Structures 27 (2020) 1722–1739

Contents lists available at ScienceDirect

Structures

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

Billiards-inspired optimization algorithm; a new meta-heuristic method

A. Kaveh*, M. Khanzadi, M. Rastegar Moghaddam

School of Civil Engineering, Iran University of Science and Technology, Narmak, Tehran 16846-13114, Iran

ARTICLE INFO

Keywords:
Metaheuristics
Billiards-inspired optimization algorithm
Mathematical functions
Optimal structural design

ABSTRACT

This paper introduces a new physics-based meta-heuristic algorithm called Billiards-inspired Optimization Algorithm (BOA). The optimization process of this algorithm resembles the billiards game. Each solution candidate is considered as a multi-dimensional billiards ball and the best obtained solutions as pockets. When the balls encounter other balls, vector algebra and conservation laws determine final positions of the balls in optimization search space. In order to evaluate the validity of the proposed approach, twenty-three mathematical functions and seven constrained engineering benchmark problems are studied. The results indicated that the BOA can play an effective role in the field of optimization.

1. Introduction

Nowadays, the modern world is growing up rapidly. Large scale structures, extensive transportation networks, complicated machines and comprehensive computer networks are most involved in this progress. Optimal design of these systems plays a prominent role in their success and survival. Mathematical programming and meta-heuristics are two main approaches for solving complex optimization models. The first group, although guarantees optimality, they are sometimes time consuming and rely on some initial requirements and gradient information of the objective function [1], whereas meta-heuristics find approximate optimal solution in reasonable time. It should be added that the latter stochastic methods can be used in all disciplines.

Nature-inspired meta-heuristic algorithms can be classified in three categories. First one is evolutionary process-based group. These algorithms employ mechanisms inspired by biological evolution, such as reproduction, mutation, recombination and selection. Genetic Algorithm (GA), the most popular algorithm in this category, was introduced by Holland [2] in 1975 based on the concept of Darwin's theory. Afterwards, Goldberg extended GA in 1989 [3]. Evolution strategy (ES) developed by Rechenberg [4] is another type of this group.

Many meta-heuristic algorithms are inspired by the social behavior of animals and form the second group of meta-heuristics, called swarm-based group. Particle Swarm Optimizer (PSO) formulated by Eberhart and Kennedy [5], and Ant Colony Optimization (ACO) proposed by Dorigo et al. [6] are two most commonly used algorithms of this group. PSO is inspired from movement of organisms in a bird flock or fish school and ACO mimics the behavior of ants in finding food. Artificial bee colony (ABC) [7], Cuckoo Search (CS) [8], Bat-inspired Algorithm (BA) [9], Firefly Algorithm (FA) [10], Dolphin Echolocation (DE) [11], Migrating Birds Optimization (MBO) [12], Grey Wolf Optimizer (GWO) [1], Moth-flame optimization algorithm (MFO) [13], Whale Optimization Algorithm (WOA) [14], Grasshopper Optimization Algorithm (GOA) [15], and Water Strider Algorithm (WSA) [16] are other examples of swarm-based algorithms.

Physical phenomena have motivated many scientists and has led to the development of many algorithms such as Simulated Annealing (SA) proposed by Kirkpatrick et al. [17], the Big Bang–Big Crunch algorithm (BB–BC) proposed by Erol and Eksin [18], the Colliding Bodies Optimization (CBO) presented by Kaveh and Mahdavi [19], Ray Optimization (RO) by Kaveh and Khayatazad [20], Small-World Optimization Algorithm (SWOA) by Haifeng et al. [21], Charged System Search (CSS) by Kaveh and Talatahari [22] and Multi-Verse Optimizer (MVO) by Mirjalili et al. [23]. Physical phenomena-based algorithms are the third and an attractive subset of meta-heuristics. Many recent metaheuristic algorithms and their applications can be found in Kaveh [22], Kaveh and Mahjoubi [23], Kaveh and Dadras Eslamlou [24] and Kaveh et al. [25].

This paper presents a new optimization algorithm based on billiards, as a pervasive game in the world. Billiards is bound up with many physical laws, especially energy and momentum conservations. This algorithm which is called Billiards-inspired Optimization Algorithm (BOA) consists of some balls as searching agents and some pockets as convergence targets. Balls are divided into two groups, cue balls and ordinary balls. Each cue ball moves toward its target ball and propels it to pockets. The process is repeated until the satisfaction of the termination criterion. After this introduction, Section 2 describes

* Corresponding author.
E-mail address: alikaveh@iust.ac.ir (A. Kaveh).

https://doi.org/10.1016/j.istruc.2020.07.058
Received 21 March 2020; Received in revised form 16 June 2020; Accepted 24 July 2020
Available online 10 August 2020
2352-0124/

inspiration principles and mechanism of algorithm in two distinct subsections. Also, sensitivity analysis is presented in Section 3. In order to validate BOA, some mathematical functions and optimization results are provided in Section 4. This is followed by Section 5 which includes some well-studied engineering problems that have various constraints. The last three problems are sizing optimization of truss structures that must satisfy multiple frequency constraints. The final section is devoted to concluding remarks.

2. Billiards-inspired optimization algorithm

In the first part of this section, billiards game and its physical background, as the inspiration source of the present algorithm, is examined. Then, the mechanism of the BOA is explained in the second part.

2.1. Inspiration

2.1.1. Billiards game

Billiards as a general class of games is played with a long stick called cue, which is used to strike billiards balls, and thereby cause them to move around a cloth-covered billiards table bounded by elastic cushions attached to the confining rails of the table, Fig. 1.

Many countries have been credited with the invention of the game, but in fact, nothing is really known about the origin of billiards. It may be inferred that it developed from variety of games in which propelling a ball was a main feature. Different forms of the game tend to be played in certain countries, though many of the games cross many national boundaries. Carom or French billiards is played with three balls on a table that has no pockets. The other principal games are played on tables which have six pockets, one at each corner and one in each of the long sides. These games include English billiards, snooker and pool. The differences between these kinds of billiards are about number of balls, size of the pockets and rules [24].

As an example, pool is played with fifteen numbered balls, in addition to one white cue ball; the balls are in solid colors and stripes. Pool includes wide variety of games and each one has specific name. The most popular version among recreational players is probably eight-ball. The goal of eight-ball is to sink the one of two types of balls followed by the Eight-ball (black ball). To begin game, the fifteen object balls are placed in a pyramid layout with its apex on a spot near the foot of the table. The first player then stands at the head of the table and drives the cue ball into the balls to break them apart. To continue game, he is usually required either to sink a ball or to drive balls into better situations. When a player fails to pocket a ball, his opponent continues game. The first player, who succeeds to pocket all balls plus the Eight-ball according to predefined rules, wins the game [25].

Fig. 1. A billiards shot.

Briefly, billiards is about manipulating the balls accurately on the table along different trajectories to achieve game purposes [26]. This game includes many technical and tactical nature challenges [27]. Therefore, adopting techniques and strategies that have foundation in physics and mathematics, beside practicing hard, can lead to professionalism in billiards [28].

2.1.2. Mechanics of billiards

Billiards sports are so vulnerable to the natural laws of physics. The physics behind them mainly involves collisions between balls. Collision between two billiards balls is almost elastic. In perfectly elastic collisions, the kinetic energies of balls are conserved before and after collision, in addition to the sum of both momenta. When two balls contact each other, the forces between the balls during the collision are directed along an imaginary line that connects their centers, in other words impact line. To account for this fact, the impact velocities decompose in two components, parallel component and perpendicular component. The first one is parallel to impact line of balls and the second component is perpendicular to it. In the following this will be investigated in more details [29].

The connecting vector, $\overrightarrow{m_1m_2}$ originates from the center of the first colliding ball to the center of the second ball, at the moment of collision. The unit vector of connecting vector is determined as Eq. (1), where r_1 and r_2 are radii of the balls.

$$\vec{e}_{\parallel} = \frac{\overrightarrow{m_1m_2}}{r_1 + r_2} \tag{1}$$

By assuming that this collision is completely elastic, the kinetic energy and momentum conservation equations expressed in this system have the following form:

$$\frac{1}{2}m_1v_{1.\parallel}^2 + \frac{1}{2}m_2v_{2.\parallel}^2 + \frac{1}{2}m_1v_{1.\perp}^2 + \frac{1}{2}m_2v_{2.\perp}^2 = \frac{1}{2}m_1v_{1.\parallel}'^2 + \frac{1}{2}m_2v_{2.\parallel}'^2 + \frac{1}{2}m_1v_{1.\perp}'^2 + \frac{1}{2}m_2v_{2.\perp}'^2 \tag{2}$$

$$\begin{aligned} m_1v_{1.\parallel} + m_2v_{2.\parallel} &= m_1v_{1.\parallel}' + m_2v_{2.\parallel}' \\ m_1v_{1.\perp} + m_2v_{2.\perp} &= m_1v_{1.\perp}' + m_2v_{2.\perp}' \end{aligned} \tag{3}$$

where v_1 and v_2 are velocities of first ball and second ball before collision and v_1' and v_2' are velocities of them after collision. Also symbols $\parallel$ and $\perp$ denote parallel and perpendicular components, respectively. The parameters, m_1 and m_2 represent the masses of the balls. Fig. 2 depicts decomposition of velocities into these two components. The parallel and the perpendicular components of the impact velocities are expressed as:

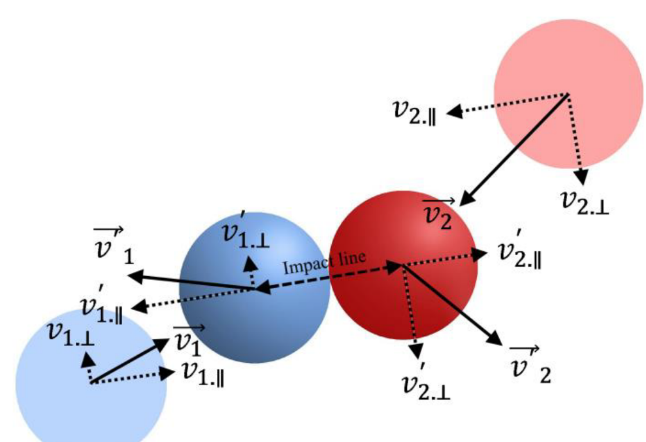

Fig. 2. Decomposition of velocities into parallel and perpendicular components.

$$\vec{v}_{1.\parallel} = (\vec{v}_1 \cdot \vec{e}_\parallel)\vec{e}_\parallel. \qquad \vec{v}_{1.\perp} = \vec{v}_1 - v_{1.\parallel}\vec{e}_\parallel \tag{4}$$

$$\vec{v}_{2.\parallel} = (\vec{v}_2 \cdot \vec{e}_\parallel)\vec{e}_\parallel. \qquad \vec{v}_{2.\perp} = \vec{v}_2 - v_{2.\parallel}\vec{e}_\parallel \tag{5}$$

Since the forces are applied along the impact line only, the perpendicular components of the momenta are conserved for each ball separately; therefore, the perpendicular components of the velocities remain unchanged.

$$m_1\vec{v}_{1.\perp} = m_1\vec{v'}_{1.\perp} \tag{6}$$

$$m_2\vec{v}_{2.\perp} = m_2\vec{v'}_{2.\perp} \tag{7}$$

Similarly, the parts of the kinetic energies corresponding to the perpendicular components remain unchanged before and after the collision. Therefore, the sum of kinetic energies corresponding to the parallel components of the velocities is conserved as well:

$$\frac{1}{2}m_1v_{1.\parallel}^2 + \frac{1}{2}m_2v_{2.\parallel}^2 = \frac{1}{2}m_1v'^2_{1.\parallel} + \frac{1}{2}m_2v'^2_{2.\parallel} \tag{8}$$

Similarly, the sum the parallel components of the momenta is also conserved.

$$m_1v_{1.\parallel} + m_2v_{2.\parallel} = m_1v'_{1.\parallel} + m_2v'_{2.\parallel} \tag{9}$$

Eqs. (8) and (9) with a little manipulation could be rewritten as:

$$m_1(v_{1.\parallel}^2 - v'^2_{1.\parallel}) = m_2(v'^2_{2.\parallel} - v_2^2) \tag{10}$$

$$m_1(v_{1.\parallel} - v'_{1.\parallel}) = m_2(v'_{2.\parallel} - v_{2.\parallel}) \tag{11}$$

Eq. (10) can be rearranged as:

$$m_1(v_{1.\parallel} - v'_{1.\parallel})(v_{1.\parallel} + v'_{1.\parallel}) = m_2(v'_{2.\parallel} - v_{2.\parallel})(v'_{2.\parallel} + v_{2.\parallel}) \tag{12}$$

By dividing Eq. (12) by Eq. (11), we have

$$v_{1.\parallel} + v'_{1.\parallel} = v_{2.\parallel} + v'_{2.\parallel} \tag{13}$$

This shows that the sums of the parallel velocities before and after the collision are equal.

According to Eqs. (9) and (13), final parallel components of velocities after the collision are determined as:

$$v'_{1.\parallel} = \frac{m_1 - m_2}{m_1 + m_2}v_{1.\parallel} + \frac{2m_2}{m_1 + m_2}v_{2.\parallel} \tag{14}$$

$$v'_{2.\parallel} = \frac{2m_1}{m_1 + m_2}v_{1.\parallel} + \frac{m_2 - m_1}{m_1 + m_2}v_{2.\parallel} \tag{15}$$

If the balls have equal masses, Eqs. (14) and (15) become as below, indicating that balls switch only the parallel component of their velocities, although perpendicular components remain unaltered.

$$v'_{1.\parallel} = v_{2.\parallel} \ . \quad v'_{2.\parallel} = v_{1.\parallel} \tag{16}$$

The final velocity vectors are determined by adding both components as

$$\vec{v'}_1 = v'_{1.\parallel}\vec{e}_\parallel + \vec{v}_{1.\perp} \tag{17}$$

$$\vec{v'}_2 = v'_{2.\parallel}\vec{e}_\parallel + \vec{v}_{2.\perp} \tag{18}$$

Aforementioned concepts and facts could be generalized to collisions in multi-dimensional spaces. When two balls collide together, the force between the balls is directed along an imaginary line that connects their centers. So along this direction, there is simply a one dimensional collision problem, and velocities in this direction are calculated by Eqs. (14) and (15) and other components of velocities remain unchanged.

2.2. Mechanism of the algorithm

As mentioned before, the proposed meta-heuristic algorithm is inspired from billiards game. In this algorithm, each solution candidate, which contains a number of decision variables, is considered as a multi-dimensional billiards ball. These balls are actually the agents of the optimization problem and each dimension of them represents a variable. Briefly, the process starts with initial generation of balls with random distribution, then some best ones are selected as pockets. Afterwards, balls are divided in two groups, ordinary balls and cue balls. Each cue ball hits its target ball and moves it toward a pocket. When cue balls encounter other balls, kinematic and collision laws are governing. They specify moving directions and final status of collided balls. The BOA pseudo code is shown in Fig. 3 and its steps are outlined in the following:

```
set N=number of balls;
    M=number of variables
    K= number of pockets;
    ET= escaping threshold;
    iter=0;
Initialize 2N balls and K pockets by Eq. (19);
while (iter< iteration limit)
    Evaluate balls and pocket position according to objective
function;
    Update pocket memory and population;
    Create ordinary ball and cue ball groups;
    for each pair ball
        Select a destination pocket by utilizing the roulette-wheel
selection mechanism;
    end
    Update position of current ordinary ball by Eq. (22);
    Calculate velocity of ordinary ball after collision by Eq. (23);
    Calculate velocity of cue ball after collision by Eq. (25);
    Update position of current cue ball by Eq. (26);
    if (rand<ET)
        Regenerate a random dimension of balls by Eq. (27);
    end
        Check the boundary condition limitations and correct the
balls if they are not within the defined range;
    iter=iter+1;
end
Return the best pocket as final solution.
```

Fig. 3. Pseudo code of the BOA.

Table 1
Properties of the Rastrigin function.

Function name	Function	Variables' Range	f_{min}
Rastrigin	$F(x) = \sum_{i=1}^{2} [x_i^2 - 10\cos(2\pi x_i) + 10]$	$[-5.12, 5.12]$	0

Step 1: Initialization

The first population of balls is distributed randomly in the search space as:

$$B_{n.m}^0 = var_m^{min} + rand_{[0,1]}(var_m^{max} - var_m^{min}). \quad n = 1.2.\ 3\ldots 2N; \quad m = 1.2.\ 3\ldots M \tag{19}$$

where, $B_{n.m}^0$ determines the initial value of the m^{th} variable of n^{th} ball. var_m^{max} and var_m^{min} are the maximum and the minimum allowable values for m^{th} variable. $rand_{[0,1]}$ is a random number with uniform distribution in the interval [0, 1], and M and 2 N are the number of variables and agents, respectively. Also some pockets are generated by previous equation randomly in search space. Number of pockets is determined by user.

Step 2: Evaluation

In this step, balls and pockets' positions are evaluated according to the objective function.

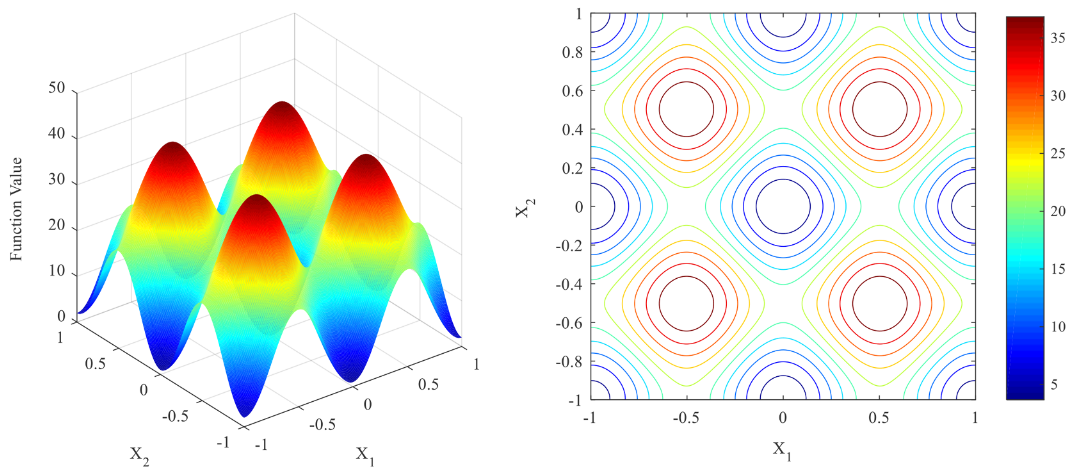

Fig. 4. Perspective view of the Rastrigin function and the corresponding contours.

Table 2
Variation range of parameters for the sensitivity analysis.

Number of agents	Number of iterations	Number of pockets	ω	ES
30	200	1–25	0.5–1	0–0.5

Table 3
Properties of the uni-modal functions.

Function	Range	f_{min}
$F_1(x) = \sum_{i=1}^{n} x_i^2$	$X \in [-100, 100]^{30}$	0
$F_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	$X \in [-10, 10]^{30}$	0
$F_3(x) = \sum_{i=1}^{n} \left(\sum_{j-1}^{i} x_j \right)^2$	$X \in [-100, 100]^{30}$	0
$F_4(x) = \max_i \{\lvert x_i \rvert, 1 \leqslant i \leqslant n\}$	$X \in [-100, 100]^{30}$	0
$F_5(x) = \sum_{i=1}^{n} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	$X \in [-30, 30]^{30}$	0
$F_6(x) = \sum_{i=1}^{n} ([x_i + 0.5])^2$	$X \in [-100, 100]^{30}$	0
$F_7(x) = \sum_{i=1}^{n} i x_i^4 + random[0,1)$	$X \in [-1.28, 1.28]^{30}$	0

Step 3: Determining the pockets

BOA pockets play two roles in this algorithm, a target for balls that provides the exploitation ability of the algorithm and a memory that saves the first K top solutions found so far. Existence of this memory improves the performance of algorithm without increasing the computational cost. This memory is updated and substituted with best balls found positions in each iteration.

Step 4: Grouping the balls

After determining the pockets, balls are sorted according to their finesses, and then they are divided in two equal groups, ordinary balls and cue balls. Superior half includes ordinary balls (i. e. $n = 1,..., N$) and the second one determines cue balls (i. e. $n = N + 1,..., 2N$). Each

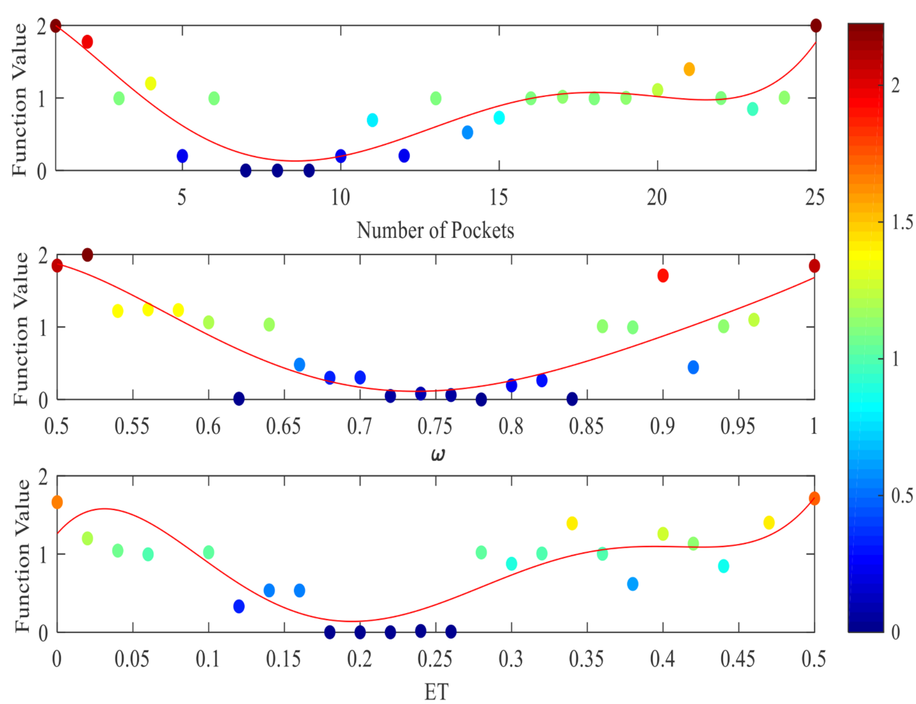

Fig. 5. Behavior of the BOA in the process of the sensitivity analysis.

Table 4
Properties of the multi-modal functions.

Function	Range	f_{min}
$F_8(x)=\sum_{i=1}^{n}-x_i\sin(\sqrt{\lvert x_i\rvert})$	$X\in[-500,\ 500]^{30}$	-418.9829×5
$F_9(x)=\sum_{i=1}^{n}[x_i^2-10\cos(2\pi x_i)+10]$	$X\in[-5.12,\ 5.12]^{30}$	0
$F_{10}(x)=-20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}\right)-\exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right)+20+e$	$X\in[-32,\ 32]^{30}$	0
$F_{11}(x)=\frac{1}{4000}\sum_{i=1}^{n}x_i^2-\prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right)+1$	$X\in[-600,\ 600]^{30}$	0
$F_{12}(x)=\frac{\pi}{n}\left\{10\sin(\pi y_1)+\sum_{i=1}^{n-1}(y_i-1)^2[1+10sin^2(\pi y_{i+1})]+(y_n-1)^2\right\}+\sum_{i=1}^{n}u(x_i,10,100,4)$	$X\in[-50,\ 50]^{30}$	0
$y_i=1+\frac{x_i+1}{4}$		
$u(x_i,a,k,m)=\begin{cases}k(x_i-a)^m & x_i>a\\ 0 & -a<x_i<a\\ k(-x_i-a)^m & x_i<-a\end{cases}$		
$F_{13}(x)=0.1\left\{sin^2(3\pi x_1)+\sum_{i=1}^{n}(x_i-1)^2[1+sin^2(3\pi x_i+1)]+(x_n-1)^2[1+sin^2(2\pi x_n)]\right\}+\sum_{i=1}^{n}u(x_i,5,100,4)$	$X\in[-50,\ 50]^{30}$	0

Table 5
Properties of the fixed-dimensional multi-modal functions.

Function	Range	f_{min}
$F_{14}(x)=\left(\frac{1}{500}+\sum_{j=1}^{25}\frac{1}{j+\sum_{i=1}^{2}(x_i-a_{ij})^6}\right)^{-1}$	$X\in[-65,\ 65]^{2}$	1
$F_{15}(x)=\sum_{i=1}^{11}\left[a_i-\frac{x_1(b_i^2+b_ix_2)}{b_i^2+b_ix_3+x_4}\right]^2$	$X\in[-5,\ 5]^{4}$	0.00030
$F_{16}(x)=4x_1^2-2.1x_1^4+\frac{1}{3}x_1^6+x_1x_2-4x_2^2+4x_2^4$	$X\in[-5,\ 5]^{2}$	-1.0316
$F_{17}(x)=\left(x_2-\frac{5.1}{4\pi^2}x_1^2+\frac{5}{\pi}x_1-6\right)^2+10\left(1-\frac{1}{8\pi}\right)\cos x_1+10$	$X\in[-5,\ 5]^{2}$	0.398
$F_{18}(x)=$ $[1+(x_1+x_2+1)^2(19-14x_1+3x_1^2-14x_2+6x_1x_2+3x_2^2)]$ $\times[30+(2x_1-3x_2)^2\times(18-32x_1+12x_1^2+48x_2-36x_1x_2+27x_2^2)]$	$X\in[-2,\ 2]^{2}$	3
$F_{19}(x)=-\sum_{i=1}^{4}c_i\ \exp\left(-\sum_{j=1}^{3}a_{ij}(x_j-p_{ij})^2\right)$	$X\in[1,3]^{3}$	-3.86
$F_{20}(x)=-\sum_{i=1}^{4}c_i\ \exp\left(-\sum_{j=1}^{6}a_{ij}(x_j-p_{ij})^2\right)$	$X\in[0,\ 1]^{6}$	-3.32
$F_{21}(x)=-\sum_{i=1}^{5}[(X-a_i)(X-a_i)^T+c_i]^{-1}$	$X\in[0,\ 10]^{4}$	-10.1532
$F_{22}(x)=-\sum_{i=1}^{7}[(X-a_i)(X-a_i)^T+c_i]^{-1}$	$X\in[0,\ 10]^{4}$	-10.4028
$F_{23}(x)=-\sum_{i=1}^{10}[(X-a_i)(X-a_i)^T+c_i]^{-1}$	$X\in[0,\ 10]^{4}$	-10.5363

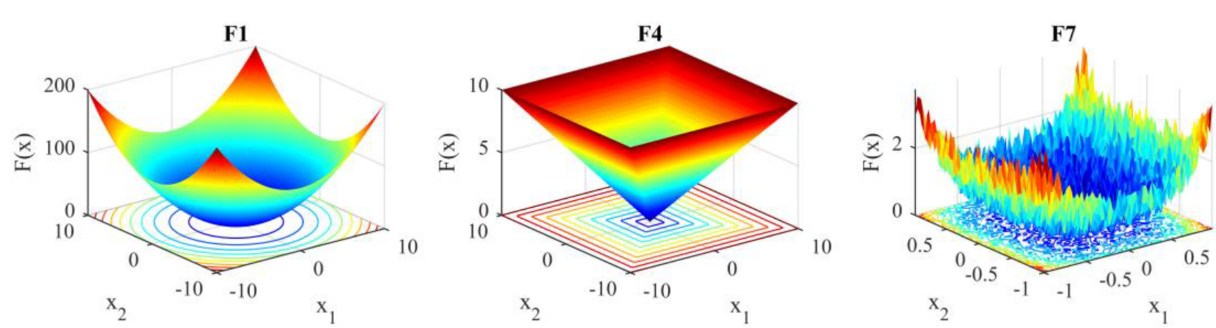

Fig. 6. 2D perspective views of some uni-modal functions.

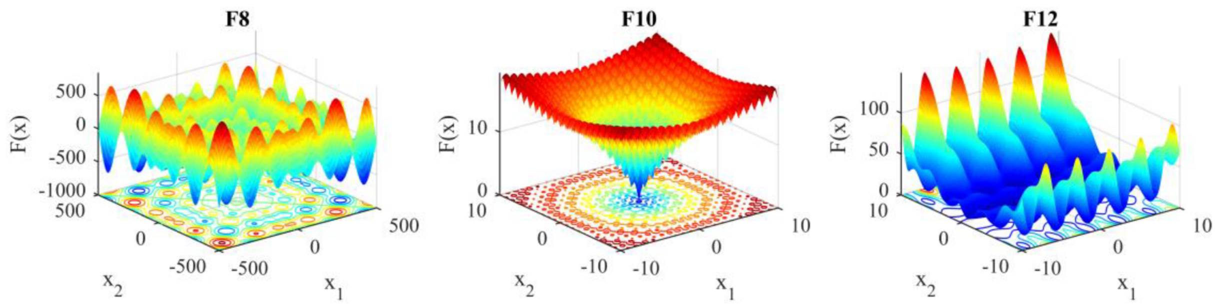

Fig. 7. 2D perspective views of some multi-modal functions.

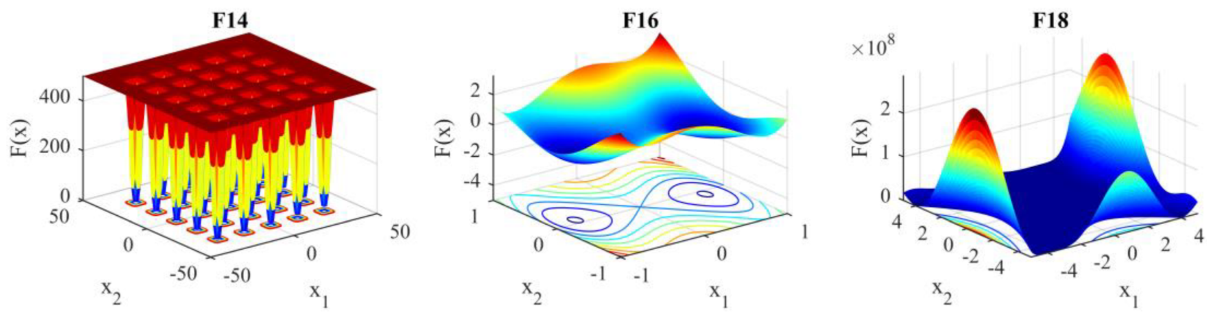

Fig. 8. 2D perspective views of some fixed-dimensional multi-modal functions.

Table 6
Comparison of the statistical results for uni-modal functions.

Functions	Quantity	BOA	ECBO	WOA	PSO	GOA	WSA	IGWO	GA	GSA	FEP
F1	Ave.	1.42E−64	0.014981	1.42E−43	3.72E−04	8.924014	1.09E−50	2.6526E−46	0.493651	2.53E−16	0.00057
	Std. dev.	4.46E−62	0.009619	3.86E−35	0.000935	7.366225	3.99E−50	6.2455E−46	0.976457	9.67E−17	0.00013
F2	Ave.	1.32E−38	0.003078	2.39E−25	2.20E−02	36.16146	4.89E−28	6.2336E−28	0.027773	0.055655	0.0081
	Std. dev.	1.87E−37	0.001012	2.03E−21	0.079000	17.87108	1.94E−27	6.7665E−28	0.053506	194,074	0.00077
F3	Ave.	17.40900	6676.084	43789.54	75.72718	2599.891	0.014089	6.8778E−10	4682.494	896.5347	0.016
	Std. dev.	15.34927	2308.023	13324.09	33.77983	1760.702	0.011180	1.8841E−09	1974.782	318.9559	0.014
F4	Ave.	2.17E−15	14.72304	6.651892	1.148165	5.978819	0.000490	7.4596E−12	9.282368	7.35487	0.3
	Std. dev.	7.07E−12	2.376884	4.220696	0.288924	2.954771	0.000354	1.3516E−11	2.208485	1.741452	0.5
F5	Ave.	25.74747	618.3821	28.21277	83.21137	526.1095	32.42146	26.6287	481.1478	67.54309	5.06
	Std. dev.	0.029795	519.0368	0.564472	51.00950	325.6448	29.52849	0.5501	481.3205	62.22534	5.87
F6	Ave.	0.075961	0.018717	0.453929	0.000171	106.3820	0	0.5128	0.398834	2.5E−16	0
	Std. dev.	0.003099	0.011134	0.220565	0.000146	107.8309	0	0.3822	0.472675	1.74E−16	0
F7	Ave.	0.000436	0.077259	0.004782	0.163603	0.587710	0.006433	0.0012	0.011947	0.089441	0.1415
	Std. dev.	0.000730	0.019375	0.004616	0.059297	0.245901	0.0018394	8.36664E−04	0.006386	0.04339	0.3522

Table 7
Comparison of the statistical results for multi-modal functions.

Functions	Quantity	BOA	ECBO	WOA	PSO	GOA	WSA	IGWO	GA	GSA	FEP
F8	Ave.	−11175.9	−11911.0	−10562.6	−4889.97	−7286.43	−9354.74	−6.2291E + 03	−10348.4	−2821.07	−12554.5
	Std. dev.	1466.041	211.2423	1562.556	1384.132	500.4084	653.1757	942.4604	389.5341	493.0375	52.6
F9	Ave.	0	8.420601	0	53.33217	166.6363	40.56002	0	9.177337	25.96841	0.046
	Std. dev.	0	2.352613	0	14.06120	35.81810	10.78416	0	2.255343	7.470068	0.012
F10	Ave.	8.88E−16	0.118860	4.09E−15	0.224646	12.64091	1.88E−14	8.7041E−15	1.123709	0.062087	0.018
	Std. dev.	9.86E−32	0.155987	2.23E−15	0.416934	6.554900	4.52E−15	1.7203E−15	0.558145	0.23628	0.0021
F11	Ave.	0	0.012818	1.11E−16	0.008880	1.708783	0.016042	0	0.293621	27.70154	0.016
	Std. dev.	0	0.008800	0.054574	0.009813	0.716727	0.020111	0	0.246095	5.040343	0.022
F12	Ave.	0.005633	0.078097	0.022028	0.031103	4.627045	1.57E−32	0.0303	0.128613	1.799617	9.20E−06
	Std. dev.	0.001536	0.070091	0.019745	0.025860	2.087953	5.57E−48	0.0212	0.153503	0.95114	3.60E−06
F13	Ave.	0.071574	0.106004	0.502757	0.007760	3.122240	1.35E−32	0.4533	0.177732	8.899084	0.00016
	Std. dev.	0.034192	0.097929	0.218434	0.002042	1.279501	5.57E−48	0.2504	0.139143	7.126241	0.000073

cue ball belongs to its same rank in superior group. This grouping method is borrowed from CBO [19].

Step 5: Allocating the pockets to balls

By utilizing the roulette-wheel selection mechanism, a destination pocket for each ordinary ball is selected. Pockets with lower objective function value bodes more merit.

Suggested probability for choosing a pocket is as the following:

$$P_k = \frac{e^{-\beta f_k}}{\sum_k e^{-\beta f_k}}; \quad k = 1,2,\ldots,K \tag{20}$$

Table 8
Comparison of the statistical results for fixed-dimensional multi-modal functions.

Functions	Quantity	BOA	ECBO	WOA	PSO	GOA	WSA	IGWO	GA	GSA	FEP
F14	Ave.	0.998004	0.998004	1.494227	3.562454	0.998004	0.998004	1.1303	1.130409	5.859838	1.22
	Std. dev.	0.705966	0.000000	3.502703	2.480270	5.06E − 16	1.13E − 16	0.5034	0.430993	3.831299	0.56
F15	Ave.	0.000369	0.004632	0.000942	0.000923	0.006711	0.000549	5.9778E − 04	0.001722	0.003673	0.0005
	Std. dev.	0.000319	0.000187	0.000402	0.000241	0.013470	0.00032	2.6297E − 04	0.003555	0.001647	0.00032
F16	Ave.	− 1.03163	− 1.03162	− 1.03163	− 1.03163	− 1.03163	− 1.03163	− 1.0316	− 1.03163	− 1.03163	− 1.03
	Std. dev.	1.41E − 07	0.000016	1.12E − 09	0	2.09E − 13	5.68E − 16	2.7209E − 06	1.27E − 15	4.88E − 16	4.90E − 07
F17	Ave.	0.397887	0.397887	0.376981	0.397887	0.398890	0.397887	0.397887	0.397887	0.397887	0.398
	Std. dev.	0	0	3.57E − 05	0	3.98E − 06	0	8.4047E − 07	0	0	1.50E − 07
F18	Ave.	3.000014	3.000000	3.000047	3.000000	1.92E + 01	3	3.000017	3	3	3.02
	Std. dev.	3.20E − 05	1.82E − 09	0.000198	2.98E − 15	1.453994	2.91E − 15	2.0705E − 05	1.69E − 15	4.17E − 15	0.11
F19	Ave.	− 3.86281	− 3.86278	− 3.85457	− 3.86278	− 3.54504	− 3.86278	N/A	− 3.86278	− 3.86278	− 3.86
	Std. dev.	2.66E − 15	4.69E − 10	0.009716	2.63E − 12	0.433311	2.46E − 15	N/A	2.71E − 15	2.29E − 15	0.000014
F20	Ave.	− 3.26854	− 3.25066	− 3.21236	− 3.25066	− 3.28614	− 3.25066	N/A	− 3.28633	− 3.31778	− 3.27
	Std. dev.	0.084124	0.057294	0.082641	0.058245	0.059569	0.059241	N/A	0.055415	0.023081	0.059
F21	Ave.	− 8.10257	− 6.13871	− 7.35090	− 5.63886	− 4.88727	− 6.72819	N/A	− 6.99664	− 5.95512	− 5.52
	Std. dev.	2.528422	3.437738	2.449949	2.851678	2.929784	3.378711	N/A	3.696605	3.737079	1.59
F22	Ave.	− 9.732711	− 6.91297	− 8.02790	− 6.68168	− 5.19768	− 7.35819	N/A	− 8.46037	− 6.68447	− 5.53
	Std. dev.	0.259246	3.383608	3.049680	1.853804	3.385219	3.609873	N/A	3.285714	2.014088	2.12
F23	Ave.	− 10.5300	− 8.95150	− 9.42641	− 8.95842	− 6.71956	− 8.30703	N/A	− 8.57634	− 10.5364	− 6.57
	Std. dev.	2.686381	2.928897	3.195802	8.88E − 15	3.624282	3.499898	N/A	3.324285	2.6E − 15	3.14

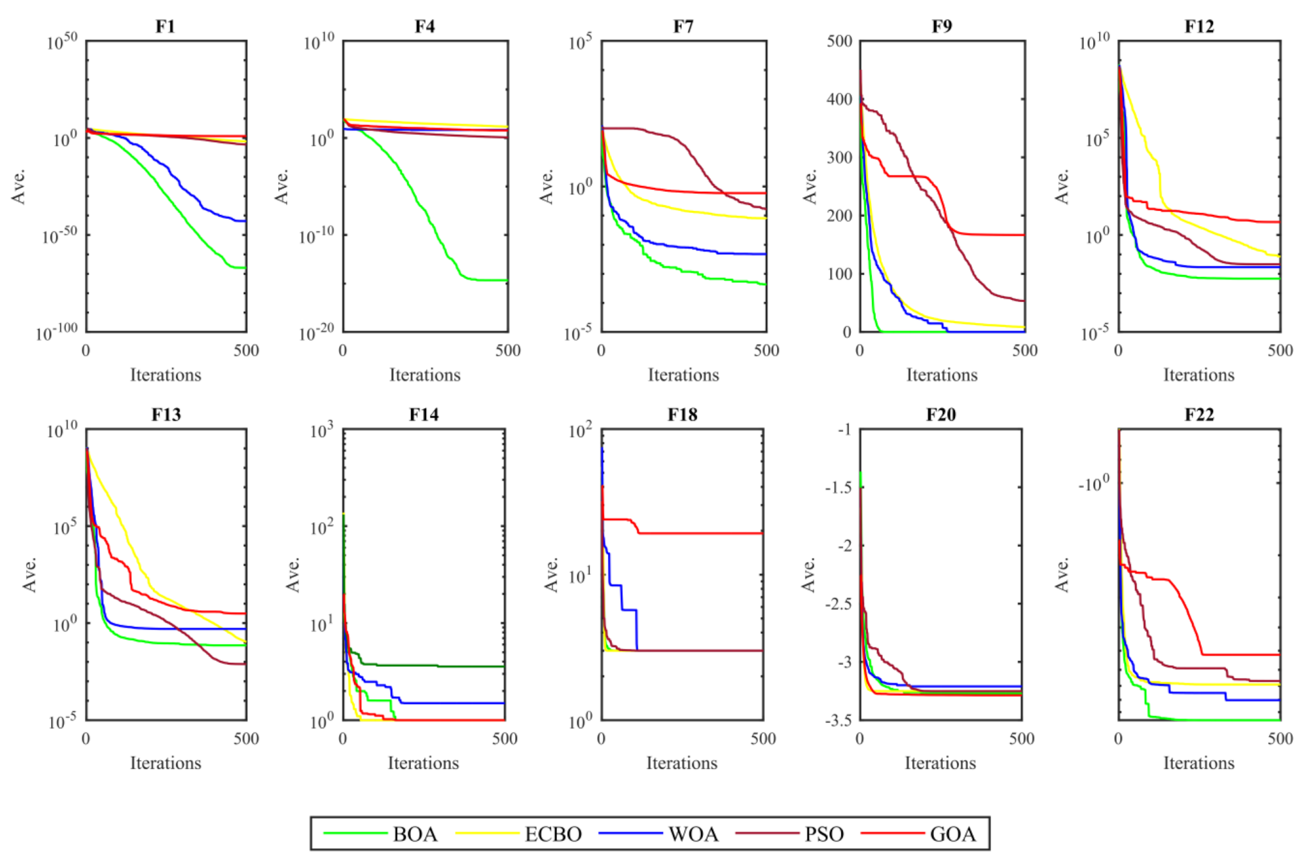

Fig. 9. Comparison of convergence speed of experimented meta-heuristics in some functions.

where β is the selection pressure which is greater than zero and f_k is the objective function value of the k^{th} pocket. So, the probability of selecting a pocket is increased when it is more efficient. Now, current cue balls hit their target balls and move them toward the pockets.

Step 6: Updating the position of the balls

The new positions of ordinary balls after the collision are in the surrounding of their pockets. These positions depend on the shot precision. With the aim of improving the exploitation ability, the error decreases along the search process. Following equation determines the new positions of the ordinary balls.

$$B^{new}_{n.m} = rand_{[-ER,ER]}(1 - PR)(B^{old}_{n,m} - P^{n}_{k,m}) + P^{n}_{k,m} , \quad n = 1.2.\ 3...N \tag{21}$$

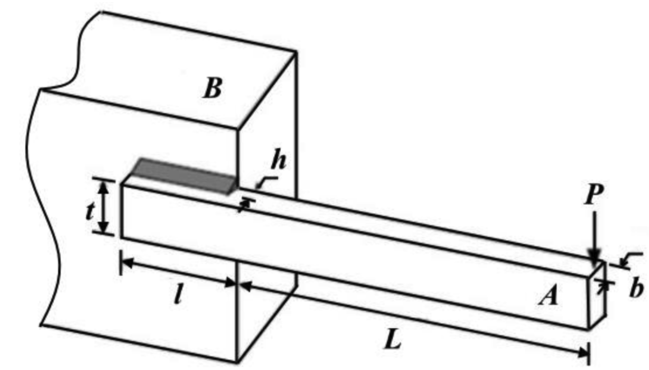

Fig. 10. Schematic of the welded beam.

Table 9
Comparison of optimized designs found for the welded beam problem.

Algorithm	Optimum variables				Optimum Cost
	$x_1\,(h)$	$x_2\,(l)$	$x_3\,(t)$	$x_4\,(b)$	
APPROX [34]	0.2444	6.2189	8.2915	0.2444	2.3815
Simplex [34]	0.2792	5.6256	7.7512	0.2796	2.5307
David [34]	0.2434	6.2552	8.2915	0.2444	2.3841
Random [34]	0.4575	4.7313	5.0853	0.6600	4.1185
GA [35]	0.2489	6.1730	8.1789	0.2533	2.4331
GA [36]	0.208800	3.420500	8.997500	0.21000	1.748310
CPSO [39]	0.202369	3.544214	9.04821	0.205723	1.728024
GSA [32]	0.182129	3.856979	10.0000	0.202376	1.879952
HS [37]	0.2442	6.2231	8.2915	0.2443	2.3807
Improved HS [38]	0.20573	3.47049	9.03662	0.2057	1.7248
RO [20]	0.203687	3.528467	9.004233	0.207241	1.735344
CSS [22]	0.20582	3.468109	9.038024	0.205723	1.724866
TEO [40]	0.205681	3.472305	9.035133	0.206480	1.725284
WSA[16]	0.20573	3.470489	9.036624	0.20573	1.724852
GOA	0.203341	3.514773	9.061732	0.205605	1.730491
PSO	0.202662	3.536398	9.040177	0.205772	1.729875
WOA	0.202503	3.54026	9.039309	0.205722	1.729615
ECBO	0.205801	3.469548	9.035075	0.205801	1.725113
BOA	0.205732	3.470471	9.036572	0.205732	1.724862

Table 10
Comparison of the statistical results for the welded beam problem.

Algorithm	Statistical result			
	Best	Average	Worst	Std. dev.
GA [36]	1.748309	1.771973	1.785835	0.011220
CPSO [39]	1.728024	1.748831	1.782143	0.012926
CSS [22]	1.724866	1.739654	1.759479	0.008064
TEO [40]	1.725284	1.768040	1.931161	0.0581661
WSA[16]	1.724852	1.724908	1.725068	4.15E−05
GOA	1.730491	1.738918	1.743646	0.002151
PSO	1.729875	1.730125	1.739497	0.013581
WOA	1.729615	1.731521	1.739928	0.021617
ECBO	1.725113	1.728513	1.736728	0.005129
BOA	1.724862	1.724981	1.725220	0.011812

$$PR = \frac{iter}{iter_{max}} \tag{22}$$

where, $B^{new}_{n,m}$ and $B^{old}_{n,m}$ are the new and the old values of the m^{th} variable from the n^{th} ordinary ball, respectively. The $P^{n}_{k,m}$ is the m^{th} variable of the k^{th} pocket which belongs to the n^{th} pair of ordinary ball. PR is the precision rate. $rand_{[-ER,ER]}$ is a random number with uniform distribution in the interval [-ER,ER]. ER is the error rate, $iter$ and $iter_{\max}$ are current iteration number and maximum iteration number, respectively.

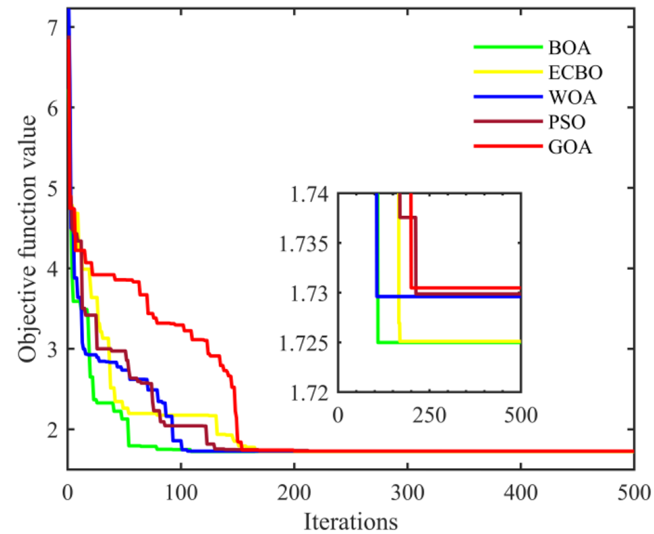

Fig. 11. Comparison of convergence curves obtained for the welded beam problem.

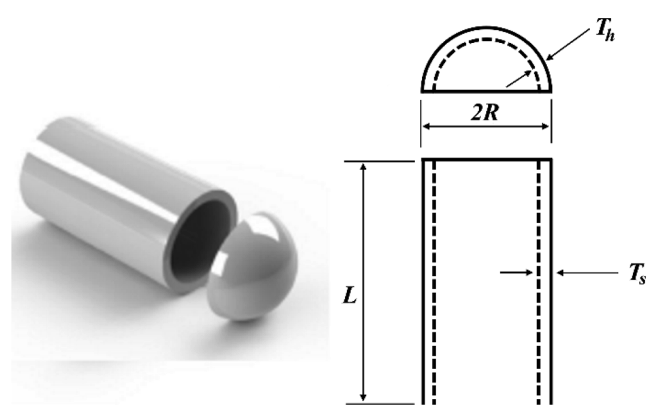

Fig. 12. Schematic of the capped pressure vessel.

Table 11
Comparison of optimized designs found for the pressure vessel problem.

Algorithm	Optimum Variables				Optimum Cost
	$x_1\,(T_s)$	$x_2\,(T_h)$	$x_3\,(R)$	$x_4\,(L)$	
Branch & bound [41]	1.1250	0.6250	47.700	117.7010	8129.1036
GA [36]	0.8125	0.4345	40.3239	200	6288.7445
ES [42]	0.8125	0.4375	42.098087	176.640518	6059.7456
CPSO [39]	0.8125	0.4375	42.091266	176.7465	6061.0777
GSA [32]	1.125	0.625	55.9886598	84.4542025	8538.8359
CDE [44]	0.8125	0.4375	42.098411	176.637690	6059.7340
Improved HS [38]	1.125	0.625	58.29015	43.69268	7197.730
CSS [22]	0.8125	0.4375	42.103624	176.572656	6059.0888
BA [43]	0.8125	0.4375	42.098445	176,636,595	6059.7143
CBO [19]	0.779946	0.385560	40.409065	198.76232	5889.911
TEO [40]	0.779151	0.385296	40.369858	199.301899	5887.511073
GOA	0.797319	0.394470	41.31048	186.67063	5920.5633
PSO	0.980288	0.484884	50.78309	92.288971	6331.9854
WOA	0.836179	0.413426	43.31855	162.10975	5993.9212
ECBO	0.778491	0.384810	40.33204	199.9	5888.0241
BOA	0.778324	0.384750	40.32634	199.92164	5886.1681

Table 12
Comparison of the statistical results for the pressure vessel problem.

Algorithm	Statistical result			
	Best	Average	Worst	Std. dev.
GA [36]	6288.7445	6293.8432	6308.1497	7.4133
CPSO [39]	6061.0777	6147.13	6469.32	130.9297
CSS [22]	6059.0888	6067.9062	6085.4765	10.2564
BA [43]	6059.7143	6179.13	6318.95	137.223
CBO [19]	5889.911	5934.201	6213.006	63.5417
TEO [40]	5887.511073	5942.565917	6134.187981	62.2212
GOA	5920.5633	6081.5124	6331.985448	60.21842
PSO	6331.9854	6521.6742	6909.181971	100.5142
WOA	5993.9212	6125.8516	6331.985448	65.12842
ECBO	5888.0241	6108.1762	6303.186211	64.20321
BOA	5886.1681	6081.3465	6366.846215	63.51425

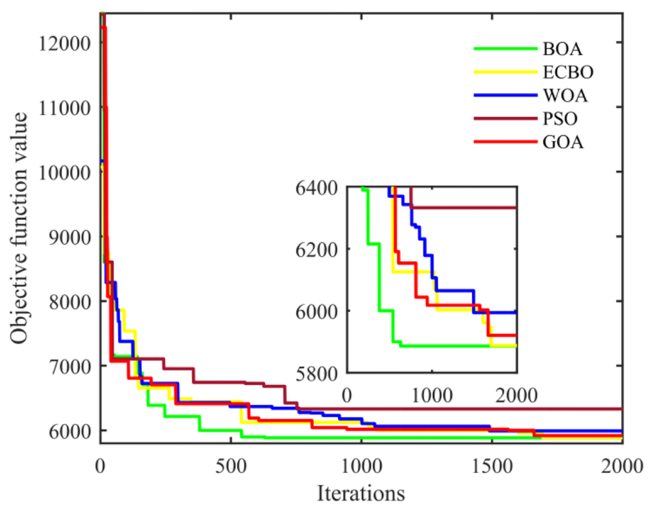

Fig. 13. Comparison of convergence curves obtained for the capped pressure vessel problem.

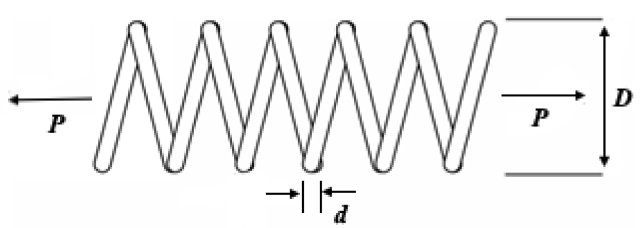

Fig. 14. Schematic of the tension/compression spring.

The positions of cue balls after the collision depend on their velocities. The following equations are utilized to calculate the velocities of balls after the collision. Velocities of the ordinary balls are calculated as:

$$\overrightarrow{v'_n} = \sqrt{2a\,\overrightarrow{B_n^{old}B_n^{new}}}\ \widehat{B_n^{old}B_n^{new}}, \quad n = 1.2.\ 3\ldots N \tag{23}$$

where, $\overrightarrow{v'_n}$ is the velocity of the n^{th} ordinary ball after the collision; $\overrightarrow{B_n^{old}B_n^{new}}$ is the movement vector of the n^{th} ball and $\widehat{B_n^{o}B_n^{n}}$ is the unit movement vector of the n^{th} ordinary ball after collision; Note that, the sign indicates corresponded unit vector. The parameter a is the acceleration rate and is set to unity. The sign "·" denotes the dot product. Also, the velocities of cue balls before and after collision are calculated as:

Table 13
Comparison of optimized designs found for the tension/compression spring problem.

Algorithm	Optimum Variables			Optimum Cost
	$x_1(d)$	$x_2(D)$	$x_3(N)$	
Belegundu [45]	0.050000	0.315900	14.250000	0.012833
GA [35]	0.051480	0.351661	11.632201	0.012704
GA [36]	0.051989	0.363965	10.890522	0.012681
CPSO [39]	0.051728	0.357644	11.244543	0.012674
ES [42]	0.051643	0.355360	11.397926	0.012698
GSA [32]	0.050276	0.323680	13.525410	0.012702
CSS [22]	0.051744	0.358532	11.165704	0.0126384
BA [43]	0.05169	0.35673	11.2885	0.012665
CDE [44]	0.051609	0.354714	11.410831	0.0126702
CBO [19]	0.051894	0.3616740	11.007846	0.0126697
TEO [40]	0.051775	0.3587919	11.168390	0.012665
GOA	0.051809	0.359746	11.10933	0.0126692
PSO	0.051488	0.352020	11.56846	0.0126690
WOA	0.051422	0.350390	11.67269	0.0126695
ECBO	0.051435	0.350642	11.65781	0.0126698
BOA	0.051645	0.355750	11.34177	0.0126644

Table 14
Comparison of the statistical results for the tension/compression spring problem.

Algorithm	Statistical result			
	Best	Average	Worst	Std. dev.
GA [36]	0.0126810	0.0127420	0.012973	5.9000E−5
CPSO [39]	0.0126747	0.012730	0.012924	5.1985E−5
ES [42]	0.012698	0.013461	0.016485	9.6600E−4
CSS [22]	0.0126384	0.012852	0.013626	8.3564E−5
BA[43]	0.012665	0.013501	0.016895	0.00142
CBO [19]	0.126697	0.1272964	0.128808	5.00376E−5
TEO [40]	0.012665	0.012685	0.012715	4.4079E−6
GOA	0.0126692	0.0127717	0.013568	2.0781E−3
PSO	0.0126690	0.0127912	0.013841	0.0032562
WOA	0.0126695	0.0127812	0.014056	0.0021531
ECBO	0.0126698	0.0127649	0.013463	1.7746E−4
BOA	0.0126644	0.0127408	0.013611	5.0331E−5

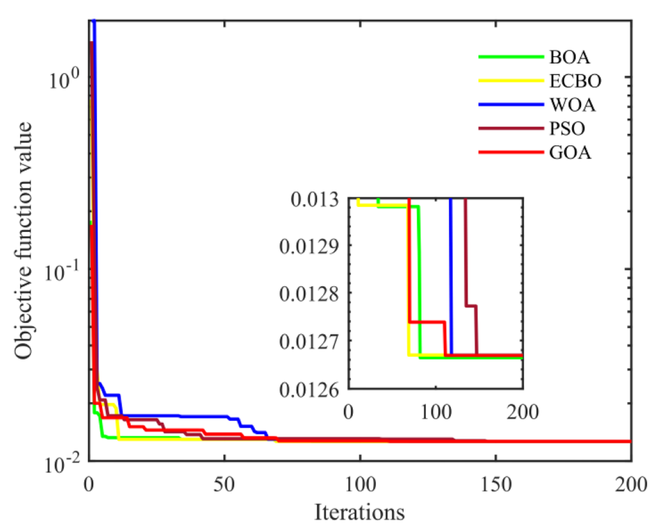

Fig. 15. Comparison of convergence curves obtained for the tension/compression spring problem.

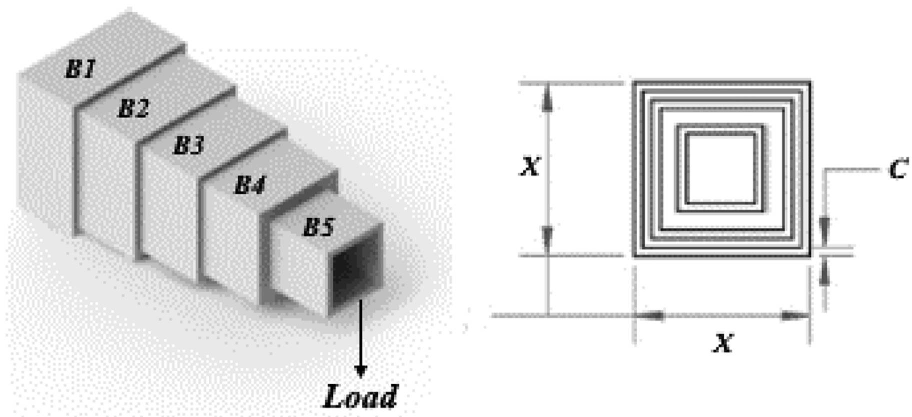

Fig. 16. Schematic of the cantilever beam.

Table 15
Comparison of optimized designs found for the cantilever beam problem.

Algorithm	Optimum Variables					Optimum Cost
	x_1	x_2	x_3	x_4	x_5	
GCA_I [46]	6.0100	5.3000	4.4900	3.4900	2.1500	1.3400
GCA_II [46]	6.0100	5.3000	4.4900	3.4900	2.1500	1.3400
MMA [46]	6.0100	5.3000	4.4900	3.4900	2.1500	1.3400
CS[47]	6.0089	5.3049	4.5023	3.5077	2.1504	1.33999
SOS [48]	6.01878	5.30344	4.49587	3.49896	2.15564	1.33996
GOA	6.010043	4.795002	4.455917	3.529235	2.153681	1.306898
PSO	6.033067	4.819485	4.524332	3.436107	2.131126	1.306913
WOA	5.973752	4.862526	4.486167	3.487862	2.129218	1.306626
ECBO	5.915208	4.908425	4.489401	3.478983	2.149223	1.306733
BOA	5.969612	4.887111	4.461483	3.475294	2.145763	1.306610

Table 16
Comparison of the statistical results for the cantilever beam problem.

Algorithm	Statistical result			
	Best	Average	Worst	Std. dev.
SOS [48]	1.33996	1.33997	N/A	1.1e-5
GOA	1.306898	1.308186	1.313259	0.000121
PSO	1.306913	1.307158	1.313724	0.000841
WOA	1.306626	1.307241	1.312675	0.001431
ECBO	1.306733	1.308635	1.317664	0.002728
BOA	1.306610	1.306812	1.310110	0.000222

$$\vec{v}_{n+N} = \frac{\|\vec{v}_n'\|}{\widehat{B_n^{old} B_n^{new}} \cdot \widehat{B_{n+N}^{old} B_n^{old}}} \widehat{B_{n+N}^{old} B_n^{old}}, \quad n = 1.2.\ 3...N \tag{24}$$

$$\overrightarrow{v_{n+N}'} = \omega\left(1 - \frac{iter}{iter_{max}}\right)(\overrightarrow{v_{n+N}} - \vec{v}_n') \tag{25}$$

where, $\overrightarrow{v_{n+N}'}$ and $\overrightarrow{v_{n+N}}$ are the velocities of the n^{th} cue ball after and before the collision, respectively. B_{n+N}^{old} is the position of the n^{th} cue ball before billiard stick. Here ω is a user defined parameter in interval [0,1) that adjusts the movement of cue balls.

Considering the cue balls' velocities and kinematic relations, updated positions of cue balls are determined as:

$$\overrightarrow{B_{n+N}^{new}} = \frac{\overrightarrow{v_{n+N}'}}{2a} \overrightarrow{v_{n+N}'} + \overrightarrow{B_n^{old}}, \quad n = 1.2.\ 3...N \tag{26}$$

Step 7: Escaping from local optima

The BOA mechanism has exploration ability inherently. In addition, to avoid from being trapped in local optima, an escaping threshold like ET within (0,1) is considered that specifies whether a dimension (variable) of an updated ball must be changed or not. For each updated ball ET is compared with $rand$ which is a random number uniformly distributed within (0,1). If $rand < ET$, a random dimension of the updated ball is regenerated as:

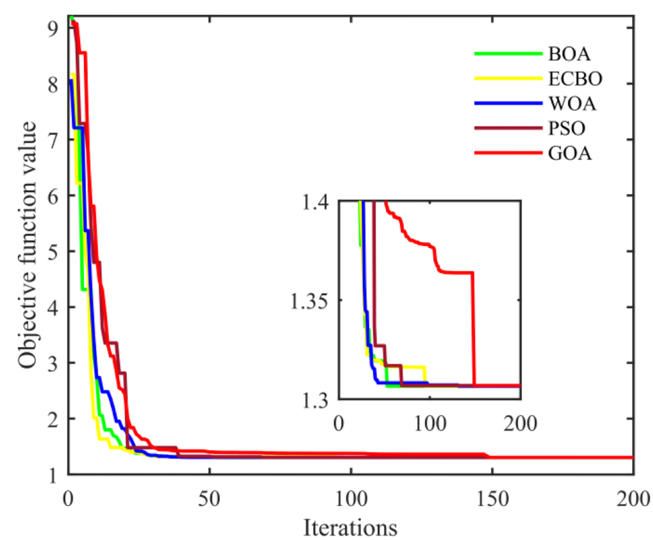

Fig. 17. Comparison of convergence curves obtained for the cantilever beam problem.

$$B_{n.m} = var_m^{min} + rand_{[0,1]}(var_m^{max} - var_m^{min}) \tag{27}$$

Step 8: Checking the boundary condition limitations

In the process of updating the position of balls, the final position of balls might be located outside of the defined range, in other words, the

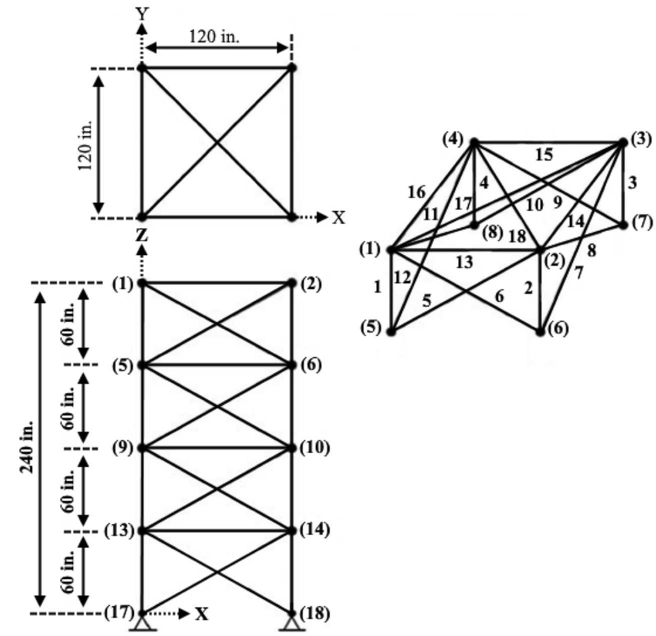

Fig. 18. Schematic of the72-bar truss.

balls may fall out of the pool table. In this case, the dimension of the ball which has violated the boundaries, must be corrected.

Step 9: Checking the termination criterion

The searching process will be terminated after satisfying a specific criterion such as a fixed number of iterations. If the criterion is satisfied, the process will be terminated and the best solution, i.e. the best pocket will be delivered. Otherwise, it will be continued.

3. Sensitivity analysis on the BOA

Sensitivity analysis has a great role in parameter tuning for meta-heuristics which can facilitate the calibration stage by focusing on the sensitive parameters. A sensitivity analysis determines the effect of different values of independent variables on a particular outcome under a given set of assumptions.

BOA has three adjustable parameters, number of pockets, ω and ET. These parameters have significant effects on exploitation and exploration rate. Sensitivity of BOA on variation of these parameters is investigated for Rastrigin function. This function is a non-convex function used as a performance test problem for optimization algorithms. Properties of this function are provided in Table 1 and its perspective view in a limited range of variables is presented in Fig. 4. Variation range of parameters is shown in Table 2. Fig. 5 is prepared to demonstrate the behavior of BOA during this parameter changing process.

4. Mathematical validation of BOA

Validation is an important step in developing a new meta-heuristic algorithm. Exploitation and exploration are two necessary features that will be investigated in this section, besides the convergence behavior of BOA. To evaluate the performance of the proposed approach, 23 well-known mathematical optimization benchmark functions are taken from optimization literature. These functions are classified into three groups: uni-modal, multi-modal, and fixed-dimensional multi-modal. Tables 3–5 summarized these functions and their properties. Also, the typical 2D perspective views of some functions are depicted in Figs. 6–8. In some figures, the ranges of the variables are limited to illustrate the variations.

The BOA algorithm was coded in MATLAB and mathematical test functions were solved with 30 agents and 500 iterations. In order to gather significant statistical results, i.e., average optimized cost functions and corresponding standard deviation, each test function was solved 30 times independently. So, the obtained results were compared with those of ECBO [30], WOA [14], PSO [5], GOA [15], WSA [16], IGWO [31], GA [2], GSA [32], and FEP [33]. This set of algorithms includes well-known and recent ones. Note that some of the comparative statistical results are quoted from Refs. [14,16] and the others are experimented in this research.

4.1. Exploitation ability evaluation

The exploitation of optimum solution which is also known as intensification is defined as the ability of the optimization algorithm to exploit the best solution by searching in the promising regions. Since uni-modal functions have only one global optimum, they deserve to be selected for evaluating the exploitation ability. Statistical results obtained for uni-modal functions are reported in Table 6. Results and comparison with other meta-heuristics confirm that the BOA has good exploitation ability and produces competitive results. More precisely, it

Table 17
Comparison of optimized designs found for the special 72-bar truss structure.

Design variables	Optimum areas (cm^2)								
	FA [50]	CSS-BBBC [51]	TLBO [52]	CBO [53]	GOA	PSO	WOA	ECBO	BOA
A_{1-4}	3.3411	2.845	3.5491	3.7336	4.2891	3.2809	3.2636	3.3939	3.5371
A_{5-12}	7.7587	8.301	7.9676	7.9355	7.8137	7.6495	7.8131	8.1258	7.9216
A_{13-16}	0.6450	0.645	0.645	0.645	0.6450	0.6450	0.6512	0.6452	0.6450
A_{17-18}	0.6450	0.645	0.645	0.645	0.6450	0.6450	0.6453	0.6450	0.6450
A_{19-22}	9.0202	8.202	8.1532	8.3765	8.1199	8.1850	8.3112	7.5922	7.9588
A_{23-30}	8.2567	7.043	7.9667	8.0889	8.2015	8.3453	8.0972	7.7164	7.9266
A_{31-34}	0.6450	0.645	0.645	0.645	0.6450	0.6517	0.6450	0.6450	0.6450
A_{35-36}	0.6450	0.645	0.645	0.645	0.6450	0.6470	0.6450	0.6452	0.6450
A_{37-40}	12.0450	16.328	12.9272	12.9491	12.588	13.570	11.882	12.691	13.053
A_{41-48}	8.0401	8.299	8.1226	8.0524	8.0967	8.0238	8.0233	8.1411	8.0650
A_{49-52}	0.6450	0.645	0.6452	0.645	0.6450	0.6698	0.6511	0.6456	0.6450
A_{53-54}	0.6450	0.645	0.645	0.645	0.6450	0.6450	0.6450	0.6450	0.6450
A_{55-58}	17.3800	15.048	17.0524	16.6629	16.834	16.717	18.385	18.088	17.135
A_{59-66}	8.0561	8.268	8.0618	8.0557	8.0167	8.1260	8.1932	8.1528	8.2072
A_{67-70}	0.6450	0.645	0.645	0.645	0.6450	0.6450	0.6450	0.6450	0.6450
A_{71-72}	0.6450	0.645	0.645	0.645	0.6450	0.6499	0.6450	0.6460	0.6450
Best weight (kg)	327.691	327.507	327.568	327.74	327.897	328.007	327.942	327.845	327.584
Average weight (kg)	329.89	N/A	328.684	328.2	328.324	328.321	328.215	328.152	337.93
Std. dev. (kg)	2.59	N/A	0.73	0.54	1.011	0.992	0.851	0.521	0.848

Table 18
Natural frequencies (Hz) evaluated at the optimum designs of the special 72-bar truss problem.

Frequency number	Natural frequencies (Hz)								
	FA	CSS-BBBC	TLBO	CBO	GOA	PSO	WOA	ECBO	BOA
1	4	4	4	4	4.000	4.000	4.000	4.000	4.000
2	4	4	4	4	4.000	4.000	4.000	4.000	4.000
3	6	6.004	6	6	6.000	6.000	6.001	6.000	6.000
4	6.2468	6.2491	6.2515	6.267	6.284	6.235	6.226	6.226	6.246
5	9.038	8.9726	9.0799	9.101	9.145	9.033	9.044	9.065	9.072

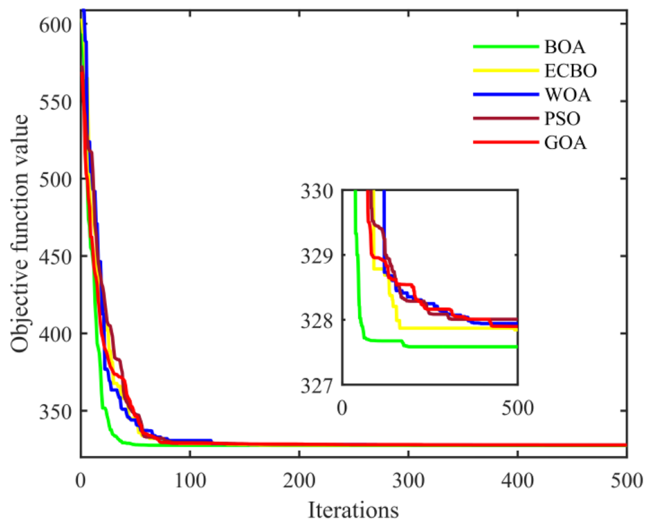

Fig. 19. Comparison of convergence curves obtained for the special 72-bar truss problem.

has the highest efficiency in average cost for functions F1, F2, F4, and F7 and outperforms the majority of algorithms in the other test functions. After BOA, the FEP achieved to get the best average cost in two cases. GA and IGWO are both stood in third place by superiority in one function.

4.2. Exploration ability evaluation

Exploration of the search space, known as diversification, is described as the ability of the optimization algorithm to escape from local optima and explore for a better solution. To assess this feature, functions with multiple local optima, besides one global optimum, can be used. The number of local optima in both multi-modal and fixed dimensional multi-modal functions increases exponentially with the number of variables. It is worth mentioning that fixed dimensional multi-modal functions provide different search space from multi-modal functions because their formulation prevents researchers to adjust the number of variables.

Statistical results obtained by BOA in multi-modal and fixed-dimensional multi-modal functions are shown in Tables 7 and 8, respectively. BOA has found the best average result for functions F9, F10, F11, F15, F16, F19, F21, and F22. In other functions, BOA delivers competitive results. It should be noted that low standard deviations of BOA in almost all cases are evidence of its stability.

4.3. Convergence speed evaluation

Convergence speed to the final solution is another important feature of meta-heuristics. In this section, this feature is evaluated and the results indicate the high and competitive speed of BOA. Fig. 9 demonstrates the average of best so far costs in each iteration over 30 runs for BOA, ECBO, WOA, PSO, and GOA in some cases as an example.

5. Performance of the BOA on constrained engineering problems

Robustness and efficiency of the introduced algorithm were analyzed in previous section. The constrained problems have various constraints that divide the candidate solutions of the algorithms into feasible and infeasible groups. In this section, performance of the BOA is examined on some well-studied constrained engineering problems. These problems have been solved using a wide variety of other meta-heuristics in the literature. The last three problems are sizing optimization of truss structures that must satisfy multiple frequency constraints. It is worth noting that the penalty approach is employed for constraint handling for the sake of simplicity, while other approaches are also applicable.

Problem 1. *Design of a welded beam*

This problem is the optimization of a welded beam. The schematic of the selected beam and its characteristics are represented in Fig. 10. The objective is to find the optimum design variables to minimize the fabrication cost of the beam. An optimum design must satisfy various constraints including shear stress (τ), bending stress (σ), buckling load (P_c), and end deflection (δ). The design variables, $\{x_1, x_2, x_3, x_4\}$ are representative of the thickness of the weld (h), the length of the clamped bar (l), the height of the bar (t), and the thickness of the bar (b), respectively. The mathematical formulation of the problem can be stated as:

Minimize: $f_{cost}\ (X) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14.0 + x_2)$
Subjected to:

$$g_1(X) = \tau(X) - \tau_{max} \leqslant 0$$

$$g_2(X) = \sigma(X) - \sigma_{max} \leqslant 0$$

$$g_3(X) = \delta(X) - \delta_{max} \leqslant 0$$

$$g_4(X) = x_1 - x_4 \leqslant 0$$

$$g_5(X) = P - P_C(X) \leqslant 0$$

$$g_6(X) = 0.125 - x_1 \leqslant 0$$

$$g_7(X) = 0.10471x_1^2 + 0.04811x_3x_4(14 + x_2) - 5 \leqslant 0$$

The ranges of design variables are:

$$0.1 < x_{1,4} < 2$$

$$0.1 < x_{2,3} < 10$$

where:

$$\tau(X) = \sqrt{(\tau')^2 + 2\tau'\tau\ \frac{x_2}{2R} + (\tau)^2};\ \tau' = \frac{P}{\sqrt{2}x_1x_2};\ \tau = \frac{MR}{J};\ M = P\left(L + \frac{x_2}{2}\right);\ R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2}$$

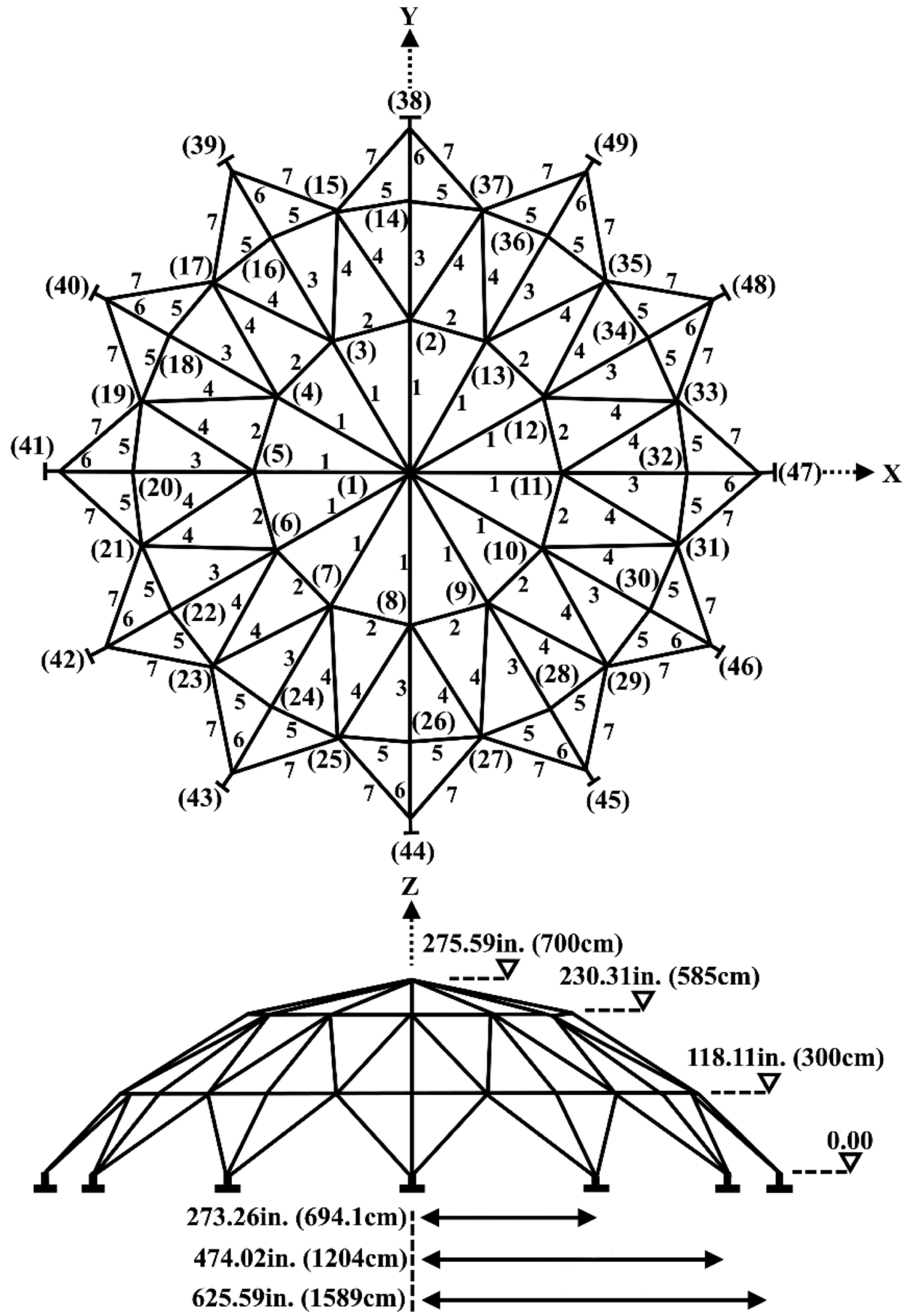

Fig. 20. Schematic of the spatial 120-bar dome-shaped truss structure.

Table 19
Comparison of optimized designs found for the 120-bar dome-shaped truss problem.

Design variable	Optimum areas (cm^2)							
	CSS-BBBC [51]	DPSO [54]	CBO [53]	GOA	PSO	WOA	ECBO	BOA
AG_1	17.478	19.607	19.7738	20.0010	20.44348	20.0010	19.4383	19.6510
AG_2	49.076	41.29	40.6757	39.7517	37.84965	39.7517	40.7565	40.0820
AG_3	12.365	11.136	11.6056	10.3984	10.61059	10.3984	10.6039	10.6041
AG_4	21.979	21.025	21.4601	20.8541	21.2157	20.8541	21.4650	21.3557
AG_5	11.19	10.06	9.8104	9.3987	9.630011	9.3987	9.7224	9.6938
AG_6	12.59	12.758	12.2866	11.7625	11.49923	11.7625	11.9888	11.8605
AG_7	13.585	15.414	15.1417	15.4534	15.31791	15.4534	14.4207	14.6551
Best weight (kg)	9046.34	8890.48	8890.69	8720.574	8726.367	8720.574	8712.55	8710.006
Average weight (kg)	N/A	8895.99	8945.64	8881.198	8921.651	8863.85	9112.21	8810.215
Std. dev. (kg)	N/A	4.26	38.33	35.98	10.32	34.65	37.23	56.854

Table 20
Natural frequencies (Hz) evaluated at the optimum designs of the 120-bar dome-shaped truss problem.

Frequency number	Natural frequencies (Hz)							
	CSS-BBBC	DPSO	CBO	GOA	PSO	WOA	ECBO	BOA
1	9	9.0001	9	9.0000	9.0000	9.0000	9.0002	9.0000
2	11.007	11.0007	1	11.0003	11.0003	11.0000	11.0000	11.0000
3	11.018	11.0053	1	11.0003	11.0003	11.0001	11.0000	11.0000
4	11.026	11.0129	1.01	11.0024	11.0024	11.0001	11.0001	11.0000
5	11.048	11.0471	1.049	11.0692	11.0692	11.0659	11.0659	11.0660

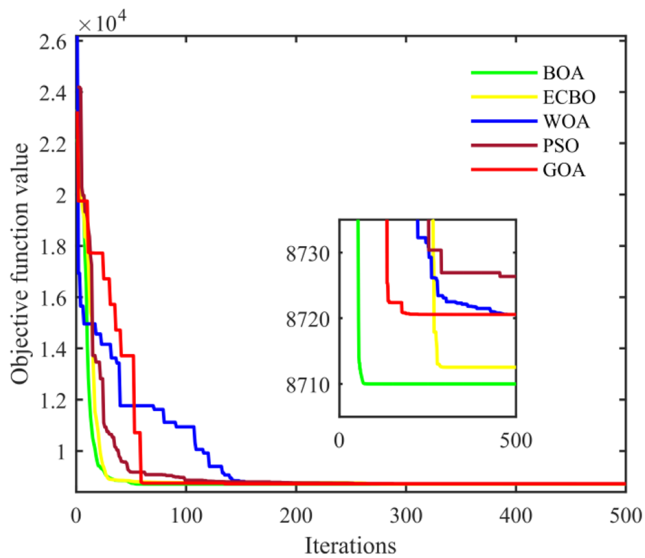

Fig. 21. Comparison of convergence curves obtained for the 120-bar dome problem.

$$J = 2\left\{\sqrt{2}x_1x_2\left[\frac{x_2^2}{12} + \left(\frac{x_1+x_3}{2}\right)^2\right]\right\};\ \sigma(X) = \frac{6PL}{x_4x_3^2};\ \delta(X) = \frac{4PL^3}{Ex_4x_3^3};\ P_c = \frac{4.013E\sqrt{x_3^2x_4^6/36}}{L^2}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right)$$

Also, constants are considered as:

$P = 6000$ lb, $L = 14$ in, E = 30 × 10^6 psi, G = 12 × 10^6 psi, $\tau_{max} = 13{,}600$ psi, $\sigma_{max} = 30{,}000$ psi, and $\delta_{max} = 0.25$ *in*.

This problem has been tackled by many researchers. Ragsdell and Philips [34] employed different mathematical based approaches like APPROX, David, Simplex, and Random in this case. Besides this, many researchers adopted meta-heuristic methods. For example, Deb [35] and Coello [36] utilized GA-based methods. Lee and Geem [37] and Mahdavi et al. [38] applied a different version of HS. He and Wang [39] used effective co-evolutionary particle swarm optimization. The results of the mentioned researches besides other algorithms are listed in Table 9. This table compares the optimum design of the BOA with other methods documented in the literature or experimented in this research. It appears that BOA is one of the top three algorithms. This problem is solved via 40 agents through 500 iterations in 30 independent runs. The statistical results of some algorithms are reported in Table 10. Also, the convergence histories of the GOA, PSO, WOA, ECBO, and BOA are compared in Fig. 11. As can be seen, BOA converges to a better design within a smaller number of iteration.

Problem 2. *Design of the cylindrical pressure vessel*

The optimal design of a cylindrical pressure vessel is the second selected problem. The vessel is designed for a working pressure of 3000 *psi* and a minimum volume of $750ft^3$ regarded the provisions and codes. As shown in Fig. 12, this vessel is capped at both ends by hemispherical heads. The objective of the problem is to minimize the total fabrication cost of the vessel including welding, material, and forming costs. Design variables $\{x_1,x_2,x_3,x_4\}$ are representative of shell thickness (T_s), spherical head thickness (T_h), the inner radius (R), and length of the cylindrical section without head (L), respectively.

The mathematical formulation of the problem is as follows:

Minimize:

$$f_{cost}\ (X) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3$$

Subjected to:

$$g_1(X) = 0.0193x_3 - x_1 \leqslant 0$$

$$g_2(X) = 0.00954x_3 - x_2 \leqslant 0$$

$$g_3(X) = 750 \times 1728 - \pi x_3^2x_4 - \frac{4}{3}\pi x_3^3 \leqslant 0$$

$$g_4(X) = x_4 - 240 \leqslant 0$$

The ranges of design variables are:

$$0 < x_{1,2} < 99$$

$$10 < x_{3,4} < 240$$

Many optimization algorithms have been applied to this problem including mathematical methods [41], GA-based co-evolution model [36], evolution strategy [42], a co-evolutionary PSO [39], colliding bodies optimization [19], charged system search [22], bat algorithm [43], thermal exchange optimization [40], etc. Table 11 compares the optimal solution of BOA with the literature. It can be seen that BOA found the best design overall which is about 0.02% less than TEO best result. Present work is performed using 40 search agents through 2000 iterations in 30 independent runs. Table 12 shows the statistical results yielded by various algorithms. It appears that BOA performed well in terms of best and average optimized cost. The convergence histories of the employed algorithms in this problem are shown in Fig. 13.

Problem 3. *Design of a tension/compression spring*

This problem is concerned with the design of a tension/compression spring in such a way that its weight is minimum, subject to constraints on shear stress, surge frequency, and deflection. Fig. 14 shows the schematic of this spring. Design variables,$\{x_1,x_2,x_3\}$ are representative of the wire diameter (d), the mean coil diameter (D), and the number of active coils (N), respectively. This problem can be formulated as follows:

Minimize: $f_{cost}\ (X) = (x_3 + 2)x_2x_1^2$

Subjected to:

$$g_1(X) = 1 - \frac{x_2^3x_3}{71785x_1^4} \leqslant 0$$

$$g_2(X) = \frac{4x_2^2 - x_1x_2}{12566(x_2x_1^3 - x_1^4)} + \frac{1}{5108x_1^2} - 1 \leqslant 0$$

$$g_3(X) = 1 - \frac{140.45x_1}{x_2^2x_3} \leqslant 0$$

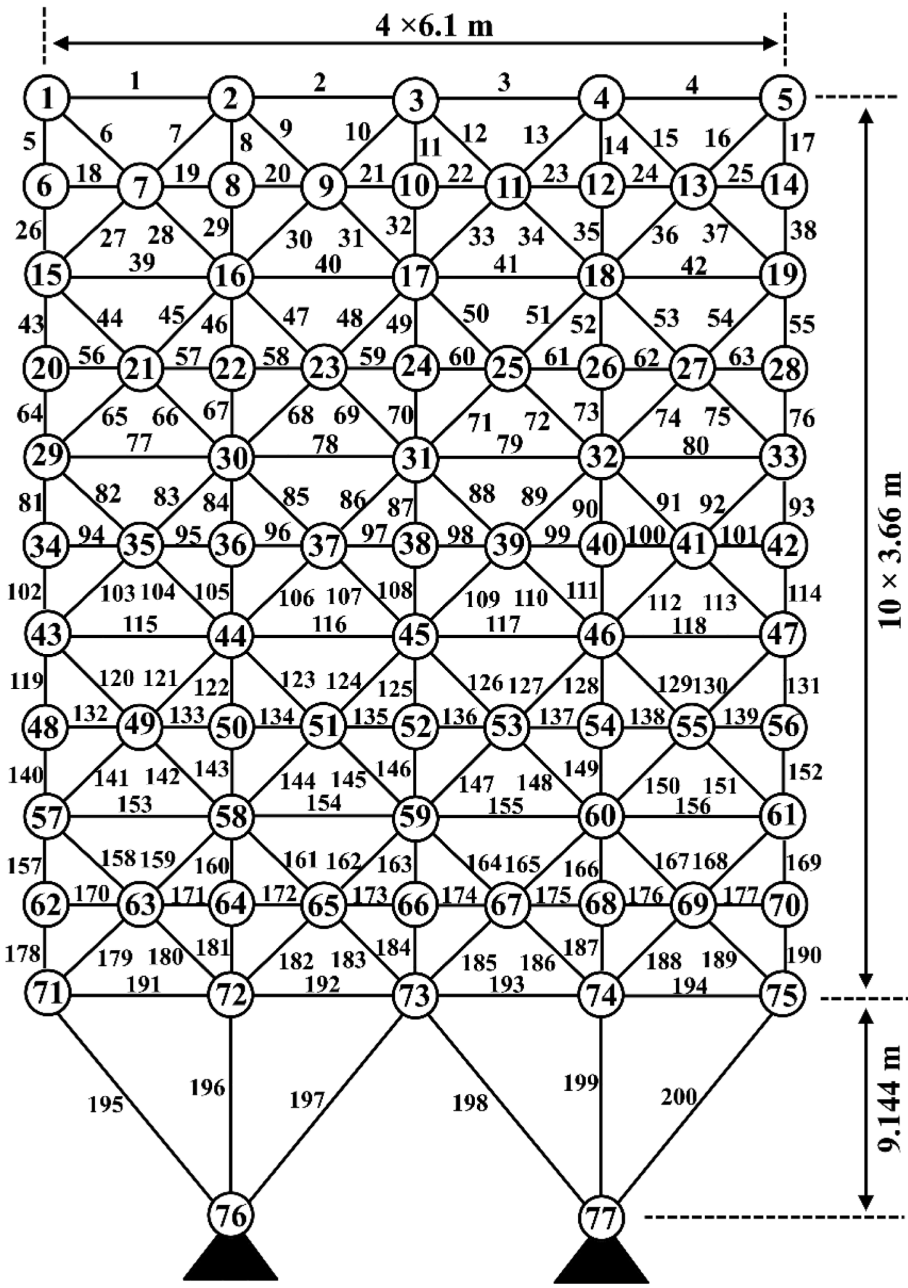

Fig. 22. Schematic of the 200-bar truss structure.

$$g_4(X) = \frac{x_1 + x_2}{1.5} - 1 \leqslant 0$$

The ranges of design variables are:

$0.05 \leqslant x_1 \leqslant 2$

$0.25 \leqslant x_2 \leqslant 1.3$

$2 \leqslant x_3 \leqslant 15$

The optimal solution of BOA is compared with the literature algorithms in Table 13. From this table, it can be seen that the best design found by BOA is better than the others, except CSS. The improvement achieved by BOA is marginal concerning the solutions of BA and TEO. Table 14 shows the literature statistical results and results of this research after 30 independent runs. In this case, the number of agents and the maximum number of iterations are considered as 40 and 200, respectively. According to Table 14, BOA achieved a good performance in terms of best and average optimized cost. Fig. 15 shows that ECBO and BOA had a close compete in finding optimal design and eventually BOA prospered.

Problem 4. *Design of a cantilever beam*

This problem is focused on minimizing the weight of a cantilever beam with five hollow square blocks. As shown in Fig. 16, the first block is rigidly supported and a vertical load exerted on the fifth. Design variables $\{x_1, x_2, x_3, x_4, x_5\}$ define the dimensions of the cross-section of the cubes, respectively. The problem is formulated using classical beam theory as below:

Minimize: $f_{cost}\ (X) = 0.0624(x_1 + x_2 + x_3 + x_4 + x_5)$

Subjected to: $g_1(X) = \frac{61}{x_1^3} + \frac{27}{x_2^3} + \frac{19}{x_3^3} + \frac{7}{x_4^3} + \frac{1}{x_5^3} - 1 \leqslant 0$

The ranges of design variables are:

$0.01 < x_{1,2,3,4,5} < 100$

This case is not widely employed by researchers but it deserves to be used as a benchmark in this field. Chickermane and Gea [46], Gandomi

Table 21
Comparison of optimized designs found for the 200-bar truss problem.

Design variable	Optimum areas (cm^2)							
	CSS-BBBC [51]	TLBO [52]	CBO [53]	GOA	PSO	WOA	ECBO	BOA
A_{1-4}	0.2934	0.303	0.3059	0.2961	0.4877	0.3501	0.2866	0.2981
$A_{5,8,11,14,17}$	0.5561	0.4479	0.4476	0.4569	0.2706	0.4250	0.4853	0.4736
$A_{19,20,21-24}$	0.2952	0.1001	0.1	0.1003	0.1000	0.1004	0.1028	0.1008
$A_{18,25-56,63,94,101,132,139,170,177}$	0.197	0.1	0.1001	0.1000	0.1000	0.1020	0.1025	0.1000
$A_{26,29,32,35,38}$	0.834	0.5124	0.4944	0.5039	0.6135	0.4829	0.5598	0.5345
$A_{6-7,9,10,12,13,15-16,27-28,30-31,33-34,36-37}$	0.6455	0.8205	0.8369	0.8386	0.8708	0.8299	0.8245	0.8229
A_{39-42}	0.177	0.1	0.1001	0.1035	0.1081	0.1045	0.1021	0.1000
$A_{43,46,49,52,55}$	1.4796	1.4499	1.5514	1.4398	1.6976	1.4489	1.4375	1.3812
A_{57-62}	0.4497	0.1001	0.1	0.1000	0.1000	0.1016	0.1000	0.1000
$A_{64,67,70,73,76}$	1.4556	1.5955	1.5286	1.5895	1.7198	1.6199	1.6424	1.6066
$A_{44-45,47-48,50-51,53-54,65-66,68-69,71-72,74-75}$	1.2238	1.1556	1.1547	1.1486	1.1452	1.1553	1.1579	1.1581
A_{77-80}	0.2739	0.1242	0.1	0.1202	0.1000	0.1000	0.1009	0.1000
$A_{81,84,87,90,93}$	1.9174	2.9753	2.998	3.0466	2.9066	2.7375	3.1157	3.0708
A_{95-100}	0.117	0.1	0.1017	0.1004	0.1003	0.1000	0.1000	0.1044
$A_{102,105,108,111,114}$	3.5535	3.2553	3.2475	3.2331	3.3081	3.1764	3.2212	3.2326
$A_{82-83,85-86,88-89,91-92,103-104,106-107,109-110,112-113}$	1.336	1.5762	1.5213	1.5966	1.6180	1.5712	1.5359	1.6082
$A_{115-118}$	0.6289	0.268	0.3996	0.1004	0.3339	0.2305	0.3061	0.2512
$A_{119,122,125,128,131}$	4.8335	5.0692	4.7557	5.3762	5.0458	5.1743	5.4883	5.1247
$A_{133-138}$	0.6062	0.1	0.1002	0.1229	0.1000	0.1007	0.1000	0.1035
$A_{140,143,146,149,152}$	5.4393	5.4281	5.1359	5.9681	5.3249	5.3179	5.1982	5.3433
$A_{120-124,126-127,129-130,141-142,144-145,147-148,150-151}$	1.8435	2.0942	2.1181	2.0762	2.1984	2.0051	2.2095	2.1805
$A_{153-156}$	0.8955	0.6985	0.92	0.1000	0.8336	0.5832	1.0544	0.6994
$A_{157,160,163,166,169}$	8.1759	7.6663	7.3084	7.6385	7.7886	7.8175	7.4783	7.2997
$A_{171-176}$	0.3209	0.1008	0.1185	0.2509	0.2798	0.1000	0.1018	0.1071
$A_{178,181,184,187,190}$	10.98	7.9899	7.6901	8.0899	8.0564	8.0421	6.7824	7.8809
$A_{158-159,161-162,164-165,167-168,179-180,182-183,185-186,188-189}$	2.9489	2.8084	3.0895	2.7349	2.9365	2.6096	3.0148	2.8910
$A_{191-194}$	10.5243	10.4661	10.6462	10.5580	10.0359	11.1607	10.5456	10.5500
$A_{195,197,198,200}$	20.4271	21.2466	20.719	21.6723	20.7479	21.9486	20.9355	21.1919
$A_{196,199}$	19.0983	10.734	1.7463	10.2576	11.2706	9.8345	11.1289	10.4725
Best weight (kg)	2298.61	2156.541	2161.15	2163.939	2172.157	2159.729	2163.06	2157.82
Average weight (kg)	N/A	2157.547	2447.52	2275.651	2198.213	2238.158	2330.78	2162.23
Std. dev. (kg)	N/A	1.545	301.29	265.21	19.52	260.38	361.66	203.55

Table 22
Natural frequencies (Hz) evaluated at the optimum designs of the 200-bar truss problem.

Frequency number	Natural frequencies (Hz)							
	CSS-BBBC	TLBO	CBO	GOA	PSO	WOA	ECBO	BOA
1	5.01	5	5	5.000	5.000	5.000	5.0000	5.0001
2	12.911	12.2171	12.221	12.167	13.280	12.530	12.2001	12.2295
3	15.416	15.03796	15.088	15.041	15.000	15.047	15.1745	15.1074
4	17.033	16.70476	16.759	16.599	17.125	16.679	16.7822	16.6725
5	21.426	21.40164	21.419	21.372	21.171	21.407	21.2865	21.4076

et al. [47], and Cheng and Prayogo [48] have investigated this problem in the literature. The optimal designs given in Table 15 indicate that BOA achieved a solution that outperforms all other methods. After BOA, WOA and ECBO are ranked second and third, respectively. Table 16 demonstrates statistical results of the considered algorithms over 30 runs via 20 agents after 200 iterations, versus Symbiotic Organisms Search (SOS) statistics. The standard deviation of SOS is much better than BOA, but BOA is superior in terms of best and average results. The convergence curves of employed meta-heuristics are compared in Fig. 17.

Problem 5. *Design of the special 72-bar truss*

Truss members sustain either tensile or compressive forces. Sizing optimization of truss structures is a common structural design problem including various constraints such as displacements, stress, buckling, and natural frequencies [49]. Cross-sectional areas of members are design variables of this problem, x_i. The objective function is to minimize the weight of structures as formulated below:

$$\text{Minimize: } f_{cost}(X) = \sum_{i=1}^{ng} x_i \sum_{j=1}^{nm(i)} \rho \ L_j$$

Subjected to:

$$g_j(X) \leqslant 0, \quad j = 1,2,3,\ldots,nc$$

$$x_{min} \leqslant x_i \leqslant x_{max}$$

where, ng is the number of design groups; $nm(i)$ is the number of members for the i^{th} group; L_j presents the length of the j^{th} member; and ρ is the material density. $g_j(X)$ denote the design constraints, and nc is the number of the constraints. x_{min} and x_{max} are minimum and maximum values of cross-sectional areas, respectively.

The schematic of the spatial 72-bar truss taken as fifth test problem is shown in Fig. 18. Because of structural symmetry the elements are classified into 16 groups. The material density and the elastic modulus are considered 2767.99 kg/m^3 and 68.95 GPa for all the members, respectively. A load of magnitude 2268 kg is attached to the nodes 1 through 4 as nonstructural masses. The allowable cross-sectional area of all elements is between 0.645 cm^2 and 20 cm^2. The natural frequency constraints are as:

$f_1 = 4$ Hz and $f_3 \geqslant 6$ Hz.

Table 17 lists the optimal designs and the statistical results found by litriture algorithm versus BOA. The lightest design belongs to CSS-BBBC

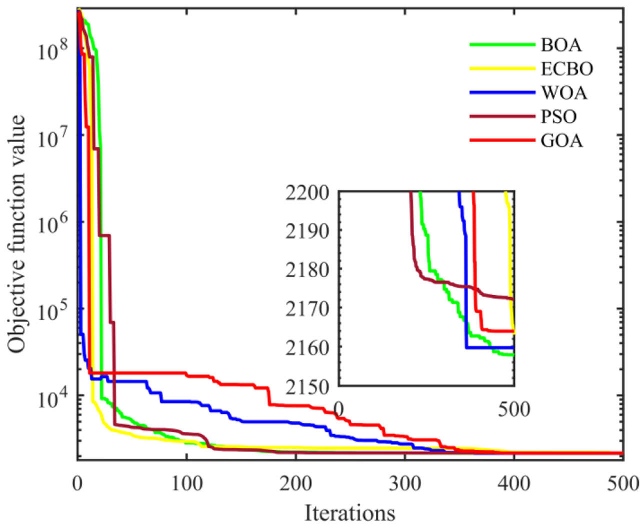

Fig. 23. Comparison of convergence curves obtained for the 200-bar truss problem.

and BOA stands in second place. Table 18 shows natural frequencies at optimum design of used algorithms. Fig. 19 shows the convergence curves of employed meta heuristic and illustrates that, already in the initial iterations, BOA has found the lighter design and finally has converged to the best design. It is noteworthy that in this study, the truss sizing optimization problems are solved via 40 agents through 500 iterations in 30 independent runs.

Problem 6. *Design of the* 120-*bar dome-shaped truss structure*

In this test problem, the dome-shaped truss schematized in Fig. 20 is selected for optimization. The members are divided into seven groups considering structural symmetry. The material density and the elastic modulus are considered as 7971.81 kg/m^3 and 210 GPa for all the members, respectively. Loads 3000 kg at node 1, 500 kg at nodes 2–13, and 100 kg at the other nodes are added. The range of cross-sectional areas is from 1 cm^2 to 129.3 cm^2. The natural frequency constraints are as:

$f_1 \geqslant 9$ Hz and $f_2 \geqslant 11$ Hz.

Optimization results are compared in Table 19, In this case, BOA found the best design overall. From Table 20, it can be seen that none of the frequency constraints are violated at optimum designs. Fig. 21 depicts convergence histories of the algorithms for obtaining optimal designs.

Problem 7. *Design of the planar* 200-*bar truss structure*

The last test problem regards the optimal design of a planar 200-bar. The schematic of this structure is shown in Fig. 22. Because of structural symmetry, elements are categorized into 29 groups. The material density and the elastic modulus are considered as 7860 kg/m^3 and 210 GPa for all the members, respectively. Like in the previous cases, some nonstructural masses are attached to the structure. In this case, 100 kg are imposed to upper nodes. Lower bound of cross-sectional areas is limited to 0.1 cm^3. Three natural frequency constraints are as:

$f_1 \geqslant 5$ Hz, $f_2 \geqslant 10$ Hz and $f_3 \geqslant 15$ Hz.

Table 21 contains the optimal solutions obtained by several algorithms. The best weight of BOA is 2157.82 kg while it is 2156.541 kg for TLBO. Table 22 confirms that all the considered methods satisfy frequencies constraints. The good convergence behavior of the BOA in comparison to the other algorithms can be seen from Fig. 23.

6. Conclusions

In this study, inspired from billiards game, a new meta-heuristic is developed which is named as Billiards-inspired Optimization Algorithm (BOA). Several physics laws are involved in this game, including momentum and energy conservations. These laws plus vector algebra form the basis of BOA. The proposed algorithm has a simple formulation that makes it easy to implement. To evaluate the exploitation and exploration ability of BOA, an extensive investigation is performed on some mathematical optimization functions. The BOA results were compared with those of ECBO, WOA, PSO, GOA, WSA, IGWO, GA, GSA, and FEP. This is a major challenge in solving real problems. For further investigation, seven constrained engineering problems are solved in this study. Comparisons of BOA's optimization results with those of the other methods confirm applicability, efficiency and superiority of the BOA to find the optimal solution. As future research, BOA can be applied to various real problems and can be compared with other meta-heuristic algorithms. It is worth noting that a multi-objective version of BOA is also under development.

Declaration of Competing 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] Mirjalili S, Mirjalili SM, Lewis A. Grey wolf optimizer. Adv Eng Softw 2014;69:46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007.
[2] Holland JH. Adaptation in Natural and Artificial Systems. USA: University of Michigan Press; 1975.
[3] Goldberg D. Genetic Algorithms in Search, Optimization, and Machine Learning, Addison Wesley, Reading, MA. 1989.
[4] Rechenberg I. Evolutionsstrategien Berlin Heidelberg: Springer; 1978. p. 83–114. https://doi.org/10.1007/978-3-642-81283-5_8.
[5] Kennedy J, Eberhart R. Particle swarm optimization. Neural networks. Proceedings IEEE Int Conf 1995. p. 1942–8. https://doi.org/10.1109/ICNN.1995.488968.
[6] Dorigo M, Birattari M, Stutzle T. Ant colony optimization. IEEE Comput Intell Mag 2006;1:28–39. https://doi.org/10.1109/MCI.2006.329691.
[7] Basturk B, Karaboga D. An artificial bee colony (ABC) algorithm for numeric function optimization. IEEE Swarm Intell Symp Indianapolis 2006:12–4.
[8] Yang X-S, Deb S. Engineering optimisation by cuckoo search. Int J Math Model Numer Optim 2010;1:330–43. https://doi.org/10.1504/IJMMNO.2010.035430.
[9] Yang X-S. A New Metaheuristic Bat-Inspired Algorithm. Studies. Berlin, Heidelberg: Springer; 2010.
[10] Yang XS. Firefly algorithm, stochastic test functions and design optimisation. Int J Bio-Inspired Comput 2010;2:78. https://doi.org/10.1504/IJBIC.2010.032124.
[11] Kaveh A, Farhoudi N. A new optimization method: Dolphin echolocation. Adv Eng Softw 2013;59:53–70. https://doi.org/10.1016/j.advengsoft.2013.03.004.
[12] Duman E, Uysal M, Alkaya AF. Migrating Birds Optimization: a new metaheuristic approach and its performance on quadratic assignment problem. Inf Sci (Ny) 2012;217:65–77. https://doi.org/10.1016/j.ins.2012.06.032.
[13] Mirjalili S. Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm. Knowledge-Based Syst 2015;89:228–49. https://doi.org/10.1016/j.knosys.2015.07.006.
[14] Mirjalili S, Lewis A. The whale optimization algorithm. Adv Eng Softw 2016;95:51–67. https://doi.org/10.1016/j.advengsoft.2016.01.008.
[15] Saremi S, Mirjalili S, Lewis A. Grasshopper Optimisation Algorithm: theory and application. Adv Eng Softw 2017;105:30–47. https://doi.org/10.1016/j.advengsoft.2017.01.004.
[16] Kaveh A, Dadras Eslamlou A. Water strider algorithm : A new metaheuristic and applications 2020;25:520–41. https://doi.org/10.1016/j.istruc.2020.03.033.
[17] Kirkpatrick S, Gelatt CD, Vecchi MP. Optimization by Simulated Annealing. New Ser 1983;220:671–80.
[18] Erol OK, Eksin I. A new optimization method: big bang-big crunch. Adv Eng Softw 2006;37:106–11. https://doi.org/10.1016/J.ADVENGSOFT.2005.04.005.
[19] Kaveh A, Mahdavi VR. Colliding bodies optimization: a novel meta-heuristic method. Comput Struct 2014;139:18–27. https://doi.org/10.1016/j.compstruc.2014.04.005.
[20] Kaveh A, Khayatazad M. A new meta-heuristic method: Ray Optimization. Comput Struct 2012;112–113:283–94. https://doi.org/10.1016/j.compstruc.2012.09.003.
[21] Du H, Wu X, Zhuang J. Small-world optimization algorithm for function optimization Berlin Heidelberg: Springer; 2006. p. 264–73. https://doi.org/10.1007/11881223_33.
[22] Kaveh A, Talatahari S. A novel heuristic optimization method : charged system 2010;289:267–89. DOI:10.1007/s00707-009-0270-4.
[23] Mirjalili S, Mirjalili SM, Hatamlou A. Multi-Verse Optimizer: a nature-inspired algorithm for global optimization. Neural Comput Appl 2016;27:495–513. https://doi.org/10.1007/s00521-015-1870-7.
[24] Alciatore D. The amazing world of billiards physics 2007.

[25] Briesath J. Billiards: the official rules & records book. Iowa: Billiard Congress of America; 1992.
[26] Mathavan S, Jackson MR, Parkin RM. A theoretical analysis of billiard ball dynamics under cushion impacts. Proc Inst Mech Eng Part C J Mech Eng Sci 2010;224:1863–73. https://doi.org/10.1243/09544062JMES1964.
[27] Landry J-F, Dussault J-P. AI optimization of a billiard player. J Intell Robot Syst 2007;50:399–417. https://doi.org/10.1007/s10846-007-9172-7.
[28] Teng C. A Mathematical Analysis of Billiards Games : Amplification Factor, Stability of Ball Path, and Player ' s Pattern. Citeseer 2008.
[29] Čepič M. Elastic collisions of smooth spherical objects: finding final velocities in four simple steps. Am J Phys 2019;87:200–7. https://doi.org/10.1119/1.5089753.
[30] Kaveh A, Ilchi Ghazaaan M. Enhanced colliding bodies optimization for design problems with continuous and discrete variables. Adv Eng Softw 2014;77:66–75. https://doi.org/10.1016/j.advengsoft.2014.08.003.
[31] Kaveh A, Zakian P. Improved GWO algorithm for optimal design of truss structures. Eng Comput 2018;34:685–707. https://doi.org/10.1007/s00366-017-0567-1.
[32] Rashedi E, Nezamabadi-pour H, Saryazdi S. GSA: A Gravitational Search Algorithm. Inf Sci (Ny) 2009;179:2232–48. DOI:10.1016/J.INS.2009.03.004.
[33] Yao X, Liu Y, Lin G. Evolutionary programming made faster. IEEE Trans Evol Comput 1999;3:82–102. https://doi.org/10.1109/4235.687880.
[34] Ragsdell KM, Phillips DT. Optimal design of a class of welded structures using geometric programming. J Eng Ind 1976;98:1021. https://doi.org/10.1115/1.3438995.
[35] Deb K. Optimal design of a welded beam via genetic algorithms. AIAA J 1991;29:2013–5. https://doi.org/10.2514/3.10834.
[36] Coello Coello CA. Use of a self-adaptive penalty approach for engineering optimization problems. Comput Ind 2000;41:113–27. https://doi.org/10.1016/S0166-3615(99)00046-9.
[37] Lee KS, Geem ZW. A new structural optimization method based on the harmony search algorithm. Comput Struct 2004;82:781–98. https://doi.org/10.1016/J.COMPSTRUC.2004.01.002.
[38] Mahdavi M, Fesanghary M, Damangir E. An improved harmony search algorithm for solving optimization problems. Appl Math Comput 2007;188:1567–79. https://doi.org/10.1016/J.AMC.2006.11.033.
[39] He Q, Wang L. An effective co-evolutionary particle swarm optimization for constrained engineering design problems. Eng Appl Artif Intell 2007;20:89–99. https://doi.org/10.1016/J.ENGAPPAI.2006.03.003.
[40] Kaveh A, Dadras A. A novel meta-heuristic optimization algorithm: thermal exchange optimization. Adv Eng Softw 2017;110:69–84. https://doi.org/10.1016/j.advengsoft.2017.03.014.
[41] Sandgren E. Nonlinear integer and discrete programming in mechanical design optimization. J Mech Des 1990;112:223. https://doi.org/10.1115/1.2912596.
[42] Mezura-Montes E, Coello CAC. A simple multimembered evolution strategy to solve constrained optimization problems. IEEE Trans Evol Comput 2005;9:1–17. https://doi.org/10.1109/TEVC.2004.836819.
[43] Gandomi AH, Yang X-S, Alavi AH, Talatahari S. Bat algorithm for constrained optimization tasks. Neural Comput Appl 2013;22:1239–55. https://doi.org/10.1007/s00521-012-1028-9.
[44] Huang F, Wang L, He Q. An effective co-evolutionary differential evolution for constrained optimization. Appl Math Comput 2007;186:340–56. https://doi.org/10.1016/J.AMC.2006.07.105.
[45] Belegundu AD. A study of mathematical programming methods for structural optimization John Wiley & Sons Ltd; 1982. https://doi.org/10.1002/nme.1620210904.
[46] Chickermane H, GEA HC. Structural Optimization Using a New Local Approximation Method. Int J Numer Methods Eng 2002;39:829–46. DOI:10.1002/(sici)1097-0207(19960315)39:5 < 829::aid-nme884 > 3.0.co;2-u.
[47] Gandomi AH, Yang X-S, Alavi AH. Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. Eng Comput 2013;29:17–35. https://doi.org/10.1007/s00366-011-0241-y.
[48] Cheng M-Y, Prayogo D. Symbiotic Organisms Search: a new metaheuristic optimization algorithm. Comput Struct 2014;139:98–112. https://doi.org/10.1016/J.COMPSTRUC.2014.03.007.
[49] Kaveh A, Ilchi Ghazaaan M. Meta-heuristic algorithms for optimal design of real-size structures Springer International Publishing; 2018. https://doi.org/10.1007/978-3-319-78780-0.
[50] Miguel LFF, Miguel LFF. Expert Systems with Applications Shape and size optimization of truss structures considering dynamic constraints through modern metaheuristic algorithms 2012;39:9458–67. DOI:10.1016/j.eswa.2012.02.113.
[51] Kaveh A, Zolghadr A. Truss optimization with natural frequency constraints using a hybridized CSS-BBBC algorithm with trap recognition capability. Comput Struct 2012;102–103:14–27. https://doi.org/10.1016/j.compstruc.2012.03.016.
[52] Farshchin M, Camp CV, Maniat M. Optimal design of truss structures for size and shape with frequency constraints using a collaborative optimization strategy. Expert Syst Appl 2016;66:203–18. https://doi.org/10.1016/j.eswa.2016.09.012.
[53] Kaveh A, Ilchi Ghazaaan M. Vibrating particles system algorithm for truss optimization with multiple natural frequency constraints. Acta Mech 2017;228:307–22. https://doi.org/10.1007/s00707-016-1725-z.
[54] Kaveh A, Zolghadr A. Democratic PSO for truss layout and size optimization with frequency constraints. Comput Struct 2014;130:10–21. https://doi.org/10.1016/j.compstruc.2013.09.002.