Computers and Structures 195 (2018) 99–112

Contents lists available at ScienceDirect

Computers and Structures

journal homepage: www.elsevier.com/locate/compstruc

An adaptive hybrid evolutionary firefly algorithm for shape and size optimization of truss structures with frequency constraints

Qui X. Lieu, Dieu T.T. Do, Jaehong Lee *

Department of Architectural Engineering, Sejong University, 209 Neungdong-ro, Gwangjin-gu, Seoul 05006, Republic of Korea

ARTICLE INFO

Article history:
Received 6 March 2017
Accepted 12 June 2017

Keywords:
Adaptive hybrid evolutionary firefly algorithm (AHEFA)
Differential evolution (DE)
Firefly algorithm (FA)
Elitist technique
Truss structures
Frequency constraints

ABSTRACT

This paper presents a novel adaptive hybrid evolutionary firefly algorithm (AHEFA) for shape and size optimization of truss structures under multiple frequency constraints. This algorithm is a hybridization of the differential evolution (DE) algorithm and the firefly algorithm (FA). An automatically adapted parameter is utilized to select an appropriate mutation scheme for an effective trade-off between the global and local search abilities. An elitist technique is applied to the selection phase to choose the best individuals. Accordingly, the convergence rate is significantly improved with the high solution accuracy. Six numerical examples are examined for the validity of the present algorithm.

1. Introduction

Since the pioneering paper on shape and size optimization problems of truss structures with multiple frequency constraints was published by Bellagamba and Yang [1], researches in this field have been rapidly developed and have attracted many scientists all over the world during over the past decades. For instance, an optimality algorithm based on uniform Lagrangian density for resizing and a scaling procedure to locate the constraint boundary were used by Grandhi and Venkayya [2]. Wang et al. [3] developed an optimality criterion under a single constraint based on differentiation of the Lagrangian function. The finite element force method associated with the sequential quadratic programming (SQP) approach was delivered by Sedaghati et al. [4,5]. Wei et al. [6] proposed the niche hybrid parallel genetic algorithm (NHPGA) to save the computational cost and enhance the solution accuracy. Gomes [7] employed the particle swarm optimization (PSO) algorithm for simultaneous shape and size optimization problems of truss structures with multiple frequency constraints. Miguel and Fadel Miguel [8] utilized the harmony search (HS) and the firefly algorithm (FA) for addressing this type of problems. Based on a hybridization of the enhanced charged system search (CSS) and the big bang-big crunch (BBBC) approaches with trap recognition capability, Kaveh and Zolghadr [9] suggested the so-called CSS-BBBC algorithm. Subsequently, these authors also additionally proposed the democratic PSO [10], the cyclical parthenogenesis algorithm (CPA) [11] and the tug of war optimization (TWO) [12]. The hybrid optimality criterion (OC) and genetic algorithm (GA) method for solving such problems were reported by Zuo et al. [13]. Khatibinia and Naseralavi [14] introduced the orthogonal multi-gravitational search algorithm (OMGSA). Kaveh and Ghazaan [15] combined the aging leader and challengers (ALC) with the PSO to produce a novel algorithm named the ALC-PSO. By integrating the harmony search-based mechanism into the ALC-PSO, the new so-called HALC-PSO was also exhibited by the above authors. More recently, Tejani et al. [16] proposed an adaptive symbiotic organisms search (SOS) algorithm. Ho-Huu et al. [17] suggested another variant of the DE named the roulette wheel selection-elitist-differential evolution (ReDE). An improved evolution (IDE) algorithm was released by Ho-Huu et al. [18] as well. Farshchin et al. successfully developed the multi-class teaching–learning-based optimization (MC-TLBO) [19] and the school-based optimization (SBO) [20], and so on.

As indicated in [2], natural frequencies and constraints of the foregoing optimization problems are highly nonlinear, non-convex and greatly sensitive to shape changes and member sizes of truss structures. Mode shapes and corresponding obtained frequencies may switch during the optimization process, and this causes many difficulties for the convergence. Therefore, a proper optimization algorithm utilized to solve such problems is really

* Corresponding author.
E-mail addresses: lieuxuanqui@gmail.com (Q.X. Lieu), thanhdieu0801@gmail.com (D.T.T. Do), jhlee@sejong.ac.kr (J. Lee).

https://doi.org/10.1016/j.compstruc.2017.06.016
0045-7949/

essential. Obviously, restricted applications of gradient-based algorithms could be recognized due to their disadvantages. In particular, these methods always require sensitivity analyses relating to derivatives of the objective function and constraints with regard to each of design variables whose performances are relatively complex and expensive, even impossible in many cases. Moreover, local optimum solutions may be trapped if a given set of initial values in a specified search space is not chosen carefully. For these shortcomings, non-gradient-based algorithms, known as metaheuristic approaches, have been rapidly developed and have received much considerable attention of many researchers. These algorithms use stochastic searching techniques to randomly select potential solutions within a predetermined search space, and thus they are completely free from sensitivity analyses and almost demand less mathematical analyses. Consequently, these approaches are easy to perform and very robust in finding global optimal solutions for optimization problems concerning highly nonlinear and non-convex properties. However, the computational cost of the search process is fairly expensive and often time-consuming as an optimal solution must be looked for over the entire search space without any definite directions. Additionally, the stochastic searching techniques differently defined in each method are also one of the key factors that affect the solution accuracy.

Among non-gradient-based optimization methods, differential evolution (DE) algorithm which was first developed by Storn and Price [21] is one of the commonly used nature-inspired population-based approaches. This algorithm has been broadly applied to a wide range of disciplines due to its effectiveness and robustness in searching a global optimal solution in continuous spaces. A large number of its variants have been introduced later to improve several drawbacks relating to the computational speed, convergence properties and the optimal solution accuracy [22–28]. Another highly effective way to improve the DE is to combine it with other available optimization algorithms for benefiting from the synergy and overcoming individual drawbacks of each of hybridized original algorithms. Accordingly, many hybrid DE methods have been developed by mixing it with other global search approaches like GA [29,30], HS [31], biogeography-based optimization (BBO) [32], gradient based real-coded population-based incremental learning (RCPBIL) algorithm [33], ant colony optimization (ACO) [34], bacterial foraging-based optimization (BFO) [35], PSO [36], invasive weed optimization (IWO) [37], simulated annealing (SA) [38,39], covariance matrix adaptation evolutionary strategies (CMA-ES) [40,41], fireworks algorithm (FWA) [42], gravitational search algorithm (GSA) [43], and artificial bee colony (ABC) algorithm [44], etc. Some other hybrid approaches of the DE integrated with local search methods could be found in [45–48]. For a more detailed discussion, the interested reader should refer to a comprehensive review of the DE particularly surveyed by Das et al. [49].

As one of the alternative population-based algorithms, Yang [50–52] developed a new approach named the firefly algorithm (FA). As stated by the author, the method can be considered as a generalization to the PSO, DE, and SA algorithms by setting its some parameters with specific values. The above three algorithms are thus only special cases of the FA. Consequently, the FA inherits the advantages of all those algorithms and can perform effectively. Indeed, this approach has captured the attention of many researchers and been successfully applied to the hardest optimization problems [53]. The interested readers are encouraged to consult the above references for a more detailed description of the algorithm as well as its applications. Although the FA is of the above good features, the computational cost of the FA is still relatively expensive since its searching technique is also metaheuristic like the classical DE and other non-gradient-based methods. For improvements, some hybrid firefly algorithms have been proposed by blending with other methods, i.e. the Lévy flight search [54], GA [55,56], ant colony [57], DE [58,59] and so forth, with an essential aim that such hybrid algorithms will outperform in terms of the solution accuracy and convergence rate from their collaborations.

Although all the aforementioned works have been extensively applied to various optimization problems in many engineering and scientific areas with fairly prominent achievements [53,49], hybrid algorithms in the framework of the DE and the FA that have the ability to improve the convergence speed for the reduction of computational cost and time-consuming process as well as to enhance the solution accuracy appear to be very limited apart from two researches of Abdullah et al. [58,59]. These authors proposed the so-called hybrid evolutionary firefly algorithm (HEFA) for the evaluation of nonlinear biological model parameters. In that algorithm, the population obtained in the previous generation is sorted according to the fitness and then divided into two sub-populations, i.e. potential and weak ones. The mutation operator "rand/1" in the DE is applied to the weak sub-population to enhance the solution search capability while the other is substantially implemented in the manner of the FA. All the rest of the procedure absolutely follows the FA without any other integrations of the DE. Although results showed that this algorithm requires less computation time than the PSO, FA and Nelder-Mead for finding good solutions, its convergence to the optimal solution still demands a highly large number of finite element (FE) analyses that will lead to the increase of the computational cost. Furthermore, its validation to shape and size optimization problems of truss structures with multiple frequency constraints has still not been examined yet so far. Therefore, this work is executed to deal with the afore-discussed issues.

In this study, a novel adaptive hybrid evolutionary firefly algorithm (AHEFA) as a hybridization of the DE method and the FA is proposed for the improvement on the convergence speed and the solution accuracy. An automatically adapted parameter computed from the deviation of objective function between the best individual and the whole population in the previous generation is utilized to select an appropriate mutation scheme for the performance in the mutation phase. The balance between the global exploration and local exploitation abilities is hence enhanced effectively. Accordingly, the convergence speed and the accuracy of achieved optimal solutions are improved considerably. Furthermore, an elitist technique is employed for the selection phase to choose a new population for the next generation containing the best individuals from the mixture of the target and trial individuals. This technique helps to speed up the convergence rate of the proposed algorithm as well. The validity of the AHEFA is then confirmed by testing for shape and size optimization problems of truss structures with multiple frequency constraints. Optimal results attained by the proposed method are compared with those given by other algorithms in the literature.

The remainder of this article is constructed as follows. The statement of shape and size optimization problem of truss structures with multiple frequency constraints is built up in Section 2. A detailed discussion of the DE approach, the FA, and the proposed AHEFA is provided in Section 3. Section 4 presents six most widely investigated benchmark numerical examples to illustrate the effectiveness and robustness of the AHEFA. Finally, Section 5 ends with conclusions.

2. Problem statement

For optimization problems of truss structures under multiple frequency constraints, the aim is to design member sizes or/and the shape of the structure so that its weight is minimized. In which, member cross-sectional areas or/and nodal coordinates are consid-

ered as continuous design variables. The connectivity information of the structure is predetermined and assumed to be unchanged during the optimization process. Additionally, each variable is designed within a predefined feasible region. This problem can thus be mathematically stated as

$$\begin{aligned} &\text{Minimize}: \quad f(\mathbf{A},\mathbf{x}) = \sum_{i=1}^{m} \rho_i A_i L_i(x_j), \\ &\text{Subject to}: \begin{cases} \omega_k \leqslant \bar{\omega}_k, \\ \omega_l \geqslant \bar{\omega}_l, \\ A_{i,\min} \leqslant A_i \leqslant A_{i,\max}, \\ x_{j,\min} \leqslant x_j \leqslant x_{j,\max}, \end{cases} \end{aligned} \tag{1}$$

where $\mathbf{A} = \{A_1, \ldots, A_i, \ldots, A_m\}$ and $\mathbf{x} = \{x_1, \ldots, x_j, \ldots, x_n\}$ are the design variable vectors of cross-sectional areas and nodal coordinates, respectively; $(m+n)$ is the entire number of design variables; m is the total number of members of the structure; n symbolizes the whole number of constraints on nodal coordinates; ρ_i and L_i stand for the material density and the length of the ith member, respectively; ω_k and ω_l denote the kth and lth natural frequencies of the structure, respectively; $\bar{\omega}_k$ and $\bar{\omega}_l$ denote the upper and lower bounds corresponding to ω_k and ω_l, respectively; $A_{i,\min}$ and $A_{i,\max}$ are the lower and upper bounds of A_i, respectively; $x_{j,\min}$ and $x_{j,\max}$ are the lower and upper bounds of x_j, respectively.

In order to transform the constrained optimization problem expressed in Eq. (1) into an unconstrained one, the penalty function method which is one of the most frequently used constraint-handling approaches [9,60] is utilized in this work. Consequently, the above problem can be rewritten as follows

$$\begin{aligned} &\text{Minimize}: \quad f_{\text{cost}}(\mathbf{A},\mathbf{x}) = (1+\varepsilon_1 \upsilon)^{\varepsilon_2} f(\mathbf{A},\mathbf{x}), \\ &\upsilon = \sum_{r=1}^{p} \max\{0, g_r(\mathbf{A},\mathbf{x})\}, \end{aligned} \tag{2}$$

where υ stands for the sum of violations of the design constraints; $g_r(\mathbf{A},\mathbf{x})$ stands for the rth constraint; p denotes the number of constraints; the values of parameters ε_1 and ε_2 depend on the exploration and exploitation rates of the search space. In this study, ε_1 is taken as 1, while ε_2 is set to be 1.5 at the beginning of the iteration and monotonously increased by 0.05 in each iteration step until it reaches 3.

3. Optimization algorithm

3.1. Differential evolution algorithm

For the first iteration, four basic steps are implemented in the classical DE including (i) initialization, (ii) mutation, (iii) crossover, and (iv) selection. While only the last three steps are repeatedly done in subsequent iterations until satisfying stopping criteria which depend on each individual optimization problem and are defined by users. A brief introduction to the above steps is provided in detail below.

- *Initialization*

An optimization problem including np individuals is randomly initialized in a given continuous search space. In which, the ith individual $(i = 1, 2, \ldots, np)$ is a vector consisting of d design variables and is given by the following form

$$x_{i,j}^{t=0} = x_{\min,j} + rand_{ij}[0,1]\left(x_{\max,j} - x_{\min,j}\right), \quad j = 1, 2, \ldots, d, \tag{3}$$

where $rand_{ij}[0,1]$ is a uniformly distributed random number within [0,1] and is independently created for each term in the ith vector; $x_{\min,j}$ and $x_{\max,j}$ are the predefined minimum and maximum bounds of $x_{i,j}$, respectively. Note that the superscript $(t=0)$ stands for $x_{i,j}$ at the first iteration. Generally, let us adopt the following notation for defining the ith individual vector at the t current iteration $(t = 0, 1, \ldots, t_{\max})$,

$$\mathbf{x}_i^t = \left\{x_{i,1}^t, x_{i,2}^t, \ldots, x_{i,j}^t, \ldots, x_{i,d}^t\right\}. \tag{4}$$

- *Mutation*

After the first step, a mutant vector $\mathbf{v}_i^t$ is generated from the target vectors $\mathbf{x}_i^t$ at the current iteration via mutation. Five most commonly utilized mutation schemes are given as follows

$$\text{rand}/1: \quad \mathbf{v}_i^t = \mathbf{x}_{R_1}^t + F\left(\mathbf{x}_{R_2}^t - \mathbf{x}_{R_3}^t\right), \tag{5a}$$

$$\text{best}/1: \quad \mathbf{v}_i^t = \mathbf{x}_{\text{best}}^t + F\left(\mathbf{x}_{R_1}^t - \mathbf{x}_{R_2}^t\right), \tag{5b}$$

$$\text{current}-\text{to}-\text{best}/1: \quad \mathbf{v}_i^t = \mathbf{x}_i^t + F\left(\mathbf{x}_{\text{best}}^t - \mathbf{x}_i^t\right) + F\left(\mathbf{x}_{R_1}^t - \mathbf{x}_{R_2}^t\right), \tag{5c}$$

$$\text{rand}/2: \quad \mathbf{v}_i^t = \mathbf{x}_{R_1}^t + F\left(\mathbf{x}_{R_2}^t - \mathbf{x}_{R_3}^t\right) + F\left(\mathbf{x}_{R_4}^t - \mathbf{x}_{R_5}^t\right), \tag{5d}$$

$$\text{best}/2: \quad \mathbf{v}_i^t = \mathbf{x}_{\text{best}}^t + F\left(\mathbf{x}_{R_1}^t - \mathbf{x}_{R_2}^t\right) + F\left(\mathbf{x}_{R_3}^t - \mathbf{x}_{R_4}^t\right), \tag{5e}$$

where R_1, R_2, R_3, R_4 and R_5 are the randomly chosen integer numbers within $[1, np]$ in such a way that all of those are totally different from the index i; the scale factor F is randomly selected in $(0, 1]$ to control the deviation from two target vectors, and $\mathbf{x}_{\text{best}}^t$ is the best individual vector corresponding to the best objective function at the current iteration.

It can be seen that $v_{i,j}^t$ obtained by one of the above mutation operators may be violated its lower and upper bounds. Therefore, in order to strictly satisfy the boundary constraints, it is returned to the search space by the following formula

$$v_{i,j}^t = \begin{cases} 2x_{\min,j} - v_{i,j}^t, & \text{if } v_{i,j}^t < x_{\min,j}, \\ 2x_{\max,j} - v_{i,j}^t, & \text{if } v_{i,j}^t > x_{\max,j}, \\ v_{i,j}^t, & \text{otherwise}. \end{cases} \tag{6}$$

- *Crossover*

Next, crossover scheme is used to enhance the diversity of individual vectors in the current population. Other variants of this scheme could be found in [61,62]. Owing to their limitations for several problems as pointed out by Das et al. [49], binomial crossover has been used more extensively. According to this scheme, the ith trial vector $\mathbf{u}_i^t = \left\{u_{i,1}^t, u_{i,2}^t, \ldots, u_{i,j}^t, \ldots, u_{i,d}^t\right\}$ is produced by mixing the target vector $\mathbf{x}_i^t$ and the mutant vector $\mathbf{v}_i^t$ as

$$u_{i,j}^t = \begin{cases} v_{i,j}^t, & \text{if } j = K \quad \text{or} \quad rand_{ij}[0,1] \leqslant Cr, \\ x_{i,j}^t, & \text{otherwise}, \end{cases} \tag{7}$$

where K is an integer number randomly chosen in $[1, np]$, and Cr is the crossover control parameter in $[0.7, 1]$.

- *Selection*

This step aims to select better individuals in the current population for the performance of the next iteration by comparing each pair of the objective function values $f(\cdot)$ for the individual vectors of $\mathbf{x}_i^t$ and $\mathbf{u}_i^t$. This strategy can be expressed as

$$\mathbf{x}_i^{t+1} = \begin{cases} \mathbf{u}_i^t, & \text{if } f(\mathbf{u}_i^t) \leqslant f(\mathbf{x}_i^t), \\ \mathbf{x}_i^t, & \text{otherwise}. \end{cases} \tag{8}$$

3.2. Firefly algorithm

Based on the flashing patterns and behavior of fireflies, Yang firstly introduced the firefly algorithm (FA) [50–52] which is also one of the nature-inspired metaheuristic algorithms. In order to build up formulas for the FA, three flashing characteristics of the fireflies are idealized as follows:

(i) All fireflies are unisex. This means that the attraction of a firefly to others is greatly independent of their sex;
(ii) A firefly's light intensity characterizes its attractiveness. The stronger the light intensity is the more attractive it becomes. Consequently, the less attractive fireflies will move toward the more attractive ones. Both features are inversely proportional to their distance;
(iii) A firefly's light intensity is affected or determined by the landscape of the objective function.

The main procedure of the FA is briefly introduced as follows. Firstly, a population consisting of *np* fireflies is randomly generated as the DE, where each firefly denotes an individual vector consisting of *d* design variables. Next, a new position of the *i*th firefly is determined by the following equation

$$\mathbf{v}_i^t = \mathbf{x}_i^t + \beta(\mathbf{x}_k^t - \mathbf{x}_i^t) + \alpha^t \boldsymbol{\epsilon}_i, \tag{9}$$

where the second term stands for the attraction; the third term is randomization with α being a number randomly chosen in $[0,1]$, and $\boldsymbol{\epsilon}_i$ is a random number vector given according to the Gaussian distribution, i.e. $\boldsymbol{\epsilon}_i = rand_{i,j}[0,1] - 0.5$. As suggested by Yang [50–52], in order to improve the convergence performance, α^t is gradually decreased based on the formula as $\alpha^{t+1} = (1 - 0.98)\alpha^t$ with $\alpha^0 = 0.5$. Additionally, the attractiveness β is given by the following expression

$$\beta = \beta_0 e^{-\gamma r_{i,k}^2}, \tag{10}$$

where the subscript k $(k = 1, 2, \ldots, np)$ denotes the kth individual whose light intensity is stronger than that of the ith individual. In other words, its objective function value is better; β_0 is the initial attractiveness at $r_{i,k} = 0$ and is set to be 1; γ is the light absorption coefficient in $[0, \infty)$ and depends on every particular problem to be minimized. In this study, the appropriate value of γ is suggested as 1, and $r_{i,k}$ stands for the Euclidean distance between two fireflies $\mathbf{x}_i^t$ and $\mathbf{x}_k^t$ in the current population and is defined by

$$r_{i,k} = \left\|\mathbf{x}_i^t - \mathbf{x}_k^t\right\| = \sqrt{\sum_{j=1}^{d}\left(x_{i,j}^t - x_{k,j}^t\right)^2}. \tag{11}$$

If an updated individual exceeds its boundary constraint, it will be penalized to come back the specified search space as follows

$$v_{i,j}^t = \begin{cases} v_{\min,j}, & \text{if } v_{i,j}^t < x_{\min,j}, \\ v_{\max,j}, & \text{if } v_{i,j}^t > x_{\max,j}, \\ v_{i,j}^t, & \text{otherwise.} \end{cases} \tag{12}$$

Apart from the first step, all performances are repeatedly carried out until the optimization process ends.

3.3. Adaptive hybrid evolutionary firefly algorithm

In fact, almost all global optimization algorithms must seek an optimal solution over a given entire search domain, a considerable amount of computational cost and time-consuming search are thus always required. The solution accuracy and the convergence speed are significantly affected by the exploration and exploitation capabilities of stochastic searching techniques, respectively. The exploration ability is very good at the global search but the convergence rate is slow. Conversely, the exploitation ability is good at the convergence speed, yet it focuses on only searching feasible solutions in a local region. For an effective balance of these two abilities to improve the convergence speed and the solution accuracy, a novel adaptive hybrid evolutionary firefly algorithm (AHEFA) hybridized from the DE algorithm and the FA is proposed in this study.

Substituting β from Eq. (10) into Eq. (9), replacing $\alpha^t \boldsymbol{\epsilon}_i$ in Eq. (9) with $\alpha\left(\mathbf{x}_{R_2}^t - \mathbf{x}_{R_3}^t\right)$, and then collecting the coefficients of $\mathbf{x}_i^t$, the equation is now rewritten as

$$\mathbf{v}_i^t = \left(1 - \beta_0 e^{-\gamma r_{i,R_1}^2}\right)\mathbf{x}_i^t + \beta_0 e^{-\gamma r_{i,R_1}^2}\mathbf{x}_{R_1}^t + \alpha\left(\mathbf{x}_{R_2}^t - \mathbf{x}_{R_3}^t\right), \tag{13}$$

where three integer numbers R_1, R_2 and R_3 $(R_1 \neq R_2 \neq R_3)$ are randomly formed from the mutation strategy "rand/1" in the DE. R_1 corresponds to the best objective function $f\left(\mathbf{x}_{R_1}^t\right)$ which is compared with $f\left(\mathbf{x}_{R_2}^t\right)$ and $f\left(\mathbf{x}_{R_3}^t\right)$; β_0 and γ are taken as those given in the FA; α serves as the scale factor F in the DE, and both are suggested to be 0.8.

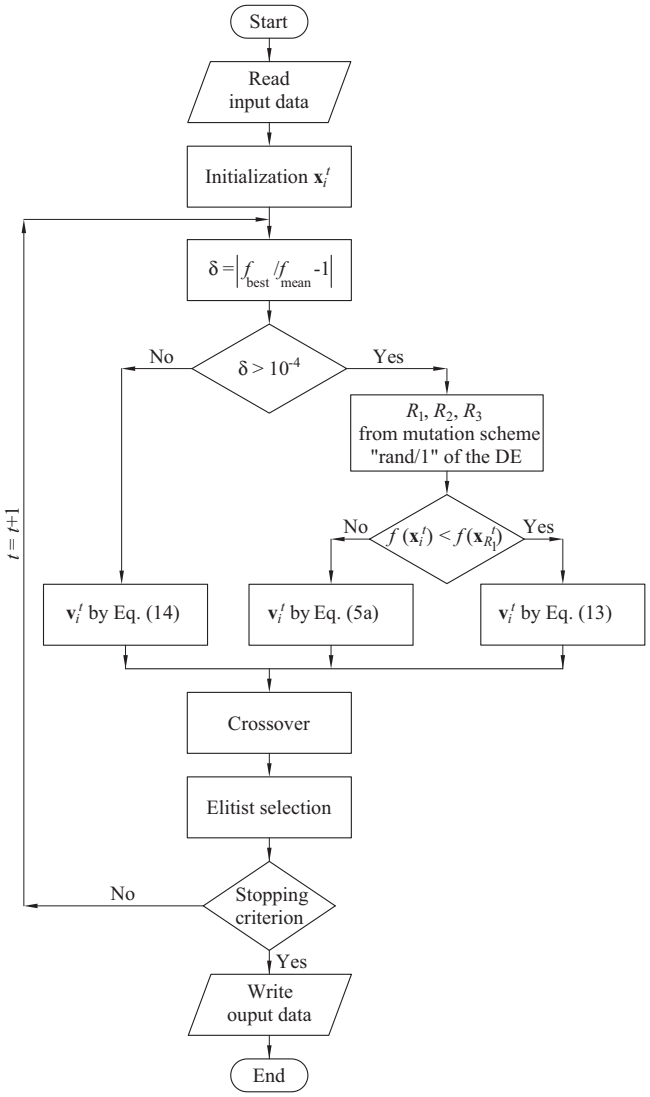

Fig. 1. The AHEFA flowchart.

Table 1
Material properties, cross-sectional area bounds, and frequency constraints for different problems.

Problem	Young's modulus E (N/m^2)	Material density ρ (kg/m^3)	Cross-sectional area bounds (m^2)	Frequency constraints (Hz)
10-bar planar truss	6.98×10^{10}	2770	$0.645 \times 10^{-4} \leqslant A$	$\omega_1 \geqslant 7, \omega_2 \geqslant 15, \omega_3 \geqslant 20$
72-bar space truss	6.98×10^{10}	2770	$0.645 \times 10^{-4} \leqslant A$	$\omega_1 = 4, \omega_3 \geqslant 6$
120-bar dome truss	2.1×10^{11}	7971.81	$0.0001 \leqslant A \leqslant 0.01293$	$\omega_1 \geqslant 9, \omega_2 \geqslant 11$
200-bar planar truss	2.1×10^{11}	7860	$0.1 \times 10^{-4} \leqslant A$	$\omega_1 \geqslant 5, \omega_2 \geqslant 10, \omega_3 \geqslant 15$
37-bar planar truss	2.1×10^{11}	7800	$1 \times 10^{-4} \leqslant A$	$\omega_1 \geqslant 20, \omega_2 \geqslant 40, \omega_3 \geqslant 60$
52-bar dome truss	2.1×10^{11}	7800	$0.0001 \leqslant A \leqslant 0.001$	$\omega_1 \leqslant 15.9155, \omega_2 \geqslant 28.6479$

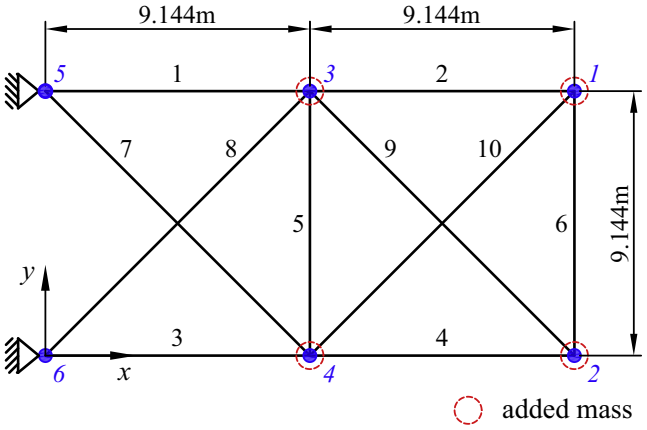

Fig. 2. The 10-bar planar truss.

If the ith firefly's attractiveness $\mathbf{x}_i^t$ is less than the R_1^{th}'s one, its new location is updated according to the above equation, otherwise it is computed by Eq. (5a) as in the DE.

Obviously, the term $\alpha\left(\mathbf{x}_{R_2}^t - \mathbf{x}_{R_3}^t\right)$ produces a wider search space for the FA, the global search ability is hence enhanced significantly. However, the convergence speed of the problem becomes slowly due to the decreased local search capacity. This issue is quite similar to the mutation operator "rand/1" in the DE [63]. As can be found that the diversity of mutant vectors in many problems is more stable after many iterations, this leads the search space to be narrowed gradually. Accordingly, the local search ability in those feasible regions is dramatically improved if the mutation operator "best/1" is employed instead. The aim of this scheme is to pay attention of other individuals in the current population only to the best individual, and thus the aforementioned issue can be handled simply. By neglecting the third term in Eq. (13) and replacing $\mathbf{x}_{R_1}^t$ with $\mathbf{v}_i^t$, which is given as Eq. (5b) of the DE. The new solutions are now updated by the following equation

$$\mathbf{v}_i^t = \left(1 - \beta_0 e^{-\gamma r_{i,\text{best}}^2}\right)\mathbf{x}_i^t + \beta_0 e^{-\gamma r_{i,\text{best}}^2}\left[\mathbf{x}_{\text{best}}^t + F\left(\mathbf{x}_{R_1}^t - \mathbf{x}_{R_2}^t\right)\right], \tag{14}$$

where $r_{i,\text{best}}$ is the distance between the ith individual and the best one.

In order to effectively switch between the global exploration and local exploitation abilities in seeking the global optimal solution and enhancing the convergence rate as discussed above, an automatically adapted parameter δ is introduced to control this adaption as follows

$$\delta = |f_{\text{mean}}/f_{\text{best}} - 1|, \tag{15}$$

where f_{mean} is the mean objective function value of the whole population, while f_{best} is the objective function value of the best individual in the previous generation. This parameter changes during the optimization process and tends towards the gradual reduction. Its value is usually very large for a few first iterations. Nevertheless, it is smaller and more stable for many subsequent iterations when the individuals in the population become less diversity. If δ is greater than a specified value ε, either Eq. (13) or Eq. (5a) is employed, otherwise Eq. (14) is utilized. Note that if ε is too small, the convergence speed is slow since the global search is mainly performed over the whole search process. On the contrary, the problem converges more quickly due to the applied local search, but local solutions may occur. The value ε is appropriately set in this study as 10^{-4}.

After this adaptive mutation scheme, the check for the boundary constraint of mutant vectors and the performance in the crossover phase are similarly executed as the DE.

Next, an elitist selection technique suggested by Padhye et al. [64] is utilized. As reported by the authors, this technique helps

Table 2
Comparison of optimal results of the 10-bar planar truss obtained by different algorithms.

Design variable	PSO	HS	CSS-BBBC	ReDE	This study		
A_i (cm^2)	[7]	[8]	[9]	[17]	DE	FA	AHEFA
1	37.712	34.282	35.274	35.1565	35.1623	35.8759	35.1714
2	9.959	15.653	15.463	14.7605	14.7263	15.1591	14.7203
3	40.265	37.641	32.110	35.1187	34.9776	34.6947	35.1074
4	16.788	16.058	14.065	14.7275	14.7206	14.9899	14.6986
5	11.576	1.069	0.645	0.6450	0.6450	0.6450	0.6451
6	3.955	4.740	4.880	4.5558	4.5600	4.5172	4.5593
7	25.308	22.505	24.046	23.7199	23.6961	23.6926	23.7330
8	21.613	24.603	24.340	23.6304	23.6730	23.1115	23.6795
9	11.576	12.867	13.343	12.3827	12.4643	12.0852	12.3987
10	11.186	12.099	13.543	12.4580	12.4801	12.7050	12.4231
Best weight (kg)	537.98	534.99	529.09	524.4542	524.4561	524.6413	524.4516
Number of FE analyses	–	20000	–	8300	11460	13060	5860
Worst weight (kg)	–	–	–	530.6431	530.6683	532.0342	530.9038
Average weight (kg)	540.89	537.68	–	524.7635	525.0996	526.5281	525.1623
Standard deviation	6.84	2.49	–	1.1129	1.8564	2.5827	1.9155

Table 3
The first eight optimal natural frequencies of the 10-bar planar truss obtained by different algorithms.

Frequency number	PSO	HS	CSS-BBBC	ReDE	This study		
	[7]	[8]	[9]	[17]	DE	FA	AHEFA
1	7.000	7.0028	7.000	7.0000	7.0000	7.0000	7.0000
2	17.786	16.7429	16.119	16.1924	16.1854	16.2090	16.1920
3	20.000	20.0548	20.075	20.0000	20.0000	20.0000	20.0000
4	20.063	20.3351	20.457	20.0002	20.0002	20.0000	20.0000
5	27.776	28.5232	29.149	28.5517	28.5784	28.4053	28.5551
6	30.939	29.2911	29.761	–	29.0068	29.1042	28.9588
7	47.297	49.0342	47.950	–	48.5608	48.7728	48.5777
8	52.286	51.7451	51.215	–	51.0736	51.3196	51.0712

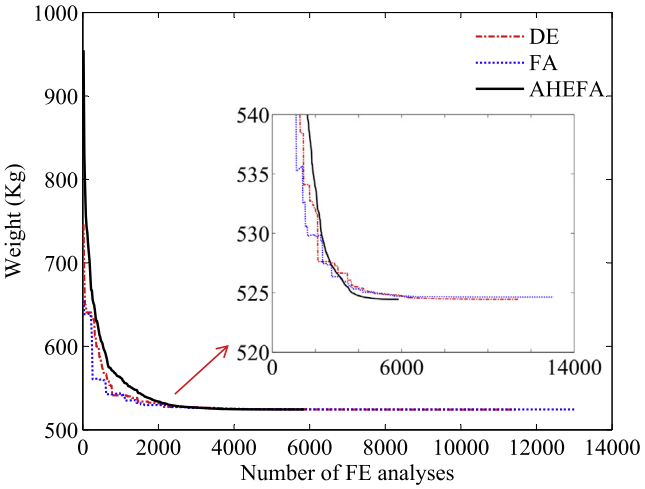

Fig. 3. The weight convergence histories of the 10-bar planar truss obtained using the DE, FA and AHEFA.

to choose the best individuals for the next iteration, and the convergence rate is hence better. This strategy is succinctly summarized as follows. Firstly, the target individuals $x_{i,j}^t$ and the trial individuals $u_{i,j}^t$ are mixed together to create a population including $2np$ individuals. Then, np best individuals $\mathbf{x}_i^{t+1}$ are selected from the above population to generate a new population for the next iteration. Obviously, this population inherits candidate target individuals that can be ignored if the original selection phase only based on the comparison of each pair of $x_{i,j}^t$ with $u_{i,j}^t$ is employed. By this way, the solution search space is reduced, the convergence speed is thus improved considerably. The flowchart of the proposed AHEFA is depicted in Fig. 1.

4. Numerical examples

In this section, six most commonly used numerical examples including four ones regarding size optimization and the others concerning shape and size optimization are explored by the DE, FA and the proposed AHEFA. For all investigated examples, the population size np is set as 20. The iterative process of all the presented methods is terminated when either the value of δ is less than or equal to 10^{-6} or the maximum number of FE analyses reaches 20000. Otherwise, it will be continued until satisfying the above stopping criterion. Material properties, cross-sectional area bounds, and frequency constraints are tabulated in Table 1. The two-node linear bar element [65] is utilized for FE analysis and code structures are performed by Matlab program. 10 independent runs are implemented for each problem due to the stochastic nature of metaheuristic algorithms. Subsequently, statistical results are presented in terms of the best weight and the corresponding number of FE analyses, the worst weight, average weight and standard deviation. Only design variable values achieved from the best run are reported for comparison with those given by other optimization approaches to illustrate the accuracy and robustness of the proposed algorithm.

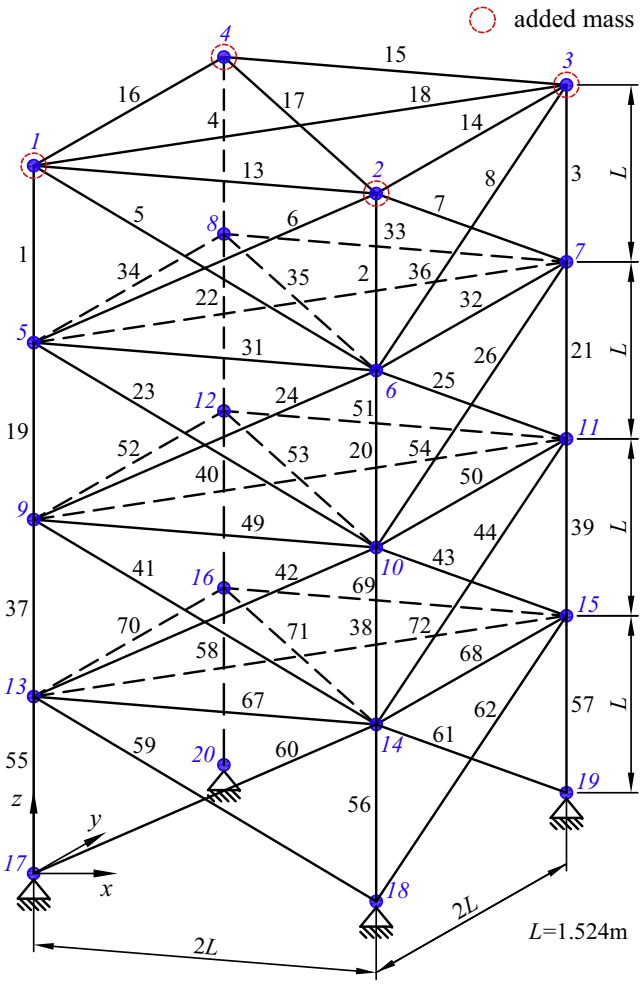

Fig. 4. The 72-bar space truss.

4.1. Size optimization

4.1.1. 10-bar planar truss

As the simplest example, a planar truss consisting of 10 bars shown in Fig. 2 is first exhibited herein. Cross-sectional areas of all bars are treated as continuous design variables. Four non-structural masses of 454 kg are located at corresponding four free

Table 4
Comparison of optimal results of the 72-bar space truss obtained by different algorithms.

Design variable	Member group	PSO	HS	CSS-BBBC	HALC-PSO	ReDE	This study		
A_i (cm^2)		[7]	[8]	[9]	[15]	[17]	DE	FA	AHEFA
1	1–4	2.987	3.6803	2.854	3.3437	3.5327	3.5964	3.7495	3.5612
2	5–12	7.849	7.6808	8.301	7.8688	7.8303	7.7600	7.9218	7.8736
3	13–16	0.645	0.6450	0.645	0.6450	0.6453	0.6450	0.6450	0.6450
4	17–18	0.645	0.6450	0.645	0.6450	0.6459	0.6450	0.6450	0.6451
5	19–22	8.765	9.4955	8.202	8.1626	8.0029	7.9011	7.8082	7.9710
6	23–30	8.153	8.2870	7.043	7.9502	7.9135	7.9426	8.0432	7.8928
7	31–34	0.645	0.6450	0.645	0.6452	0.6451	0.6450	0.6450	0.6450
8	35–36	0.645	0.6461	0.645	0.6452	0.6451	0.6450	0.6450	0.6451
9	37–40	13.450	11.4510	16.328	12.2668	12.7626	12.7248	13.1948	12.5404
10	41–48	8.073	7.8990	8.299	8.1845	7.9657	8.0114	7.8882	7.9639
11	49–52	0.645	0.6473	0.645	0.6451	0.6452	0.6451	0.6451	0.6459
12	53–54	0.645	0.6450	0.645	0.6451	0.6450	0.6451	0.6450	0.6462
13	55–58	16.684	17.4060	15.048	17.9632	16.9041	16.9809	16.5127	17.1323
14	59–66	8.159	8.2736	8.268	8.1292	8.0434	8.0398	7.9034	8.0216
15	67–70	0.645	0.6450	0.645	0.6450	0.6451	0.6450	0.6450	0.6450
16	71–72	0.645	0.6450	0.645	0.6450	0.6473	0.6451	0.6450	0.6451
Best weight (kg)		328.823	328.334	327.507	327.77	324.2467	324.2436	324.3678	324.2376
Number of FE analyses			50000	–	8000	10840	20000	11920	8860
Worst weight (kg)			–	–	–	324.4811	324.4478	326.0852	325.0968
Average weight (kg)			332.64	–	327.99	324.3219	324.2904	325.1738	324.4109
Standard deviation			2.39	–	0.19	0.0516	0.0564	0.6785	0.2420

Table 5
The first five optimal natural frequencies of the 72-bar space truss obtained by different algorithms.

Frequency number	PSO	HS	CSS-BBBC	HALC-PSO	ReDE	This study		
	[7]	[8]	[9]	[15]	[17]	DE	FA	AHEFA
1	4.000	4.0000	4.0000	4.000	4.0000	4.0000	4.0000	4.0000
2	4.000	4.0000	4.0000	4.000	4.0000	4.0000	4.0000	4.0000
3	6.000	6.0000	6.0040	6.000	6.0001	6.0000	6.0000	6.0000
4	6.219	6.2723	6.2491	6.230	6.2762	6.2738	6.2864	6.2740
5	8.976	9.0749	8.9726	9.041	9.1073	9.1087	9.1453	9.1137

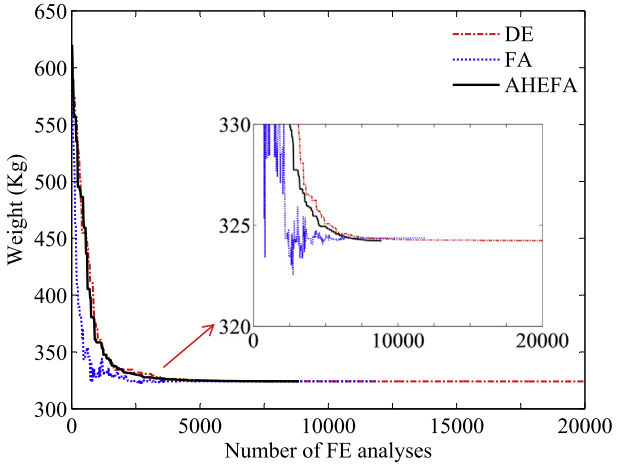

Fig. 5. The weight convergence histories of the 72-bar space truss obtained using the DE, FA and AHEFA.

nodes as indicated in the same figure. Various algorithms have been previously used to optimize this problem such as OC [2,3], SQP [4,5], NHPGA [6], PSO [7], HS and FA [8], CSS-BBBC [9], democratic PSO [10], CPA [11], TWO [12], hybrid OC-GA [13], OMGSA [14], adaptive SOS [16], ReDE [17], IDE [18], MC-TLBO [19], SBO [20].

Table 2 shows a comparison of optimal results obtained by the present algorithms with different approaches. It can be seen that the optimal weight achieved by the proposed AHEFA is better than those of other methods. Table 3 represents the first eight optimal natural frequencies obtained by the present work and different researches. It is clear that all the frequencies strictly satisfy the allowable constraints. Fig. 3 shows the weight convergence histories obtained using the DE, FA and AHEFA for this structure. As observed, the convergence speed of the AHEFA is much faster than the DE and the FA. This algorithm rapidly finds the optimal solution with only 5860 FE analyses, while the others require a larger number of FE analyses for the convergence performance, i.e. DE with 11460 FE analyses, and FA with 13060 FE analyses.

4.1.2. 72-bar space truss

The second example aims to optimize a 72-bar space truss whose geometry and finite element representation are depicted in Fig. 4. Cross-sectional areas of all bars are categorized into 16 groups which are equivalent to 16 design variables as shown in Table 4. Four non-structural masses of 2270 kg are attached to 4 upper nodes. Also, as one of the benchmark problems, this structure has been studied by various researchers that can be listed in the following literature [4,7–9,14–16,11,12,17–20].

Optimal results obtained by the present methods with different algorithms are provided in Table 4 for comparison. As can be observed that the AHEFA yields the smallest weight in comparison with the others. Table 5 shows the first five optimal natural frequencies obtained by the proposed method and different algorithms. None of any violated frequency constraints is observed as reported in the table. The weight convergence histories obtained using the DE, FA and AHEFA for this example is given in Fig. 5. It is found that the AHEFA needs far fewer the number of FE analyses to gain the optimal solution than the others.

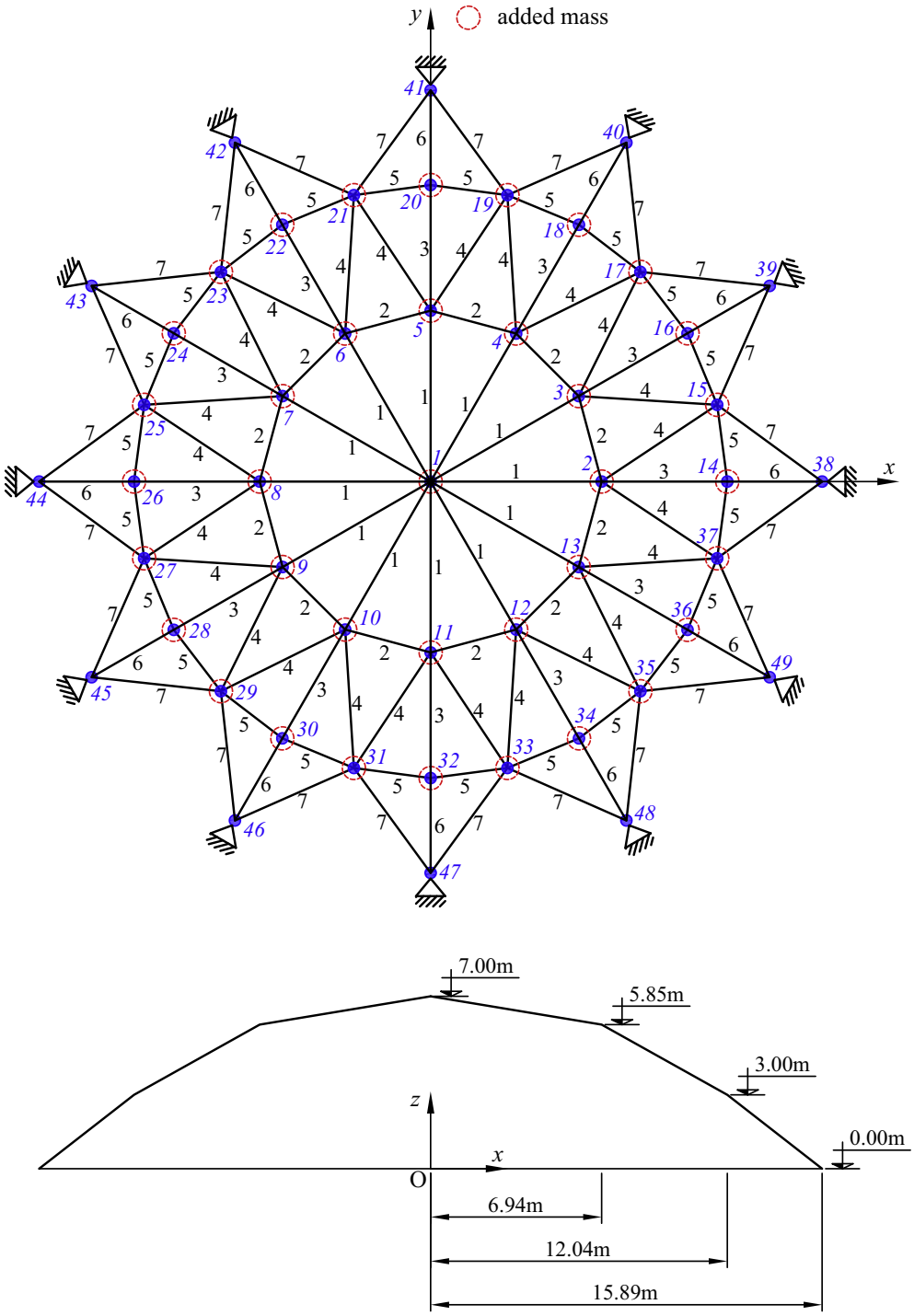

Fig. 6. The 120-bar dome truss.

4.1.3. 120-bar dome truss

Next, the 120-bar dome truss described in Fig. 6 is investigated. Node 1, nodes 2–13, and the rest of all free nodes are supplemented constant concentrated masses of $m_1 = 3000$, $m_2 = 500$ and $m_3 = 100$ kg, respectively. This is also a benchmark problem that has been previously examined by a variety of studies such as [9,10,14–18].

Table 6 presents a comparison of optimal results obtained by this work with those available in the literature. It is noticed that in this case, the AHEFA has the smallest optimal weight compared with the others. Optimal results of the first five natural frequencies obtained by the present study and different works are given in Table 7. Obviously, none of the frequency constraints is violated. Fig. 7 shows the weight convergence histories obtained using the DE, FA and AHEFA for this problem. As revealed in the figure, the convergence rate of the AHEFA is faster than the others.

4.1.4. 200-bar planar truss

A planar truss containing 200 bars shown in Fig. 8 is examined as the last example of the type of size optimization problem of truss structures with multiple frequency constraints. At upper nodes of the structure, non-structural masses of 100 kg are added

Table 6
Comparison of optimal results of the 120-bar dome truss obtained by different algorithms.

Design variable	CSS-BBBC	HALC-PSO	ReDE	This study		
A_i (cm^2)	[9]	[15]	[17]	DE	FA	AHEFA
1	17.478	19.8905	19.5131	19.4867	19.3030	19.5094
2	49.076	40.4045	40.3914	40.4260	41.0317	40.3867
3	12.365	11.2057	10.6066	10.6099	10.5580	10.6033
4	21.979	21.3768	21.1415	21.0910	21.0490	21.1168
5	11.190	9.8669	9.8057	9.8491	9.9086	9.8221
6	12.590	12.7200	11.7781	11.7639	11.8906	11.7735
7	13.585	15.2236	14.8163	14.8556	14.7781	14.8405
Best weight (kg)	9046.34	8889.96	8707.3221	8707.2715	8708.2350	8707.2559
Number of FE analyses	–	17000	5080	6000	13840	3560
Worst weight (kg)	–	–	8707.9138	8707.6493	8745.1771	8708.0154
Average weight (kg)	–	8900.39	8707.5233	8707.3937	8717.5865	8707.5580
Standard deviation	–	6.38	0.1543	0.1035	13.7261	0.2535

Table 7
The first five optimal natural frequencies of the 120-bar dome truss obtained by different algorithms.

Frequency number	CSS-BBBC	HALC-PSO	ReDE	This study		
	[9]	[15]	[17]	DE	FA	AHEFA
1	9.000	9.000	9.0000	9.0000	9.0000	9.0000
2	11.007	11.000	11.0000	11.0000	11.0000	11.0000
3	11.018	11.000	11.0000	11.0000	11.0000	11.0000
4	11.026	11.010	11.0000	11.0000	11.0000	11.0000
5	11.048	11.050	11.0667	11.0670	11.0672	11.0669

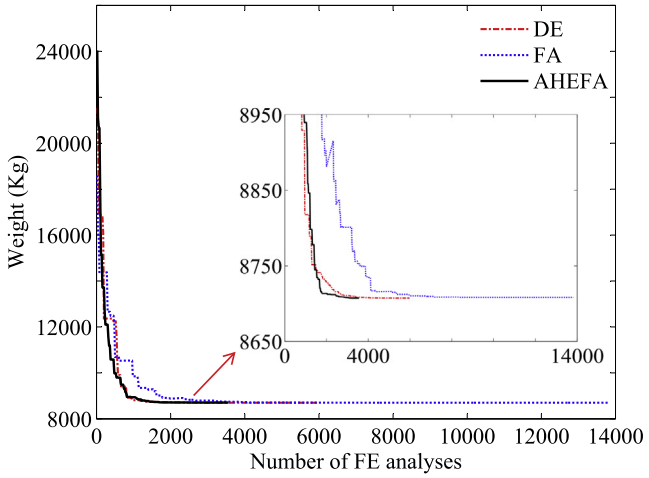

Fig. 7. The weight convergence histories of the 120-bar dome truss obtained using the DE, FA and AHEFA.

for the free vibration analysis. The structure is categorized into 29 member groups with regard to 29 design variables as indicated in Table 8. This example has been formerly considered in detail by numerous researchers with different methods [2,9,13–16,19,20].

As the previously illustrated examples, optimal results obtained by the present algorithms and different approaches are summarized in Table 8 for comparison. As expected, the best and worst weights, average weight and standard deviation given by the AHEFA are better than many methods. Furthermore, the number of FE analyses used to gain the optimal weight is the least as well. Note that although the best weight provided by the HALC-PSO is smaller than those by the AHEFA, the number of FE analyses for the convergence performance is larger than those of the proposed algorithm. Therefore, its computational cost is higher than the AHEFA. Optimal natural frequencies obtained by various

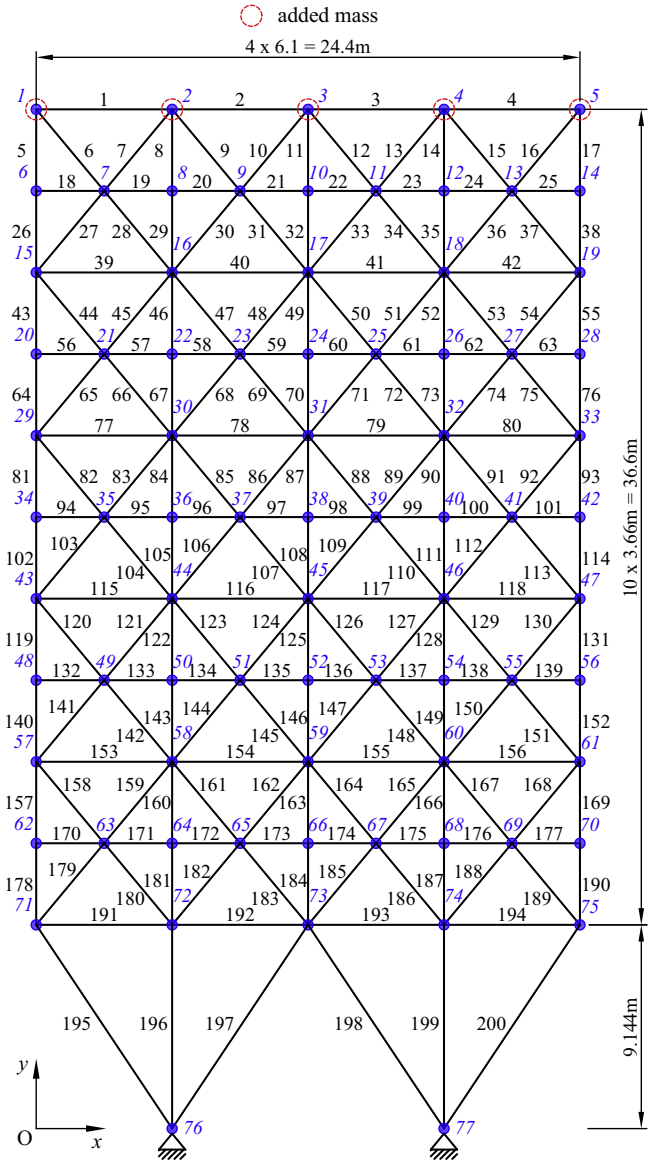

Fig. 8. The 200-bar planar truss.

algorithms are displayed in Table 9. It is recognized that all the constraints are not violated. The weight convergence histories obtained using the DE, FA and AHEFA for this structure are shown

Table 8
Comparison of optimal results of the 200-bar planar truss obtained by different algorithms.

Design variable A_i (cm^2)	Member group	CSS-BBBC [9]	HALC-PSO [15]	ALC-PSO [15]	This study		
					DE	FA	AHEFA
1	1, 2, 3, 4	0.2934	0.3072	0.2750	0.3035	0.6918	0.2993
2	5, 8, 11, 14, 17	0.5561	0.4545	0.4264	0.4528	0.5410	0.4508
3	19, 20, 21, 22, 23, 24	0.2952	0.1000	0.1000	0.1000	0.1000	0.1001
4	18, 25, 56, 63, 94, 101, 132, 139, 170, 177	0.1970	0.1000	0.1000	0.1000	0.1000	0.1000
5	26, 29, 32, 35, 38	0.8340	0.5080	0.7000	0.5162	0.5796	0.5123
6	6, 7, 9, 10, 12, 13, 15, 16, 27, 28, 30, 31, 33, 34, 36, 37	0.6455	0.8276	0.7948	0.8203	0.7950	0.8205
7	39, 40, 41, 42	0.1770	0.1023	0.1003	0.1004	0.1022	0.1011
8	43, 46, 49, 52, 55	1.4796	1.4357	1.5402	1.4393	1.3900	1.4156
9	57, 58, 59, 60, 61, 62	0.4497	0.1007	0.1000	0.1003	0.1000	0.1000
10	64, 67, 70, 73, 76	1.4556	1.5528	1.7544	1.5918	1.8472	1.5742
11	44, 45, 47, 48, 50, 51, 53, 54, 65, 66, 68, 69, 71, 72, 74, 75	1.2238	1.1529	1.1213	1.1641	1.1453	1.1597
12	77, 78, 79, 80	0.2739	0.1522	0.1000	0.1319	0.1009	0.1338
13	81, 84, 87, 90, 93	1.9174	2.9564	2.8381	2.9561	2.4956	2.9672
14	95,96, 97, 98, 99, 100	0.1170	0.1003	0.1000	0.1003	2.2732	0.1000
15	102, 105, 108, 111, 114	3.5535	3.2242	3.3936	3.2491	3.4189	3.2722
16	82, 83, 85, 86, 88, 89, 91, 92, 103, 104, 106, 107, 109, 110, 112, 113	1.3360	1.5839	1.5849	1.5949	1.6362	1.5762
17	115, 116, 117, 118	0.6289	0.2818	0.1000	0.2525	0.1000	0.2562
18	119, 122, 125, 128, 131	4.8335	5.0696	5.2642	5.1567	6.7800	5.0956
19	133, 134, 135, 136, 137, 138	0.6062	0.1033	0.1000	0.1004	6.2286	0.1001
20	140, 143, 146, 149, 152	5.4393	5.4657	5.7884	5.4938	6.0192	5.4546
21	120, 121, 123, 124, 126, 127, 129, 130, 141, 142, 144, 145, 147, 148, 150, 151	1.8435	2.0975	2.0218	2.1094	2.2978	2.0933
22	153, 154, 155, 156	0.8955	0.6598	0.4600	0.6731	0.3474	0.6737
23	157, 160, 163, 166, 169	8.1759	7.6585	7.8414	7.6922	7.8789	7.6498
24	171, 172, 173, 174, 175, 176	0.3209	0.1444	0.2983	0.1150	3.9626	0.1178
25	178, 181, 184, 187, 190	10.9800	8.0520	8.1844	8.0035	12.6765	8.0682
26	158, 159, 161, 162, 164, 165, 167, 168, 179, 180, 182, 183, 185, 186, 188, 189	2.9489	2.7889	2.7756	2.7794	3.2229	2.8025
27	191, 192, 193, 194	10.5243	10.4770	10.1639	10.5173	6.8088	10.5040
28	195, 197, 198, 200	20.4271	21.3257	21.4137	21.2292	18.4113	21.2935
29	196, 199	19.0983	10.5111	10.9083	10.7286	20.2728	10.7410
Best weight (kg)		2298.61	2156.73	2162.99	2160.7747	2441.3773	2160.7445
Number of FE analyses		–	13000	20000	19180	13140	11300
Worst weight (kg)		–	–	–	2171.2052	3261.9231	2161.3802
Average weight (kg)		–	2157.14	2562.07	2162.2495	2753.6381	2161.0393
Standard deviation		–	0.2413	328.55	3.0003	226.5117	0.1783

Table 9
The first six optimal natural frequencies of the 200-bar planar truss obtained by different algorithms.

Frequency number	CSS-BBBC [9]	HALC-PSO [15]	ALC-PSO [15]	This study		
				DE	FA	AHEFA
1	5.010	5.000	5.000	5.0000	5.0000	5.0000
2	12.911	12.254	12.136	12.2301	14.6752	12.1821
3	15.416	15.044	15.210	15.0277	15.0001	15.0160
4	17.033	16.718	16.723	16.7054	18.5869	16.6837
5	21.426	21.461	21.101	21.4238	21.0200	21.3547
6	21.613	21.524	21.526	21.4435	23.4380	21.4168

in Fig. 9. Clearly, the convergence rate of the AHEFA is always better than the others.

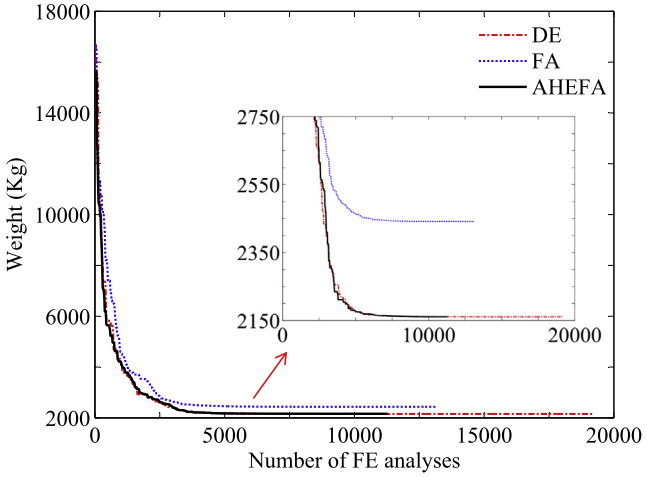

Fig. 9. The weight convergence histories of the 200-bar planar truss obtained using the DE, FA and AHEFA.

4.2. Shape and size optimization

4.2.1. 37-bar planar truss

A simply supported 37-bar planar truss is studied as the first shape and size optimization example to demonstrate the effectiveness and robustness of the AHEFA. Fig. 10 depicts the initial shape and a finite element representation of this structure. A constant lumped mass of 10 kg is placed at each of free nodes on the lower chord and is assumed to be unchanged during the design process. All bars of the lower chord are fixedly assigned to a constant cross-sectional area of 4×10^{-3} m^2, whereas the others possess an initial cross-sectional area of 1×10^{-4} m^2. Note that all the nodes on the upper chord are permitted to move vertically, their y-axis coordinates are thus taken as design variables. Furthermore, nodal coordinates and member areas are connected so that the structure is symmetric. Consequently, nineteen design variables containing

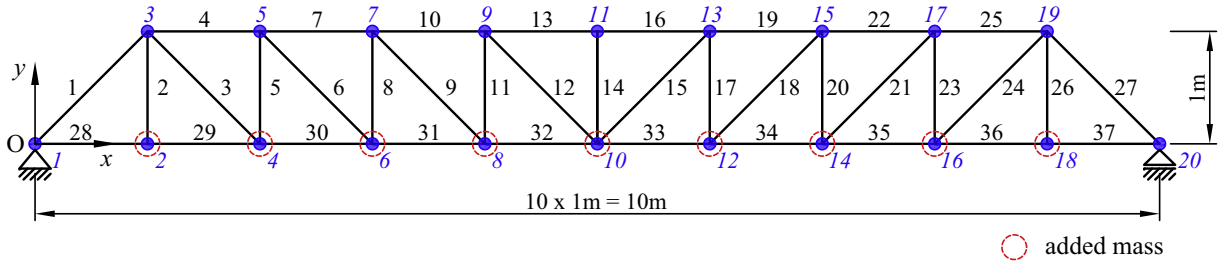

Fig. 10. The initial shape of the 37-bar planar truss.

Table 10
Comparison of optimal results of the 37-bar planar truss obtained by different algorithms.

Design variable	PSO	HS	OMGSA	HALC-PSO	ReDE	This study		
Y_j (m); A_i (cm^2)	[7]	[8]	[14]	[15]	[17]	DE	FA	AHEFA
Y_3, Y_{19}	0.9637	0.8415	1.0064	0.9750	0.9533	0.9940	0.8651	0.9589
Y_5, Y_{17}	1.3978	1.2409	1.4274	1.3577	1.3414	1.3688	1.2206	1.3450
Y_7, Y_{15}	1.5929	1.4464	1.6171	1.5520	1.5319	1.5528	1.4032	1.5355
Y_9, Y_{13}	1.8812	1.5334	1.7984	1.6920	1.6528	1.6974	1.5033	1.6668
Y_{11}	2.0856	1.5971	1.8720	1.7688	1.7280	1.7737	1.5741	1.7397
A_1, A_{27}	2.6797	3.2031	2.5017	2.9652	2.9608	2.7067	2.6225	2.8210
A_2, A_{26}	1.1568	1.1107	1.0949	1.0114	1.0052	1.0090	1.2146	1.0019
A_3, A_{24}	2.3476	1.1871	0.8891	1.0090	1.0014	1.0050	1.0010	1.0001
A_4, A_{25}	1.7182	3.3281	2.5172	2.4601	2.5994	2.5848	2.8617	2.5308
A_5, A_{23}	1.2751	1.4057	1.2119	1.2300	1.1949	1.1647	1.1598	1.2210
A_6, A_{21}	1.4819	1.0883	1.3147	1.2064	1.2165	1.2534	1.0933	1.2429
A_7, A_{22}	4.6850	2.1881	2.1197	2.4245	2.4303	2.4588	2.9790	2.4718
A_8, A_{20}	1.1246	1.2223	1.4223	1.4618	1.3644	1.4518	1.5042	1.4018
A_9, A_{18}	2.1214	1.7033	1.5948	1.4328	1.5548	1.4674	1.5779	1.5061
A_{10}, A_{19}	3.8600	3.1885	2.4784	2.5000	2.5247	2.3872	3.5802	2.5604
A_{11}, A_{17}	2.9817	1.0100	1.1896	1.2319	1.1946	1.2811	1.2247	1.2146
A_{12}, A_{15}	1.2021	1.4074	1.6461	1.3669	1.3163	1.3340	1.2327	1.3605
A_{13}, A_{16}	1.2563	2.8499	2.0878	2.2801	2.4465	2.4126	3.2194	2.3992
A_{14}	3.3276	1.0269	0.5008	1.0011	1.0003	1.0012	1.2760	1.0000
Best weight (kg)	377.20	361.50	359.97	359.93	359.8066	359.8957	361.5685	359.8121
Number of FE analyses	–	20000	242700	10000	13740	20000	14620	8640
Worst weight (kg)	–	–	363.84	–	360.5492	360.9547	391.2681	360.1072
Average weight (kg)	381.20	362.04	361.96	360.23	359.9944	360.0748	376.4367	359.9193
Standard deviation	4.26	0.52	1.868	0.24	0.1493	0.2982	10.6489	0.0871

Table 11
The first five optimal natural frequencies of the 37-bar planar truss obtained by different algorithms.

Frequency number	PSO	HS	OMGSA	HALC-PSO	ReDE	This study		
	[7]	[8]	[14]	[15]	[17]	DE	FA	AHEFA
1	20.0001	20.0037	20.022	20.0216	20.0005	20.0001	20.0000	20.0000
2	40.0003	40.0050	40.010	40.0098	40.0004	40.0002	40.0000	40.0001
3	60.0001	60.0082	60.049	60.0017	60.0022	60.0005	60.0000	60.0002
4	73.0440	77.9753	76.451	76.7857	76.4734	76.3599	76.4486	76.7801
5	89.8240	96.2564	96.297	96.3543	96.3820	96.1073	98.4713	96.4007

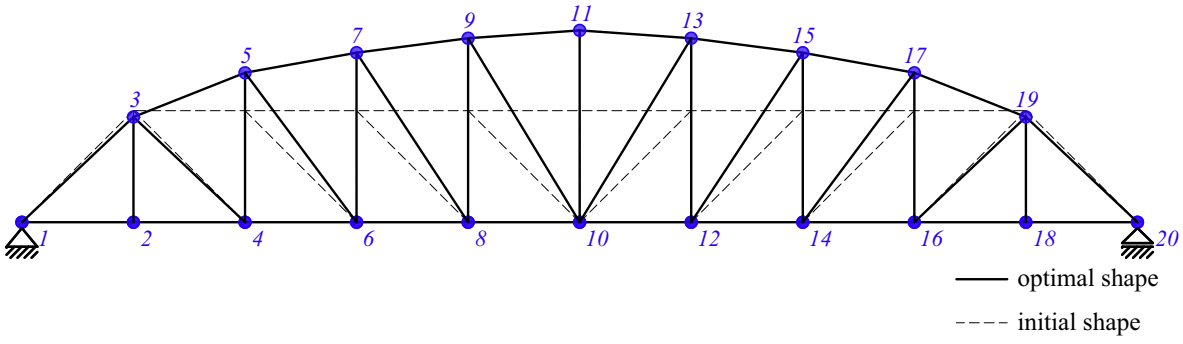

Fig. 11. The optimal shape of the 37-bar planar truss obtained using the AHEFA.

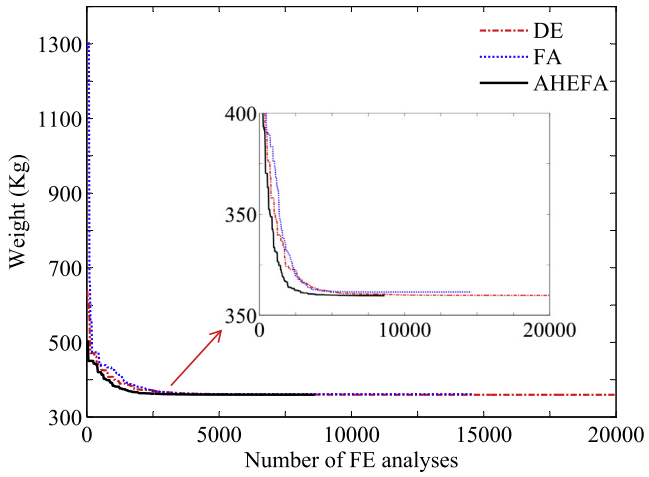

Fig. 12. The weight convergence histories of the 37-bar planar truss obtained using the DE, FA and AHEFA.

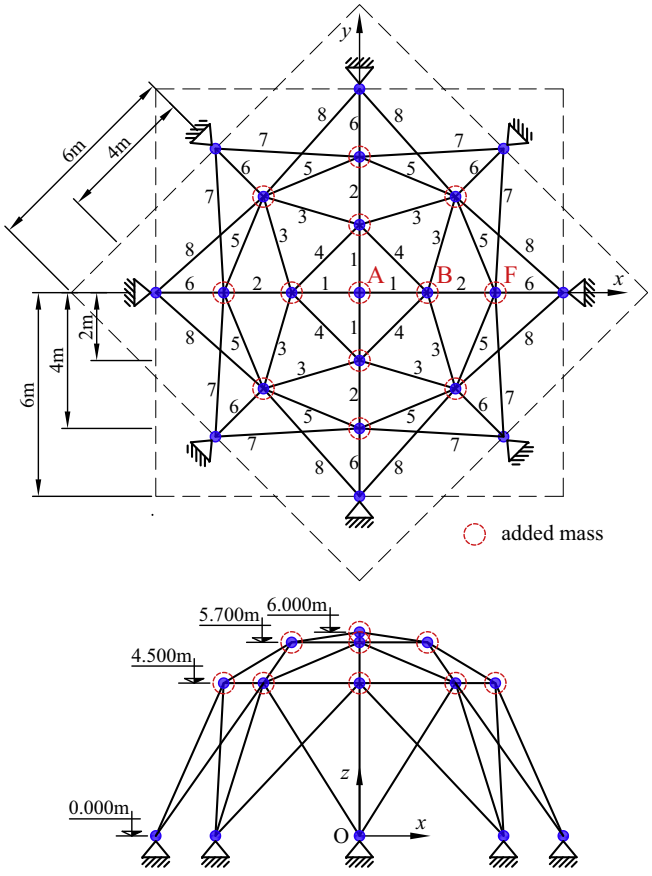

Fig. 13. The initial shape of the 52-bar dome truss.

five nodal coordinate and fourteen cross-sectional area ones are redesigned. This structure which is a well-known benchmark problem was first done by Wang et al. [3], and has been studied later by many authors [6–8,10–12,14–20].

Optimal results obtained by the present algorithms and different approaches including cross-sectional areas, nodal coordinates, corresponding weights and statistics results are provided in Table 10. It is seen that the AHEFA results in better solutions than the others. Note that the optimal weight obtained by the ReDE [17] is slightly lighter than those of the proposed method, yet this method requires more computational cost in terms of the number of FE analyses than the AHEFA. Table 11 represents the first five optimal natural frequencies attained by the present study and different ones. As observed, all the optimized frequencies are free from any violations of constraints. Fig. 11 describes the optimal shape of the truss structure obtained using the AHEFA. It can be stated that this final shape is quite similar to those performed by the previously published works. It is fairly sensible and can be used for practical designs with the high feasibility. Finally, as expected, the convergence rate obtained by the AHEFA is always much faster than the DE and FA as expressed in Fig. 12.

4.2.2. 52-bar dome truss

A last example for the shape and size optimization problem done herein is to optimize a 52-bar dome truss. The initial model of the structure is depicted in Fig. 13. All members of the structure are divided into eight variable groups for optimization design as labeled in the same figure. Each free node is added a concentrated mass of 50 kg. The three coordinates (x, y, z) of each of free nodes are allowed to shift within the range $[-2, +2]$m, and they are considered as design variables as well. Note that the whole structure must preserve its symmetry during the design process. Thus, there are 13 independent design variables including 5 shape and 8 sizing variables. All member areas are initially assigned a value of 2×10^{-4} m^2. A first study on this problem was carried out by Lin et al. [66]. Other investigations have been subsequently conducted by many researchers such as [3,6–12,14–20].

Table 12 shows a comparison of optimal results obtained by the present algorithms and different approaches. Without any doubts regarding the effectiveness and robustness of the AHEFA, the optimal weight achieved by the proposed method is the best compared with those given by the others as indicated in the table. Table 13 displays the first five optimal natural frequencies for different optimization methods. It is easily seen that no any violations of constraints are observed in all the optimized results. Fig. 14 sketches the final optimal shape of the structure given by the AHEFA after the design process. It should be noted that analogous configurations were also found out by various studies in the literature. Fig. 15 shows the weight convergence histories obtained using the DE, FA and AHEFA for this structure. Again, the AHEFA shows its effectiveness and robustness in significantly improving the convergence speed compared with the DE and the FA as well as many other algorithms but still achieving good solution accuracy.

5. Conclusions

In this paper, a novel AHEFA based on a hybridization of the DE algorithm and the FA is proposed for dealing with shape and size optimization problems of truss structures under multiple frequency constraints. An automatically adapted parameter computed from the deviation of objective function between the best individual and the whole population in the previous generation is used to select an appropriate mutation operator for the performance in the mutation phase. This leads the global exploration and local exploitation capabilities to be balanced effectively. Additionally, an elitist technique is employed for the selection phase to choose a new population for the next generation consisting of the best individuals from the mixture of target and trial individuals. Feasible solutions in the target individual can thus be kept for the next generation instead of the trial ones only. Six numerical benchmark examples with highly nonlinear and non-convex properties including four ones for size optimization and two ones for shape and size optimization are tested to verify the effectiveness and robustness of the proposed approach. Attained results indicated that the convergence speed of the proposed method is dra-

Table 12
Comparison of optimal results of the 52-bar dome truss obtained by different algorithms.

Design variable	PSO	HS	CSS-BBBC	HALC-PSO	ReDE	This study		
Z_j, X_j (m); A_i (cm^2)	[7]	[8]	[9]	[15]	[17]	DE	FA	AHEFA
Z_A	5.5344	4.7374	5.3310	5.9362	6.0188	6.0180	6.1715	5.9953
X_B	2.0885	1.5643	2.1340	2.2416	2.2976	2.2736	2.2657	2.3062
Z_B	3.9283	3.7413	3.7190	3.7309	3.7417	3.7609	3.8937	3.7308
X_F	4.0255	3.4882	3.9350	3.9630	3.9996	4.0000	3.9826	4.0000
Z_F	2.4575	2.6274	2.5000	2.5000	2.5001	2.5001	2.6075	2.5000
A_1	0.3696	1.0085	1.0000	1.0001	1.0000	1.0000	1.0004	1.0000
A_2	4.1912	1.4999	1.3056	1.1654	1.0852	1.0980	1.0606	1.0832
A_3	1.5123	1.3948	1.4230	1.2323	1.1968	1.1882	1.1792	1.2014
A_4	1.5620	1.3462	1.3851	1.4323	1.4503	1.4098	1.3119	1.4527
A_5	1.9154	1.6776	1.4226	1.3901	1.4216	1.3922	1.4629	1.4212
A_6	1.1315	1.3704	1.0000	1.0001	1.0001	1.0000	1.0008	1.0000
A_7	1.8233	1.4137	1.5562	1.6024	1.5614	1.5684	1.4730	1.5570
A_8	1.0904	1.9378	1.4485	1.4131	1.3878	1.4182	1.5728	1.3904
Best weight (kg)	228.381	214.94	197.309	194.85	193.2048	193.2462	196.5993	193.2024
Number of FE analyses	–	20000	–	7500	16200	20000	18630	12120
Worst weight (kg)	–	–	–	–	202.3656	202.3394	342.6344	202.3941
Average weight (kg)	234.30	229.88	–	196.85	195.4260	195.1199	231.4418	198.7290
Standard deviation	5.22	12.44	–	2.38	3.8587	3.5851	55.8303	4.4108

Table 13
The first five optimal natural frequencies of the 52-bar dome truss obtained by different algorithms.

Frequency number	PSO	HS	CSS-BBBC	HALC-PSO	ReDE	This study		
	[7]	[8]	[9]	[15]	[17]	DE	FA	AHEFA
1	12.751	12.2222	12.987	11.4339	11.6107	11.5493	11.7391	11.6629
2	28.649	28.6577	28.648	28.6480	28.6482	28.6480	28.6485	28.6480
3	28.649	28.6577	28.679	28.6480	28.6499	28.6481	28.6490	28.6481
4	28.803	28.6618	28.713	28.6482	28.6500	28.6481	28.6845	28.6482
5	29.230	30.0997	30.262	28.6848	28.6510	28.6485	28.7505	28.6611

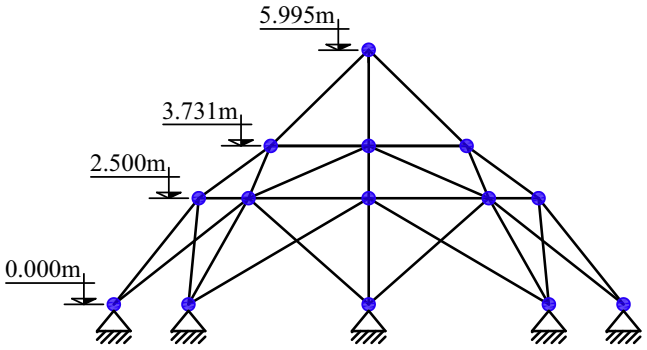

Fig. 14. The optimal shape of the 52-bar dome truss obtained using the AHEFA.

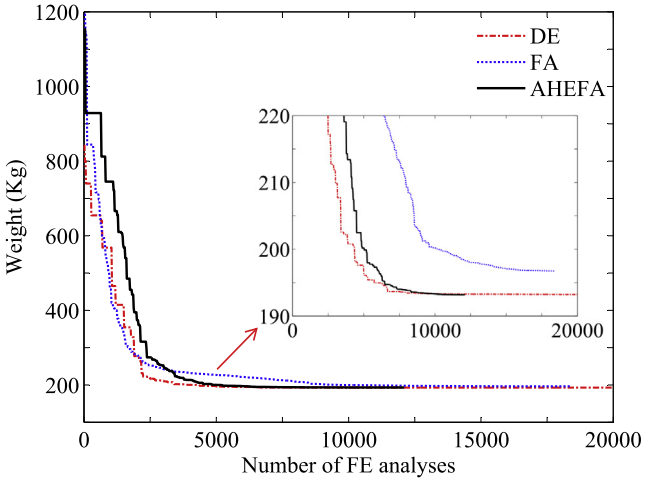

Fig. 15. The weight convergence histories of the 52-bar dome truss obtained using the DE, FA and AHEFA.

matically improved in comparison with the DE and the FA as well as numerous approaches in the literature, while the solution accuracy is still better than many others. The implementation of the present method is fairly simple as no substantial changes in code structures are observed compared with the DE and the FA. Therefore, the AHEFA is very promising to extend its applications to various engineering optimization problems in many fields with the high feasibility whose computational cost is a matter of great interest.

Acknowledgment

This research was supported by a grant (NRF-2017R1A4A1015660) from NRF (National Research Foundation of Korea) funded by MEST (Ministry of Education and Science Technology) of Korean government.

References

[1] Bellagamba L, Yang TY. Minimum-mass truss structures with constraints on fundamental natural frequency. AIAA J 1981;19:1452–8.
[2] Grandhi RV, Venkayya VB. Structural optimization with frequency constraints. AIAA J 1988;26:858–66.
[3] Wang D, Zhang WH, Jiang JS. Truss optimization on shape and sizing with frequency constraints. AIAA J 2004;42:622–30.
[4] Sedaghati R, Suleman A, Tabarrok B. Structural optimization with frequency constraints using the finite element force method. AIAA J 2002;40:382–8.
[5] Sedaghati R. Benchmark case studies in structural design optimization using the force method. Int J Solids Struct 2005;42:5848–71.
[6] Wei L, Tang T, Xie X, Shen W. Truss optimization on shape and sizing with frequency constraints based on parallel genetic algorithm. Struct Multidiscip Optim 2011;43:665–82.
[7] Gomes HM. Truss optimization with dynamic constraints using a particle swarm algorithm. Expert Syst Appl 2011;38:957–68.
[8] Miguel LFF, Fadel Miguel LF. Shape and size optimization of truss structures considering dynamic constraints through modern metaheuristic algorithms. Expert Syst Appl 2012;39:9458–67.

[9] Kaveh A, Zolghadr A. Truss optimization with natural frequency constraints using a hybridized CSS-BBBC algorithm with trap recognition capability. Comput Struct 2012;102–103:14–27.
[10] Kaveh A, Zolghadr A. Democratic PSO for truss layout and size optimization with frequency constraints. Comput Struct 2014;130:10–21.
[11] Kaveh A, Zolghadr A. Cyclical parthenogenesis algorithm for layout optimization of truss structures with frequency constraints. Eng Optim 2016;0:1–18.
[12] Kaveh A, Zolghadr A. Truss shape and size optimization with frequency constraints using tug of war optimization. Asian J Civil Eng 2017;18:311–33.
[13] Zuo W, Bai J, Li B. A hybrid OC-GA approach for fast and global truss optimization with frequency constraints. Appl Soft Compt 2014;14:528–35.
[14] Khatibinia M, Naseralavi SS. Truss optimization on shape and sizing with frequency constraints based on orthogonal multi-gravitational search algorithm. J Sound Vib 2014;333:6349–69.
[15] Kaveh A, Ilchi Ghazaan M. Hybridized optimization algorithms for design of trusses with multiple natural frequency constraints. Adv Eng Softw 2015;79:137–47.
[16] Tejani GG, Savsani VJ, Patel VK. Adaptive symbiotic organisms search (SOS) algorithm for structural design optimization. J Comput Des Eng 2016;3:226–49.
[17] Ho-Huu V, Nguyen-Thoi T, Truong-Khac T, Le-Anh L, Vo-Duy T. An improved differential evolution based on roulette wheel selection for shape and size optimization of truss structures with frequency constraints. Neural Comput Appl 2016:1–19.
[18] Ho-Huu V, Vo-Duy T, Luu-Van T, Le-Anh L, Nguyen-Thoi T. Optimal design of truss structures with frequency constraints using improved differential evolution algorithm based on an adaptive mutation scheme. Autom Constr 2016;68:81–94.
[19] Farshchin M, Camp CV, Maniat M. Multi-class teaching-learning-based optimization for truss design with frequency constraints. Eng Struct 2016;106:355–69.
[20] Farshchin M, Camp CV, Maniat M. Optimal design of truss structures for size and shape with frequency constraints using a collaborative optimization strategy. Expert Syst Appl 2016;66:203–18.
[21] Storn R, Price K. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. J Glob Optim 1997;11:341–59.
[22] Wu CY, Tseng KY. Truss structure optimization using adaptive multi-population differential evolution. Struct Multidiscip Optim 2010;42:575–90.
[23] Zou D, Liu H, Gao L, Li S. A novel modified differential evolution algorithm for constrained optimization problems. Comput Math Appl 2011;61:1608–23.
[24] Zou D, Wu J, Gao L, Li S. A modified differential evolution algorithm for unconstrained optimization problems. Neurocomputing 2013;120:469–81.
[25] Jiang LL, Maskell DL, Patra JC. Parameter estimation of solar cells and modules using an improved adaptive differential evolution algorithm. Appl Energy 2013;112:185–93.
[26] Panda S, Kiran SH, Dash SS, Subramani C. A PD-type multi input single output SSSC damping controller design employing hybrid improved differential evolution-pattern search approach. Appl Soft Comput 2015;32:532–43.
[27] Panda S, Yegireddy NK. Multi-input single output SSSC based damping controller design by a hybrid improved differential evolution-pattern search approach. ISA Trans 2015;58:173–85.
[28] Do DTT, Lee S, Lee J. A modified differential evolution algorithm for tensegrity structures. Compos Struct 2016;158:11–9.
[29] Elsayed SM, Sarker RA, Essam DL. GA with a new multi-parent crossover for solving IEEE-CEC2011 competition problems. In: 2011 IEEE congress of evolutionary computation (CEC).
[30] Trivedi A, Srinivasan D, Biswas S, Reindl T. Hybridizing genetic algorithm with differential evolution for solving the unit commitment scheduling problem. Swar Evol Comput 2015;23:50–64.
[31] Liao TW. Two hybrid differential evolution algorithms for engineering design optimization. Appl Soft Comput 2010;10:1188–99.
[32] Boussaid I, Chatterjee A, Siarry P, Ahmed-Nacer M. Hybridizing biogeography-based optimization with differential evolution for optimal power allocation in wireless sensor networks. IEEE Trans Veh Tech 2011;60:2347–53.
[33] Bureerat S. Improved population-based incremental learning in continuous spaces. In: Soft computing in industrial applications. Springer; 2011.
[34] Chang L, Liao C, Lin W, Chen LL, Zheng X. A hybrid method based on differential evolution and continuous ant colony optimization and its application on wideband antenna design. Prog Elect Res 2012;122:105–18.
[35] Biswal B, Behera HS, Bisoi R, Dash PK. Classification of power quality data using decision tree and chemotactic differential evolution based fuzzy clustering. Swar Evol Comput 2012;4:12–24.
[36] Epitropakis MG, Plagianakos VP, Vrahatis MN. Evolving cognitive and social experience in particle swarm optimization through differential evolution: a hybrid approach. Inform Sci 2012;216:50–92.
[37] Basak A, Maity D, Das S. A differential invasive weed optimization algorithm for improved global numerical optimization. Appl Math Comput 2013;219:6645–68.
[38] Olenšek J, Tuma T, Puhan J, Bürmen R. A new asynchronous parallel global optimization method based on simulated annealing and differential evolution. Appl Soft Comput 2011;11:1481–9.
[39] Guo H, Li Y, Li J, Sun H, Wang D, Chen X. Differential evolution improved with self-adaptive control parameters based on simulated annealing. Swar Evol Comput 2014;19:52–67.
[40] Ghosh S, Das S, Roy S, Minhazul Islam SK, Suganthan PN. A differential covariance matrix adaptation evolutionary algorithm for real parameter optimization. Inform Sci 2012;182:199–219.
[41] Li YL, Zhan ZH, Gong YJ, Chen WN, Zhang J, Li Y. Differential evolution with an evolution path: a DEEP evolutionary algorithm. IEEE Trans Cybernet 2015;45:1798–810.
[42] Zheng YJ, Xu XL, Ling HF, Chen SY. A hybrid fireworks optimization method with differential evolution operators. Neurocomputing 2015;148:75–82.
[43] Chakraborti T, Chatterjee A, Halder A, Konar A. Automated emotion recognition employing a novel modified binary quantum-behaved gravitational search algorithm with differential mutation. Expert Syst 2015;32:522–30.
[44] Tran DH, Cheng MY, Cao MT. Hybrid multiple objective artificial bee colony with differential evolution for the time-cost-quality tradeoff problem. Knowl-Based Syst 2015;74:176–86.
[45] Nelder JA, Mead R. A simplex method for function minimization. Comput J 1965;7:308–13.
[46] Reynoso-Meza G, Sanchis J, Blasco X, Herrero JM. Hybrid DE algorithm with adaptive crossover operator for solving real-world numerical optimization problems. In: 2011 IEEE congress of evolutionary computation (CEC).
[47] Jia D, Zheng G, Khurram Khan M. An effective memetic differential evolution algorithm based on chaotic local search. Inform Sci 2011;181:3175–87.
[48] Neri F, Iacca G, Mininno E. Disturbed exploitation compact differential evolution for limited memory optimization problems. Inform Sci 2011;181:2469–87.
[49] Das S, Mullick SS, Suganthan PN. Recent advances in differential evolution – an updated survey. Swar Evol Comput 2016;27:1–30.
[50] Yang XS. Firefly algorithm. Lévy flights and global optimization. In: Research and development in intelligent systems XXVI. Springer; 2010.
[51] Yang XS. Firefly algorithm, stochastic test functions and design optimisation. Int J Bio-Insp Comput 2010;2:78–84.
[52] Yang XS. Nature-inspired optimization algorithms. The Netherlands: Elsevier Science Publishers B.V.; 2014.
[53] Fister I, Fister Jr I, Yang XS, Brest J. A comprehensive review of firefly algorithms. Swar Evol Comput 2013;13:34–46.
[54] Yang XS, Deb S. Eagle strategy using Lévy walk and firefly algorithms for stochastic optimization. In: Nature inspired cooperative strategies for optimization (NICSO 2010). Stud comp intell. Springer; 2010.
[55] Luthra J, Pal SK. A hybrid firefly algorithm using genetic operators for the cryptanalysis of a monoalphabetic substitution cipher. In: 2011 World congress on information and communication technologies.
[56] Farahani SM, Abshouri AA, Nasiri B, Meybodi MR. Some hybrid models to improve firefly algorithm performance. Int J Artificial Intell 2012;8:97–117.
[57] Aruchamy R, DrVasantha KD. A comparative performance study on hybrid swarm model for micro array data. Int J Comput Appl 2011;30:10–4.
[58] Abdullah A, Deris S, Mohamad MS, Hashim SZM. A new hybrid firefly algorithm for complex and nonlinear problem. In: Distributed computing and artificial intelligence. Springer; 2012.
[59] Abdullah A, Deris S, Anwar S, Arjunan SNV. An evolutionary firefly algorithm for the estimation of nonlinear biological model parameters. PloS One 2013;8:1–16.
[60] Kaveh A, Talatahari S. A novel heuristic optimization method: charged system search. Acta Mech 2010;213:267–89.
[61] Zhao SZ, Suganthan PN, Das S. Self-adaptive differential evolution with multi-trajectory search for large-scale optimization. Soft Comput 2011;15:2175–85.
[62] Tanabe R, Fukunaga A. Reevaluating exponential crossover in differential evolution. In: Parallel problem solving from nature – PPSN XIII. Springer; 2014.
[63] Qin AK, Huang VL, Suganthan PN. Differential evolution algorithm with strategy adaptation for global numerical optimization. IEEE Trans Evol Comput 2009;13:398–417.
[64] Padhye N, Bhardawaj P, Deb K. Improving differential evolution through a unified approach. J Glob Optim 2013;55:771–99.
[65] Reddy J. An introduction to the finite element method. New York: McGraw-Hill; 2005.
[66] Lin JH, Che WY, Yu YS. Structural optimization on geometrical configuration and element sizing with statical and dynamical constraints. Comput Struct 1982;15:507–15.