Engineering with Computers (2022) 38:2743–2767
https://doi.org/10.1007/s00366-020-01236-z

ORIGINAL ARTICLE

Colliding bodies optimization with Morlet wavelet mutation and quadratic interpolation for global optimization problems

Ali Kaveh[1] · **Majid Ilchi Ghazaan**[1] · **Fatemeh Saadatmand**[1]

Received: 1 October 2020 / Accepted: 3 December 2020 / Published online: 4 January 2021

Abstract
This paper represents a new variant of colliding bodies optimization (CBO) and the objective is to alleviate the lack of population diversity, premature convergence phenomenon, and the imbalance between the diversification and intensification of the CBO method. The CBO is a meta-heuristic algorithm based on momentum and energy laws in a one-dimensional collision between two bodies. The proposed method is designed by hybridization of the CBO with Morlet wavelet (MW) mutation and quadratic interpolation (QI) (MWQI-CBO). The Morlet wavelet mutation is employed to improve the CBO so that it can explore the search space more effectively on reaching a better solution. Besides, quadratic interpolation that utilized historically best solution is added to CBO to enhance the exploitation phase. Two new parameters are defined to have a better balance between the diversification and the intensification inclinations. The proposed algorithm is tested in 24 mathematical optimization problems including 30 design variables and compared with standard CBO and some state-of-art metaheuristics. Besides, the optimal design of five standard discrete and continuous structural design problems with various constraints such as strength, stability, displacement, and frequency constraints are studied. It is found that MWQI-CBO is quite competitive with other meta-heuristic algorithms in terms of reliability, solution accuracy, and convergence speed.

Keywords Colliding bodies optimization · Quadratic interpolation · Wavelet mutation · Structural optimization · Discrete and continuous optimization

1 Introduction

In recent years, there has been research towards developing meta-heuristic algorithms (MAs) for solving engineering optimization problems since they can be readily applicable to a wide range of problems. Unlike traditional optimization methods, MAs are problem-independent algorithms and do not require gradient information for finding suboptimal/optimal solution in optimization problems. The feasibility and effectiveness of meta-heuristic algorithms on the optimal design of structures are studied by many researchers [1–7].

Meta-heuristic algorithms (MAs) generally have a significant performance in seeking the search space but they may face some problems during their performance which can make the algorithms unable to find the optima; they can be trapped in the local minima. There have been several studies addressing this issue for different optimization algorithms. One of the operators that has been commonly used in various optimization problems is mutation. In a genetic algorithm (GA) [8], after survival of the best solutions, one candidate solution is selected from the population for regenerating by a mutation operator to enhance the diversity of the solutions, thereby ensuring a well-performed search. Ling and Leung proposed a real-coded genetic algorithm (RCGA) with new genetic operations including wavelet mutation and a crossover operator to minimize operation cost in economic load dispatch problems [9]. In HPSOWM [10], a mutation with a dynamic mutating space by incorporating Morlet wavelet mutation is utilized to prevent the algorithm from immature convergence and has performed on three industrial applications to solve the load flow problems, model the development of fluid dispensing for electronic packaging and to design a neural network-based controller. Mondal et al. introduced a differential evolution with wavelet mutation for the optimal design of linear phase finite impulse response filters [11]. A hybridized version of the gravitational search algorithm (GSA) and wavelet mutation (WM) strategy is utilized for

✉ Ali Kaveh
alikaveh@iust.ac.ir

[1] School of Civil Engineering, Iran University of Science and Technology, P.O. Box 16846-13114, Tehran, Iran

the design of an 8th-order infinite impulse response (IIR) filter [12]. In a binary hybrid particle swarm optimization introduced by Jiang et al. [13], a mutation process based on wavelet theory is adopted to improve the searchability of standard PSO and has been tested in various mathematical experiments to evaluate the algorithm validity. In SMWOA [14], the Morlet wavelet mutation is incorporated into the WOA's exploration phase to enhance the algorithm's ability to jump out of local optima and improve the convergence speed and accuracy of the algorithm. This algorithm is used for solving the three water resources forecasting in Shaanxi Province of China and the results have shown great prediction accuracy of 99.68%.

On the other hand, it can be seen that some meta-heuristic algorithms suffering from some drawbacks like inefficient search near global optima and low convergence speed, so that, they are needed to be improved by some operators. One of the reliable operators is quadratic interpolation (QI) crossover which has been widely used in MAs. Pant et al. presented a new variant of the basic particle swarm optimization (BPSO) algorithm named QIPSO for solving global optimization problems [15]. Deep and Das introduced a quadratic approximation-based hybrid genetic algorithm for function optimization in which QI is utilized for generating new offspring in crossover operation [16]. In a modified whale optimization algorithm (MWOA), a Lévy-flight strategy with a quadratic interpolation method is applied to the leader of the population to enhance solution accuracy; this method is employed to solve large-scale optimization problems [17]. In QIWOA [18] proposed by Sun et al., QI is adopted to improve the exploitation process. A QI-based teaching–learning-based optimization is introduced by Chen et al. and is applied to solve six chemical dynamic optimization problems including three parameter estimation problems and three optimal control problems [19].

Colliding bodies optimization (CBO) is a recently developed population-based algorithm introduced by Kaveh and Mahdavi [20] based on collision's concept in physics. In this method, every search agent is considered as a colliding body (CB) with specified mass and velocity that the positions of CBs will be updated after a collision occurs between two bodies to find better positions in the search space. The CBO almost has a good convergence rate but the possibility that the algorithm will be trapped in the local optima exists. By proposing enhanced CBO (ECBO) [21], tried to alleviate CBO's drawbacks using a mechanism to escape from local minima and a memory to store some best solutions. The performance of the CBO and ECBO is studied in solving various kinds of optimization problems [22, 23]. This paper represents a novel algorithm called MWQI-CBO. In this algorithm, Morlet wavelet (MW) mutation [10] and quadratic interpolation (QI) crossover [18] are employed to improve the performance of the CBO. These two mechanisms are utilized in diversification and intensification phases, respectively. Here, viability of the MWQI-CBO is examined using 24 mathematical benchmark functions and 5 structural design problems. Results show that the proposed algorithm is a robust and reliable method.

The rest of this paper is organized as follows: In Sect. 2, a brief overview of the CBO is presented. Section 3 introduces the efficient version of the CBO based on MW mutation and QI (MWQI-CBO). Section 4 utilizes benchmark mathematical functions and structural design problems to compare MWQI-CBO against CBO, ECBO, and some other well-known optimization methods. Concluding remarks are described in Sect. 5.

2 A brief explanation of the CBO algorithm

The CBO is a population-based algorithm proposed by Kaveh and Mahdavi [20], based on the collision phenomenon between two bodies which are called colliding bodies (CBs). After each collision, two CBs will move toward new positions based on updating equations which will be presented in the following.

In this method, CBs have a specified mass defined as:

$$m_k = \frac{\frac{1}{\text{fit}(k)}}{\frac{1}{\sum_{i=1}^{n}\frac{1}{\text{fit}(i)}}}, \quad k = 1, 2, \ldots, n, \tag{1}$$

where $\text{fit}(i)$ represents the objective function value of the ith candidate solution and n is the number of CBs. Obviously, a CB with good values exerts a larger mass than the bad ones. In addition, for maximization, the objective function $\text{fit}(i)$ will be replaced by $\frac{1}{\text{fit}(i)}$.

Two CBs for collision are selected from two equal groups named (1) stationary CBs and (2) moving *CBs* which are generated from organized CBs in a descending order based on their mass values. The first half of these organized CBs is for the first group and the second half of them is for the second group. Moving CBs collide stationary CBs to move them towards better positions and improve themselves' positions. The velocity of the CBs in the stationary group before the collision is zero. Thus,

$$v_i = 0, \quad i = 1, 2, \ldots, \frac{n}{2}. \tag{2}$$

The velocity of each CB in moving group before collision is

$$v_i = x_{i-\frac{n}{2}} - x_i, \quad i = \frac{n}{2} + 1, \frac{n}{2} + 2, \ldots, n. \tag{3}$$

The velocity of each stationary CB after collision is obtained by:

$$v_i' = \frac{\left(m_{i+\frac{n}{2}} + \varepsilon m_{i+\frac{n}{2}}\right) v_{i+\frac{n}{2}}}{m_i + m_{i+\frac{n}{2}}}, \quad i = 1, 2, \ldots, \frac{n}{2}. \tag{4}$$

The velocity of each moving CB after collision is defined by:

$$v_i' = \frac{\left(m_i - \varepsilon m_{i-\frac{n}{2}}\right) v_i}{m_i + m_{i-\frac{n}{2}}}, \quad i = \frac{n}{2} + 1, \frac{n}{2} + 2, \ldots, n \tag{5}$$

where ε is the coefficient of restitution (COR) which decreases linearly from 1 to zero and is defined as:

$$\varepsilon = 1 - \frac{t}{t_{\max}}, \tag{6}$$

where t is the current iteration and $t_{\max}$ is the maximum number of iterations.

New positions of CBs are evaluated based on the velocities generated after collision and the position of stationary CBs. Thus, the new position of each stationary CB is obtained by:

$$x_i^{\text{new}} = x_i + \text{rand} \circ v_i', \quad i = 1, 2, \ldots, \frac{n}{2}, \tag{7}$$

where x_i^{new},x_i and v_i' are the new position, current position and the velocity of the ith CB after collision, respectively. rand is a random vector uniformly distributed in the interval of [− 1,1] and the sign "∘" denotes an element-by-element multiplication.

The new position of each moving CB is defined by:

$$x_i^{\text{new}} = x_{i-\frac{n}{2}} + \text{rand} \circ v_i', \quad i = \frac{n}{2} + 1, \frac{n}{2} + 2, \ldots, n. \tag{8}$$

For further details, the reader may refer to [20].

3 Colliding bodies optimization with Morlet wavelet mutation and quadratic interpolation (MWQI-CBO)

Standard CBO has some shortcomings such as lack of diversity and trapping into local optimum. To prevent these issues and improve solution stability, Morlet wavelet mutation is used in the exploration phase due to its fine-tuning ability. However, the standard CBO has a good performance in the exploitation phase but its convergence rate is low. To improve the exploitation phase, quadratic interpolation crossover is used in this algorithm to improve the search near the global-best search agent. The implementation of the exploration and exploitation phase is controlled by two parameters A and B. Details of the proposed algorithm are described more in the following subsections.

3.1 Improving exploration phase

The CBO can easily be trapped in local optima which can prevent the algorithm to search the whole search space. To improve the exploration phase, reliability of the search and the stability of solutions, Morlet wavelet mutation is employed in this study. Based on "Wavelet theory", certain seismic signals can be modeled by combining translations and expansions of an oscillatory function within finite duration called a "wavelet". A continuous time function $\psi(x)$ is called a "mother wavelet" or a "wavelet" if it satisfies the following properties [10].

Property 1

$$\int_{-\infty}^{+\infty} \psi(x) \mathrm{d}x = 0. \tag{9}$$

This equation shows that the total negative and positive momentum of $\psi(x)$ is equal.

On the other hand, it is possible to show that the admissibility condition implies that $\hat{\psi}(0) = 0$ so that a wavelet must integrate to zero. Notice that $\hat{\psi}$ is Fourier transform of wavelet ψ, and the admissibility condition is defined as follows [10]:

$$0 < C_\psi < +\infty, \quad C_\psi = \int_{-\infty}^{+\infty} \frac{|\hat{\psi}(v)|^2}{|v|} \mathrm{d}v. \tag{10}$$

Property 2

$$\int_{-\infty}^{+\infty} |\psi(x)|^2 \mathrm{d}x < \infty. \tag{11}$$

Morlet wavelet is the example of $\psi(x)$ which integrates to zero (property 1) and over 99% of total energy of the function is contained in [− 2.5,2.5] (property 2).

The implementation of Morlet wavelet mutation in MWQI-CBO is controlled by parameter A, which is defined as:

$$A = 0.9 + 0.1 \times \frac{t}{t_{\max}}, \tag{12}$$

where t is the current iteration and $t_{\max}$ is the maximum number of iterations. By increasing the iterations, the probability of using MW is reduced. When *rand* which is a random number in the interval of [0,1] is more than A, the positions of a candidate solution is updated by the following formula.

$$x_{ij}^{\text{new}} = \begin{cases} x_{ij}(t) + \sigma\left(\text{varmax} - x_{ij}(t)\right) & \text{if} \quad \sigma > 0 \\ x_{ij}(t) + \sigma\left(x_{ij}(t) - \text{varmin}\right) & \text{if} \quad \sigma \le 0 \end{cases}, \tag{13}$$

where x_{ij} is the jth variable of CB i, varmax and varmin are the upper and lower bounds of each variable. $\sigma = \frac{1}{\sqrt{a}}\psi\left(\frac{\phi_j}{a}\right)$ is the coefficient of wavelet mutation and $\psi(x) = \mathrm{e}^{-\frac{x^2}{2}}\cos(5x)$ is Morlet wavelet mutation where ϕ_j is a random number in the range of [− 2.5a, 2.5a]. a is the scaling parameter of MW and increases from 1 to S as the number of iterations rises.

$$a = s\left(\frac{1}{s}\right)^{\left(1-\frac{t}{t_{\max}}\right)}, \tag{14}$$

where S is a constant number is set to 10,000. When $t/t_{\max}$ is 0, a is 1 and as a result, σ is obtained as 1. By substituting that in Eq. (13), varmax will be obtained which shows that the whole search space has the chance to be explored. In the other side, if $t = t_{\max}$, a is equal to $S = 10{,}000$, so that, the value of σ will become so small which leads the algorithm to search in a smaller search area.

3.2 Improving exploitation phase

The CBO generally has a good exploitation phase but to obtain fast convergence and improve its intensification, quadratic interpolation crossover is utilized in this study which is a local search operator. Mathematically, QI uses a parabola that curve passes through three points to find the minimum point of the curve in D-dimensional space [18]. QI crossover is defined as:

$$x_i = 0.5 \times \frac{\left(y_i^2 - z_i^2\right)\cdot f(X^*) + \left(z_i^2 - x_i^{*2}\right)\cdot f(Y) + \left(x_i^{*2} - y_i^2\right)\cdot f(Z)}{\left(y_i - z_i\right)\cdot f(X^*) + \left(z_i - x_i^*\right)\cdot f(Y) + \left(x_i^* - y_i\right)\cdot f(Z)}, \quad i = 1, 2, \ldots, d, \tag{15}$$

where $f(x^*)$, $f(y)$, and $f(z)$ are the fitness of the three distinct search agents X^*, Y, and Z, respectively. $X^* = (x_1, x_2, \ldots, x_d)$ is the global-best search agent, $Y = (y_1, y_2, \ldots, y_d)$ and $Z = (z_1, z_2, \ldots, z_d)$ are two randomly selected from current population. $X = (x_1, x_2, \ldots, x_d)$ is the generated solution vector by QI crossover which is around the optimal solution vector. The implementation of QI is controlled by parameter B which is set to 0.15.

3.3 Procedure of the proposed MWQI-CBO

According to the previous parts, the following steps introduce the MWQI-CBO algorithm.

Step 1 Initialize MWQI-CBO algorithm parameters. The positions of all CBs are randomly set within predefined ranges and the objective function is evaluated for each CB:

$$x_i^0 = x_{\min} + \text{random} \circ \left(x_{\min} - x_{\max}\right), i = 1, 2, \ldots, n, \tag{16}$$

where x_i^0 is the initial position of the ith CB, $x_{\min}$ and $x_{\max}$ are the lower and upper bounds of each variables in the search space; random is a randomly generated vector which each component is in the interval [0,1]; n is the number of CBs.

Step 2 To enhance the convergence speed, some of the historically best CBs are replaced with the worst CBs in the current population.

Step 3 Solution candidates are divided into stationary and moving groups.

Step 4 The positions of each two colliding bodies are updated by the following procedure.

If rnd > A, the new location is updated by Eq. (13); otherwise, the equations proposed by CBO or QI is employed for updating CBs. If rnd < B, QI is adopted for updating candidate solutions; otherwise, the new positions are calculated by Eqs. (7) and (8). rnd is a random number uniformly distributed in the range of [0,1].

Step 5 With a predefined probability, one of the components of each CB is changed to make population diversity. The new component is determined randomly in the search space. The selected probability is usually set to a small value and only one dimension is expected to be regenerated to protect the structure of the candidate solution.

Step 6 When the terminal condition is met, the optimization process is terminated; Otherwise, go to step 2 for a new round.

4 Experiments and optimization results

4.1 Experiment 1: mathematical optimization problems

In this experiment, 24 benchmark functions are adopted from [24] to evaluate the performance of the proposed method. These functions are divided into unimodal (f_1–f_{12}) and multimodal (f_{13}–f_{24}) functions which are described in Table 1. The unimodal and multimodal functions are generally taken to test the exploitation and exploration phases, respectively. These functions have either a narrow valley, basin, or a huge number of local optima, which are challenging for optimization algorithms. For all mathematical problems, population size is set to 20 with the maximum function evaluation of

Table 1 Mathematical benchmark functions

Function	Equation	Geometry shape	$f_{\min}$
Sphere	$f_1 = \sum_{i=1}^{D} x_i^2$	Unimodal	0
Noncontiguous-sphere	$f_2 = \sum_{i=1}^{D} \left(x_i^2 + 0.5\right)^2$	Unimodal	0
Uniform-distribution perturbed Ellipsoid	$f_3 = \sum_{i=1}^{D} i x_i^2 + \bigcup(0,\delta)$	Unimodal	0
Rosenbrock	$f_4 = \sum_{i=1}^{D-1} (100(x_{i+1} - x_i^2)^2 + \left(x_i - 1\right)^2)$	Unimodal	0
Schwefel	$f_5 = \sum_{i=1}^{D} \lvert x_i\rvert + \prod_{i=1}^{D} \lvert x_i\rvert$	Unimodal	0
1st Schwefel	$f_6 = \sum_{i=1}^{D} \left(\sum_{j=1}^{i} x_j\right)^2$	Unimodal	0
Alpine	$f_7 = \sum_{i=1}^{D} (\left\lvert x_i \sin\left(x_i\right)\right\rvert + 0.1 x_i)$	Unimodal	0
Different power	$f_8 = \sum_{i=1}^{D} \lvert x_i\rvert^{i+1}$	Unimodal	0
Bent Cigar	$f_9 = x_1^2 + 10^6 \sum_{i=1}^{D} x_i^2$	Unimodal	0
Discus	$f_{10} = 10^6 x_1^2 + \sum_{i=1}^{D} x_i^2$	Unimodal	0
Zakharov	$f_{11} = \sum_{i=1}^{D} x_i^2 + \left(\sum_{i=1}^{D} 0.5 x_i\right)^2 + \left(\sum_{i=1}^{D} 0.5 x_i\right)^4$	Unimodal	0
High conditioned elliptic	$f_{12} = \sum_{i=1}^{D} (10^6)^{\frac{i-1}{D-1}} x_i^2$	Unimodal	0
Rastrigin	$f_{13} = \sum_{i=1}^{D} (x_i^2 - 10\cos\left(2\pi x_i\right) + 10)$	Multimodal	0
2nd Schwefel	$f_{14} = \sum_{i=1}^{D} \lvert x_i\rvert \, \lvert\sin(\sqrt{\lvert x_i\rvert})\rvert$	Multimodal	0
Ackley	$f_{15} = -20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right) - \exp(\frac{1}{D}\sum_{i=1}^{D}\cos\left(2\pi x_i\right) + 20 + e$	Multimodal	0
Griewank	$f_{16} = \frac{1}{4000}\sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D}\cos\left(\frac{x_i}{\sqrt{i}} + 1\right)$	Multimodal	0
Schaffer	$f_{17} = 0.5 + \frac{\sin^2\left(\sqrt{\sum_{i=1}^{D} x_i^2}\right) - 0.5}{1 + 0.1\sum_{i=1}^{D} x_i^2}$	Multimodal	0
Levy	$f_{18} = \sin^2\left(\pi w_1\right) + \sum_{i=1}^{D-1} (w_i - 1)^2(1 + 10\sin^2\left(\pi w_i + 1\right)) + \left(w_D - 1\right)^2(1 + \sin^2\left(2\pi w_D\right))$	Multimodal	0
Weierstrass	$f_{19} = \sum_{i=1}^{D}\left(\sum_{k=0}^{k\max}\left(a^k\cos\left(2\pi b^k\left(x_i + 0.5\right)\right)\right)\right) - D\sum_{k=0}^{k\max} a^k\cos\left(\pi b^k\right)$	Multimodal	0
Katsuura	$f_{20} = \frac{10}{D^2}\prod_{i=1}^{D}(1 + i\sum_{j=1}^{32}\frac{\left\lvert 2^j x_i - \mathrm{round}\left(2^j x_i\right)\right\rvert}{2^j})^{\frac{10}{D^{1.2}}} - \frac{10}{D^2}$	Multimodal	0
Expanded Schaffer	$f_{21} = f_{12}\left(x_1, x_2\right) + f_{12}\left(x_2, x_3\right) + \ldots + f_{12}\left(x_{D-1}, x_D\right) + f_{12}\left(x_D, x_1\right)$	Multimodal	0
HappyCat	$f_{22} = \left\lvert\sum_{i=1}^{D} x_i^2 - D\right\rvert^{\frac{1}{4}} + \frac{\frac{1}{2}\sum_{i=1}^{D} x_i^2 + \sum_{i=1}^{D} x_i}{D} + \frac{1}{2}$	Multimodal	0

Table 1 (continued)

Function	Equation	Geometry shape	$f_{\min}$
HGBat	$f_{23} = \left\lvert \left(\sum_{i=1}^{D} x_i^2 \right)^2 - \left(\sum_{i=1}^{D} x_i \right)^2 \right\rvert^{\frac{1}{2}} + \frac{\frac{1}{2}\sum_{i=1}^{D} x_i^2 + \sum_{i=1}^{D} x_i}{D} + \frac{1}{2}$	Multimodal	0
Expanded Griewank	$f_{24} = f_{10}(f5(x_1, x_2)) + f_{10}(f5(x_2, x_3)) + \cdots + f_{10}(f5(x_{D-1}, x_D)) + f_{10}(f5(x_D, x_1))$	Multimodal	0

$w_i = 1 + \frac{x_i - 1}{4} (\forall i = 1, \ldots, D), a = \frac{1}{2}, b = 3, k\max = 20$

Table 2 Performance comparison for the 24 mathematical benchmark functions

Function	G-PSO [25]	PSO-DTT [26]	PSO-DFCM [24]	CBO	ECBO	MWQI-CBO
f_1	3.4733e−15	4.5759e−30	1.010e−38	1.0059e−16	4.0987e−17	**0**
f_2	**0**	**0**	**0**	**0**	**0**	**0**
f_3	0.0079	0.0059	**0.0021**	0.0758	0.0206	0.01073
f_4	11.536	17.993	0.2639	4.7013	0.0704	**2.4996e−04**
f_5	0.54216	3.3967e−05	2.3276e−09	1.75e−13	1.0542e−07	**0**
f_6	1.0513e−06	1.2530e−08	2.7669e−07	38.4346	2.9832	**0**
f_7	5.7395e−07	1.7238e−07	1.1705e−11	7.8902e−12	1.0899e−05	**0**
f_8	9.91533	6.1246e−13	8.5617e−17	6.7636e+06	8.786e−49	**0**
f_9	1.0000e−18	1.4076e−23	2.762e−26	9.1774e−14	5.912e−12	**0**
f_{10}	1.3158e−25	3.1181e−30	1.4563e−40	6.4884e−16	7.9111e−04	**0**
f_{11}	3.1189e−25	1.671e−27	2.0309e−35	3.7753e+04	1.6484e+04	**0**
f_{12}	1.1826e+05	8.5218e−10	5.8791e−25	1.1210e−16	7.9077e−10	**0**
f_{13}	21.8891	**0**	1.1997e−15	136.3085	1.959e−10	**0**
f_{14}	5.4424e−10	9.3069e−14	3.1774e−14	4.7985e−13	4.5752e−08	**0**
f_{15}	20	2.66e−15	8.8817e−08	20.8543	20.0042	**8.8818e−16**
f_{16}	0.0369	1.1102e−18	**0**	**0**	**0**	**0**
f_{17}	0.39813	0.44918	2.4520e−04	0.4989	0.4492	**0**
f_{18}	5.67708e−25	3.2198e−27	**3.0815e−33**	148.9869	1.3082e−17	7.5290e−21
f_{19}	28.1026	13.5372	0.0072	32.9654	1.7672	**0**
f_{20}	**0**	**0**	**0**	**0**	0.0026	**0**
f_{21}	11.492	11.492	2.7153e−04	11.2481	2.0928e−14	**0**
f_{22}	0.25112	0.24713	0.2106	0.1408	0.2670	**0.1371**
f_{23}	3.7537e−04	3.0054e−04	**2.3162e−06**	33.9933	0.1804	0.0363
f_{24}	2.8654	2.7687	1.4319	28.7074	1.4793	0

The smallest values in each row are bolded

100,000 and 20 independent runs were performed under 30 dimensions. In addition, the search space for all problems is − 100 ~ 100.

4.1.1 Results analysis

In Table 2, optimal values of MWQI-CBO, CBO, ECBO and several variants of PSO including gravitational PSO (G-PSO) [25], PSO using dynamic tournament topology (PSO-DTT) [26], and particle swarm optimization with damping factor and cooperative mechanism (PSO-DFCM) [24] have been compared. As illustrated in Table 2, for all the unimodal functions except f_3, the MWQI-CBO has the best performance among six algorithms. In f_3, PSO-DFCM gains the best results although it cannot find the optimal value. Moreover, among 12 unimodal functions, the proposed method achieved the global best for 10 functions (f_1, f_2, f_5–f_{12}) which can verify the algorithm's great performance in exploitation. The other five algorithms can find the global optima for f_2. For multimodal functions, the proposed algorithm outperforms five others in most functions except for f_{18} and f_{23}. Furthermore, MWQI-CBO can achieve the local optima for 8 functions (f_{13}, f_{14}, f_{16}, f_{17}, f_{19}–f_{21}, and f_{24}) that can prove a good exploration ability of the algorithm. PSO-DTT can also find the optimal value for f_{13} as well as MWQI-CBO. All the algorithms except ECBO have the capability to find the global optimum for f_{20}.

Table 3 Statistical results of mathematical benchmark functions for CBO, ECBO, and MWQI-CBO

Function	CBO	ECBO	MWQI-CBO
f_1			
Best	1.0059e−16	4.0987e−17	**0**
Mean	1.1666e−09	1.415e−15	**0**
Std	5.3109e−09	3.0259e−15	**0**
Rank	3(–)	2(–)	1
f_2			
Best	**0**	**0**	**0**
Mean	**0**	**0**	**0**
Std	**0**	**0**	**0**
Rank	1(~)	1(~)	1
f_3			
Best	0.0758	0.0206	0.01073
Mean	0.1691	0.0383	**0.0187**
Std	0.0381	0.0103	**0.0054**
Rank	3(~)	2(~)	1
f_4			
Best	4.7013	0.0704	**2.4996e−04**
Mean	5.4358e+03	193.7732	**42.3602**
Std	1.4068e+04	319.9334	**68.3999**
Rank	3(–)	2(–)	1
f_5			
Best	1.75e−13	1.0542e−07	**0**
Mean	1.2463e−06	3.7685e−07	**0**
Std	4.1586e−06	2.018e−07	**0**
Rank	3(–)	2(–)	1
f_6			
Best	38.4346	2.9832	**0**
Mean	495.4065	9.3523	**0.0141**
Std	338.5972	14.1745	**0.0143**
Rank	3(–)	2(–)	1
f_7			
Best	7.8902e−12	1.0899e−05	**0**
Mean	4.0664e−07	2.0691e−04	**2.5178e−07**
Std	1.5105e−07	2.0671e−04	2.4753e−07
Rank	2(~)	3(–)	1
f_8			
Best	6.7636e+06	8.786e−49	**0**
Mean	5.6722e+19	1.8706e−37	**0**
Std	2.3445e+20	7.8777e−37	**0**
Rank	3(–)	2(–)	1
f_9			
Best	9.1774e−12	5.912e−12	**0**
Mean	2.5374e−06	5.6164e−10	**0**
Std	7.8352e−05	1.7047e−10	**0**
Rank	3(–)	2(–)	1
f_{10}			
Best	6.4884e−16	7.9111e−04	**0**
Mean	2.3062e−10	271.8075	**0**
Std	6.1054e−10	439.6761	**0**
Rank	2(–)	3(–)	1
f_{11}			
Best	3.7753e+04	1.6484e+04	**0**
Mean	1.6528e+05	4.1933e+04	**1.3349e+03**
Std	9.0981e+04	2000	**1.7329e+03**
Rank	3(–)	2(–)	1
f_{12}			
Best	1.1210e−16	7.9077e−10	**0**
Mean	1.9459e−07	1.3838e−05	**0**
Std	8.5823e−07	5.2898e−05	**0**
Rank	2(–)	3(–)	1
f_{13}			
Best	136.3085	1.959e−10	**0**
Mean	418.4701	0.3982	**0.398**
Std	205.6728	**0.5004**	0.6771
Rank	3(–)	2(~)	1
f_{14}			
Best	4.7985e−13	4.5752e−08	**0**
Mean	1.2423e−07	1.9622e−06	**7.0513e−08**
Std	3.879e−07	2.3141e−06	**1.2808e−07**
Rank	2(–)	3(–)	**1**
f_{15}			
Best	20.8543	20.0042	**8.8818e−16**
Mean	20.9786	20.009	**10.0557**
Std	0.0535	**0.0043**	8.0026
Rank	3(–)	2(–)	1
f_{16}			
Best	0	0	**0**
Mean	0.0399	0.0205	**0.0100**
Std	0.05	0.0246	**0.017**
Rank	3(~)	2(~)	1
f_{17}			
Best	0.4989	0.4492	**0**
Mean	0.4994	0.4701	**0.3997**
Std	**3.0287e−04**	0.0054	0.1384
Rank	3(–)	2(–)	1
f_{18}			
Best	148.9869	1.3082e−17	**7.5290e−21**
Mean	729.8776	1.611e−15	**4.0941e−20**
Std	395.4678	2.2347e−14	**3.1553e−20**
Rank	3(–)	2(–)	1
f_{19}			
Best	32.9654	1.7672	**0**
Mean	40.1779	2.2571	**0.4026**
Std	2.1766	**0.6144**	0.7791
Rank	3(–)	2(–)	1
f_{20}			
Best	**0**	0.0026	**0**
Mean	**0**	0.0053	0.0011
Std	**0**	0.0016	0.0015

Table 3 (continued)

Function	CBO	ECBO	MWQI-CBO
Rank	1(+)	3(~)	2
f_{21}			
Best	11.2481	2.0928e−14	**0**
Mean	12.2710	2.4343	**0.5574**
Std	**0.5998**	2.6164	1.1638
Rank	3(–)	2(–)	1
f_{22}			
Best	0.1408	0.2670	**0.1371**
Mean	0.4586	0.4677	**0.3668**
Std	0.1541	0.1383	**0.1113**
Rank	2(~)	3(~)	1
f_{23}			
Best	0.2332	0.1804	**0.0363**
Mean	0.5468	0.4537	**0.279**
Std	0.3019	0.2124	**0.1136**
Rank	3(–)	2(–)	1
f_{24}			
Best	33.9933	1.4793	**0**
Mean	151.4149	2.3237	**2.441**
Std	128.5221	**0.527**	1.2359
Rank	3(–)	2(~)	1
Average rank	2.67	2.21	**1.042**
Overall rank	3	2	**1**

The smallest values in each row are bolded

In Table 3, the results of CBO, ECBO, and MWQI-CBO algorithms for mathematical benchmark functions are shown in more detail such as the best fitness values among 20 independent runs (Best), the mean values of them (Mean), and also standard deviations (Std). In addition, for each function, algorithms are ranked based on their mean values and the overall rank based on the average rank for each algorithm is also presented in this table. +, − and, ~ demonstrate that CBO or ECBO's performance is statistically superior, inferior, or similar to the performance of the MWQI-CBO. Besides, superior results are highlighted with boldface letters in the corresponding table. It is evident that MWQI-CBO acquires the best results for most functions in comparison with two others.

In terms of unimodal functions, MWQI-CBO achieves very competitive results. For f_1 and f_2 which are simple functions, easy to converge, and usually test for evaluating the convergence rate of the algorithms, the proposed algorithm obtained the global optimum in each run; other algorithms perform as well as MWQI-CBO for merely f_2. Besides, for f_5, f_8–f_{10}, and f_{12}, MWQI-CBO performs astonishing and the results show the solution stability and also efficient local search ability of the proposed algorithm due to utilizing quadratic interpolation crossover. It can be observed that there are not significant differences between algorithms for f_3, in spite of that, MWQI-CBO has the least mean and Std values. Although the proposed algorithm has better execution and can find the global best for f_{11} which there is a wide discrepancy at different dimensions, according to the mean value, it rarely occurs to achieve the optimum result. On the whole, it can be recognized from the results of unimodal functions, MWQI-CBO has been successful to enhance standard CBO's exploitation and there is a good balance between exploration and exploitation phases.

For multimodal functions, as can be seen from Table 3, MWQI-CBO obtained the first rank for most cases. For f_{13}, f_{14}, f_{17}, and f_{21} with a large number of local optima, the proposed algorithm is capable to find the global optima. For remain functions, MWQI-CBO is more successful than others except for f_{20}. As can be observed, CBO can find the global best in each run and the suggested algorithm achieved the second rank while it has the capability of finding the global optimum. f_{15} is characterized by a nearly flat outer region with a large hole at the center and poses a risk for optimization algorithms to be trapped in one of its many local minima. MWQI-CBO performs well for this case in comparison with other algorithms which can be understood the high potential of the proposed algorithm's exploration ability. For f_{16} which has many widespread local minima, MWQI-CBO gains better results with less mean and Std cost values and all algorithms can sometimes find global optima. For f_{18}, f_{19}, and f_{24}, both ECBO and MWQI-CBO achieve much better results compared to CBO and in all of them, MWQI-CBO obtained the first rank. It can be understood that CBO without mutation operator or improvement in the exploration phase, cannot execute an efficient search and it does need an operator to prevent it from sticking in local optima. Based on the overall results, it can be perceived that the exploration capacity of the CBO is extended as a result of the using Morlet wavelet mutation.

In conclusion, MWQI-CBO is observed to give more successful and robust results for most of the mathematical benchmark functions including both unimodal and multimodal functions, and as can be seen, it totally ranks first.

4.1.2 Wilcoxon's rank sum test

The Wilcoxon's rank sum test is a non-parametric test to detect a significant difference between the behaviors of the algorithms which is done at 5% level of significance [27]. As is shown in Table 4, there are two parameters: p value and h value. The p value of the test returned as a positive scalar from 0 to 1. p is the probability of observing a test statistic as or more extreme than the observed value under the null hypothesis. h value returned as a logical value of 1 or 0. When the p value is less than 5% or h value equals to 1, it means that there is a statistically significant difference

Table 4 Results of Wilcoxon's rank sum test for MWQI-CBO, ECBO, and CBO

Function	ECBO	CBO
f_1		
p value	8.8575e−05	8.8575e−05
h value	1	1
f_2		
p value	1	1
h value	0	0
f_3		
p value	8.8575e−05	8.8575e−05
h value	1	1
f_4		
p value	0.0479	4.4934e−04
h value	1	1
f_5		
p value	8.8575e−05	8.8575e−05
h value	1	1
f_6		
p value	8.8575e−05	8.8575e−05
h value	1	1
f_7		
p value	8.8575e−05	0.0522
h value	1	0
f_8		
p value	8.8449e−05	8.8575e−05
h value	1	1
f_9		
p value	8.8575e−05	8.8575e−05
h value	1	1
f_{10}		
p value	8.8575e−05	8.8575e−05
h value	1	1
f_{11}		
p value	8.8575e−05	8.8575e−05
h value	1	1
f_{12}		
p value	8.8575e−05	8.8575e−05
h value	1	1
f_{13}		
p value	0.1913	8.8575e−05
h value	0	1
f_{14}		
p value	8.8449e−05	8.8575e−05
h value	1	1
f_{15}		
p value	3.9023e−04	8.8575e−05
h value	1	1
f_{16}		
p value	0.0582	0.0126
h value	0	1
f_{17}		
p value	8.7949e−05	8.8199e−05
h value	1	1
f_{18}		
p value	8.8575e−05	8.8575e−05
h value	1	1
f_{19}		
p value	1.4013e−04	8.8575e−05
h value	1	1
f_{20}		
p value	0.0032	1.3183e−04
h value	1	1
f_{21}		
p value	4.4934e−04	8.8575e−05
h value	1	1
f_{22}		
p value	0.0022	0.0793
h value	1	0
f_{23}		
p value	1	0.004
h value	0	1
f_{24}		
p value	0.7089	8.8575e−05
h value	0	1

between MWQI-CBO and other algorithms; otherwise, there is a little difference between them. As can be seen from Table 4, there is a significant difference between MWQI-CBO and CBO for most functions except for f_2, f_7, and f_{22}. CBO performs better than MWQI-CBO for f_{20}. Besides, for MWQI-CBO and ECBO, there is a considerable difference for all functions except for $f_2, f_{13}, f_{16}, f_{23}$, and f_{24}.

4.1.3 Convergence rate analysis

The convergence curves of MWQI-CBO, CBO, and ECBO for 24 mathematical benchmark functions are compared in Fig. 1. It can be witnessed that the proposed algorithm has a faster convergence speed compared with other algorithms for most functions. While CBO and ECBO are trapped in local optima for $f_1, f_5, f_8, f_9, f_{10}$, and f_{12} that are unimodal functions, MWQI-CBO can escape and find global optimum in less than 1600 iterations. For f_8, the proposed algorithm performs brilliantly and can reach the global optima in nearly 50 iterations. For f_7, although CBO shows a better convergence rate than MWQI-CBO, it eventually is trapped in local optimum, but as can be seen, the suggested algorithm is successful to find the global best. In multimodal functions, the fast convergence rate of the suggested method and its high ability to escape from local optima can be observed

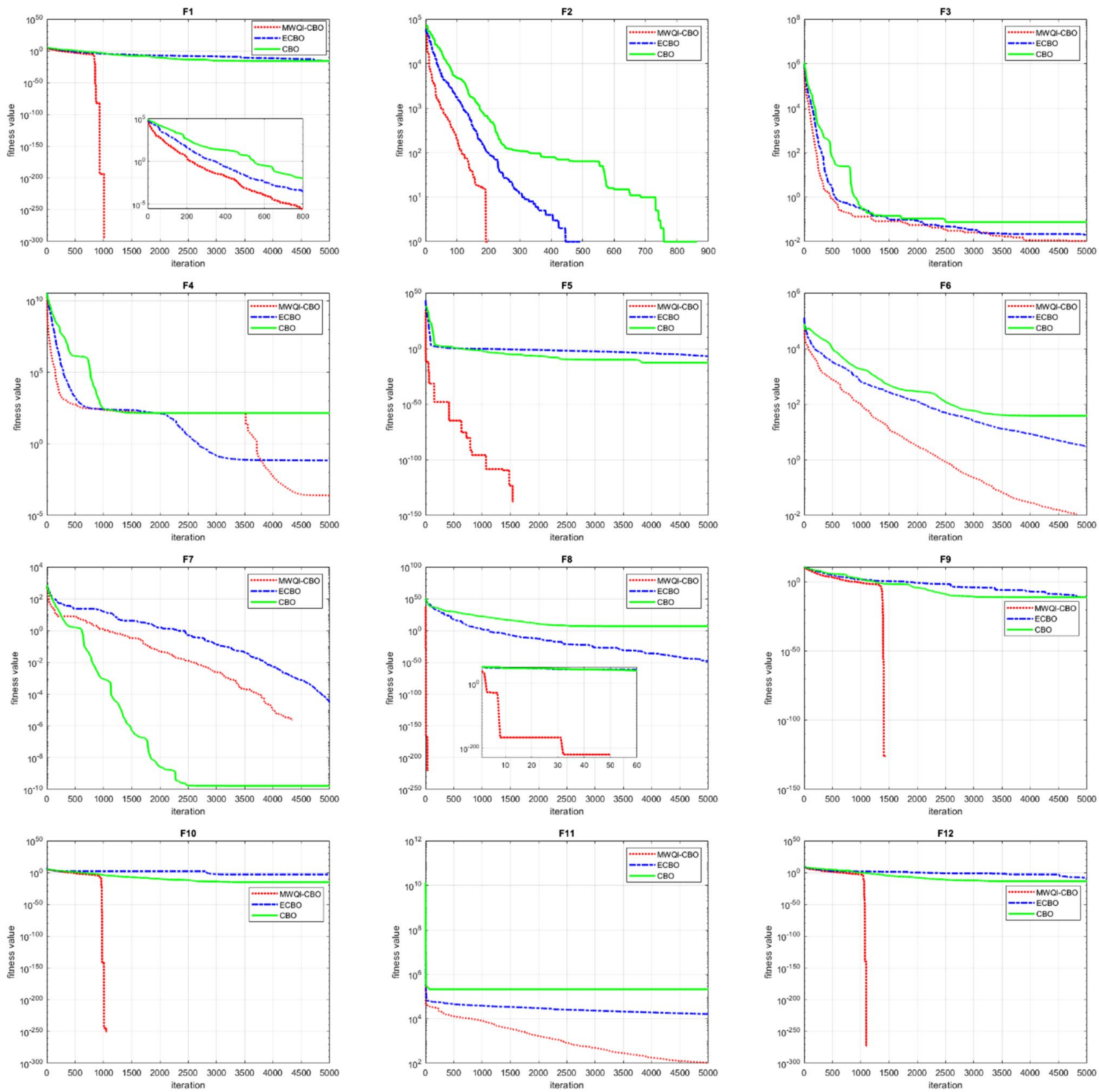


Fig. 1 Convergence curves for mathematical benchmark functions

clearly for f_{13}, f_{15}–f_{18}, and f_{24}. For f_{21}–f_{23}, MWQI-CBO has a slower convergence rate than ECBO but as time continues, it converges to superior results. For f_{14}, in spite of CBO's faster convergence in early iterations, MWQI-CBO can finally reach the global optima which demonstrates efficient search of the proposed algorithm. On the whole, these plots indicate that the convergence rate of standard CBO has been developed successfully due to the efficient improvement of its both exploration and exploitation processes.

4.2 Experiment 2: structural design problems

Sizing optimization of truss and frame structures are frequent structural design problems. Here, five benchmark examples are provided to demonstrate the effectiveness, robustness, and efficiency of the proposed method. These problems are subjected to various constraints such as displacements, stress, buckling, and natural frequencies. To reduce statistical errors, each test is repeated 20 times

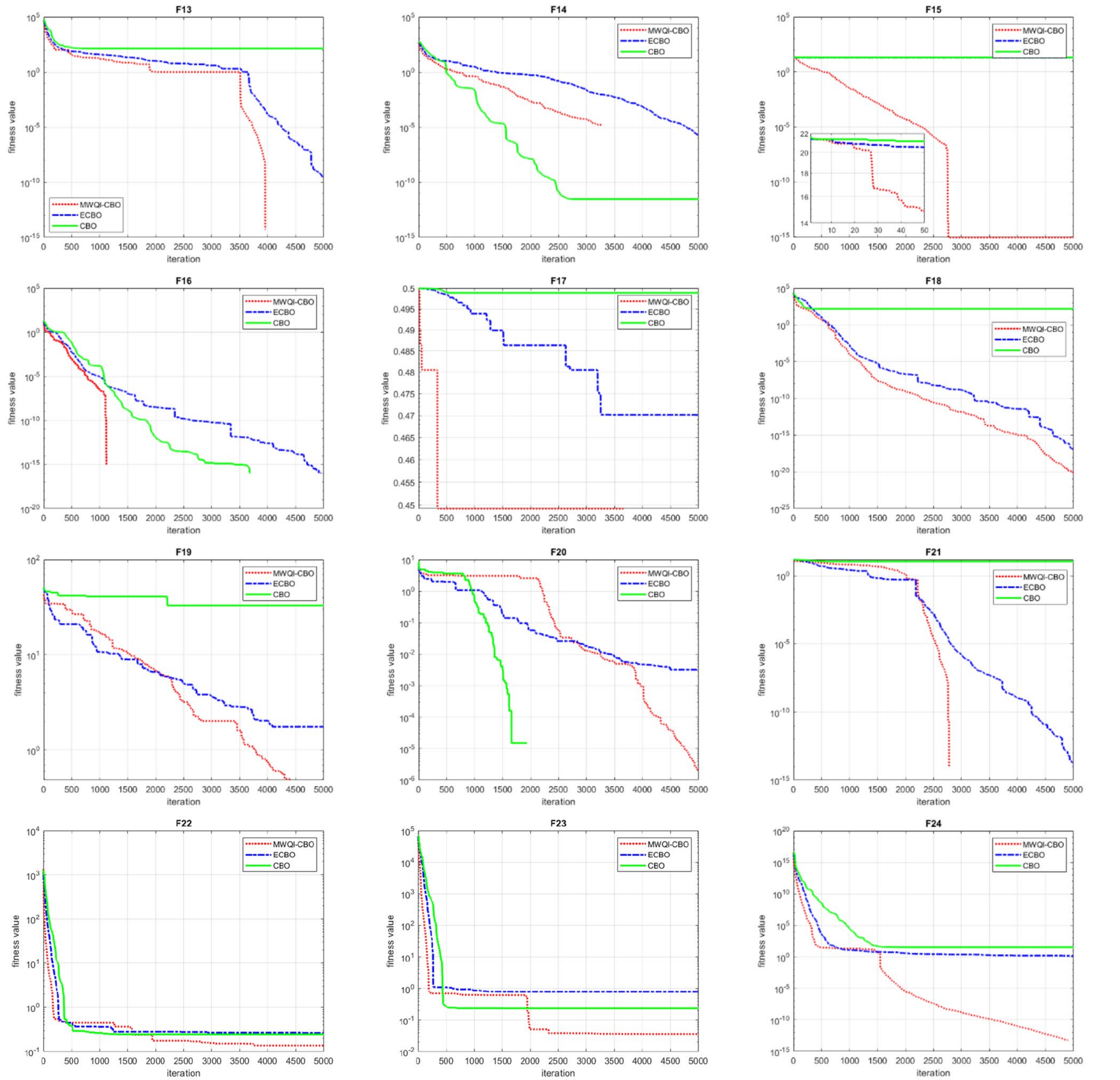


Fig. 1 (continued)

independently. The algorithms are coded in MATLAB and the structures are analyzed using the direct stiffness method by our own codes.

4.2.1 A 200-bar planar truss problem

The first structural optimization problem is the optimal design of a 200-bar planar truss schematized in Fig. 2. Due to the symmetry, the elements are divided into 29 groups. The modulus of elasticity is 210 GPa and the material density is 7860 kg/m^3 for all elements. The minimum cross-sectional area of all members is 0.1 cm^2. Non-structural masses of 100 kg are attached to the upper nodes. The first three natural frequencies of the structure must satisfy the following limitations: $f_1 \geq 5$ Hz, $f_2 \geq 10$ Hz, and $f_3 \geq 15$ Hz.

Optimal structures found by LCA-Tie-2 (league championship algorithm with tie concept) [28], ISOS (improved symbiotic organisms search) [29], differential evolution (DE) [30], AHEFA (adaptive hybrid evolutionary firefly algorithm) [30], CBO [31], ECBO [31], and MWQI-CBO

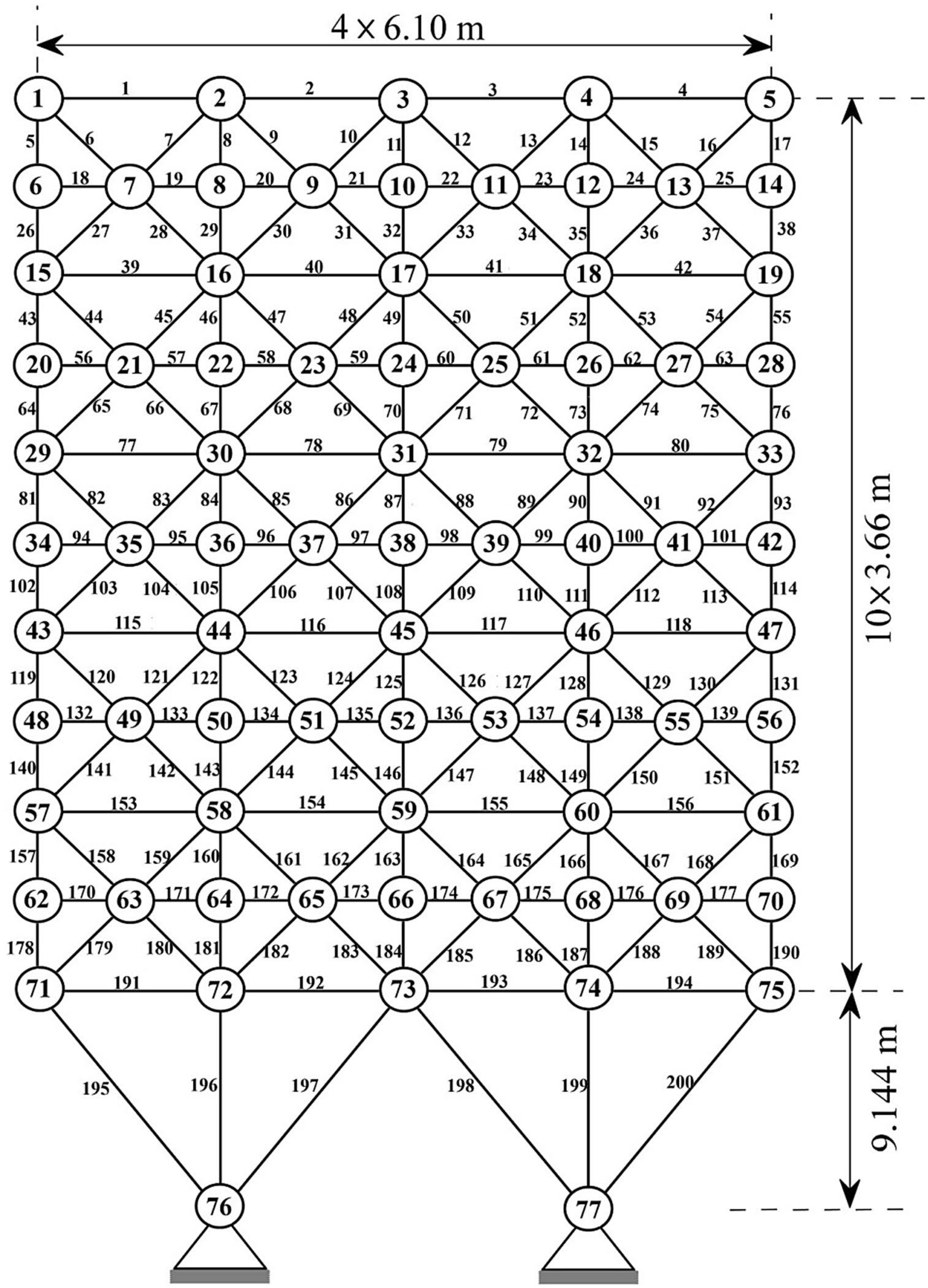


Fig. 2 Schematic of the 200-bar planar truss

are compared in Table 5. It can be seen that the lightest design (i.e., 2157.06 kg) is obtained by the MWQI-CBO. The mean of the independent runs for the proposed method is 2159.88 kg which is less than those of all other methods. Table 6 reports the natural frequencies of the optimized structures and it is clear that none of the frequency constraints are violated. Figure 3 shows the convergence curves of the best results found by CBO, ECBO, and MWQI-CBO. The MWQI-CBO converges to the optimum solution after 15,060 analyses. The CBO and ECBO get the optimal solution after 10,500 and 14,700 analyses, respectively. It should be mentioned that the proposed algorithm achieved the best designs of CBO and ECBO after 8240 and 13,680 analyses, respectively.

4.2.2 The 3-bay 15-story frame problem

The configuration, applied loads and the numbering of member groups for the 3-bay 15-story frame is shown in Fig. 4. This frame consists of 64 joints and 105 members. The modulus of elasticity is 29,000 ksi (200 GPa) and the yield stress is 36 ksi (248.2 MPa) for all members. The effective length

Table 5 Performance comparison for the 200-bar planar truss problem

Element group	Areas (cm^2)						
	LCA-Tie-2 [28]	ISOS [29]	DE [30]	AHEFA [30]	CBO [31]	ECBO [31]	MWQI-CBO
1	0.30891160	0.3072	0.3035	0.2993	0.3059	0.2993	0.2966
2	0.48871045	0.5075	0.4528	0.4508	0.4476	0.4497	0.4657
3	0.10162941	0.1001	0.1000	0.1001	0.1000	0.1000	0.1008
4	0.10657586	0.1000	0.1000	0.1000	0.1001	0.1	0.1002
5	0.54794212	0.5893	0.5162	0.5123	0.4944	0.5137	0.5077
6	0.81394811	0.8328	0.8203	0.8205	0.8369	0.7914	0.8253
7	0.11532799	0.1431	0.1004	0.1011	0.1001	0.1013	0.1001
8	1.29042334	1.3600	1.4393	1.4156	1.5514	1.4129	1.4194
9	0.11282050	0.1039	0.1003	0.1000	0.1000	0.1019	0.1002
10	1.56294014	1.5114	1.5918	1.5742	1.5286	1.6460	1.6222
11	1.14548904	1.3568	1.1641	1.1597	1.1547	1.1532	1.1746
12	0.18455251	0.1024	0.1319	0.1338	0.1000	0.1000	0.1013
13	2.92990485	2.9024	2.9561	2.9672	2.9980	3.1850	2.9609
14	0.11534915	0.1000	0.1003	0.1000	0.1017	0.1034	0.1006
15	3.29811115	3.4120	3.2491	3.2722	3.2475	3.3126	3.2534
16	1.60489863	1.4819	1.5949	1.5762	1.5213	1.5920	1.5706
17	0.29433408	0.2587	0.2525	0.2562	0.3996	0.2238	0.2417
18	5.25387865	4.8291	5.1567	5.0956	4.7557	5.1227	5.154
19	0.10219255	0.1499	0.1004	0.1001	0.1002	0.1050	0.1065
20	5.44086222	5.5090	5.4938	5.4546	5.1359	5.3707	5.46
21	2.02955602	2.2221	2.1094	2.0933	2.1181	2.0645	2.1291
22	0.57323199	0.6113	0.6731	0.6737	0.9200	0.5443	0.656
23	7.47936823	7.3398	7.6922	7.6498	7.3084	7.6497	7.4562
24	0.28990234	0.1559	0.1150	0.1178	0.1185	0.1000	0.1616
25	7.85261204	8.6301	8.0035	8.0682	7.6901	7.6754	8.0675
26	2.86794791	2.8245	2.7794	2.8025	3.0895	2.7178	2.8185
27	10.38474350	10.8563	10.5173	10.5040	10.6462	10.8141	10.4169
28	21.59152982	20.9142	21.2292	21.2935	20.7190	21.6349	21.3471
29	10.25871058	10.5305	10.7286	10.7410	11.7463	10.3520	10.4155
Best weight (kg)	2159.96	2169.4590	2160.7747	2160.7445	2161.15	2158.08	2157.06
Average optimized weight (kg)	2168.21	2244.6372	2162.2495	2161.0393	2447.52	2159.93	2159.88
Standard deviation on average weight (kg)	9.51	43.4808	3.0003	0.1783	301.29	1.57	2.94

Table 6 Natural frequencies (Hz) evaluated at the optimum designs of the 200-bar planar truss problem

Frequency number	Natural frequencies (Hz)						
	LCA-Tie-2 [28]	ISOS [29]	DE [30]	AHEFA [30]	CBO [31]	ECBO [31]	MWQI-CBO
1	5.000015	5.0000	5.0000	5.0000	5.000	5.000	5.000
2	12.363073	12.4477	12.2301	12.1821	12.221	12.189	12.179
3	15.173504	15.2332	15.0277	15.0160	15.088	15.048	15.058
4	16.728441	N/A	16.7054	16.6837	16.759	16.643	16.673
5	21.576253	N/A	21.4238	21.3547	21.419	21.342	21.365
6	21.688359	N/A	21.4435	21.4168	21.501	21.382	21.520

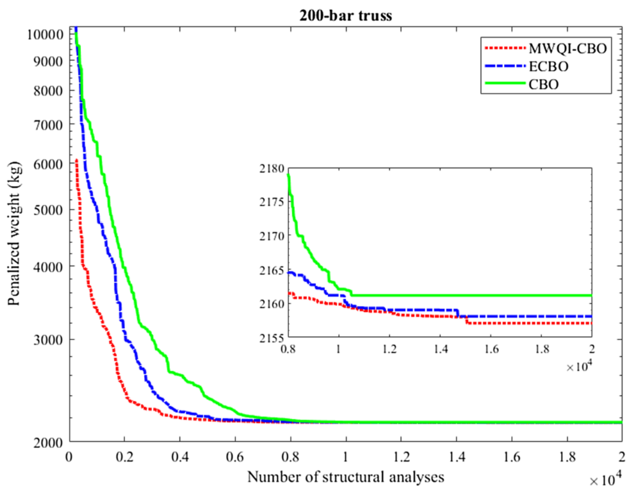


Fig. 3 Convergence curves for the 200-bar planar truss problem

factors of the members are calculated as $k_x \geq 0$ for a sway-permitted frame and the out-of-plane effective length factor is specified as $k_y = 1.0$. Each column is considered as non-braced along its length, and the non-braced length for each beam member is specified as one-fifth of the span length. Limitations on displacement and strength are imposed according to the provisions of the AISC [32] as follows:

(a) Maximum lateral displacement:

$$\frac{\Delta_{\mathrm{T}}}{H} - R \leq 0, \tag{17}$$

where Δ_T is the maximum lateral displacement; H is the height of the frame structure; and R is the maximum drift index which is equal to 1/300.

(b) The inter-story displacements:

$$\frac{d_i}{h_i} - R_{\mathrm{I}} \leq 0, \quad i = 1, 2, \ldots, ns, \tag{18}$$

where d_i is the inter-story drift; h_i is the story height of the ith floor; ns is the total number of stories; R_{I} is the inter-story drift index (1/300).

(c) Strength constraints:

$$\begin{cases} \frac{P_{\mathrm{u}}}{2\varphi_{\mathrm{c}}P_{\mathrm{n}}} + \frac{M_{\mathrm{u}}}{\varphi_{\mathrm{b}}M_{\mathrm{n}}} - 1 \leq 0, & \text{for} \frac{P_{\mathrm{u}}}{\varphi_{\mathrm{c}}P_{\mathrm{n}}} < 0.2 \\ \frac{P_{\mathrm{u}}}{\varphi_{\mathrm{c}}P_{\mathrm{n}}} + \frac{8M_{\mathrm{u}}}{9\varphi_{\mathrm{b}}M_{\mathrm{n}}} - 1 \leq 0, & \text{for} \frac{P_{\mathrm{u}}}{\varphi_{\mathrm{c}}P_{\mathrm{n}}} \geq 0.2 \end{cases}, \tag{19}$$

where P_{u} is the required strength (tension or compression); P_{n} is the nominal axial strength (tension or compression); φ_{c} is the resistance factor ($\varphi_{\mathrm{c}}=0.9$ for tension, $\varphi_{\mathrm{c}}=0.85$ for compression); M_{u} is the required flexural strengths; M_{n} is the nominal flexural strengths; φ_{b} denotes the flexural resistance reduction factor ($\varphi_{\mathrm{b}}=0.90$).

The nominal tensile strength for yielding in the gross section is calculated by:

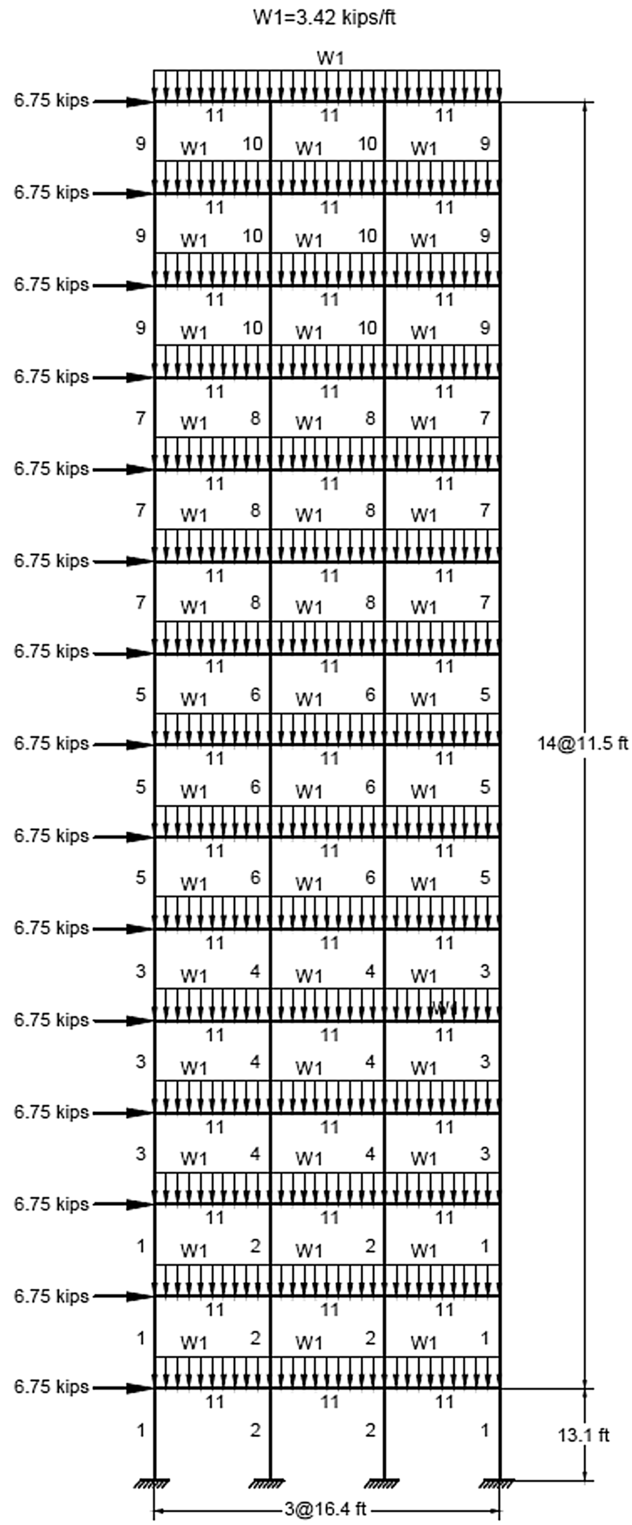


Fig. 4 Schematic of the 3-bay 15-story frame

$$P_n = A_g \,.\, F_y. \tag{20}$$

The nominal compressive strength of a member is computed as:

Table 7 Performance comparison for the 3-bay 15-story frame problem

Element group	Optimal *W*-shaped sections						
	DSOS [34]	CS [35]	TLBO [35]	WEO [35]	CBO [23]	ECBO [23]	MWQI-CBO
1	W16×100	W14×109	W12×96	W21×111	W24×104	W14×99	W14×90
2	W32×152	W27×161	W27×161	W27×146	W40×167	W27×161	W36×170
3	W12×79	W27×84	W27×84	W30×90	W27×84	W27×84	W27×84
4	W27×114	W24×104	W24×104	W21×101	W27×114	W24×104	W24×104
5	W21×93	W14×61	W10×68	W24×68	W21×68	W14×61	W14×61
6	W12×79	W30×90	W30×90	W27×84	W30×90	W30×90	W30×90
7	W21×55	W14×48	W8×48	W18×55	W8×48	W14×48	W14×48
8	W14×61	W21×68	W24×68	W10×60	W21×68	W14×61	W14×61
9	W14×22	W6×25	W8×28	W16×36	W14×34	W14×30	W14×34
10	W14×43	W14×43	W10×39	W18×35	W8×35	W12×40	W8×35
11	W21×48	W21×44	W21×50	W14×48	W21×50	W21×44	W21×44
Best weight (lb)	91,248	87,469	87,735	87,745	93,795	86,986	86,917
Average optimized weight (lb)	N/A	99,674	95,206	94,912	98,738	88,410	88,353
Standard deviation on average weight (lb)	N/A	24,308	11,346	18,101	N/A	N/A	1,948

$$P_{\mathrm{n}} = A_{\mathrm{g}} \cdot F_{\mathrm{cr}}, \tag{21}$$

where

$$\begin{cases} F_{\mathrm{cr}} = (0.658^{\lambda_{\mathrm{c}}^2})F_y, & \text{for} \quad \lambda_{\mathrm{c}} \le 1.5 \\ F_{\mathrm{cr}} = \left(\frac{0.877}{\lambda_{\mathrm{c}}^2}\right)F_y, & \text{for} \quad \lambda_{\mathrm{c}} > 1.5 \end{cases}, \tag{22}$$

$$\lambda_{\mathrm{c}} = \frac{kl}{r\pi}\sqrt{\frac{F_y}{E}}, \tag{23}$$

where A_{g} is the cross-sectional area of a member, and k is the effective length factor that is calculated by (Dumonteil [33]):

$$k = \sqrt{\frac{1.6G_A G_B + 4.0(G_A + G_B) + 7.5}{G_A + G_B + 7.5}}, \tag{24}$$

where G_A and G_B are stiffness ratios of columns and girders at the two end joints A and B of the column section, respectively.

In addition, in this example, the sway of the top story is limited to 9.25 in (23.5 cm).

The results found by discrete symbiotic organisms search (DSOS) [34], cuckoo search (CS) [35], teaching–learning-based optimization (TLBO) [35], water evaporation optimization (WEO) [35], CBO [23], ECBO [23], and MWQI-CBO algorithms are summarized in Table 7. MWQI-CBO achieves the lightest design (i.e., 86,917 lb). The best design obtained by DSOS, CS, TLBO, WEO, CBO, and ECBO are 91,248 lb, 87,469 lb, 87,735 lb, 87,745 lb, 93,795 lb, and 86,986 lb, respectively. The proposed method has better performance in terms of the average optimized weight and standard deviation on average weight which are 88,353 lb, and 1948 lb, respectively. Convergence histories are depicted in Fig. 5. The required number of structural analyses to achieve the best design by CBO, ECBO, and MWQI-CBO are 9520, 9000, and 14,420 analyses, respectively. MWQI-CBO found the best designs of CBO after 6420 analyses. Figure 6 demonstrates the existing stress ratios and inter-story drifts for the best designs of proposed algorithm. The

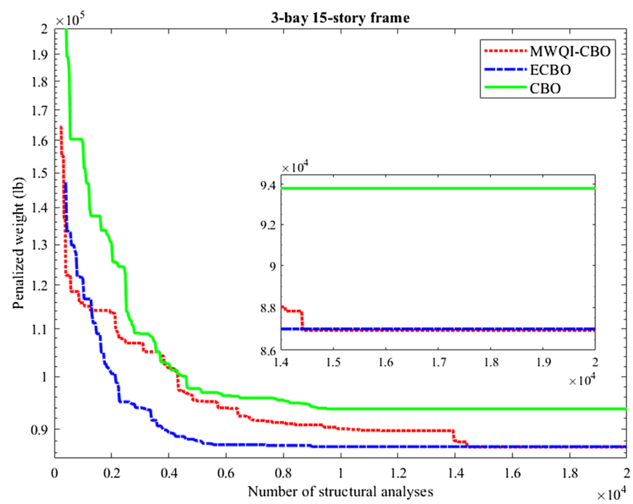


Fig. 5 Convergence curves for the 3-bay 15-story frame problem

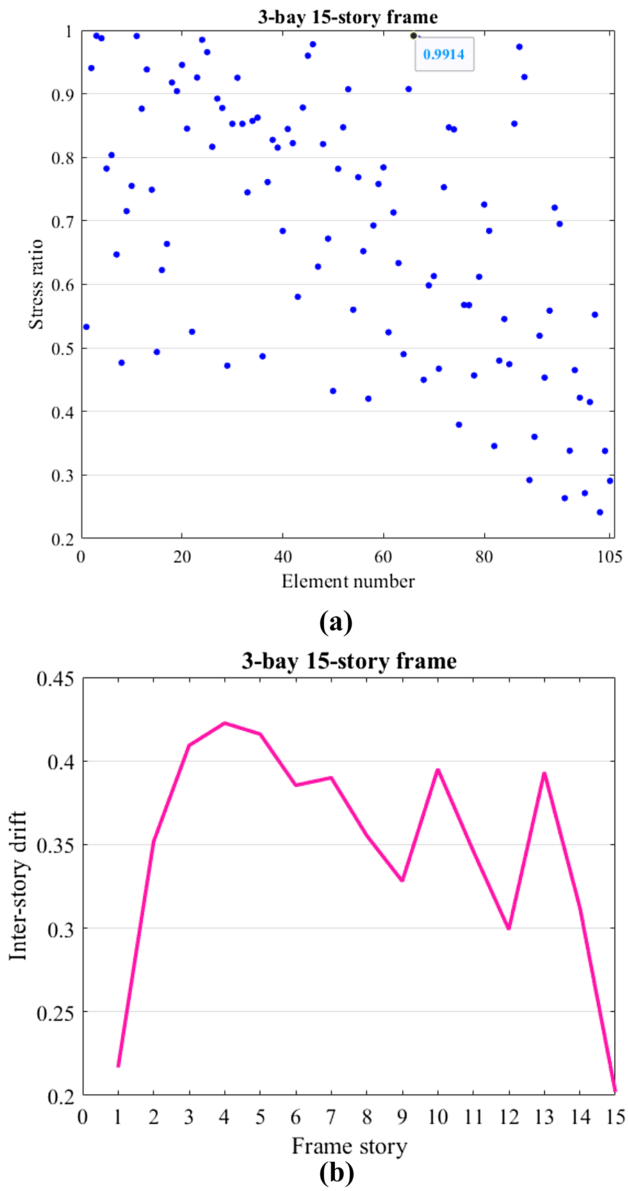


Fig. 6 Constraint margins for the best design obtained by MWQI-CBO for the 3-bay 15-story frame problem: **a** element stress ratio and **b** inter-story drift

maximum stress ratio for the best designs of the MWQI-CBO is 99.14%.

4.2.3 The 3-bay 24-story frame problem

The third structural optimization problem is the size optimization of a 3-bay 24-story frame depicted in Fig. 7. The material has a modulus of elasticity equal to $E = 29.732$ Msi (205 GPa) and a yield stress of $f_y = 33.4$ ksi (230.3 MPa). It consists of 168 members that are collected in 20 groups (16 column groups and 4 beam groups). Each of the four beam element groups is chosen from all 267 W-shapes, while the

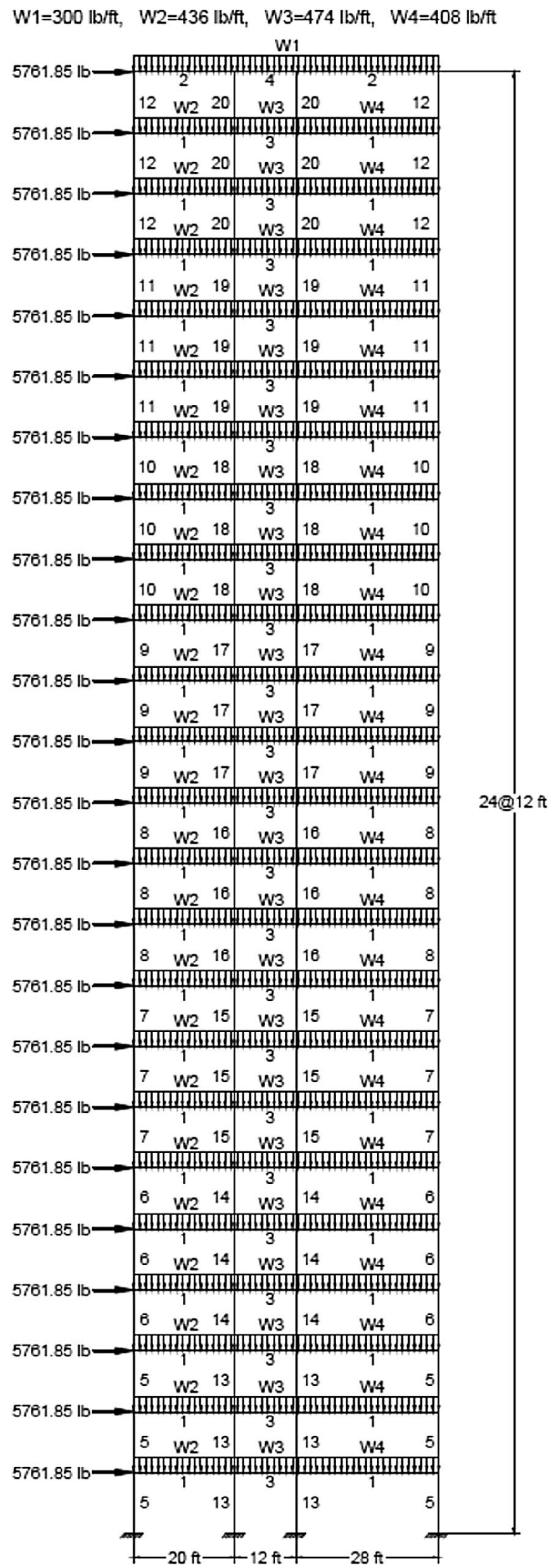


Fig. 7 Schematic of the 3-bay 24-story frame

Table 8 Performance comparison for the 3-bay 24-story frame problem

Element group	Optimal *W*-shaped sections						
	SBO [36]	CS [35]	TLBO [35]	WEO [35]	CBO [23]	ECBO [23]	MWQI-CBO
1	W30×90	W14×176	W14×145	W14×159	W14×132	W14×145	W14×145
2	W8×18	W14×109	W14×90	W14×120	W14×120	W14×132	W14×109
3	W21×48	W14×99	W14×99	W14×90	W14×145	W14×99	W14×109
4	W6×8.5	W14×99	W14×74	W14×82	W14×82	W14×90	W14×82
5	W14×152	W14×74	W14×53	W14×48	W14×61	W14×74	W14×68
6	W14×120	W14×38	W14×90	W14×48	W14×43	W14×38	W14×38
7	W14×109	W14×30	W14×30	W14×43	W14×38	W14×38	W14×34
8	W14×74	W14×22	W14×22	W14×22	W14×22	W14×22	W14×22
9	W14×82	W14×90	W14×90	W14×61	W14×99	W14×99	W14×99
10	W14×43	W14×109	W14×120	W14×68	W14×109	W14×99	W14×109
11	W14×34	W14×99	W14×99	W14×74	W14×82	W14×99	W14×99
12	W12×19	W14×82	W14×99	W14×68	W14×90	W14×82	W14×90
13	W14×109	W14×68	W14×90	W14×145	W14×74	W14×68	W14×74
14	W14×109	W14×61	W14×38	W14×48	W14×61	W14×61	W14×68
15	W14×99	W14×38	W14×43	W14×26	W14×30	W14×30	W14×34
16	W14×99	W14×22	W14×22	W14×22	W14×22	W14×22	W14×22
17	W14×68	W30×90	W30×90	W33×130	W27×102	W30×90	W30×90
18	W14×61	W6×15	W6×15	W8×18	W8×18	W6×15	W6×15
19	W14×34	W24×55	W24×68	W21×44	W24×55	W24×55	W24×55
20	W14×22	W6×8.5	W6×8.5	W6×8.5	W6×8.5	W6×8.5	W6×8.5
Best weight (lb)	202,422	202,482	202,626	203,058	215,874	201,618	201,906
Average optimized weight (lb)	209,560	230,342	218,853	222,880	225,071	209,644	206,025
Standard deviation on average weight (lb)	7052	65,703	37,979	66,839	N/A	N/A	3202

16 column element groups are limited to W14 sections. The effective length factors of the members are calculated as $k_x \geq 0$ for a sway-permitted frame and the out-of-plane effective length factor is specified as $k_y = 1.0$. All columns and beams are considered as non-braced along their lengths. The frame is designed following the LRFD specification and uses an inter-story drift displacement constraint similar to the previous problem AISC [32].

Table 8 presents the results obtained by school-based optimization (SBO) [36], cuckoo search (CS) [35], teaching–learning-based optimization (TLBO) [35], water evaporation optimization (WEO) [35], CBO [23], ECBO [23], and MWQI-CBO. The lightest design (i.e., 201,618 lb) is obtained by the ECBO. After that, the best design found by MWQI-CBO is better than those of the other methods (201,906 lb). The best weight found by SBO, CS, TLBO, WEO, and CBO are 202,422 lb, 202,482 lb, 202,626 lb, 203,058 lb, and 215,874 lb, respectively. The proposed method is the most robust optimizer, achieving the lowest average weight over the independent optimization runs. It also performs better than other algorithms in terms of standard deviation on average weight. The SBO, CS, TLBO, WEO, CBO, ECBO, and MWQI-CBO algorithms achieved the optimal solutions after 14,572, 18,760, 8760, 15,465, 8280, 15,360, and 12,760 structural analyses, respectively. The proposed method obtained the best design of CBO after 4540 structural analyses. Convergence history diagrams are depicted in Fig. 8. Element stress ratio and inter-story drift evaluated at the best design optimized by MWQI-CBO are shown in Fig. 9. The maximum stress ratio is 95.95% and the maximum inter-story drift is 47.92.

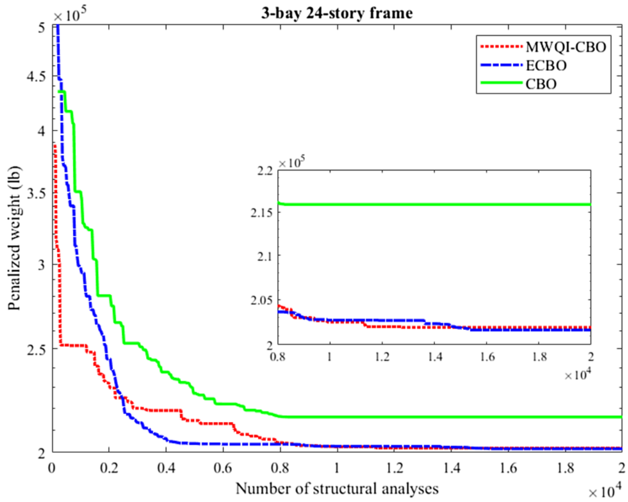


Fig. 8 Convergence curves for the 3-bay 24-story frame problem

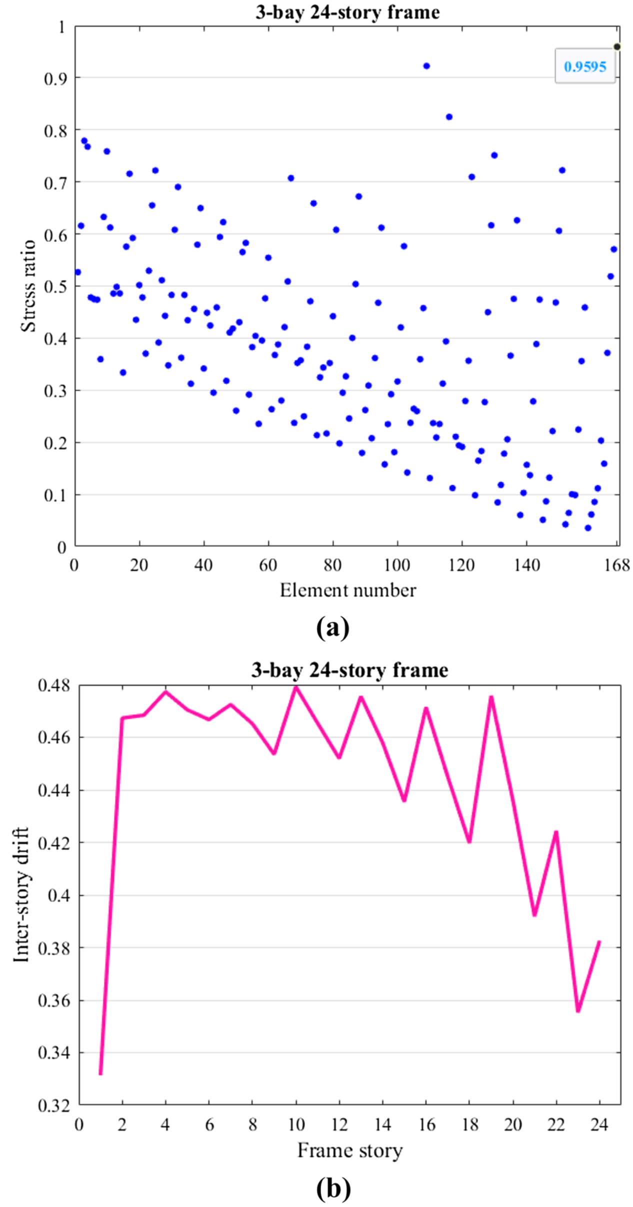


Fig. 9 Constraint margins for the best design obtained by MWQI-CBO for the 3-bay 24-story frame problem: **a** element stress ratio and **b** inter-story drift

4.2.4 The spatial 582-bar tower truss problem

The 582-bar tower truss is schematized in Fig. 10. The members are divided into 32 groups, because of structural symmetry. A single-load case is considered consisting of lateral loads of 1.12 kips (5.0 kN) applied in both x- and y-directions and vertical loads of − 6.74 kips (− 30 kN) applied in z-direction to all free nodes of the tower. Cross-sectional areas of elements are selected from a discrete list of W-shaped standard steel sections based on area and radii of gyration properties. Cross-sectional areas of elements can vary between 6.16 and 215 in^2 (i.e., between 39.74 and 1387.09 cm^2). Limitation on stress and stability of truss elements are imposed according to the provisions of AISC [37] as follows.

The allowable tensile stresses for tension members are calculated as:

$$\sigma_i^+ = 0.6F_y, \tag{25}$$

where F_y is the yield strength.

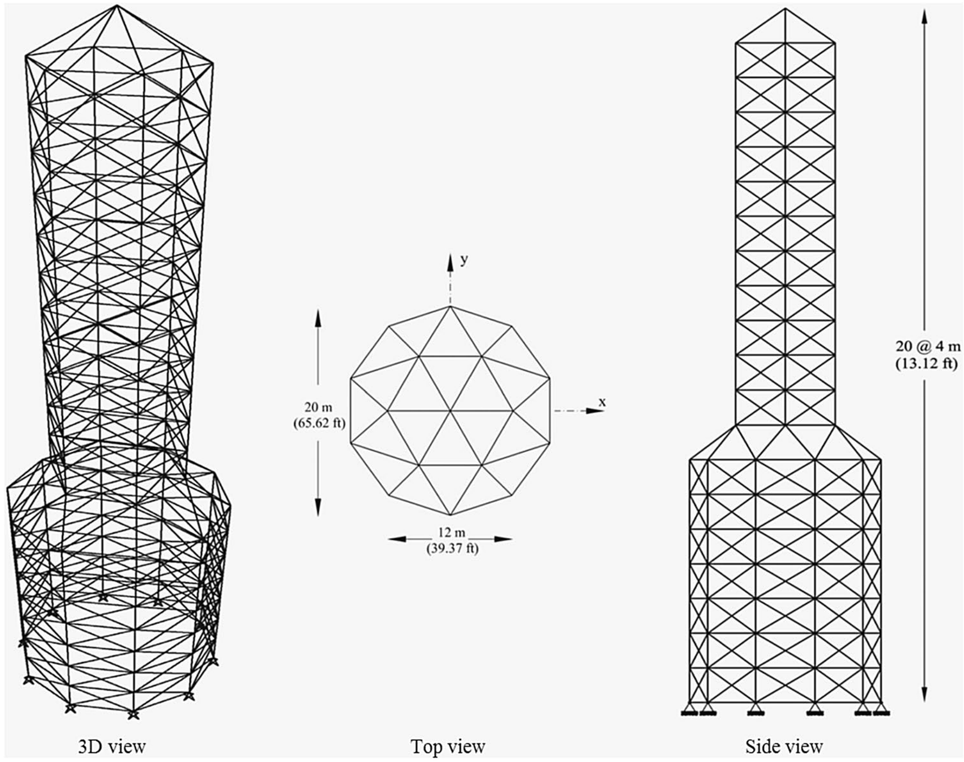


Fig. 10 Schematic of the 582-bar tower truss

The allowable stress limits for compression members are calculated depending on two possible failure modes of the members known as elastic and inelastic buckling. Therefore,

$$\sigma_i^- = \begin{cases} \left[\left(1 - \frac{\lambda_i^2}{2C_c^2}\right)F_y\right] / \left[\frac{5}{3} + \frac{3\lambda_i}{8C_c} - \frac{\lambda_i^3}{8C_c^3}\right] & \text{for} \quad \lambda_i < C \\ \frac{12\pi^2 E}{23\lambda_i^2} & \text{for} \quad \lambda_i \geq C_c \end{cases}, \tag{26}$$

where E is the modulus of elasticity; λ_i is the slenderness ratio $\left(\lambda_i = kl_i / r_i\right)$; C_c denotes the slenderness ratio dividing the elastic and inelastic buckling regions $C_c = \sqrt{2\pi^2 E / F_y}$; k is the effective length factor (k is set equal to 1 for all truss members); L_i is the member length; and r_i is the minimum radius of gyration.

The maximum slenderness ratio is limited to 300 for tension members, and it is recommended to be 200 for compression members. Moreover, nodal displacements in all coordinate directions must be less than ± 3.15 in (i.e., ± 8 cm) for this example.

Table 9 lists the optimal designs found by particle swarm optimization (PSO) [38], whale optimization algorithm (WOA) [39], enhanced whale optimization algorithm (EWOA) [39], CBO [23], ECBO [23], and MWQI-CBO. The proposed algorithm obtained the lightest design compared to other methods that is 1,295,562 in^3. Moreover, the average optimized volume and the standard deviation on average volume of MWQI-CBO (1,305,095 in^3 and 5320 in^3) are less than those of all other methods. The best designs found by the PSO, WOA, EWOA, CBO, and ECBO are

Table 9 Performance comparison for the spatial 582-bar tower truss problem

Element group	Optimal *W*-shaped sections					
	PSO [38]	WOA [39]	EWOA [39]	CBO [23]	ECBO [23]	MWQI-CBO
1	W8×21	W8×21	W8×21	W8×21	W8×21	W8×21
2	W12×79	W14×90	W14×90	W14×82	W14×90	W14×90
3	W8×24	W8×24	W8×24	W8×28	W8×24	W8×24
4	W10×60	W14×61	W10×60	W12×50	W14×61	W12×58
5	W8×24	W8×24	W8×24	W8×24	W8×24	W8×24
6	W8×21	W8×21	W8×21	W8×21	W8×21	W8×21
7	W8×48	W10×49	W14×48	W12×53	W10×49	W10×45
8	W8×24	W8×24	W8×24	W12×26	W8×24	W8×24
9	W8×21	W8×21	W8×21	W8×21	W8×21	W8×21
10	W10×45	W10×39	W10×49	W14×48	W14×43	W10×54
11	W8×24	W8×24	W8×24	W8×24	W8×24	W8×24
12	W10×68	W12×72	W16×67	W14×61	W12×72	W12×65
13	W14×74	W14×74	W18×76	W14×82	W12×72	W14×74
14	W8×48	W12×50	W10×49	W12×50	W10×54	W10×49
15	W18×76	W10×77	W18×76	W14×74	W12×65	W14×74
16	W8×31	W8×31	W8×31	W8×40	W8×31	W8×31
17	W8×21	W10×49	W14×61	W12×53	W10×60	W14×61
18	W16×67	W8×24	W8×24	W6×25	W8×24	W8×24
19	W8×24	W8×21	W8×21	W8×21	W8×21	W8×21
20	W8×21	W14×48	W14×34	W8×40	W14×43	W8×40
21	W8×40	W6×25	W8×24	W8×24	W8×24	W8×24
22	W8×24	W10×22	W8×21	W8×21	W8×21	W8×21
23	W8×21	W8×21	W8×21	W12×26	W8×21	W8×28
24	W10×22	W8×24	W8×24	W12×26	W8×24	W8×24
25	W8×24	W8×21	W8×21	W10×22	W8×21	W8×21
26	W8×21	W10×22	W10×22	W10×22	W8×21	W8×21
27	W8×21	W8×24	W8×24	W6×25	W8×24	W8×24
28	W8×24	W8×21	W8×21	W8×21	W8×21	W8×21
29	W16×21	W8×21	W8×21	W8×21	W8×21	W8×21
30	W8×21	W8×24	W8×24	W8×24	W8×24	W8×24
31	W8×24	W8×21	W8×21	W8×21	W8×21	W8×21
32	W8×24	W8×28	W8×24	W6×25	W8×24	W8×24
Best volume (in^3)	1,366,674	1,302,038	1,295,738	1,334,994	1,296,776	1,295,562
Average optimized volume (in^3)	1,371,667	1,349,290	1,310,836	1,345,429	1,306,728	1,305,095
Standard deviation on average volume (in^3)	N/A	N/A	N/A	9116	7536	5320

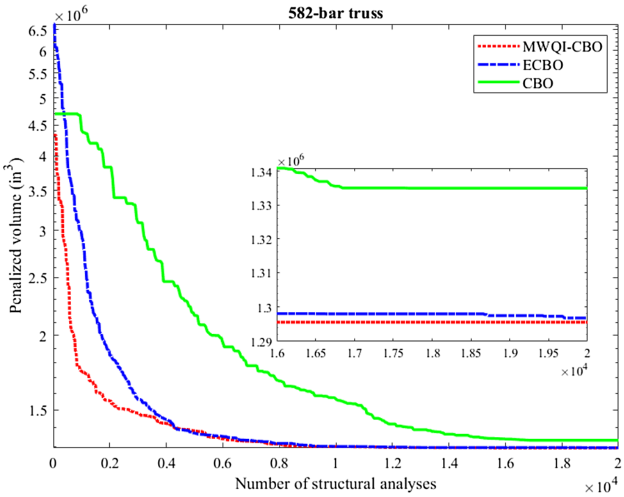
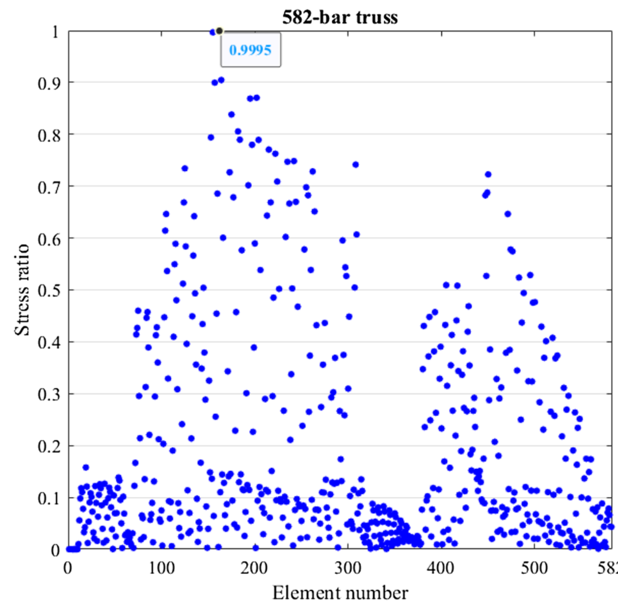
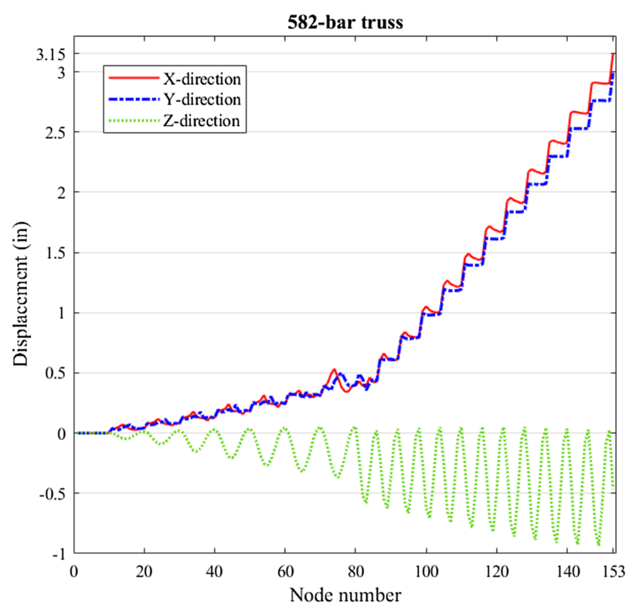


Fig. 11 Convergence curves for the 582-bar tower truss problem



(a)



(b)

Fig. 12 Constraint margins for the best design obtained by MWQI-CBO algorithm for the 582-bar tower truss problem: **a** element stress ratio and **b** nodal displacements

1,366,674 in^3, 1,302,038 in^3, 1,295,738 in^3, 1,334,994 in^3, and 1,296,776 in^3, respectively. Convergence histories are demonstrated in Fig. 11. It should be noted that the proposed method requires 15,560 structural analyses to find the optimum solution while WOA, EWOA, CBO, and ECBO require 18,840, 19,300, 17,700, and 19,700 structural analyses, respectively. Stress ratios and nodal displacements in all directions evaluated for the best design achieved by MWQI-CBO are shown Fig. 12. The maximum stress ratio and the maximum nodal displacement are 99.95% and 3.1488 in, respectively.

4.2.5 The 600-bar single-layer dome truss problem

The sizing optimization of a 600-bar single-layer dome structure schematized in Fig. 13 is the last test case. The entire structure is composed of 216 nodes and 600 elements. Figure 14 shows a substructure in more detail for nodal numbering. The cross-sectional area of each of the member in this substructure is considered to be an independent variable. Therefore, this is a size optimization problem with 25 variables. Table 10 presents the coordinates of the nodes in the Cartesian coordinate system. The elastic modulus is

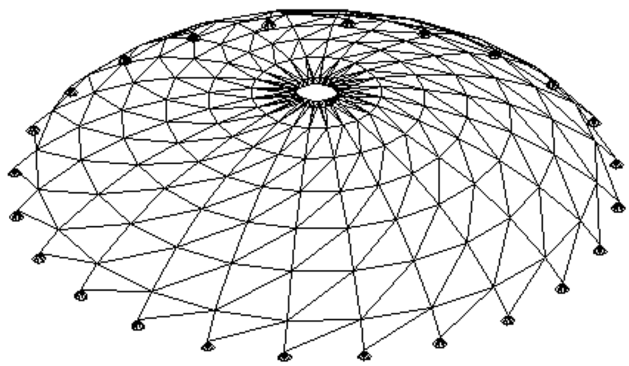


Fig. 13 Schematic of the 600-bar single-layer dome truss

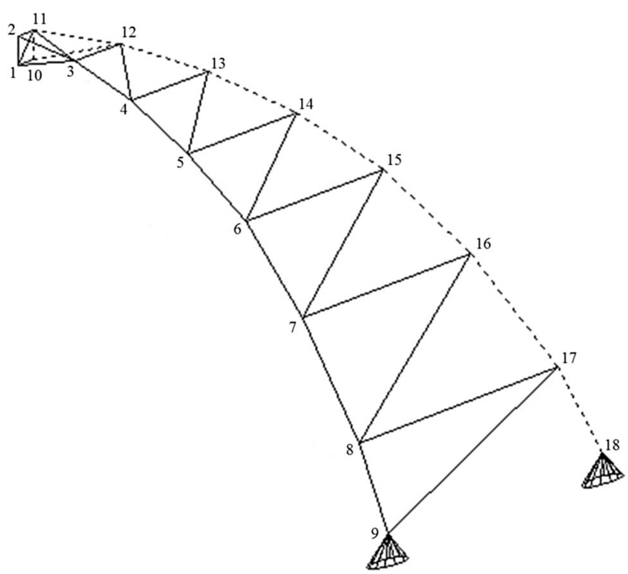


Fig. 14 Details of a substructure of the 600-bar single-layer dome truss

Table 10 Coordinates of the nodes for the 600-bar single-layer dome truss problem

Node number	Coordinates (*x*, *y*, *z*)	Node number	Coordinates (*x*, *y*, *z*)
1	(1,0,7)	10	(0.9659,0.2588,7)
2	(1,0,7.5)	11	(0.9659,0.2588,7.5)
3	(3,0,7.25)	12	(2.8978,0.7765,7.25)
4	(5,0,6.75)	13	(4.8296,1.2941,6.75)
5	(7,0,6)	14	(6.7615,1.8117,6)
6	(9,0,5)	15	(8.6933,2.3294,5)
7	(11,0,3.5)	16	(10.6251,2.8471,3.5)
8	(13,0,1.5)	17	(12.5570,3.3646,1.5)
9	(14,0,0)	18	(13.5230,3.6235,0)

200 GPa and the material density is 7850 kg/m^3 for all elements. Non-structural masses of 100 kg are attached to each free node. The minimum and maximum admissible cross-sectional areas are 1 cm^2 and 100 cm^2, respectively. The first frequency is required to be $f_1 \geq 5$ Hz and the third frequency is required to be $f_3 \geq 7$ Hz.

The optimized designs found by democratic particle swarm optimization (DPSO) [40], harmony search (HS) [41], cyclical parthenogenesis algorithm (CPA) [41], CBO [23], ECBO [23], and MWQI-CBO are compared in Table 11. The MWQI-CBO obtained the lightest design which is 6147.96 kg while it is 6344.55 kg for DPSO, 6357.59 kg for HS, 6336.85 kg for CPA, 6182.01 kg for CBO, and 6171.51 kg for ECBO. The average optimized weight and the standard deviation on average weight of the ECBO is less than those of all other methods. Frequency constraints are satisfied by all methods (see Table 12). Figure 15 compares the convergence curves of the best results obtained by CBO, ECBO, and MWQI-CBO. The MWQI-CBO requires 16,560 structural analyses to find the optimum solution while CBO and ECBO require 17,940, and 19,020 structural analyses, respectively.

5 Concluding remarks

In this work, a new variant of CBO (MWQI-CBO) is proposed considering Morlet wavelet mutation and quadratic interpolation. The CBO generally has a week performance in exploration and can easily be trapped in local optima. MW is a mutation operator that ensures a very good search in the search space and QI is a local operator near the so-far-best agent. In all 24 mathematical experiments, MWQI-CBO is compared with G-PSO, PSO-DTT, PSO-DFCM, CBO, and ECBO methods. The results are also analyzed by Wilcoxon's rank sum test. The obtained statistical results show that the proposed algorithm is very competitive and often superior compared to the algorithms used in the experiments. To illustrate the efficiency and applicability of the MWQI-CBO, five structural design problems are also studied and the proposed algorithm demonstrates better performance than the considered metaheuristics. To sum up, comprehensive experiments have validated that MWQI-CBO has a good global search capacity and performs effectively and reliably when compared to standard CBO, ECBO, and some state-of-art metaheuristics.

Table 11 Performance comparison for the 600-bar single-layer dome truss problem

Element number (nodes)	Areas (cm^2)					
	DPSO [40]	HS [41]	CPA [41]	CBO [23]	ECBO [23]	MWQI-CBO
1 (1–2)	1.365	1.439	1.155	1.2404	1.4305	1.1414
2 (1–3)	1.391	1.425	1.304	1.3797	1.3941	1.1930
3 (1–10)	5.686	4.942	4.178	5.2597	5.5293	4.9972
4 (1–11)	1.511	1.677	1.335	1.2658	1.0469	1.3359
5 (2–3)	17.711	18.331	18.375	17.2255	16.9642	16.4705
6 (2–11)	36.266	36.074	39.914	38.2991	35.1892	40.9204
7(3–4)	13.263	13.407	13.609	12.2234	12.2171	12.5481
8 (3–11)	16.919	17.066	16.470	15.4712	16.7152	16.8270
9 (3–12)	13.333	13.122	14.108	11.1577	12.5999	12.3559
10 (4–5)	9.534	10.061	10.038	9.4636	9.5118	9.8049
11 (4–12)	9.884	9.827	9.514	8.8250	8.9977	8.8128
12 (4–13)	9.547	9.388	9.329	9.1021	9.4397	8.9853
13 (5–6)	7.866	7.083	6.938	6.8417	6.8864	7.4324
14 (5–13)	5.529	5.697	5.545	5.2882	4.2057	4.4777
15 (5–14)	7.007	7.139	6.763	6.7702	7.2651	6.7637
16 (6–7)	5.462	5.082	5.209	5.1402	6.1693	5.3079
17 (6–14)	3.853	3.295	3.842	5.1827	3.9768	3.7870
18 (6–15)	7.432	7.663	8.112	7.4781	8.3127	7.5167
19 (7–8)	4.261	4.100	4.252	4.5646	4.1451	4.3198
20 (7–15)	2.253	1.882	2.227	1.8617	2.4042	1.9381
21 (7–16)	4.337	4.725	4.582	4.8797	4.3038	4.8992
22 (8–9)	4.028	3.860	3.336	3.5065	3.2539	3.2783
23 (8–16)	1.954	2.280	1.725	2.4546	1.8273	1.8130
24 (8–17)	4.709	4.912	4.675	4.9128	4.8805	4.8722
25 (9–17)	1.410	1.502	1.673	1.2324	1.5276	1.9181
Best weight (kg)	6344.55	6357.59	6336.85	6182.01	6171.51	6147.96
Average optimized weight (kg)	6674.71	6631.48	6376.01	6226.37	6191.50	6215.29
Standard deviation on average weight (kg)	473.21	304.09	90.39	60.12	39.08	51.42

Table 12 Natural frequencies (Hz) evaluated at the optimum designs of the 600-bar single-layer dome truss problem

Frequency number	Natural frequencies (Hz)					
	DPSO [40]	HS [41]	CPA [41]	CBO [23]	ECBO [23]	MWQI-CBO
1	5.000	5.000	5.000	5.000	5.002	5.004
2	5.000	5.000	5.000	5.000	5.003	5.004
3	7.000	7.000	7.000	7.000	7.001	7.000
4	7.000	7.001	7.000	7.000	7.001	7.000
5	7.000	7.001	7.000	7.001	7.002	7.001

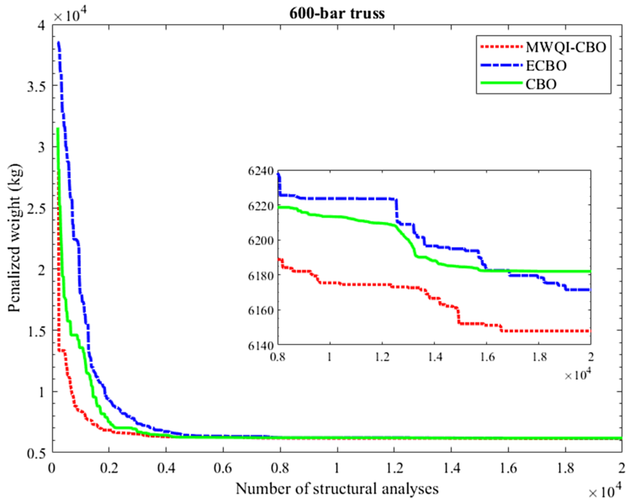


Fig. 15 Convergence curves for the 600-bar dome truss problem

Compliance with ethical standards

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Kaveh A (2017) Advances in metaheuristic algorithms for optimal design of structures, 2nd edn. Springer, Cham. https://doi.org/10.1007/978-3-319-05549-7
2. Artar M, Daloğlu AT (2016) Optimum weight design of steel space frames with semi-rigid connections using harmony search and genetic algorithms. Neural Comput Appl 29(11):1089–1100. https://doi.org/10.1007/s00521-016-2634-8
3. Techasen T, Wansasueb K, Panagant N, Pholdee N, Bureerat S (2018) Simultaneous topology, shape, and size optimization of trusses, taking account of uncertainties using multi-objective evolutionary algorithms. Eng Comput 35(2):721–740. https://doi.org/10.1007/s00366-018-0629-z
4. Maheri MR, Talezadeh M (2018) An enhanced imperialist competitive algorithm for optimum design of skeletal structures. Swarm Evolu Comput 40:24–36. https://doi.org/10.1016/j.swevo.2017.12.001
5. Millan-Paramo C, Filho JE (2019) Exporting water wave optimization concepts to modified simulated annealing algorithm for size optimization of truss structures with natural frequency constraints. Eng Comput. https://doi.org/10.1007/s00366-019-00854-6
6. Kaveh A, Kabir MZ, Bohlool M (2019) Optimum design of three-dimensional steel frames with prismatic and non-prismatic elements. Eng Comput 36(3):1011–1027. https://doi.org/10.1007/s00366-019-00746-9
7. Kumar S, Tejani GG, Pholdee N, Bureerat S (2020) Multi-objective modified heat transfer search for truss optimization. Eng Comput. https://doi.org/10.1007/s00366-020-01010-1
8. Goldberg DE (1989) Genetic algorithms in search, optimization, and machine learning. Choice Revi Online 27(02):27-0936, Addison Wesley. https://doi.org/10.5860/choice.27-0936
9. Ling SH, Leung FH (2006) An improved genetic algorithm with average-bound crossover and wavelet mutation operations. Soft Comput 11(1):7–31. https://doi.org/10.1007/s00500-006-0049-7
10. Ling S, Iu H, Chan K, Lam H, Yeung B, Leung F (2008) Hybrid particle swarm optimization with wavelet mutation and its industrial applications. IEEE Trans Syst Man Cybern Part B (Cybern) 38(3):743–763. https://doi.org/10.1109/tsmcb.2008.921005
11. Mondal S, Ghoshal SP, Kar R, Mandal D (2012) Differential evolution with wavelet mutation in digital finite impulse response filter design. J Optim Theory Appl 155(1):315–324. https://doi.org/10.1007/s10957-012-0028-3
12. Saha SK, Kar R, Mandal D, Ghoshal SP (2013) Gravitational search algorithm with wavelet mutation applied for optimal IIR band pass filter design. In: 2013 international conference on communication and signal processing. https://doi.org/10.1109/iccsp.2013.6577005
13. Jiang F, Xia H, Anh Tran Q, Minh Ha Q, Quang Tran N, Hu J (2017) A new binary hybrid particle swarm optimization with wavelet mutation. Knowl Based Syst 130:90–101. https://doi.org/10.1016/j.knosys.2017.03.032
14. Guo W, Liu T, Dai F, Xu P (2020) An improved whale optimization algorithm for forecasting water resources demand. Appl Soft Comput 86:105925. https://doi.org/10.1016/j.asoc.2019.105925
15. Pant M, Radha T, Singh VP (2007) A new particle swarm optimization with quadratic interpolation. In: International conference on computational intelligence and multimedia applications (ICCIMA 2007). https://doi.org/10.1109/iccima.2007.95
16. Deep K, Das KN (2008) Quadratic approximation based hybrid genetic algorithm for function optimization. Appl Math Comput 203(1):86–98. https://doi.org/10.1016/j.amc.2008.04.021
17. Sun Y, Wang X, Chen Y, Liu Z (2018) A modified whale optimization algorithm for large-scale global optimization problems. Expert Syst Appl 114:563–577. https://doi.org/10.1016/j.eswa.2018.08.027
18. Sun Y, Yang T, Liu Z (2019) A whale optimization algorithm based on quadratic interpolation for high-dimensional global optimization problems. Appl Soft Comput 85:105744. https://doi.org/10.1016/j.asoc.2019.105744
19. Chen X, Mei C, Xu B, Yu K, Huang X (2018) Quadratic interpolation based teaching-learning-based optimization for chemical dynamic system optimization. Knowl Based Syst 145:250–263. https://doi.org/10.1016/j.knosys.2018.01.021
20. Kaveh A, Mahdavi VR (2014) Colliding bodies optimization method for optimum design of truss structures with continuous variables. Adv Eng Softw 70:1–12. https://doi.org/10.1016/j.advengsoft.2014.01.002
21. Kaveh A, Ilchi Ghazaan M (2014) Enhanced colliding bodies optimization for design problems with continuous and discrete variables. Adv Eng Softw 77:66–75. https://doi.org/10.1016/j.advengsoft.2014.08.003
22. Kaveh A, Mahdavi VR (2015) Colliding bodies optimization algorithms. Colliding Bodies Optim. https://doi.org/10.1007/978-3-319-19659-6_2
23. Kaveh A, Ilchi Ghazaan M (2018) Meta-heuristic algorithms for optimal design of real-size structures. Springer, Switzerland. https://doi.org/10.1007/978-3-319-78780-0
24. He M, Liu M, Wang R, Jiang X, Liu B, Zhou H (2019) Particle swarm optimization with damping factor and cooperative mechanism. Appl Soft Comput 76:45–52. https://doi.org/10.1016/j.asoc.2018.11.050
25. Yadav A, Deep K, Kim JH, Nagar AK (2016) Gravitational swarm optimizer for global optimization. Swarm Evol Comput 31:64–89. https://doi.org/10.1016/j.swevo.2016.07.003
26. Wang L, Yang B, Orchard J (2016) Particle swarm optimization using dynamic tournament topology. Appl Soft Comput 48:584–596. https://doi.org/10.1016/j.asoc.2016.07.041

27. Derrac J, García S, Molina D, Herrera F (2011) A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol Comput 1(1):3–18. https://doi.org/10.1016/j.swevo.2011.02.002
28. Jalili S, Kashan AH, Hosseinzadeh Y (2017) League championship algorithms for optimum design of pin-jointed structures. J Comput Civ Eng 31(2):04016048. https://doi.org/10.1061/(asce)cp.1943-5487.0000617
29. 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
30. Lieu QX, Do DT, 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
31. Kaveh A, Ilchi Ghazaan M (2015) Enhanced colliding bodies algorithm for truss optimization with frequency constraints. J Comput Civ Eng 29(6):04014104. https://doi.org/10.1061/(asce)cp.1943-5487.0000445
32. American Institute of Steel Construction (AISC) (2001) Manual of steel construction: load and resistance factor design. ISBN-10:156240517
33. Dumonteil P (1992) Simple equations for effective length factors. Eng J AISC 29(3):111–115
34. Talatahari S (2016) Symbiotic organisms search for optimum design of and grillage system. Asian J Civ Eng 17(3):299–313
35. Kaveh A, Biabani Hamedani K, Milad Hosseini S, Bakhshpoori T (2020) Optimal design of planar steel frame structures utilizing meta-heuristic optimization algorithms. Structures 25:335–346. https://doi.org/10.1016/j.istruc.2020.03.032
36. Farshchin M, Maniat M, Camp CV, Pezeshk S (2018) School based optimization algorithm for design of steel frames. Eng Struct 171:326–335. https://doi.org/10.1016/j.engstruct.2018.05.085
37. American Institute of Steel Construction (AISC) (1989) Manual of steel construction: allowable stress design, Chicago, USA. ISBN-10:999460693X
38. Hasançebi O, Çarbaş S, Doğan E, Erdal F, Saka M (2009) Performance evaluation of metaheuristic search techniques in the optimum design of real size pin jointed structures. Comput Struct 87(5–6):284–302. https://doi.org/10.1016/j.compstruc.2009.01.002
39. Kaveh A, Ilchi Ghazaan M (2016) Enhanced whale optimization algorithm for sizing optimization of skeletal structures. Mech Based Des Struct Mach 45(3):345–362. https://doi.org/10.1080/15397734.2016.1213639
40. Kaveh A, Zolghadr A (2016) Optimal analysis and design of large-scale domes with frequency constraints. Smart Struct Syst 18(4):733–754. https://doi.org/10.12989/sss.2016.18.4.733
41. Kaveh A, Zolghadr A (2017) Optimal design of cyclically symmetric trusses with frequency constraints using cyclical parthenogenesis algorithm. Adv Struct Eng 21(5):739–755. https://doi.org/10.1177/1369433217732492