AUT Journal of Civil Engineering

AUT J. Civil Eng., 6(2) (2022) 295-318
DOI: 10.22060/ajce.2023.21732.5810

PSOHHO Hybrid Optimization Algorithm for Truss Optimization

M. Yassami, P. Ashtari*

Department of Engineering, Faculty of Civil Engineering, University of Zanjan, Zanjan, Iran.

ABSTRACT: Numerous algorithms have recently been invented with varying strengths and weaknesses, none of which is the best for all cases. Herein, a hybrid optimization method known as a PSOHHO optimization algorithm is presented. There are two methods for combining algorithms: parallel and sequential. We adopted the parallel method and optimized the algorithm's performance. We cover the weaknesses of one algorithm with the strengths of another algorithm using a new method of combination. In this method, using several formulas, the top populations are exchanged between the two algorithms, and a new population is created. With this ability, the strengths of an algorithm can be used to compensate for the weaknesses of the other algorithm. In this method, no changes are made to the algorithms. The main goal is to use existing algorithms. This method aims to attain the optimal solution in the shortest time possible. Two algorithms of particle swarm optimization (PSO) and Harris Hawks optimization (HHO) were used to present this method and five truss samples were considered to confirm the performance of this method. Based on the results, this method has rapid convergence speed and acceptable results compared to the other methods. It also yields better results than its basic algorithms.

Review History:

Received: Aug, 29, 2022
Revised: Jan, 26, 2023
Accepted: Mar, 04, 2023
Available Online: Mar, 20, 2023

Keywords:

Meta-heuristic algorithms
Hybrid algorithm
Optimization
Truss
PSOHHO

1- Introduction

Optimization is a method to identify the best solution in the shortest time possible. Due to the recent expansion in parameters, mathematical models are not adequate anymore and, as such, the use of meta-heuristic models has grown. The solutions proposed by meta-heuristic algorithms are better and more precise solutions for engineering problems [1,2]. The main difference between meta-heuristic and commercial methods is that the former does not need a gradient to solve problems. They are also superior due to their easier application and their ability to search the entire space, which will lead to better solutions. The majority of the optimization algorithms are inspired by nature, animals' behaviors, or laws of physics. Generally, optimization algorithms can be divided into three main categories: physics-based, population-based, and evolutionary-based [3]. The genetic algorithm (GA) is the most well-known optimization algorithm [4,5] and has different stages including selection, crossover, and mutation. These phases prevent the algorithm from being trapped in local optima. In fact, randomness is a feature of this algorithm. GA is frequently applied due to its simplicity and the absence of numerous equations in it. Another population-based algorithm is the particle swarm optimization (PSO) algorithm, which is widely used today [6]. This algorithm was developed based on the social behavior of some animals and two parameters of speed and position, which are updated in each stage. This algorithm is based on the best previous experience of each population. The PSO has memory contrary to GA. In practice, this ability helps find the local optimal solution during the algorithm's stages. In each state, the best personal experience of the population is compared to the best overall experience and is replaced if it is superior. This also improves the convergence speed [7].

Many algorithms have been presented in recent years, including GA [5], PSO [8], Harris Hawk's optimization (HHO) [9], charged system search (CSS) [10], ant colony optimization (ACO) [11], grey wolf optimizer (GWO) [12], pathfinder algorithm (PFA) [13], poor and rich optimization algorithm (PRO) [14], Gaussian quantum-behaved particle swarm optimization (QPSO) [15], weighted differential evolution algorithm (WDE) [16], particle swarm optimization +(PSO+) [17], and optimization booster algorithm (OBA) [18].

Some papers have focused on optimizing two- and three-dimensional trusses, including cyclical parthenogenesis algorithm (CPA) [19], Jaya algorithm (JA) [20], improved GWO (IGWO) [21], mean-variance mapping optimization (MVMO) [22], accelerated multi-gravitational search algorithm (AMGSA) [23], firefly algorithm (FA) [24], chaotic coyote algorithm (COA) [25], spiral water cycle algorithm (SWCA) [26], artificial bee colony algorithm (ABC) [27], parameter-free Jaya algorithm (PFJA) [28],

*Corresponding author's email: ashtari@znu.ac.ir

Table 1. PSO hybridizations.

PSO hybridization with	References	Proposed	Conclusion
Grey wolf optimization (GWO)	[38]	Improving the exploration of PSO	Convergence to more optimal solutions with fewer iterations
Sine cosine algorithm (SCA)	[39]	Escaping the local optima	Successfully applied to real constrained engineering problems and provides better solutions than other methods
Whale optimization algorithm (WOA)	[40]	For the exploration phase in an uncertain environment	Very good convergence
Spotted hyena optimization (SHO)	[41]	Improving the hunting strategy of the spotted hyena optimizer	The algorithm performs better than other metaheuristic algorithms.
Gravitational search algorithm (GSA)	[42]	Improving the exploration of PSO	Superior performance in terms of accuracy, reliability, and efficiency compared to PSO, GSA, and other recently developed hybrid variants

interactive fuzzy search algorithm (IFSA) [29], mixed integer linear optimization (MILO) [30], hybrid artificial physics optimization and big bang-big crunch (HPBA) [30], and heuristic dragonfly algorithm (HAD) [31].

The hybridization of two or more algorithms, instead of developing and proposing new optimization methods, has also been popularized in recent years. The hybridization of algorithms helps compensate for the weaknesses of one algorithm with the strengths of another algorithm. Numerous studies are focused on this topic [32,33]. Yaseen [34] introduced a new hybrid optimization algorithm based on a bat algorithm (BA) and the PSO algorithm (HB-SA). Nenavath [35] used the hybrid sine–cosine algorithm with the teaching–learning-based optimization algorithm (SCA–TLBO). DEVARAPALLI adopted the sequential method and optimized the algorithm's performance. The velocity function update in each iteration of the PSO technique has been adopted to avoid being trapped in local search space with HHO [36].

1- 1- PSO algorithm

The PSO algorithm is based on the social behavior of fish and birds [6]. This algorithm searches the space based on the existing population that is also present in the GA. Therefore, PSO is a population-based algorithm. First, a large population is randomly created. Subsequently, the speed, fitness, and objective functions are determined for this population. A member of this population is selected as the best member. In the following stages, speed and position are calculated using Eqs. (1) and (2). In this stage, after calculating the fitness function, the best position is determined. If the best existing position is superior to the previous best position, the previous position will be replaced with this new position.

$$V_i(t) = W \times V_i(t-1) + C_1 \times rand_1\left(P_{Pbest} - X_i(t-1)\right) + C_2 \times rand_2 \times \left(P_{gbest} - X_i(t-1)\right) \quad (1)$$

$$X_i(t) = X_i(t-1) + V_i(t) \quad (2)$$

where $X_i(t)$ is the existing position, $V_i(t)$ is the existing speed, C_1 and C_2 are constants, P_{Pbest} is the best personal experience or the best local position, Pgbest is the best general position, W helps us approach optimization solutions as the optimization process proceeds, and $rand_1$ and $rand_2$ are constant numbers that will make the search of the space random, ensuring that the algorithm remains random; these random numbers vary from 0 to 1. Table 1 presents some algorithms hybridized with the PSO. Among the weaknesses of the PSO algorithm, we can mention exploration weakness that leads to a local optimum convergence. Different aspects of the PSO algorithm can be improved; initial population, inertia parameter w, and acceleration factors c1 and c2 are a few examples of these aspects. The PSO algorithm only accepts better solution positions but overlooks inferior solution positions with the potential to find the global optimal solution, which makes the algorithm easily fall into premature convergence [37].

1- 2- HHO algorithm

The HHO algorithm is inspired by the participatory behavior and chasing of the prey by Harris hawks in nature [9]. HHO is basically a gradient-free population-based

optimization algorithm. The central idea of HHO is based on the cooperative hunting behavior of Harris hawk bird, especially their chasing style of prey which is called surprise pounce or also known as the 'seven kill' strategy. In this smart process, several hawks collaborate to chase the prey from different directions to confuse it. Evidently, the prey's escape pattern is proportional to the hawks' pursuit model. Birds collaborate in the attack. The leader of the group attacks the target, follows it, and abruptly soars out of sight, and the next Harris Hawk joins the chase. These hawks can demonstrate a variety of chasing patterns based on the dynamic nature of scenarios and the prey's escape patterns. This mathematically mimics the dynamic behaviors to develop an optimization algorithm. The HHO algorithm has three phases. The first phase is the exploration ability, which is formulated as follows:

Exploration phase:

$$X(t+1)=\begin{cases} X_{rand}(t)-r_1\left|X_{rand}(t)-2r_2X(t)\right| & q\geq 0.5 \\ (X_{rabbit}(t)-X_m(t))-r_3\left(LB+r_4(UB-LB)\right) & q<0.5 \end{cases} \quad (3)$$

where *X(t)* is the current position, *X(t+1)* is the next position of the hawk, $X_{rabbit}(t)$ is the prey's position, q, r_1, r_2, r_3, r_4 are random numbers between 0 and 1, $X_{rand}(t)$ is the random position selected from among the available positions, $X_m(t)$ is the average of the existing positions, and LB and UB are the lower and upper bounds of the data, respectively. $X_m(t)$ is calculated using the equation below:

$$X_m(t)=\frac{1}{N}\sum_{i=1}^{N}X_i(t) \quad (4)$$

where $X_i(t)$ is the hawk's position in iteration t and N is the total number of hawks. After certain stages, the hawk approaches the target (prey). Then, convergence must increase. This is why the energy equations are used:

$$E=2E_0\left(1-\frac{t}{T}\right) \quad (5)$$

E_0 is the baseline energy, T is the maximum value of iteration, t is the current stage, and E is energy. In this phase, if is the search phase and , the exploitation phase will occur.

Exploitation Phase:

In the next stage, hawks surprise the prey and hunt it. Four cases may arise:

- Soft besiege: When r ≥ 0.5 and |E| ≥ 0.5

$$X(t+1)=\Delta X(t)-E\left|JX_{rabbit}(t)-X(t)\right| \quad (6)$$

$$\Delta X(t)=X_{rabbit}(t)-X(t) \quad (7)$$

$$J=2\left(1-r^5\right) \quad (8)$$

- Hard besiege: When r ≥ 0.5 and |E| < 0.5

$$X(t+1)=X_{rabbit}(t)-E\left|\Delta X(t)\right| \quad (9)$$

- Soft besiege with rapid dives: When |E| ≥ 0.5 still but r < 0.5

$$Y=X_{rabbit}(t)-E\left|JX_{rabbit}(t)-X(t)\right| \quad (10)$$

$$Z=Y+S\times LF(D) \quad (11)$$

Here, D is the dimension of the problem, and S is the random function with the size D x 1. LF is the Levy flight function obtained from Eq. (12).

$$LF(X)=0.01\times\frac{u\times\sigma}{|v|^{\frac{1}{\beta}}},\ \sigma=\left(\frac{\Gamma(1+\beta)\times\sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right)\times\beta\times2^{\left(\frac{\beta-1}{2}\right)}}\right)^{\frac{1}{\beta}} \quad (12)$$

where u and v are random values between 0 and 1. β is a default constant set to 1.5.

$$X(t+1)=\begin{cases} Y & if\ F(Y)<F(X(t)) \\ Z & if\ F(Z)<F(X(t)) \end{cases} \quad (13)$$

- Hard besiege with rapid dives: When |E| < 0.5 and r < 0.5

$$X(t+1)=\begin{cases} Y & if\ F(Y)<F(X(t)) \\ Z & if\ F(Z)<F(X(t)) \end{cases} \quad (14)$$

Here, Z and Y are calculated based on Eqs. (15) and (16).

$$Y=X_{rabbit}(t)-E\left|JX_{rabbit}(t)-X_m(t)\right| \quad (15)$$

$$Z=Y+S\times LF(D) \quad (16)$$

Traps in the local optima and weak exploration are also some of the weak points of the HHO algorithm, which some researchers have tried to fix. Dhawale in his paper, improved the exploration and exploitation phase of HHO using a chaotic variant of the present optimizer [43]. Table 2 briefly presents some algorithms hybridized using the HHO algorithm.

Table 2. HHO hybridizations.

HHO hybridization with	References	Proposed	Conclusion
Imperialist competitive algorithm (ICA)	[44]	Improving the exploration of HHO	Competitive performance compared to HHO, ICA, and some other well-established algorithms
Moth-flame optimization (MFO)	[45]	Improving the exploration of HHO	The HHO-MFO algorithm outperforms its competitors in a majority of case studies.
Simulate annealing (SA)	[46]	Escaping the local optima	Superior results compared to other algorithms
Grey wolf optimizer's (GWO)	[47]	Balancing exploration and exploitation	Better accuracy, the smaller size of selected features, in much lower computational time

Herein, we propose a novel hybrid method based on the two algorithms of PSO and HHO to optimize truss structures. This new method is called the PSOHHO hybrid algorithm, and all its stages will be presented in MATLAB. One of the main goals of this article is to provide a method to use the capabilities of the existing algorithms.

The remainder of this study is organized as follows: Section 2 shows the proposed optimization methods. The optimization examples and results are discussed in Section 3, where extensive comparisons with methods in the literature are presented. Finally, Section 4 summarizes the main findings of the study.

2- Proposed Optimization Methods

There are two methods for merging algorithms:

1) In the first method, two algorithms are used simultaneously. In the beginning, the population of each algorithm is half the initial population (e.g., if the hybrid algorithm has a total initial population of 100 members, the population of each algorithm is 50). When algorithms are run, the objective function values of the algorithms are compared at each stage, and the one that has a higher fitness value shares its population with the other. However, the population of each algorithm remains constant from the beginning to the end. The proposed algorithm leads to faster convergence and better responses.

2) In the second method, similar to the first one, the two algorithms are run simultaneously with half the initial population. However, by comparing the function values in the following steps, the algorithm that has a better function value will have more population in proportion. For instance, if the initial population includes 100 members, each algorithm has 50 members at the beginning; nevertheless, by comparing the fitness values in the next steps, the population of one algorithm may reach 60 members and the other algorithm will have 40 members.

In this paper, we employed the first method using a proposed relation. However, Eqs. (17) and (18) can also be used for other optimization methods. The proposed combination makes it possible to use existing algorithms without any alterations. Each algorithm has certain advantages and disadvantages, and it cannot be said with certainty which algorithm is the best. As such, when we use a combination of algorithms, the disadvantages of one method are covered by the other one's advantages. This combination is simple because it does not alter the original algorithms. The basic algorithms have been selected as an example, and this method can be used to combine other algorithms, which is a critical advantage of this method. Here, we used PSO and HHO for the presentation of our method. The reason for using PSO and HHO algorithms is because of the investigated impact of different computational complexity in the performance of each of these algorithms, which improves the performance of the presented method. PSO has a convincing exploitation ability but poor exploration ability (required for a good starting position), on the other hand, the HHO algorithm has a high exploration ability hence it gives a good starting point. Therefore, these two unique features of both algorithms are fused to obtain a hybrid PSO-HHO

If best fitness PSO < best fitness HHO:

$$X^{new}(\mathrm{i}) = \mathrm{X_m} + F_1 \mathrm{rand}_1 \left(\mathrm{X_{bPSO}} - \mathrm{X_m}\right) \tag{17}$$

If best fitness PSO ≥ best fitness HHO:

$$X^{new}(\mathrm{i}) = \mathrm{X_m} + F_2 \mathrm{rand}_2 \left(\mathrm{X_{bHHO}} - \mathrm{X_m}\right) \tag{18}$$

where X^{new} is the new location, X_m is the mean i^{th} best location of PSO and HHO, $rand_1$ and $rand_2$ are random numbers between 0 and 1, and F_1 and F_2 are the constants and equal to 1.5; X_{bPSO} is the i^{th} best location of PSO, and X_{bHHO} is the i^{th} best location of HHO.

Table 3. Algorithm pseudo-code of the PSOHHO algorithm.

Inputs: Population size (N) and maximum number of iterations (T)
Outputs: The location of the population and its fitness value
Create the random population X (i = 1, 2, . . . ,N) for every algorithm
While (the stopping condition is not met) do
Run PSO
Update the placement and velocity of the present agent by Eqs. (1) & (2)
Run HHO
Calculate the fitness values of the Hawks
Set X_{rabbit} as the placement of the rabbit (best location)
For (each hawk (X_i)) do
Update the first energy E0 and jump strength
Update the E using Eq. (5)
if (|E| ≥ 1) then
Update Eq. (3)
if (|E| < 1)
if (r ≥ 0.5 and |E| ≥ 0.5)
Update the placement vector using Eq. (6)
else if (r ≥ 0.5 and |E| < 0.5)
Update the location vector using Eq. (9)
else if (r < 0.5 and |E| ≥ 0.5
Update the location vector using Eq. (13)
else if (r < 0.5 and |E| < 0.5)
Update the placement vector using Eq. (14)
Return X_{rabbit}
If best fitness PSO < best fitness HHO
Update the placement vector using Eq. (17)
If best fitness PSO ≥ best fitness HHO
Update the placement vector using Eq. (18)
For n = 1: N mutation
d = 0.012*unifrnd (-1,1)*(UB(j)-LB(j))
Xn(i)+d
if $X^{new}(i)$ obtains better fitness function values Eq. (19)
Replace *Xi* with *Xnew*
Update X_b

In Eqs. (17) and (18), locations move toward a better point due to using the mean and best locations. This type of movement leads to more space searches, and the mean location moves toward more optimization. This formula aims to find the local optimum. However, it does not lose X_m from its memory.

In the end, a value is added or subtracted. This point avoids getting stuck in the local optimum, and it is obtained by the following equation:

$$d = 0.01 \times rand(-1,1) \times \left(\mathrm{Ub}_j - \mathrm{Lb}_j\right)_{j=1,\ldots n} \quad (19)$$

where Lb_j and Ub_j are the lower and upper bounds of the data, respectively, and rand (-1,1) indicates a generated random number between -1 and 1. Table 3 presents the algorithm pseudo-code of the PSOHHO algorithm. Fig. 1 shows the PSOHHO flowchart.

3- Numerical Examples

In the present study, the PSHHO hybridization method is adopted to optimize truss structures. To assess the method's efficiency, five truss structures, including 10-bar 2D, 25-bar 3D, 72-bar 3D, 200-bar 2D, and 942-bar 3D trusses, were examined. All these examples were from known cases that have recently been examined by numerous researchers. In the truss structure, the objective function and the constraints are obtained from Eq. (20).

Minimize weight:

$$\begin{aligned} & W(A) = \sum_{j=1}^{n} \gamma_j A_j L_j \\ & \sigma_j \le \sigma_{max} \\ & \Delta_k \le \Delta_{max} \\ & A_{min} \le A_j \le A_{max} \end{aligned} \quad (20)$$

where γ_j, A_j, and L_j indicate the specific weight, cross-section area, and length, respectively. Moreover, n denotes the number of truss cross-sections. The cross-sections in these problems are assumed to be variable and chosen between A_{min} and A_{max}. σ_j and Δk denote stress and displacement, respectively.

Since most metaheuristic methods were originally developed to solve unconstrained optimization tasks, it is

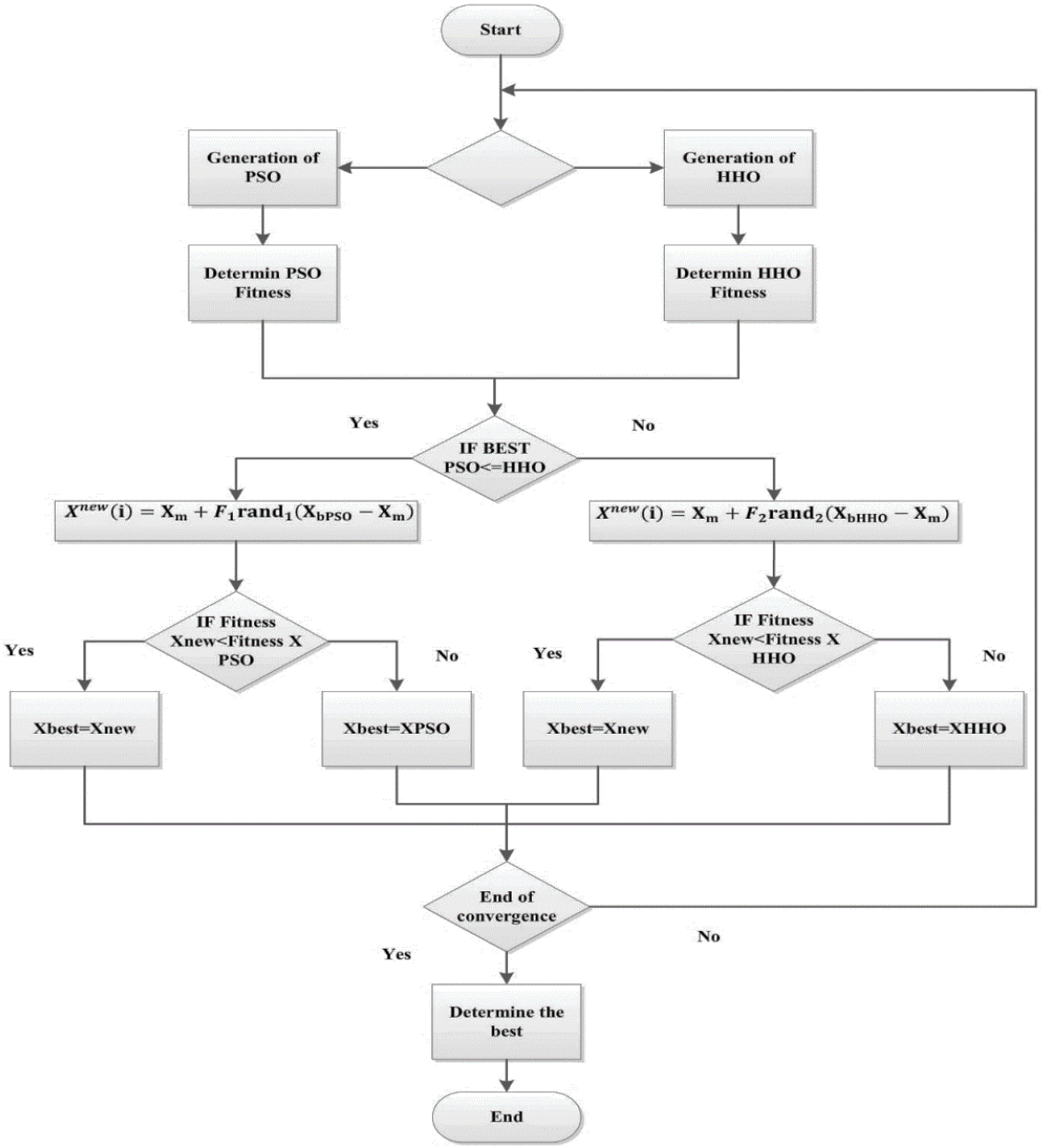

Fig. 1. The PSOHHO flowchart.

first necessary to convert the present design problem into an unconstrained one. In this study, a normalized constraint handling methodology is adopted where stress/ displacement values for every structural member and node are compared with the maximum allowable stress and displacement limits. The modified structural constraints are hence written as:

$$g_i^{\sigma}(x) = \frac{\sigma_i - \sigma_{max}}{\sigma_{max}} \le 0 \tag{21}$$

$$g_i^{\delta}(x) = \frac{\delta_i - \delta_{max}}{\delta_{max}} \le 0 \tag{22}$$

where $g_i^{\sigma}(x)$ is the normalized normal stress constraint for the ith member, σ_{max} is the maximum allowable normal stress limit for both tension and compression, $g_i^{\delta}(x)$ is the normalized displacement constraint for the ith node, δ_{max} is the maximum permissible nodal displacement value, nN is the number of nodes in the truss, and nE is the number of elements in the truss.

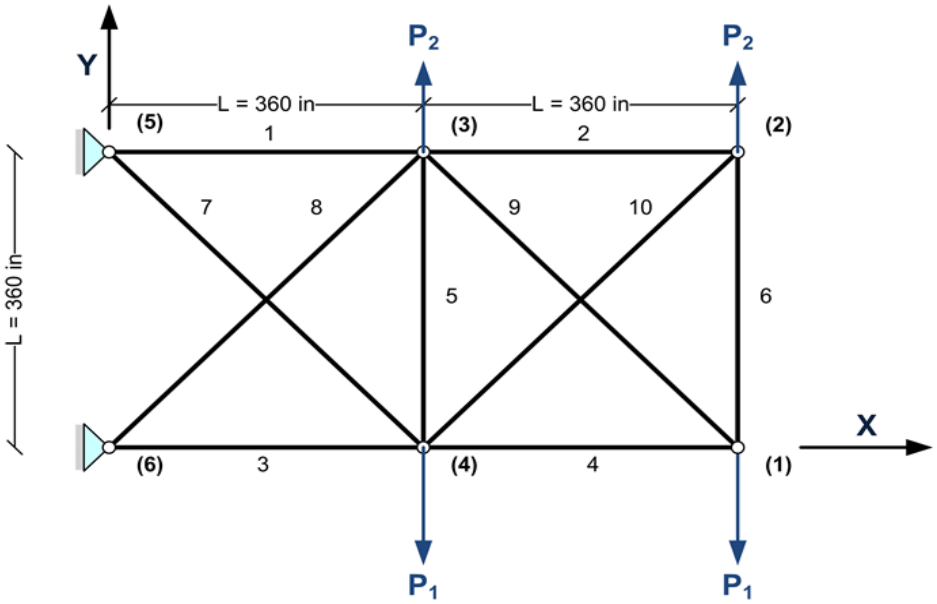

Fig. 2. The 10-bar truss problem.

Constraint values are governed by the following relation:

$$if\ g_i(x) > 0\ then\ c_i = g_i(x) \tag{23}$$

$$elseif\ g_i(x) \le 0\ then\ c_i = 0 \tag{24}$$

Stress and displacement constraints to be satisfied are handled by using a penalty function. The penalized objective function φ(x) is obtained as the product between the truss weight W(A) and the penalty function C as follows:

$$\varphi(x) = \left(\sum_{j=1}^{n} \gamma_j A_j L_j\right)(1+C)^{\varepsilon} \tag{25}$$

$$C = \sum_{i=1}^{nE}\left(\frac{\sigma_i - \sigma_{max}}{\sigma_{max}}\right) + \sum_{j=1}^{nN}\left(\frac{\Delta_i - \Delta_{max}}{\Delta_{max}}\right) \tag{26}$$

where the variables ε, E, and N in Eq. (26) are the penalty exponent, element, and node, respectively. The results were compared with those of other studies. The number of independent runs was chosen as 30, and the population size of 100 was selected. Finally, the best solution, the average of solutions, the standard deviation (Std), and the number of function evaluations (NFE) required for convergence were obtained.

3- 1- The 10-bar 2D truss

This case aims to minimize the 10-bar 2D truss depicted in Fig. 2 [48]. This example has 10 variables of (A1, ..., A10).

where W is the objective function, g is the constraint, σ denotes the stress, δ indicates the displacement, L is the length of the beam, σ_{allow} is the maximum allowable stress, δ max is the maximum allowable displacement, and A is the cross-section. In this example, the elasticity coefficient is 10^7 psi and the specific gravity is 0.1 lb/in^3. The stress limit is 25 ksi and the maximum horizontal and vertical displacement of the truss points is assumed to be 2 in. The cross-section limit for this case is 0.1 in^2 and 35 in^2. Two loadings are assumed for the 10-bar truss. In the first case, P_2 = 0 and P_1 = 100 kips, while in the second case, P_2 = 50 kips and P_1 = 150 kips. To determine the best final solution, in both loading states, 3000 analyses were performed. Figs. 3 and 4 depict the convergence trend for loading states 1 and 2.

Tables 4 and 5 list the results for loading states 1 and 2. For case 1 (Table 4), the best solution was achieved for the proposed PSOHHO algorithm (5060.80 lb), and the best rank was first. Evidently, the outcome of this method can compete with that of other methods, and this method has a faster convergence speed. In the second case (Table 5), 4677.26 lb was achieved as the best outcome. The best rank for the second state was second. The best outcome was obtained after 2417 analyses in the first state, while it was attained after 2386 analyses in the second state.

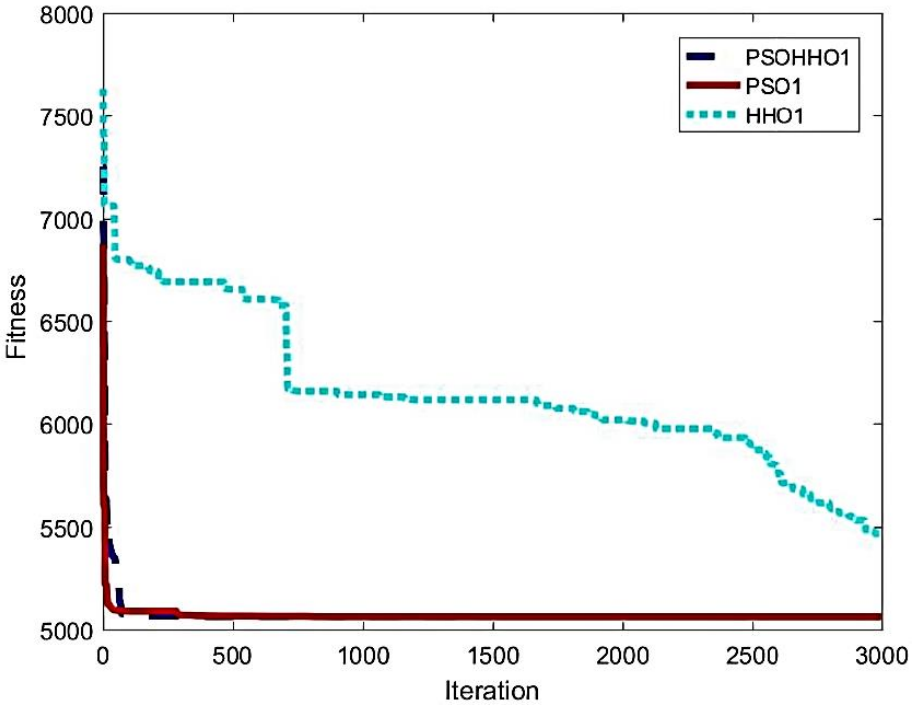

Fig. 3. The convergence curve of the 10-bar truss (Case 1).

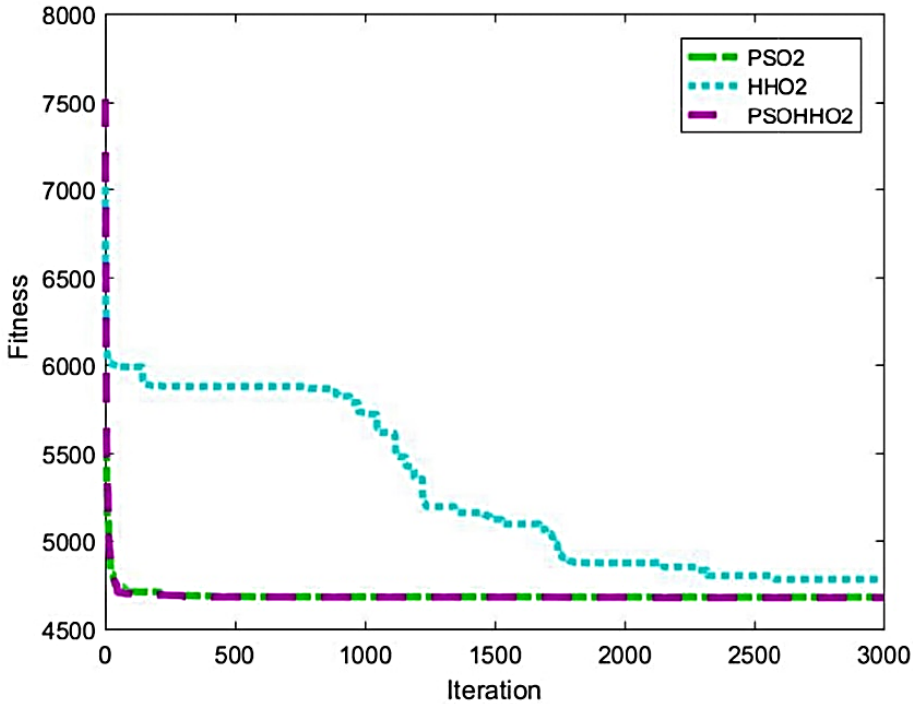

Fig. 4. The convergence curve of the 10-bar truss (Case 2).

Table 4. The results of the 10-bar truss problem (Case 1).

Variables	Optimal cross-sectional areas (in^2)							
	[53]	[52]	[51]	[50]	[49]	PSO	HHO	**PSOHHO**
A_1	30.5349	30.5383	30.5755	30.5069	30.501	30.6279	33.7109	30.3810
A_2	0.1	0.1	0.1	0.1	0.1	0.1	0.8885	0.1
A3	23.1893	23.1759	23.3368	23.302	23.198	22.9722	28.9113	23.2979
A4	15.2035	15.2483	15.1497	15.165	15.247	15.1772	11.5141	15.2853
A5	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
A6	0.5490	0.55377	0.5276	0.5436	0.5551	0.5386	0.8456	0.5616
A7	7.4613	7.45847	7.4458	7.4612	7.4562	7.5060	13.2335	7.4291
A8	21.0572	21.0269	20.9892	21.113	21.035	21.2239	17.6452	20.9283
A9	21.5170	21.5223	21.5236	21.413	21.526	21.4247	21.88	21.6454
A10	0.1	0.1	0.1	0.1	0.1	0.1	0.8601	0.1
Best weight(lb)	5060.85	5060.85	5060.99	5060.99	5060.85	5061.13	5464.75	5060.80
Average weight(lb)	5065.41	5060.87	5062.09	N/A	5061.23	5061.72	5464.76	5061.61
Std	5.2797	0.0215	2.05	N/A	0.53	1.41	0.1	0.011
NFE	14200	20000	19540	13800	7920	5800	15320	4834
Rank	2	2	3	3	2	4	5	1

Table 5. The results of the 10-bar truss problem (Case 2).

Variables	Optimal cross-sectional areas (in^2)							
	[49]	[50]	[51]	[53]	[54]	PSO	HHO	**PSOHHO**
A_1	23.62	23.5236	23.5804	23.523	23.6319	23.6954	22.4243	23.6251
A_2	0.1	0.1	0.1003	0.1	0.1	0.1	0.1	0.1
A3	25.434	25.2852	25.1582	25.285	25.3424	26.0611	21.881	25.2417
A4	14.351	14.3716	14.1801	14.371	14.5964	14.5368	11.8223	14.4404
A5	0.1	0.1	0.1002	0.1	0.1	0.1	0.1	0.1
A6	1.9701	1.9697	1.9708	1.969	1.9769	1.9696	2.2627	1.9695
A7	12.339	12.3917	12.4511	12.391	12.3446	12.0814	13.9816	12.3305
A8	12.712	12.8332	12.9349	12.833	12.6697	12.1956	16.4234	12.6087
A9	20.346	20.3288	20.3595	20.328	20.2586	20.5517	21.9958	20.5317
A10	0.1	0.1	0.1001	0.1	0.1	0.1	0.1	0.1
Best weight(lb)	4677.06	4676.92	4677.31	4776.92	4676.92	4680.06	4782.15	4676.96
Average weight(lb)	4677.97	4692.71	4679.06	4692.71	4680.30	4683.17	4795.23	4679.72
Std	0.33	45.7824	2.07	45.7824	3.82	1.95	10.04	0.044
NFE	7920	13640	19890	14000	5000	6100	16800	4772
Rank	3	1	4	6	1	5	7	2

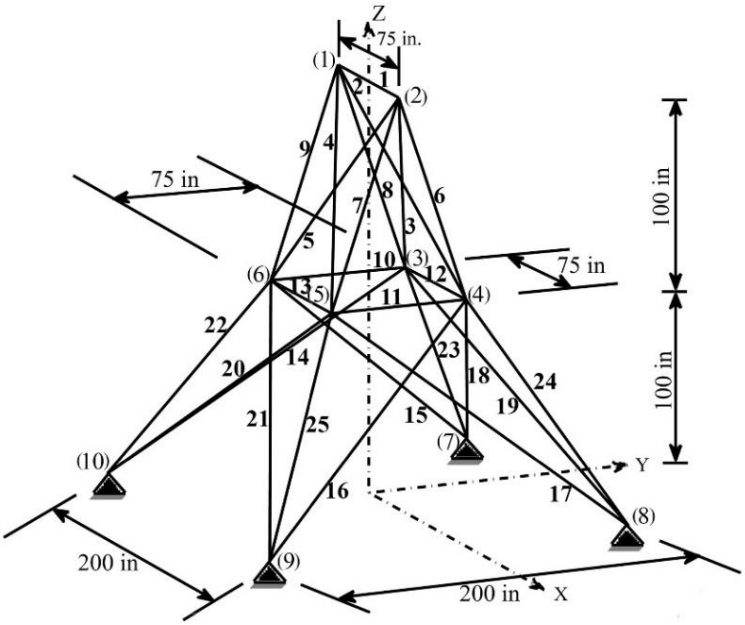

Fig. 5. The spatial 25-bar truss problem.

Table 6. Load cases for the spatial 25-bar truss.

node	Case 1			Case 2		
	F_x(kips)	F_y(kips)	F_z(kips)	F_x(kips)	F_y(kips)	F_z(kips)
1	0	20	-5	1	10	-5
2	0	-20	-5	0	10	-5
3	0	0	0	0.5	0	0
6	0	0	0	0.5	0	0

Table 7. Stresses for the 3D 25-bar truss.

Element Group		Compressive stress limitations (ksi)	Tensile stress limitations (ksi)
1	A_1	35.092	40.0
2	A_2-A_5	11.590	40.0
3	A_6-A_9	17.305	40.0
4	A_{10}-A_{11}	35.092	40.0
5	A_{12}-A_{13}	35.092	40.0
6	A_{14}-A_{17}	6.759	40.0
7	A_{18}-A_{21}	6.959	40.0
8	A_{22}-A_{25}	11.082	40.0

3- 2- The 25-bar 3D truss

Fig. 5 displays the 25-bar 3D truss. The specific gravity and elasticity coefficient are assumed to be 0.1 lb/in^3 and 10^7 psi, respectively. The maximum displacement is assumed to be 0.35 in. The 25 bars of this structure are divided into eight groups. The two types of loading for this truss are given in Table 6. The stress limit is presented in Table 7. The cross-section limit is assumed to be 0.01 to 3.4 in^2. The number of analyses for reaching the final solution is assumed to be 3000.

The final results for the 25-bar truss are presented in Table 8. For the proposed method, the value of 544.15 lb was attained after 2989 iterations, which indicates a high convergence speed. The final weight can compete with that presented in other sources. The rank of the proposed method was 1.

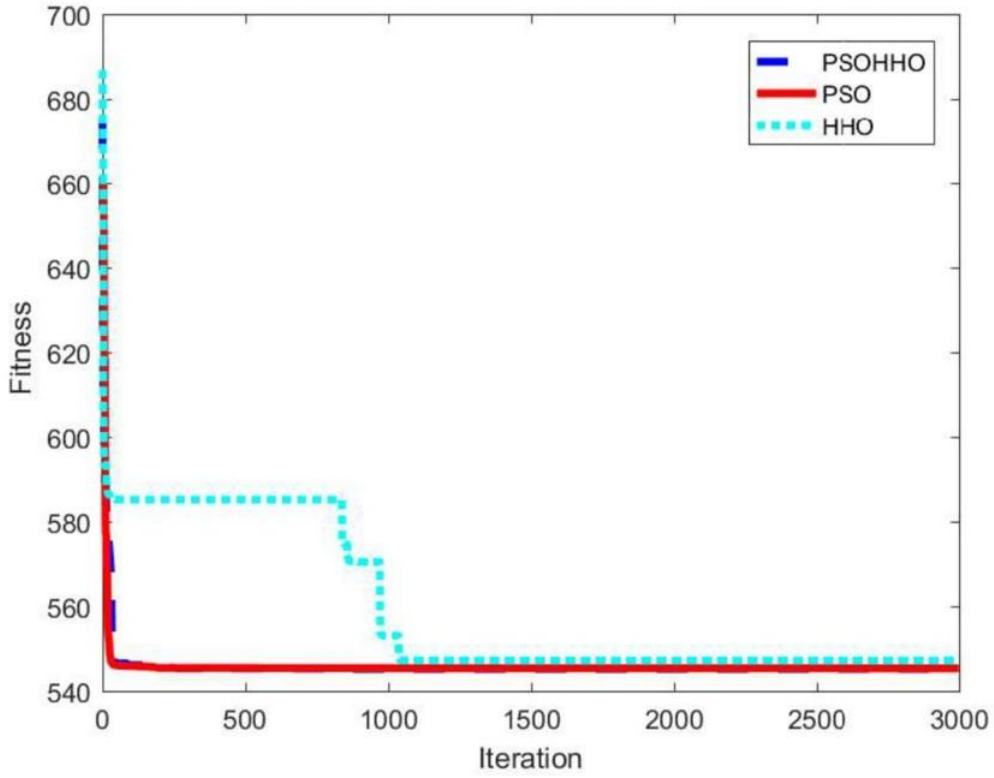

Fig. 6. The convergence curve of the 25-bar truss.

Table 8. The results of the 25-bar truss problem.

Variables	Optimal cross-sectional areas (in^2)							
	[21]	[51]	[52]	[53]	[55]	PSO	HHO	**PSOHHO**
A_1	0.0124	0.01	0.0100	0.0100	0.01	0.01	0.01	0.01
A_2-A_5	1.9624	1.9814	1.9825	1.9903	2.007	1.9177	2.1293	1.9664
A_6-A_9	3.0204	3.0023	3.0004	2.9881	3.001	3.0952	2.7317	3.0260
A_{10}-A_{11}	0.0266	0.0100	0.0100	0.1	0.01	0.01	0.0100	0.01
A_{12}-A_{13}	0.0109	0.0100	0.0100	0.1	0.01	0.01	0.0100	0.01
A_{14}-A_{17}	0.6841	0.6827	0.6832	0.6857	0.661	0.6875	0.7082	0.6782
A_{18}-A_{21}	1.6862	1.6778	1.6775	1.6764	1.620	1.6911	1.7095	1.6800
A_{22}-A_{25}	2.6526	2.6612	2.6610	2.6613	2.668	2.6272	2.6919	2.6600
Best weight(lb)	545.48	545.16	545.16	545.16	544.92	545.31	547.11	544.15
Average weight(lb)	549.67	545.22	545.16	545.91	545.13	545.36	548.15	545.23
Std	2.8113	0.083	0.00162	1.0893	0.401	0.123	1.024	0.031
NFE	5640	19750	20000	6500	N/A	7340	11500	5978
Rank	5	3	3	3	2	4	6	1

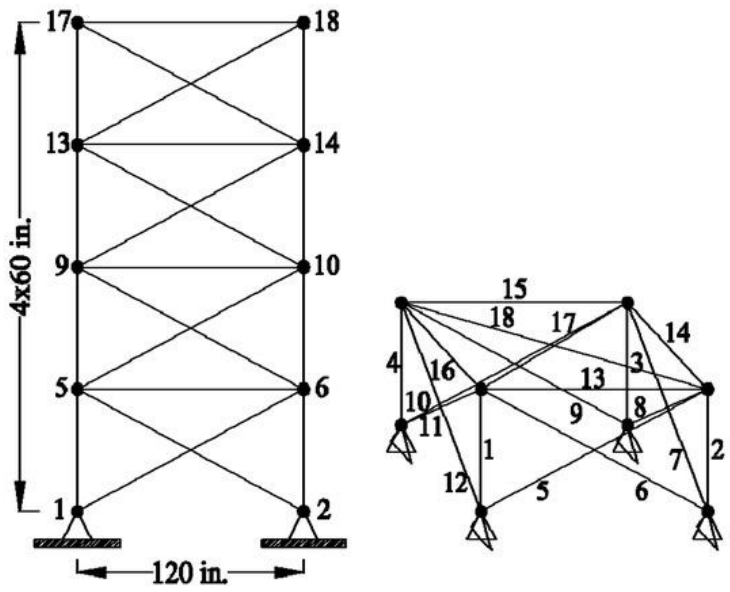

Fig. 7. The spatial 72-bar truss problem.

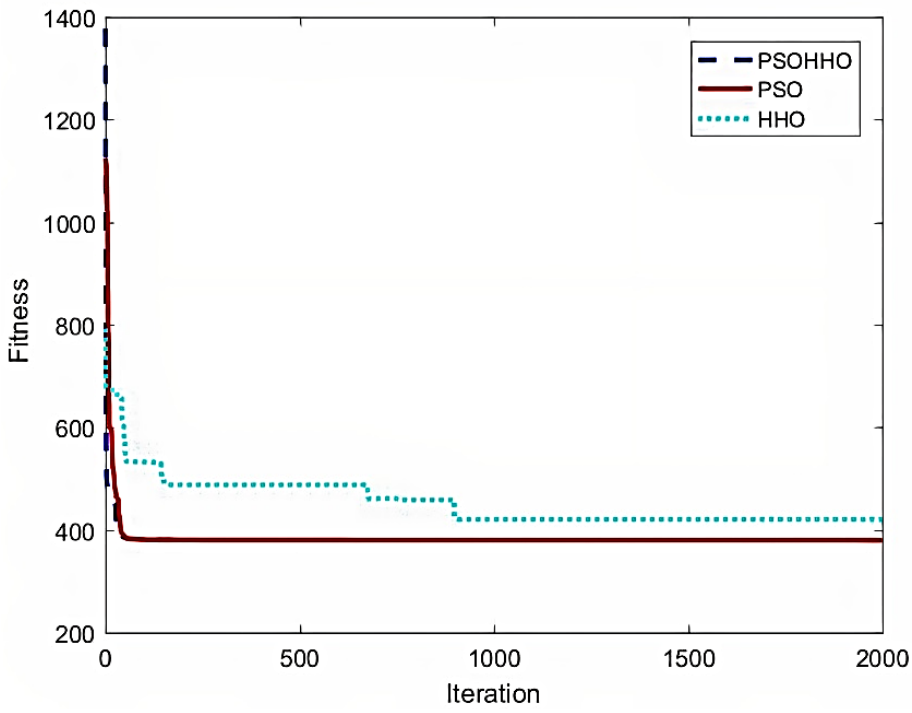

Fig. 8. The convergence curve of the 72-bar truss.

3- 3- The 72-bar 3D truss

This truss is displayed in Fig. 7. It contains 72 bars classified into 16 groups. The allowable stress for this case is 25 ksi at the assumed pressure and traction. The elasticity coefficient is assumed to be 10^7 psi, and the specific gravity in cross-sections is considered to be 0.1 lb/in^3. The maximum displacement of the point is 0.25 in. In the optimization process, the minimum cross-section area of 0.1 in^2 is applied. The magnitude of the loads applied and their classification are given in Table 9. Table 10 shows the member group of the spatial 72-bar truss. The results of this case after 2000 analyses are presented in Table 11. Fig. 8 depicts the process of optimization.

According to Table 11, the value of 379.54 lb was obtained by the PSOHHO algorithm after 1931 analyses, indicating the rapid convergence of this method. The best rank for the 72-bar truss was two. Fig. 8 shows that the proposed method has faster convergence than the HHO and PSO algorithms.

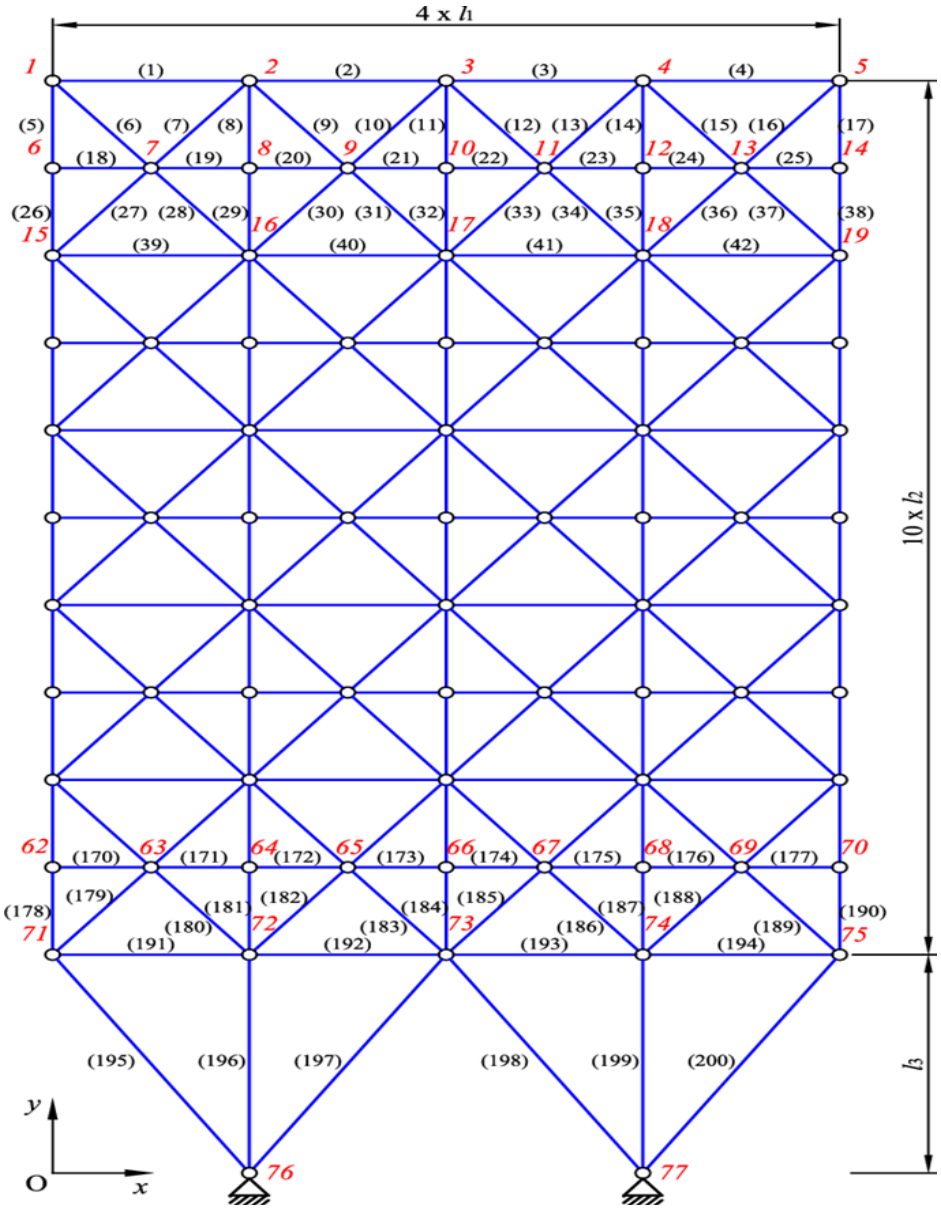

Fig. 9. The 200-bar truss problem.

3- 4- The 200-bar 2D truss

The 200-bar 2D truss is a well-known truss in optimization (Fig. 9). It comprises 200 bars classified into 2 groups (Table 13). There are three states for loading (Table 12). The elasticity coefficient of 10^7 psi and the specific gravity of 0.273 lb/in^3 are assumed for the bars. The allowable stress at the pressure and traction is 10 ksi, and there is no constraint on displacement. The minimum cross-section area is 0.1 in^2. The solutions of this case after 4000 analyses are presented in Table 14.

The final solution obtained by the PSOHHO method is 25076.43 lb after 3884 analyses, demonstrating its rapid convergence. A comparison of the convergence speed of the proposed method and other algorithms is presented in Fig. 10. In this example, the best rank for PSOHHO was one.

Table 9. Load cases for the spatial 72-bar truss.

node	Case 1			Case 2		
	F_x(kips)	F_y(kips)	F_z(kips)	F_x(kips)	F_y(kips)	F_z(kips)
17	5	5	-5	0	0	-5
18	0	0	0	0	0	-5
19	0	0	0	0	0	-5
20	0	0	0	0	0	-5

Table 10. Member group of the spatial 72-bar truss.

group	element	group	element
1	A_1-A_4	9	A_{37}-A_{40}
2	A_5-A_{12}	10	A_{41}-A_{48}
3	A_{13}-A_{16}	11	A_{49}-A_{52}
4	A_{17}-A_{18}	12	A_{53}-A_{54}
5	A_{19}-A_{22}	13	A_{55}-A_{58}
6	A_{23}-A_{30}	14	A_{59}-A_{66}
7	A_{31}-A_{34}	15	A_{67}-A_{70}
8	A_{35}-A_{36}	16	A_{71}-A_{72}

Table 11. The results of the 72-bar truss problem.

Variables	Optimal cross-sectional areas (in^2)							
	[21]	[51]	[52]	[53]	[56]	PSO	HHO	**PSOHHO**
A_1	1.8585	1.8618	1.88468	1.929788435	1.8910	1.9119	1.3536	1.9066
A_2	0.5021	0.5206	0.51372	0.508996112	0.5131	0.5101	0.5107	0.5140
A_3	0.1002	0.0105	0.1	0.1	0.1	0.1	0.1534	0.1
A_4	0.1	0.0100	0.1	0.1	0.1	0.1	0.1798	0.1
A_5	1.3011	1.2455	1.27107	1.246709769	1.2697	1.2514	1.9478	1.2490
A_6	0.5151	0.5177	0.51080	0.512783923	0.5097	0.5154	0.5925	0.5131
A_7	0.1	0.0101	0.1	0.1	0.1	0.1	0.1	0.1
A_8	0.1001	0.0100	0.1	0.1	0.1	0.1	0.1805	0.1
A_9	0.5311	0.5327	0.52589	0.52977944	0.5201	0.5191	0.4296	0.5330
A_{10}	0.5122	0.5109	0.51627	0.517240625	0.5175	0.5176	0.5840	0.5169
A_{11}	0.1008	0.0100	0.1	0.1	0.1	0.1	0.1	0.1
A_{12}	0.1030	0.1205	0.1	0.100000001	0.1	0.1	0.6109	0.1
A_{13}	0.1560	0.1655	0.15647	0.156445307	0.1566	0.1565	0.1440	0.1565
A_{14}	0.5472	0.5397	0.54479	0.543968566	0.5457	0.5449	0.6557	0.5417
A_{15}	0.4202	0.4554	0.41210	0.41055331	0.4107	0.4033	0.3395	0.4086
A_{16}	0.5793	0.5995	0.56840	0.562437566	0.5679	0.5727	0.2729	0.5678
Best weight(lb)	379.76	363.98	379.61	379.65	379.56	379.64	420.00	379.54
Average weight(lb)	380.68	364.35	379.62	380.29	379.67	379.79	450.02	379.73
Std	0.7315	0.2188	0.0038	0.5243	0.127	0.0032	33.831	0.0021
NFE	11960	19860	33600	12000	9000	4660	9580	3862
Rank	7	1	4	6	3	5	8	2

Table 12. Load cases for the planner 200-bar truss.

node	Case 1	Case 2	Case 3
Load(lb)	1000	10000	
Direction	X	Y	
Nodes	1,6,15,20,29,34,43, 48,57,62,71	1-6,8,10,12,14-20,22,24,26, 28-34, 36,38,40, 42-48,50, 52,54,56- 62, 64,66,68,70-75	Load cases 1 and 2 acting simultaneously

Table 13. Member grouping details for the planar 200-bar truss.

Group	Member Number	Group	Member Number
A1	1, 2, 3,4	A16	82, 83, 85, 86, 88, 89, 91, 92, 103,104, 106, 107, 109, 110, 112, 113
A2	5, 8, 11, 14, 17	A17	115, 116, 117, 118
A3	19, 20, 21, 22, 23, 24	A18	119, 122, 125, 128, 131
A4	25, 56, 63, 94, 101, 132, 139, 170, 177	A19	133, 134, 135, 136, 137, 138
A5	26, 29, 32, 35, 38	A20	140, 143, 146, 149, 152
A6	6, 7, 9, 10, 12, 13, 15, 16, 27, 28, 30, 31, 33, 34, 36, 37	A21	120, 121, 123, 124, 126, 127, 129, 130, 141, 142, 144, 145, 147, 148, 150, 151
A7	39, 40, 41, 42	A22	153, 154, 155, 156
A8	43, 46, 49, 52, 55	A23	157, 160, 163, 166, 169
A9	57, 58, 59, 60, 61, 62	A24	171, 172, 173, 174, 175, 176
A10	64, 67, 70, 73, 76	A25	178, 181, 184, 187, 190
A11	44, 45, 47, 48, 50, 51, 53, 54, 65, 66, 68, 69, 71, 72, 74, 75	A26	158, 159, 161, 162, 164, 165, 167, 168, 179, 180, 182, 183, 185, 186, 188, 189
A12	77, 78, 79, 80	A27	191, 192, 193, 194
A13	81, 84, 87, 90, 93	A28	195, 197, 198, 200
A14	95, 96, 97, 98, 99, 100	A29	196,199
A15	102, 105, 108, 111, 114		

Table 14. The results of the 200-bar truss problem.

Variables	Optimal cross-sectional areas (in^2)							
	[20]	[21]	[51]	[52]	[53]	PSO	HHO	**PSOHHO**
A_1	0.147258	0.1024	0.1144	0.144758	0.1471	0.1270	11.5191	0.1691
A_2	0.940434	0.9654	0.9443	0.943058	0.9399	0.9777	1.4260	1.0384
A_3	0.100109	0.1391	0.1310	0.101225	0.1000	0.1000	0.9944	0.2174
A_4	0.100098	0.1741	0.1016	0.100001	0.1	0.1000	4.1363	0.1
A_5	1.941704	1.9613	2.0353	1.943059	1.9399	1.9897	3.2044	2.0074
A_6	0.296783	0.2899	0.3126	0.296271	0.2965	0.1204	5.9409	0.1998
A_7	0.100096	0.1294	0.1679	0.103267	0.1000	0.4548	0.9317	0.1241
A_8	3.106749	3.1511	3.1541	3.114355	3.1049	3.8642	3.6499	3.2032
A_9	0.100095	0.1251	0.1003	0.102462	0.1000	0.1000	1.9094	0.1094
A_{10}	4.108109	4.0627	4.1005	4.114354	4.1049	4.2385	3.9358	4.4946
A_{11}	0.403975	0.4131	0.4350	0.400374	0.4037	0.3514	0.8594	0.2454
A_{12}	0.193079	0.4043	0.1148	0.113995	0.1906	0.1000	1.6149	0.1536
A_{13}	5.434236	5.3357	5.3823	5.388609	5.4298	6.8563	13.3499	5.5229
A_{14}	0.100095	0.2632	0.1607	0.100012	0.1006	1.1606	2.8119	0.1106
A_{15}	6.434203	6.3226	6.4152	6.388601	6.4298	6.6874	6.3849	6.3991
A_{16}	0.575306	0.7972	0.5629	0.533194	0.5739	0.5956	1.6907	0.3312
A_{17}	0.135485	0.1791	0.4010	0.394526	0.1332	0.1000	1.4669	0.6629
A_{18}	7.980200	8.1268	7.9735	7.941942	7.9744	7.4970	5.3782	7.4228
A_{19}	0.100157	0.1141	0.1092	0.100949	0.1000	1.0235	2.2753	0.3712
A_{20}	8.980345	9.1337	9.0155	8.941920	8.9744	8.4960	6.2781	8.4226
A_{21}	0.709002	0.8000	0.8628	0.834785	0.7064	0.9117	4.2669	0.8138
A_{22}	0.437247	0.2487	0.2220	0.151136	0.4339	0.1000	9.4266	1.0067
A_{23}	10.89123	11.2008	11.025	10.94004	10.8790	10.4553	10.1191	10.3599
A_{24}	0.100150	0.1136	0.1397	0.100028	0.1	0.1000	11.0087	0.1
A_{25}	11.89141	12.1703	12.034	11.94004	11.8790	11.4526	14.5686	11.3601
A_{26}	1.049144	0.9947	1.0043	0.897270	1.0453	0.7150	3.4915	1.1006
A_{27}	6.610648	6.3377	6.5762	6.848813	6.6300	7.3215	4.0494	8.5567
A_{28}	10.77913	10.5338	10.726	10.88481	10.7827	15.00	14.8294	11.3949
A_{29}	13.87830	14.0917	13.966	13.74952	13.8691	15.00	14.8329	13.4384
Best weight(lb)	25463.53	25771.77	25374.8	25453.77	25448.88	27200.46	49343.06	25076.43
Average weight(lb)	25477.47	26699.19	26613.4	25455.67	25531.69	27351.22	49343.11	26258.22
Std	24.12	410.401	702.8	2.337	42.1634	180.21	0.12	18.68
NFE	31580	23760	19410	80500	96600	14250	17980	7768
Rank	5	6	2	4	3	7	8	1

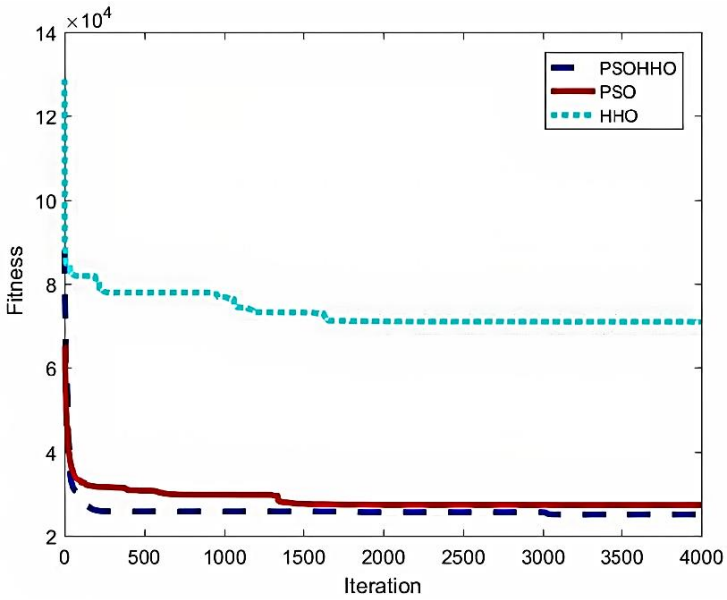

Fig. 10. The convergence curve of the 200-bar truss.

Table 15. Load cases for the spatial 942-bar truss.

Case number	Load (lb)	Direction	Nodes
1	-3000	Z	Each node in the first section
2	-6000	Z	At each node in the second section
3	-9000	Z	At each node in the third section
4	1000	X	At each node on the right side
5	1500	X	At each node on the left side
6	1000	Y	All nodes of the tower

3- 5- The 942-bar 3D truss

The final case is a 942-bar 3D truss. This structure has 26 stories and 942 bars classified into 59 groups (Fig. 11). In this case, like the previous structures, the elasticity coefficient of 10^7 psi and the specific gravity of 0.1 lb/in^3 are assumed. Six types of loads are considered for loading (Table 15). The cross-section limits are in the 1-200 in^2 range. The stress constraint for both states of pressure and traction is 25 ksi. The maximum displacement for the upper points is 15 in. Solutions of this case after 13000 analyses are presented in Table 16.

According to Table 16, the solution of the PSOHHO method is 131044.45 lb, which is acceptable compared to the other methods. In this example, the best rank for PSOHHO was one. The convergence speed of the proposed method is presented in Fig. 12.

Table 16. The results of the 942-bar truss problem.

Variables	Optimal cross-sectional areas (in^2)							
	[20]	[21]	[53]	[56]	[58]	PSO [57]	HHO	**PSOHHO**
A_1	1.045258	4.2489	1.000106287	1	1.00	1.00	1.024	1.002
A_2	1.001630	1.7702	1.000000908	1	1.00	1.00	1.000	1.002
A_3	3.549999	1.5892	3.166316514	1	3.01	6.999	3.3541	3.321
A_4	1.924590	1.5235	1.815748019	1	1.75	3.00	1.9647	1.824
A_5	1.000032	1.0265	1.000004059	1	1.00	2.00	1.0057	1.0387
A_6	15.337079	15.3979	14.47644046	14	14.27	17	14.9675	14.586
A_7	3.108905	2.8825	3.00668249	4	2.93	3	3.4672	3.1351
A_8	6.589077	6.9912	7.035993176	5	1.00	20	7.0534	7.986
A_9	16.569661	11.2039	16.37606449	5	1.00	38	17.8646	16.148
A_{10}	2.553777	2.7262	2.379222386	22	9.38	9	2.4682	2.459
A_{11}	6.433946	8.1921	6.438189777	1	4.43	2	6.3947	6.458
A_{12}	5.812166	6.2178	5.602718173	4	4.54	8	5.6487	5.726
A_{13}	15.836882	16.5585	15.1181383	19	16.14	23	15.9647	15.1347
A_{14}	2.196943	2.3668	2.124871462	2	2.33	3	2.3245	2.214
A_{15}	4.324553	4.1519	4.098857115	4	7.51	26	4.3651	4.236
A_{16}	1.000047	1.2370	1.000000108	1	1.00	1	1.0000	1.254
A_{17}	21.973772	22.3006	21.68022386	21	22.47	50.98	21.5672	21.125
A_{18}	2.674909	2.9996	2.597362396	3	2.70	3	2.7698	2.432
A_{19}	8.722646	7.7559	7.870074834	14	13.58	26	7.9621	7.532
A_{20}	1.000032	1.1283	1.000000101	1	1.00	1	1.0000	1.02
A_{21}	29.898613	28.2646	27.75329707	35	28.93	20	28.3547	27.232
A_{22}	3.249223	3.1924	3.135263149	3	3.23	2	3.8632	3.125
A_{23}	16.995624	16.3965	15.84340544	18	23.87	20	16.3254	15.631
A_{24}	25.510407	22.6095	26.31743388	24	41.67	30	26.8134	26.352
A_{25}	37.634066	40.0759	40.69496614	36	36.02	66.98	39.6340	41.332
A_{26}	1.220731	5.3549	1.149987375	1	6.41	22	1.4302	1.624
A_{27}	11.944077	9.2695	11.64990143	11	23.79	2	12.0364	11.035
A_{28}	16.515003	15.0911	16.07570037	14	28.39	10	16.8032	16.927
A_{29}	14.822892	14.0704	13.90458804	14	19.38	15	14.6325	13.024
A_{30}	15.983565	15.1962	14.52601036	23	20.31	14	14.9364	14.726
A_{31}	38.514252	37.1490	35.38504225	38	31.41	31	35.6327	35.932

A_{32}	3.323571	3.1643	3.207432961	3	2.57	4	3.7340	3.021
A_{33}	3.189674	3.4414	2.532916182	2	4.18	5	3.5237	2.825
A_{34}	2.822370	2.2813	2.506554054	3	3.33	2	2.7032	2.453
A_{35}	1.001323	1.0166	1.000000003	1	1.00	1	1.0000	1
A_{36}	1.002606	1.4089	1.00000006	1	1.00	1	1.0000	1
A_{37}	59.530117	59.6649	57.28451296	70	47.11	45.98	58.6378	56.276
A_{38}	3.250054	3.3173	3.163254209	3	2.35	4	3.6210	4.054
A_{39}	2.068093	2.0249	2.146376942	2	3.79	12	2.7302	2.635
A_{40}	3.084539	2.3953	2.850268391	3	3.30	4	2.8371	2.721
A_{41}	1.000717	1.0554	1.000000264	1	1.00	1	1.0005	1
A_{42}	1.239938	1.2294	1.000616023	1	1.00	6	1.0273	1
A_{43}	79.891179	79.5798	76.72647796	91	63.33	54	79.3619	75.586
A_{44}	3.299488	3.2875	3.154725029	3	3.21	3	3.5067	4.324
A_{45}	1.964128	1.9028	2.064116875	2	4.86	8	2.0396	2.254
A_{46}	3.489718	3.2460	3.279900541	2	2.22	8	3.4631	2.864
A_{47}	1.000032	1.0277	1.000079442	1	1.00	2	1.0024	1.001
A_{48}	1.000032	1.0898	1.000000176	1	1.00	3	1.0000	1.035
A_{49}	97.181471	93.8836	91.16688452	102	76.93	56	91.9642	92.132
A_{50}	3.322281	3.0634	3.23458801	4	3.54	3	3.5487	3.665
A_{51}	1.002997	1.7246	1.000000003	1	3.91	8	1.0003	2.924
A_{52}	3.651629	3.9313	3.600181305	3	2.25	4	3.8964	3.756
A_{53}	7.226228	8.1063	6.583921302	10	11.44	31	6.9547	6.246
A_{54}	4.544599	9.8391	3.785500472	11	11.64	24	3.5314	3.486
A_{55}	41.411074	42.7529	41.72633586	46	36.94	87.97	45.3984	41.336
A_{56}	1.002207	1.1219	1.000000003	1	1.00	6	1.2358	1.001
A_{57}	64.803517	63.0179	63.417437	65	48.10	36	67.5319	62.468
A_{58}	2.525618	2.6542	2.3264112	3	5.88	12	2.9365	2.247
A_{59}	1.000054	1.6685	1.0000000	1	1.00	4	1.0024	1.022
Best weight(lb)	137,344.35	136,311.1322	131,984.40	141,860	134,120	157,984.29	138,862.34	131,044.45
Average weight(lb)	137379.616	137453.6697	135,768.12	144,231	135244.7	177196.69	140,531.24	135328.61
Std	38.346	673.8566	2289.1491	3342	1497.06	13073.63	2863.42	1832.27
NFE	58274	28000	140800	32500	75000	41174	10458	14608
Rank	5	4	2	7	3	8	6	1

4- Conclusion

Here, truss structures were optimized using a novel PSOHHO hybrid method. This algorithm uses a distinct combination of PSO and HHO. The proposed algorithm consistently outperforms the standard PSO and HHO. In this method, the two algorithms work in parallel, but the population is exchanged using the proposed relation. This relation can be applied to other similar algorithms. The main purpose of this proposed relation is to use the capabilities of existing algorithms without modifying the principle of the algorithm. Both these algorithms have certain strengths and drawbacks; with this method, one method's strengths compensate for the other's weaknesses. It can be concluded that the hybrid algorithm PSOHHO yielded the best results among the examined algorithms. Five well-known trusses were assumed to evaluate the performance and functioning of the proposed algorithm. Based on the findings, the proposed hybrid method had rapid convergence and the final solutions were more acceptable than those of the other algorithms. In most examples, this method's rank was satisfactory.

It is recommended that other algorithms be combined using the proposed method. Various structural examples can also be utilized to test the performance of the proposed algorithm.

Declaration of Competing Interest

The authors declare that they have no conflict of interest.

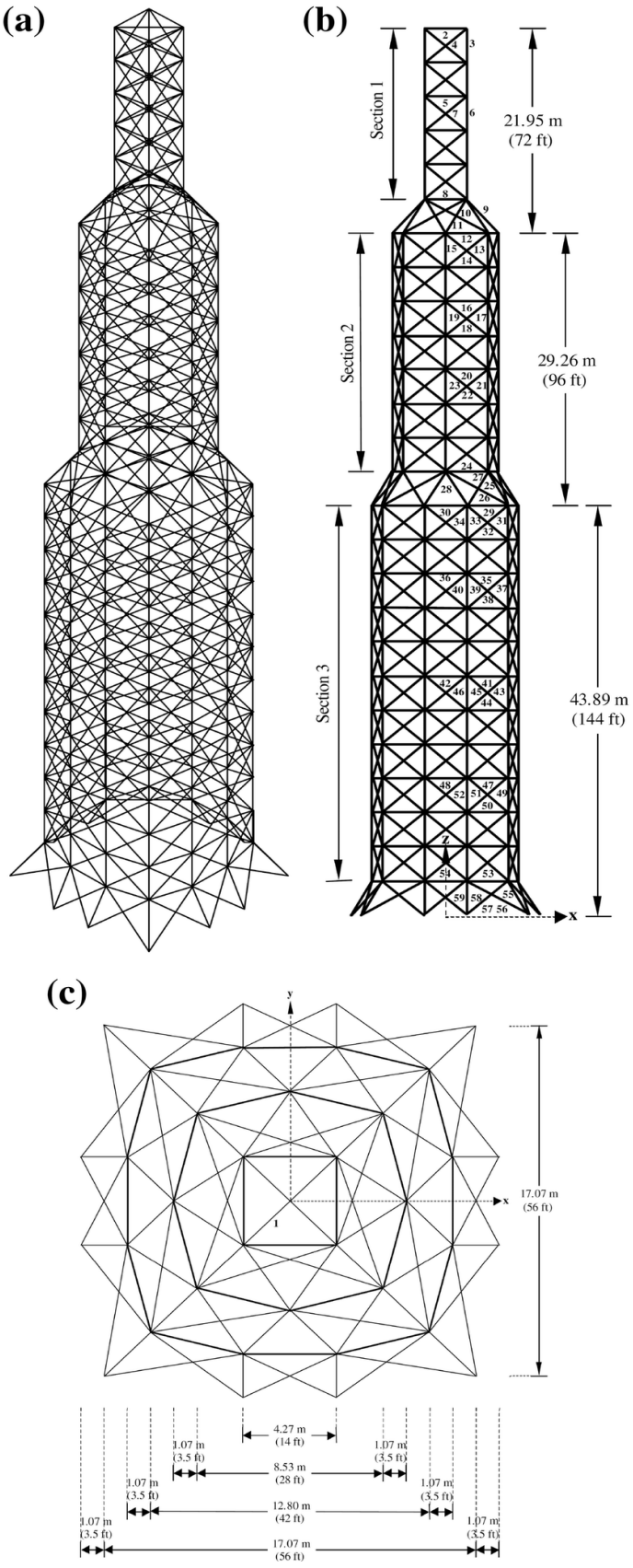

Fig. 11. The spatial 942-bar truss problem.

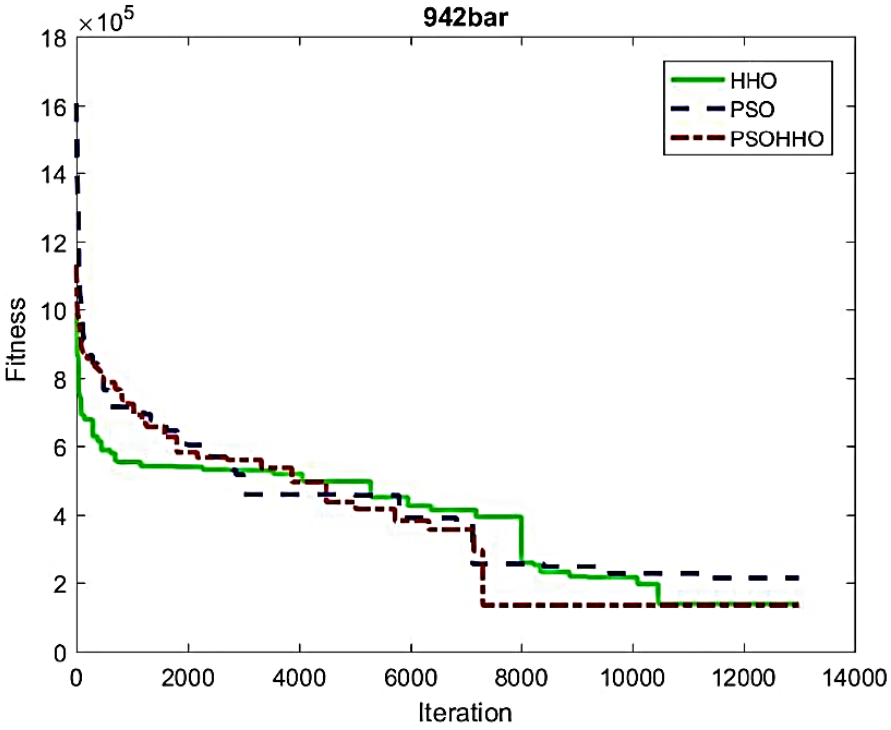

Fig. 12. The convergence curve of the 942-bar truss.

References

[1] D.B. Fogel, Evolutionary Computation: Toward a New Philosophy of Machine Intelligence, 3rd Editio ed., John Wiley & Sons, Inc., Hoboken, NJ, USA, 2005.

[2] X. Yao, Evolutionary Computation: Theory and Applications, in: Evolutionary Computation: Theory and Applications, WORLD SCIENTIFIC, 1999, pp. 1-36.

[3] S. Khalilpourazari, S. Khalilpourazary, A lexicographic weighted Tchebycheff approach for multi-constrained multi-objective optimization of the surface grinding process, Engineering Optimization, 49 (2017) 878-895.

[4] J.H. Holland, Genetic Algorithms, Scientific American, 267 (1992) 66-73.

[5] J.H. Holland, J.S. Reitman, Cognitive systems based on adaptive algorithms, ACM SIGART Bulletin, (1977) 49-49.

[6] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proceedings of ICNN'95 - International Conference on Neural Networks, IEEE, 1995, pp. 1942-1948.

[7] J. Kennedy, R.C. Eberhart, Swarm Intelligence, 1 st ed., Morgan Kaufmann, 2002.

[8] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science, IEEE, 1995, pp. 39-43.

[9] A.A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, Harris Hawks optimization: Algorithm and applications, Future Generation Computer Systems, 97 (2019) 849-872.

[10] A. Kaveh, S. Talatahari, A novel heuristic optimization method: charged system search, Acta Mechanica, 213 (2010) 267-289.

[11] D. Karaboga, B. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, Journal of Global Optimization, 39 (2007) 459-471.

[12] S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey Wolf Optimizer, Advances in Engineering Software, 69 (2014) 46-61.

[13] H. Yapici, N. Cetinkaya, A new meta-heuristic optimizer: Pathfinder algorithm, Applied Soft Computing, 78 (2019) 545-568.

[14] S.H. Samareh Moosavi, V.K. Bardsiri, Poor and rich optimization algorithm: A new human-based and multi populations algorithm, Engineering Applications of Artificial Intelligence, 86 (2019) 165-181.

[15] L.d.S. Coelho, Gaussian quantum-behaved particle swarm optimization approaches for constrained engineering design problems, Expert Systems with Applications, 37 (2010) 1676-1683.

[16] P. Civicioglu, E. Besdok, M.A. Gunen, U.H. Atasever, Weighted differential evolution algorithm for numerical function optimization: a comparative study with cuckoo search, artificial bee colony, adaptive differential evolution, and backtracking search optimization

algorithms, Neural Computing and Applications, 32 (2020) 3923-3937.

[17] M. Kohler, M.M.B.R. Vellasco, R. Tanscheit, PSO+: A new particle swarm optimization algorithm for constrained problems, Applied Soft Computing, 85 (2019) 105865.

[18] S.H. Pakzad-Moghaddam, H. Mina, P. Mostafazadeh, A novel optimization booster algorithm, Computers & Industrial Engineering, 136 (2019) 591-613.

[19] A. Kaveh, A. Zolghadr, Cyclical parthenogenesis algorithm for layout optimization of truss structures with frequency constraints, Engineering Optimization, 49 (2017) 1317-1334.

[20] S.O. Degertekin, L. Lamberti, I.B. Ugur, Sizing, layout and topology design optimization of truss structures using the Jaya algorithm, Applied Soft Computing, 70 (2018) 903-928.

[21] A. Kaveh, P. Zakian, Improved GWO algorithm for optimal design of truss structures, Engineering with Computers, 34 (2018) 685-707.

[22] M. Aslani, P. Ghasemi, A.H. Gandomi, Constrained mean-variance mapping optimization for truss optimization problems, The Structural Design of Tall and Special Buildings, 27 (2018) e1449.

[23] M. Khatibinia, H. Yazdani, Accelerated multi-gravitational search algorithm for size optimization of truss structures, Swarm and Evolutionary Computation, 38 (2018) 109-119.

[24] S. Talatahari, A.H. Gandomi, G.J. Yun, Optimum design of tower structures using Firefly Algorithm, The Structural Design of Tall and Special Buildings, 23 (2014) 350-361.

[25] J. Pierezan, L. dos Santos Coelho, V. Cocco Mariani, E. Hochsteiner de Vasconcelos Segundo, D. Prayogo, Chaotic coyote algorithm applied to truss optimization problems, Computers & Structures, 242 (2021) 106353.

[26] H.F. Eid, L. Garcia-Hernandez, A. Abraham, Spiral water cycle algorithm for solving multi-objective optimization and truss optimization problems, Engineering with Computers, 38 (2022) 963–973.

[27] F.K.J. Jawad, C. Ozturk, W. Dansheng, M. Mahmood, O. Al-Azzawi, A. Al-Jemely, Sizing and layout optimization of truss structures with artificial bee colony algorithm, Structures, 30 (2021) 546-559.

[28] S.O. Degertekin, G. Yalcin Bayar, L. Lamberti, Parameter free Jaya algorithm for truss sizing-layout optimization under natural frequency constraints, Computers & Structures, 245 (2021) 106461.

[29] A. Mortazavi, Size and layout optimization of truss structures with dynamic constraints using the interactive fuzzy search algorithm, Engineering Optimization, 53 (2021) 369-391.

[30] M. Shahabsafa, R. Fakhimi, W. Lei, S. He, J.R.R.A. Martins, T. Terlaky, L.F. Zuluaga, Truss topology design and sizing optimization with guaranteed kinematic stability, Structural and Multidisciplinary Optimization, 63 (2021) 21-38.

[31] F.K.J. Jawad, M. Mahmood, D. Wang, O. AL-Azzawi, A. AL-JAMELY, Heuristic dragonfly algorithm for optimal design of truss structures with discrete variables, Structures, 29 (2021) 843-862.

[32] R. Chelouah, P. Siarry, Genetic and Nelder–Mead algorithms hybridized for a more accurate global optimization of continuous multiminima functions, European Journal of Operational Research, 148 (2003) 335-348.

[33] S.-K.S. Fan, E. Zahara, A hybrid simplex search and particle swarm optimization for unconstrained optimization, European Journal of Operational Research, 181 (2007) 527-548.

[34] Z.M. Yaseen, M.F. Allawi, H. Karami, M. Ehteram, S. Farzin, A.N. Ahmed, S.B. Koting, N.S. Mohd, W.Z.B. Jaafar, H.A. Afan, A. El-Shafie, A hybrid bat–swarm algorithm for optimizing dam and reservoir operation, Neural Computing and Applications, 31 (2019) 8807-8821.

[35] H. Nenavath, R.K. Jatoth, Hybrid SCA–TLBO: a novel optimization algorithm for global optimization and visual tracking, Neural Computing and Applications, 31 (2019) 5497-5526.

[36] R. Devarapalli, V. Kumar, Power system oscillation damping controller design: A novel approach of integrated HHO-PSO algorithm, Archives of Control Sciences, 31 (2021) 553-591.

[37] Y. Li, Y. Peng, S. Zhou, IMPROVED PSO ALGORITHM FOR SHAPE AND SIZING OPTIMIZATION OF TRUSS STRUCTURE, Journal of Civil Engineering and Management, 19 (2013) 542-549.

[38] F.A. Şenel, F. Gökçe, A.S. Yüksel, T. Yiğit, A novel hybrid PSO–GWO algorithm for optimization problems, Engineer ing with Computers, 35 (2019) 1359-1373.

[39] S.N. Chegini, A. Bagheri, F. Najafi, PSOSCALF: A new hybrid PSO based on Sine Cosine Algorithm and Levy flight for solving optimization problems, Applied Soft Computing, 73 (2018) 697-726.

[40] I.N. Trivedi, P. Jangir, A. Kumar, N. Jangir, R. Totlani, A Novel Hybrid PSO–WOA Algorithm for Global Numerical Functions Optimization, in: Advances in Computer and Computational Sciences 2018, pp. 53-60.

[41] G. Dhiman, A. Kaur, A Hybrid Algorithm Based on Particle Swarm and Spotted Hyena Optimizer for Global Optimization in: Soft Computing for Problem Solving, Springer Singapore, Singapore, 2019, pp. 599-615.

[42] S. Jiang, C. Zhang, S. Chen, Sequential Hybrid Particle Swarm Optimization and Gravitational Search Algorithm with Dependent Random Coefficients, Mathematical Problems in Engineering, 2020 (2020) 1-17.

[43] D. Dhawale, V.K. Kamboj, P. Anand, An improved Chaotic Harris Hawks Optimizer for solving numerical and engineering optimization problems, Engineering with Computers, (2021).

[44] A. Kaveh, P. Rahmani, A.D. Eslamlou, An efficient hybrid approach based on Harris Hawks optimization and imperialist competitive algorithm for structural optimization, Engineering with Computers, 38 (2022) 1555-1583.

[45] M. Abd Elaziz, D. Yousri, S. Mirjalili, A hybrid Harris

hawks-moth-flame optimization algorithm including fractional-order chaos maps and evolutionary population dynamics, Advances in Engineering Software, 154 (2021) 102973.

[46] M. Abdel-Basset, W. Ding, D. El-Shahat, A hybrid Harris Hawks optimization algorithm with simulated annealing for feature selection, Artificial Intelligence Review, 54 (2021) 593-637.

[47] R. Al-Wajih, S.J. Abdulkadir, N. Aziz, Q. Al-Tashi, N. Talpur, Hybrid Binary Grey Wolf With Harris Hawks Optimizer for Feature Selection, IEEE Access, 9 (2021) 31662-31677.

[48] T. Yokota, T. Taguchi, M. Gen, A solution method for optimal weight design problem of 10 bar truss using genetic algorithms, Computers & Industrial Engineering, 35 (1998) 367-372.

[49] R. Awad, Sizing optimization of truss structures using the political optimizer (PO) algorithm, Structures, 33 (2021) 4871-4894.

[50] H. Varaee, M.R. Ghasemi, Engineering optimization based on ideal gas molecular movement algorithm, Engineering with Computers, 33 (2017) 71-93.

[51] A. Kaveh, T. Bakhshpoori, A new metaheuristic for continuous structural optimization: water evaporation optimization, Structural and Multidisciplinary Optimization, 54 (2016) 23-43.

[52] B. Adil, B. Cengiz, Optimal design of truss structures using weighted superposition attraction algorithm, Engineering with Computers, 36 (2020) 965-979.

[53] E. Pouriyanezhad, H. Rahami, S.M. Mirhosseini, Truss optimization using eigenvectors of the covariance matrix, Engineering with Computers, 37 (2021) 2207-2224.

[54] M. Jafari, E. Salajegheh, J. Salajegheh, An efficient hybrid of elephant herding optimization and cultural algorithm for optimal design of trusses, Engineering with Computers, 35 (2019) 781-801.

[55] G. Dhiman, ESA: a hybrid bio-inspired metaheuristic optimization approach for engineering problems, Engineering with Computers, 37 (2021) 323-353.

[56] A. Kaveh, R. Mahdipour Moghanni, S.M. Javadi, Optimum design of large steel skeletal structures using chaotic firefly optimization algorithm based on the Gaussian map, Structural and Multidisciplinary Optimization, 60 (2019) 879-894.

[57] H. Cao, X. Qian, Z. Chen, H. Zhu, Enhanced particle swarm optimization for size and shape optimization of truss structures, Engineering Optimization, 49 (2017) 1939-1956.

[58] A.H. Gandomi, S. Talatahari, X.-S. Yang, S. Deb, Design optimization of truss structures using cuckoo search algorithm, The Structural Design of Tall and Special Buildings, 22 (2013) 1330-1349.

HOW TO CITE THIS ARTICLE
M. Yassami, P. Ashtari, PSOHHO Hybrid Optimization Algorithm for Truss Optimization, AUT J. Civil Eng., 6(2) (2022) 295-318.

DOI: 10.22060/ajce.2023.21732.5810