Knowledge-Based Systems 251 (2022) 109215

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Beluga whale optimization: A novel nature-inspired metaheuristic algorithm

Changting Zhong [a], Gang Li [a,*], Zeng Meng [b]

[a] Department of Engineering Mechanics, State Key Laboratory of Structural Analyses for Industrial Equipment, Dalian University of Technology, Dalian 116024, China
[b] School of Civil Engineering, Hefei University of Technology, Hefei 230009, China

ARTICLE INFO

Article history:
Received 6 May 2021
Received in revised form 7 April 2022
Accepted 3 June 2022
Available online 9 June 2022

Keywords:
Metaheuristics
Beluga whale optimization
Optimization
Swarm intelligence

ABSTRACT

In this paper, a novel swarm-based metaheuristic algorithm inspired from the behaviors of beluga whales, called beluga whale optimization (BWO), is presented to solve optimization problem. Three phases of exploration, exploitation and whale fall are established in BWO, corresponding to the behaviors of pair swim, prey, and whale fall, respectively. The balance factor and probability of whale fall in BWO are self-adaptive which play significant roles to control the ability of exploration and exploitation. Besides, the Levy flight is introduced to enhance the global convergence in the exploitation phase. The effectiveness of the proposed BWO is tested using 30 benchmark functions, with qualitative, quantitative and scalability analysis, and the statistical results are compared with 15 other metaheuristic algorithms. According to the results and discussion, BWO is a competitive algorithm in solving unimodal and multimodal optimization problems, and the overall rank of BWO is the first in the scalability analysis of benchmark functions among compared metaheuristic algorithms through the Friedman ranking test. Finally, four engineering problems demonstrate the merits and potential of BWO in solving complex real-world optimization problems. The source code of BWO is currently available to public: https://ww2.mathworks.cn/matlabcentral/fileexchange/112830-beluga-whale-optimization-bwo/.

1. Introduction

Over the last decades, the requirements for optimization techniques becomes more and more evident with the ever-increasing complexity and difficulty of real-world optimization problem. Metaheuristic algorithm is a class of stochastic search approach which exhibits great performance in dealing with multimodal, non-continuous, and non-differentiable problems [1–4]. Therefore, metaheuristic algorithms are very popular which have been widely used in solving real-world optimization problems in different fields, such as cloud computing [5,6], scheduling [7], neural network [8], feature selection [9], image segmentation [10], fuzzy control [11], photovoltaic models [12], civil engineering [13,14], reliability-based design [15], and so on.

Four characteristics can be summarized in metaheuristic algorithms: simplicity, flexibility, derivation-free mechanism, and local optima avoidance [16]. First, the metaheuristic algorithm is usually inspired from the phenomenon of nature, evolutionary, behaviors of animals or humans, which is based on a well-understand concept and easy to implement. Second, metaheuristic algorithm is flexible to deal with different types of optimization problems, due to the black box for any objective problem. Third, most metaheuristic algorithms do not require derivative information during the optimization process, which make highly acceptable for problems with unknown gradient information and convenient to non-embedded analysis combined with the commercial finite element software. Finally, the metaheuristic algorithm is suitable for solving challenging optimization problems with a great number of local optima, especially for real-world optimization problems.

In recent years, several representative algorithms are highly concerned in the community of metaheuristic, such as particle swarm optimization (PSO) [17], genetic algorithm (GA) [18], differential evolution (DE) [19]. It should be noted that the inspiration of metaheuristic from nature is different. For instance, PSO is inspired by the behavior of flocking birds or fishes, who are represented as the candidate solutions by traveling in the search space. GA mimics the evolutionary process by the stochastic behaviors of selection, reproduction and mutation. A great many of novel metaheuristics have been developed and applied in different fields, and the comprehensive literature is provided in Section 2.

* Corresponding author.
E-mail addresses: zhongct@dlut.edu.cn (C. Zhong), ligang@dlut.edu.cn (G. Li), mengz@hfut.edu.cn (Z. Meng).

https://doi.org/10.1016/j.knosys.2022.109215
0950-7051/

For each metaheuristic algorithm, it is difficult to balance the exploration and exploitation to find the global optimum. Exploration phase controls the global search space, related to the escaping from local search or local optima stagnation. Exploitation is the capacity of local search to improve the performance. In general, metaheuristic algorithms have different operators and mechanisms, so metaheuristic algorithms provide different performances in solving optimization problems due to the different searching ability for the exploration and exploitation.

According to the No Free Lunch (NFL) theorem [20], the average performance of all metaheuristic algorithms is the same when solving all optimization problems. In other words, the best algorithm which can solve all optimization problems does not exist, implies that when the prior knowledge (algorithmic parameters, convergence criterion) is given for a metaheuristic algorithm to solve a specific problem, the performances for different algorithms are not equal. It is still a challenge to find the most suitable algorithm for each specific type of optimization problem. Each metaheuristic algorithm has different characteristics due to the different inspirations from natural or biological behaviors. A metaheuristic algorithm needs comprehensive tests from a lot of benchmark functions and real-world applications in different fields, in order to evaluate the performance and find suitable application range with continuous improvement. The above reasons support the innovation and design of metaheuristic algorithms to solve different optimization problems.

In this paper, a novel metaheuristic algorithm, named Beluga Whale Optimization (BWO), is presented for solving optimization problems. BWO is a swarm-based algorithm which is inspired from the behaviors of beluga whales, including swim, prey and whale fall in the sea. The exploration, exploitation and whale fall phase are constructed in the mathematical model of BWO, and the Levy flight function is utilized in the exploitative phase to increase the convergence ability of BWO. The effectiveness and robustness of BWO are tested with 30 benchmark functions problems and 4 real-world optimization problems. The performance of BWO is compared with 15 different metaheuristic algorithms, while the qualitative, quantitative and scalability analysis of BWO are implemented.

The rest of this paper is organized as follows. Section 2 introduces the related works of metaheuristic algorithms. Section 3 represents the inspiration and mathematical model of BWO. The test of benchmark functions and engineering problems for BWO and the discussion on comparing different metaheuristic algorithms are presented in Section 4. Finally, the conclusions are summarized in Section 5.

2. Literature review

The state-of-the-art of metaheuristics are discussed in this section. In recent years, a great number of metaheuristic algorithms are presented and investigated, and they can be mainly classified into four categories [21–23]: (1) swarm-based algorithms, simulating the intelligence of swarms; (2) evolutionary-based algorithms, inspired from the evolutionary phenomenon in nature; (3) physics or chemistry-based algorithms, inspired from the physical phenomenon or chemistry; (4) social or human-based algorithms, inspired from human or social behaviors. A summary of well-known and recently metaheuristic algorithms is listed in Table 1.

In the swarm-based metaheuristic algorithms, PSO is the most popular algorithm proposed by Kennedy and Eberhart [27] in 1995. Ant colony optimization (ACO) is another popular and classical metaheuristic algorithm presented by Dorigo et al. [29] which is inspired from the foraging behaviors of ants, based on the communication of chemical pheromone trails to find the shortest paths between their locations and food sources. Artificial bee colony (ABC) algorithm [34] was inspired from the foraging behaviors of bees consisted of three groups: employed bees, onlooker bees and scouts. Several recently developed swarm-based metaheuristic algorithms also have been attracted much attention. Grey wolf optimizer (GWO) was presented by Mirjalili et al. [48] inspired from the foraging behavior of grey wolves considering four groups: alpha, beta, delta, and omega, and the cooperative hunting in grey wolfs is simulated. Whale optimization algorithm (WOA) [22] was presented to mimic the foraging behavior of whale with bubble-net feeding maneuver, which has good convergence for optimization problems. Salp swarm algorithm (SSA) was developed by Mirjalili et al. [58], inspired from the behaviors of the salp chain with leader and followers, which has also attracted much attention in other fields [88]. Harris hawks optimization (HHO) was presented by Heidari et al. [63], simulating the behaviors of preying of Harris hawks with four different chasing patterns, which has good ability for solving engineering optimization problems [89]. Marine predator algorithm was presented by Faramarzi et al. [74] inspired from the behaviors of predator and prey, while the foraging mechanism relies on the velocity ratio, with Levy and Brownian movement during their habitats. Seagull optimization algorithm (SOA) [65] was proposed to solve optimization problems, which mimics the foraging behaviors of seagulls with migration phase and attacking phase. Furthermore, other concerned swarm-based metaheuristic algorithms presented so far include krill herd algorithm [45], monarch butterfly optimization [53], lion optimization algorithm [54], pity beetle algorithm [60], squirrel search algorithm [61], butterfly optimization algorithm [62], slime mould algorithm [79], golden eagle optimizer [84], red fox optimization [85], and so on.

The second category of metaheuristic algorithm is the evolutionary-based algorithm. For instance, GA was presented in 1975 by Holland [21] as one of pioneers in metaheuristics, inspired from the Darwin's theory about the natural competition, which is suitable to solve a variety of optimization problems [90, 91]. Differential evolution (DE) was developed by Storn and Price [28], usually as a popular algorithm to solve optimization problems [92]. Biogeography-based optimization [36] was derived from the migration and mutation of biological organism, while the best solution is obtained from updating the habitat suitability index by the migration and mutation. Moreover, variants of evolutionary-based metaheuristics have been developed, such as evolution strategy [93], gene expression programming [94], memetic algorithm [95].

In the third category, numerous metaheuristic algorithms have been developed based on the physics or chemistry, including simulated annealing [25], bacteria foraging optimization [31], gravitational search algorithm [38], big-bang big-crunch algorithm [33], charged system search [39], ray optimization [46], stochastic fractal search [54], equilibrium optimizer [23], sine cosine algorithm [56], water cycle algorithm [47], thermal exchange optimization [59], and so on. Simulated annealing [25] is a single-solution based heuristic algorithm inspired from the physical law about the metal's cool and anneal, and it is successful to solve complex optimization problems [96]. Gravitational search algorithm [38] was inspired from the law of gravity that the particles are attracted toward by the weight of mass, and find the best solution during optimization process. Moreover, hybrid metaheuristic algorithms inspired from physics or chemistry were also presented, including simulated-annealing PSO [97], sine–cosine salp swarm algorithm [98].

The last class of metaheuristic algorithm is based on social or human behaviors. Brain storm optimization [42] was developed by Shi which mimics the intense ideological collision from people, while each idea is a candidate solution, and the

Table 1
Metaheuristic optimization algorithms.

Author	Algorithm	Year	Category
Holland [24]	Genetic Algorithm, GA	1975	Evolutionary
Kirkpatrick et al. [25]	Simulated Annealing, SA	1983	Physics
Glover [26]	Tabu Search, TS	1986	Human
Kennedy and Eberhart [27]	Particle Swarm Optimization, PSO	1995	Swarm
Storn and Price [28]	Differential Evolution, DE	1997	Evolutionary
Dorigo et al. [29]	Ant Colony Optimization, ACO	1999	Swarm
Geem et al. [30]	Harmony Search, HS	2001	Human
Passino [31]	Bacteria Foraging Optimization, BFO	2002	Swarm
Li et al. [32]	Artificial Fish Swarm Algorithm, AFSA	2002	Swarm
Erol and Eksin [33]	Big Bang Big Crunch, BBBC	2006	Physics
Karaboga et al. [34]	Artificial Bee Colony, ABC	2007	Swarm
Atashpaz-Gargari et al. [35]	Imperialist Competitive Algorithm, ICA	2007	Human
Simon [36]	Biogeography-Based Optimization, BBO	2008	Evolutionary
Yang and Deb [37]	Cuckoo Search, CS	2009	Swarm
Rashedi et al. [38]	Gravitational Search Algorithm, GSA	2009	Physics
Kaveh et al. [39]	Charged System Search, CSS	2010	Physics
Lam and Victor [40]	Chemical Reaction Optimization, CRO	2012	Chemistry
Yang et al. [41]	Bat Algorithm, BA	2011	Swarm
Shi [42]	Brain Storm Optimization, BSO	2011	Human
Rao et al. [43]	Teaching Learning Based Optimization, TLBO	2011	Human
Yang et al. [44]	Flower Pollination Algorithm, FPA	2012	Swarm
Gandomi et al. [45]	Krill Herd, KH	2012	Swarm
Kaveh et al. [46]	Ray Optimization, RO	2012	Physics
Eskandar et al. [47]	Water Cycle Algorithm, WCA	2012	Physics
Mirjalili et al. [48]	Grey Wolf Optimizer, GWO	2014	Swarm
Cheng et al. [49]	Symbiotic Organisms Search, SOS	2014	Physics
Kashan [50]	League Championship Algorithm, LCA	2014	Human
Mirjalili [51]	Ant Lion Optimizer, ALO	2015	Swarm
Mirjalili [52]	Moth–Flame Optimization, MFO	2015	Swarm
Wang et al. [53]	Monarch Butterfly Optimization, MBO	2015	Swarm
Salimi [54]	Stochastic Fractal Search, SFS	2015	Physics
Kaveh et al. [55]	Water Evaporation Optimization, WEO	2016	Physics
Mirjalili et al. [22]	Whale Optimization Algorithm, WOA	2016	Swarm
Mirjalili [56]	Sine Cosine Algorithm, SCA	2016	Physics
Saremi et al. [57]	Grasshopper Optimization Algorithm, GOA	2017	Swarm
Mirjalili et al. [58]	Salp Swarm Algorithm, SSA	2017	Swarm
Kaveh et al. [59]	Thermal Exchange Optimization, TEO	2017	Physics
Kallioras et al. [60]	Pity Beetle Algorithm, PBA	2018	Swarm
Jain et al. [61]	Squirrel Search Algorithm, SSA	2018	Swarm
Arora and Singh [62]	Butterfly Optimization Algorithm, BOA	2019	Swarm
Heidari et al. [63]	Harris Hawks Optimization, HHO	2019	Swarm
Hashim et al. [64]	Henry Gas Solubility Optimization, HGSO	2019	Physics
Dhiman et al. [65]	Seagull Optimization Algorithm, SOA	2019	Swarm
Masadeh et al. [66]	Sea Lion Optimization Algorithm, SLOA	2019	Swarm
Sulaiman et al. [67]	Barnacles Mating Optimizer, BMO	2020	Evolutionary
Kaveh et al. [68]	Billiards-Inspired Optimization, BIO	2020	Physics
Hayyolalam et al. [69]	Black Widow Optimization Algorithm, BWOA	2020	Swarm
Khishe et al. [70]	Chimp Optimization Algorithm, COA	2020	Swarm
Faramarzi et al. [23]	Equilibrium Optimizer, EO	2020	Physics
Askari et al. [71]	Heap-Based Optimizer, HBO	2020	Human
Jahangiri et al. [72]	Interactive Autodidactic School, IAS	2020	Human
Houssein et al. [73]	Lévy Flight Distribution, LFD	2020	Human
Faramarzi et al. [74]	Marine Predator Algorithm, MPA	2020	Swarm
Bouchekara [75]	Most Valuable Player Algorithm, MVPA	2020	Human
Salih et al. [76]	Nomadic People Optimizer, NPO	2020	Human

(continued on next page)

Table 1 (*continued*).

Author	Algorithm	Year	Category
Kaveh et al. [77]	Plasma Generation Optimization, PGO	2020	Physics
Askari et al. [78]	Political Optimizer, PO	2020	Human
Li et al. [79]	Slime Mould Algorithm, SMA	2020	Swarm
Kaur et al. [80]	Tunicate Swarm Algorithm, TSA	2020	Swarm
Abualigah et al. [81]	Arithmetic Optimization Algorithm, AOA	2021	Human
Zitouni et al. [82]	Archerfish Hunting Optimizer, AHO	2021	Swarm
Meng et al. [83]	Carnivorous Plant Algorithm, CPA	2021	Swarm
Mohammadi-Balani et al. [84]	Golden Eagle Optimizer, GEO	2021	Swarm
Polap et al. [85]	Red Fox Optimization, RFO	2021	Swarm
Zitouni et al. [86]	Solar System Algorithm, SSA	2021	Physics
Emami [87]	Stock exchange trading optimization, SETO	2021	Human

solution update is conducted by clustering ideas and fusion. Teaching–learning-based optimization [43] was inspired from the behaviors of teaching and learning process in class, while students learn knowledge not only from teachers, but also from students. TLBO is proven as a high-quality algorithm in the field of metaheuristic [99]. Other well-known or recent metaheuristics about social or humans include Tabu Search [26], harmony search [30], political optimizer [78], imperialist competitive algorithm [35], league championship algorithm [50], Interactive autodidactic school [72], arithmetic optimization algorithm [81], and so on.

3. Beluga whale optimization (BWO)

This section introduces the details of the proposed BWO which is inspired from the behaviors of beluga whales, including swimming, preying and whale fall. The mathematical model of BWO is given as follows, as well as the procedure and the complexity. Finally, the comparison of BWO and other popular metaheuristic algorithms in mechanism is provided.

3.1. Inspiration

The beluga whale (Delphinapterus leucas) [100] is a member of whale living in the sea, famous for the pure white color of adults, earning the title "canary of the sea" by producing many different sounds. A beluga whale has a round and stocky body which is medium-sized with 3.5–5.5 m length, about 1500 kg weight. Belugas have sharp vision and hearing ability, and they move and hunt by sound. The main distributions of beluga whales are the Arctic and subarctic ocean in the worldwide, including Alaska, northwest Canada and off Ellesmere Island. Some belugas are housed in aquariums, and they have smiling appearance and graceful movement, as shown in Fig. 1(a).

Even though the social behaviors of beluga whales are not fully explored, beluga whales have some conspicuous social-sexual behaviors in their living behaviors documented for beluga whales in human care [101]. For example, beluga whale shows social-sexual behavior under S-posture with the body in the lateral swim, or shows agonistic behavior under vertical S-posture with head jerks. The beluga whale can swim laterally with pectoral fin raised which extends pectoral fin away from the body so that the fin is perpendicular to the body. They can dive or surface in synchronize or mirrored manner when swimming, known as milling. They can also bubble releasing from the blow hole, open mouth fully to beg for diet, and swim rapidly for or away from creatures. Besides, they can be seen to be playing, swimming, vocalizing around each other, and they have a great of curiosity towards humans.

In fact, beluga whales are highly social animals, and they can gather in groups vary with 2 to 25 members, averaging with 10 members. In Fig. 1(b), belugas are omnivorous including but not limited to shrimp, worms, codfishes, trout and salmon. When summer comes, a great many of creatures gather in some estuaries, so whales gather up and diet. Beluga whales usually bring their prey into their mouth by suction due to the teeth that are not sharp. Sometimes beluga whales with coordinating groups attack and feed on fish by steering the fish into shallow water. Besides, beluga whales are under threat from killer whales, polar bears, and human for high population density in estuaries during summer. Some whales may die and fall into the deep-sea during migration, which called "whale fall" [102], giving plenty of food for a large number of creatures without sun and oxygen, as Fig. 1(c) shown.

Inspired from the behaviors of beluga whales including swimming, preying and whale fall, we firstly develop a novel metaheuristic algorithm named beluga whale optimization (BWO). The mathematical model of BWO is established as follows.

3.2. Mathematical model of beluga whale optimization

The BWO algorithm mimics the behaviors of beluga whales such as swimming, preying and whale fall. Similar to other metaheuristics, BWO contains the exploration phase and the exploitation phase. The exploration phase guarantees the global searching ability in the design space by the random selection of beluga whales, and the exploitation phase controls the local search in the design space. To model the behaviors, the beluga whales are regarded as search agents which can move in search space by changing their position vectors. Moreover, the probability of whale fall is considered in BWO which changes the positions of beluga whales.

Due to the population-based mechanism of BWO, beluga whales are regarded as the search agents, while each beluga whale is a candidate solution, which is updated during optimization. The matrix to positions of search agents is modeled as:

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,d} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,d} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,d} \end{bmatrix} \tag{1}$$

where n is the population size of beluga whales, d represents the dimension of design variables. For all the beluga whales, the corresponding fitness values are stored as follows:

$$F_X = \begin{bmatrix} f\left(x_{1,1}, x_{1,2}, \ldots, x_{1,d}\right) \\ f\left(x_{2,1}, x_{2,2}, \ldots, x_{2,d}\right) \\ \vdots \\ f\left(x_{n,1}, x_{n,2}, \ldots, x_{n,d}\right) \end{bmatrix} \tag{2}$$

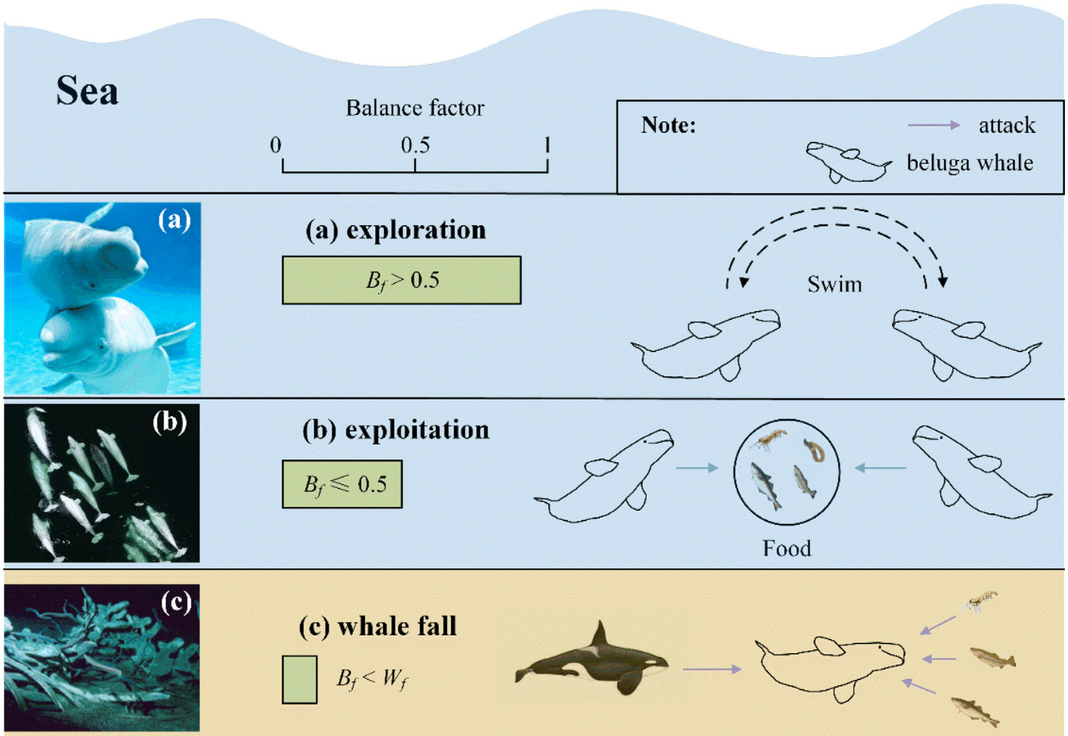

Fig. 1. Behaviors of beluga whales, (a) swim, corresponding to exploration phase; (b) foraging, corresponding to exploitation phase, (c) whale fall, for whale fall phase.
Source: https://constative.com/animals/whale-watching, https://www.sealuxe.ca/blog/narwhal, Ref. [96].

The BWO algorithm can transfer from exploration to exploitation, depending on the balance factor B_f, which is modeled as:

$$B_f = B_0(1 - T/2T_{max}) \tag{3}$$

where T is the current iteration, T_{max} is the maximum iterative number, B_0 randomly changes between (0, 1) at each iteration. The exploration phase happens when the balance factor B_f > 0.5 while the exploitation phase happens when B_f ≤0.5. With the increasing of iteration T, the fluctuation range of B_f is reduced from (0, 1) to (0, 0.5), illustrating the significant change of probabilities for exploitation and exploration phase, while the probability of exploitation phase is increased with the ever-increasing iteration T.

3.2.1. Exploration phase

The exploration phase of BWO is established by considering the swim behavior of beluga whales. According to the behaviors documented for beluga whales in human care, beluga whales can perform social-sexual behaviors under different postures [101], such as the pair swim of two beluga whales closely together with synchronized or mirrored manner, as Fig. 1 shown. Therefore, the positions of search agents are determined by the pair swim of beluga whales, and the positions for beluga whales are updated as follows:

$$\begin{cases} X_{i,j}^{T+1} = X_{i,p_j}^{T} + \left(X_{r,p_1}^{T} - X_{i,p_j}^{T}\right)(1+r_1)\sin(2\pi r_2), & j = even \\ X_{i,j}^{T+1} = X_{i,p_j}^{T} + \left(X_{r,p_1}^{T} - X_{i,p_j}^{T}\right)(1+r_1)\cos(2\pi r_2), & j = odd \end{cases} \tag{4}$$

where T is the current iteration, $X_{i,j}^{T+1}$ is the new position for the ith beluga whale on the jth dimension, p_j (j = 1, 2, …, d) is a random number selected from d-dimension, X_{i,p_j}^{T} is the position of the ith beluga whale on p_j dimension, X_{i,p_j}^{T} and X_{r,p_1}^{T} are the current positions for ith and rth beluga whale (r is a randomly selected beluga whale), r_1 and r_2 are random number between (0, 1), $\sin(2\pi r_2)$ and $\cos(2\pi r_2)$ mean fins of the mirrored beluga whales are toward the surface. According to the dimension chosen by odd and even number, the updated position reflects the synchronous or mirror behaviors of beluga whale in swimming or diving. Two random numbers r_1 and r_2 are used to enhance the random operators in the exploration phase.

3.2.2. Exploitation phase

The exploitation phase of BWO is inspired from the preying behavior of beluga whales. Beluga whales can cooperatively forage and move according to location of near beluga whales. Therefore, the beluga whales prey by sharing the information of positions for each other, considering the best candidate and others. The strategy of Levy flight [103] is introduced in the exploitative phase of BWO to enhance the convergence. We supposed that they can catch the prey with Levy flight strategy, and the mathematical model is expressed as:

$$X_i^{T+1} = r_3 X_{best}^{T} - r_4 X_i^{T} + C_1 \cdot L_F \cdot \left(X_r^{T} - X_i^{T}\right) \tag{5}$$

where T is current iteration, X_i^T and X_r^T are current position for the ith beluga whale and a random beluga whale, X_i^{T+1} is the position of new position of the ith beluga whale, X_{best}^T is the best position among beluga whales, r_3 and r_4 are random number between (0, 1), $C_1 = 2r_4(1\text{-}T/T_{max})$ is the random jump strength that measuring the intensity of Levy flight.

L_F is the Levy flight function [103], calculated as follows:

$$L_F = 0.05 \times \frac{u \times \sigma}{|v|^{1/\beta}} \tag{6}$$

$$\sigma = \left(\frac{\Gamma(1+\beta) \times \sin(\pi\beta/2)}{\Gamma((1+\beta)/2) \times \beta \times 2^{(\beta-1)/2}}\right)^{1/\beta} \tag{7}$$

where u and v are normally distributed random numbers, β is the default constant equal to 1.5.

3.2.3. Whale fall

During the migration and foraging, the beluga whales are threaten from killer whales, polar bears, and humans. Most beluga whales are smart and can escape from threats by sharing information with each other. However, a little number of beluga whales are not survived and fallen into the deep seabed. The phenomenon is called "whale fall", feeding a great number of creatures. Numerous sharks and invertebrates gather together to feed the dead body of whale, and the exposed bones and bodies of the dead whale attract abundant aggregation of hair crustaceans. Finally, the skeleton is decomposed or occupied by bacteria and corals for decades.

To model the behavior of whale fall in each iteration, we select a probability of whale fall from the individuals in the population as our subjective assumption to simulate small changes in the groups. We assume that these beluga whales either moved elsewhere or were shot down and fallen into the deep sea. In order to ensure the number of population size constant, the positions of beluga whales and step size of whale fall are using to establish the updated position. The mathematical model is expressed as:

$$X_i^{T+1} = r_5 X_i^T - r_6 X_r^T + r_7 X_{step} \tag{8}$$

where r_5, r_6, and r_7 are random numbers between (0, 1), X_{step} is the step size of whale fall established as:

$$X_{step} = (u_b - l_b) \exp\left(-C_2 T / T_{\max}\right) \tag{9}$$

where C_2 is the step factor which is related to the probability of whale fall and population size ($C_2 = 2W_f \times n$), u_b and l_b are upper and lower boundary of variables, respectively. It can be seen that the step size is affected by the boundaries of design variables, iteration and maximum iterative number.

In this model, the probability of whale fall (W_f) is calculated as a linear function:

$$W_f = 0.1 - 0.05 T / T_{\max} \tag{10}$$

The probability of whale fall is decreased from 0.1 in the initial iteration to 0.05 in the last iteration, indicates that when beluga whales are more close to food source during optimization process, the danger of beluga whale decreases.

3.3. The procedure of BWO

According to the previous theory, BWO consists of three main phases: the exploration phase simulating the swimming behavior, the exploitation phase mimicking the preying behavior, and the whale fall phase inspired from the fall of beluga whale. During the optimization process, the whale fall phase is implemented when the exploration phase and the exploitation phase are finished in each iteration. The main procedure of BWO is provided in this section. The flowchart and pseudo code of the BWO algorithm are illustrated in Fig. 2 and Algorithm 1, respectively.

Step 1: Initialization

The algorithm parameters, including population size n and maximum iterative number T_{max}, are determined. The initial positions of all beluga whales are randomly generated within the search space, and the fitness values are obtained based on the objective function.

Step 2: Update on exploration and exploitation phase

Each beluga whale is decided to enter the exploration phase or exploitation phase based on the balance factor B_f. If B_f > 0.5 for a beluga whale, the updating mechanism is enter into the exploration phase, and the position of beluga whale is updated by Eq. (4). If B_f<0.5, the updating is controlled by the exploitation phase, and the position of a beluga whale is updated using Eq. (5). Then, the fitness values of new positions are calculated and sorted to find the optimum result in the current iteration.

Step 3: Update on the whale fall phase

Some beluga whales may die and fall into the deep sea, and the probability of whale fall W_f is calculated in each iteration. Therefore, the position of a beluga whale is updated by Eq. (8).

Step 4: Terminating condition check

If the current iteration is larger than the maximum iterative number, the BWO algorithm stops. Otherwise, repeat from Step 2.

3.4. Computational complexity

The computational complexity of BWO is an important metric to judge its performance which includes three processes: initialization, fitness evaluation, and updating of the beluga whales. Note that with the beluga whales, the computational complexity of the initialization process is O(n). In exploration and exploitation phase, the computational complexity is calculated as O($n \times T_{max}$), where n is the number of beluga whales, T_{max} is the maximum iterative number. In the whale fall phase, the computational complexity is affected by the probability of whale fall W_f and balance factor B_f, which can be approximated as O($0.1 \times n \times T_{max}$). Therefore, the computational complexity of BWO is evaluated approximately as O($n \times (1+1.1 \times T_{max})$).

3.5. Conceptual difference of BWO and WOA

The BWO and whale optimization algorithm (WOA) both have the common characteristics, such as population-based algorithms, exploration phase and exploitation phase, inspired from whales. The differences between BWO and WOA are as follows. First, the preying behavior of WOA is inspired from the spiral moving of humpback, while the preying behavior of BWO is inspired from the beluga whale, by updating positions based on own position, food solution (best fitness value) and other beluga whales, without spiral moving. Second, WOA does not have the whale fall phase, while BWO has the whale fall phase to get out of local optimum. Third, WOA does not have the Levy flight mechanism, which is introduced into the BWO.

4. Experimental results and discussion

The performance of proposed BWO algorithm is tested with 30 well-known benchmark problems, and the results are compared with other 15 metaheuristic algorithms. First, the details of benchmark problems are provided, and the experimental setup and compared algorithms are presented. Then, the qualitative analysis, quantitative analysis and scalability analysis for BWO are implemented. Moreover, 4 real-world optimization problems are employed to test the performance of BWO.

4.1. Benchmark problems and experimental setup

To test the proposed BWO, 30 different benchmark problems are tested which are divided into three classes: unimodal, multimodal, and composition functions. The unimodal test functions (F1–F9) reveal the exploitation performance and the multimodal functions (F10–F24) can challenge the exploration ability. Besides, the composition functions (F25–F30) test the local optimum avoidance of algorithm. Details of benchmark functions can be found in Suganthan [104] and Liang [105]. The mathematical formulation and properties of benchmark functions are summarized in Tables 2–4, where D denotes the dimension, Range represents the bound of design variable, f_{min} is the optimal value.

15 different metaheuristic algorithms are compared with the proposed BWO algorithm, including PSO [27], DE [28], AOA [81], BBBC [33], BBO [36], CSA [37], GSA [38], GWO [48], HHO [63], MFO [52], RO [46], SSA [58], SOA [65], TLBO [43], WOA [22]. These

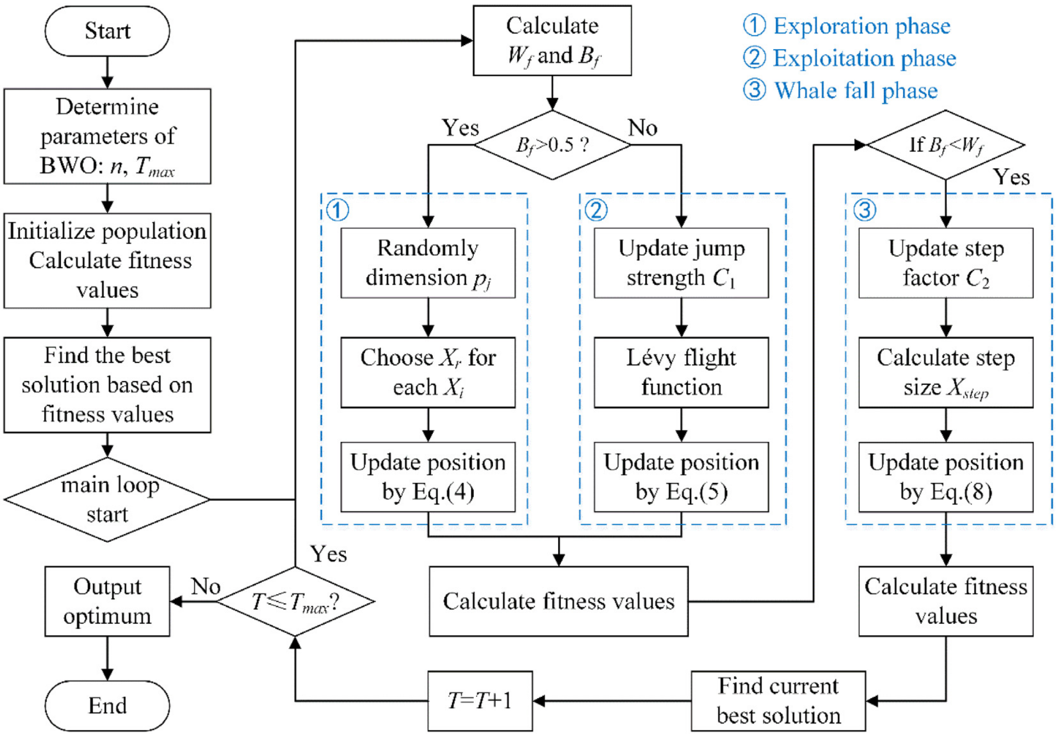

Fig. 2. Flowchart of the proposed BWO.

Table 2
Details of benchmark problems for unimodal functions.

Name	Function	D	Range	$f_{\min}$
Sphere	$F_1(x) = \sum_{i=1}^{D} x_i^2$	30	[−100, 100]	0
Schwefel's 2.22	$F_2(x) = \sum_{i=1}^{D} \lvert x_i \rvert + \prod_{i=1}^{D} \lvert x_i \rvert$	30	[−10, 10]	0
Powell Sum	$F_3(x) = \sum_{i=1}^{D} \lvert x_i \rvert^{i+1}$	30	[−1, 1]	0
Schwefel's 1.2	$F_4(x) = \sum_{i=1}^{D} \left(\sum_{j=1}^{D} x_j \right)^2$	30	[−100, 100]	0
Schwefel's 2.21	$F_5(x) = \max_i \{ \lvert x_i \rvert, 1 \le i \le D \}$	30	[−100, 100]	0
Rosenbrock	$F_6(x) = \sum_{i=1}^{D-1} \left[100 \left(x_{i+1} - x_i^2 \right)^2 + (x_i - 1)^2 \right]$	30	[−30, 30]	0
Step	$F_7(x) = \sum_{i=1}^{D} (x_i + 0.5)^2$	30	[−100, 100]	0
Quartic	$F_8(x) = \sum_{i=1}^{D} i x_i^4 + random[0, 1)$	30	[−1.28, 1.28]	0
Zakharov	$F_9(x) = \sum_{i=1}^{D} x_i^2 + \left(\sum_{i=1}^{D} 0.5 i x_i \right)^2 + \left(\sum_{i=1}^{D} 0.5 i x_i \right)^4$	30	[−5, 10]	0

compared algorithms are classical or recently metaheuristic algorithms which were widely used in solving optimization problems. The algorithmic parameters for these algorithms are tabulated in Table 5. For all algorithms, the population size (n) and maximum iterative number (T_{max}) are set as 50 and 1000 in each problem, while 30 independent runs are implemented for each algorithm to verify the robustness. All experiments are implemented using MATLAB 2018b on Windows 10, with the computer of 128 GB RAM and core i9-10900K CPU.

4.2. Qualitative analysis

The qualitative measures for the performance of BWO are discussed in this section. The test functions include 5 unimodal functions (F1–F4, F8) and 7 multimodal functions (F10–F14, F16, F19). In Fig. 3, the qualitative analysis consists of six subfigure for each benchmark function, including: (1) landscape of benchmark functions; (2) search history of search agents; (3) iterative curve of balance factor; (4) the trajectory in the first dimension; (5) the

Algorithm 1: The pseudo code of BWO algorithm

Input: Algorithmic parameters (population size, maximum iteration)
Output: The best solution

Initialize the population and evaluate fitness values, obtain the best solution (P*)
While $T \le T_{max}$ **Do**
  Obtain probability of whale fall W_f by Eq. (10) and balance factor B_f by Eq. (3)
  For each beluga whale (X_i) **Do**
    If $B_f(i) > 0.5$
      // In the exploration phase
      Generate p_j ($j = 1,2,\ldots,d$) randomly from dimension
      Choose a beluga whale X_r randomly
      Update new position of i-th beluga whale using Eq. (4)
    Else If $B_f(i) \le 0.5$
      // In the exploitation phase
      Update the random jump strength C_1 and calculate the Levy flight function
      Update new position of i-th beluga whale using Eq. (5)
    End If
    Check the boundaries of new positions and evaluate the fitness values
  End For
  For each beluga whale (X_i) **Do**
    // the whale fall phase
    If $B_f(i) \le W_f$
      Update the step factor C_2
      Calculate the step size X_{step}
      Update new position of i-th beluga whale using Eq. (8)
      Check the boundaries of new position and calculate fitness value
    End If
  End For
  Find the current best solution P^*
  $T = T+1$
End While
Output the best solution

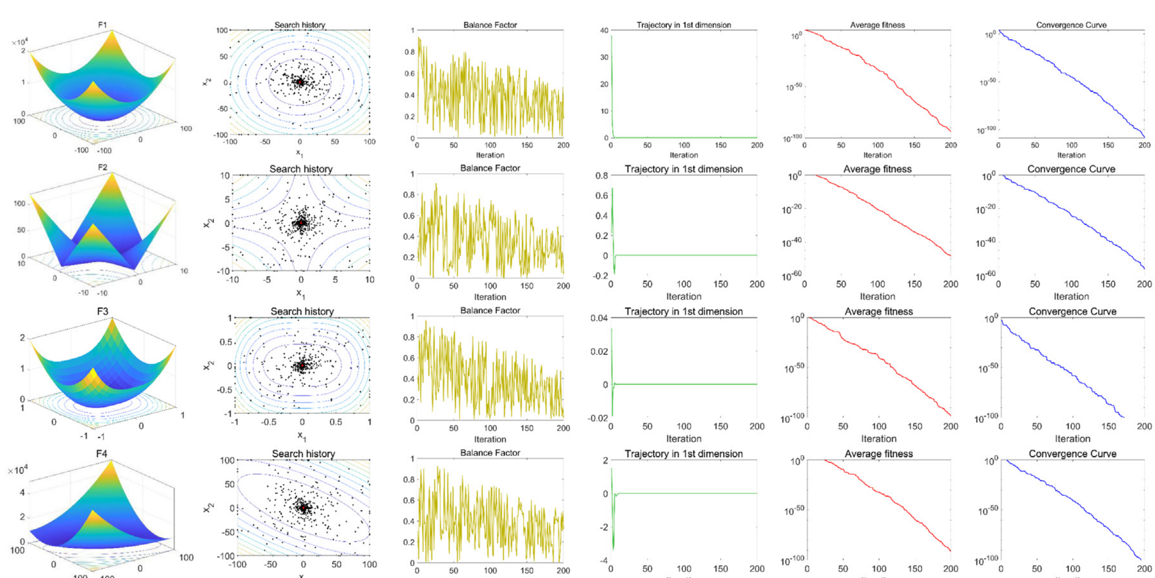

Fig. 3. Qualitative results of BWO, include: (1) function's landscape; (2) search history; (3) balance factor; (4) trajectory of 1st dimension; (5) average fitness; (6) convergence curve.

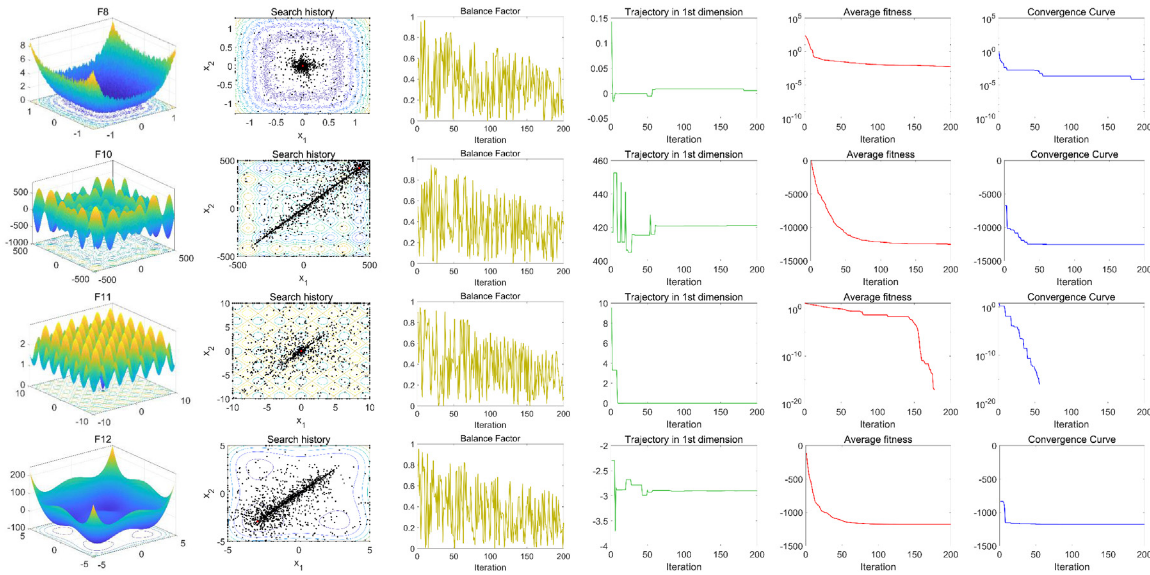

Fig. 3. (*continued*).

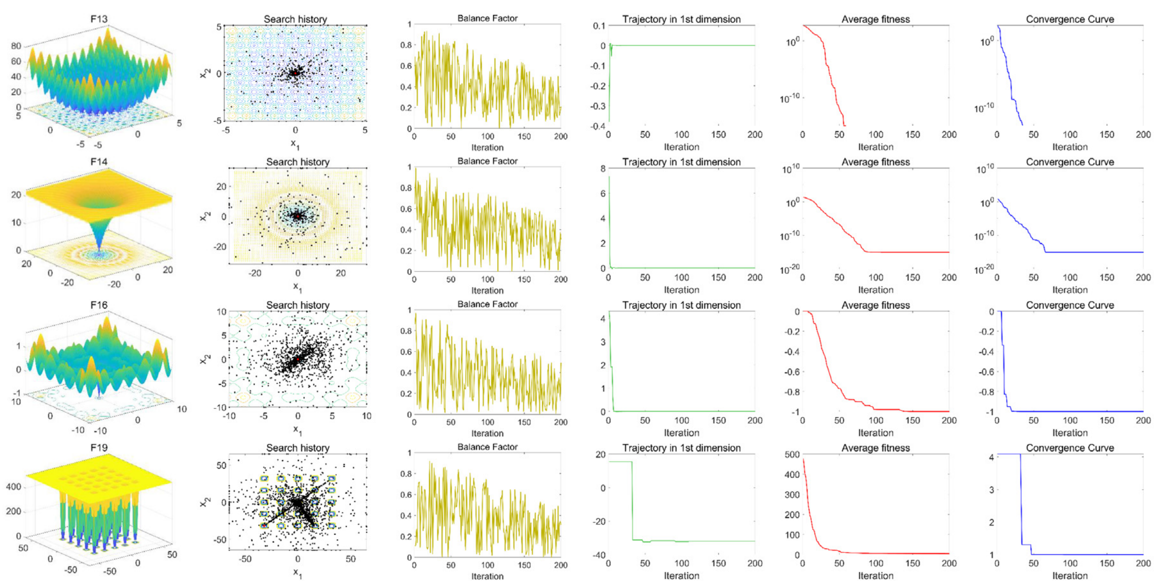

Fig. 3. (*continued*).

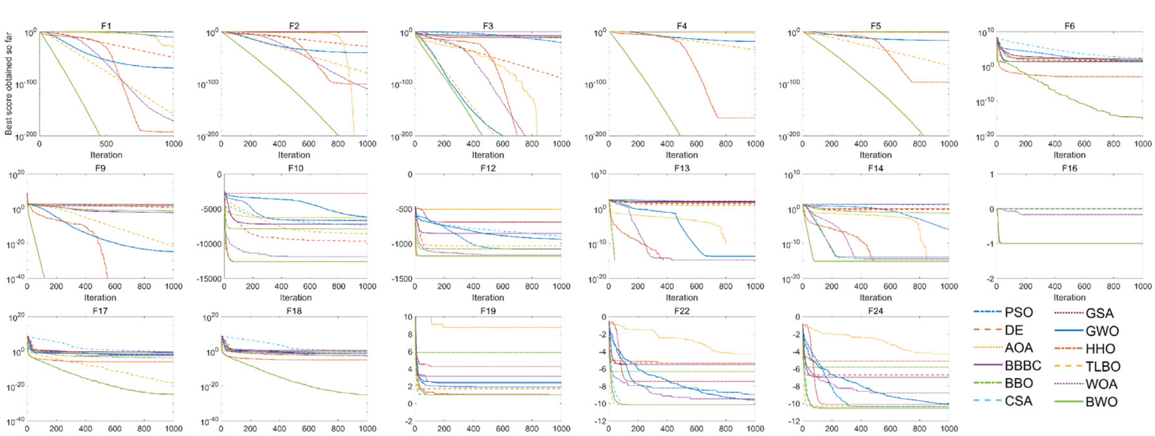

Fig. 4. Convergence curves of algorithms for different functions.

Table 3
Details of benchmark problems for multi-modal functions.

Name	Function	D	Range	$f_{\min}$
Schwefel	$F_{10}(x) = -\sum_{i=1}^{n}\left(x_i \sin\sqrt{\lvert x_i\rvert}\right)$	30	[−500, 500]	−418.98 ×D
Periodic	$F_{11}(x) = 1 + \sum_{i=1}^{D}\sin^2(x_i) - \exp\left(\sum_{i=1}^{D}x_i^2\right)$	30	[−10, 10]	0
Styblinski–Tang	$F_{12}(x) = 0.5\sum_{i=1}^{D}\left(x_i^4 - 16x_i^2 + 5x_i\right)$	30	[−5, 5]	−39.166 ×D
Rastrigin	$F_{13}(x) = \sum_{i=1}^{D}\left[x_i^2 - 10\cos(2\pi x_i) + 10\right]$	30	[−5.12, 5.12]	0
Ackley 1	$F_{14}(x) = -20\exp\left(-0.2\sqrt{\left(\sum_{i=1}^{D}x_i^2\right)/D}\right) - \exp\left(\left(\sum_{i=1}^{D}\cos(2\pi x_i)\right)/D\right) + 20 + e$	30	[−32, 32]	0
Griewank	$F_{15}(x) = \sum_{i=1}^{D}x_i^2/4000 - \prod_{i=1}^{D}\cos\left(x_i/\sqrt{i}\right) + 1$	30	[−600, 600]	0
Xin–She Yang N.4	$F_{16}(x) = \left(\sum_{i=1}^{D}\sin^2(x_i) - \exp\left(-\sum_{i=1}^{D}x_i^2\right)\right) \exp\left(-\sum_{i=1}^{D}\sin^2\sqrt{\lvert x_i\rvert}\right)$	30	[−10, 10]	−1
Penalized	$F_{17}(x) = \frac{\pi}{D}\left\{10\sin(\pi y_1) + \sum_{i=1}^{D-1}(y_i-1)^2\left[1 + 10\sin^2(\pi y_{i+1})\right] + (y_n-1)^2\right\} + \sum_{i=1}^{D}u(x_i, 10, 100, 4)$	30	[−50, 50]	0
Penalized2	$F_{18}(x) = 0.1\left\{\sin^2(3\pi x_1) + \sum_{i=1}^{D}(x_i-1)^2\left[1+\sin^2(3\pi x_i+1)\right] + (x_D-1)^2\left[1+\sin^2(2\pi x_D)\right]\right\} + \sum_{i=1}^{D}u(x_i, 5, 100, 4)$	30	[−50, 50]	0
Foxholes	$F_{19}(x) = \left[\frac{1}{500} + \sum_{j=1}^{25}\frac{1}{j + \sum_{i=1}^{2}\left(x_i - a_{ij}\right)^6}\right]^{-1}$	2	±65.536	0.998
Kowalik	$F_{20}(x) = \sum_{i=1}^{11}\left\lvert a_i - \frac{x_1\left(b_i^2 + b_i x_2\right)}{b_i^2 + b_i x_3 + x_4}\right\rvert^2$	4	[−5, 5]	0.000308
Six Hump Camel	$F_{21}(x) = 4x_1^2 - 2.1x_1^4 + x_1^6/3 + x_1x_2 - 4x_2^2 + 4x_2^4$	2	[−5, 5]	−1.0316
Shekel 5	$F_{22}(x) = -\sum_{i=1}^{5}\left\lvert(x_i - a_i)(x_i - a_i)^T + c_i\right\rvert^{-1}$	4	[0, 10]	−10.1532
Shekel 7	$F_{23}(x) = -\sum_{i=1}^{7}\left\lvert(x_i - a_i)(x_i - a_i)^T + c_i\right\rvert^{-1}$	4	[0, 10]	−10.4028
Shekel 10	$F_{24}(x) = -\sum_{i=1}^{10}\left\lvert(x_i - a_i)(x_i - a_i)^T + c_i\right\rvert^{-1}$	4	[0, 10]	−10.5364

average fitness of search agents; (6) the convergence curve of the best candidate solution.

The search history displays the location and distribution of search agents. It can be seen that the exploration ability of BWO is achieved by search agents that are expanding in the whole search space, displayed in unimodal functions (F1–F4, F8) and multi-modal functions (F11–F14, F16). Moreover, BWO achieves fast convergence due to the search trajectory clustered near the global best solution. For multi-modal functions F10 and F19 with a great many of local optimum, the search history of BWO presents a nearly linear search pattern to avoid local optimum and ensure the global solution. The results indicate that BWO has good balance between the exploration phase and exploitation phase in solving optimization problems.

The balance factor of BWO shows the transformation for the exploration phase and exploitation phase. Among results, the balance factor is changed with the increasing iteration T. In the initial iterative step, the probability of entering exploration phase and exploitation phase is equal. With increasing of iteration, the probability of exploitation is increased and equals to one in the final iteration. The trajectory of the first search agent in the first dimension can represent the primary exploratory behavior of BWO. Results show that the fast oscillation is happened in the primary phase, while the slight oscillation is happened in the anaphase. This behavior can ensure the global convergence of BWO.

The average fitness curves and convergence curves for different functions are also provided in Fig. 3. Among the results, the rapid changes happen in the initial iterations and diminishing changes happen in the follower iteration. This phenomenon also reveals the transition from the exploration phase to exploitative phase, which corresponding to the behavior of the balance factor. The convergence curves show that how the optimal fitness values improve during the optimization process.

4.3. Quantitative analysis

Although the exploration and exploitation ability of BWO is proved by qualitative analysis, the performance of BWO is not fully investigated. In this subsection, the statistical measures to quantify the performance of BWO are presented. For each algorithm, 30 independent runs are implemented in each problem,

Table 4
Details of benchmark problems for composition functions.

Function	D	Range	f_{min}
F_{25}(CF1): $f_1, f_2, f_3, \ldots, f_{10}$ = Sphere Function $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [1, 1, 1, \ldots, 1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}] = [5/100, 5/100, 5/100, \ldots, 5/100]$	10	[−5, 5]	0
F_{26}(CF2): $f_1, f_2, f_3, \ldots, f_{10}$ = Griewank's Function $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [1, 1, 1, \ldots, 1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}] = [5/100, 5/100, 5/100, \ldots, 5/100]$	10	[−5, 5]	0
F_{27}(CF3): $f_1, f_2, f_3, \ldots, f_{10}$ = Griewank's Function $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [1, 1, 1, \ldots, 1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}] = [1, 1, 1, \ldots, 1]$	10	[−5, 5]	0
F_{28}(CF4): f_1, f_2 = Ackley's Function, f_3, f_4 = Rastrigin's Function, f_5, f_6 = Weierstrass Function, f_7, f_8 = Griewank's Function, f_9, f_{10} = Sphere's Function $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [1, 1, 1, \ldots, 1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}]$ = [5/32, 5/32, 1, 1, 5/0.5, 5/0.5, 5/100, 5/100, 5/100, 5/100]	10	[−5, 5]	0
F_{29}(CF5): f_1, f_2 = Rastrigin's Function, f_3, f_4 = Weierstrass Function, f_5, f_6 = Griewank's Function, f_7, f_8 = Ackley's Function, f_9, f_{10} = Sphere's Function $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [1, 1, 1, \ldots, 1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}]$ = [1/5, 1/5, 5/0.5, 5/0.5, 5/100, 5/100, 5/32, 5/32, 5/100, 5/100]	10	[−5, 5]	0
F_{30}(CF6): f_1, f_2 = Rastrigin's Function, f_3, f_4 = Weierstrass Function, f_5, f_6 = Griewank's Function, f_7, f_8 = Ackley's Function, f_9, f_{10} = Sphere's Function $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}]$ = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1] $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}]$ = [0.1*1/5, 0.2*1/5, 0.3*5/0.5, 0.4*5/0.5, 0.5*5/100, 0.6*5/100, 0.7*5/32, 0.8*5/32, 0.9*5/100, 1*5/100]	10	[−5, 5]	0

and the statistical results (average and standard deviation) are summarized in Table 6. It is worth mentioning that the best average and standard deviation values for benchmark functions are bold in Table 6.

Based on the results from 9 unimodal functions (F1 to F9) in Table 6, it is obvious that BWO outperformed compared algorithms on 7 unimodal functions (F1–F6, F9), and it achieves first rank in unimodal test functions in total. BWO obtains the first rank on both the average and the standard deviation in functions from F1 to F6, and F9, while BWO achieves the second rank (2/16) after DE in function F7 and the second rank (2/16) after AOA in function F8. For other compared algorithms, AOA achieves global optimum results on functions F2 and F3, GWO, HHO, TLBO, WOA achieves first rank on function F3. It can be stated that high exploitation ability of BWO is verified in unimodal functions.

Due to the great number of local optima in multimodal functions, the exploration ability of the proposed BWO can be evaluated. From the results on 15 multimodal functions (F10 to F24) in Table 6, BWO provides first rank both on average and standard deviation values in totally 10 multimodal functions (F10–F19), achieves first rank on average value on 4 multimodal functions (F21–F24), and achieves second rank on average value on function F20. The average values of each multimodal functions obtained by BWO are in the top two rank. BWO can provide competitive results in the composition functions (F25–F30), and rank first of standard deviations in composition functions (F27–F28). For other top ranking compared metaheuristic algorithms, HHO achieves best results in totally 6 multimodal functions (F11, F13–F16, F19), CSA obtains best results in 5 multimodal functions (F19, F21–F24), TLBO and AOA achieves best results in 3 functions. In general, the statistical results of 30 benchmark functions demonstrate that BWO has good accuracy and robustness in unimodal and multimodal functions.

Fig. 4 provides the convergence curves of PSO, AOA, GWO, HHO, TLBO, WOA and the proposed BWO for some benchmark functions. From the convergence curves of functions F1–F6, F9–F10, F12–F14, F16–F18, BWO achieves the highest accuracy of fitness values and fastest convergence in these unimodal and multimodal functions, while some algorithms cannot obtain the global solution due to the local optima stagnation. For functions F19, F22 and F24, BWO provides competitive ability to convergence. It is worth to mention that BWO achieves the global optimum in functions F1, F2, F3, F4, F5, F9, and F13. It can be seen that the exploitative ability of BWO is fully performed for dealing with unimodal functions, while the explorative ability of BWO is performed in multimodal functions. The operators in exploration and exploitation phase can ensure the global convergence of BWO.

4.4. Scalability analysis

To test the scalability of BWO, 18 test functions (F1–F18) with four different dimensions (30, 50, 100, 500) are tested and compared with the 11 different metaheuristics including PSO, DE, AOA, BBBC, BBO, CSAA, GSA, GWO, HHO, MFO, WOA. Population size and maximum iterative number for each algorithm are fixed at 50 and 1000 for compared algorithms in different cases, with 30 independent runs. In this subsection, 72 cases for the whole scalability analysis are considered. The results are tabulated in Table 7 for unimodal and multimodal functions respectively. From these results, it is observed that the performance of BWO is superior in most cases than compared algorithms because BWO achieves the best average and standard deviation values in 67 cases among 72 cases (93%), which is higher than AOA (15.3%), HHO (5.6%), WOA (5.6%), GWO (4.2%) and other algorithms (0%).

More than the given evaluation metrics (average and standard deviation), Friedman ranking test [106] has been applied in the scalability analysis for the above-mentioned algorithms in a statistical way. The ranks of comparative algorithms using 18 test functions (F1–F18) with different dimensions (30, 50, 100, and 500) are investigated by the Friedman ranking test, and the results of compared algorithms are tabulated in Table 8. The results show that the proposed BWO is ranked first compared to other comparative algorithms, while HHO, WOA and GWO are ranking second, third and fourth, respectively. In conclusion, BWO outperforms than other compared algorithms in scalability of different dimensions through unimodal and multimodal benchmark functions.

Table 5
Algorithmic parameters for metaheuristics.

Algorithm	Parameters	Values
# all algorithms	Population size, maximum iterative number, replication times	50, 1000, 30
PSO	Cognitive and social constant Inertia weight linearly decreased at interval	$c_1 = 2$, $c_2 = 2$ [0.9 0.2]
DE	Scaling factor, crossover probability	0.5, 0.5
AOA	Sensitive parameter, Control parameter	$\alpha = 5$, $\mu = 0.5$
BBBC	controlling parameter of weight average of particles Parameter of limiting size of initial search space	0.2 1
BBO	Probability of modifying a habitat Probability limits of immigrations Size of each step I and E Probability of mutation	1 [0, 1] 1 1 0.1
CSA	Step size parameter, discovery rate of alien eggs	1, 0.2
GSA	G0, α, Rnorm, Rpower	100, 40, 2, 1
GWO	Convergence parameter a decreased at interval	[2 0]
HHO	Probability thresholds of escaping, escaping energy	0.5, 0.5
MFO	Convergence constant spiral factor	a = [-2 -1] $b = 1$
RO	Stoch, d	0.35, 7.5
SSA	Leader position update probability	0.5
SOA	Control parameter f_c	[2, 0] 2
TLBO	Teacher factor	[1, 2]
WOA	Probability of encircling mechanism, spiral factor	0.5, 1
BWO	Probability of whale fall decreased at interval W_f	[0.1 0.05]

4.5. Real-world application

In this section, four constrained optimization problems from real-world engineering are chosen to test the performance of proposed BWO, including cantilever beam design problem, welded beam design problem, tension/compression spring design problem, and pressure vessel design problem. These real-world optimization problems are usually constrained optimization problems, while they need to be solved by metaheuristics equipped with a constraint handling technique. Common constraint handling techniques consist of penalty functions, decoders, special operators, separation of objective function, feasibility rules, stochastic ranking, ε-constrained method, and so on. Among these constraint handling techniques, the penalty function method is very popular due to its simplicity and efficiency. Therefore, the static penalty function method is utilized to penalize the objective function, with a positive constant that has a large value. It should be noted that population size and maximum iterative number are 50 and 1000 respectively, and 30 independent runs are implemented. The comparing metaheuristic algorithms include PSO, AOA, BBBC, GWO, GSA, WOA, HHO, and more algorithms reported from literature such as GEO, GA, RO, BOA. The best optimal cost for each metaheuristic algorithm is chosen to compare in four optimization problems. Details of engineering benchmark problems are described as following.

4.5.1. Cantilever beam design problem

A cantilever beam with five attached hollow blocks is considered, as displayed in Fig. 5, while the total weight of the structure is minimized. The mathematical model of the cantilever beam is established which can be shown in Appendix. Table 9 shows the best results obtained from BWO and other compared algorithms, while the optimal design variables and optimum cost of weight are provided. According to the results, PSO, GWO, GSA achieves better performance on the optimal cost, while the result of BWO is very close to the global best solution.

4.5.2. Welded beam design problem

This is a mechanical engineering problem in the field of structural optimization, in minimizing the weight of a welded beam, including four design variables: thickness of weld (h), length (L), height (t), and thickness of the bar (b). The structure of welded beam is shown in Fig. 6, and the mathematical formulation is formulated as shown in Appendix. The results of welded beam design problem solved by BWO and compared algorithms are summarized in Table 10, including four design variables and optimum value of weight. The results show that BWO outperforms AOA, GSA, WOA, HHO, RO, BOA, and provides competitive results compared to PSO, BBBC, GWO. The convergence of BWO in solving multiple nonlinear constraints efficiently is proved.

4.5.3. Tension/compression spring design problem

In this optimization, we consider a tension/compression spring design problem with the objective of minimizing weight subject to constraints on minimum deflection, shear stress, surge frequency and some box constraints. Four design variables are wire diameter (d), mean coil diameter (D) and number of active coils (N). The structure of tension/compression spring is shown in Fig. 7. The optimization formulation of spring design is provided in Appendix. The results of tension/compression spring design problem are tabulated in Table 11, which is solved by BWO and other metaheuristic algorithms. The results show that BWO can provide competitive results compared with metaheuristics.

4.5.4. Pressure vessel design problem

A well-known real-world optimization problem is pressure vessel design problem considering mixed type of variables (continuous/discrete). The objective function is to minimize the construction weight under the maximum pressure and minimum volume. The design variables consider coating thickness of cylinder (T_s), coating thickness of hemispherical cover (T_h), radius of the cylinder without the shell (R), length of the cylinder (L). The structure of pressure vessel design problem is shown in Fig. 8, and the mathematical formulation is shown in Appendix. Table 12 shows the statistical results of pressure vessel design problem, which is solved by BWO and compared algorithms. The results show that BWO can provide a better solution than AOA, BBBC, GWO, GSA, WOA, HHO, and it has a competitive result with PSO. In all, BWO can provide a competitive solution in the pressure vessel design problem.

5. Conclusions

In this paper, a novel metaheuristic algorithm named beluga whale optimization (BWO) was presented. BWO was inspired from the behaviors of beluga whales such as swim, prey and whale fall, while consists of three phases: exploration phase, exploitation phase and whale fall phase. In BWO, the pair swims of beluga whales are used to construct the updating mechanism for beluga whales in the exploration phase, and the Levy flight is introduced to enhance the global convergence in the exploitation phase. Moreover, the whale fall phase is considered in BWO

Table 6
Results of 30 benchmark functions for compared algorithms.

Fun	Method	PSO	DE	AOA	BBBC	BBO	CSA	GSA	GWO
F1	Aver	5.41E −11	4.85E −50	1.47E −28	5.84E −01	5.68E −02	3.68E −01	2.91E+00	2.53E −70
	STD	2.72E −10	1.54E −49	8.03E −28	9.35E −02	2.05E −02	1.20E −01	8.42E+00	3.95E −70
F2	Aver	4.33E+00	1.35E −28	**0**	4.00E+00	7.25E −02	3.91E −01	2.76E −01	4.36E −41
	STD	6.79E+00	6.65E −28	**0**	1.96E+01	1.31E −02	9.54E −02	3.52E −01	3.94E −41
F3	Aver	2.97E −22	4.85E −90	**0**	2.94E −08	1.13E −10	5.52E −13	4.56E −12	**0**
	STD	1.36E −21	1.77E −89	**0**	9.93E −09	1.93E −10	8.18E −13	1.14E −11	**0**
F4	Aver	6.58E+00	1.78E+01	2.94E −03	2.03E+00	2.65E+01	2.56E+04	2.95E+02	1.61E −19
	STD	3.58E+00	1.74E+01	1.07E −02	5.83E −01	8.22E+00	4.52E+03	1.14E+02	5.60E −19
F5	Aver	4.46E −01	1.90E+00	1.99E −02	3.09E −01	3.80E −01	6.02E+00	4.46E+00	1.51E −17
	STD	1.36E −01	1.37E+00	2.07E −02	2.38E −02	6.99E −02	8.12E −01	1.28E+00	1.81E −17
F6	Aver	5.18E+01	3.72E+01	2.79E+01	1.04E+02	1.09E+02	2.18E+02	1.49E+02	2.64E+01
	STD	4.52E+01	3.73E+01	4.75E −01	1.44E+02	1.18E+02	8.03E+01	1.05E+02	7.05E −01
F7	Aver	1.93E −11	**5.65E −33**	2.42E+00	5.83E −01	5.73E −02	3.72E −01	1.33E+00	2.92E −01
	STD	6.63E −11	**7.97E −33**	2.14E −01	7.00E −02	1.93E −02	1.40E −01	3.34E+00	2.47E −01
F8	Aver	2.54E+00	1.20E −02	**2.32E −05**	1.25E −02	5.80E −03	3.92E −02	5.25E −02	4.74E −04
	STD	3.22E+00	4.65E −03	**1.71E −05**	4.18E −03	1.71E −03	1.29E −02	2.81E −02	3.22E −04
F9	Aver	1.67E+02	4.94E+00	1.70E+02	6.20E −03	7.40E −02	1.82E+02	4.49E+01	2.61E −25
	STD	9.19E+01	7.44E+00	4.34E+01	7.32E −04	2.15E −02	2.65E+01	1.31E+01	1.39E −24
F10	Aver	−6674.506	−9636.729	−6258.369	−7208.450	−7880.209	−7410.897	−2785.141	−6141.680
	STD	778.501	990.341	489.998	1151.516	693.209	766.157	304.292	744.421
F11	Aver	2.80E+00	1.00E+00	**0**	1.22E+00	1.00E+00	5.59E+00	8.33E+00	1.45E+00
	STD	8.26E −01	2.86E −16	**0**	1.20E+00	1.69E −04	2.42E −01	1.84E+00	9.57E −01
F12	Aver	−1077.442	−1074.614	−508.232	−851.418	−1076.03	−894.723	−688.460	−936.995
	STD	3.07E+01	3.86E+01	5.32E+01	2.44E+02	3.32E+01	5.02E+01	2.38E+02	5.96E+01
F13	Aver	7.71E+01	3.33E+01	**0**	1.22E+02	7.27E+01	1.98E+02	6.91E+01	2.27E −14
	STD	2.61E+01	7.95E+00	**0**	3.16E+01	2.25E+01	1.56E+01	7.51E+01	7.40E −14
F14	Aver	1.21E −06	7.08E −01	**8.88E −16**	2.03E+01	5.86E −02	1.12E+01	1.10E+00	1.36E −14
	STD	1.40E −06	7.18E −01	**0**	9.69E −02	1.29E −02	5.83E+00	7.43E −01	2.41E −15
F15	Aver	7.15E −03	5.17E −03	7.38E −02	6.96E −01	1.71E −01	6.63E −01	1.62E+01	1.67E −03
	STD	6.41E −03	7.77E −03	4.23E −02	5.03E −02	6.16E −02	9.33E −02	4.72E+00	4.46E −03

Fun	Method	HHO	MFO	RO	SSA	SOA	TLBO	WOA	BWO
F1	Aver	2.6E −193	6.67E+02	4.07E −01	2.31E −09	4.22E −19	4.9E −159	2.0E −171	**0**
	STD	3.8E −192	2.54E+03	4.93E −10	9.90E −02	6.56E −19	1.2E −158	3.9E −169	**0**
F2	Aver	1.9E −101	3.47E+01	1.99E+00	7.99E+00	1.10E −12	8.55E −80	5.6E −110	**0**
	STD	7.6E −101	2.47E+01	2.97E+01	3.23E −01	1.47E −12	5.96E −80	3.0E −109	**0**
F3	Aver	**0**	3.19E −21	1.64E −09	1.40E −08	2.13E −69	**0**	**0**	**0**
	STD	**0**	9.79E −21	7.82E −09	2.90E −09	1.17E −68	**0**	**0**	**0**
F4	Aver	2.3E −166	1.84E+04	1.66E+01	1.31E+00	1.79E −09	1.04E −35	1.34E+04	**0**
	STD	1.2E −164	1.20E+04	2.85E+00	1.83E+01	2.12E −09	2.78E −35	7.95E+03	**0**
F5	Aver	8.46E −98	5.89E+01	2.14E+00	3.25E −03	4.08E −06	3.88E −65	3.25E+01	**0**
	STD	4.43E −97	1.09E+01	8.72E −03	1.05E+00	4.30E −06	2.43E −65	3.20E+01	**0**
F6	Aver	1.11E −03	1.26E+04	8.75E+01	1.01E+02	2.77E+01	2.11E+01	2.66E+01	**2.20E −15**
	STD	1.25E −03	3.09E+04	1.20E+02	4.57E+01	6.05E −01	1.01E+00	3.14E −01	**7.47E −15**
F7	Aver	1.05E −05	1.99E+03	4.82E −01	2.35E −09	2.00E+00	4.26E −18	4.09E −03	3.35E −28
	STD	2.17E −05	4.04E+03	4.12E −10	8.58E −02	4.22E −01	1.09E −17	1.58E −03	7.72E −28
F8	Aver	3.37E −05	2.85E+00	1.77E −02	2.83E −02	1.23E −03	4.50E −04	9.28E −04	2.63E −05
	STD	3.48E −05	7.40E+00	1.14E −02	1.03E −02	9.40E −04	1.46E −04	9.32E −04	1.89E −05
F9	Aver	1.49E −114	2.73E+02	7.35E −01	2.33E+02	1.12E −11	2.55E −22	4.63E+02	**0**
	STD	8.13E −114	8.99E+01	4.01E+01	1.48E −01	2.77E −11	5.72E −22	7.99E+01	**0**
F10	Aver	−12569.403	−8866.212	−6861.988	−7855.684	−5535.212	−8551.773	−11871.72	**-12569.48**
	STD	0.168	1004.906	708.607	286.249	605.848	657.194	1290.836	**1.85E −12**
F11	Aver	**0**	4.14E+00	2.84E+00	1.00E+00	2.22E+00	3.27E+00	6.06E −01	**0**
	STD	**0**	8.14E −01	5.09E −12	3.51E −01	5.61E −01	1.81E+00	5.92E −01	**0**
F12	Aver	−1174.984	−1033.147	−984.319	−1025.136	−756.736	−1031.262	−1157.982	**-1174.985**
	STD	0.001	38.489	40.259	35.727	59.239	35.625	39.667	**2.31E −13**
F13	Aver	**0**	1.41E+02	6.92E+01	1.28E+02	2.27E+00	9.64E+00	1.89E −15	**0**
	STD	**0**	3.81E+01	3.45E+01	2.09E+01	4.56E+00	5.33E+00	1.04E −14	**0**
F14	Aver	**8.88E −16**	9.77E+00	1.64E+00	9.56E+00	2.00E+01	4.44E −15	4.32E −15	**8.88E −16**
	STD	**0**	9.57E+00	9.80E+00	6.95E −01	1.32E −03	0	2.18E −15	**0**
F15	Aver	**0**	9.04E+00	1.15E+00	8.94E −03	9.23E −03	**0**	1.24E −03	**0**
	STD	**0**	2.75E+01	1.11E −02	4.70E −02	1.67E −02	**0**	6.77E −03	**0**

(*continued on next page*)

Table 6 (*continued*).

Fun	Method	PSO	DE	AOA	BBBC	BBO	CSA	GSA	GWO
F16	Aver	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	STD	4.28E −14	1.44E −43	4.12E −08	1.18E −15	7.24E −17	2.61E −12	2.23E −10	4.65E −17
F17	Aver	1.04E −02	4.10E −01	3.11E −01	2.27E −01	1.67E −04	4.01E −01	4.74E −01	2.58E −02
	STD	0.0316	0.9322	0.0458	7.13E −01	1.15E −04	0.2009	0.3984	1.25E −02
F18	Aver	3.30E −03	3.25E −01	2.77E+00	2.65E −02	1.81E −03	4.88E −01	3.43E+00	3.40E −01
	STD	5.12E −03	8.48E −01	0.0980	3.52E −03	0.0011	1.70E −01	4.82E+00	1.47E −01
F19	Aver	1.8887	1.6900	8.6419	3.1321	5.8620	**0.9980**	4.2439	2.4375
	STD	1.4503	1.4704	4.4084	2.7264	4.1764	**0.0000**	2.0800	2.9447
F20	Aver	4.29E −03	2.96E −03	1.86E −02	4.11E −03	7.26E −03	4.53E −04	7.08E −03	4.38E −03
	STD	7.45E −03	4.63E −03	3.13E −02	7.40E −03	9.29E −03	9.89E −05	5.94E −03	8.13E −03
F21	Aver	**−1.0316**	**−1.0316**	**−1.0316**	−0.9658	−1.0044	**−1.0316**	**−1.0316**	**−1.0316**
	STD	6.71E −16	6.65E −16	5.27E −08	1.89E −01	1.49E −01	**6.78E −16**	7.85E −05	2.31E −09
F22	Aver	−8.9833	−5.3877	−4.3412	−5.5401	−6.3197	**−10.1532**	−7.4610	−9.6179
	STD	2.1578	3.3050	1.1936	3.0202	3.6962	**6.79E −15**	3.0189	1.6387
F23	Aver	−7.6985	−6.4625	−4.4008	−6.0997	−4.5616	**−10.4029**	−9.7975	−10.2256
	STD	3.0156	3.3876	1.0401	3.6710	3.0381	**8.73E −16**	1.8563	0.9704
F24	Aver	−10.3577	−6.6911	−4.3193	−6.9915	−5.8264	**−10.5364**	−10.1098	−10.0856
	STD	0.9787	3.7695	1.4549	3.6927	3.6918	**0.0000**	1.6847	1.7518
F25	Aver	149.3204	140.4844	312.3490	156.2867	184.4267	135.1694	172.7969	173.7346
	STD	**26.9050**	32.7585	86.2482	45.7597	65.4265	31.3281	41.8315	43.6018
F26	Aver	206.6816	222.2623	369.6884	218.0039	287.7389	**201.8746**	244.5104	241.6097
	STD	33.0759	30.5024	76.5944	42.7071	90.1737	29.9475	48.8622	45.4280
F27	Aver	674.6445	686.4329	882.7467	681.7742	802.3838	**650.5829**	730.5822	737.7796
	STD	71.5399	68.1736	48.0055	89.0926	97.9160	73.1954	97.0344	60.4306
F28	Aver	783.6841	801.4167	886.4679	800.5310	897.2850	**766.3891**	878.0059	866.7182
	STD	42.6887	49.1554	31.4703	46.1891	33.8325	47.6313	41.4531	56.0731
F29	Aver	169.5487	180.6458	357.8420	201.1217	258.6728	**157.8441**	222.1998	206.3271
	STD	35.0465	32.5792	117.9601	55.1358	88.6471	32.3033	42.4960	46.1577
F30	Aver	677.9054	845.6565	894.1299	794.3321	894.4781	690.4053	825.4644	895.2381
	STD	89.7194	102.3429	29.6626	118.0199	**6.0840**	84.1000	105.6190	13.3725

Fun	Method	HHO	MFO	RO	SSA	SOA	TLBO	WOA	BWO
F16	Aver	**-1.0000**	0.0000	0.0000	0.0000	0.0000	0.0000	−0.1667	**-1.0000**
	STD	**0**	3.99E −12	3.68E −24	3.71E −13	5.62E −13	6.98E −16	3.79E −01	**0**
F17	Aver	9.26E −07	3.21E −01	1.51E+00	9.75E −01	1.66E −01	1.04E −18	2.99E −03	**1.86E −25**
	STD	1.53E −06	0.4531	1.1580	1.42E+00	1.10E −01	5.10E −18	7.51E −03	**8.28E −25**
F18	Aver	1.15E −05	6.36E −02	1.14E −01	2.56E −03	1.63E+00	2.02E −02	3.76E −02	**9.74E −26**
	STD	1.30E −05	2.92E −01	4.73E −03	0.0381	2.06E −01	3.84E −02	5.43E −02	**0.0000**
F19	Aver	**0.9980**	1.4272	**0.9980**	**0.9980**	1.6594	**0.9980**	2.3052	**0.9980**
	STD	**0.0000**	1.0274	**0.0000**	**0.0000**	0.9513	**0.0000**	2.9624	**0.0000**
F20	Aver	3.51E −04	8.92E −04	3.42E −04	1.39E −03	1.17E −03	**3.13E −04**	5.42E −04	3.25E −04
	STD	1.68E −04	3.63E −04	3.59E −03	2.96E −05	2.33E −04	2.66E −05	3.25E −04	**1.11E −05**
F21	Aver	**−1.0316**	**−1.0316**	**−1.0316**	**−1.0316**	**−1.0316**	**−1.0316**	**−1.0316**	**−1.0316**
	STD	2.48E −13	**6.78E −16**	7.00E −16	5.74E −06	3.75E −07	**6.78E −16**	4.20E −11	4.71E −05
F22	Aver	−5.3893	−7.3193	−9.9447	−1.7228	−4.9578	**−10.1532**	−9.4689	**−10.1532**
	STD	1.2722	3.5783	1.1155	0.3874	4.3068	0.0000	1.7610	1.17E −08
F23	Aver	−5.4413	−9.0821	−10.1049	−1.4201	−8.5734	−10.2088	−8.8225	**−10.4029**
	STD	1.3473	2.7052	0.7753	0.4272	3.4855	1.06E+00	2.7171	4.61E −06
F24	Aver	−5.1282	−9.3082	−10.4382	−2.0853	−8.3477	**−10.3555**	−8.7842	**−10.5364**
	STD	0.0003	2.8034	1.0856	0.0852	3.4972	0.9906	2.7423	4.30E −06
F25	Aver	155.2586	147.8998	240.4322	145.2500	196.6751	**133.7601**	155.7147	174.6018
	STD	48.9607	37.4161	39.6909	45.9586	44.4405	27.2017	39.6519	35.2997
F26	Aver	229.6278	229.5731	296.6702	209.0209	243.2509	211.3231	216.4554	236.8907
	STD	48.0805	**29.6414**	30.8210	52.6802	43.9610	29.8545	41.0352	39.6911
F27	Aver	772.6255	689.9632	843.6061	694.8218	740.6820	667.1531	715.3396	882.0657
	STD	134.2617	91.9611	68.4757	61.7283	75.1389	62.0381	73.0237	**47.2217**
F28	Aver	852.4864	869.8141	862.0093	789.3298	848.3185	886.7015	826.8512	893.6029
	STD	68.8470	38.2799	43.0620	32.9723	47.3674	23.6080	52.4768	**21.9930**
F29	Aver	175.1339	190.0917	236.9524	183.1179	221.8144	173.4133	190.5187	200.8624
	STD	45.8903	30.2297	36.3149	46.9697	55.4124	**23.5941**	40.6247	39.1863
F30	Aver	727.3212	849.4370	838.1801	**568.4113**	629.0601	890.1025	772.2097	850.2291
	STD	161.7314	79.5565	47.5887	63.7352	67.2416	28.6180	136.1041	98.3454

Table 7
Results of test functions (F1–F18) with 30, 50, 100 and 500 dimensions.

Fun	Dim	Method	PSO	DE	AOA	BBBC	BBO	CSAA	GSA	GWO	HHO	MFO	WOA	BWO
F1	30	Aver	3.98E −11	5.63E −50	1.38E −39	6.07E −01	5.18E −02	3.59E −01	7.16E −01	2.58E −70	8.1E −189	2.33E+03	1.1E −170	**0**
		STD	1.86E −10	2.68E −49	7.54E −39	6.56E −02	1.56E −02	1.52E −01	1.50E+00	5.94E −70	0	5.04E+03	0	**0**
	50	Aver	5.89E −05	3.59E −27	9.26E −19	1.65E+00	5.44E+00	8.07E+01	2.84E+02	1.47E −51	6.5E −197	7.67E+03	1.2E −171	**0**
		STD	8.89E −05	6.04E −27	5.07E −18	1.53E −01	9.59E −01	1.97E+01	1.10E+02	3.00E −51	0	8.98E+03	0	**0**
	100	Aver	8.21E −01	2.68E −08	1.69E −02	6.78E+00	1.31E+02	2.50E+03	3.34E+03	1.67E −34	1.4E −191	3.44E+04	2.6E −169	**0**
		STD	4.68E −01	4.23E −08	8.71E −03	6.50E −01	1.27E+01	3.56E+02	7.62E+02	1.21E −34	0	1.75E+04	0	**0**
	500	Aver	1.72E+03	5.30E+04	5.61E −01	5.35E+02	7.47E+03	7.07E+04	3.36E+04	2.79E −14	2.3E −191	9.16E+05	1.3E −166	**0**
		STD	1.46E+02	1.32E+04	3.34E −02	1.84E+02	2.84E+02	3.87E+03	1.42E+03	1.14E −14	0	3.38E+04	0	**0**
F2	30	Aver	2.00E+00	2.32E −29	**0**	7.06E+00	6.78E −02	3.78E −01	1.34E −01	5.67E −41	3.7E −101	3.17E+01	1.3E −110	**0**
		STD	4.07E+00	4.45E −29	**0**	2.53E+01	1.23E −02	9.63E −02	1.14E −01	4.18E −41	1.5E −100	2.29E+01	4.0E −110	**0**
	50	Aver	2.60E+01	3.33E −01	**0**	1.22E+07	9.93E −01	1.27E+01	4.79E+00	1.40E −30	2.5E −101	6.91E+01	1.0E −109	**0**
		STD	1.81E+01	1.83E+00	**0**	6.67E+07	9.27E −02	2.00E+00	1.21E+00	1.18E −30	9.5E −101	3.30E+01	3.4E −109	**0**
	100	Aver	1.07E+02	2.00E+00	**0**	1.37E+32	7.63E+00	7.12E+01	3.45E+01	7.25E −21	8.5E −100	1.57E+02	2.0E −105	**0**
		STD	3.54E+01	4.84E+00	**0**	7.48E+32	4.71E −01	8.49E+00	3.47E+00	3.72E −21	4.05E −99	3.46E+01	1.1E −104	**0**
	500	Aver	1.42E+03	5.35E+02	1.42E −04	Inf	1.68E+02	5.14E+02	Inf	5.85E −09	5.6E −100	2.18E+03	4.0E −104	**0**
		STD	3.15E+02	5.06E+01	3.85E −04	Inf	5.56E+00	2.52E+01	Inf	1.28E −09	2.52E −99	7.51E+01	2.2E −103	**0**
F3	30	Aver	1.09E −23	2.18E −90	**0**	3.11E −08	9.12E −11	5.94E −13	4.39E −12	**0**	**0**	2.28E −20	**0**	**0**
		STD	4.07E −23	1.04E −89	**0**	9.34E −09	2.00E −10	8.26E −13	7.15E −12	**0**	**0**	1.20E −19	**0**	**0**
	50	Aver	1.84E −10	5.21E −51	**0**	3.90E −08	2.84E −11	1.92E −10	2.06E −11	**0**	**0**	9.92E −09	**0**	**0**
		STD	6.30E −10	2.82E −50	**0**	1.13E −08	3.54E −11	2.06E −10	6.43E −11	**0**	**0**	5.40E −08	**0**	**0**
	100	Aver	3.46E −01	2.10E −27	**0**	5.35E −08	2.33E −10	2.24E −09	2.29E −10	**0**	**0**	7.21E −05	**0**	**0**
		STD	4.03E −01	5.06E −27	**0**	1.50E −08	7.20E −10	3.33E −09	7.88E −10	**0**	**0**	3.09E −04	**0**	**0**
	500	Aver	2.19E+00	2.84E −14	**0**	8.51E −08	4.97E −11	2.14E −08	3.99E −09	5.16E −06	**0**	2.00E −01	**0**	**0**
		STD	4.64E −01	1.02E −13	**0**	2.78E −08	6.07E −11	3.10E −08	1.06E −08	2.33E −05	**0**	2.18E −01	**0**	**0**
F4	30	Aver	7.35E+00	1.70E+01	2.91E −03	1.92E+00	2.68E+01	2.35E+04	3.29E+02	2.38E −18	3.41E −163	1.60E+04	1.24E+04	**0**
		STD	4.14E+00	1.04E+01	9.44E −03	4.35E −01	8.70E+00	3.64E+03	1.58E+02	9.61E −18	2.22E −162	1.33E+04	7.98E+03	**0**
	50	Aver	3.85E+02	9.47E+03	4.95E −02	3.93E+01	5.04E+02	9.41E+04	1.40E+03	7.90E −10	6.0E −157	5.28E+04	8.57E+04	**0**
		STD	1.42E+02	4.33E+03	4.31E −02	9.97E+00	1.36E+02	8.36E+03	4.01E+02	1.94E −09	3.18E −156	2.73E+04	2.65E+04	**0**
	100	Aver	9.04E+03	1.37E+05	6.11E −01	9.71E+03	1.15E+04	3.85E+05	7.11E+03	3.68E −01	1.11E −132	1.74E+05	6.91E+05	**0**
		STD	2.56E+03	2.21E+04	5.59E −01	2.10E+03	1.87E+03	2.37E+04	2.57E+03	1.67E+00	6.09E −132	5.11E+04	1.17E+05	**0**
	500	Aver	4.03E+05	4.29E+06	2.79E+01	9.26E+05	8.64E+05	8.91E+06	6.93E+05	9.35E+04	1.39E −87	3.76E+06	2.74E+07	**0**
		STD	8.29E+04	6.82E+05	1.57E+01	7.86E+04	6.13E+04	6.84E+05	4.21E+05	5.04E+04	7.62E −87	6.09E+05	7.95E+06	**0**
F5	30	Aver	3.84E −01	2.33E+00	1.16E −02	3.19E −01	4.21E −01	6.17E+00	4.69E+00	2.14E −17	1.71E −98	5.25E+01	3.43E+01	**0**
		STD	1.08E −01	1.74E+00	1.83E −02	3.29E −02	6.40E −02	7.20E −01	1.39E+00	3.24E −17	4.75E −98	1.09E+01	2.91E+01	**0**
	50	Aver	1.95E+00	1.80E+01	4.81E −02	3.31E+00	1.55E+00	1.43E+01	9.03E+00	1.72E −11	6.29E −96	7.88E+01	5.95E+01	**0**
		STD	2.74E −01	3.98E+00	7.89E −03	3.21E+00	1.21E −01	1.54E+00	1.35E+00	1.65E −11	3.42E −95	3.82E+00	3.20E+01	**0**
	100	Aver	7.10E+00	3.87E+01	8.74E −02	2.79E+01	4.37E+00	2.35E+01	1.37E+01	1.66E −04	1.59E −96	9.20E+01	6.93E+01	**0**
		STD	1.31E+00	5.06E+00	1.14E −02	3.51E+00	1.73E −01	2.08E+00	1.84E+00	3.86E −04	8.22E −96	2.77E+00	2.82E+01	**0**
	500	Aver	2.46E+01	6.00E+01	1.61E −01	6.22E+01	4.56E+01	3.56E+01	2.15E+01	5.38E+01	2.41E −94	9.87E+01	7.68E+01	**0**
		STD	1.02E+00	2.58E+00	7.02E −03	1.89E+00	2.44E+00	1.71E+00	1.28E+00	7.55E+00	1.30E −93	3.89E −01	2.33E+01	**0**
F6	30	Aver	6.37E+01	2.94E+01	2.81E+01	1.06E+02	6.94E+01	1.98E+02	1.31E+02	2.67E+01	1.45E −03	6.69E+03	2.66E+01	**1.99E −15**
		STD	5.31E+01	2.62E+01	4.59E −01	1.27E+02	4.16E+01	7.15E+01	8.12E+01	8.48E −01	2.01E −03	2.27E+04	2.74E −01	**8.54E −15**
	50	Aver	1.42E+02	8.29E+01	4.84E+01	1.28E+02	3.95E+02	4.53E+03	3.18E+03	4.68E+01	1.75E −03	1.07E+07	4.69E+01	**2.77E −15**
		STD	8.68E+01	3.74E+01	3.17E −01	1.83E+02	5.76E+02	1.46E+03	2.88E+03	6.84E −01	2.26E −03	2.77E+07	4.17E −01	**1.09E −14**
	100	Aver	1.31E+03	3.41E+03	9.88E+01	2.77E+02	2.43E+03	3.95E+05	1.04E+05	9.72E+01	3.93E −03	4.86E+07	9.73E+01	**1.63E −14**
		STD	7.20E+02	1.64E+04	1.25E −01	1.88E+02	5.88E+02	1.06E+05	4.26E+04	9.64E −01	5.79E −03	5.33E+07	5.24E −01	**8.60E −14**
	500	Aver	6.20E+06	4.24E+07	4.99E+02	4.04E+05	5.02E+05	2.52E+07	3.64E+06	4.97E+02	1.69E −02	3.80E+09	4.95E+02	**3.16E −14**
		STD	6.65E+05	1.43E+07	6.33E −02	2.61E+05	4.15E+04	3.49E+06	5.20E+05	2.94E −01	1.71E −02	2.23E+08	2.78E −01	**8.11E −14**
F7	30	Aver	2.13E −11	**6.16E −33**	2.37E+00	5.64E −01	6.59E −02	3.66E −01	7.37E −01	4.46E −01	1.24E −05	1.35E+03	1.15E −02	2.83E −28
		STD	6.15E −11	**2.03E −32**	2.43E −01	7.98E −02	2.33E −02	1.55E −01	2.33E+00	2.94E −01	1.08E −05	5.77E+03	3.64E −02	4.20E −28
	50	Aver	4.08E −05	2.75E −27	6.30E+00	1.67E+00	5.49E+00	7.47E+01	3.47E+02	1.49E+00	2.20E −05	5.34E+03	4.90E −02	**9.37E −28**
		STD	7.81E −05	7.00E −27	3.46E −01	1.36E −01	1.06E+00	1.66E+01	1.84E+02	5.92E −01	4.10E −05	7.77E+03	3.28E −02	**2.66E −27**
	100	Aver	1.04E+00	2.90E −08	1.68E+01	6.56E+00	1.34E+02	2.59E+03	2.94E+03	7.72E+00	3.93E −05	2.92E+04	5.41E −01	**7.86E −27**

(continued on next page)

Table 7 (*continued*).

Fun	Dim	Method	PSO	DE	AOA	BBBC	BBO	CSAA	GSA	GWO	HHO	MFO	WOA	BWO
		STD	1.18E+00	5.17E−08	6.77E−01	4.21E−01	1.38E+01	4.68E+02	7.36E+02	7.70E−01	7.91E−05	1.25E+04	1.54E−01	**2.36E−26**
	500	Aver	1.67E+03	5.79E+04	1.14E+02	4.91E+02	7.41E+03	7.23E+04	3.35E+04	8.76E+01	2.02E−04	9.25E+05	9.54E+00	**1.08E−26**
		STD	1.50E+02	1.25E+04	1.24E+00	1.88E+02	2.17E+02	5.34E+03	2.06E+03	2.08E+00	3.19E−04	3.51E+04	1.94E+00	**2.41E−26**
F8	30	Aver	3.53E+00	1.07E−02	**1.36E−05**	1.20E−02	6.09E−03	3.91E−02	4.30E−02	5.92E−04	3.53E−05	2.48E+00	1.59E−03	3.49E−05
		STD	5.14E+00	4.56E−03	**1.26E−05**	4.10E−03	1.88E−03	1.09E−02	1.94E−02	3.97E−04	4.02E−05	4.84E+00	2.43E−03	2.31E−05
	50	Aver	2.97E+01	4.97E−02	**1.80E−05**	3.45E−02	1.86E−02	1.24E−01	2.50E−01	8.84E−04	3.00E−05	2.07E+01	1.36E−03	3.96E−05
		STD	2.53E+01	1.82E−02	**1.70E−05**	8.07E−03	3.97E−03	3.05E−02	1.08E−01	4.65E−04	1.87E−05	2.11E+01	1.49E−03	3.60E−05
	100	Aver	2.62E+02	3.99E−01	**2.05E−05**	1.30E−01	8.92E−02	9.63E−01	2.80E+00	1.57E−03	5.09E−05	8.47E+01	1.18E−03	4.09E−05
		STD	1.29E+02	1.98E−01	**2.00E−05**	2.74E−02	2.01E−02	2.05E−01	6.79E−01	5.94E−04	4.73E−05	5.54E+01	1.32E−03	3.25E−05
	500	Aver	2.53E+04	3.64E+02	**3.14E−05**	2.39E+01	5.74E+00	1.81E+02	4.75E+02	7.70E−03	4.03E−05	2.83E+04	1.05E−03	3.33E−05
		STD	3.04E+03	1.35E+02	**2.17E−05**	4.36E+00	3.47E−01	2.65E+01	1.07E+02	2.15E−03	4.06E−05	2.39E+03	1.27E−03	3.83E−05
F9	30	Aver	1.85E+02	4.96E+00	1.63E+02	6.31E−03	6.58E−02	1.82E+02	4.78E+01	8.03E−27	4.10E−136	2.57E+02	4.65E+02	**0**
		STD	1.30E+02	8.20E+00	5.75E+01	9.33E−04	1.80E−02	2.64E+01	1.72E+01	1.69E−26	1.16E−135	1.20E+02	1.11E+02	**0**
	50	Aver	7.50E+02	2.65E+02	6.48E+02	7.70E+00	1.07E+00	5.85E+02	1.99E+02	9.11E−14	3.49E−67	7.56E+02	8.74E+02	**0**
		STD	2.69E+02	8.12E+01	8.46E+01	6.96E+00	2.44E−01	6.56E+01	3.72E+01	2.09E−13	1.91E−66	1.35E+02	1.10E+02	**0**
	100	Aver	2.91E+03	1.51E+03	1.88E+03	9.06E+02	5.46E+01	1.96E+03	5.27E+02	1.42E−02	5.23E−36	2.35E+03	1.65E+03	**0**
		STD	4.58E+02	2.07E+02	1.94E+02	1.44E+02	1.44E+01	1.61E+02	7.67E+01	2.31E−02	2.86E−35	2.01E+02	2.63E+02	**0**
	500	Aver	8.04E+14	1.32E+04	1.46E+05	9.21E+03	5.13E+03	1.46E+04	2.36E+19	2.47E+03	1.41E+03	1.80E+04	8.04E+03	**0**
		STD	2.86E+15	6.62E+02	7.02E+05	3.95E+02	5.19E+02	1.39E+03	6.24E+18	4.28E+02	2.48E+03	7.58E+02	8.94E+02	**0**
F10	30	Aver	−6.86E+03	−9.81E+03	−6.26E+03	−6.97E+03	−8.14E+03	−7.53E+03	−2.82E+03	−6.40E+03	−1.26E+04	−8.53E+03	−1.16E+04	**−1.26E+04**
		STD	7.60E+02	7.20E+02	5.11E+02	1.32E+03	7.32E+02	8.65E+02	3.36E+02	6.22E+02	6.80E−01	8.44E+02	1.55E+03	**1.85E−12**
	50	Aver	−1.08E+04	−1.57E+04	−7.86E+03	−1.17E+04	−1.27E+04	−1.11E+04	−3.67E+03	−9.20E+03	−2.09E+04	−1.34E+04	−1.91E+04	**−2.09E+04**
		STD	1.50E+03	1.24E+03	6.97E+02	1.78E+03	7.71E+02	1.40E+03	8.11E+02	1.46E+03	6.51E+01	1.64E+03	2.15E+03	**3.70E−12**
	100	Aver	−1.99E+04	−3.01E+04	−1.13E+04	−2.39E+04	−2.26E+04	−1.62E+04	−5.42E+03	−1.78E+04	−4.18E+04	−2.42E+04	−3.93E+04	**−4.19E+04**
		STD	3.11E+03	1.14E+03	8.06E+02	1.57E+03	1.18E+03	2.62E+03	7.46E+02	1.48E+03	4.17E+02	2.68E+03	3.52E+03	**7.40E−12**
	500	Aver	−9.63E+04	−1.27E+05	−2.58E+04	−1.03E+05	−9.04E+04	−3.98E+04	−1.14E+04	−6.34E+04	−2.09E+05	−7.51E+04	−1.96E+05	**−2.09E+05**
		STD	1.60E+04	5.03E+03	1.34E+03	3.19E+04	2.99E+03	7.05E+03	1.72E+03	1.01E+04	2.09E+00	4.35E+03	2.16E+04	**8.88E−11**
F11	30	Aver	2.99E+00	1.02E+00	**0**	1.51E+00	1.00E+00	5.57E+00	8.36E+00	1.24E+00	**0**	4.40E+00	5.60E−01	**0**
		STD	9.32E−01	7.51E−02	**0**	1.93E+00	1.95E−04	3.82E−01	2.10E+00	3.01E−01	**0**	1.06E+00	5.74E−01	**0**
	50	Aver	5.62E+00	1.03E+00	**0**	4.25E+00	1.05E+00	1.06E+01	1.41E+01	1.70E+00	**0**	7.27E+00	5.00E−01	**0**
		STD	1.89E+00	9.03E−02	**0**	5.98E+00	1.13E−02	7.63E−01	5.42E+00	2.05E+00	**0**	1.36E+00	7.62E−01	**0**
	100	Aver	1.62E+01	1.15E+00	**0**	3.12E+01	2.24E+00	2.55E+01	3.49E+01	1.67E+00	**0**	1.51E+01	4.90E−01	**0**
		STD	4.86E+00	2.30E−01	**0**	5.74E+00	1.51E−01	1.21E+00	6.83E+00	5.00E−01	**0**	1.44E+00	1.31E+00	**0**
	500	Aver	1.34E+02	1.85E+01	3.59E−06	1.72E+02	5.72E+01	1.43E+02	2.00E+02	2.41E+00	**0**	9.78E+01	3.76E−01	**0**
		STD	5.09E+00	2.16E+00	1.87E−06	2.93E+00	1.83E+00	3.38E+00	9.75E+00	4.64E−01	**0**	3.05E+00	2.06E+00	**0**
F12	30	Aver	−1.08E+03	−1.08E+03	−5.05E+02	−8.60E+02	−1.07E+03	−8.79E+02	−6.91E+02	−9.29E+02	−1.17E+03	−1.03E+03	−1.16E+03	**−1.17E+03**
		STD	2.85E+01	2.92E+01	5.05E+01	2.33E+02	3.13E+01	4.64E+01	2.47E+02	5.65E+01	1.29E−03	3.23E+01	4.94E+01	**2.31E−13**
	50	Aver	−1.74E+03	−1.75E+03	−7.31E+02	−1.31E+03	−1.76E+03	−1.18E+03	−8.92E+02	−1.40E+03	−1.96E+03	−1.67E+03	−1.91E+03	**−1.96E+03**
		STD	4.46E+01	5.05E+01	7.90E+01	4.54E+02	3.92E+01	4.24E+01	3.77E+01	7.11E+01	1.23E−03	6.10E+01	9.05E+01	**4.63E−13**
	100	Aver	−3.29E+03	−3.42E+03	−1.17E+03	−3.06E+03	−3.40E+03	−1.93E+03	−1.72E+03	−2.43E+03	−3.92E+03	−3.26E+03	−3.88E+03	**−3.92E+03**
		STD	1.29E+02	5.58E+01	1.04E+02	6.98E+02	6.23E+01	6.09E+01	4.83E+02	1.26E+02	5.60E−03	1.13E+02	9.17E+01	**2.32E−12**
	500	Aver	−9.90E+03	−1.55E+04	−3.86E+03	−1.58E+04	−1.62E+04	−7.42E+03	−6.78E+03	−8.41E+03	−1.96E+04	−9.24E+03	−1.95E+04	**−1.96E+04**
		STD	5.55E+02	2.50E+02	3.47E+02	2.35E+03	1.75E+02	1.18E+02	1.57E+02	4.82E+02	6.68E−03	2.47E+02	2.55E+02	**2.79E−12**
F13	30	Aver	9.11E+01	3.25E+01	**0**	1.10E+02	6.40E+01	1.95E+02	4.30E+01	3.06E−01	**0**	1.42E+02	**0**	**0**
		STD	2.68E+01	6.23E+00	**0**	2.81E+01	1.28E+01	1.30E+01	5.95E+01	1.16E+00	**0**	2.95E+01	**0**	**0**
	50	Aver	2.35E+02	7.70E+01	**0**	2.44E+02	1.25E+02	3.89E+02	7.21E+01	7.58E−15	**0**	2.98E+02	**0**	**0**
		STD	3.48E+01	1.55E+01	**0**	5.58E+01	2.44E+01	2.39E+01	1.01E+02	1.97E−14	**0**	4.47E+01	**0**	**0**
	100	Aver	6.70E+02	2.35E+02	**0**	5.61E+02	3.54E+02	8.85E+02	3.12E+02	2.56E−01	**0**	6.93E+02	**0**	**0**
		STD	7.76E+01	3.82E+01	**0**	8.52E+01	4.49E+01	2.23E+01	2.27E+02	8.08E−01	**0**	6.75E+01	**0**	**0**
	500	Aver	5.43E+03	2.29E+03	2.53E−06	3.59E+03	3.65E+03	4.83E+03	3.26E+03	4.52E+00	**0**	6.21E+03	**0**	**0**
		STD	1.38E+02	1.67E+02	4.16E−06	1.67E+02	1.26E+02	5.84E+01	1.98E+02	6.32E+00	**0**	1.55E+02	**0**	**0**
F14	30	Aver	2.01E−06	8.51E−01	**8.88E−16**	1.96E+01	5.66E−02	9.01E+00	9.47E−01	1.44E−14	**8.88E−16**	1.05E+01	4.09E−15	**8.88E−16**
		STD	2.67E−06	1.07E+00	**0**	3.65E+00	8.73E−03	5.00E+00	6.88E−01	2.54E−15	**0**	9.64E+00	2.70E−15	**0**
	50	Aver	1.62E−01	1.93E+00	**8.88E−16**	1.97E+01	7.27E−01	1.91E+01	3.25E+00	2.63E−14	**8.88E−16**	1.93E+01	4.20E−15	**8.88E−16**
		STD	3.57E−01	7.01E−01	**0**	3.66E+00	1.13E−01	1.69E+00	3.68E−01	3.74E−15	**0**	1.23E+00	2.27E−15	**0**
	100	Aver	2.13E+00	5.63E+00	4.76E−05	2.05E+01	2.81E+00	1.99E+01	5.86E+00	6.99E−14	**8.88E−16**	1.98E+01	3.61E−15	**8.88E−16**
		STD	3.48E−01	1.98E+00	2.61E−04	5.50E−02	1.08E−01	5.19E−01	4.49E−01	4.73E−15	**0**	3.40E−01	2.22E−15	**0**
	500	Aver	8.70E+00	1.70E+01	7.48E−03	2.07E+01	6.23E+00	2.00E+01	9.19E+00	7.60E−09	**8.88E−16**	2.01E+01	4.09E−15	**8.88E−16**
		STD	3.23E−01	5.47E−01	3.13E−04	2.41E−02	8.51E−02	1.72E−04	2.48E−01	1.49E−09	**0**	1.07E−01	2.53E−15	**0**
F15	30	Aver	6.32E−03	5.08E−03	8.23E−02	6.83E−01	1.77E−01	6.72E−01	1.65E+01	1.12E−03	**0**	1.51E+01	1.59E−03	**0**
		STD	7.86E−03	9.85E−03	6.16E−02	5.95E−02	5.74E−02	1.20E−01	4.23E+00	4.84E−03	**0**	3.42E+01	8.73E−03	**0**
	50	Aver	5.42E−03	3.46E−02	3.69E−01	9.05E−01	1.04E+00	1.70E+00	9.79E+01	1.34E−03	**0**	5.79E+01	4.85E−03	**0**
		STD	6.94E−03	9.38E−02	1.79E−01	4.70E−02	1.20E−02	1.42E−01	1.28E+01	5.96E−03	**0**	5.54E+01	1.88E−02	**0**

(continued on next page)

Table 7 (*continued*).

Fun	Dim	Method	PSO	DE	AOA	BBBC	BBO	CSAA	GSA	GWO	HHO	MFO	WOA	BWO
	100	Aver	1.83E −02	6.37E −02	3.48E+02	1.05E+00	2.15E+00	2.29E+01	6.08E+02	1.00E −03	**0**	2.75E+02	**0**	**0**
		STD	1.60E −02	1.04E −01	1.86E+02	1.10E −02	1.09E −01	3.96E+00	4.56E+01	3.92E −03	**0**	1.10E+02	**0**	**0**
	500	Aver	1.63E+00	5.07E+02	8.25E+03	5.54E+00	6.79E+01	6.42E+02	8.34E+03	2.82E −03	**0**	8.13E+03	**0**	**0**
		STD	6.44E −02	9.83E+01	1.65E+03	8.48E −01	2.61E+00	5.66E+01	1.35E+02	7.46E −03	**0**	3.24E+02	**0**	**0**
F16	30	Aver	8.25E −24	1.74E −43	5.96E −08	3.07E −15	1.59E −16	7.10E −12	1.73E −10	3.44E −15	−1	5.38E −12	−2.00E −01	−1
		STD	2.87E −23	3.68E −44	4.59E −08	2.16E −15	5.87E −17	2.78E −12	2.56E −10	1.82E −14	**0**	1.42E −11	4.07E −01	**0**
	50	Aver	5.32E −22	1.97E −48	1.73E −12	5.30E −23	8.73E −23	1.05E −18	2.43E −16	6.15E −24	−1	7.12E −20	−2.33E −01	−1
		STD	1.03E −21	2.76E −48	1.72E −12	4.33E −23	2.85E −23	5.62E −19	3.86E −16	6.57E −24	**0**	1.62E −19	4.30E −01	**0**
	100	Aver	2.22E −40	1.21E −42	9.49E −24	1.15E −42	1.33E −42	1.04E −35	3.45E −35	2.90E −41	−1	1.43E −37	−2.67E −01	−1
		STD	3.91E −40	3.59E −43	1.70E −23	8.20E −46	1.72E −44	8.50E −36	2.85E −35	7.26E −41	**0**	5.37E −37	4.50E −01	**0**
	500	Aver	0	0	0	0	0	0	0	0	−1	0	−4.33E −01	−1
		STD	0	0	0	0	0	0	0	0	**0**	0	5.04E −01	**0**
F17	30	Aver	4.90E −14	5.26E −01	3.25E −01	6.79E −02	1.59E −04	4.09E −01	4.23E −01	2.24E −02	1.13E −06	3.09E −01	7.92E −02	**2.92E −26**
		STD	9.05E −14	7.98E −01	3.65E −02	2.00E −01	1.15E −04	2.20E −01	3.51E −01	1.43E −02	1.93E −06	7.47E −01	4.30E −01	**5.20E −26**
	50	Aver	1.87E −02	3.92E −01	6.12E −01	3.61E+00	7.39E −03	4.87E+00	1.53E+00	5.55E −02	4.51E −07	3.41E+07	3.09E −03	**7.43E −27**
		STD	4.37E −02	6.42E −01	2.67E −02	1.69E+00	2.01E −03	1.23E+00	6.31E −01	2.03E −02	6.52E −07	1.11E+08	3.55E −03	**1.82E −26**
	100	Aver	6.69E −01	3.04E+00	8.23E −01	8.96E+00	2.73E −01	2.29E+01	4.55E+00	1.78E −01	3.06E −07	7.55E+07	5.75E −03	**7.60E −28**
		STD	5.94E −01	1.84E+00	2.35E −02	1.74E+00	9.81E −02	9.36E+00	1.16E+00	5.24E −02	3.57E −07	1.19E+08	2.30E −03	**1.35E −27**
	500	Aver	9.23E+03	1.47E+07	1.05E+00	9.19E+01	7.56E+01	7.96E+05	2.37E+01	6.93E −01	1.58E −07	8.76E+09	1.48E −02	**6.30E −29**
		STD	6.48E+03	7.11E+06	9.78E −03	2.02E+01	9.17E+01	3.47E+05	9.37E+00	3.60E −02	3.02E −07	7.57E+08	3.90E −03	**1.76E −28**
F18	30	Aver	3.30E −03	2.83E −01	2.77E+00	2.54E −02	1.86E −03	4.66E −01	3.07E+00	3.17E −01	8.72E −06	2.81E −01	4.62E −02	**3.89E −25**
		STD	5.12E −03	7.57E −01	1.30E −01	4.02E −03	7.08E −04	1.66E −01	3.22E+00	1.81E −01	1.21E −05	6.67E −01	5.02E −02	**7.10E −25**
	50	Aver	4.36E −03	9.84E −01	4.80E+00	7.64E −02	2.22E −01	2.40E+01	3.69E+01	1.33E+00	1.03E −05	2.73E+07	2.35E −01	**5.79E −26**
		STD	6.65E −03	1.32E+00	1.15E −01	1.08E −02	5.07E −02	7.02E+00	1.43E+01	3.16E −01	1.61E −05	1.50E+08	1.78E −01	**1.03E −25**
	100	Aver	4.47E+00	3.74E+01	9.92E+00	2.05E+00	5.34E+00	5.47E+04	1.50E+02	5.61E+00	1.51E −05	2.18E+08	6.90E −01	**2.45E −26**
		STD	2.55E+00	2.45E+01	6.51E −02	7.22E+00	4.91E −01	4.20E+04	7.82E+01	3.71E −01	2.14E −05	2.63E+08	3.43E −01	**4.01E −26**
	500	Aver	2.86E+05	9.58E+07	5.02E+01	3.96E+03	2.72E+04	2.11E+07	3.84E+05	4.52E+01	2.19E −05	1.65E+10	4.63E+00	**7.19E −27**
		STD	8.59E+04	4.41E+07	3.21E −02	2.80E+03	1.27E+04	5.00E+06	1.82E+05	4.30E −01	2.77E −05	1.28E+09	1.62E+00	**1.89E −26**

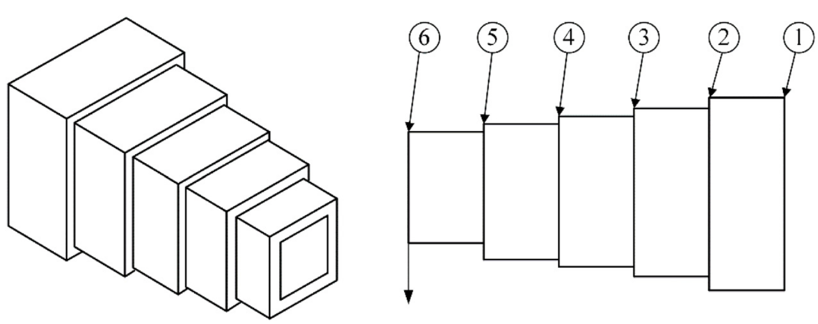

Fig. 5. Schematic of cantilever beam.

according to the probability of whale fall. The balance factor and probability of whale fall are self-adaptive which greatly affect the capacity of exploration and exploitation.

To test the performance of BWO, a series of experiments based on benchmark functions were conducted, consisting of qualitative analysis, quantitative analysis, and test of scalability. First, the qualitative analysis is implemented, including the search history, the curve of balance factor, trajectory, the average fitness curve and the convergence curve. The results demonstrate that BWO can provide good balance between the exploration phase

Table 8
Ranking-based Friedman test for algorithms on functions (F1–F18) with different dimensions.

Fun	Dim	PSO	DE	AOA	BBBC	BBO	CSAA	GSA	GWO	HHO	MFO	WOA	BWO
F1	30	7	5	6	10	8	9	11	4	2	12	3	**1**
	50	7	5	6	8	9	10	11	4	2	12	3	**1**
	100	7	5	6	8	9	10	11	4	2	12	3	**1**
	500	7	10	5	6	8	11	9	4	2	12	3	**1**
F2	30	10	6	**1**	11	7	9	8	5	4	12	3	**1**
	50	10	6	**1**	12	7	9	8	5	4	11	3	**1**
	100	10	6	**1**	12	7	9	8	5	4	11	3	**1**
	500	9	8	5	11	6	7	12	4	3	10	2	**1**
F3	30	7	6	**1**	12	11	9	10	**1**	**1**	8	**1**	**1**
	50	9	6	**1**	12	8	10	7	**1**	**1**	11	**1**	**1**
	100	12	6	**1**	10	8	9	7	**1**	**1**	11	**1**	**1**
	500	12	5	1	9	6	8	7	10	**1**	11	**1**	**1**
F4	30	6	7	4	5	8	12	9	3	2	11	10	**1**
	50	6	9	4	5	7	12	8	3	2	10	11	**1**
	100	6	9	4	7	8	11	5	3	2	10	12	**1**
	500	5	10	3	8	7	11	6	4	2	9	12	**1**
F5	30	6	8	4	5	7	10	9	3	2	12	11	**1**
	50	6	10	4	7	5	9	8	3	2	12	11	**1**
	100	6	10	4	9	5	8	7	3	2	12	11	**1**
	500	5	9	3	10	7	6	4	8	2	12	11	**1**
F6	30	7	6	5	9	8	11	10	4	2	12	3	**1**
	50	8	6	5	7	9	11	10	3	2	12	4	**1**
	100	7	9	5	6	8	11	10	3	2	12	4	**1**
	500	9	11	5	6	7	10	8	4	2	12	3	**1**
F7	30	3	**1**	11	9	6	7	10	8	4	12	5	2
	50	4	2	9	7	8	10	11	6	3	12	5	**1**
	100	5	2	8	6	9	10	11	7	3	12	4	**1**
	500	7	10	5	6	8	11	9	4	2	12	3	**1**
F8	30	12	7	**1**	8	6	9	10	4	3	11	5	2
	50	12	8	**1**	7	6	9	10	4	2	11	5	3
	100	12	8	**1**	7	6	9	10	5	3	11	4	2
	500	11	9	**1**	7	6	8	10	5	3	12	4	2
F9	30	10	6	8	4	5	9	7	3	2	11	12	**1**
	50	10	7	9	5	4	8	6	3	2	11	12	**1**
	100	12	7	9	6	4	10	5	3	2	11	8	**1**
	500	11	7	10	6	4	8	12	3	2	9	5	**1**
F10	30	9	4	11	8	6	7	12	10	2	5	3	**1**
	50	9	4	11	7	6	8	12	10	2	5	3	**1**
	100	8	4	11	6	7	10	12	9	2	5	3	**1**
	500	6	4	11	5	7	10	12	9	2	8	3	**1**
F11	30	9	6	2	8	5	11	12	7	3	10	4	**1**
	50	9	5	2	8	6	11	12	7	3	10	4	**1**
	100	9	5	2	11	7	10	12	6	3	8	4	**1**
	500	9	6	3	11	7	10	12	5	2	8	4	**1**
F12	30	4	5	12	10	6	9	11	8	2	7	3	**1**
	50	6	5	12	9	4	10	11	8	2	7	3	**1**
	100	6	4	12	8	5	10	11	9	2	7	3	**1**
	500	7	6	12	5	4	10	11	9	2	8	3	**1**
F13	30	9	6	2	10	8	12	7	5	3	11	4	**1**
	50	9	7	2	10	8	12	6	5	3	11	4	**1**
	100	10	6	2	9	8	12	7	5	3	11	4	**1**
	500	11	6	4	8	9	10	7	5	2	12	3	**1**
F14	30	6	8	2	12	7	10	9	5	3	11	4	**1**
	50	6	8	2	12	7	10	9	5	3	11	4	**1**
	100	6	8	5	12	7	11	9	4	2	10	3	**1**
	500	7	9	5	12	6	10	8	4	2	11	3	**1**
F15	30	6	5	7	10	8	9	12	3	2	11	4	**1**
	50	5	6	7	8	9	10	12	2	3	11	4	**1**
	100	5	6	11	7	8	9	12	4	2	10	3	**1**
	500	5	8	11	6	7	9	12	4	2	10	3	**1**
F16	30	5	4	12	7	6	10	11	8	2	9	3	**1**
	50	8	4	12	6	7	10	11	5	2	9	3	**1**
	100	8	5	12	4	6	10	11	7	2	9	3	**1**
	500	8	6	12	5	4	10	9	7	2	11	3	**1**

(continued on next page)

Table 8 (*continued*).

Fun	Dim	PSO	DE	AOA	BBBC	BBO	CSAA	GSA	GWO	HHO	MFO	WOA	BWO
F17	30	2	12	9	6	4	10	11	5	3	8	7	**1**
	50	5	7	8	10	4	11	9	6	2	12	3	**1**
	100	6	8	7	10	5	11	9	4	2	12	3	**1**
	500	9	11	5	8	7	10	6	4	2	12	3	**1**
F18	30	4	8	11	5	3	10	12	9	2	7	6	**1**
	50	3	7	9	4	5	10	11	8	2	12	6	**1**
	100	5	9	8	4	6	11	10	7	2	12	3	**1**
	500	8	11	5	6	7	10	9	4	2	12	3	**1**
Friedman mean rank		7.50	6.74	5.96	7.92	6.64	9.76	9.49	5.13	2.28	10.38	4.60	1.08
rank		8	7	5	9	6	11	10	4	2	12	3	1

Table 9
Comparison results for cantilever beam design problem.

Algorithms	Optimal values for variables					f_{opt}
	x_1	x_2	x_3	x_4	x_5	
GEO [84]	6.0157	5.3093	4.4944	3.5016	2.1527	13.3652
GA [84]	6.0439	5.2981	4.4836	3.4868	2.1618	13.3656
PSO	5.9788	4.8777	4.4680	3.4764	2.1382	13.0325
AOA	6.7929	6.1445	5.3520	2.5689	2.2459	14.3800
BBBC	5.9623	4.8987	4.4739	3.4935	2.1120	13.0333
GWO	5.9828	4.8711	4.4626	3.4853	2.1373	13.0325
GSA	5.9770	4.8787	4.4632	3.4813	2.1389	13.0325
WOA	5.9548	5.2006	4.1404	3.5160	2.2279	13.0951
HHO	6.0109	4.8220	4.5015	3.4459	2.1621	13.0346
BWO	6.0351	4.8313	4.4690	3.4503	2.1587	13.0358

Table 10
Comparison results for welded beam design problem.

Algorithms	Optimal values for variables				f_{opt}
	x_1 (h)	x_2 (L)	x_3 (t)	x_4 (b)	
RO [46]	0.2037	3.5285	9.0042	0.2072	1.7353
BOA [68]	0.2057	3.4705	9.0366	0.2057	1.7249
PSO	0.2057	3.2531	9.0366	0.2057	1.6952
AOA	0.2042	3.3493	10	0.2062	1.8757
BBBC	0.2055	3.2569	9.0377	0.2057	1.6957
GWO	0.2057	3.2541	9.0361	0.2058	1.6956
GSA	0.1250	6.2285	8.5101	0.2320	2.0287
WOA	0.1940	3.4480	9.1304	0.2068	1.7282
HHO	0.2066	3.2758	9.0153	0.2067	1.7033
BWO	0.2059	3.2665	9.0229	0.2064	1.6997

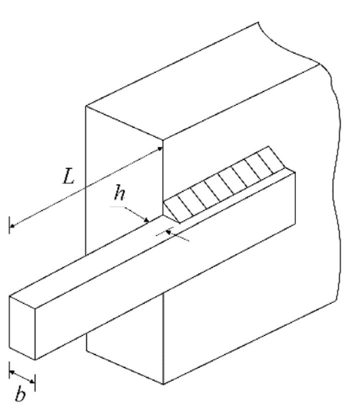

Fig. 6. Schematic of welded beam.

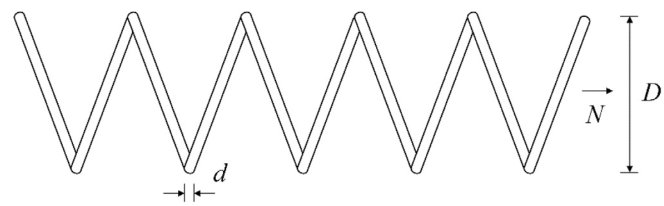

Fig. 7. Schematic of tension/compression spring.

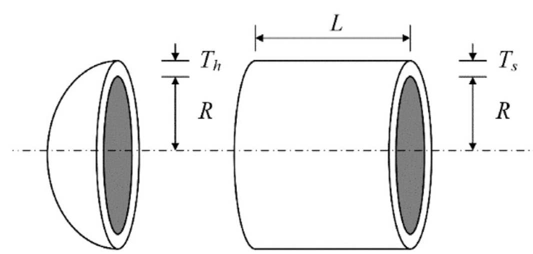

Fig. 8. Schematic of pressure vessel.

Table 11
Comparison results for tension/compression spring design problem.

Algorithms	Optimal values for variables			f_{opt}
	x_1 (d)	x_2 (D)	x_3 (N)	
PSO	0.0514	0.3500	11.6944	0.012667
AOA	0.0500	0.3105	15.0000	0.013195
BBBC	0.0517	0.3570	11.2753	0.012670
GWO	0.0519	0.3626	10.9606	0.012672
GSA	0.0513	0.3486	11.7834	0.012667
WOA	0.0525	0.3759	10.2481	0.012676
HHO	0.0517	0.3581	11.2106	0.012665
BWO	0.0517	0.3568	11.3132	0.012703

Table 12
Comparison results for pressure vessel design problem.

Algorithms	Optimal values for variables				f_{opt}
	x_1 (T_s)	x_2 (T_h)	x_3 (R)	x_4 (L)	
PSO	0.7911	0.3911	40.9912	190.8581	5907.979
AOA	0.9393	0.6038	46.0635	184.5142	8569.154
BBBC	0.7989	0.3993	41.3750	186.2517	5947.589
GWO	0.7782	0.3853	40.3197	200.0000	5887.323
GSA	0.9391	0.4642	48.6595	109.3493	6221.299
WOA	0.7816	0.3855	40.3196	200.0000	5912.401
HHO	0.7784	0.4128	40.3296	199.8615	5966.674
BWO	0.7796	0.3921	40.3598	199.4567	5912.114

and the exploitation phase in solving benchmark optimization problems. Second, 30 well-known benchmark functions with unimodal functions, multimodal functions and composite functions were conducted to test the BWO, while the results are compared with 15 other metaheuristic algorithms. The results show that BWO achieves the first rank in 23 out of 30 functions, which is competitive among the compared algorithms. Third, the results of the scalability analysis from benchmark functions F1–F18 with different dimensions indicate that BWO achieves the best average and standard deviation values in 67 out of 72 cases (93%), which is higher than AOA (15.3%), HHO (5.6%), WOA (5.6%), GWO (4.2%) and other algorithms (0%). Finally, four engineering problems in different fields were implemented, including cantilever beam design problem, welded beam design problem, tension/compression spring design problem, and pressure vessel design problem. The results of engineering problems demonstrate the practical metrics of the proposed BWO.

Based on the above results, analysis and discussion of the experiments support the following conclusions:

(1) BWO is a derivative-free optimization technique and easy to implement.

(2) BWO can provide good ability to balance the exploration and exploitation phase to ensure the global convergence.

(3) BWO performs well for unimodal and multimodal functions, especially outstanding for scalability analysis, and provides competitiveness for composite functions.

(4) BWO is competitive in solving real-world engineering problems.

In our future work, the binary version of BWO can be investigated to solve discrete problems. Several learning operators can be also incorporated with BWO such as comprehensive learning, opposition-based learning. Besides, the multi-objective BWO is expected to develop. Moreover, BWO can also be expanded to solve different optimization problems in various fields, such as neural networks, feature selection, shop scheduling, photovoltaic models, big data applications, and so on.

CRediT authorship contribution statement

Changting Zhong: Conceptualization, Methodology, Software, Validation, Writing – original draft, Writing – review & editing. **Gang Li:** Conceptualization, Methodology, Supervision, Funding acquisition, Writing – review & editing. **Zeng Meng:** Conceptualization, Supervision, Writing – review & editing.

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.

Acknowledgments

The support of the National Key Research and Development Program, China (Grant No. 2019YFA0706803) and the National Natural Science Foundation of China (Grant No. 11872142) is greatly appreciated.

Appendix

I — Cantilever beam design problem

Find $\vec{x} = [x_1, x_2, x_3, x_4, x_5]$

Min $f(\vec{x}) = 0.0624\,(x_1 + x_2 + x_3 + x_4 + x_5)$

s.t. $g_1(\vec{x}) = \dfrac{61}{x_1^3} + \dfrac{37}{x_2^3} + \dfrac{19}{x_3^3} + \dfrac{7}{x_4^3} + \dfrac{1}{x_5^3} - 1 \le 0$

Range $0.01 \le x_i \le 100 (i = 1, 2, \ldots, 5)$

II — Welded beam design problem

Find $\vec{x} = [x_1, x_2, x_3, x_4] = [h \quad L \quad t \quad b]$

Min $f(\vec{x}) = 1.10471 x_1^2 x_2 + 0.04811 x_3 x_4 (14 + x_2)$

s.t.

$$g_1(\vec{x}) = \tau(\vec{x}) - \tau_{\max} \le 0,$$
$$g_2(\vec{x}) = \sigma(\vec{x}) - \sigma_{\max} \le 0,$$
$$g_3(\vec{x}) = x_1 - x_4 \le 0,$$
$$g_4(\vec{x}) = 1.10471 x_1^2 + 0.04811 x_3 x_4 (14.0 + x_2) - 5.0 \le 0,$$
$$g_5(\vec{x}) = 0.125 - x_1 \le 0,$$
$$g_6(\vec{x}) = \delta(\vec{x}) - \delta_{\max} \le 0,$$
$$g_7(\vec{x}) = P - P_c(\vec{x}) \le 0$$

Range $0.1 \le x_1 \le 2, \quad 0.1 \le x_2 \le 10, \quad 0.1 \le x_3 \le 10, \quad 0.1 \le x_4 \le 2,$

$$\text{where } \tau(\vec{x}) = \sqrt{(\tau')^2 + 2\tau'\tau''\frac{x_2}{2R} + (\tau'')^2},$$
$$\tau' = \frac{P}{\sqrt{2}x_1x_2}, \tau'' = \frac{MR}{J}, M = P\left(L + \frac{x_2}{2}\right),$$
$$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2},$$
$$J = 2\left\{\sqrt{2}x_1x_2\left[\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2\right]\right\},$$
$$\sigma(\vec{x}) = \frac{6PL}{x_4x_3^2}, \delta(\vec{x}) = \frac{6PL^3}{Ex_3^2x_4},$$
$$P_c(\vec{x}) = \frac{4.013E\sqrt{\frac{x_3^2x_4^6}{36}}}{L^2}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right),$$
$$P = 6000lb, L = 14 \text{ in.}, E = 30 \times 10^6 \text{ psi},$$
$$G = 12 \times 10^6 \text{psi},$$
$$\tau_{max} = 13600 \text{ psi}, \sigma_{max} = 30000 \text{ psi}, \delta_{max} = 0.25 \text{ in.}$$

III — Tension/compression spring design problem

Find $\vec{x} = [x_1, x_2, x_3] = [d \quad D \quad N]$

Min $f(\vec{x}) = (x_3 + 2)\, x_2 x_1^2$

s.t.

$$g_1(\vec{x}) = 1 - \frac{x_2^3 x_3}{71785 x_1^4} \le 0,$$
$$g_2(\vec{x}) = \frac{4x_2^2 - x_1x_2}{12566\left(x_2x_1^3 - x_1^4\right)} + \frac{1}{5108x_1^2} \le 0,$$
$$g_3(\vec{x}) = 1 - \frac{140.45x_1}{x_2^2x_3} \le 0,$$
$$g_4(\vec{x}) = \frac{x_1 + x_2}{1.5} - 1 \le 0,$$

Range $0.05 \le x_1 \le 2.00, \quad 0.23 \le x_2 \le 1.30, \quad 2.00 \le x_3 \le 15.0.$

IV — Pressure vessel design problem

Find $\vec{x} = [x_1, x_2, x_3, x_4] = [T_s \quad T_h \quad R \quad L]$

Min $f(\vec{x}) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3$

s.t.

$$g_1(\vec{x}) = -x_1 + 0.0193x_3 \le 0,$$
$$g_2(\vec{x}) = -x_2 + 0.00954x_3 \le 0,$$
$$g_3(\vec{x}) = -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \le 0,$$

$$g_4\left(\overrightarrow{x}\right) = x_4 - 240 \le 0$$

Range $\quad 0\le x_1\le 99, \quad 0\le x_2\le 99, \quad 10\le x_3\le 200, \quad 10\le x_4\le 200,$

References

[1] J. Del Ser, E. Osaba, D. Molina, X.-S. Yang, S. Salcedo-Sanz, D. Camacho, S. Das, P.N. Suganthan, C.A. Coello Coello, F. Herrera, Bio-inspired computation: Where we stand and what's next? Swarm Evol. Comput. 48 (2019) 220–250, http://dx.doi.org/10.1016/j.swevo.2019.04.008.

[2] A. Kaveh, Advances in Metaheuristics Algorithms for Optimal Design of Structures, third ed., Springer International Publishing, Switzerland, 2021.

[3] R. Tanabe, H. Ishibuchi, A review of evolutionary multimodal multiobjective optimization, IEEE Trans. Evol. Comput. 24 (1) (2020) 193–200, http://dx.doi.org/10.1109/TEVC.2019.2909744.

[4] H. Khattab, A. Sharieh, B.A. Mahafzah, Most valuable player algorithm for solving minimum vertex cover problem, Int. J. Adv. Comput. Sci. Appl. 10 (8) (2019) 159–167, http://dx.doi.org/10.14569/IJACSA.2019.0100821.

[5] E.H. Houssein, A.G. Gad, Y.M. Wazery, P.N. Suganthan, Task scheduling in cloud computing based on meta-heuristics: review, taxonomy, open challenges, and future trends, Swarm Evol. Comput. 62 (2021) 100841, http://dx.doi.org/10.1016/j.swevo.2021.100841.

[6] R. Masadeh, N. Alsharman, A. Sharieh, B.A. Mahafzah, A. Abdulrahman, Task scheduling on cloud computing based on sea lion optimization algorithm, Int. J. Web Inf. Syst. 17 (2) (2021) 99–116, http://dx.doi.org/10.1108/IJWIS-11-2020-0071.

[7] B.A. Mahafzah, R. Jabri, O. Murad, Multithreaded scheduling for program segments based on chemical reaction optimizer, Soft Comput. 25 (2021) 2741–2766, http://dx.doi.org/10.1007/s00500-020-05334-4.

[8] P. Singh, S. Chaudhury, B.K. Panigrahi, Hybrid MPSO-CNN: Multi-level particle swarm optimized hyperparameters of convolutional neural network, Swarm Evol. Comput. 63 (2021) 100863, http://dx.doi.org/10.1016/j.swevo.2021.100863.

[9] D. Paul, A. Jain, S. Saha, J. Mathew, Multi-objective PSO based online feature selection for multi-label classification, Knowl.-Based Syst. 222 (2021) 106966, http://dx.doi.org/10.1016/j.knosys.2021.106966.

[10] D. Zhao, L. Liu, F. Yu, A.A. Heidari, M. Wang, D. Oliva, K. Muhammad, H. Chen, Ant colony optimization with horizontal and vertical crossover search: fundamental visions for multi-threshold image segmentation, Expert Syst. Appl. 167 (2021) 114122, http://dx.doi.org/10.1016/j.eswa.2020.114122.

[11] M. Gheisarnejad, An effective hybrid harmony search and cuckoo optimization algorithm based fuzzy PID controller for load frequency control, Appl. Soft Comput. 65 (2018) 121–138, http://dx.doi.org/10.1016/j.asoc.2018.01.007.

[12] D. Yousri, M. Abd Elaziz, D. Oliva, L. Abualigah, M.A.A. Al-qaness, A.A. Ewees, Reliable applied objective for identifying simple and detailed photovoltaic models using modern metaheuristics: comparative study, Energy Convers. Manag. 223 (2020) 113279, http://dx.doi.org/10.1016/j.enconman.2020.113279.

[13] G. Li, H. Hu, Risk design optimization using many-objective evolutionary algorithm with application to performance-based wind engineering of tall buildings, Struct. Saf. 48 (2014) 1–14, http://dx.doi.org/10.1016/j.strusafe.2014.01.002.

[14] A. Kaveh, Applications of Metaheuristic Optimization Algorithms in Civil Engineering, Springer, Switzerland, 2017.

[15] Z. Meng, G. Li, X. Wang, S.M. Said, A.R. Yildiz, A comparative study of metaheuristic algorithms for reliability-based design optimization problems, Arch. Comput. Methods Eng. 28 (2021) 1853–1869, http://dx.doi.org/10.1007/s11831-020-09443-z.

[16] X.S. Yang, Recent Advances in Swarm Intelligence and Evolutionary Computation, Middlesex University, Springer, London, 2015, http://dx.doi.org/10.1007/978-3-319-13826-8.

[17] E.H. Houssein, A.G. Gad, K. Hussain, P.N. Suganthan, Major advances in particle swarm optimization: theory, analysis and application, Swarm Evol. Comput. 63 (2021) 100868, http://dx.doi.org/10.1016/j.swevo.2021.100868.

[18] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197, http://dx.doi.org/10.1109/4235.996017.

[19] B.Y. Qu, P.N. Suganthan, J.J. Liang, Differential evolution with neighborhood mutation for multimodal optimization, IEEE Trans. Evol. Comput. 16 (5) (2012) 601–614, http://dx.doi.org/10.1109/TEVC.2011.2161873.

[20] D.H. Wolpert, W.G. Macready, No free lunch theorem for optimization, IEEE Trans. Evol. Comput. 1 (1) (1997) 67–82, http://dx.doi.org/10.1109/4235.585893.

[21] C. Zhong, G. Li, Comprehensive learning Harris hawks-equilibrium optimization with terminal replacement mechanism for constrained optimization problems, Expert Syst. Appl. 192 (2022) 116432, http://dx.doi.org/10.1016/j.eswa.2021.116432.

[22] S. Mirjalili, A. Lewis, The whale optimization algorithm, Adv. Eng. Softw. 95 (2016) 51–67, http://dx.doi.org/10.1016/j.advengsoft.2016.01.008.

[23] A. Faramarzi, M. Heidarinejad, B. Stephens, S. Mirjalili, Equilibrium optimizer: a novel optimization algorithm, Knowl.-Based Syst. 191 (2020) 105190, http://dx.doi.org/10.1016/j.knosys.2019.105190.

[24] J.H. Holl, Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, Michigan, 1975.

[25] S. Kirkpatrick, C.D. Gelatt Jr., M.P. Vecchi, Optimization by simulated annealing, Science 220 (4598) (1983) 671–680, http://dx.doi.org/10.1126/science.220.4598.671.

[26] F. Glover, Future paths for integer programming and links to artificial intelligence, Comput. Oper. Res. 13 (5) (1986) 533–549, http://dx.doi.org/10.1016/0305-0548(86)90048-1.

[27] J. Kennedy, R.C. Eberhart, Particle swarm optimization, in: Proceedings of IEEE International Conference on Neural Networks, Perth, 1995, pp. 1942–1948, http://dx.doi.org/10.1109/ICNN.1995.488968.

[28] R. Storn, K. Price, Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces, J. Global Optim. 11 (1997) 341–359, http://dx.doi.org/10.1023/A:1008202821328.

[29] M. Dorigo, G. Di Caro, Ant colony optimization: a new meta-heuristic, in: Proceedings of the 1999 Congress on Evolutionary Computation-CEC99, Washington DC, 1999, pp. 1470–1477, http://dx.doi.org/10.1109/CEC.1999.782657.

[30] Z.W. Geem, J.H. Kim, G.V. Loganathan, A new heuristic optimization algorithm: harmony search, Simulation 76 (2) (2001) 60–68, http://dx.doi.org/10.1177/003754970107600201.

[31] K.M. Passino, Biomimicry of bacterial foraging for distributed optimization and control, IEEE Control Syst. Mag. 22 (3) (2002) 52–67, http://dx.doi.org/10.1109/MCS.2002.1004010.

[32] X.L. Li, A New Intelligent Optimization-Artificial Fish Swarm Algorithm, (Ph.D. thesis), Zhejiang University, China. (in Chinese), 2003.

[33] O.K. Erol, I. Eksin, A new optimization method: big bang-big crunch, Adv. Eng. Softw. 37 (2) (2006) 106–111, http://dx.doi.org/10.1016/j.advengsoft.2005.04.005.

[34] D. Karaboga, B. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, J. Global Optim. 39 (3) (2007) 459–471, http://dx.doi.org/10.1007/s10898-007-9149-x.

[35] E. Atashpax-Gargari, C. Lucas, Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition, in: 2007 IEEE Congress on Evolutionary Computation, CEC 2007, 2007, pp. 4661–4662, http://dx.doi.org/10.1109/CEC.2007.4425083.

[36] D. Simon, Biogeography-based optimization, IEEE Trans. Evol. Comput. 12 (6) (2008) 702–713, http://dx.doi.org/10.1109/TEVC.2008.919004.

[37] X.S. Yang, S. Deb, Cuckoo search via lévy flights, in: World Congress on Nature & Biologically Inspired Computing, NaBIC, Coimbatore, 2009, pp. 210–214, http://dx.doi.org/10.1109/NABIC.2009.5393690.

[38] E. Rashedi, H. Nezamabadi-pour, S. Saryazdi, GSA: a gravitational search algorithm, Inform. Sci. 179 (2009) 2232–2248, http://dx.doi.org/10.1016/j.ins.2009.03.004.

[39] A. Kaveh, S. Talatahari, A novel heuristic optimization method: charged system search, Acta Mech. 213 (2010) 267–289, http://dx.doi.org/10.1007/s00707-009-0270-4.

[40] A.Y.S. Lam, V.O.K. Li, Chemical-reaction-inspired metaheuristic for optimization, IEEE Trans. Evol. Comput. 14 (3) (2010) 381–399, http://dx.doi.org/10.1109/TEVC.2009.2033580.

[41] X.S. Yang, A.H. Gandomi, Bat algorithm: a novel approach for global engineering optimization, Eng. Comput. 29 (5) (2012) 464–483, http://dx.doi.org/10.1108/02644401211235834.

[42] Y.H. Shi, An optimization algorithm based on brainstorming process, Int. J. Swarm Intell. Res. 2 (4) (2011) 35–62, http://dx.doi.org/10.4018/jsir.2011100103.

[43] R.V. Rao, V.J. Savsani, D.P. Vakharia, Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems, Comput. Aided Des. 43 (2011) 303–315, http://dx.doi.org/10.1016/j.cad.2010.12.015.

[44] X.S. Yang, Flower pollination algorithm for global optimization, in: Unconventional Computation and Natural Computation, Springer, 2012, pp. 240–249, http://dx.doi.org/10.1007/978-3-642-32894-7_27.

[45] A.H. Gandomi, A.H. Alavi, Krill herd: A new bio-inspired optimization algorithm, Commun. Nonlinear Sci. Numer. Simul. 17 (12) (2012) 4831–4835, http://dx.doi.org/10.1016/j.cnsns.2012.05.010.

[46] A. Kaveh, M. Khayatazad, A new metaheuristic method: ray optimization, Comput. Struct. 112–113 (2012) 283–294, http://dx.doi.org/10.1016/j.compstruc.2012.09.003.

[47] H. Eskandar, A. Sadollah, A. Bahreininejad, M. Hamdi, Water cycle algorithm - A novel metaheuristic optimization method for solving constrained engineering optimization problems, Comput. Struct. 110–111 (2012) 151–166, http://dx.doi.org/10.1016/j.compstruc.2012.07.010.

[48] S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey wolf optimizer, Adv. Eng. Softw. 69 (2014) 46–61, http://dx.doi.org/10.1016/j.advengsoft.2013.12.007.

[49] M.Y. Cheng, D. Prayogo, Symbiotic organisms search: a new metaheuristic optimization algorithm, Comput. Struct. 139 (2014) 98–112, http://dx.doi.org/10.1016/j.compstruc.2014.03.007.
[50] A.H. Kashan, League championship algorithm (LCA): an algorithm for global optimization inspired by sport championships, Appl. Soft Comput. 16 (2014) 171–200, http://dx.doi.org/10.1016/j.asoc.2013.12.005.
[51] S. Mirjalili, The ant lion optimizer, Adv. Eng. Softw. 83 (2015) 80–98, http://dx.doi.org/10.1016/j.advengsoft.2015.01.010.
[52] S. Mirjalili, Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm, Knowl.-Based Syst. 89 (2015) 228–249, http://dx.doi.org/10.1016/j.knosys.2015.07.006.
[53] G.G. Wang, S. Deb, Z. Cui, Monarch butterfly optimization, Neural Comput. Appl. 31 (2019) 1995–2014, http://dx.doi.org/10.1007/s00521-015-1923-y.
[54] H. Salimi, Stochastic fractal search. A powerful metaheuristic algorithm, Knowl.-Based Syst. 75 (2015) 1–18, http://dx.doi.org/10.1016/j.knosys.2014.07.025.
[55] A. Kaveh, T. Bakhshpoori, Water evaporation optimization: a novel physically inspired optimization algorithm, Comput. Struct. 167 (2016) 69–85, http://dx.doi.org/10.1016/j.compstruc.2016.01.008.
[56] S. Mirjalili, SCA: A sine cosine algorithm for solving optimization problems, Knowl.-Based Syst. 96 (2016) 120–133, http://dx.doi.org/10.1016/j.knosys.2015.12.022.
[57] S. Saremi, S. Mirjalili, A. Lewis, Grasshopper optimization algorithm: theory and application, Adv. Eng. Softw. 105 (2017) 30–47, http://dx.doi.org/10.1016/j.advengsoft.2017.01.004.
[58] S. Mirjalili, A.H. Gandomi, S.Z. Mirjalili, S. Saremi, H. Faris, S.M. Mirjalili, Salp swarm algorithm: a bio-inspired optimizer for engineering design problems, Adv. Eng. Softw. 114 (2017) 163–191, http://dx.doi.org/10.1016/j.advengsoft.2017.07.002.
[59] A. Kaveh, A. Dadras, A novel meta-heuristic optimization algorithm: thermal exchange optimization, Adv. Eng. Softw. 110 (2017) 69–84, http://dx.doi.org/10.1016/j.advengsoft.2017.03.014.
[60] N.A. Kallioras, N.D. Lagaros, D.N. Avtzis, Pity beetle algorithm – a new metaheuristic inspired by the behavior of bark beetles, Adv. Eng. Softw. 121 (2018) 147–166, http://dx.doi.org/10.1016/j.advengsoft.2018.04.007.
[61] M. Jain, V. Singh, A. Rani, A novel nature-inspired algorithm for optimization: squirrel search algorithm, Swarm Evol. Comput. 44 (2019) 148–175, http://dx.doi.org/10.1016/j.swevo.2018.02.013.
[62] S. Arora, S. Singh, Butterfly optimization algorithm: a novel approach for global optimization, Soft Comput. 23 (2019) 715–734, http://dx.doi.org/10.1007/s00500-018-3102-4.
[63] A.A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, Harris hawks optimization: algorithm and applications, Future Gener. Comput. Syst. 97 (2019) 849–872, http://dx.doi.org/10.1016/j.future.2019.02.028.
[64] F.A. Hashim, E.H. Houssein, M.S. Mabrouk, W. Al-Atabany, S. Mirjalili, Henry gas solubility optimization: a novel physics-based algorithm, Future Gener. Comput. Syst. 101 (2019) 646–667, http://dx.doi.org/10.1016/j.future.2019.07.015.
[65] G. Dhiman, V. Kumar, Seagull optimization algorithm: theory and its applications for large-scale industrial engineering problems, Knowl.-Based Syst. 165 (2019) 169–196, http://dx.doi.org/10.1016/j.knosys.2018.11.024.
[66] R. Masadeh, B.A. Mahafzah, A. Sharieh, Sea lion optimization algorithm, Int. J. Adv. Comput. Sci. Appl. 10 (5) (2019) 388–395, http://dx.doi.org/10.14569/IJACSA.2019.0100548.
[67] M.H. Sulaiman, Z. Mustaffa, M.M. Saari, H. Daniyal, Barnacles mating optimizer: a new bio-inspired algorithm for solving engineering optimization problems, Eng. Appl. Artif. Intell. 87 (2020) 103330, http://dx.doi.org/10.1016/j.engappai.2019.103330.
[68] A. Kaveh, M. Khanzadi, M. Rastegar Moghaddam, Billiards-inspired optimization algorithm: a new meta-heuristic method, Structures 27 (2020) 1722–1739, http://dx.doi.org/10.1016/j.istruc.2020.07.058.
[69] V. Hayyolalam, A.A.P. Kazem, Black widow optimization algorithm: a novel meta-heuristic approach for solving engineering optimization problems, Eng. Appl. Artif. Intell. 87 (2020) 103249, http://dx.doi.org/10.1016/j.engappai.2019.103249.
[70] M. Khishe, M.R. Mosavi, Chimp optimization algorithm, Expert Syst. Appl. 149 (2020) 113338, http://dx.doi.org/10.1016/j.eswa.2020.113338.
[71] Q. Askari, M. Saeed, I. Younas, Heap-based optimizer inspired by corporate rank hierarchy for global optimization, Expert Syst. Appl. 161 (2020) 113702, http://dx.doi.org/10.1016/j.eswa.2020.113702.
[72] M. Jahangiri, M.A. Hadianfard, M.A. Najafgholipour, M. Jahangiri, M.R. Gerami, Interactive autodidactic school: a new metaheuristic optimization algorithm for solving mathematical and structural design optimization problems, Comput. Struct. 235 (2020) 106268, http://dx.doi.org/10.1016/j.compstruc.2020.106268.
[73] E.H. Houssein, M.R. Saad, F.A. Hashim, H. Shaban, M. Hassaballah, Lévy flight distribution: a new metaheuristic algorithm for solving engineering optimization problems, Eng. Appl. Artif. Intell. 94 (2020) 103731, http://dx.doi.org/10.1016/j.engappai.2020.103731.
[74] A. Faramarzi, M. Heidarinejad, S. Mirjalili, A.H. Gandomi, Marine predators algorithm: a nature-inspired metaheuristic, Expert Syst. Appl. 152 (2020) 113377, http://dx.doi.org/10.1016/j.eswa.2020.113377.
[75] H.R.E.H. Bouchekara, Most valuable player algorithm: a novel optimization algorithm inspired from sport, Oper. Res. 20 (2020) 139–195, http://dx.doi.org/10.1007/s12351-017-0320-y.
[76] S.Q. Salih, A.A. Alsewari, A new algorithm for normal and large-scale optimization problems: Nomadic people optimizer, Neural Comput. Appl. 32 (2020) 10359–10386, http://dx.doi.org/10.1007/s00521-019-04575-1.
[77] A. Kaveh, H. Akbari, S.M. Hosseini, Plasma generation optimization: a new physically-based metaheuristic algorithm for solving constrained optimization problems, Eng. Comput. 38 (4) (2020) 1554–1606, http://dx.doi.org/10.1108/EC-05-2020-0235.
[78] Q. Askari, I. Younas, M. Saeed, Political optimizer: a novel socio-inspired meta-heuristic for global optimization, Knowl.-Based Syst. 195 (2020) 105709, http://dx.doi.org/10.1016/j.knosys.2020.105709.
[79] S. Li, H. Chen, M. Wang, A.A. Heidari, S. Mirjalili, Slime mould algorithm: a new method for stochastic optimization, Future Gener. Comput. Syst. 111 (2020) 300–323, http://dx.doi.org/10.1016/j.future.2020.03.055.
[80] S. Kaur, L.K. Awasthi, A.L. Sangal, G. Dhiman, Tunicate swarm algorithm: a new bio-inspired based metaheuristic paradigm for global optimization, Eng. Appl. Artif. Intell. 90 (2020) 103541, http://dx.doi.org/10.1016/j.engappai.2020.103541.
[81] L. Abualigah, A. Diabat, S. Mirjalili, M.A. Elaziz, A.H. Gandomi, The arithmetic optimization algorithm, Comput. Methods Appl. Mech. Engrg. 376 (2021) 113609, http://dx.doi.org/10.1016/j.cma.2020.113609.
[82] F. Zitouni, S. Harous, A. Belkeram, L.E.B. Hammou, The Archerfish hunting optimizer: A novel metaheuristic algorithm for global optimization, Arab. J. Sci. Eng. (2021) http://dx.doi.org/10.1007/s13369-021-06208-z.
[83] O.K. Meng, O. Pauline, S.C. Kiong, A carnivorous plant algorithm for solving global optimization problems, Appl. Soft Comput. 98 (2021) 106833, http://dx.doi.org/10.1016/j.asoc.2020.106833.
[84] A. Mohammadi-Balani, M.D. Nayeri, A. Azar, M. Taghizadeh-Yazdi, Golden eagle optimizer: a nature-inspired metaheuristic algorithm, Comput. Ind. Eng. 152 (2021) 107050, http://dx.doi.org/10.1016/j.cie.2020.107050.
[85] D. Połap, M. Woźniak, Red fox optimization algorithm, Expert Syst. Appl. 166 (2021) 114107, http://dx.doi.org/10.1016/j.eswa.2020.114107.
[86] F. Zitouni, S. Harous, R. Maamri, The solar system algorithm: a novel metaheuristic method for global optimization, IEEE Access 9 (2021) 4542–4565, http://dx.doi.org/10.1109/ACCESS.2020.3047912.
[87] H. Emami, Stock exchange trading optimization algorithm: a human-inspired method for global optimization, J. Supercomput. (2021) http://dx.doi.org/10.1007/s11227-021-03943-w.
[88] C. Zhong, M. Wang, C. Dang, W. Ke, Structural reliability assessment by salp swarm algorithm-based FORM, Qual. Reliab. Eng. Int. 36 (2020) 1224–1244, http://dx.doi.org/10.1002/qre.2626.
[89] C. Zhong, M. Wang, C. Dang, W. Ke, S. Guo, First-order reliability method based on Harris hawks optimization for high-dimensional reliability analysis, Struct. Multidiscip. Optim. 62 (2020) 1951–1968, http://dx.doi.org/10.1007/s00158-020-02587-3.
[90] G. Li, H. Lu, X. Liu, A hybrid genetic algorithm and optimality criteria method for optimum design of RC tall buildings under multi-load cases, Struct. Des. Tall Special Build. 19 (2010) 656–678, http://dx.doi.org/10.1002/tal.505.
[91] M. Jafari, H. Moussavian, M.H.B. Chaleshtari, Optimum design of perforated orthotropic and laminated composite plates under in-plane loading by genetic algorithm, Struct. Multidiscip. Optim. 57 (2018) 341–357, http://dx.doi.org/10.1007/s00158-017-1758-5.
[92] M. Bilal Pant, H. Zaheer, L. Garcia-Hernandez, A. Abraham, Differential evolution: a review of more than two decades of research, Eng. Appl. Artif. Intell. 90 (2020) 103479, http://dx.doi.org/10.1016/j.engappai.2020.103479.
[93] F.E. Fernandes Jr., G.G. Yen, Pruning of generative adversarial neural networks for medical imaging diagnostics with evolution strategy, Inform. Sci. 558 (2021) 91–102, http://dx.doi.org/10.1016/j.ins.2020.12.086.
[94] Q. Lu, S. Zhou, F. Tao, J. Luo, Z. Wang, Enhancing gene expression programming based on space partition and jump for symbolic regression, Inform. Sci. 547 (2021) 553–567, http://dx.doi.org/10.1016/j.ins.2020.08.061.
[95] S. Bansal, N. Baliyan, Bi-MARS: a bi-clustering based memetic algorithm for recommender systems, Appl. Soft Comput. 97 (2020) 106785, http://dx.doi.org/10.1016/j.asoc.2020.106785.
[96] B. Suman, P. Kumar, A survey of simulated annealing as a tool for single and mutliobjective optimization, J. Oper. Res. Hist. 57 (2006) 1143–1160, http://dx.doi.org/10.1057/palgrave.jors.2602068.
[97] F. Javidrad, M. Nazari, A new hybrid particle swarm and simulated annealing stochastic optimization method, Appl. Soft Comput. 60 (2017) 634–654, http://dx.doi.org/10.1016/j.asoc.2017.07.023.

[98] N. Neggaz, A.A. Ewees, M.A. Elaziz, M. Mafarja, Boosting salp swarm algorithm by sine cosine algorithm and disrupt operator for feature selection, Experts Syst. Appl. 145 (2020) 113103, http://dx.doi.org/10.1016/j.eswa.2019.113103.
[99] F. Zou, D. Chen, Q. Xu, A survey of teaching-learning-based optimization, Neurocomputing 335 (2019) 366–383, http://dx.doi.org/10.1016/j.neucom.2018.06.076.
[100] W.F. Perrin, B. Würsig, J.G.M. Thewissen, Encyclopedia of Marine Mammals, second ed., Elsevier, 2009, pp. 108–112, http://dx.doi.org/10.1016/B978-0-12-373553-9.X0001-6.
[101] H. Hill, S. Dietrich, D. Yeater, M. McKinnon, M. Miller, S. Aibel, A. Dove, Developing a catalog of socio-sexual behaviors of beluga whales (Delphinapterus leucas) in the care of humans, Animal Behav. Cogn. 2 (2) (2015) 105–123, http://dx.doi.org/10.12966/abc.05.01.2015.
[102] C.R. Smith, A.G. Golver, T. Treude, N.D. Higgs, D.J. Amon, Whale-fall ecosystems: recent insights into ecology, paleoecology, and evolution, Ann. Rev. Mar. Sci. 7 (2015) 571–596, http://dx.doi.org/10.1146/annurev-marine-010213-135144.
[103] R.N. Mantegna, Fast, accurate algorithm for numerical simulation of Lévy stable stochastic processes, Phys. Rev. E 49 (5) (1994) 4677–4683, http://dx.doi.org/10.1103/PhysRevE.49.4677.
[104] P.N. Suganthan, N. Hansen, J. Liang, K. Deb, Y. Chen, A. Auger, S. Tiwari, Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization, Technical Report, Nanyang Technological University and KanGAL Report Number 2005005, Singapore, 2005, pp. 1–50, https://www.ntu.edu.sg/home/EPNSugan.
[105] J. Liang, P. Suganthan, K. Deb, Novel composition test functions for numerical global optimization, in: Proceedings 2005 IEEE Swarm Intelligence Symposium, 2005, SIS 2005, Pasadena, 2005, pp. 68–75, http://dx.doi.org/10.1109/SIS.2005.1501604.
[106] J. Derrac, S. García, D. Molina, F. Herrera, 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) (2011) 3–18, http://dx.doi.org/10.1016/j.swevo.2011.02.002.