,000 function evaluations.

5. Conclusions

This paper gathers many self-adaptive meta-heuristic algorithms and implements them on truss sizing optimisation with stress, displacement and buckling constraints. Based on the six

Table 11. Best rEsults Obtained from This Study for Case IV-VI

Problems	Case IV		Case V		Case VI	
Buckling Analysis	No	Yes	No	Yes	No	Yes
Optimiser	ICMLSP	L-SHADE	ICMLSP	L-SHADE	L-SHADE	SPS-L-SHADE-EIG
Set of optimum design variable (Detailed in section 2)	1.8718 0.5120 0.1001 0.1000 1.2617 0.5143 0.1000 0.1001 0.5345 0.5140 0.1001 0.1002 0.1563 0.5442 0.4133 0.5803	1.9688 1.9257 0.3415 0.1001 1.5198 1.9361 0.4739 0.8243 1.4980 1.7757 0.1004 0.3359 1.4693 2.2854 1.6476 2.7727	1.8655 0.5159 0.0100 0.0100 1.2891 0.5183 0.0100 0.0103 0.5305 0.5238 0.0100 0.1199 0.1658 0.5357 0.4493 0.5589	1.9521 1.9138 0.3129 0.0100 1.5213 1.9369 0.5121 0.9188 1.5059 1.7608 0.0127 0.2350 1.4868 2.2415 1.6705 2.8074	0.1161 0.9964 0.1115 0.1000 2.0027 0.2831 0.1301 3.0565 0.2275 4.0566 0.4615 0.1185 5.4152 0.1903 6.4146 0.5987 0.1781 8.0388 0.1119 9.0195 0.7627 0.2368 10.9545 0.1045 12.0040 0.9461 6.5914 10.8910 13.8166	1.3371 2.5234 0.4490 0.8701 3.9995 2.9417 1.3491 4.7238 0.4212 5.7211 3.8652 1.3861 6.9662 0.5488 7.9665 4.4182 1.4343 9.6447 0.4474 10.6536 4.8760 1.7949 12.4617 0.9154 13.4759 5.7288 5.7879 27.3881 28.8912
Weight (Ib)	379.65	1,265.25	363.87	1,253.23	25,553.20	58,168.61
Max stress constraint (load 1)	-0.3389	-0.8550	-0.3242	-0.8534	0	-0.8851
Max stress constraint (load 2)	0	-0.8693	0	-0.8686	0	-0.2089
Max stress constraint (load 3)	None	None	None	None	0	-0.1644
Max displacement constraint (load 1)	0	-0.5842	0	-0.5815	None	None
Max displacement constraint (load 2)	-0.0132	-0.7103	-0.0094	-0.7080	None	None
Max displacement constraint (load 3)	None	None	None	None	None	None
Max buckling constraint (load 1)	56.2119	0	915.7888	0	116.4961	-0.0018
Max buckling constraint (load 2)	72.3094	0	402.5494	0	34.4200	-0.0002
Max buckling constraint (load 3)	None	None	None	None	60.1155	-0.0001
No. of analyses	10,000	10,000	10,000	10,000	20,000	20,000

Table 12. Comparative Results for the Trusses with Ungrouped Design Variables

Design problems	Objective function (Weight, Ib)	
	Best (mean±std)	Second best (mean±std)
25-bar truss with ungrouped design variables	CMAES (545.127±0.5256)	SHADE (545.316±0.6408)
72-bar truss with ungrouped design variables	L-SHADE (394.032±0.2488)	SPS-L-SHADE-EIG (394.292±0.444)

traditional test problems, the performances of the optimisers are compared. The results shown that the top three meta-heuristics for all test problems with and without buckling constraints, are L-SHADE, SHADE and SPS-L-SHADE-EIG. The constraint violation check reveals that buckling is one important design constraint for truss sizing design. The results reported in this paper are said to be the new baselines for those who want to develop meta-heuristics for solving optimal truss sizing.

Acknowledgements

The authors are graceful for the support from the Thailand Research Fund (TRF) grant number BRG5880014.

References

Ahrari, A. and Atai, A. A. (2013). "Fully stressed design evolution strategy for shape and size optimization of truss structures." *Computers & Structures*, Vol. 123, pp. 58-67, DOI: 10.1016/j.compstruc.2013.04.013.

Ahrari, A., Atai, A. A., and Deb, K. (2015). "Simultaneous topology, shape and size optimization of truss structures by fully stressed design based on evolution strategy." *Engineering Optimization*, Vol. 47, No. 8, pp. 1063-1084, DOI: 10.1080/0305215X.2014.947972.

Baluja, S. (1994). *Population-Based Incremental Learning: A Method for Integrating Genetic Search Based Function Optimization and Competitive Learning*, In Population-Based Incremental Learning: A Method for Integrating Genetic Search Based Function Optimization and Competitive Learning Ed CMU-CS-94-163, t.r. Carnagie Mellon University, Pittsburgh.

Baykasoğlu, A. and Ozsoydan, F. B. (2014). "An improved firefly algorithm for solving dynamic multidimensional knapsack problems." *Expert Systems with Applications*, Vol. 41, No. 8, pp. 3712-3725, DOI: 10.1016/j.eswa.2013.11.040.

Beck, A., Ben-Tal, A., and Tetruashvili, L. (2010). "A sequential parametric convex approximation method with applications to nonconvex truss topology design problems." *Journal of Global Optimization*, Vol. 47, No. 1, pp. 29-51, DOI: 10.1007/s10898-009-9456-5.

Bekdaş, G., Nigdeli, S. M., and Yang, X.-S. (2015). "Sizing optimization of truss structures using flower pollination algorithm." *Applied Soft Computing*, Vol. 37, pp. 322-331, DOI: 10.1016/j.asoc.2015.08.037.

Bekdas, G., Nigdeli, S. M., and Türakin, O. H. (2017a). "Non-Linear programing for sizing optimization of truss structures." *International Journal of Theoretical and Applied Mechanics*, Vol. 1, pp. 274-284.

Bekdas, G., Nigdeli, S. M., and Yang, X. S. (2017b). "Size optimization of truss structures employing flower pollination algorithm without grouping structural members." *International Journal of Theoretical and Applied Mechanics*, Vol. 1, pp. 269-273.

Brest, J., Greiner, S., Boskovic, B., Mernik, M., and Zumer, V. (2006). "Self-Adapting control parameters in differential evolution: A comparative study on numerical benchmark problems." *Evolutionary Computation, IEEE Transactions on*, Vol. 10, No. 6, pp. 646-657, DOI: 10.1109/TEVC.2006.872133.

Bureerat, S. (2011). *Improved Population-Based Incremental Learning in Continuous Spaces,* In Improved Population-Based Incremental Learning in Continuous Spaces Eds Gaspar-Cunha, A., Takahashi, R., Schaefer, G. & Costa, L. pp. 77-86. Springer Berlin Heidelberg, DOI: 10.1007/978-3-642-20505-7_6.

Bureerat, S. and Pholdee, N. (2015). "Optimal truss sizing using an adaptive differential evolution algorithm." *Journal of Computing in Civil Engineering*, 04015019, DOI: 10.1061/(ASCE)CP.1943-5487.0000487#sthash.zn4UhHlF.dpuf.

Camp, C. V. and Farshchin, M. (2014). "Design of space trusses using modified teaching–learning based optimization." *Engineering Structures,* Vols. 62-63, pp. 87-97, DOI: 10.1016/j.engstruct.2014.01.020.

Dede, T. and Ayvaz, Y. (2015). "Combined size and shape optimization of structures with a new meta-heuristic algorithm." *Applied Soft Computing*, Vol. 28, pp. 250-258, DOI: 10.1016/j.asoc.2014.12.007

Degertekin, S. O. (2012). "Improved harmony search algorithms for sizing optimization of truss structures." *Computers & Structures*, Vols. 92-93, pp. 229-241, DOI: 10.1016/j.compstruc.2011.10.022.

Degertekin, S. O. and Hayalioglu, M. S. (2013). "Sizing truss structures using teaching-learning-based optimization." *Computers & Structures*, Vol. 119, pp. 177-188, DOI: 10.1016/j.compstruc.2012.12.011.

Degertekin, S. O., Lamberti, L., and Hayalioglu, M. S. (2017). "Heat transfer search algorithm for sizing optimization of truss structures." *Latin American Journal of Solids and Structures*, Vol. 14, No. 3, pp. 373-397, DOI: 10.1590/1679-78253297.

Elsayed, S. M., Sarker, R. A., Essam, D. L., and Hamza, N. M. (2014). *Testing united multi-operator evolutionary algorithms on the CEC2014 real-parameter numerical optimization*, In Testing united multi-operator evolutionary algorithms on the CEC2014 real-parameter numerical optimization pp. 1650-1657, DOI: 10.1109/CEC.2014.6900308.

Flager, F., Adya, A., Haymaker, J., and Fischer, M. (2014). "A bi-level hierarchical method for shape and member sizing optimization of steel truss structures." *Computers & Structures*, Vol. 131, pp. 1-11, DOI: 10.1016/j.compstruc.2013.10.004.

García-Martínez, C., Lozano, M., Herrera, F., Molina, D., and Sánchez, A. M. (2008). "Global and local real-coded genetic algorithms based on parent-centric crossover operators." *European Journal of Operational Research*, Vol. 185, No. 3, pp. 1088-1113, DOI: 10.1016/j.ejor.2006.06.043.

Gholizadeh, S. and Barzegar, A. (2012). "Shape optimization of structures for frequency constraints by sequential harmony search algorithm." *Engineering Optimization*, Vol. 45, No. 6, pp. 627-646, DOI: 10.1080/0305215X.2012.704028.

Hajela, P. (1990). "Genetic search - An approach to the nonconvex optimization problem." *AIAA Journal*, Vol. 28, No. 7, pp. 1205-1210, DOI: 10.2514/3.25195.

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.*, Vol. 11, No. 1, pp. 1-18, DOI: 10.1162/106365603321828970.

Hosseini, K., Nodoushan, E. J., Barati, R., and Shahheydari, H. (2016). "Optimal design of labyrinth spillways using meta-heuristic algorithms." *KSCE Journal of Civil Engineering*, Vol. 20, No. 1, pp. 468-477, DOI: 10.1007/s12205-015-0462-5.

Husseinzadeh Kashan, A. (2011). "An efficient algorithm for constrained global optimization and application to mechanical engineering design: League Championship Algorithm (LCA)." *Computer-Aided Design*, Vol. 43, No. 12, pp. 1769-1792, DOI: 10.1016/j.cad.2011.07.003.

Jia, G., Wang, Y., Cai, Z., and Jin, Y. (2013). "An improved (μ+λ)-constrained differential evolution for constrained optimization." *Information Sciences*, Vol. 222, pp. 302-322, DOI: 10.1016/j.ins.2012.01.017.

Jung, J., Jayakrishnan, R., and Nam, D. (2015). "High coverage point-to-point transit: Hybrid evolutionary approach to local vehicle routing." *KSCE Journal of Civil Engineering*, Vol. 19, No. 6, pp. 1882-1891, DOI: 10.1007/s12205-014-0069-2.

Kaveh, A. and Ilchi Ghazaan, M. (2015). "Hybridized optimization algorithms for design of trusses with multiple natural frequency constraints." *Advances in Engineering Software*, Vol. 79, pp. 137-147, DOI: 10.1016/j.advengsoft.2014.10.001.

Kaveh, A. and Ilchi Ghazaan, M. (2016). "Optimal design of dome truss structures with dynamic frequency constraints." *Structural and Multidisciplinary Optimization*, Vol. 53, No. 3, pp. 605-621, DOI: 10.1007/s00158-015-1357-2.

Kaveh, A. and Javadi, S.M. (2014). "Shape and size optimization of trusses with multiple frequency constraints using harmony search and ray optimizer for enhancing the particle swarm optimization algorithm." *Acta Mechanica*, Vol. 225, No. 6, pp. 1595-1605, DOI: 10.1007/s00707-013-1006-z.

Kaveh, A. and Khayatazad, M. (2013). "Ray optimization for size and shape optimization of truss structures." *Computers & Structures*, Vol. 117, pp. 82-94, DOI: 10.1016/j.compstruc.2012.12.010

Kaveh, A. and Mahdavi, V. R. (2015). "A hybrid CBO–PSO algorithm for optimal design of truss structures with dynamic constraints." *Applied Soft Computing*, Vol. 34, pp. 260-273, DOI: 10.1016/j.asoc.2015.05.010.

Kaveh, A. and Talatahari, S. (2009). "Particle swarm optimizer, ant colony strategy and harmony search scheme hybridized for optimization of truss structures." *Computers and Structures*, Vol. 87, Issues 5-6, pp. 267-283, DOI: 10.1016/j.compstruc.2009.01.003.

Kaveh, A. and Talatahari, S. (2010). "A novel heuristic optimization method: Charged system search." *Acta Mechanica*, Vol. 213, No. 3, pp. 267-289, DOI: 10.1007/s00707-009-0270-4.

Kaveh, A. and Zolghadr, A. (2012). "Truss optimization with natural frequency constraints using a hybridized CSS-BBBC algorithm with trap recognition capability." *Computers & Structures,* Vols. 102-103, pp. 14-27, DOI: 10.1016/j.compstruc.2012.03.016.

Kaveh, A. and Zolghadr, A. (2014). "Enhanced bat algorithm for optimal design of skeletal structure." *Asian Journal of Civil Engineering (Building and Housing),* Vol. 15, pp. 179-212.

Kaveh, A., Bakhshpoori, T., and Afshari, E. (2014). "An efficient hybrid particle swarm and swallow swarm optimization algorithm." *Computers & Structures*, Vol. 143, pp. 40-59, DOI: 10.1016/j.compstruc. 2014. 07.012.

Kaveh, A., Sheikholeslami, R., Talatahari, S., and Keshvari-Ilkhichi, M. (2014). "Chaotic swarming of particles: A new method for size optimization of truss structures." *Advances in Engineering Software,* Vol. 67, pp. 136-147, DOI: 10.1016/j.advengsoft.2013.09.006.

Khatibinia, M. and Yazdani, H. (2017). "Accelerated multi-gravitational search algorithm for size optimization of truss structures." *Swarm and Evolutionary Computation, in press*, DOI: 10.1016/j.swevo. 2017.07.001.

Lei, C., Hai-lin, L., Chaoda, P., and Shengli, X. (2015). "An improved covariance matrix leaning and searching preference algorithm for solving CEC 2015 benchmark problems." *In An improved covariance matrix leaning and searching preference algorithm for solving CEC 2015 benchmark problems,* pp. 1041-1045, DOI: 10.1109/CEC. 2015.7257004.

Lei, C., Hai-lin, L., Zhe, Z., and Shengli, X. (2014). "An evolutionary algorithm based on Covariance Matrix Leaning and Searching Preference for solving CEC 2014 benchmark problems." *In An evolutionary algorithm based on Covariance Matrix Leaning and Searching Preference for solving CEC 2014 benchmark problems,* pp. 2672-2677, DOI: 10.1109/CEC.2014.6900594.

Li, J.-P. (2014). "Truss topology optimization using an improved species-conserving genetic algorithm." *Engineering Optimization*, Vol. 47, No. 1, pp. 107-128, DOI: 10.1080/0305215X.2013.875165

Liang, J. J., Qin, A. K., Suganthan, P. N., and Baskar, S. (2006). "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions." *IEEE Transactions on Evolutionary Computation*, Vol. 10, No. 3, pp. 281-295, DOI: 10.1109/TEVC.2005.857610.

Mallipeddi, R., Suganthan, P. N., Pan, Q. K., and Tasgetiren, M. F. (2011). "Differential evolution algorithm with ensemble of parameters and mutation strategies." *Applied Soft Computing*, Vol. 11, No. 2, pp. 1679-1696, DOI: 10.1016/j.asoc.2010.04.024.

Meng, T. and Pan, Q.-K. (2017). "An improved fruit fly optimization algorithm for solving the multidimensional knapsack problem." *Applied Soft Computing,* Vol. 50, pp. 79-93, DOI: 10.1016/j.asoc. 2016.11.023.

Miguel, L. F. F. and Fadel Miguel, L. F. (2012). "Shape and size optimization of truss structures considering dynamic constraints through modern metaheuristic algorithms." *Expert Systems with Applications*, Vol. 39, No. 10, pp. 9458-9467, DOI: 10.1016/j.eswa. 2012.02.113.

Mortazavi, A. and Toğan, V. (2017). "Sizing and layout design of truss structures under dynamic and static constraints with an integrated particle swarm optimization algorithm." *Applied Soft Computing*, Vol. 51, pp. 239-252, DOI: 10.1016/j.asoc.2016.11.032.

Muthiah-Nakarajan, V. and Noel, M. M. (2016). "Galactic Swarm Optimization: A new global optimization metaheuristic inspired by galactic motion." *Applied Soft Computing*, Vol. 38, pp. 771-787, DOI: 10.1016/j.asoc.2015.10.034.

Nariman, N. A. (2016). "Aerodynamic stability parameters optimization and global sensitivity analysis for a cable stayed Bridge." *KSCE Journal of Civil Engineering*, pp. 1-16, DOI: 10.1007/s12205-016-0962-y.

Noilublao, N. and Bureerat, S. (2011). "Simultaneous topology, shape and sizing optimisation of a three-dimensional slender truss tower using multiobjective evolutionary algorithms." *Computers & Structures*, Vol. 89, Nos. 23-24, pp. 2531-2538, DOI: 10.1016/j.compstruc. 2011.08.010.

Noilublao, N. and Bureerat, S. (2013). "Simultaneous topology, shape, and sizing optimisation of plane trusses with adaptive ground finite elements using MOEAs." *Mathematical Problems in Engineering*, Vol. 2013, Article ID 838102, 9 pages, DOI: 10.1155/2013/838102.

Pholdee, N. and Bureerat, S. (2013). "Hybridisation of real-code population-based incremental learning and differential evolution for multiobjective design of trusses." *Information Sciences*, Vol. 223, pp. 136-152, DOI: 10.1016/j.ins.2012.10.008.

Pholdee, N. and Bureerat, S. (2014a). "Hybrid real-code population-based incremental learning and approximate gradients for multi-objective truss design. *Engineering Optimization*, Vol. 46, No. 8, pp. 1032-1051, DOI: 10.1080/0305215X.2013.823194.

Pholdee, N. and Bureerat, S. (2014b). "Comparative performance of meta-heuristic algorithms for mass minimisation of trusses with dynamic constraints." *Advances in Engineering Software*, Vol. 75, pp. 1-13, DOI: 10.1016/j.advengsoft.2014.04.005.

Qin, A. K., Huang, V. L., and Suganthan, P. N. (2009). "Differential evolution algorithm with strategy adaptation for global numerical optimization." *IEEE Transactions on Evolutionary Computation*, Vol. 13, No. 2, pp. 398-417, DOI: 10.1109/TEVC.2008.927706.

Rao, R. V., Savsani, V. J., and Vakharia, D. P. (2011). "Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems." *Computer-Aided Design*, Vol. 43, No. 3, pp. 303-315, DOI: 10.1016/j.cad.2010.12.015.

Rashedi, E., Nezamabadi-pour, H., and Saryazdi, S. (2009). "GSA: A Gravitational Search Algorithm." *Information Sciences*, Vol. 179, No. 13, pp. 2232-2248, DOI: 10.1016/j.ins.2009.03.004.

Richardson, J., Adriaenssens, S., Bouillard, P., and Filomeno Coelho, R. (2012). "Multiobjective topology optimization of truss structures with kinematic stability repair." *Structural and Multidisciplinary Optimization*, Vol. 46, No. 4, pp. 513-532, DOI: 10.1007/s00158-012-0777-5.

Richardson, J.N ., Coelho, R. F., and Adriaenssens, S. (2016). "A unified stochastic framework for robust topology optimization of continuum and truss-like structures." *Engineering Optimization*, Vol. 48, No. 2, pp. 334-350, DOI: 10.1080/0305215X.2015.1011152.

Sallam, K. M., Sarker, R. A., Essam, D. L., and Elsayed, S. M. (2015). "Neurodynamic differential evolution algorithm and solving CEC2015 competition problems." *In Neurodynamic differential evolution algorithm and solving CEC2015 competition problems,* pp. 1033-1040, DOI: 10.1109/CEC.2015.7257003.

Sheikholeslami, R., Khalili, B. G., Sadollah, A., and Kim, J. (2016). "Optimization of reinforced concrete retaining walls via hybrid firefly algorithm with upper bound strategy." *KSCE Journal of Civil Engineering*, Vol. 20, No. 6, pp. 2428-2438, DOI: 10.1007/s12205-015-1163-9.

Shu-Mei, G., Tsai, J. S. H., Chin-Chang, Y., and Pang-Han, H. (2015). "A self-optimization approach for L-SHADE incorporated with eigenvector-based crossover and successful-parent-selecting framework on CEC 2015 benchmark set." *Evolutionary Computation (CEC), 2015 IEEE Congress on,* pp. 1003-1010, DOI: 10.1109/CEC.2015.7256999.

Stolpe, M. and Svanberg, K. (2001). "On the trajectories of the epsilon-relaxation approach for stress-constrained truss topology optimization." *Structural and Multidisciplinary Optimization*, Vol. 21, No. 2, pp. 140-151, DOI: 10.1007/s001580050178.

Talaei, A. S., Nasrollahi, A., and Ghayekhloo, M. (2016). "An automated approach for optimal design of prestressed concrete slabs using PSOHS." *KSCE Journal of Civil Engineering*, pp. 1-10, DOI: 10.1007/s12205-016-1126-9, DOI: 10.1007/s12205-016-1126-9.

Tan, Y. and Zhu, Y. (2010). *Fireworks Algorithm for Optimization*, In Fireworks Algorithm for Optimization Eds Tan, Y., Shi, Y. & Tan, K. pp. 355-364. Springer Berlin Heidelberg.

Tanabe, R. and Fukunaga, A. (2013). "Evaluating the performance of SHADE on CEC 2013 benchmark problems." *Evolutionary Computation (CEC), 2013 IEEE Congress on,* pp. 1952-1959, DOI: 10.1109/CEC.2013.6557798.

Tanabe, R. and Fukunaga, A. S. (2014). "Improving the search performance of SHADE using linear population size reduction." *Evolutionary Computation (CEC), 2014 IEEE Congress on*, pp. 1658-1665, DOI: 10.1109/CEC.2014.6900380.

Teh, Y. S. and Rangaiah, G. P. (2003). "Tabu search for global optimization of continuous functions with application to phase equilibrium calculations." *Computers & Chemical Engineering*, Vol. 27, No. 11, pp. 1665-1679, DOI: 10.1016/S0098-1354(03)00134-0.

Wang, Y., Cai, Z., and Zhang, Q. (2012). "Enhancing the search ability of differential evolution through orthogonal crossover." *Information Sciences*, Vol. 185, No. 1, pp. 153-177, DOI: 10.1016/j.ins.2011.09.001.

Yang, B., Zhang, Q., and Li, H. (2015). "Solving truss topological optimization with discrete design variables via swarm intelligence." *KSCE Journal of Civil Engineering*, Vol. 19, No. 4, pp. 952-963, DOI: 10.1007/s12205-015-0501-2.

Yang, B., Zhang, Q., and Zhou, Z. (2015). "Solving truss topological optimization via Swarm Intelligence." *KSCE Journal of Civil Engineering*, Vol. 19, No. 7, pp. 1962-1972, DOI: 10.1007/s12205-015-0218-2.

Yang, X.-S. and Deb, S. (2010). "Engineering optimisation by cuckoo search." *International Journal of Mathematical Modelling and Numerical Optimisation*, Vol. 1, No. 4, pp. 330-343, DOI: 10.1504/IJMMNO.2010.03543.

Yang, X. S., and Hossein Gandomi, A. (2012). "Bat algorithm: a novel approach for global engineering optimization." *Engineering Computations*, Vol. 29, No. 5, pp. 464-483, DOI: 10.1108/02644401211235834.

Yong, W. and Zixing, C. (2012). "Combining multiobjective optimization with differential evolution to solve constrained optimization problems." *IEEE Transactions on Evolutionary Computation*, Vol. 16. No. 1, pp. 117-134, DOI: 10.1109/TEVC.2010.2093582.

Yong, W., Zixing, C., and Qingfu, Z. (2011). "Differential evolution with composite trial vector generation strategies and control parameters." *IEEE Transactions On Evolutionary Computation*, Vol. 15, No. 1, pp. 55-66, DOI: 10.1109/TEVC.2010.2087271.

Zhang, J. and Sanderson, A.C. (2009). "JADE: Adaptive differential evolution with optional external archive." *IEEE Transactions on Evolutionary Computation*, Vol. 13, No. 5, pp. 945-958, DOI: 10.1109/TEVC.2009.2014613.

Zhao, H., Zhao, M., and Zhu, C. (2016). "Reliability-based optimization of geotechnical engineering using the artificial bee colony algorithm." *KSCE Journal of Civil Engineering*, Vol. 20, No. 5, pp. 1728-1736, DOI: 10.1007/s12205-015-0117-6.