Engineering with Computers (2021) 37:763–777
https://doi.org/10.1007/s00366-019-00854-6

ORIGINAL ARTICLE

Exporting water wave optimization concepts to modified simulated annealing algorithm for size optimization of truss structures with natural frequency constraints

Carlos Millan-Paramo[1,2] · **João Elias Abdalla Filho**[1]

Received: 6 June 2019 / Accepted: 23 August 2019 / Published online: 31 August 2019
© Springer-Verlag London Ltd., part of Springer Nature 2019

Abstract
This paper proposes an improved version of a recently proposed modified simulated annealing algorithm (MSAA) named as an improved MSAA (I-MSAA) to tackle the size optimization of truss structures with frequency constraint. This kind of problem is problematic because its feasible region is non-convex while the boundaries are highly non-linear. The main motivation is to improve the exploitative behavior of MSAA, taking concept from water wave optimization metaheuristic (WWO). An interesting concept of WWO is its *breaking* operation. Thirty functions extracted from the CEC2014 test suite and four benchmark truss optimization problems with frequency constraints are explored for the validity of the proposed algorithm. Numerical results indicate that I-MSAA is more reliable, stable and efficient than those found by other existing metaheuristics in the literature.

Keywords Truss structures · Size optimization · Frequency constraints · Modified simulated annealing algorithm (MSAA) · Water wave optimization (WWO) · Metaheuristics

1 Introduction

Most engineering structures are subjected to dynamic loads (e.g., fatigue, seismic, and shock loads) which can produce unwanted vibrations. The natural frequencies are essential parameters that are useful to avoid resonance and improve the dynamic behavior of a structure [1, 2]. In addition, engineering structures should be as light as possible. However, minimizing the weight of structures can be considered as a difficult problem to solve because the reduction of weight generates conflict with the frequency bounds. These constraints are highly non-linear, non-convex, and implied with respect to the variables of design [3]. Therefore, this has led to difficulty in the use of gradient-based optimizers [2]. Under such circumstances, the metaheuristic algorithms can serve as appropriate alternatives due to the ability to search global minima in highly modal and multidimensional spaces.

The first works to address this problem employed classical techniques such as Gauss method [4], bi-factor algorithm [5], optimality criteria [1, 6], and integrated force method [7, 8]. Since then many researchers in the field of structural optimization have introduced several metaheuristics for solving this kind of optimization problem. Table 1 presents the most important works that involve metaheuristics methods to solve this problem. Although several metaheuristics have been introduced to solve this problem, most of them are population based, undergo many steps along with several parameters that make them difficult to understand and code. Also, there are same procedures in recent metaheuristics which make them similar. Because of this, the researchers usually are confused to select a metaheuristic. According to the No Free Lunch Theorem in the field of optimization, there is no algorithm to solve all optimization problems. This indicates that a new improved algorithm has potential to solve a group of problems better than the existing algorithms.

Contrary to the previous works, this paper aims to improve the performance of the recently proposed modified

✉ Carlos Millan-Paramo
carlos.millan@unisucre.edu.co

João Elias Abdalla Filho
joaofilho@utfpr.edu.br

[1] Postgraduate Program in Civil Engineering (PPGEC), Universidade Tecnológica Federal do Paraná, St. Dep. Heitor Alencar Furtado, 5000, Curitiba, Paraná CEP 81280-340, Brazil

[2] Department of Civil Engineering, Universidad de Sucre, Sincelejo, Colombia

Table 1 Main works in optimization of truss structures with frequency constraints

References	Metaheuristic
Lingyun et al. [9, 10]	Niche hybrid parallel genetic algorithm (NHPGA) Parallel genetic algorithm
Gomes [11]	Particle swarm optimization (PSO)
Miguel and Fadel Miguel [12]	Harmony search (HS) Firefly algorithm (FA)
Kaveh and Zolghadr [13]	Hybridization of the charged system search and the big bang-big crunch algorithms (CSS-BBC)
Kaveh and Zolghadr [14]	Charged system search (CSS)
Kaveh and Zolghadr [15]	Democratic particle swarm optimization (DPSO)
Kaveh and Mahdavi [16]	Colliding bodies optimization (CBO)
Khatibinia and Naseralevi [17]	Orthogonal multi-gravitational search algorithm (OMGSA)
Kaveh and Ilchi Ghazaan [18]	Hybridization of the particle swarm optimization with an aging leader and challengers (ALC-PSO and HALC-PSO)
Farshchin and Camp [19]	School-based optimization (SBO)
Goncalves et al. [20]	Search group algorithm (SGA)
Farshchin and Camp [21]	Multi-class teaching–learning-based optimization (MC-TLBO)
Kaveh and Zolghadr [22]	Cyclical parthenogenesis algorithm (CPA)
Kaveh and Ilchi Ghazaan [23]	Vibrating particles system (VPS)
Kaveh and Zolghadr [24]	Tug of war optimization (TWO)
Ho-Huu et al. [25]	Differential evolution (ReDE)
Tejani et al. [26, 27]	Symbiotic organisms search (SOS)
Lieu et al. [28]	Adaptive hybrid evolutionary firefly algorithm (AHEFA)

simulated annealing algorithm (MSAA) and adapt it better for structure design problems. MSAA is a simple single-solution algorithm based on the behavior of atomic arrangements in liquid or solid materials during the annealing process introduced by Millat et al. [29] for solving global optimization problems and it has been applied in structural optimization problems with success [30–32]. Regardless of the successful application of MSAA, this algorithm estimates the global optimum of a given problem in three phases: preliminary exploration, search step, and probability of accepting. First, a preliminary exploration is realized to choose the starting point of search. Second, the transition from the start point to the new point is done by a search step. Third, the range of probability of accepting a worse solution is reduced [29].

In the search step phase, the new solution for comparison is not randomly generated. From the starting point determined in the preliminary exploration phase, a search step is generated to determine the neighboring state. This step depends on a radius (R) of action that gradually decreases as the temperature of the system decreases. The transition from starting point to the new point (search step) is performed by the addition of random numbers that are between $[-R, R]$. This phase leads solution to jump into non-visited regions (exploration) and permits local search of visited regions (exploitation). However, the exploitation capability of this phase is considerably low as compared to the exploratory capability. This causes the algorithm to consume a large number of unused function evaluations (FEs) at low temperatures. To overcome this drawback, the aim of this work is to investigate whether the basic concepts underlying water wave optimization (WWO) can be exported to improve the MSAA. This variation pretends to allow a good balance between exploration and exploitation throughout the optimization process. The validity of the improved MSAA (based on WWO) is confirmed by testing for a diverse set of benchmark problems and applied to size optimization problems of truss structures with frequency constraints. Optimal results attained by I-MSAA are compared with other metaheuristics in the literature.

The remainder of this article is structured as follows. The MSAA is briefly presented in Sect. 2. Section 3 describes the improvement in the MSAA. In Sect. 4, the 30 benchmark functions proposed in the CEC2014 special session on single objective real-parameter numerical optimization [33] are used to demonstrate the effectiveness of the proposed algorithm. Section 5 describes the mathematical formulation of truss optimization with frequency constraints. Section 6 presents four most widely investigated benchmark numerical examples to illustrate the efficiency of the I-MSAA. Finally, in Sect. 7, our final conclusions are presented.

2 Modified simulated annealing algorithm

The MSAA [29] is a single-solution algorithm based on the cooling process of molten metals through annealing process. MSAA has three main stages that differentiate it from the simulated annealing (SA) proposed by Kirkpatrick [34]:

1. The starting point is not randomly generated but is selected by a preliminary exploration where the algorithm performs a scan in the search space and is given by the following equation:

$$\mathbf{X}_{P\times N} = \mathbf{I}_{P\times N} X_{\mathrm{L}} + \mathbf{rand}_{P\times N}(X_{\mathrm{U}} - X_{\mathrm{L}}), \tag{1}$$

where P is the number of points that are desired in the search space; N the number of dimensions of the problem; $\mathbf{I}_{P\times N}$ the identity matrix of size $P\times N$; X_{L} the lower limit of the problem; X_{U} the upper limit of the problem, and $\mathbf{rand}_{P\times N}$ is the matrix of random numbers (pure randomness) between 0 and 1 of size $P\times N$. To start the optimization process with MSAA, all points generated with (1) are evaluated in the objective function of the problem and the smallest value (in the case of searching the minimum value of the function) is chosen as the starting point of the search.

2. The new solution for comparison is not randomly generated. From the starting point determined in the preliminary exploration stage, a search step is generated to determine the neighboring state. This step depends on a radius (R) of action that gradually decreases as the temperature of the system decreases. The transition from the starting point to the new point is performed by the addition of random numbers that are within the defined radius. This enables the algorithm to execute a global exploration at high temperatures and a local exploration at low temperatures. The radius is updated as follows:

$$R_{\mathrm{i}+1} = R_{\mathrm{i}} \times \alpha, \tag{2}$$

where R_{i} is the initial radius cycle and α is the radius reduction coefficient.

3. If the cost function of the new solution is higher than the best value, the acceptance of the new solution depends on the following equation:

$$P = \frac{1}{1 + e^{(\Delta f/T)}}, \tag{3}$$

where P is the probability of accepting the new solution; Δf the difference between the quality of the new solution and the quality of the current solution; T the temperature of the system; and e is the Euler number. This probability is in a range between 0 and 0.5, allowing the algorithm to have a lower range of acceptance of worse solutions. For more details, see [29]. The flowchart of the MSAA is illustrated in Fig. 1.

3 Improvement in the MSAA

The capability to balance intensification and diversification during a search determines the efficiency of a specific metaheuristic algorithm. Diversification (exploration) ensures, usually by randomization, that the algorithm explores the search space efficiently. Intensification (exploitation) aims to identify the best solution and select, during the process, a succession of best solutions. In the search step phase of the MSAA, the trial point is generated by the addition of random numbers that are defined within a radius (Fig. 1). This phase works mainly to improve exploitation capabilities of the search process. Although the MSAA has demonstrated its ability to find near global regions within a reasonable time, it is comparatively inefficient in performing local searches [31]. This is because the radius must be tuned for each problem, affecting the speed convergence of the algorithm. Thus, this paper proposes the I-MSAA as a new and superior MSAA algorithm variant to improve MSAA algorithm local search capabilities and balance-associated intensification and diversification components. The proposed algorithm introduces a concept drawn from water wave optimization (WWO) [35] to replace the search step phase.

WWO was introduced by Zheng [35] and is inspired by shallow water wave models. The WWO has three important phases for finding optimal solutions. They are: propagation, breaking, and refraction phase. In propagation phase, the wave is propagated to a random position exactly once in an iteration. If a wave attains a lower sea depth (for minimization), it breaks into solitary waves which are formed in the breaking phase. Thus, breaking is used for the intensive search (exploitation) in search spaces by producing random solitary waves around the current best position. While in the refraction phase, the algorithm explores the search space for any other best solution and avoids search inactiveness. As our interest is to improve the exploitation of the MSAA, the breaking phase used in WWO is implemented in the I-MSAA.

3.1 Breaking

According to Zheng [35], when a wave moves to a position where the water depth is below a threshold value, the wave crest velocity exceeds the wave celerity. Consequently, the crest becomes steeper and steeper, and finally the wave breaks into a train of solitary waves. In WWO, the breaking operation performs a local search around the wave to simulate wave breaking. In the problem-solving process, k is randomly selected (where k is a random number between 1 and a predefined number $k_{\max}$), and at each dimension d generate a solitary wave x as:

$$x'(d) = x(d) + N(0,1) \times \beta L(d), \tag{4}$$

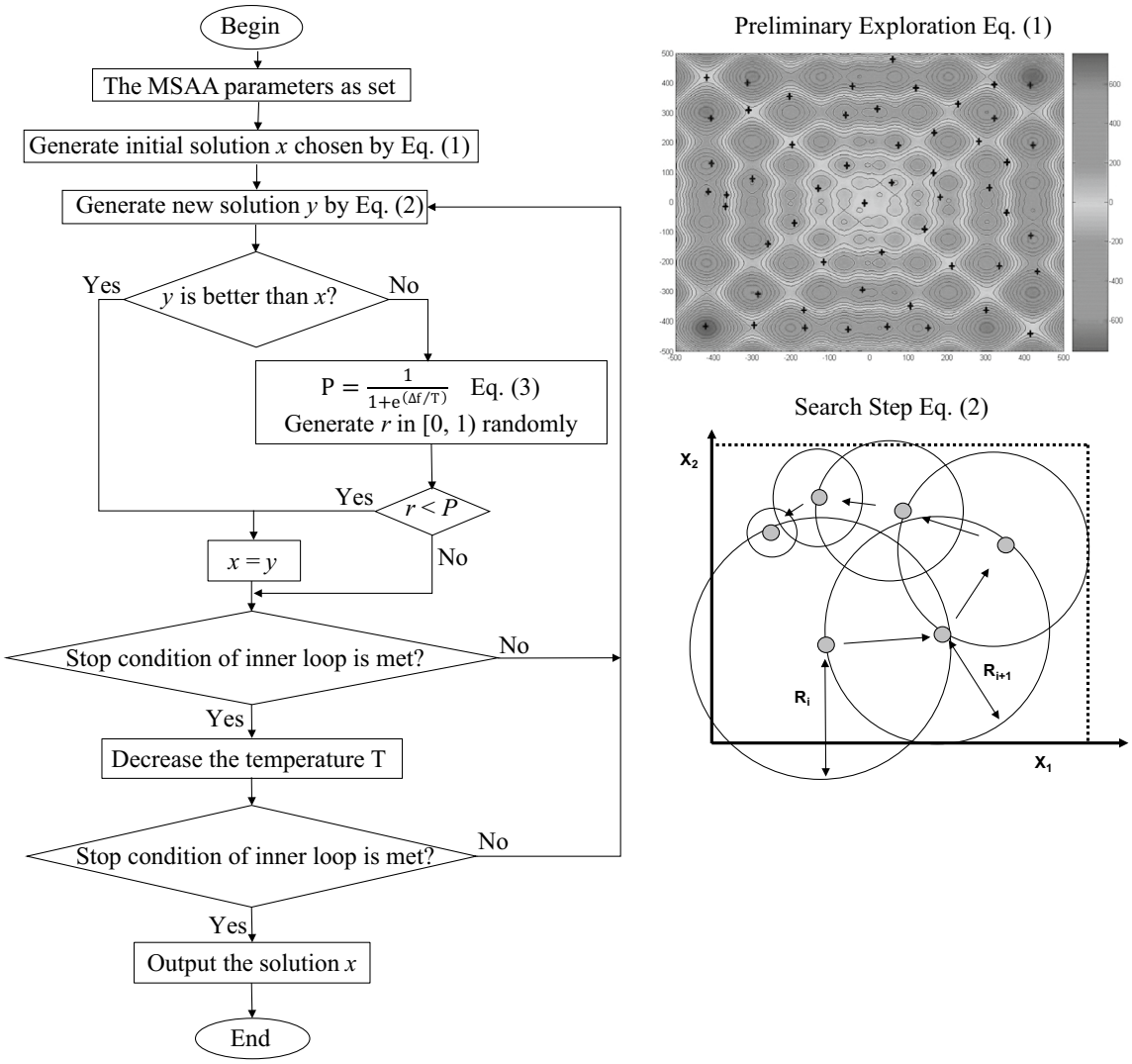

Fig. 1 The MSAA flowchart

where β is the breaking coefficient; $N(0, 1)$ a Gaussian random number with mean 0 and standard deviation 1, and $L(d)$ is the length of the dth dimension of the search space. According to Zheng [35], it is recommended to set β to 0.001–0.01, and k_{max} to min(12, D/2), where D is the dimension of the problem.

This concept is exported to MSAA. Thus, the search step phase of the MSAA is replaced by breaking phase to improve the convergence ability and set a good balance between exploration and exploitation.

4 The 30 benchmark functions of the CEC2014

The validity of the proposed I-MSAA is confirmed on 30 benchmark functions of the CEC2014 special session on single objective real-parameter numerical optimization [33]. The benchmark suite covers various types of function optimization problems, as summarized in Table 2. The algorithm is coded in Matlab program and executed

Table 2 Summary of the CEC 2014 benchmark functions [33]

Type	Function	Optimum
Unimodal	f_1: rotated high conditioned elliptic function	100
	f_2: rotated bent cigar function	200
	f_3: rotated discus function	300
Multimodal	f_4: shifted and rotated Rosenbrock function	400
	f_5: shifted and rotated Ackley's function	500
	f_6: shifted and rotated Weierstrass function	600
	f_7: shifted and rotated Griewank's function	700
	f_8: shifted Rastrigin's function	800
	f_9: Shifted and rotated Rastrigin's function	900
	f_{10}: shifted Schwefel's function	1000
	f_{11}: shifted and rotated Schwefel's function	1100
	f_{12}: shifted and rotated Katsuura function	1200
	f_{13}: shifted and rotated HappyCat function	1300
	f_{14}: Shifted and rotated HGBat function	1400
	f_{15}: shifted and rotated expanded Griewank's plus Rosenbrock's function	1500
	f_{16}: shifted and rotated expanded Scaffe's f_6 function	1600
Hybrid	f_{17}: hybrid function 1 (f_9, f_8, f_1)	1700
	f_{18}: hybrid function 2 (f_2, f_{12}, f_8)	1800
	f_{19}: hybrid function 3 (f_7, f_6, f_4, f_{14})	1900
	f_{20}: hybrid function 4 (f_{12}, f_3, f_{13}, f_8)	2000
	f_{21}: hybrid function 5 ($f_{14}, f_{12}, f_4, f_9, f_1$)	2100
	f_{22}: hybrid function 6 ($f_{10}, f_{11}, f_{13}, f_9, f_5$)	2200
Composition	f_{23}: composition function 1 (f_4, f_1, f_2, f_3, f_1)	2300
	f_{24}: composition function 2 (f_{10}, f_9, f_{14})	2400
	f_{25}: composition function 3 (f_{11}, f_9, f_1)	2500
	f_{26}: composition function 4 ($f_{11}, f_{13}, f_1, f_6, f_7$)	2600
	f_{27}: composition function 5 ($f_{14}, f_9, f_{11}, f_6, f_1$)	2700
	f_{28}: composition function 6 ($f_{15}, f_{13}, f_{11}, f_{16}, f_1$)	2800
	f_{29}: composition function 7 (f_{17}, f_{18}, f_9)	2900
	f_{30}: composition function 8 (f_{20}, f_{21}, f_{22})	3000

using an Intel Core i7-3630QM system 2.4 GHz with 8 GB RAM. For result verification, the comparison is made between several optimization algorithms (IWO, BBO, GSA, HuS, BA, WWO, SOS, ISOS, MSAA, and I-MSAA). In this study, 30D functions are used with search ranges as [− 100, 100] and set the FEmax to 150,000. All results are collected from 60 independent runs on each test function. For all examples, the population size (preliminary exploration), initial temperature (T_{initial}), final temperature (T_{final}), $\text{np}_{\max}$, β, and $k_{\max}$ are set as 200, 1, 1×10^{-3}, 300, 0.001, and min(12, D/2), respectively. Sensitivity analyses on these parameters are investigated in [29, 31, 32, 35]. Statistical tests are essential to check significance improvements by a proposed method over the existing methods. Therefore, the Friedman rank test on the results of I-MSAA, MSAA, and other state-of-the-art algorithms is used. The test is performed on the average and standard deviation (SD) of functional values obtained.

Table 3 shows the comparative average of fitness value. It can be seen that I-MSAA gives best results for unimodal functions, multimodal functions, and composition functions. In hybrid functions, I-MSAA is only surpassed by the WWO and IWO algorithms. Furthermore, I-MSAA ranks better compared to MSAA for all type functions. Finally, I-MSAA ranks first for overall performance. Table 4 shows the comparative SD of fitness value. The results indicate that I-MSAA is second best among the considered algorithms. These results confirm the merits of the proposed algorithm.

5 Problem definition

The goal of the problem is designing the member sizes of the structure so that its weight is minimized while satisfying some constraints on the natural frequencies. Member cross-sectional areas are considered as continuous design

Table 3 Comparative average of fitness values of the CEC2014 (the results of first eight algorithms are as per [27])

Function	WWO	BA	Hus	GSA	BBO	IWO	SOS	ISOS	MSAA	I-MSAA
f_1	6.281E+05	3.166E+08	5.556E+06	1.441E+07	2.726E+07	1.463E+06	1.027E+06	9.822E+05	9.954E+05	8.521E+05
f_2	3.304E+02	2.571E+10	1.007E+04	8.771E+03	4.012E+06	1.767E+04	2.132E+02	2.053E+02	2.842E+02	2.069E+02
f_3	5.268E+02	7.200E+04	5.020E+02	4.538E+04	1.310E+04	8.167E+03	9.389E+02	7.790E+02	7.803E+02	5.275E+02
Friedman value of f_1–f_3	8	30	15	23	26	21	14	8	13	7
Friedman rank of f_1–f_3	2	10	6	8	9	7	5	2	4	1
f_4	4.170E+02	3.698E+03	5.069E+02	6.764E+02	5.388E+02	5.003E+02	4.683E+02	4.598E+02	4.702E+02	4.202E+02
f_5	5.200E+02	5.210E+02	5.207E+02	5.200E+02	5.202E+02	5.200E+02	5.206E+02	5.203E+02	5.209E+02	5.200E+02
f_6	6.060E+02	6.364E+02	6.231E+02	6.196E+02	6.140E+02	6.022E+02	6.109E+02	6.105E+02	6.133E+02	6.065E+02
f_7	7.000E+02	9.107E+02	7.000E+02	7.000E+02	7.010E+02	7.000E+02	7.000E+02	7.000E+02	7.000E+02	7.000E+02
f_8	8.011E+02	1.070E+03	9.401E+02	8.005E+02	8.775E+02	8.437E+02	8.521E+02	8.147E+02	8.020E+02	8.007E+02
f_9	9.611E+02	1.250E+03	1.012E+03	1.060E+03	9.514E+02	9.461E+02	9.705E+02	9.543E+02	9.652E+02	9.475E+02
f_{10}	1.582E+03	6.426E+03	2.254E+03	4.392E+03	1.002E+03	2.565E+03	2.107E+03	1.157E+03	1.103E+03	1.002E+03
f_{11}	3.349E+03	8.152E+03	3.303E+03	5.099E+03	3.247E+03	2.887E+03	4.017E+03	2.882E+03	3.103E+03	2.901E+03
f_{12}	1.200E+03	1.203E+03	1.200E+03	1.200E+03	1.200E+03	1.200E+03	1.201E+03	1.200E+03	1.201E+03	1.200E+03
f_{13}	1.300E+03	1.304E+03	1.300E+03	1.300E+03	1.301E+03	1.300E+03	1.300E+03	1.300E+03	1.300E+03	1.300E+03
f_{14}	1.400E+03	1.473E+03	1.400E+03	1.400E+03	1.400E+03	1.400E+03	1.400E+03	1.400E+03	1.400E+03	1.400E+03
f_{15}	1.503E+03	1.945E+05	1.517E+03	1.503E+03	1.515E+03	1.504E+03	1.518E+03	1.511E+03	1.514E+03	1.509E+03
f_{16}	1.610E+03	1.613E+03	1.612E+03	1.614E+03	1.610E+03	1.610E+03	1.611E+03	1.609E+03	1.611E+03	1.610E+03
Friedman value of f_4–f_{16}	93	69	81	51	90	55	77	29	93	69
Friedman rank of f_4–f_{16}	8	5	7	3	8	4	6	1	8	5
f_{17}	2.662E+04	4.641E+06	1.981E+05	5.786E+05	4.299E+06	8.644E+04	1.432E+05	1.767E+05	1.502E+05	1.205E+05
f_{18}	2.026E+03	1.219E+08	3.781E+03	2.290E+03	2.842E+04	5.787E+03	8.320E+03	5.690E+03	6.214E+03	5.601E+03
f_{19}	1.908E+03	2.005E+03	1.931E+03	1.995E+03	1.928E+03	1.908E+03	1.923E+03	1.908E+03	1.956E+03	1.908E+03
f_{20}	5.364E+03	1.936E+04	3.866E+04	2.242E+04	3.141E+04	2.993E+03	5.770E+03	6.983E+03	6.185E+03	4.126E+03
f_{21}	3.867E+04	1.095E+06	6.046E+04	1.706E+05	4.856E+05	3.907E+04	6.860E+04	9.112E+04	7.045E+04	4.105E+04
f_{22}	2.482E+03	3.134E+03	3.073E+03	3.161E+03	2.723E+03	2.346E+03	2.496E+03	2.475E+03	2.746E+03	2.506E+04
Friedman value of f_{17}–f_{22}	10	55	38	44	47	15	30	28	37	26
Friedman rank of f_{17}–f_{22}	1	10	7	8	9	2	5	4	6	3
f_{23}	2.615E+03	2.589E+03	2.616E+03	2.564E+03	2.617E+03	2.615E+03	2.615E+03	2.615E+03	2.616E+03	2.615E+03
f_{24}	2.631E+03	2.601E+03	2.658E+03	2.600E+03	2.635E+03	2.618E+03	2.600E+03	2.600E+03	2.602E+03	2.600E+03
f_{25}	2.708E+03	2.706E+03	2.725E+03	2.700E+03	2.712E+03	2.705E+03	2.700E+03	2.700E+03	2.700E+03	2.700E+03
f_{26}	2.700E+03	2.703E+03	2.785E+03	2.800E+03	2.706E+03	2.700E+03	2.700E+03	2.700E+03	2.702E+03	2.700E+03
f_{27}	3.093E+03	3.321E+03	4.965E+03	3.720E+03	3.397E+03	3.081E+03	3.266E+03	3.243E+03	3.315E+03	3.102E+03
f_{28}	3.888E+03	4.535E+03	5.415E+03	5.282E+03	3.801E+03	3.693E+03	3.847E+03	3.768E+03	3.895E+03	3.770E+03
f_{29}	4.094E+03	4.611E+06	2.078E+06	5.215E+04	1.492E+05	1.640E+04	1.724E+06	5.733E+05	4.521E+04	5.204E+03
f_{30}	5.652E+03	1.981E+05	1.672E+04	1.914E+04	1.621E+04	9.196E+03	5.941E+03	5.377E+03	5.875E+03	5.670E+03

Table 3 (continued)

Function	WWO	BA	Hus	GSA	BBO	IWO	SOS	ISOS	MSAA	I-MSAA
Friedman value of f_{23}–f_{30}	56	75	52	61	33	37	26	44	23	56
Friedman rank of f_{23}–f_{30}	8	10	7	9	3	5	2	6	1	8
Overall Friedman value	93	270	221	188	215	120	171	117	171	85
Overall Friedman rank	2	10	9	7	8	4	5	3	5	1

variables. Each variable should be chosen within a permissible range. The mathematical formulation for this problem can be expressed as follows:

$$\text{Minimize} \quad W(A) = \sum_{i=1}^{n} \rho_i A_i L_i$$
$$\text{Subject to} \begin{cases} \omega_j \le \omega_j^* \\ \omega_k \ge \omega_k^* \\ A_{i,\min} \le A_i \le A_{i,\max} \end{cases}, \tag{4}$$

where W is the weight of the structure; n the total number of members of the structure; ρ_i, A_i, and L_i stand for the material density, the cross-sectional area, and the length of the ith member, respectively; ω_j and ω_k the jth and kth natural frequencies of the structure, respectively; ω_j^* and ω_k^* the upper and lower bounds corresponding to ω_j and ω_k, respectively; $A_{i,\min}$ and $A_{i,\max}$ are the lower and upper bounds of A_i, respectively.

6 Truss problems and discussions

To evaluate the feasibility and validity of the I-MSAA, the following classical truss sizing problems (Fig. 2) are optimized and the results are compared with the previous results obtained through various existing metaheuristics: (1) 10-bar planar truss; (2) 72-bar space truss; (3) 120-bar dome truss, and (4) 200-bar planar truss. The design considerations of the problems are given in Table 5.

In all design examples, the parameters used in the I-MSAA are: (1) as it is demonstrated in the previous studies [29, 31, 32], the population size (preliminary exploration), initial temperature (T_{initial}), and final temperature (T_{final}) are set as 200, 1, 1×10^{-3}, respectively; (2) according to [35], β and $k_{\max}$ are set as 0.001 and min(12, D/2), respectively; (3) according to [32], the maximum number of perturbations ($\text{np}_{\max}$) at the same temperature can be chosen in the range of 100–300. Evidence gathered from sensitivity analysis led to set $\text{np}_{\max}$ as 200. These numbers of perturbations have been obtained in this work by examining its effect to find a balance between accuracy and computational cost for each of the problems.

The iterative process is terminated when the algorithm reaches the final temperature. The proposed I-MSAA algorithm and finite element analysis are coded in Matlab program and executed using an Intel Core i7-3630QM system 2.4 GHz with 8 GB RAM. Statistical results, obtained for 100 independent runs, are presented in terms of the best weight, average weight, standard deviation (SD), the corresponding iterations number (NI), and frequency responses. It is important to note that all presented I-MSAA designs

Table 4 Comparative SD of fitness values of the CEC2014 (the results of first eight algorithms are as per [27])

Function	WWO	BA	Hus	GSA	BBO	IWO	SOS	ISOS	MSAA	I-MSAA
f_1	2.445E+05	1.047E+08	2.620E+06	1.319E+07	1.672E+07	5.717E+05	7.329E+05	7.055E+05	7.295E+05	3.013E+05
f_2	2.022E+02	7.554E+09	6.013E+03	2.903E+03	1.549E+06	8.673E+03	2.028E+01	1.718E+01	5.421E+01	1.820E+01
f_3	1.846E+02	1.755E+04	5.406E+02	1.043E+04	1.276E+04	2.693E+03	5.276E+02	6.239E+02	9.541E+02	2.013E+02
f_4	3.636E+01	1.974E+03	3.662E+01	5.151E+01	3.835E+01	2.880E+01	3.175E+01	3.572E+01	3.954E+01	3.215E+01
f_5	7.000E−04	4.810E−02	7.830E−02	6.000E−04	4.220E−02	3.800E−03	8.010E−02	6.670E−02	4.125E−02	2.140E−03
f_6	2.620E+00	1.559E+00	2.178E+00	1.832E+00	2.354E+00	1.122E+00	2.568E+00	2.396E+00	2.466E+00	2.379E+00
f_7	6.300E−03	3.232E+01	5.560E−02	1.000E−03	2.640E−02	1.210E−02	2.140E−02	1.830E−02	2.563E−01	1.425E−02
f_8	2.336E+00	2.565E+01	1.273E+01	2.063E−01	2.069E+01	1.011E+01	1.232E+01	3.349E+00	1.365E+01	1.985E+00
f_9	1.110E+01	4.413E+01	2.599E+01	1.743E+01	1.144E+01	1.139E+01	2.408E+01	1.344E+01	2.547E+01	1.352E+01
f_{10}	3.616E+02	5.187E+02	4.332E+02	3.610E+02	6.800E−01	3.800E+02	3.441E+02	4.084E+01	2.145E+02	3.541E+01
f_{11}	2.892E+02	3.622E+02	4.655E+02	5.673E+02	5.116E+02	4.477E+02	8.351E+02	4.541E+02	5.273E+02	3.015E+02
f_{12}	5.610E−02	3.339E−01	7.770E−02	1.000E−03	5.620E−02	1.480E−02	1.833E−01	5.730E−02	2.541E−01	1.474E−01
f_{13}	6.410E−02	5.483E−01	6.500E−02	6.650E−02	1.061E−01	6.500E−02	8.640E−02	7.100E−02	1.012E−01	8.214E−02
f_{14}	4.410E−02	1.395E+01	4.740E−02	4.230E−02	1.992E−01	1.191E−01	1.296E−01	5.120E−02	1.147E−01	4.625E−02
f_{15}	7.753E−01	1.403E+05	3.270E+00	7.297E−01	4.298E+00	8.484E−01	3.798E+00	3.717E+00	4.125E+00	7.584E−01
f_{16}	4.667E−01	1.904E−01	7.249E−01	3.428E−01	5.923E−01	6.144E−01	6.059E−01	7.314E−01	5.474E−01	5.596E−01
f_{17}	1.240E+04	1.790E+06	1.605E+05	2.199E+05	4.192E+06	6.847E+04	1.590E+05	1.645E+05	9.855E+04	2.010E+04
f_{18}	1.252E+02	1.003E+08	2.247E+03	3.779E+02	1.967E+04	3.690E+03	1.031E+04	5.150E+03	1.287E+04	3.818E+02
f_{19}	1.378E+00	2.032E+01	3.315E+01	3.432E+01	2.769E+01	1.655E+00	2.661E+01	1.787E+00	8.714E+00	1.413E+00
f_{20}	3.177E+03	1.028E+04	8.493E+03	1.386E+04	1.760E+04	7.004E+02	3.295E+03	3.393E+03	5.177E+03	3.236E+03
f_{21}	3.556E+04	7.507E+05	4.243E+04	6.529E+04	3.346E+05	2.301E+04	8.009E+04	1.078E+05	1.525E+05	4.015E+04
f_{22}	1.429E+02	2.054E+02	2.673E+02	2.500E+02	2.344E+02	7.339E+01	1.515E+02	1.454E+02	2.115E+02	1.502E+02
f_{23}	1.447E−01	1.284E+02	8.448E−01	6.450E+01	1.318E+00	7.950E−02	0.000E+00	0.000E+00	4.175E−01	1.300E−04
f_{24}	6.885E+00	1.200E+00	1.249E+01	1.710E−02	5.974E+00	1.077E+01	1.300E−03	1.500E−03	5.645E−01	1.235E−03
f_{25}	2.001E+00	1.498E+01	6.269E+00	1.319E+00	3.010E+00	8.076E−01	0.000E+00	0.000E+00	1.113E+00	0.000E+00
f_{26}	6.500E−02	5.372E−01	3.533E+01	5.400E−03	2.202E+01	5.430E−02	8.620E−02	9.550E−02	9.965E−01	8.450E−02
f_{27}	5.901E+01	6.462E+01	6.825E+02	3.505E+02	6.353E+01	3.503E+01	1.463E+02	1.363E+02	6.847E+01	6.029E+01
f_{28}	3.607E+02	5.929E+02	4.613E+02	7.153E+02	9.334E+01	4.121E+01	1.901E+02	1.266E+02	2.015E+02	1.252E+02
f_{29}	3.596E+02	2.831E+06	7.705E+06	3.781E+05	1.114E+06	5.140E+03	3.468E+06	2.146E+06	5.478E+05	6.417E+05
f_{30}	7.381E+02	9.106E+04	6.583E+03	1.841E+04	6.076E+03	2.079E+03	3.249E+03	1.107E+03	5.141E+04	1.000E+03
Overall Friedman value	93	247	213.5	160	213	110.5	175.5	146.5	195	96
Overall Friedman rank	1	10	9	5	8	3	6	4	7	2

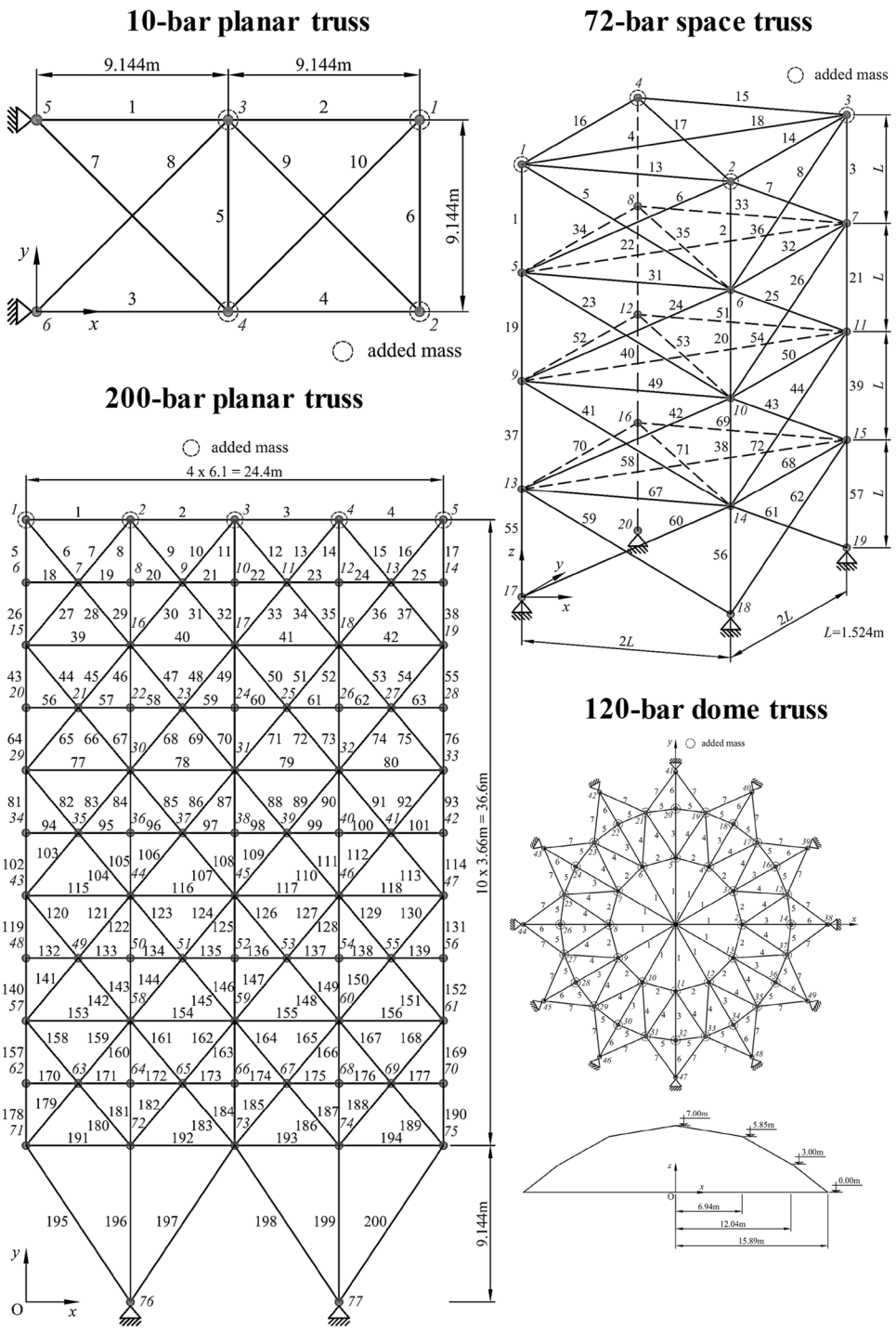

Fig. 2 Benchmark trusses' problems

are feasible. The results and discussions of the benchmark problems are explained in the following sections.

6.1 10-Bar planar truss

The first design example is the 10-bar planar truss structure shown in Fig. 2. A lumped mass of 454 kg is added in all free nodes. Table 6 presents a comparison of optimal results obtained by the proposed algorithm and other methods. The results indicate that I-MSAA (529.75 kg) achieves better optimal weight than DPSO, SBO, VPS, and MSAA algorithms, but slight heavier design than ReDe, ISOS, and AHEFA. However, the proposed algorithm requires less NI than ReDe (6200 NI for I-MSAA and 8300 NI for ReDe) to reach final solution. It can be seen that the convergence speed of the DPSO, ISOS, and AHEFA is faster than that of

Table 5 Design considerations of the benchmark trusses

	10-Bar planar truss	72-Bar space truss	120-Bar dome truss	200-Bar planar truss
Modulus of elasticity E (N/m^2)	6.98×10^{10}	6.98×10^{10}	2.1×10^{11}	2.1×10^{11}
Material density ρ (kg/m^3)	2770	2770	7971.81	7860
Cross-sectional area bounds (cm^2)	$0.645\le A\le50$	$0.645\le A\le30$	$1\le A\le129.3$	$0.1\le A\le30$
Frequency constraints (Hz)	$f_1\ge7$ $f_2\ge15$ $f_3\ge20$	$f_1=4$ $f_3\ge6$	$f_1\ge9$ $f_2\ge11$	$f_1\ge5$ $f_2\ge10$ $f_3\ge15$

Table 6 Optimal design parameters for the 10-bar planar truss by different algorithms

Variables (cm^2)	DPSO [15]	SBO [19]	VPS [23]	ReDe [25]	ISOS [27]	AHEFA [28]	MSAA	I-MSAA
A_1	35.944	35.5994	35.1471	35.1565	35.2654	35.1714	32.9710	32.5584
A_2	15.53	14.9956	14.6668	14.7605	14.6803	14.7203	15.5925	15.4787
A_3	35.285	35.4806	35.6889	35.1187	34.4273	35.1074	32.8514	32.7556
A_4	15.385	14.7646	15.0929	14.7275	14.9605	14.6986	15.5942	15.5750
A_5	0.648	0.6450	0.645	0.6450	0.6450	0.6451	0.6454	0.6454
A_6	4.583	4.6305	4.6221	4.5558	4.5927	4.5593	4.6552	4.6612
A_7	23.61	24.3272	23.5552	23.7199	23.3417	23.7330	26.1179	26.1090
A_8	23.599	23.8528	24.468	23.6304	23.8236	23.6795	26.1350	26.2576
A_9	13.135	12.6797	12.7198	12.3827	12.8497	12.3987	11.9983	11.7470
A_{10}	12.357	12.6375	12.6845	12.4580	12.5321	12.4231	11.9339	11.8823
Best weight (kg)	532.39	532.05	530.77	524.45	524.73	524.45	532.04	529.75
f_1 (Hz)	7.0000	7.0000	7.0000	7.0000	7.0001	7.0000	7.0000	7.0000
f_2 (Hz)	16.1870	16.1660	16.1599	16.1924	16.1703	16.1920	15.8458	15.8235
f_3 (Hz)	20.0000	20.0000	20.0000	20.0000	20.0024	20.0000	20.0000	20.0000
Average weight (kg)	537.80	533.45	535.64	524.76	530.03	525.16	532.06	530.11
SD (kg)	4.02	2.20	2.55	1.11	3.48	1.92	0.01	0.11
NI	6000	10,000	30,000	8300	4000	5860	7130	6200

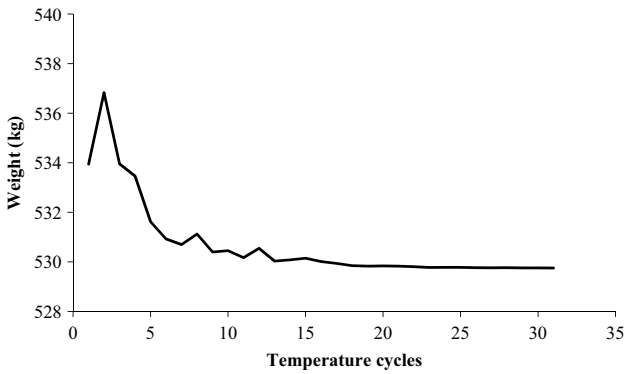

Fig. 3 Convergence curve of I-MSAA for the 10-bar planar truss

the I-MSAA (6000 NI for DPSO, 4000 NI for ISOS, 5860 for AHEFA); however, the I-MSAA is more stable than the DPSO, ISOS, and AHEFA through the best value of SD (0.11 kg for I-MSAA, 3.48 kg for ISOS, 1.92 kg for the AHEFA, and 4.02 kg for the DPSO). Finally, with respect to the SD, I-MSAA ranks second among the considered metaheuristics, only being surpassed by MSAA (0.01 kg). Natural frequencies optimal obtained by the I-MSAA show that none of the frequency constraints are violated. Figure 3 shows the convergence curve of the best design of I-MSAA for this problem.

6.2 72-Bar space truss

The 72-bar space truss shown in Fig. 2 is the second numerical example. The bars are categorized into 16 groups by considering geometrical symmetry. A lumped mass of 2770 kg is attached at all top nodes (nodes 1–4). Optimal results obtained by the I-MSAA and the other optimization algorithms published in the literature are reported in Table 7. It can be seen that the acquired result by the proposed algorithm (324.43 kg) is better than the other methods (327.51 kg for the CSS-BBBC, 327.65 kg for the DPSO, 327.55 kg for the SBO, 327.65 kg for the VPS, 325.01 kg for the ISOS, and 324.97 kg for the MSAA). Moreover, I-MSAA requires

Table 7 Optimal design parameters for the 72-bar space truss by different algorithms

Variables (cm^2)	CSS-BBBC [13]	DPSO [15]	SBO [19]	VPS [23]	ReDe [25]	ISOS [27]	AHEFA [28]	MSAA	I-MSAA
A_1–A_4	2.854	3.5498	3.4917	3.5017	3.5327	3.3563	3.5612	3.4927	3.3524
A_5–A_{12}	8.301	7.8356	7.9414	7.9340	7.8303	7.8726	7.8736	7.8573	7.7448
A_{13}–A_{16}	0.645	0.6450	0.6450	0.6450	0.6453	0.6450	0.6450	0.6450	0.6450
A_{17}–A_{18}	0.645	0.6450	0.6450	0.6450	0.6459	0.6450	0.6451	0.6474	0.6450
A_{19}–A_{22}	8.202	8.1183	8.1154	8.0215	8.0029	8.5798	7.9710	7.8897	7.5541
A_{23}–A_{30}	7.043	8.1338	8.0533	7.9826	7.9135	7.6566	7.8928	8.0057	7.8746
A_{31}–A_{34}	0.645	0.6450	0.6450	0.6450	0.6451	0.7417	0.6450	0.6450	0.6450
A_{35}–A_{36}	0.645	0.6450	0.6450	0.6450	0.6451	0.6450	0.6451	0.6454	0.6450
A_{37}–A_{40}	16.328	12.6231	12.8569	12.8175	12.7626	13.0864	12.5404	12.6034	12.5877
A_{41}–A_{48}	8.299	8.0971	8.0425	8.1129	7.9657	8.0764	7.9639	7.9616	8.0790
A_{49}–A_{52}	0.645	0.6450	0.6451	0.6450	0.6452	0.6450	0.6459	0.6451	0.6450
A_{53}–A_{54}	0.645	0.6450	0.6450	0.6450	0.6450	0.6937	0.6462	0.6450	0.6450
A_{55}–A_{58}	15.048	17.3908	17.2136	17.3362	16.9041	16.2517	17.1323	17.1604	17.8079
A_{59}–A_{66}	8.268	8.0634	8.0804	8.1010	8.0434	8.1703	8.0216	8.0368	8.0575
A_{67}–A_{70}	0.645	0.6450	0.6450	0.6450	0.6451	0.6450	0.6450	0.6450	0.6450
A_{71}–A_{72}	0.645	0.6450	0.6450	0.6450	0.6473	0.6450	0.6451	0.6450	0.6486
Best weight (kg)	327.51	327.65	327.55	327.65	324.25	325.01	324.24	324.97	324.43
f_1 (Hz)	4.0000	4.0000	4.0000	4.0000	4.0000	4.0000	4.0000	4.0000	4.0000
f_3 (Hz)	6.0040	6.0000	6.0000	6.0000	6.0001	6.0008	6.0000	6.0000	6.0000
Average weight (kg)	–	327.76	327.68	327.67	324.32	329.47	324.41	325.13	324.52
SD (kg)	–	0.06	0.07	0.02	0.05	2.66	0.24	0.18	0.07
NI	–	20,000	15,000	30,000	10,840	4000	8860	7130	6200

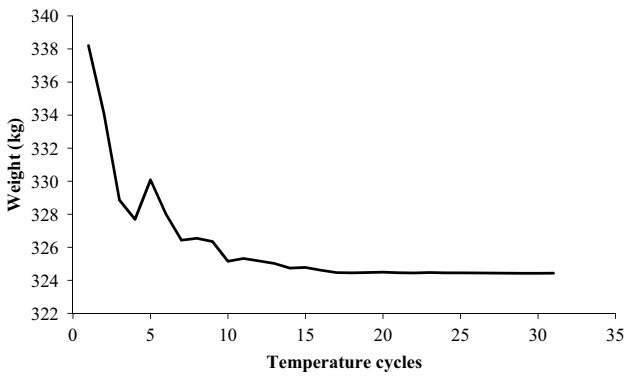

Fig. 4 Convergence curve of I-MSAA for the 72-bar space truss

fewer NI than the DPSO, SBO, VPS, ReDe, AHEFA, and MSAA (6200 NI for I-MSAA, 20,000 NI for DPSO, 15,000 NI for SOB, 30,000 for VPS, 10,840 NI for ReDe, 8860 NI for AHEFA, and 7130 for MSAA). The average weight benefit for I-MSAA is 3.24, 3.16, 3.15, 4.95, and 0.61 kg as compared to those obtained from DPSO, SBO, VPS, ISOS, and MSAA, respectively. Finally, I-MSAA obtains a low SD (0.07 kg) that evidences the stability of the proposed algorithm. Natural frequencies indicate the feasibility of the obtained design by I-MSAA. The convergence curve of the best design of I-MSAA for this problem is shown in Fig. 4.

6.3 120-Bar dome truss

The third example is the 120-bar dome truss shown in Fig. 2. A lumped mass is attached to all free nodes as follows: 3000 kg at node one, 500 kg at nodes 2 through 13 kg, and 100 kg at the rest of the nodes. The members of the structure are categorized into seven groups using symmetry about the z axis. Table 8 compares the results of I-MSAA with other optimization methods. As observed, the I-MSAA provides the best result with 8707.01 kg while the others give larger weights, namely CSS-BBBC (9046.34 kg), DPSO (8890.48 kg), CBO (8889.13 kg), HALC-PSO (8889.96 kg), VPS (8888.74 kg), ReDe (8707.32 kg), ISOS (8710.06 kg), and MSAA (8707.39 kg). Moreover, I-MSAA requires 6200 NI to converge the optimal solution, while the HALC-PSO and VPS need 17,000 and 30,000 NI, respectively. From the obtained average weight (8707.42 kg) and SD (0.08 kg) values of the proposed method, it can be seen that the I-MSAA is stable. The SD obtained with I-MSAA ranks first among the considered metaheuristics. The convergence curve of the best design of I-MSAA for this problem is shown in Fig. 5.

Table 8 Optimal design parameters for the 120-bar dome truss by different algorithms

Variables (cm^2)	CSS-BBBC [13]	DPSO [15]	CBO [16]	HALC-PSO [18]	VPS [23]	ReDe [25]	ISOS [27]	MSAA	I-MSAA
A_1	17.478	19.607	19.6917	19.8905	19.6836	19.5131	19.6662	20.0425	19.6068
A_2	49.076	41.290	41.1421	40.4045	40.9581	40.3914	39.8539	39.4775	40.5483
A_3	12.365	11.136	11.1550	11.2057	11.3325	10.6066	10.6127	13.6425	13.4167
A_4	21.979	21.025	21.3207	21.3768	21.5387	21.1415	21.2901	20.4928	20.2411
A_5	11.190	10.060	9.8330	9.8669	9.8867	9.8057	9.7911	9.0488	9.1521
A_6	12.590	12.758	12.8520	12.7200	12.7116	11.7781	11.7899	15.2658	15.8831
A_7	13.585	15.414	15.1602	15.2236	14.9330	14.8163	14.7437	12.9846	12.9856
Best weight (kg)	9046.34	8890.48	8889.13	8889.96	8888.74	8707.32	8710.06	8707.39	8707.01
f_1 (Hz)	9.000	9.0001	9.0000	9.0000	9.0000	9.0000	9.0001	9.0000	9.0000
f_2 (Hz)	11.007	11.0007	11.0000	11.0000	11.0000	11.0000	10.9998	11.0000	11.0000
Average weight (kg)	–	8895.99	8891.25	8900.39	8896.04	8707.52	8728.56	8709.96	8707.42
SD (kg)	–	4.26	1.79	6.38	6.65	0.15	14.23	3.43	0.08
NI	–	6000	6000	17,000	30,000	5080	4000	3100	6200

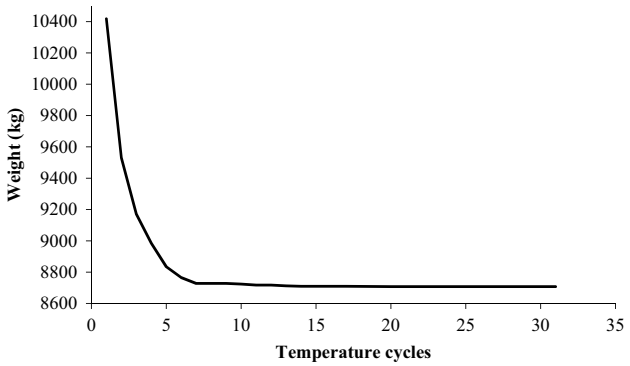

Fig. 5 Convergence curve of I-MSAA for the 120-bar dome truss

6.4 200-Bar planar truss

The last design problem is the 200-bar planar truss (Fig. 2). A lumped mass is attached at all top nodes (nodes 1–5). The bars are grouped into 29 by seeing symmetry as reported in the previous studies. Table 9 provides a comparison of optimal results obtained by the I-MSAA and different metaheuristics. It can be seen that the best weight obtained by I-MSAA (2156.83 kg) is better than those given by AHEFA (2160.74 kg), CSS-BBBC (2298.61 kg), ISOS (2169.46 kg), SOS (2180.32 kg), and MSAA (2157.28 kg). The I-MSSA design is slightly worse than SBO (2156.51 kg) and HALC-PSO (2156.73 kg); however, the convergence speed of the I-MSAA is faster than these algorithms (23,000 NI for SBO and 13,000 NI for HALC-PSO). The results also indicate that, I-MSAA is more stable than SOS, ISOS, and MSAA with the smallest SD (1.13 kg for I-MSAA, 83.59 kg for SOS, 43.48 kg for ISOS, and 2.96 for MSAA). Frequency values show that all constraints of the 200-bar planar truss are satisfied by the I-MSAA. Figure 6 shows the convergence curve of the best design of I-MSAA for this problem.

7 Conclusions

This paper proposes how to improve the modified simulated annealing algorithm by including breaking operator, a concept borrowed from water wave optimization. The breaking operator is implemented to improve exploitation ability of MSAA in the search process. The new algorithm is called I-MSAA. Thirty benchmark functions of the CEC2014 and four classical truss sizing problems with frequency constraints are tested to verify the effectiveness and robustness of the proposed algorithm. In the benchmark functions, I-MSAA is better or competitive for obtaining results based on the mean and SD of functional values obtained over the stated runs as compared to MSAA and other metaheuristics. In the truss optimization, I-MSAA always achieve a better design than MSAA. Additionally, the numerical results show the ability of this algorithm to produce competitive results compared to those of the other metaheuristic algorithms presented in

Table 9 Optimal design parameters for the 200-bar planar truss by different algorithms

Variables (cm^2)	CSS-BBBC [13]	HALC-PSO [18]	SBO [19]	SOS [26]	ISOS [27]	AHEFA [28]	MSAA	I-MSAA
A_1	0.2934	0.3072	0.3040	0.4781	0.3072	0.2993	0.3034	0.3181
A_2	0.5561	0.4545	0.4478	0.4481	0.5075	0.4508	0.5177	0.4603
A_3	0.2952	0.1000	0.1000	0.1049	0.1001	0.1001	0.1000	0.1000
A_4	0.1970	0.1000	0.1000	0.1045	0.1000	0.1000	0.1000	0.1000
A_5	0.8340	0.5080	0.5075	0.4875	0.5893	0.5123	0.5699	0.5271
A_6	0.6455	0.8276	0.8219	0.9353	0.8328	0.8205	0.8187	0.8066
A_7	0.1770	0.1023	0.1003	0.1200	0.1431	0.1011	0.1000	0.1009
A_8	1.4796	1.4357	1.4240	1.3236	1.3600	1.4156	1.4361	1.5387
A_9	0.4497	0.1007	0.1001	0.1015	0.1039	0.1000	0.1000	0.1001
A_{10}	1.4556	1.5528	1.5929	1.4827	1.5114	1.5742	1.4599	1.6293
A_{11}	1.2238	1.1529	1.1597	1.1384	1.3568	1.1597	1.1381	1.1467
A_{12}	0.2739	0.1522	0.1275	0.1020	0.1024	0.1338	0.1205	0.1318
A_{13}	1.9174	2.9564	2.9765	2.9943	2.9024	2.9672	2.9032	2.8387
A_{14}	0.1170	0.1003	0.1001	0.1562	0.1000	0.1000	0.1006	0.1000
A_{15}	3.5535	3.2242	3.2456	3.4330	3.4120	3.2722	3.7168	2.7781
A_{16}	1.3360	1.5839	1.5818	1.6816	1.4819	1.5762	1.5246	1.5820
A_{17}	0.6289	0.2818	0.2566	0.1026	0.2587	0.2562	0.2056	0.1409
A_{18}	4.8335	5.0696	5.1118	5.0739	4.8291	5.0956	5.1494	5.7784
A_{19}	0.6062	0.1033	0.1001	0.1068	0.1499	0.1001	0.1021	0.1015
A_{20}	5.4393	5.4657	5.4337	6.0176	5.5090	5.4546	5.3291	4.8444
A_{21}	1.8435	2.0975	2.1016	2.0340	2.2221	2.0933	1.9882	2.0156
A_{22}	0.8955	0.6598	0.6794	0.6595	0.6113	0.6737	0.6782	0.4538
A_{23}	8.1759	7.6585	7.6581	6.9003	7.3398	7.6498	7.9359	6.4039
A_{24}	0.3209	0.1444	0.1006	0.2020	0.1559	0.1178	0.3222	0.6062
A_{25}	10.9800	8.0520	7.9468	6.8356	8.6301	8.0682	8.9235	9.2760
A_{26}	2.9489	2.7889	2.7835	2.6644	2.8245	2.8025	2.5618	2.8030
A_{27}	10.5243	10.4770	10.5277	12.1430	10.8563	10.5040	10.4026	11.6835
A_{28}	20.4271	21.3257	21.3027	22.2484	20.9142	21.2935	21.3538	21.2372
A_{29}	19.0983	10.5111	10.6207	8.9378	10.5305	10.7410	10.6476	9.7778
Best weight (kg)	2298.61	2156.73	2156.51	2180.32	2169.46	2160.74	2157.28	2156.83
f_1 (Hz)	5.010	5.000	5.000	5.0001	5.0000	5.0000	5.0000	5.0000
f_2 (Hz)	12.911	12.254	12.2141	13.4306	12.4477	12.1821	12.3405	12.3482
f_3 (Hz)	15.416	15.044	15.0192	15.2645	15.2332	15.0160	15.0001	15.0028
Average weight (kg)	–	2157.14	2156.79	2303.30	2244.64	2161.04	2161.74	2157.94
SD (kg)	–	0.24	0.21	83.59	43.48	0.18	2.96	1.13
NI	–	13,000	23,000	10,000	10,000	11,300	6200	6200

the literature in terms of best weight, SD, and NI required by the optimization process. The presented data on average weight and SD of optimized weight obtained from 100 independent runs prove the robustness of I-MSAA.

The I-MSAA is simple to implement and it can be easy to extend for various engineering optimization problems such as: simultaneous shape and topology optimization of truss structures, frame optimization, and reliability-based design optimization problems.

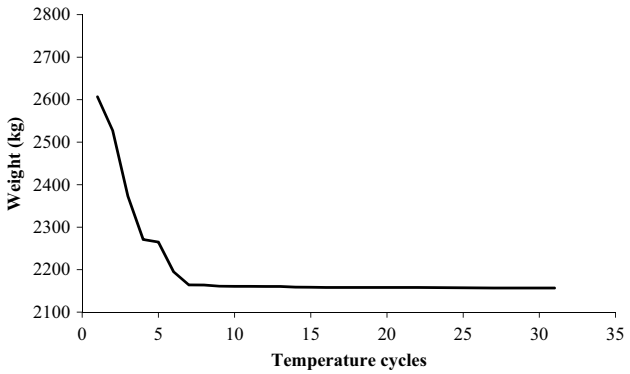

Fig. 6 Convergence curve of I-MSAA for the 200-bar planar truss

Acknowledgements The authors are thankful to Universidade Tecnológica Federal do Paraná, and for the scholarship granted to the first author by CAPES.

References

1. Grandhi RV, Venkayya VB (1988) Structural optimization with frequency constraints. AIAA J 26:858–866. https://doi.org/10.2514/3.9979
2. Pholdee N, Bureerat S (2014) Comparative performance of meta-heuristic algorithms for mass minimisation of trusses with dynamic constraints. Adv Eng Softw 75:1–13. https://doi.org/10.1016/j.advengsoft.2014.04.005
3. Grandhi RV (1993) Structural optimization with frequency constraints—a review. AIAA J 31:2296–2330. https://doi.org/10.2514/3.9979
4. Bellagamba L, Yang TY (1981) Minimum-mass truss structures with constraints on fundamental natural frequency. AIAA J 19:1452–1458. https://doi.org/10.2514/3.7875
5. Lin JH, Che WY, Yu YS (1982) Structural optimization on geometrical configuration and element sizing with statical and dynamical constraints. Comput Struct 15:507–515. https://doi.org/10.1016/0045-7949(82)90002-5
6. Ko F-T, Wang BP (1991) An improved method of optimality criteria for structural optimization. Comput Struct 41:629–636. https://doi.org/10.1016/0045-7949(91)90175-L
7. Sedaghati R, Suleman A, Tabarrok B (2002) Structural optimization with frequency constraints using the finite element force method. AIAA J 40:382–388
8. Sedaghati R (2005) Benchmark case studies in structural design optimization using the force method. Int J Solids Struct 42:5848–5871. https://doi.org/10.1016/j.ijsolstr.2005.03.030
9. Lingyun W, Mei Z, Guangming W, Guang M (2005) Truss optimization on shape and sizing with frequency constraints based on genetic algorithm. Comput Mech 35:361–368. https://doi.org/10.1007/s00466-004-0623-8
10. Wei L, Tang T, Xie X, Shen W (2011) Truss optimization on shape and sizing with frequency constraints based on parallel genetic algorithm. Struct Multidiscip Optim 43:665–682. https://doi.org/10.1007/s00158-010-0600-0
11. Gomes HM (2011) Truss optimization with dynamic constraints using a particle swarm algorithm. Expert Syst Appl 38:957–968. https://doi.org/10.1016/j.eswa.2010.07.086
12. Miguel LFF, Fadel Miguel LF (2012) Shape and size optimization of truss structures considering dynamic constraints through modern metaheuristic algorithms. Expert Syst Appl 39:9458–9467. https://doi.org/10.1016/j.eswa.2012.02.113
13. Kaveh A, Zolghadr A (2012) Truss optimization with natural frequency constraints using a hybridized CSS-BBBC algorithm with trap recognition capability. Comput Struct 102–103:14–27. https://doi.org/10.1016/j.compstruc.2012.03.016
14. Kaveh A, Zolghadr A (2011) Shape and size optimization of truss structures with frequency constraints using enhanced charged system search algorithm. Asian J Civ Eng 12:487–509
15. Kaveh A, Zolghadr A (2014) Democratic PSO for truss layout and size optimization with frequency constraints. Comput Struct 130:10–21. https://doi.org/10.1016/j.compstruc.2013.09.002
16. Kaveh A, Mahdavi VR (2015) Colliding-bodies optimization for truss optimization with multiple frequency constraints. J Comput Civ Eng 29:04014078. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000402
17. Khatibinia M, Naseralavi S (2014) Truss optimization on shape and sizing with frequency constraints based on orthogonal multi-gravitational search algorithm. J Sound Vib 333:6349–6369. https://doi.org/10.1016/j.jsv.2014.07.027
18. Kaveh A, Ilchi Ghazaan M (2015) Hybridized optimization algorithms for design of trusses with multiple natural frequency constraints. Adv Eng Softw 79:137–147. https://doi.org/10.1016/j.advengsoft.2014.10.001
19. Farshchin M, Camp CV, Maniat M (2016) Optimal design of truss structures for size and shape with frequency constraints using a collaborative optimization strategy. Expert Syst Appl 66:203–218. https://doi.org/10.1016/j.eswa.2016.09.012
20. Gonçalves MS, Lopez RH, Miguel LFF (2015) Search group algorithm: a new metaheuristic method for the optimization of truss structures. Comput Struct 153:165–184. https://doi.org/10.1016/j.compstruc.2015.03.003
21. Farshchin M, Camp CV, Maniat M (2016) Multi-class teaching–learning-based optimization for truss design with frequency constraints. Eng Struct 106:355–369. https://doi.org/10.1016/j.engstruct.2015.10.039
22. Kaveh A, Zolghadr A (2017) Cyclical parthenogenesis algorithm for layout optimization of truss structures with frequency constraints. Eng Optim 49:1317–1334. https://doi.org/10.1080/0305215X.2016.1245730
23. Kaveh A, Ilchi Ghazaan M (2017) Vibrating particles system algorithm for truss optimization with multiple natural frequency constraints. Acta Mech 228:307–322. https://doi.org/10.1007/s00707-016-1725-z
24. Kaveh A, Zolghadr A (2017) Truss shape and size optimization with frequency constraints using tug of war optimization. Asian J Civ Eng 18:311–313
25. Ho-Huu V, Nguyen-Thoi T, Truong-Khac T et al (2018) An improved differential evolution based on roulette wheel selection for shape and size optimization of truss structures with frequency constraints. Neural Comput Appl 29:167–185. https://doi.org/10.1007/s00521-016-2426-1
26. Tejani GG, Savsani VJ, Patel VK (2016) Adaptive symbiotic organisms search (SOS) algorithm for structural design optimization. J Comput Des Eng 3:226–249. https://doi.org/10.1016/j.jcde.2016.02.003
27. Tejani GG, Savsani VJ, Patel VK, Mirjalili S (2018) Truss optimization with natural frequency bounds using improved symbiotic organisms search. Knowl Based Syst 143:162–178. https://doi.org/10.1016/j.knosys.2017.12.012
28. Lieu QX, Do DTT, Lee J (2018) An adaptive hybrid evolutionary firefly algorithm for shape and size optimization of truss structures with frequency constraints. Comput Struct 195:99–112. https://doi.org/10.1016/j.compstruc.2017.06.016

29. Millán Páramo C, Begambre Carrillo O, Millán Romero E (2014) Proposal and validation of a modified Simulated annealing algorithm for solving optimization problems. Rev Int Métodos Numéricos para Cálculo y Diseño en Ing 30:264–270. https://doi.org/10.1016/j.rimni.2013.10.003
30. Millán Páramo C, Begambre Carrillo O (2016) Solving topology optimization problems using the modified simulated annealing algorithm. Rev int métodos numér cálc diseño ing 32:65–69. https://doi.org/10.1016/j.rimni.2014.11.005
31. Millan-Paramo C (2018) Modified simulated annealing algorithm for discrete sizing optimization of truss structure. Jordan J Civ Eng 12:683–697
32. Millan-Paramo C, Filho J (2019) Modified simulated annealing algorithm for optimal design of steel structures. Rev int métodos numér cálc diseño ing 35:1–12. https://doi.org/10.23967/j.rimni.2019.03.003
33. Liang JJ, Qu BY, Suganthan PN (2013) Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization. Comput Intell Lab Zhengzhou Univ Zhengzhou China Tech Report, Nanyang Technol Univ Singapore
34. Kirkpatrick S, Gelatt CD, Vecchi MP (1983) Optimization by simulated annealing. Science (80-) 220:671–680. https://doi.org/10.1126/science.220.4598.671
35. Zheng Y-J (2015) Water wave optimization: a new nature-inspired metaheuristic. Comput Oper Res 55:1–11. https://doi.org/10.1016/j.cor.2014.10.008

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.