Knowledge-Based Systems 235 (2022) 107645

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

A reinforcement learning brain storm optimization algorithm (BSO) with learning mechanism

Fuqing Zhao [a,*], Xiaotong Hu [a], Ling Wang [b], Jinlong Zhao [a], Jianxin Tang [a], Jonrinaldi [c]

[a] School of Computer and Communication Technology, Lanzhou University of Technology, Lanzhou 730050, China
[b] Department of Automation, Tsinghua University, Beijing, 10084, China
[c] Department of Industrial Engineering, Universitas Andalas, Padang, 25163, Indonesia

ARTICLE INFO

Article history:
Received 9 February 2021
Received in revised form 1 July 2021
Accepted 30 September 2021
Available online 3 November 2021

Keywords:
Brain storm optimization
Q-learning
Mutation strategies
Self-learning mechanism

ABSTRACT

Brain storm optimization algorithm (BSO), which is inspired by brain storm process of human, has been adopted as an efficient optimizer for various complex problems. A reinforcement learning brain storm optimization algorithm (RLBSO) to improve the performance of BSO is proposed in this paper. Four mutation strategies are designed to enhance the search capability of the algorithm in different stages. Elites are adopted as the guidance to ensure the quality of the population. The global best is utilized to guide the search direction of individuals. The cluster centers are employed to improve the exploitation ability of the algorithm. The historical individuals are utilized to the increase the diversity of the population. The Q-learning mechanism is introduced to guide the selection of strategies according to the historical information fed back by the corresponding strategies. A self-learning mechanism, which is based on the evolutionary state of the population and the experience of previous successful individuals, is utilized to determine the update method of individual. The RLBSO algorithm is tested on the CEC 2017 benchmark test suite and a practical engineering problem. The results show that RLBSO has better performance than the other state of the arts algorithms.

1. Introduction

1.1. Motivation

Complex engineering problems [1,2], which exist widely in the real world, have the properties of high dimension, multi-modal and large scale. Initially, precise solution methods such as branch and bound method [3] and linear programming [4] are used to solve the complex engineering problems. The mathematical method is an effective tool to solve small-scale problems. However, the mathematical methods tend to fall into local optimum when facing multi-modal and composition problems. In addition, a mathematical model of the problem is to be established at first when adopting mathematical methods as the optimizer. However, various complex engineering problems in application are difficult to establish a mathematical model. Therefore, a meta-heuristic algorithm [5] which is independent of the characteristics of the problem to solve np-hard problems is utilized by experts and scholars. Meta-heuristic algorithms are divided into three categories, including algorithms based on natural phenomena, physical phenomena and emerging algorithms. The first is inspired by natural phenomena, such as Particle Swarm Optimization algorithm (PSO) [6], Ant Colony Optimization algorithm (ACO) [7]. The second is inspired by physical phenomena, such as Gravitational Search Algorithm (GSA) [8,9], Simulated Annealing (SA) [10], Water Wave Optimization (WWO) [11]. The third is some other emerging intelligent algorithms: Differential Evolution algorithm (DE) [12,13], and Estimation Distribution Algorithm (EDA) [14]. The process of metaheuristic algorithm to solve a problem is iteration, that is, the individuals are constantly looking for the optimal value in the solution space. However, aimless searching of individuals in the solution space are not able to quickly find the global optimal solution or satisfactory suboptimal solution, but also increase the calculation cost. Therefore, theoretical guidance is necessary.

Brain storm optimization algorithm, which is proposed by Shi [15,16] in 2011, is a swarm intelligent algorithm. As a kind of meta-heuristic algorithm, an unsupervised learning mechanism—clustering, is added in BSO. Clustering operation, which is the core of BSO algorithm, is the main difference between BSO and other algorithms. The population is divided into different sub populations, with each sub population as a neighborhood. The local optimal solution is generated in the neighborhood, which increases the speed and probability of finding the optimal solution and improves the solving efficiency of the algorithm. In addition,

* Corresponding author.
E-mail addresses: Fzhao2000@hotmail.com (F. Zhao), huxt826@163.com (X. Hu), wangling@tsinghua.edu.cn (L. Wang), 1409776617@qq.com (J. Zhao), 582971672@qq.com (J. Tang), jonrinaldi@eng.unand.ac.id (Jonrinaldi).

https://doi.org/10.1016/j.knosys.2021.107645
0950-7051/

humans are the smartest animals in the world, so it is thought that BSO algorithm inspired by human brainstorming behavior is smarter than algorithms inspired by other animals.

The generation of new solutions is a significant operation in BSO algorithm. The performance of the BSO is largely determined by the design and selection of the mutation strategy. To solve the variable non-separable problem and increase the rotation invariance of the algorithm, a variety of mutation strategies need to be designed to suit different individuals. In addition, the updating strategies of BSO are usually to exchange local neighborhood information between individuals blindly or randomly, which inevitably leads to the deterioration of the population evolution during searching. The updating strategies with learning ability need to be proposed to ensure the continuous learning of the algorithm in the search process to ensure the evolutionary ability of the population.

Reinforcement learning is used to describe and solve the problem that agents maximize returns or achieve specific goals in the process of interacting with the environment through learn strategies. Reinforcement learning takes learning as a process of trial evaluation, in which an agent chooses an action for the environment, and the state changes after the environment accepts the action. At the same time, a reinforcement signal is generated to feed back to the agent, and the agent chooses the next action according to the reinforcement signal and the current state of the environment. The principle of selection is to increase the probability of positive reinforcement. Reinforcement learning ideas have been applied to a large number of researches and achieved good results.

A new strategy gradient method [17] is proposed for reinforcement learning. The method alternates between sampled data by interacting with the environment, and uses stochastic gradient ascent to optimize the "proxy" objective function. The results show that the proposed "Proximal Policy Optimization" is superior to other online strategy gradient methods, and achieves a good balance among sample complexity, simplicity and persistence. Liu et al. proposed a weighted relative frequency of the maximal reward (WRFMR) algorithm [18], which utilizes a weighted parameter and action probability to balance exploration and exploitation and accelerate convergence to the optimal joint action. Each agent needs to share status and immediate rewards without observing the actions of other agents. The simulation results show that WRFMR algorithm is better than other algorithms in learning speed and success rate. An adaptive actor-criticism approach that considers the action strategies of other agents is proposed [19]. The approach successfully learns strategies that require complex multi-agent coordination. In addition, a training scheme is introduced that uses a set of policies for each agent to generate robust multi-agent policies. The proposed approach has advantages over existing approaches in collaborative and competitive scenarios where proxy populations are able to discover a variety of physical and information coordination strategies.

Three hybrid algorithms using reinforcement learning and meta-heuristic methods to solve global optimization problems is introduced [20]. The search agent tries to find a global optimum to avoid falling into the trap of local optimum. Comparing with the classical meta-heuristic method, the proposed algorithm has high success rate and balanced performance in the exploration and development stage. A method based on RL rules [21] is proposed to guide the use of evolutionary algorithm (EA) in constraint optimization. Firstly, the RL proximal strategy optimization agent is trained to master the rules/constraints of matching part of the problem. The RL injection experience is used to guide various evolutionary/stochastic algorithms, such as GA, SA, PSO, DE and natural evolution strategies. The results show that the performance of RL-guided EA is better than that of independent evolutionary algorithm.

The idea of information feedback is an extremely important part of reinforcement learning and has been widely used. A new mutation strategy "DE/current-to-pbest" with optional external archiving and adaptive update control parameters is implemented in [22]. "DE/current-to-pbest" is a generalization of the classic "DE/current-to-best", while the optional archiving operation uses historical data to provide information on the direction of progress. Both operations diversify the population and improve the convergence performance. JADE with an external archive shows promising results in terms of relatively high dimensions. A new variant of JADE is proposed [23], which incorporates chaotic local search (CLS) mechanism into JADE to alleviate the premature convergence problem of DE. Using the ergodicity and non-repeatability of chaos, it is possible to diversify the population and thus have the opportunity to explore the vast search space. Because of its inherent local development capability, its embedded CLS take advantage of small areas to refine the solutions obtained by JADE. Multiple chaotic mappings are individually, randomly, parallel, and memory-selectively incorporated into the CLS. The results show that the proposed algorithm has better performance than JADE and some other advanced optimization algorithms. A hierarchical knowledge guided backtracking search algorithm (HKBSA) with self-learning strategy is proposed [24]. A distribution vector of disadvantaged individuals is constructed by sampling method, and a feedback individual generation mechanism is proposed by combining self-learning strategy and elite learning strategy. Comparing with the advanced BSA algorithms, HKBSA algorithm is superior to other algorithms in terms of convergence speed, solution precision and stability.

Reinforcement learning is used to guide the improvement of the performance of BSO in view of the issues of large randomness and the lack of theoretical guidance, and the successful application of RL in existing swarm intelligence algorithms.

1.2. Literature review

In recent years, BSO has been used to solve continuous optimization problems and practical engineering problems [25–29].

The clustering method is improved to decrease the time complexity and cost of calculation. The clustering method in basic BSO is replaced by a random grouping strategy in the proposed algorithm (RGBSO) [30]. The new clustering operation reduces the computing time of the algorithm. Besides, a new dynamic step size control strategy is used to balance the exploration and exploitation capability of the algorithm at different stages of the iteration. The performance of RGBSO is better than that of BSO algorithm, but it does not work well in solving practical application problems. An improved BSO algorithm (MBSO) [31] was proposed by Zhan et al. A simple grouping method (SGM) is used to reduce the computational cost of the algorithm. An idealized difference strategy (IDS), which effectively avoid the algorithm falling into local optimum, is proposed to replace Gaussian random search strategy. A method of clustering in the target space (BSOOS) is proposed [32]. Individuals are divided into elite individuals and ordinary individuals in the target space in the BSOOS. Each individual in the target space is one-dimensional, so the calculation time of this method depends only on the population size. This algorithm can effectively solve high dimension problems.

Some other algorithms have been incorporated into the BSO to further improve the search capability of the BSO algorithm. An BSO algorithm is proposed based on a graph theory [33]. The population is transformed into an undirected weight graph. The length of the Hamiltonian ring is used to judge whether the algorithm is in a poor state. When the algorithm falls into local optimum, new individuals are generated to improve the old ones. A BSO algorithm combined with distribution estimation (EDBSO) [34] is proposed by Luo et al. The EDBSO divides the discussion process into intra-cluster and inter-cluster discussions. The joint probability distribution, mean and variance of the two clusters are calculated for each combined cluster in the inter-cluster discussion process. Distribution estimation is used to generate new individuals. The EDBSO effectively contributes to helping the algorithm find the global optimal. The convergence of Gaussian estimation algorithm is not replaced by other estimation methods. The chaotic local search (CLS) [35], which is combined with BSO algorithm, is utilized to enhance the local search ability of BSO. When the algorithm goes to a standstill, the dynamic mechanism of CLS disperses individuals and preserves the diversity of the population. The robustness of CLS in solving practical engineering problems is not studied.

Other aspects of the BSO are designed to improve the efficiency of the algorithm. A control method for evaluating the diversity of BSO population [36] is proposed to improve the performance of BSO. A distance-based diversity metric and fitness-based diversity metric are used to implement the adaptive algorithm parameters. The performance of the proposed method is verified by CEC2017 test suite and neuron model training task. Experimental results show that the method is effective. An adaptive step structure (ASBSO) and a successful memory selection strategy are proposed [37] to increase the efficiency and robustness. The method, which is an adaptive step size based on memory selection, uses multiple step sizes to modify the generation process of new solutions, so as to provide flexible search according to the corresponding problem and convergence period. A new memory mechanism is used to evaluate and store the improvement of the solution to determine the possibility of step size selection. The results show that ASBSO has significantly improved solution quality, scalability and robustness. The performance of the BSO is improved by introducing a global optimal version combined with updates of each variable and fitness-based grouping [38]. In addition, the proposed algorithm also includes a reinitialization scheme triggered by the current state of the population. The introduced global best BSO (GBSO) is compared with other BSO variants and global best versions of other meta-heuristics. The results show that GBSO is superior to the previous BSO variants in various classical functions and different problem scales. The relationship between individuals in BSO is constructed by population interaction network (PIN) [39] to analyze the performance of the algorithm theoretically from the perspective of population evolution. Experiments of different dimensions, parameters, combinatorial parameter Settings and related algorithms are carried out. The experimental results show that the average degree frequency of BSO satisfies the power law distribution in the low-dimensional function, which indicates that the algorithm has the best performance in the three dimensions.

A definition of population diversity is proposed to measure the search capability of the algorithm [40]. Two strategies are used to reinitialize the population. The first is to reinitialize the half solutions after several generations of population renewal. The second is to gradually reduce the amount of reinitialized resolution during the search process. The performance of BSO is improved in this work. The relationship between the diversity of the population and the performance of BSO is the future work. A re-initialization mechanism [41] is proposed to balance the search capability of the algorithm in different stages of exploration and exploitation. Whether or not a reinitialization operation is performed depends on the current state of the population. Besides, the updating method of step size is modified, considering the size of search space. The improved BSO has good performance compared with the two BSO variants. However, the algorithm is not compared to other population-based algorithms. An algorithm based on multi-population framework (MBSO) [42] is proposed. The individuals are grouped into elite individuals and ordinary individuals in the target space. The updating formula independent of the maximum number of iterations is used when updating the population. The improved BSO algorithm can effectively solve the multi-modal optimization problems. Four search strategies are designed to enhance the exploration and exploitation capability of the algorithm at different stages [43]. An adaptive parameter is set to dynamically select the strategy. A random grouping strategy (RMBSO) [44] is used to represent k-means clustering method. Three methods are used to update the population. Besides, the global and local optima are used to guide the individuals. A micro-relaxed selection mechanism is used to replace the greedy selection to improve the diversity of the population. RMBSO is superior to the most advanced brain storm optimization algorithm and several improved intelligent algorithms.

The learning mechanism, which makes full use of historical experience, is added on the BSO to guide the evolution of the population. An adaptive learning strategy brain storm optimization, which adaptively imitates, explores and learns to generate new individuals, is proposed [45]. The performance of the proposed algorithm is better than that of BSO. However, the proposed BSO is not compared with other learning algorithms. A collaborative learning strategy (BSO CLS) [46] is proposed to iteratively update the candidate solutions. The other solutions and the weights obtained from the fitness values of the other solutions are combined linearly. This algorithm performs better than some other BSO variants on large-scale problems. The active learning strategy, which is based on the knowledge of active learning in machine learning, is used to improve the performance of BSO algorithms [47]. Moreover, the computational cost of the algorithm is reduced by dynamically varying the clustering period. An orthogonal learning framework (OLBSO) [48] is designed. An exploration and exploitation engine are designed to improve the learning mechanism of the BSO in the OLBSO. Different update vectors are set by orthogonal decision mechanism. The modified algorithm can effectively optimize complex functions. A learning automata mechanism, which adaptively selects the optimal parameters and search strategy for each individual, is proposed [49]. Three auxiliary transfer vectors are designed to enhance the local search and global search capability of the algorithm in different stages.

1.3. Contribution

A reinforcement learning brain storm optimization algorithm is proposed to enhance the ability of BSO. Four search strategies and crossover operation are designed in the RLBSO. The combination of the feedback of the successful experience and the evolutionary state of the population is used to guide the individual to choose the appropriate strategy. Specifically, the RLBSO has the following three advantages.

- Four mutation strategies, which makes full use of the global optimum, elitists, and historical individuals, are designed to increase the possibility of the algorithm for finding a superior solution.
- The Q-learning mechanism and the feedback information of successful history are utilized to guide the algorithm to choose the appropriate strategy.
- A self-learning strategy selection mechanism is designed to provide guidance for the timing of strategy selection according to the evolutionary state of the population.

The structure of this paper is organized as follows. The basic BSO algorithm is introduced in Section 2. The proposed RLBSO algorithm is illustrated in Section 3. The test and the results of the RLBSO is demonstrated in Section 4. The RLBSO for real-world engineering problem is tested in Section 5. Section 6 is the conclusion.

2. The brain storm optimization algorithm

The brain storm optimization algorithm is a promising swarm intelligence algorithm, including three main operations: clustering, new solution generation and selection. Each individual represents an idea produced by the process of brain storm in the BSO. Firstly, N solutions are generated randomly between [−100,100] in the search space.

$$X_{i,j}^{t} = L_j + rand * (R_j - L_j) \tag{1}$$

where i is the index of the current solution, and j is the dimension of each solution. R_j and L_j are the up and low boundary, respectively. *rand* is a random number in [0,1]. After initialization, the fitness values $f(X_{i,j}^{t})$ of the N individuals are calculated in the objective space. The main four operations of the BSO are as follows.

Step1: the k-means algorithm is used to divide the N individuals into M clusters. The best individual in each cluster is regarded as the center.

$$\text{find } X_{c,j}^{k} \in K \text{suchthat } \forall X_{i,j}^{k} \in K_{i,j}^{k}, f(X_{c,j}^{k}) < f(X_{i,j}^{k}) \tag{2}$$

where $X_{c,j}^{k}$ is the individual with the minimum fitness value, and $X_{i,j}^{k}$ is the random solution in the cluster K.

Step2: a cluster center is selected by a probability of 0.2, and then it is replaced by a random solution.

Step3: there are two stages in generating the new solutions. $X_{selected,j}^{t}$ is constructed by two methods in the first stage, determined by the probability $P = 0.8$.

One individual is used to construct the $X_{selected,j}^{t}$ in the first method. The probability P_{global} is used to determine whether to select a center or a random individual in a cluster.

$$X_{selected,j}^{t} = \begin{cases} X_{r,j}^{t}, r1 < P_{global} \\ X_{c,j}^{t}, otherwise \end{cases} \tag{3}$$

where $X_{r,j}^{t}$ and $X_{c,j}^{t}$ are a random solution and the center in a cluster, respectively. the probability of selection is $P_{global} = 0.4$.

Two individuals are used to construct the $X_{selected,j}^{t}$ in the second method. The probability P_{local} is used to determine whether to select two centers or two random individuals in two clusters.

$$X_{selected,j}^{t} = \begin{cases} \omega 1 * X_{r1,j}^{t} + (1-\omega 1) * X_{r2,j}^{t}, \text{if } r2 < P_{local} \\ \omega 1 * X_{c1,j}^{t} + (1-\omega 1) * X_{c2,j}^{t}, otherwise \end{cases} \tag{4}$$

where $X_{r1,j}^{t}$ and $X_{r2,j}^{t}$ are two random solutions in two clusters. $X_{c1,j}^{t}$ and $X_{c2,j}^{t}$ are the centers of the two clusters. The probability of selection is $P_{local} = 0.5$.

The combination of Gauss distribution and ξ is added on the $X_{selected,j}^{t}$ to generate the $X_{new,j}^{t}$ in the second stage. The is a random step size. The details are as follows.

$$X_{new,j}^{t} = X_{selected,j}^{t} + \xi * G(\mu, \sigma) \tag{5}$$

$$\xi = \log sig\left(\frac{0.5 * T - t}{s}\right) * rand\,(0, 1) \tag{6}$$

where the logsig() is a transfer function, and T is the number of maximum iteration, and t is the number of current iteration, and s is the slope.

Step4: the fitness value of the current solution is compared with that of the historical solution.

The solution, which has the better fitness value, is chosen to enter the next generation.

$$X_{i}^{t+1} = \begin{cases} X_{new,j}^{t}, & \text{if } f\left(X_{new,j}^{t}\right) > f\left(X_{i,j}^{t}\right) \\ X_{i,j}^{t}, & \text{otherwise} \end{cases} \tag{7}$$

3. The reinforcement learning brain storm optimization algorithm (RLBSO)

The reinforcement learning brain storm optimization is proposed in this section. The pseudocode is shown in Algorithm 1. The flow chart is shown in Fig. 1.

3.1. Four mutation strategies

Four individual update strategies are proposed to enhance the adaptability of the algorithm in different iteration stages. The four strategies have certain advantages in the exploration and exploitation stages of the algorithm. The control factor is $\lambda = e^{1-T/(T-t+1)}$. The four mutation strategies are shown in Fig. 2, with green dots representing new individuals. Multiple strategies are detailed below.

The pink dots represent elite individuals in Fig. 2(a). Elites are utilized to guide the search direction of individuals to ensure the quality of the population in the first strategy. A differential step size is constructed to make the population cover the undeveloped area. The control factor λ increases exponentially to make sure that the algorithm carries out a comprehensive search in the early stage and performs an accurate local search in the later stage.

$$X_{selected,j}^{t} = X_{best,j}^{t} + \lambda (X_{i1,j}^{t} - X_{i2,j}^{t}) \tag{8}$$

where, the elites are the top 10 individuals, and $X_{i1,j}^{t}$ and $X_{i2,j}^{t}$ are two different individuals from different clusters. The first strategy balances the global search and local search abilities of the RLBSO while maintaining the evolutionary ability of the population.

The global best individual is used to guide the current individual to search in the second strategy. The global best is represented by red dot in Fig. 2(b).

Algorithm 1 The RLBSO algorithm.

```
Input: ObjFun, N, T, M, R_j, L_j
Output:globalminimum, globalminimizer
Randomly initialize N solutions in population P.
For i form 1 to N do
   Calculate the fitness of the solutions.
End for
While t < T
   Divide the N individuals into M clusters.
   Select the best individual in cluster j as the center.
   If t ≤ 1
      Generate new individuals using four strategies randomly.
      Calculate the Q-learning table and get the represent individual M_x .
   else
      Update the population according to Algorithm 2.
      Calculate the Q-learning table and get the represent individual M_x .
   End if
   fitnessP_best = min(fitness)
   globalminimum = fitnessP_best
End while
```

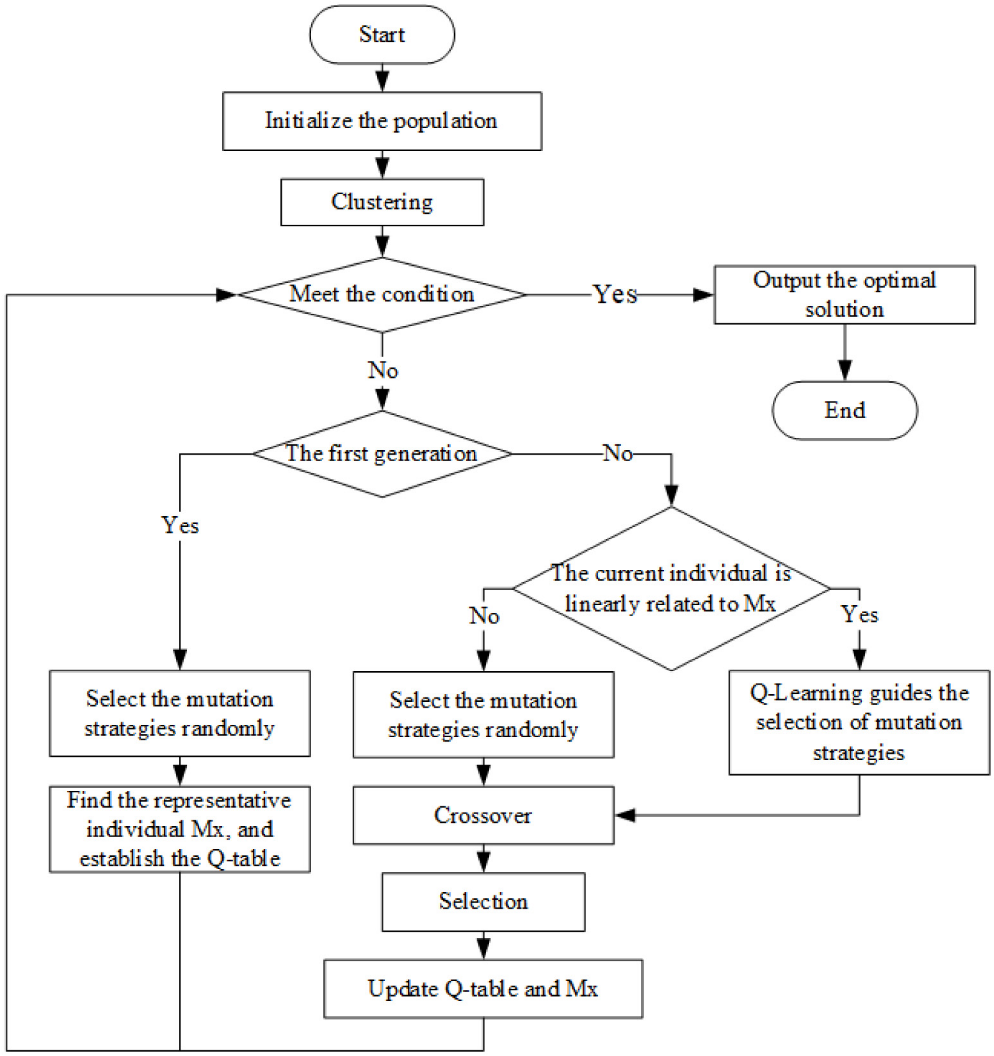


Fig. 1. The flow chart of RLBSO.

The differences between the individuals become smaller and smaller with the iteration of the population. The two differential items are used as the search step, and the search range of the population is gradually reduced. The population transforms from exploration to exploitation.

$$X^t_{selected,j} = X^t_{i,j} + \lambda(X^t_{gbest,j} - X^t_{i,j}) + \lambda(X^t_{p1,j} - X^t_{p2,j}) \tag{9}$$

where, $X^t_{i,j}$ is the current individual. $X^t_{gbest,j}$ is the global best. $X^t_{p1,j}$ and $X^t_{p2,j}$ are two different individuals from a cluster.

The cluster centers are utilized to guide individuals to search in a small area, which increases the local search capability of the algorithm in the third strategy. It is shown in Fig. 2(c).

$$X^t_{selected,j} = X^t_{p\_center,j} + \lambda(X^t_{p1,j} - X^t_{p2,j}) \tag{10}$$

where, $X^t_{p\_center,j}$ is the center of a cluster, $X^t_{p1,j}$ and $X^t_{p2,j}$ are two different individuals in cluster p.

The loss of population diversity is a serious phenomenon in the iteration process of swarm algorithms. The diversity of the population has an important effect on the convergence speed of the solution.

Individuals from historical population are used to generate new individuals in the fourth strategy, which enriches the individuals in the population. The three gray dots represent historical individuals in Fig. 2(d). The strategy effectively avoids the RLBSO falling into a local optimum during the solution process.

$$X^t_{selected,j} = X^t_{i,j} + F * (X^{t-1}_{i1,j} - X^t_{i,j}) + F * (X^{t-1}_{i2,j} - X^{t-1}_{i3,j}) \tag{11}$$

where, $X^t_{i,j}$ is the current individual. $X^{t-1}_{i1,j}$, $X^{t-1}_{i2,j}$, and $X^{t-1}_{i3,j}$ are individuals from historical population. F is a standard normally distributed random number.

3.2. Q-learning based selection guidance

It is an effective method to improve the ability of the algorithm to find superior solutions by adopting the corresponding mutation strategy at different evolutionary stages of the algorithm iteration. The mutation strategies are proposed to adapt to different iteration stages. The timing of these four strategies is crucial.

A Q-learning mechanism is used as the method of selecting strategies to overcome the shortcoming of random selection. The Q-learning [50,51], which is related to machine learning algorithms, is mainly composed of learning agent, an environment, states, actions, and rewards. The Q-learning is widely used in game theory [52]. The details of the multi-strategy mutation selection mechanism based on Q-learning are as follows.

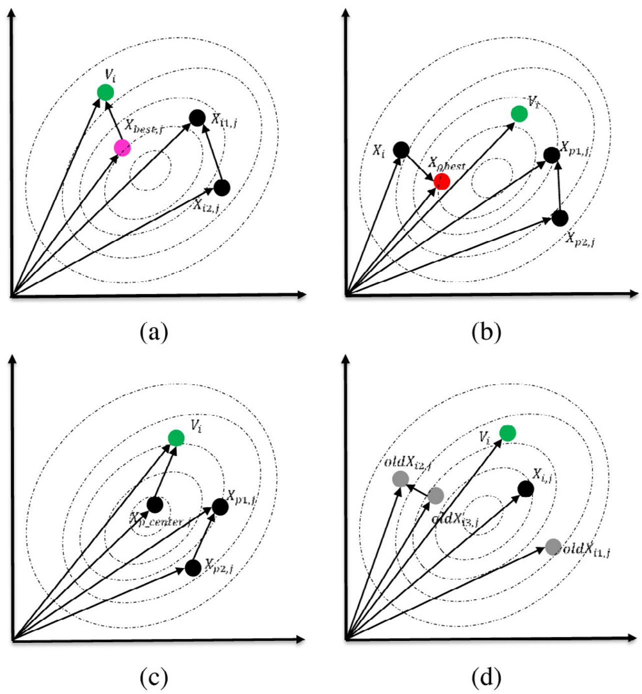


Fig. 2. The mutation strategies.

Algorithm 2 Updating method in RLBSO

Input: M_x, N, four strategies
Output: the current population
For i **from** 1 **to** N
  Calculate the correlation coefficient R.
  If $R = 1$
    According to the Q-learning table, update individuals using four strategies.
  else
    Select the four strategies randomly to update the individuals.
  End if
  Store the search experiences of current population.
  Crossover operation.
End for

First, initializing the Q-Table matrix. After population initialization, four mutation strategies are used to update the population. After the completion of the first iteration, the individuals in the current population who adopt the above four mutation strategies are counted. The four numbers of individuals who choose the four strategies are calculated. The number of successful individuals in each of the four mutation strategies is counted separately. The successful individuals are the individuals whose fitness value decreased after using each mutation strategy.

$$Q - table = \begin{bmatrix} mut^1 & score^1 \\ mut^2 & score^2 \\ \cdots & \cdots \\ mut^i & score^i \end{bmatrix} \tag{12}$$

where, mut^i is the ith mutation strategy. $score^i$ is the rewards score for the ith mutation strategy, $i = \{1, 2, 3, 4\}$.

A mutation strategy is selected according to the Q-table matrix to update the current individual from the second generation. The $score^i$ is updated after this iteration. $score^i$ is updated as follows.

$$score(t+1)^i = score(t)^i + \varepsilon \tag{13}$$

$$\varepsilon = \frac{\sum_{n=1}^{mut_n^{i'}} |1|}{\sum_{n=1}^{mut_n^i} |1|} \tag{14}$$

where, mut_n^i is the number of individuals using strategy mut^i. $mut_n^{i'}$ is the number of successful individuals after using mutation strategy mut^i.

3.3. Self-learning strategy selection mechanism

The offspring individuals are created by a series of operations performed by previous generations in the evolution of population. Therefore, individuals in the offspring population are more closely related to the individuals of the preceding generation. The successful individuals in the previous generation are chosen firstly. The new individual is instructed by the operation of the successful individual that is linearly related to itself. The correlation coefficient is adopted to measure the correlation between individuals in RLBSO.

Correlation analysis [53,54] is the analysis of two or more variables to obtain correlation coefficients for measuring the degree of correlation between two vectors. The correlation coefficient is calculated as follows.

Supposing that X and Y are two random variables, and $E(X)$ and $E(Y)$ are the expectation of X and Y, respectively. The standard deviations of X and Y are δ_X and δ_Y. They are calculated as follows.

$$\delta_X = \sqrt{D(X)} = \sqrt{E(X^2) - E^2(X)} \tag{15}$$

$$\delta_Y = \sqrt{D(Y)} = \sqrt{E(Y^2) - E^2(Y)} \tag{16}$$

The $Cov(X, Y)$ is the covariance of X and Y, which is calculated in Eq. (17).

$$\begin{aligned} Cov(X, Y) &= E((X - EX) - (Y - EY)) \\ &= E(XY) - E(X)E(Y) \end{aligned} \tag{17}$$

The correlation coefficient $\rho_{X,Y}$ is calculated as follows.

$$\begin{aligned} \rho_{X,Y} &= \frac{Cov(X, Y)}{\sigma_X \sigma_Y} \\ &= \frac{E(XY) - E(X)E(Y)}{\sqrt{E(X^2) - E^2(X)}\sqrt{E(Y^2) - E^2(Y)}} \end{aligned} \tag{18}$$

The steps of the correlation selection mechanism are as follows.

Step 1: Use four strategies randomly to update the population in the first generation.

Step 2: After the completion of the iteration, the individuals who successfully mutate using the above four mutation strategies in the current population are counted. The individual with the smallest fitness value is selected as the representative individual of this strategy from the successful individuals of each strategy. Representative individuals of the four mutation strategies are represented as M_{x1}, M_{x2}, M_{x3}, M_{x4}, respectively. The optimal individual M_x is selected from the four representative individuals as the learning object.

Step 3: Each individual x_i to be updated in the population is judged for correlation with M_x from the second generation. When there is a linear correlation between x_i and M_x, the reward and punishment scores in Q-table are used to guide the selection of the corresponding mutation strategy. If there is no linear correlation between x_i and M_x, the individual randomly selects the strategy for updating.

Step 4: Repeat steps 2–3 until the termination condition is met.

3.4. The crossover operation

A disturbance is added to the cluster center with a probability of 0.2 in the original BSO. A crossover operation is used to replace the disturbance operation in the RLBSO. The crossover operation works on some dimensions of the solution instead of changing the whole solution. The rotation invariance is enhanced, which improves the ability of finding the global optimal solution. Besides, the correlation between individuals becomes stronger and stronger in the process of algorithm iteration. It is not enough to simply re-initialize the cluster center to assist the algorithm in jumping out of the local optimal solution. The crossover operation works on some dimensions of the individual, which further increases the chance of being optimal on all dimensions.

$$x_{new,j}^{t}=\begin{cases}x_{new,j}^{t}, & \text{if } rand\le CP \text{or } j=j_{rand}\\ x_{i,j}^{t}, & otherwise\end{cases} \tag{19}$$

where, $x_{new,j}^{t}$ is the new individual, and $x_{i,j}^{t}$ is the current individual. j_{rand} is a random number from 1 to D.

The crossover probability CP increases adaptively with iteration, so that the algorithm has a large probability to search for new areas in the process of evolution.

$$CP=0.9-0.2e^{1-\frac{T}{T-t+1}} \tag{20}$$

where, T is the maximum number of iterations, and t is the current number of iterations.

3.5. Complexity analysis

In this section, the time complexity of the RLBSO is evaluated and compared with that of the original BSO. The time complexity of the original BSO detail steps is as follows.

(1) The complexity of population initialization is $O(N)$.

(2) Calculating the fitness value costs $O(N)$.

(3) The complexity of dividing the population into subpopulations using k-means is $O(kN^2)$.

(4) The perturbation generated by randomly selecting a cluster center needs $O(N^2)$.

(5) Generating the individual to be updated costs $O(N^2)$.

(6) The wasted time generated by the step size is $O(N^2)$.

(7) The computing complexity of new individual generation and fitness value is $O(N^2)$ respectively.

The overall time complexity of BSO is as follows.

$$\begin{aligned}&O(N)+O(N)+O(kN^2)+O(N^2)+O(N^2)\\&\quad+O(N^2)+O(N^2)+O(N^2)\\&=2O(N)+O(kN^2)+5O(N^2)\end{aligned}$$

The detailed calculation steps of the time complexity of RLBSO are as follows.

(1) The initialization of population needs $O(N)$.

(2) The complexity of calculating the fitness value is $O(N)$.

(3) Dividing the population into subpopulations using k-means needs $O(kN^2)$.

(4) In the guidance process of Q-learning, the computational complexity is $O(N^2)$.

(5) During the process of self-learning, the computational complexity of similarity measure is $O(N^2)$.

(6) Using different strategies to update individuals needs $O(N^2)$.

(7) The time complexity of new individual generation is $O(N^2)$.

(8) The calculation of fitness value costs $O(N^2)$.

The overall time complexity of RLBSO is as follows.

$$\begin{aligned}&O(N)+O(N)+O(kN^2)+O(N)+O(N^2)+O(N^2)+O(N^2)+O(N^2)\\&=2O(N)+O(kN^2)+5O(N^2)\end{aligned}$$

Table 1
ANOVA results for parameter settings of RLBSO.

Source	*Sum Sq.*	*d.f.*	*Mean Sq.*	*F*-ratio	*p*-value
Main effects					
N	54.4717	3	18.1572	190.36	**0**
E	0.3811	3	0.127	1.33	0.2698
$CP1$	0.5503	3	0.1834	1.92	0.1324
$CP2$	0.2893	3	0.0964	1.01	0.3923
Interactions					
$N*E$	1.336	9	0.1484	1.56	0.1427
$N*CP1$	1.4152	9	0.1572	1.65	0.1155
$N*CP2$	0.732	9	0.0813	0.85	0.5705
$E*CP1$	0.8056	9	0.0895	0.94	0.4969
$E*CP2$	0.7107	9	0.079	0.83	0.5924
$CP1*CP2$	1.729	9	0.1921	2.01	**0.048**
$N*E*CP1$	2.6942	27	0.0998	1.05	0.4224
$N*E*CP2$	2.6804	27	0.0993	1.04	0.4288
$N*CP1*CP2$	4.9159	27	0.1821	1.91	0.1039
$E*CP1*CP2$	2.5751	27	0.0954	1	0.4792
Error	7.7263	81	0.0954		
Total	83.0128	255			

Since the fifth step in the original BSO, cluster center perturbation, is deleted from RLBSO, the time complexity is relatively reduced $O(N^2)$. However, Q-learning and self-learning are increased in the RLBSO, and the computational complexity is relatively increased by $O(N)+O(N^2)$. Overall, RLBSO does not add computational complexity over the original BSO.

4. The experiments and discussion

4.1. Parameter setting

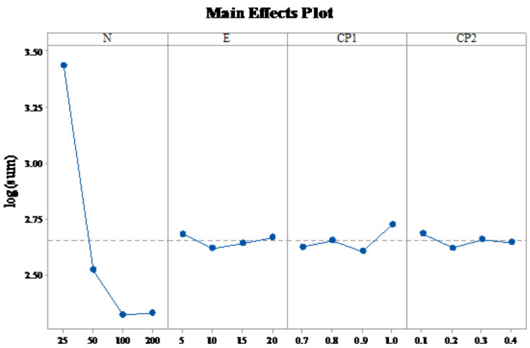
Reasonable parameters have significant influence on algorithm performance. The key parameters in this paper include population size, the number of elite individuals, and parameters in the crossover operation.

The orthogonal experiment design [55] is used to arrange the possible combinations of the four parameters to determine the value of the key parameters in this paper. According to the BSO literatures, the parameters are set as $N=\{25,50,100,200\}$, $E=\{5,10,15,20\}$, $CP1=\{0.7,0.8,0.9,1.0\}$, and $CP2=\{0.1,0.2,0.3,0.4\}$. Each parameter has 4 possible values, and 256 parameter combinations are obtained. The combination of the 256 parameters is verified in the RLBSO algorithm. Multivariate analysis of variance (ANOVA) is carried out on the experimental results, and the results of ANOVA are shown in Table 1.

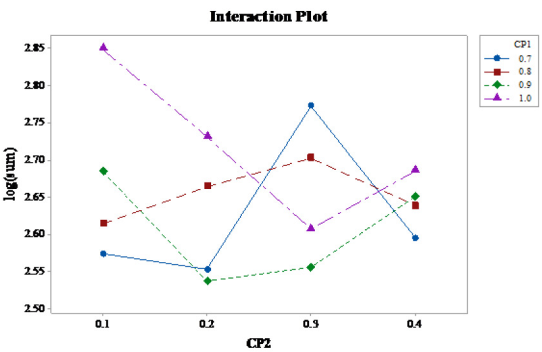
If there is a significant difference between the two variables, the p-value is less than 0.05. From Table 1, the p-value corresponding to N is less than 0.05, and the corresponding $F-$ratio is also the largest. Therefore, N has a significant impact on the algorithm. The value of $N*E$ is greater than 0.05, indicating that the interaction between N and E is significant.

Since the combined p-value of $CP1$ and $CP2$ is less than 0.05, no useful information could be obtained from the main effects plot. The values of $CP1$ and $CP2$ are 0.9 and 0.2 from the interaction plot Fig. 3(b), respectively. The horizontal axis is the values of $CP1$, and the vertical axis is the logarithm of the sum of all the functions evaluated by the algorithm with different parameter combinations.

Fig. 3(a) is the main effect plot, in which the horizontal axis is the values of the four parameters, and the vertical axis is the logarithm of the sum of all the functions evaluated by the algorithm with different parameter combinations. The optimal value corresponding to N is 100, and the optimal value of E is 10 from Fig. 3(a).



(a)



(b)

Fig. 3. Main effects plot and interaction plot of parameters.

4.2. The benchmark test

The RLBSO is compared with eight BSO variants, which are the MSBSO [43], ALBSO [45], CBSO [35], MDBSO [36], ASBSO [37], GBSO [38], OLBSO [48], and MIIBSO [56]. In addition, RLBSO is compared with four state-of-the-art algorithms, which are CJADE [23], TLBO-FL [57], PPSO [58], and EPSO [59]. The algorithms are tested on the CEC2017 benchmark test suite. All the test functions are divided into four groups in the test suite. The first group is unimodal functions, including f_1 and f_3. The second group is multimodal functions, including $f_4 - f_{10}$. The third group is hybrid functions, including $f_{11} - f_{20}$. The fourth group is composition functions, including $f_{21} - f_{30}$. The algorithms are run on a PC with a 3.4 GHZ Intel (R), Core (TM) i7- 6700 CPU, 8 GB of RAM and 64-bit OS. According to the experimental setting in the CEC2017 benchmark test suite, the algorithms are independently run 51 times on the 29 functions.

Algorithm	Parameter setting
RLBSO	$N = 100, M = 5, Elite = 10, CP1 = 0.9, CP2 = 0.2$
MSBSO	$N = 100, M = 5, P_{global} = 0.6, P_{local} = 0.8, k = 20$
ALBSO	$N = 100, M = 5, P_{5a} = 0.2, P = 0.8, P_{global} = 0.4, P_{local} = 0.5, k = 20$
OLBSO	$N = 100, M = 5, P_{5a} = 0.2, P = 0.8, P_{global} = 0.7, P_{local} = 0.7, k = 20$
CBSO	$N = 100, M = 5, P_{5a} = 0.2, P = 0.8, P_{global} = 0.4, P_{local} = 0.5, k = 20$
ASBSO	$N = 100, M = 4, P_{5a} = 0.2, P = 0.8, P_{global} = 0.4, P_{local} = 0.5, k = 10$
MDBSO	$N = 100, M = 5$
GBSO	$N = 25, M = 5, P = 0.8, P_{global} = 0.4, P_{local} = 0.5, k = 20, C_{\min} = 0.2, C_{\max} = 0.8$
MIIBSO	$N = 100, M = 5, P = 0.8, \Delta = 20, \nabla = 30$
TLBO-FL	$N = 100$
PPSO	$N = 200, \varphi_1 = -0.0025, \varphi_2 = 0.0025$
CJADE	$N = 40, CR_m = 0.5, F_m = 0.5$
EPSO	$N = 20, \omega = 0.9 - 0.2, c_1 = 2, c_2 = 2$

The dimensions are $D = 10$, $D = 30$, $D = 50$, and $D = 100$. The *MaxEFS* is $10000 * D$. The population size is $N = 100$. The parameter setting of RLBSO and eight BSO variants and four state-of-the-art algorithms is shown as follows.

The mean and standard deviation values of RLBSO and its variants with $D = 10$, $D = 30$, $D = 50$, and $D = 100$ are shown in Tables 2–5. The performance of RLBSO is better than that of the other eight comparison algorithms from Tables 2–5. The RLBSO performs better on 15 functions than the other eight comparison algorithms in 10D. The RLBSO performs better on 13 functions than the other eight comparison algorithms in 30D. The effect of RLBSO with 24 functions in 50D is better than that of the comparison algorithms. RLBSO has 22 functions in 100D that are better than the comparison algorithms. The RLBSO finds the optimal solution on functions f_1, f_3 and f_9 with $D = 10$. In general, RLBSO has significant performance in solving hybrid functions and composition functions in high dimension.

The convergence curves of nine algorithms on functions f_1, f_6, f_{15}, and f_{30} with $D = 10$, $D = 30$, $D = 50$, and $D = 100$ are shown in Figs. 4–7. f_1, f_6, f_{15}, and f_{30} are taken as representatives of the four function types. According to the test standard of CEC2017 test suite, 14 error-values are taken from the results of algorithm to draw the convergence curves. The horizontal axis represents the evaluation times of the functions; The vertical axis represents the logarithm of the error-value (the difference between the fitness value calculated by the algorithm and the optimal value for the given function). The RLBSO has a faster convergence speed than the other comparison algorithms from the curves. Therefore, RLBSO has more advantages in solving high dimension problems.

The mean and standard deviation values of RLBSO and four comparison algorithms with $D = 10$, $D = 30$, $D = 50$, and $D = 100$ are shown in Tables 6–9. The performance of RLBSO is better than that of the TLBO-FL, PPSO, EPSO. In addition, RLBSO performs better on f_3, f_{11}, f_{27}, and f_{30} than CJADE in four dimensions. The boxplots of five algorithms on functions f_1, f_6, f_{15}, and f_{30} with $D = 10$, $D = 30$, $D = 50$, and $D = 100$ are shown in Figs. 8–11. The horizontal axis represents the algorithms, and the vertical axis represents the error between the candidate solution and the optimal solution. The standard variance of RLBSO is smaller than that of the other four comparison algorithms from the boxplots. The stability of RLBSO is better than that of the other comparison algorithms.

4.3. The results of Friedman test and the Wilcoxon symbolic rank test

Friedman test is used to analyze the significant difference between the mean errors of the six algorithms on the CEC 2017 benchmark test suite. The statistical results of Friedman test of the nine algorithms with $D = 10, 30, 50$, and 100 are shown in Fig. 12. The horizontal axis represents the algorithm, and the vertical axis represents the average rank of each algorithm result. The average rank of RLBSO in different dimensions is the smallest of the nine algorithms with $\alpha = 0.05$.and $\alpha = 0.1$. RLBSO and MSBSO have no significant difference under the critical difference of confidence interval of 90% and 95%, and RLBSO has significant difference with other seven comparison algorithms.

Wilcoxon symbolic rank test is used to analyze the significant difference between the mean errors of RLBSO and the other eight BSO variants, respectively. The results of RLBSO and the other eight comparison algorithms are shown in Tables 10–13. R^+ is the sum of the rank that RLBSO is superior to the current comparison algorithms. R^- is the sum of the rank that the current comparison algorithm is superior to RLBSO. The rank sum of R^+ of RLBSO is higher than the rank sum of R^- in the eight groups of comparison result. p-value represents asymptotic significance. When $p - value < \alpha$, the corresponding α position of the algorithm is "yes", which represents that the two algorithms have significant differences.

Table 2
The results of RLBSO and eight BSO variants (10-dimensional benchmark functions).

Fun	MSBSO		ALBSO		CBSO		ASBSO		MDBSO		MIIBSO		GBSO		OLBSO		RLBSO	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	0.00E+00	0.00E+00	8.66E+06	4.33E+07	1.19E+03	1.38E+03	1.22E+03	1.75E+03	1.41E+10	4.35E+09	1.36E+05	1.26E+05	3.49E+09	9.03E+08	1.36E+03	1.88E+03	**0.00E+00**	**0.00E+00**
3	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	5.37E−12	1.62E−11	1.97E+04	2.95E+03	2.78E+02	2.55E+02	8.79E+03	2.95E+03	1.73E+03	1.59E+03	**0.00E+00**	**0.00E+00**
4	**2.35E−01**	**9.47E−01**	1.65E+00	1.31E+00	6.87E+00	1.59E+01	3.04E+00	1.70E+00	1.30E+03	4.91E+02	6.02E+00	2.28E+00	1.94E+02	5.88E+01	1.25E+01	1.79E+01	3.13E−01	1.08E+00
5	9.57E+00	4.11E+00	3.55E+01	1.34E+01	3.61E+01	1.14E+01	3.73E+01	1.27E+01	1.18E+02	1.95E+01	9.44E+00	**3.15E+00**	7.17E+01	8.71E+00	3.60E+01	1.45E+01	**9.31E+00**	4.32E+00
6	9.81E−02	1.91E−01	2.89E+01	9.64E+00	2.57E+01	9.09E+00	2.51E+01	7.96E+00	6.47E+01	1.29E+01	7.26E−02	6.42E−02	3.88E+01	5.58E+00	7.57E+00	5.33E+00	**1.66E−03**	**5.41E−03**
7	2.14E+01	4.57E+00	6.23E+01	2.02E+01	6.30E+01	2.46E+01	6.22E+01	2.45E+01	1.87E+02	**5.05E−05**	**1.72E+01**	3.24E+00	1.67E+02	2.28E+01	5.71E+01	1.96E+01	1.93E+01	5.05E+00
8	9.35E+00	4.03E+00	2.45E+01	9.02E+00	2.08E+01	8.82E+00	2.60E+01	1.04E+01	9.42E+01	4.51E+00	**6.65E+00**	**2.92E+00**	6.52E+01	8.72E+00	3.14E+01	1.21E+01	8.80E+00	3.60E+00
9	2.67E−02	1.08E−01	3.18E+02	1.84E+02	2.44E+02	1.25E+02	2.23E+02	1.13E+02	1.62E+03	1.96E+02	1.33E+00	2.17E+00	6.98E+02	1.65E+02	1.99E+02	1.55E+02	**0.00E+00**	**0.00E+00**
10	4.77E+02	2.32E+02	1.07E+03	2.60E+02	1.12E+03	2.82E+02	1.05E+03	2.61E+02	1.87E+03	**1.64E+02**	**3.37E+02**	1.91E+02	1.51E+03	1.65E+02	9.92E+02	3.44E+02	5.63E+02	2.43E+02
11	**2.70E+00**	2.35E+00	6.30E+01	3.54E+01	5.14E+01	2.85E+01	5.89E+01	3.95E+01	2.61E+03	2.10E+03	9.47E+00	3.72E+00	3.38E+02	1.51E+02	5.50E+01	4.83E+01	2.73E+00	**2.04E+00**
12	**7.09E+01**	**7.48E+01**	1.60E+04	3.73E+04	8.44E+04	9.28E+04	6.36E+04	4.31E+04	8.30E+08	5.14E+08	8.53E+04	1.21E+05	6.18E+07	3.09E+07	1.30E+06	1.19E+06	1.63E+02	1.30E+02
13	8.19E+00	3.05E+00	2.25E+03	3.04E+03	7.43E+03	5.38E+03	7.10E+03	5.70E+03	9.16E+06	1.33E+07	8.78E+03	6.87E+03	8.62E+04	6.21E+04	9.29E+03	7.57E+03	**7.41E+00**	**2.66E+00**
14	2.23E+01	**1.53E+00**	9.01E+01	8.51E+01	3.21E+02	4.43E+02	4.43E+02	8.71E+02	2.12E+02	9.39E+01	3.27E+03	3.89E+03	8.53E+02	8.46E+02	2.23E+03	3.07E+03	**1.73E+01**	8.59E+00
15	2.01E+00	1.62E+00	1.34E+02	1.64E+02	2.14E+03	2.02E+03	2.69E+03	3.33E+03	4.01E+03	2.77E+03	6.72E+03	1.03E+04	2.55E+03	3.07E+03	3.34E+03	4.44E+03	**1.91E+00**	**1.40E+00**
16	8.29E+01	9.49E+01	2.58E+02	1.11E+02	2.73E+02	1.35E+02	2.56E+02	1.38E+02	6.62E+02	1.23E+02	1.12E+02	1.04E+02	2.54E+02	9.35E+01	2.52E+02	1.37E+02	**7.99E+01**	**8.41E+01**
17	3.38E+01	2.28E+01	6.91E+01	3.79E+01	6.62E+01	2.85E+01	6.98E+01	4.06E+01	3.09E+02	7.99E+01	**1.46E+01**	1.29E+01	1.16E+02	2.30E+01	6.76E+01	4.48E+01	1.53E+01	**1.44E+01**
18	2.06E+01	**6.20E−01**	2.44E+03	3.18E+03	6.63E+03	7.44E+03	8.24E+03	8.55E+03	1.76E+07	2.50E+07	9.53E+03	1.01E+04	1.33E+06	1.51E+06	1.84E+04	1.73E+04	**1.69E+01**	7.50E+00
19	1.86E+00	**8.08E−01**	2.95E+02	1.81E+03	2.41E+03	2.75E+03	1.92E+03	2.37E+03	6.85E+03	7.79E+03	4.94E+03	5.45E+03	7.69E+03	9.07E+03	4.81E+03	6.53E+03	**1.08E+00**	8.09E−01
20	2.37E+01	9.24E+00	1.29E+02	6.09E+01	1.35E+02	5.54E+01	1.34E+02	6.98E+01	2.66E+02	5.71E+01	**4.54E+00**	**6.33E+00**	1.31E+02	2.86E+01	7.57E+01	4.85E+01	2.97E+01	4.20E+01
21	1.80E+02	5.20E+01	2.00E+02	5.68E+01	1.85E+02	6.08E+01	1.95E+02	6.35E+01	2.83E+02	4.33E+01	1.89E+02	4.31E+01	1.54E+02	**2.53E+01**	1.50E+02	5.90E+01	**1.09E+02**	3.02E+01
22	**9.70E+01**	1.86E+01	1.06E+02	5.12E+00	1.01E+02	**1.06E+00**	9.88E+01	9.93E+00	1.35E+03	3.64E+02	1.03E+02	2.31E+01	2.93E+02	1.02E+02	1.06E+02	4.25E+00	1.78E+02	2.38E+02
23	3.17E+02	5.22E+00	3.98E+02	3.53E+01	3.92E+02	2.52E+01	3.87E+02	3.02E+01	4.30E+02	2.05E+01	3.14E+02	4.99E+00	3.88E+02	2.78E+01	3.48E+02	1.82E+01	**3.13E+02**	**4.38E+00**
24	3.21E+02	7.37E+01	3.72E+02	1.36E+02	3.46E+02	1.39E+02	3.48E+02	1.38E+02	5.40E+02	4.99E+01	3.39E+02	3.51E+01	3.71E+02	5.65E+01	**3.19E+02**	1.15E+02	3.45E+02	**6.32E+00**
25	4.26E+02	2.34E+01	4.29E+02	5.22E+01	4.27E+02	2.21E+01	4.24E+02	2.55E+01	1.40E+03	3.62E+02	4.36E+02	2.11E+01	5.77E+02	4.42E+01	4.31E+02	**2.06E+01**	**4.20E+02**	2.32E+01
26	**3.40E+02**	7.89E+01	8.13E+02	3.80E+02	6.96E+02	3.21E+02	7.44E+02	3.39E+02	1.91E+03	3.67E+02	3.60E+02	1.00E+02	7.42E+02	**7.72E+01**	4.89E+02	3.07E+02	3.50E+02	1.26E+02
27	**3.74E+02**	**2.31E+00**	4.91E+02	5.44E+01	4.84E+02	3.99E+01	4.85E+02	5.13E+01	5.41E+02	4.78E+01	4.78E+02	4.52E+01	4.66E+02	2.01E+01	4.17E+02	6.84E+01	3.80E+02	3.02E+01
28	4.74E+02	3.85E+00	4.13E+02	8.29E+01	4.47E+02	1.55E+02	**3.85E+02**	6.74E+01	5.00E+02	**3.95E−03**	4.94E+02	5.25E+00	4.99E+02	1.75E+00	4.96E+02	1.16E+02	4.64E+02	3.31E+01
29	**2.49E+02**	2.09E+01	3.65E+02	8.58E+01	3.55E+02	6.94E+01	3.51E+02	8.35E+01	6.70E+02	1.11E+02	2.85E+02	2.75E+01	4.19E+02	3.90E+01	3.55E+02	5.67E+01	2.56E+02	**1.93E+01**
30	2.08E+02	**1.54E+01**	2.50E+03	3.03E+03	2.08E+05	5.15E+05	3.76E+04	2.63E+04	2.02E+07	1.44E+07	1.10E+05	1.85E+05	1.57E+06	1.40E+06	6.10E+05	4.81E+05	**2.08E+02**	1.58E+01

Table 3
The results of RLBSO and eight BSO variants (30-dimensional benchmark functions).

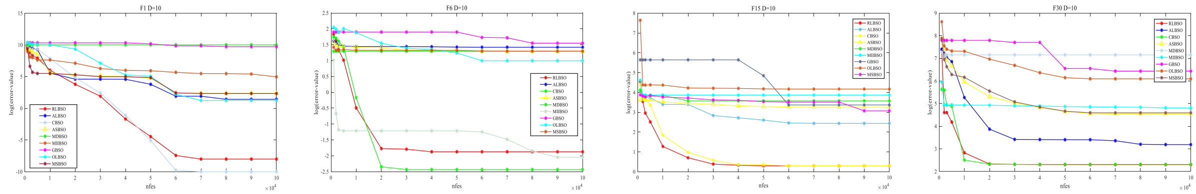
Fun	MSBSO		ALBSO		CBSO		ASBSO		MDBSO		MIIBSO		GBSO		OLBSO		RLBSO	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	1.84E−03	7.46E−03	7.47E+07	3.05E+08	1.81E+03	1.80E+03	2.16E+03	2.65E+03	8.02E+10	3.29E+09	5.81E+06	2.79E+06	5.34E+10	6.46E+09	4.70E+04	3.22E+04	**1.63E−05**	**1.16E−04**
3	**7.43E+00**	**2.85E+01**	4.58E+03	2.22E+04	2.13E+02	3.56E+02	1.67E+02	2.93E+02	9.40E+04	1.59E+02	1.11E+04	3.98E+03	1.11E+05	1.48E+04	4.82E+03	8.85E+03	3.35E+03	5.51E+03
4	**6.08E+00**	**7.16E+00**	8.78E+01	6.84E+01	8.63E+01	2.79E+01	7.32E+01	1.90E+01	2.28E+04	1.39E+03	1.26E+02	1.75E+01	1.06E+04	1.92E+03	7.22E+01	2.09E+01	2.44E+01	2.41E+01
5	**5.59E+01**	1.81E+01	1.92E+02	3.44E+01	1.86E+02	3.82E+01	2.04E+02	4.16E+01	5.07E+02	**1.08E−06**	6.52E+01	1.52E+01	4.38E+02	2.46E+01	1.85E+02	4.73E+01	5.62E+01	1.59E+01
6	1.65E+00	1.91E+00	5.53E+01	6.33E+00	5.34E+01	6.79E+00	5.32E+01	6.11E+00	1.09E+02	3.45E+00	1.79E+00	7.56E−01	8.72E+01	6.13E+00	1.31E+01	6.37E+00	**1.16E−01**	**1.65E−01**
7	8.44E+01	1.73E+01	5.70E+02	1.29E+02	4.93E+02	7.90E+01	5.08E+02	9.82E+01	8.87E+02	**5.75E−06**	1.22E+02	2.15E+01	1.42E+03	1.19E+02	2.28E+02	3.80E+01	**8.30E+01**	1.44E+01
8	5.68E+01	1.72E+01	1.47E+02	3.50E+01	1.40E+02	2.85E+01	1.38E+02	3.09E+01	4.37E+02	**3.07E−07**	**4.93E+01**	1.18E+01	3.92E+02	2.09E+01	1.73E+02	4.39E+01	5.33E+01	1.68E+01
9	1.62E+02	2.06E+02	3.54E+03	7.93E+02	3.12E+03	7.51E+02	3.30E+03	7.65E+02	1.66E+04	**1.44E−02**	4.72E+02	2.32E+02	1.41E+04	1.52E+03	5.16E+03	2.21E+03	**1.32E+01**	1.59E+01
10	3.58E+03	6.92E+02	4.47E+03	6.01E+02	4.25E+03	5.79E+02	4.32E+03	6.36E+02	8.63E+03	2.96E+02	**2.67E+03**	4.55E+02	7.46E+03	**2.51E+02**	4.73E+03	7.02E+02	2.97E+03	5.52E+02
11	**1.96E+01**	**1.03E+01**	1.48E+02	5.09E+01	1.37E+02	3.88E+01	1.38E+02	4.01E+01	1.44E+04	1.30E+03	1.92E+02	6.85E+01	7.18E+03	1.42E+03	1.77E+02	6.21E+01	2.18E+01	1.18E+01
12	3.22E+04	3.06E+04	1.00E+07	5.00E+07	1.81E+06	9.73E+05	1.96E+06	1.06E+06	2.17E+10	4.32E+09	2.01E+06	1.23E+06	5.64E+09	1.26E+09	3.59E+06	2.85E+06	**3.05E+04**	**2.96E+04**
13	**1.25E+02**	**1.43E+02**	9.60E+06	5.51E+07	5.39E+04	3.19E+04	5.14E+04	2.92E+04	1.77E+10	5.78E+09	1.24E+04	8.79E+03	3.10E+09	1.08E+09	2.40E+04	1.68E+04	2.20E+02	2.76E+02
14	4.66E+01	**1.00E+01**	5.39E+03	1.84E+04	6.70E+03	5.73E+03	5.57E+03	5.37E+03	9.09E+06	5.88E+06	5.37E+05	3.85E+05	9.57E+05	4.44E+05	2.51E+04	3.67E+04	**4.50E+01**	1.43E+01
15	7.45E+01	**1.89E+02**	6.53E+04	1.69E+05	2.92E+04	1.83E+04	2.72E+04	1.77E+04	3.37E+09	1.24E+09	4.68E+03	5.15E+03	3.03E+08	1.46E+08	8.89E+03	7.39E+03	**7.30E+01**	2.19E+02
16	9.64E+02	2.92E+02	1.57E+03	2.94E+02	1.59E+03	3.03E+02	1.48E+03	4.00E+02	5.09E+03	9.91E+02	9.51E+02	**2.82E+02**	3.08E+03	3.04E+02	1.48E+03	3.08E+02	**9.16E+02**	2.92E+02
17	5.64E+02	1.85E+02	7.38E+02	2.59E+02	7.33E+02	2.81E+02	7.64E+02	2.64E+02	5.92E+03	6.84E+03	**3.44E+02**	1.74E+02	1.36E+03	**1.68E+02**	7.29E+02	2.51E+02	4.33E+02	2.14E+02
18	1.64E+04	2.40E+04	1.12E+05	1.46E+05	1.25E+05	9.73E+04	1.29E+05	7.27E+04	1.63E+08	1.17E+08	6.24E+05	8.37E+05	1.26E+07	5.34E+06	2.70E+05	2.07E+05	**9.50E+03**	**8.67E+03**
19	1.85E+01	**5.42E+00**	6.26E+04	4.58E+04	1.33E+05	6.02E+04	1.49E+05	6.13E+04	3.28E+09	6.71E+08	4.21E+03	4.67E+03	4.42E+08	2.13E+08	2.98E+04	5.22E+04	**1.59E+01**	6.05E+00
20	3.72E+02	1.83E+02	7.27E+02	2.09E+02	7.05E+02	1.82E+02	7.16E+02	2.00E+02	1.36E+03	1.44E+02	**3.47E+02**	1.57E+02	8.11E+02	**1.06E+02**	5.73E+02	2.21E+02	4.43E+02	1.85E+02
21	**2.56E+02**	1.59E+01	4.11E+02	3.97E+01	3.85E+02	3.46E+01	4.08E+02	4.09E+01	7.68E+02	5.91E+01	2.56E+02	**1.28E+01**	5.96E+02	2.26E+01	4.06E+02	6.11E+01	2.59E+02	1.55E+01
22	3.46E+03	1.13E+03	4.66E+03	1.09E+03	3.99E+03	1.81E+03	4.09E+03	1.52E+03	8.80E+03	**3.59E+02**	2.89E+03	1.12E+03	6.01E+03	6.97E+02	**3.03E+03**	2.56E+03	3.38E+03	6.06E+02
23	4.18E+02	1.99E+01	1.05E+03	1.21E+02	1.01E+03	1.23E+02	9.79E+02	1.28E+02	1.40E+03	1.44E+02	4.41E+02	1.99E+01	9.41E+02	4.62E+01	6.10E+02	6.73E+01	**4.12E+02**	**1.55E+01**
24	4.83E+02	1.95E+01	1.13E+03	1.96E+02	1.11E+03	9.83E+01	1.11E+03	1.11E+02	1.56E+03	1.07E+02	5.23E+02	2.56E+01	1.12E+03	6.22E+01	6.78E+02	6.89E+01	**4.82E+02**	**1.38E+01**
25	**3.78E+02**	9.88E−01	3.96E+02	2.38E+01	3.89E+02	8.01E+00	3.82E+02	7.57E+00	4.71E+03	**2.20E−06**	4.29E+02	2.45E+01	3.89E+03	6.70E+02	4.07E+02	2.26E+01	3.79E+02	7.39E−01
26	**1.30E+03**	**1.77E+02**	5.75E+03	1.34E+03	5.55E+03	1.55E+03	5.67E+03	1.21E+03	1.14E+04	1.11E+03	2.19E+03	2.96E+02	7.31E+03	5.98E+02	3.38E+03	1.27E+03	1.42E+03	2.54E+02
27	**4.49E+02**	1.94E−04	1.23E+03	2.75E+02	1.22E+03	2.16E+02	1.18E+03	2.93E+02	5.51E+02	2.64E+02	5.00E+02	2.34E−04	5.00E+02	**7.07E−05**	4.96E+02	7.63E+00	5.00E+02	2.10E−04
28	5.00E+02	2.16E+00	4.14E+02	2.02E+01	**3.94E+02**	3.43E+01	3.96E+02	3.58E+01	6.11E+02	7.96E+02	5.00E+02	2.70E−04	5.00E+02	**9.81E−05**	4.33E+02	2.44E+01	4.98E+02	3.46E+00
29	7.33E+02	2.50E+02	1.61E+03	3.61E+02	1.62E+03	3.01E+02	1.59E+03	3.50E+02	9.50E+03	6.83E+03	7.77E+02	**1.69E+02**	2.06E+03	3.98E+02	1.45E+03	2.71E+02	**5.58E+02**	1.99E+02
30	**2.17E+02**	**5.28E+00**	2.16E+06	1.16E+07	5.46E+05	3.09E+05	5.02E+05	2.62E+05	2.76E+09	1.34E+09	1.28E+04	6.26E+03	2.98E+08	1.12E+08	7.18E+05	4.50E+05	2.20E+02	1.26E+01

Table 4
The results of RLBSO and eight BSO variants (50-dimensional benchmark functions).

Fun	MSBSO		ALBSO		CBSO		ASBSO		MDBSO		MIIBSO		GBSO		OLBSO		RLBSO	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	2.03E+02	1.27E+03	6.42E+07	4.58E+08	3.91E+03	3.29E+03	2.73E+03	2.52E+03	1.35E+11	**3.87E−01**	3.57E+08	1.11E+08	1.33E+11	1.17E+10	2.82E+05	1.72E+05	**6.77E−01**	4.50E+00
3	9.16E+04	4.36E+04	**1.30E+01**	**3.76E+00**	6.48E+03	2.74E+03	5.87E+03	3.27E+03	2.81E+05	5.90E+03	3.55E+04	8.34E+03	2.41E+05	2.90E+04	2.32E+01	8.06E+00	1.37E+05	5.95E+04
4	3.92E+01	2.16E+01	1.90E+02	1.45E+02	1.58E+02	5.54E+01	1.51E+02	5.02E+01	5.62E+04	**1.86E−06**	3.72E+02	5.67E+01	3.56E+04	5.89E+03	1.51E+02	4.72E+01	**3.85E+01**	2.08E+01
5	1.27E+02	3.14E+01	3.23E+02	3.80E+01	3.25E+02	4.45E+01	3.23E+02	5.35E+01	8.15E+02	**2.15E−07**	2.07E+02	2.49E+01	8.32E+02	4.12E+01	3.95E+02	1.03E+02	**1.20E+02**	3.27E+01
6	7.69E+00	5.80E+00	6.06E+01	5.66E+00	5.91E+01	6.94E+00	5.99E+01	6.31E+00	1.08E+02	**2.79E−07**	1.20E+01	2.92E+00	1.07E+02	4.42E+00	1.24E+01	4.57E+00	**1.66E+00**	2.10E+00
7	1.71E+02	2.98E+01	1.20E+03	1.96E+02	1.00E+03	1.63E+02	9.85E+02	1.41E+02	1.50E+03	**3.99E−06**	3.92E+02	3.86E+01	3.13E+03	2.36E+02	4.36E+02	9.00E+01	**1.67E+02**	3.12E+01
8	1.19E+02	2.94E+01	3.34E+02	3.74E+01	3.39E+02	3.56E+01	3.37E+02	4.25E+01	8.13E+02	4.21E+01	2.08E+02	3.26E+01	8.29E+02	3.25E+01	3.36E+02	9.26E+01	**1.13E+02**	**2.88E+01**
9	1.07E+03	1.15E+03	1.13E+04	2.39E+03	1.11E+04	1.46E+03	1.05E+04	1.48E+03	5.31E+04	**3.18E+02**	5.13E+03	1.25E+03	4.75E+04	4.98E+03	1.28E+04	4.78E+03	**3.83E+02**	5.04E+02
10	6.56E+03	1.01E+03	7.42E+03	8.26E+02	7.34E+03	8.12E+02	7.38E+03	8.85E+02	1.54E+04	5.04E+02	6.65E+03	7.27E+02	1.37E+04	**3.49E+02**	7.76E+03	8.50E+02	**5.86E+03**	7.90E+02
11	5.52E+01	3.19E+01	2.32E+02	5.29E+01	2.07E+02	5.10E+01	1.99E+02	4.06E+01	5.66E+04	4.82E+03	7.56E+02	2.83E+02	2.18E+04	3.64E+03	3.10E+02	8.46E+01	**5.44E+01**	**2.21E+01**
12	2.97E+05	2.47E+05	4.93E+07	1.89E+08	1.24E+07	6.58E+06	1.41E+07	6.99E+06	1.23E+11	1.76E+10	2.24E+07	7.10E+06	4.68E+10	8.32E+09	1.24E+07	5.88E+06	**1.92E+05**	**1.33E+05**
13	2.33E+03	2.89E+03	1.44E+07	8.36E+07	6.54E+04	6.77E+04	6.04E+04	4.00E+04	6.94E+10	1.57E+10	3.76E+04	1.53E+04	1.74E+10	3.38E+09	3.99E+04	2.49E+04	**6.09E+02**	**9.84E+02**
14	1.35E+04	**3.12E+01**	1.19E+05	5.58E+05	3.54E+04	2.67E+04	3.69E+04	2.40E+04	1.50E+08	8.38E+07	1.92E+06	1.27E+06	1.34E+07	5.60E+06	8.54E+04	7.16E+04	**8.52E+03**	1.70E+04
15	5.97E+02	1.72E+03	1.01E+06	7.00E+06	2.78E+04	1.79E+04	2.84E+04	1.64E+04	1.92E+10	4.92E+09	9.51E+03	5.36E+03	4.34E+09	1.34E+09	1.24E+04	1.04E+04	**2.97E+02**	**3.81E+02**
16	1.84E+03	4.23E+02	2.26E+03	4.95E+02	2.21E+03	4.11E+02	2.09E+03	4.50E+02	1.07E+04	1.61E+03	**1.52E+03**	4.22E+02	6.06E+03	4.43E+02	2.42E+03	5.35E+02	1.71E+03	**3.43E+02**
17	1.25E+03	3.65E+02	1.99E+03	4.38E+02	1.94E+03	4.05E+02	1.87E+03	3.86E+02	1.48E+05	4.87E+04	1.21E+03	3.49E+02	6.27E+03	1.86E+03	1.94E+03	3.78E+02	**1.20E+03**	**3.11E+02**
18	1.38E+05	1.86E+05	6.43E+05	2.83E+06	3.04E+05	1.27E+05	3.07E+05	1.46E+05	6.02E+08	2.41E+08	3.63E+06	2.42E+06	6.06E+07	1.82E+07	8.30E+05	6.53E+05	**1.07E+05**	**7.74E+04**
19	**4.35E+01**	**1.44E+01**	6.38E+05	6.47E+05	4.87E+05	2.35E+05	4.96E+05	2.08E+05	9.20E+09	1.75E+09	1.75E+04	8.73E+03	1.74E+09	5.24E+08	2.09E+04	2.14E+04	7.20E+02	4.06E+03
20	1.08E+03	3.47E+02	1.43E+03	3.02E+02	1.45E+03	2.89E+02	1.50E+03	2.84E+02	2.79E+03	1.43E+02	**7.97E+02**	2.76E+02	2.12E+03	**1.33E+02**	1.41E+03	3.85E+02	9.07E+02	2.50E+02
21	3.22E+02	**2.69E+01**	6.47E+02	6.89E+01	6.46E+02	5.95E+01	6.93E+02	8.07E+01	1.34E+03	8.42E+01	3.93E+02	3.04E+01	1.04E+03	3.69E+01	5.35E+02	1.03E+02	**3.23E+02**	3.03E+01
22	6.78E+03	1.11E+03	8.28E+03	1.10E+03	8.00E+03	7.41E+02	7.98E+03	8.57E+02	1.57E+04	5.68E+02	7.58E+03	8.34E+02	1.39E+04	**3.86E+02**	8.43E+03	1.47E+03	**6.86E+03**	1.00E+03
23	5.39E+02	4.76E+01	1.79E+03	2.05E+02	1.74E+03	2.09E+02	1.67E+03	1.98E+02	2.30E+03	1.28E+02	7.30E+02	4.39E+01	1.67E+03	5.21E+01	9.42E+02	1.13E+02	**5.32E+02**	**4.40E+01**
24	6.30E+02	3.52E+01	1.80E+03	1.10E+02	1.71E+03	2.10E+02	1.73E+03	1.91E+02	2.90E+03	1.97E+02	8.25E+02	4.04E+01	1.97E+03	1.08E+02	9.77E+02	1.06E+02	**6.04E+02**	**2.80E+01**
25	4.43E+02	1.98E+01	4.96E+02	4.48E+01	5.68E+02	2.94E+01	4.62E+02	3.74E+01	1.69E+04	**6.32E−07**	7.73E+02	4.11E+01	2.00E+04	2.24E+03	4.93E+02	4.08E+01	**4.37E+02**	1.41E+01
26	2.44E+03	4.82E+02	1.08E+04	1.53E+03	1.05E+04	8.85E+02	1.05E+04	8.22E+02	1.68E+04	**3.55E−06**	4.37E+03	5.07E+02	1.60E+04	1.33E+03	5.27E+03	2.22E+03	**2.12E+03**	6.13E+02
27	5.00E+02	1.87E−04	3.02E+03	6.16E+02	2.76E+03	6.31E+02	2.69E+03	6.62E+02	1.87E+03	1.94E+03	5.00E+02	2.32E−04	5.00E+02	**9.11E−05**	5.05E+02	2.03E+01	**5.00E+02**	1.92E−04
28	5.00E+02	2.09E−04	4.95E+02	2.67E+01	5.10E+02	1.75E+01	**4.89E+02**	1.91E+01	5.00E+02	**9.48E−08**	5.00E+02	2.24E−04	5.00E+02	7.53E−05	4.90E+02	1.05E+01	5.00E+02	2.38E−04
29	1.47E+03	4.72E+02	3.11E+03	4.59E+02	2.67E+03	4.73E+02	2.79E+03	4.82E+02	5.49E+05	5.86E+05	1.46E+03	**2.41E+02**	9.24E+03	2.31E+03	2.52E+03	4.75E+02	**1.16E+03**	3.09E+02
30	4.48E+02	6.00E+02	2.40E+07	4.99E+07	1.79E+07	1.90E+06	1.69E+07	2.04E+06	1.51E+10	4.35E+09	2.97E+06	6.03E+05	3.08E+09	7.33E+08	1.00E+07	2.42E+06	**3.07E+02**	**5.83E+01**

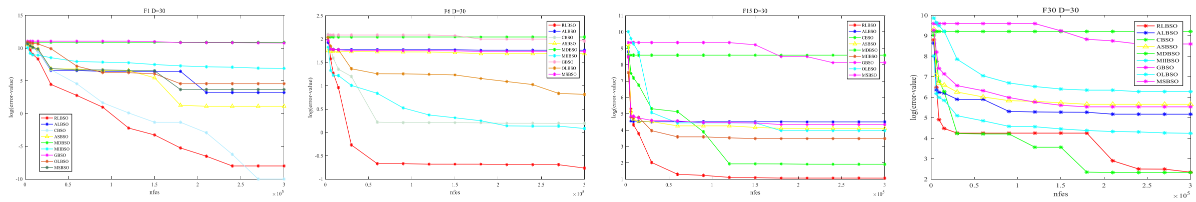
Table 5
The results of RLBSO and eight BSO variants (100-dimensional benchmark functions).

Fun	MSBSO		ALBSO		CBSO		ASBSO		MDBSO		MIIBSO		GBSO		OLBSO		RLBSO	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	1.41E+03	5.68E+03	7.87E+08	2.74E+09	1.18E+06	4.09E+05	5.90E+05	2.32E+05	2.97E+11	**1.04E+00**	1.21E+10	2.03E+09	3.93E+11	2.17E+10	1.26E+06	4.77E+05	**6.93E+02**	2.19E+03
3	1.72E+06	1.06E+06	**2.59+02**	4.67E+02	8.45E+04	5.56E+01	8.53E+04	1.38E+04	3.66E+05	**1.30E−02**	1.63E+05	1.74E+04	5.99E+05	4.57E+04	3.60E+02	6.87E+01	1.20E+05	5.16E+05
4	1.51E+02	4.76E+01	3.70E+02	4.31E+02	2.78E+02	5.00E+01	2.88E+02	6.85E+01	1.60E+05	**4.77E−07**	2.42E+03	3.71E+02	1.23E+05	1.21E+04	2.40E+02	5.16E+01	**1.41E+02**	4.59E+01
5	3.67E+02	8.81E+01	8.04E+02	6.24E+01	8.15E+02	6.85E+01	8.23E+02	7.18E+01	1.81E+03	**5.07E−08**	8.70E+02	5.60E+01	1.99E+03	6.48E+01	9.57E+02	2.37E+02	**3.08E+02**	6.50E+01
6	1.88E+01	7.88E+00	6.80E+01	5.04E+00	6.40E+01	4.04E+00	6.36E+01	3.80E+00	1.20E+02	**5.69E−09**	3.98E+01	4.12E+00	1.26E+02	3.84E+00	1.57E+01	6.02E+00	**1.54E+01**	5.97E+00
7	5.32E+02	1.21E+02	3.23E+03	3.31E+02	2.65E+03	2.72E+02	2.63E+03	2.66E+02	3.53E+03	**1.33E−07**	1.72E+03	1.74E+02	7.79E+03	4.57E+02	9.42E+02	1.90E+02	**5.31E+02**	1.20E+02
8	**3.03E+02**	5.03E+01	9.33E+02	9.63E+01	9.37E+02	7.95E+01	9.36E+02	8.20E+01	1.94E+03	**1.31E−06**	9.21E+02	5.97E+01	2.07E+03	6.45E+01	9.20E+02	2.67E+02	3.76E+02	7.84E+01
9	**5.31E+03**	3.66E+03	2.96E+04	6.04E+03	2.72E+04	2.90E+03	2.67E+04	3.20E+03	1.03E+05	**3.93E−04**	2.61E+04	4.13E+03	1.34E+05	7.54E+03	2.81E+04	9.66E+03	7.06E+03	3.84E+03
10	1.50E+04	2.06E+03	1.57E+04	1.11E+03	1.57E+04	1.23E+03	1.56E+04	1.26E+03	3.33E+04	6.29E+02	2.23E+04	1.31E+03	3.06E+04	**5.66E+02**	1.61E+04	1.43E+03	**1.46E+04**	1.69E+03
11	7.50E+02	2.92E+02	1.96E+03	2.37E+03	1.31E+03	1.72E+02	1.32E+03	**1.45E+02**	3.97E+05	5.61E+04	1.07E+04	2.37E+03	2.37E+05	2.08E+04	1.45E+03	1.94E+02	**5.89E+02**	2.57E+02
12	**9.08E+05**	5.42E+05	6.95E+08	1.26E+09	8.26E+07	2.03E+07	7.80E+07	1.63E+07	2.61E+11	**6.55E−01**	1.15E+09	2.11E+08	1.82E+11	1.90E+10	2.70E+07	8.18E+06	1.38E+06	9.48E+05
13	6.98E+03	9.01E+03	3.79E+04	1.17E+04	3.65E+04	1.26E+04	3.82E+04	1.01E+04	6.51E+10	5.54E+08	1.20E+05	1.00E+05	3.86E+10	3.91E+09	3.07E+04	1.07E+04	**4.37E+03**	**6.92E+03**
14	**2.72E+04**	**2.09E+04**	4.15E+05	2.30E+05	3.02E+05	1.26E+05	3.54E+05	1.49E+05	2.96E+08	4.70E+07	3.68E+06	1.42E+06	7.88E+07	1.71E+07	4.49E+05	2.18E+05	7.25E+04	4.49E+04
15	4.46E+03	6.35E+03	3.94E+06	2.50E+07	3.23E+04	1.23E+04	3.07E+04	1.08E+04	3.94E+10	3.09E+09	5.82E+03	**2.47E+03**	1.44E+10	2.62E+09	1.12E+04	4.27E+03	**3.53E+03**	6.39E+03
16	4.20E+03	**6.34E+02**	5.62E+03	7.38E+02	5.20E+03	7.24E+02	5.17E+03	7.52E+02	3.23E+04	4.42E+03	4.55E+03	7.79E+02	1.76E+04	1.27E+03	4.98E+03	7.67E+02	**3.93E+03**	6.38E+02
17	3.21E+03	5.91E+02	6.04E+03	9.21E+03	4.02E+03	7.40E+02	3.92E+03	**5.14E+02**	4.26E+07	2.66E+07	3.90E+03	5.97E+02	7.30E+05	5.02E+05	4.02E+03	6.34E+02	**2.97E+03**	5.82E+02
18	4.48E+05	2.57E+05	6.28E+05	2.19E+05	4.92E+05	1.76E+05	4.53E+05	**1.64E+05**	1.09E+09	3.73E+08	6.99E+06	4.08E+06	1.33E+08	3.49E+07	7.19E+05	2.81E+05	**2.61E+05**	1.74E+05
19	3.83E+03	5.61E+03	9.46E+06	2.77E+07	2.54E+06	1.08E+06	2.48E+06	1.15E+06	4.01E+10	2.01E+09	8.47E+03	**4.52E+03**	1.57E+10	3.18E+09	1.18E+05	1.21E+05	**2.01E+03**	6.43E+03
20	2.73E+03	6.11E+02	3.74E+03	5.27E+02	3.73E+03	4.69E+02	3.80E+03	4.94E+02	6.76E+03	2.96E+02	2.82E+03	4.97E+02	5.51E+03	**2.50E+02**	4.03E+03	6.09E+02	**2.72E+03**	4.94E+02
21	6.64E+02	8.96E+01	1.89E+03	2.16E+02	1.92E+03	2.00E+02	1.93E+03	1.60E+02	3.07E+03	1.50E+02	1.16E+03	**5.93E+01**	2.46E+03	7.59E+01	1.21E+03	1.78E+02	**5.77E+02**	9.67E+01
22	**1.60E+04**	1.47E+03	1.73E+04	1.39E+03	1.69E+04	**1.27E+03**	1.70E+04	1.32E+03	3.44E+04	7.42E+02	2.37E+04	1.41E+03	3.12E+04	6.19E+02	1.77E+04	1.34E+03	1.61E+04	2.13E+03
23	9.83E+02	8.64E+01	3.41E+03	3.11E+02	3.25E+03	2.96E+02	3.28E+03	2.69E+02	4.93E+03	2.49E+02	1.46E+03	**7.63E+01**	3.42E+03	1.18E+02	1.53E+03	2.18E+02	**9.51E+02**	8.06E+01
24	1.36E+03	**1.22E+02**	4.05E+03	6.36E+02	4.04E+03	6.40E+02	3.75E+03	5.51E+02	8.67E+03	4.15E+02	2.43E+03	1.37E+02	5.81E+03	1.51E+02	1.98E+03	2.25E+02	**1.35E+03**	1.61E+02
25	7.66E+02	7.03E+01	8.41E+02	3.59E+01	8.03E+02	5.92E+01	7.64E+02	7.56E+01	3.26E+04	**1.82E−07**	2.69E+03	2.96E+02	5.76E+04	5.18E+03	**7.20E+02**	5.61E+01	7.50E+02	7.66E+01
26	8.95E+03	1.49E+03	2.89E+04	1.72E+03	2.77E+04	2.21E+03	2.80E+04	1.84E+03	5.63E+04	**1.37E+02**	1.84E+04	1.42E+03	5.25E+04	2.40E+03	1.38E+04	4.25E+03	**8.69E+03**	8.85E+02
27	5.00E+02	2.48E−04	5.39E+03	1.32E+03	5.01E+03	1.45E+03	4.27E+03	1.22E+03	2.57E+03	4.48E+03	5.00E+02	2.40E−04	5.00E+02	**8.76E−05**	5.00E+02	2.19E−04	**5.00E+02**	2.90E−04
28	5.00E+02	2.22E−04	5.95E+02	4.32E+01	6.24E+02	3.99E+01	5.69E+02	3.08E+01	8.22E+03	1.33E+04	5.00E+02	1.98E−04	5.00E+02	**9.56E−05**	5.00E+02	8.74E+00	**5.00E+02**	2.05E+00
29	3.57E+03	6.94E+02	7.46E+03	8.92E+02	6.01E+03	6.87E+02	6.41E+03	6.93E+02	6.95E+06	2.44E+06	6.01E+03	6.93E+02	1.44E+05	6.25E+04	5.45E+03	6.05E+02	**3.08E+03**	**5.95E+02**
30	3.57E+03	1.03E+03	5.30E+07	1.18E+08	1.15E+07	2.88E+06	1.26E+07	3.45E+06	5.87E+10	4.54E+09	6.22E+06	6.44E+06	2.51E+10	3.87E+09	1.46E+06	4.94E+05	**4.46E+02**	**1.40E+02**



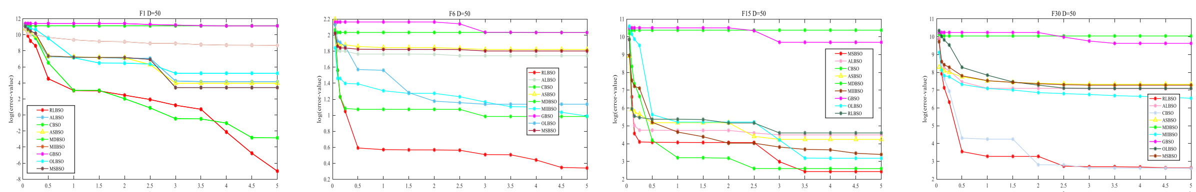
f_1 f_6 f_{15} f_{30}

Fig. 4. Convergence curves of four typical benchmark functions (10D).



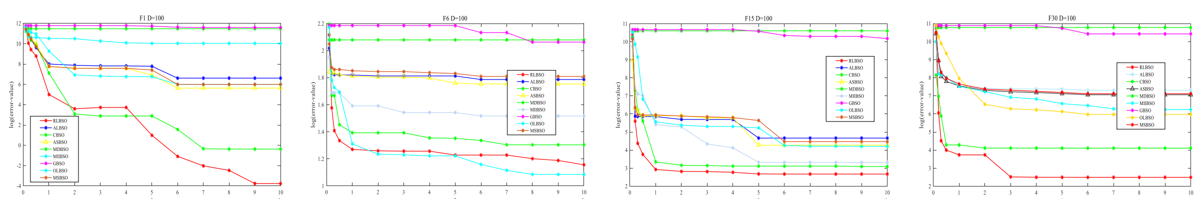
f_1 f_6 f_{15} f_{30}

Fig. 5. Convergence curves of four typical benchmark functions (30D).



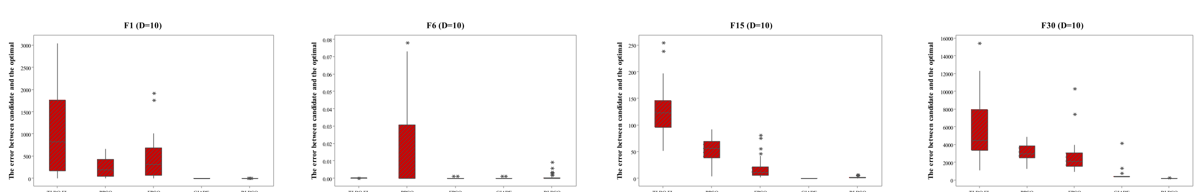
f_1 f_6 f_{15} f_{30}

Fig. 6. Convergence curves of four typical benchmark functions (50D).



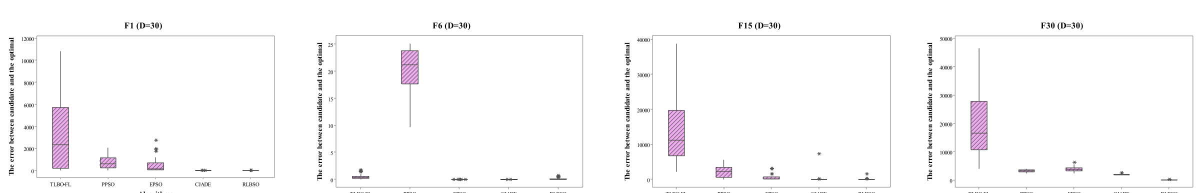
f_1 f_6 f_{15} f_{30}

Fig. 7. Convergence curves of four typical benchmark functions (100D).



f_1 f_6 f_{15} f_{30}

Fig. 8. Box plots of four typical benchmark functions (10D).



f_1 f_6 f_{15} f_{30}

Fig. 9. Box plots of four typical benchmark functions (30D).

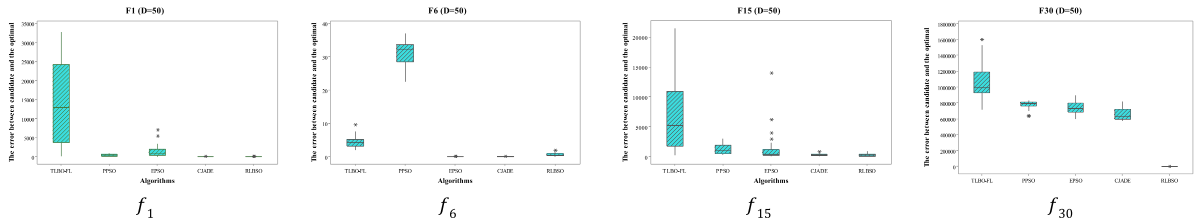


Fig. 10. Box plots of four typical benchmark functions (50D).

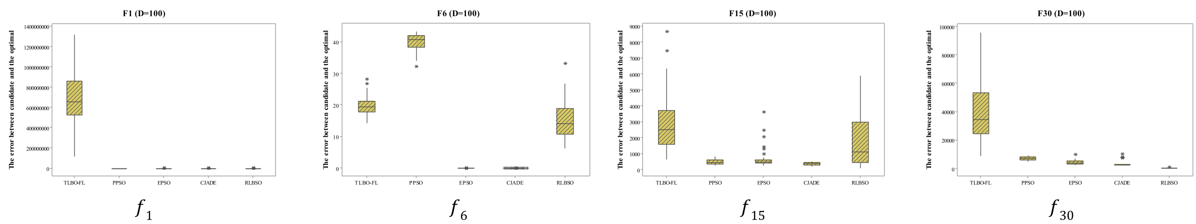


Fig. 11. Box plots of four typical benchmark functions (100D).

Table 6
The results of RLBSO and four state-of-the-art algorithms (10-dimensional benchmark functions).

Fun	TLBO-FL		PPSO		EPSO		CJADE		RLBSO	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	2.03E+03	2.46E+03	2.39E+02	2.01E+02	3.82E+02	4.15E+02	0.00E+00	0.00E+00	**0.00E+00**	**0.00E+00**
3	1.11E−04	7.84E−04	0.00E+00	0.00E+00	3.79E−14	2.94E−14	0.00E+00	0.00E+00	**0.00E+00**	**0.00E+00**
4	3.03E+00	1.17E+00	1.20E+00	9.36E−01	3.13E−01	2.63E−01	**0.00E+00**	**0.00E+00**	3.13E−01	1.08E+00
5	8.75E+00	5.57E+00	1.81E+01	5.10E+00	6.24E+00	2.51E+00	**3.08E+00**	**7.91E−01**	9.31E+00	4.32E+00
6	**0.00E+00**	4.43E−07	2.26E−01	3.09E−01	9.81E−14	6.02E−14	6.02E−14	**5.73E−14**	1.66E−03	5.41E−03
7	2.76E+01	3.97E+00	1.71E+01	2.21E+00	1.87E+01	3.53E+00	**1.37E+01**	**1.11E+00**	1.93E+01	5.05E+00
8	1.23E+01	4.40E+00	9.96E+00	2.37E+00	7.03E+00	2.60E+00	**3.40E+00**	**8.28E−01**	8.80E+00	3.60E+00
9	8.89E−03	6.36E−02	0.00E+00	0.00E+00	2.45E−14	4.72E−14	0.00E+00	0.00E+00	**0.00E+00**	**0.00E+00**
10	9.55E+02	2.14E+02	5.04E+02	1.55E+02	2.39E+02	1.26E+02	**8.52E+01**	**6.21E+01**	5.63E+02	2.43E+02
11	4.13E+00	1.46E+00	1.69E+01	5.32E+00	3.66E+00	2.24E+00	**2.39E+00**	**7.31E−01**	2.73E+00	2.04E+00
12	6.56E+04	5.50E+04	4.56E+03	2.51E+03	8.77E+03	7.57E+03	**9.02E+01**	**8.71E+01**	1.63E+02	1.30E+02
13	2.46E+03	2.21E+03	1.39E+03	1.33E+03	3.86E+02	6.64E+02	**3.85E+00**	**2.57E+00**	7.41E+00	2.66E+00
14	6.73E+01	1.83E+01	3.74E+01	1.18E+01	2.38E+01	1.46E+01	**5.02E−01**	**4.09E−01**	1.73E+01	8.59E+00
15	1.27E+02	4.35E+01	5.33E+01	2.27E+01	2.01E+01	2.21E+01	**4.17E−01**	**1.27E−01**	1.91E+00	1.40E+00
16	8.91E+00	2.19E+01	8.30E+01	7.25E+01	3.29E+00	1.67E+01	**1.08E+00**	**3.41E−01**	7.99E+01	8.41E+01
17	3.83E+01	7.83E+00	2.47E+01	7.43E+00	7.92E+00	8.11E+00	**4.06E−01**	**2.99E−01**	1.53E+01	1.44E+01
18	6.15E+03	5.63E+03	8.78E+02	7.15E+02	1.56E+03	1.53E+03	**3.09E−01**	**4.00E−01**	1.69E+01	7.50E+00
19	6.06E+01	3.18E+01	2.26E+01	1.57E+01	1.75E+01	5.20E+01	**3.58E−02**	**1.65E−02**	1.08E+00	8.09E−01
20	1.46E+01	9.45E+00	2.78E+01	8.96E+00	5.45E−01	5.68E−01	**2.72E−10**	**8.38E−10**	2.97E+01	4.20E+01
21	1.42E+02	5.17E+01	**1.05E+02**	**2.16E+01**	1.37E+02	5.24E+01	1.52E+02	4.76E+01	1.09E+02	3.02E+01
22	9.34E+01	2.31E+01	9.67E+01	**1.68E+01**	**8.08E+01**	2.95E+01	9.83E+01	1.24E+01	1.78E+02	2.38E+02
23	3.08E+02	3.84E+00	3.42E+02	1.06E+01	3.09E+02	2.93E+00	**3.05E+02**	**1.30E+00**	3.13E+02	4.38E+00
24	3.10E+02	6.90E+01	2.27E+02	1.35E+02	**1.73E+02**	1.00E+02	2.65E+02	9.71E+01	3.45E+02	**6.32E+00**
25	4.26E+02	2.26E+01	4.05E+02	**1.46E+01**	**3.84E+02**	6.93E+01	4.17E+02	2.29E+01	4.20E+02	2.32E+01
26	3.29E+02	4.62E+01	2.68E+02	7.66E+01	**2.58E+02**	6.90E+01	3.00E+02	**0.00E+00**	3.50E+02	1.26E+02
27	3.89E+02	3.28E+00	4.27E+02	1.35E+01	3.93E+02	2.53E+00	3.89E+02	**7.04E−01**	**3.80E+02**	3.02E+01
28	4.47E+02	1.61E+02	**2.95E+02**	4.20E+01	3.05E+02	8.07E+01	3.79E+02	1.32E+02	4.64E+02	**3.31E+01**
29	2.74E+02	1.38E+01	2.78E+02	1.35E+01	2.62E+02	1.37E+01	**2.44E+02**	**6.17E+00**	2.56E+02	1.93E+01
30	2.79E+05	4.92E+05	2.99E+03	8.99E+02	2.88E+03	1.75E+03	6.47E+04	2.22E+05	**2.08E+02**	**1.58E+01**

4.4. Effects of RLBSO components

The RLBSO includes three operations, which are mutation strategies, strategy selection method based on learning mechanism, and crossover operations. Three sets of experiments are designed to verify the effectiveness of the three operations. A random method is used to replace the learning-based method to select the four strategies in the first group. The mutation strategies in the basic BSO are used to replace the mutation strategies in the RLBSO in the second group. The crossover operation is removed in the third group. The three groups of experiments are tested on 29 functions in the CEC 2017 test suite and compared with RLBSO respectively. RLBSO1, RLBSO2, and RLBSO3 are used to name the three groups of experiments. The results of the three groups of experiments are shown in Table 14. The RLBSO performs better on 23 functions than the other three experiments. Four types of functions in the test suite are represented by four dot plots separately, from Fig. 13, where the horizontal axis represents the functions and the vertical axis represents the natural logarithm of the mean value of each algorithm. Intuitively, the three operations improve the performance of BSO algorithm.

Table 7
The results of RLBSO and four state-of-the-art algorithms (30-dimensional benchmark functions).

Fun	TLBO-FL		PPSO		EPSO		CJADE		RLBSO	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	3.51E+03	3.61E+03	7.50E+02	6.07E+02	4.57E+02	5.98E+02	**1.48E−14**	**3.98E−15**	1.63E−05	1.16E−04
3	3.00E+03	1.08E+03	1.13E+00	4.83E−01	**4.68E−08**	**1.27E−07**	4.52E+03	1.26E+04	3.35E+03	5.51E+03
4	9.01E+01	**2.37E+01**	4.39E+01	3.19E+01	3.17E+01	3.20E+01	3.96E+01	2.90E+01	**2.44E+01**	2.41E+01
5	3.95E+01	2.08E+01	1.12E+02	1.33E+01	5.20E+01	1.19E+01	**2.59E+01**	**3.85E+00**	5.62E+01	1.59E+01
6	4.88E−01	4.24E−01	2.03E+01	4.15E+00	1.93E−08	1.01E−07	**1.18E−13**	**2.23E−14**	1.16E−01	1.65E−01
7	1.39E+02	4.76E+01	1.35E+02	1.63E+01	9.45E+01	1.41E+01	**5.49E+01**	**4.05E+00**	8.30E+01	1.44E+01
8	3.67E+01	1.85E+01	8.10E+01	1.04E+01	5.61E+01	1.55E+01	**2.60E+01**	**3.78E+00**	5.33E+01	1.68E+01
9	3.45E+01	2.71E+01	1.36E+03	2.83E+02	7.61E+01	4.39E+01	**1.76E−03**	**1.25E−02**	1.32E+01	1.59E+01
10	6.69E+03	2.78E+02	3.13E+03	3.46E+02	2.23E+03	3.34E+02	**1.92E+03**	**2.54E+02**	2.97E+03	5.52E+02
11	8.16E+01	4.14E+01	8.43E+01	1.85E+01	5.86E+01	2.87E+01	3.16E+01	2.57E+01	**2.18E+01**	**1.18E+01**
12	5.75E+04	8.99E+04	2.77E+04	8.55E+03	2.86E+04	1.37E+04	**1.37E+03**	**9.42E+02**	3.05E+04	2.96E+04
13	2.02E+04	1.79E+04	3.19E+03	2.88E+03	1.09E+03	1.07E+03	**4.80E+01**	**3.27E+01**	2.20E+02	2.76E+02
14	7.10E+03	5.85E+03	2.32E+03	1.52E+03	5.95E+03	8.67E+03	2.73E+03	5.19E+03	**4.50E+01**	**1.43E+01**
15	2.17E+04	2.28E+04	2.13E+03	1.63E+03	5.47E+02	6.97E+02	1.79E+02	1.02E+03	**7.30E+01**	**2.19E+02**
16	4.92E+02	3.54E+02	8.46E+02	**1.54E+02**	6.38E+02	2.13E+02	**4.57E+02**	1.59E+02	9.16E+02	2.92E+02
17	1.42E+02	6.59E+01	3.31E+02	1.13E+02	1.99E+02	1.04E+02	**7.42E+01**	**2.67E+01**	4.33E+02	2.14E+02
18	3.67E+05	1.67E+05	7.01E+04	3.06E+04	1.04E+05	8.99E+04	**6.72E+03**	3.52E+04	9.50E+03	**8.67E+03**
19	1.07E+04	1.10E+04	1.71E+03	1.69E+03	8.23E+02	1.46E+03	3.05E+02	2.02E+03	**1.59E+01**	**6.05E+00**
20	2.21E+02	1.25E+02	3.48E+02	9.16E+01	2.18E+02	1.27E+02	**1.14E+02**	**5.43E+01**	4.43E+02	1.85E+02
21	2.34E+02	1.16E+01	3.06E+02	3.30E+01	2.54E+02	3.09E+01	**2.26E+02**	**3.97E+00**	2.59E+02	1.55E+01
22	1.01E+02	1.94E+00	1.00E+02	5.05E−07	1.42E+02	2.99E+02	**1.00E+02**	**1.00E−13**	3.38E+03	6.06E+02
23	3.96E+02	1.62E+01	6.79E+02	3.79E+01	4.09E+02	1.44E+01	**3.73E+02**	**5.24E+00**	4.12E+02	1.55E+01
24	4.69E+02	1.63E+01	7.39E+02	4.58E+01	4.81E+02	5.76E+01	**4.42E+02**	**4.76E+00**	4.82E+02	1.38E+01
25	4.02E+02	1.77E+01	3.85E+02	1.78E+00	3.87E+02	1.61E+00	3.87E+02	**5.35E−01**	**3.79E+02**	7.39E−01
26	1.42E+03	4.69E+02	2.04E+03	1.73E+03	7.23E+02	7.03E+02	**1.20E+03**	**8.20E+01**	1.42E+03	2.54E+02
27	5.32E+02	2.07E+01	7.08E+02	5.42E+01	5.16E+02	8.72E+00	5.04E+02	8.10E+00	**5.00E+02**	**2.10E−04**
28	4.30E+02	2.67E+01	**3.27E+02**	3.17E+01	3.31E+02	5.10E+01	3.34E+02	5.50E+01	4.98E+02	**3.46E+00**
29	6.15E+02	9.09E+01	7.81E+02	1.22E+02	6.12E+02	8.88E+01	**4.78E+02**	**2.32E+01**	5.58E+02	1.99E+02
30	2.58E+04	2.78E+04	3.32E+03	3.89E+02	3.99E+03	7.67E+02	2.12E+03	1.55E+02	**2.20E+02**	**1.26E+01**

Table 8
The results of RLBSO and four state-of-the-art algorithms (50-dimensional benchmark functions).

Fun	TLBO-FL		PPSO		EPSO		CJADE		RLBSO	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	6.06E+05	2.21E+06	3.89E+02	2.95E+02	1.66E+03	1.89E+03	**7.50E−14**	**6.91E−14**	6.77E−01	4.50E+00
3	2.57E+04	4.88E+03	8.65E+02	1.86E+02	**9.92E−01**	**1.34E+00**	2.35E+04	3.91E+04	1.37E+05	5.95E+04
4	1.90E+02	4.58E+01	9.13E+01	3.62E+01	6.86E+01	3.82E+01	4.34E+01	4.75E+01	**3.85E+01**	**2.08E+01**
5	9.67E+01	1.73E+01	2.01E+02	1.37E+01	1.26E+02	2.60E+01	**5.32E+01**	**7.71E+00**	1.20E+02	3.27E+01
6	4.51E+00	1.70E+00	3.18E+01	3.91E+00	3.58E−05	1.47E−04	**1.16E−13**	**1.59E−14**	1.66E+00	2.10E+00
7	1.74E+02	4.38E+01	2.78E+02	3.42E+01	2.00E+02	3.13E+01	**9.98E+01**	**7.28E+00**	1.67E+02	3.12E+01
8	9.33E+01	1.58E+01	1.99E+02	1.52E+01	1.30E+02	2.21E+01	**5.34E+01**	**6.22E+00**	1.13E+02	2.88E+01
9	1.30E+03	1.05E+03	6.06E+03	7.28E+02	9.44E+02	6.06E+02	**1.17E+00**	**1.22E+00**	3.83E+02	5.04E+02
10	1.27E+04	3.97E+02	5.20E+03	5.51E+02	4.13E+03	6.05E+02	**3.77E+03**	**2.39E+02**	5.86E+03	7.90E+02
11	1.69E+02	4.79E+01	1.28E+02	**1.45E+01**	1.40E+02	4.61E+01	1.39E+02	3.46E+01	**5.44E+01**	2.21E+01
12	9.16E+05	8.94E+05	5.52E+05	2.03E+05	3.97E+05	2.16E+05	**6.02E+03**	**3.22E+03**	1.92E+05	1.33E+05
13	8.01E+03	5.15E+03	8.47E+02	5.63E+02	1.10E+03	7.56E+02	**3.93E+02**	**3.53E+02**	6.09E+02	9.84E+02
14	8.63E+04	4.96E+04	1.95E+04	**1.00E+04**	1.78E+04	1.26E+04	**6.75E+03**	2.71E+04	8.52E+03	1.70E+04
15	6.88E+03	5.91E+03	1.19E+03	7.80E+02	1.25E+03	2.41E+03	3.01E+02	**1.51E+02**	**2.97E+02**	3.81E+02
16	8.46E+02	3.05E+02	1.24E+03	2.28E+02	1.16E+03	3.31E+02	**8.38E+02**	**2.07E+02**	1.71E+03	3.43E+02
17	8.53E+02	4.12E+02	1.03E+03	1.54E+02	8.44E+02	1.88E+02	**6.16E+02**	**1.17E+02**	1.20E+03	3.11E+02
18	1.15E+06	5.13E+05	2.09E+05	8.67E+04	1.06E+05	1.08E+05	**1.31E+04**	9.27E+04	1.07E+05	**7.74E+04**
19	1.45E+04	9.40E+03	8.67E+03	3.91E+03	1.60E+03	2.04E+03	**1.41E+02**	**4.95E+01**	7.20E+02	4.06E+03
20	1.04E+03	3.88E+02	7.70E+02	1.92E+02	7.73E+02	2.08E+02	**4.44E+02**	**1.16E+02**	9.07E+02	2.50E+02
21	2.81E+02	1.52E+01	4.33E+02	2.14E+01	3.34E+02	2.12E+01	**2.57E+02**	**8.54E+00**	3.23E+02	3.03E+01
22	6.56E+03	6.40E+03	5.97E+03	**9.89E+02**	3.99E+03	2.02E+03	**3.58E+03**	1.52E+03	6.86E+03	1.00E+03
23	5.66E+02	3.40E+01	1.07E+03	7.09E+01	5.95E+02	3.26E+01	**4.80E+02**	**1.00E+01**	5.32E+02	4.40E+01
24	6.75E+02	4.75E+01	1.08E+03	7.07E+01	6.95E+02	5.95E+01	**5.42E+02**	**1.08E+01**	6.04E+02	2.80E+01
25	6.16E+02	2.68E+01	5.42E+02	2.77E+01	5.30E+02	2.56E+01	5.23E+02	3.53E+01	**4.37E+02**	**1.41E+01**
26	2.93E+03	6.02E+02	5.45E+03	2.59E+03	2.04E+03	1.22E+03	**1.64E+03**	**1.19E+02**	2.12E+03	6.13E+02
27	8.68E+02	1.76E+02	1.46E+03	1.70E+02	6.76E+02	5.06E+01	5.53E+02	2.79E+01	**5.00E+02**	**1.92E−04**
28	6.12E+02	4.22E+01	**4.89E+02**	1.79E+01	**4.89E+02**	1.98E+01	4.93E+02	2.61E+01	5.00E+02	**2.38E−04**
29	1.02E+03	2.48E+02	1.53E+03	2.09E+02	9.00E+02	2.04E+02	**4.97E+02**	**7.66E+01**	1.16E+03	3.09E+02
30	1.16E+06	3.15E+05	7.81E+05	4.82E+04	7.40E+05	7.93E+04	6.61E+05	7.51E+04	**3.07E+02**	**5.83E+01**

Sign rank test is used to analyze the significant difference between the mean errors of RLBSO and the other three algorithms, respectively. The results are shown in Table 15. R^+ is the sum of the functions that RLBSO is superior to the current comparison algorithm. R^- is the sum of the functions that the current comparison algorithm is superior to RLBSO. p-values are less than $\alpha = 0.05$ and $\alpha = 0.1$, which means that RLBSO have significant differences with RLBSO1, RLBSO2, and RLBSO3.

Table 9
The results of RLBSO and four state-of-the-art algorithms (100-dimensional benchmark functions).

Fun	TLBO-FL		PPSO		EPSO		CJADE		RLBSO	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	8.59E+08	5.17E+08	1.08E+04	3.31E+03	6.42E+02	6.66E+02	**4.12E−10**	**8.54E−10**	6.93E+02	2.19E+03
3	1.77E+05	1.97E+04	7.25E+04	6.60E+03	**5.97E+03**	**3.18E+03**	1.42E+05	1.66E+05	1.20E+05	5.16E+05
4	6.99E+02	1.20E+02	2.41E+02	**3.12E+01**	2.06E+02	4.64E+01	**2.49E+01**	4.65E+01	1.41E+02	4.59E+01
5	3.41E+02	4.51E+01	5.20E+02	2.31E+01	4.43E+02	6.62E+01	**1.53E+02**	**2.18E+01**	3.08E+02	6.50E+01
6	1.96E+01	3.01E+00	4.01E+01	2.34E+00	**1.14E−04**	**2.98E−04**	7.67E−04	2.39E−03	1.54E+01	5.97E+00
7	7.80E+02	9.56E+01	7.38E+02	8.90E+01	6.59E+02	6.28E+01	**2.77E+02**	**2.23E+01**	5.31E+02	1.20E+02
8	3.71E+02	5.59E+01	5.82E+02	3.07E+01	4.31E+02	5.94E+01	**1.45E+02**	**1.86E+01**	3.76E+02	7.84E+01
9	1.97E+04	4.93E+03	1.48E+04	8.12E+02	1.04E+04	3.98E+03	**1.03E+02**	**7.05E+01**	7.06E+03	3.84E+03
10	2.90E+04	5.93E+02	1.14E+04	8.94E+02	1.09E+04	8.28E+02	**1.02E+04**	**5.45E+02**	1.46E+04	1.69E+03
11	1.05E+03	1.98E+02	8.96E+02	**7.10E+01**	9.55E+02	1.53E+02	2.99E+03	3.44E+03	**5.89E+02**	2.57E+02
12	2.72E+07	1.52E+07	4.12E+06	7.35E+05	8.14E+05	3.91E+05	**1.85E+04**	**7.42E+03**	1.38E+06	9.48E+05
13	1.40E+04	6.47E+03	1.53E+03	7.24E+02	**1.07E+03**	**4.52E+02**	4.27E+03	4.66E+03	4.37E+03	6.92E+03
14	1.92E+06	7.10E+05	3.14E+05	8.94E+04	6.96E+04	3.53E+04	**5.53E+02**	**2.49E+02**	7.25E+04	4.49E+04
15	3.73E+03	3.58E+03	4.69E+02	1.58E+02	6.86E+02	5.81E+02	**3.59E+02**	**8.53E+01**	3.53E+03	6.39E+03
16	**2.57E+03**	5.79E+02	3.18E+03	**3.25E+02**	3.34E+03	4.59E+02	2.58E+03	3.38E+02	3.93E+03	6.38E+02
17	2.19E+03	4.74E+02	2.63E+03	3.34E+02	2.60E+03	2.67E+02	**1.90E+03**	**2.55E+02**	2.97E+03	5.82E+02
18	4.59E+06	1.64E+06	6.89E+05	1.76E+05	1.81E+05	6.94E+04	**1.41E+03**	**7.34E+02**	2.61E+05	1.74E+05
19	4.15E+03	5.31E+03	5.43E+02	**3.33E+02**	4.74E+02	6.02E+02	**1.48E+03**	2.18E+03	2.01E+03	6.43E+03
20	4.23E+03	3.18E+02	2.35E+03	**2.58E+02**	2.62E+03	2.76E+02	**1.88E+03**	2.83E+02	2.72E+03	4.94E+02
21	5.99E+02	4.53E+01	1.06E+03	5.52E+01	7.03E+02	5.28E+01	**3.63E+02**	**1.64E+01**	5.77E+02	9.67E+01
22	2.95E+04	4.23E+03	1.37E+04	8.95E+02	1.24E+04	8.24E+02	**1.11E+04**	2.27E+03	1.61E+04	**2.13E+03**
23	1.17E+03	1.11E+02	2.07E+03	8.16E+01	8.25E+02	2.70E+01	**6.51E+02**	**1.46E+01**	9.51E+02	8.06E+01
24	2.02E+03	2.61E+02	1.89E+03	1.04E+02	1.34E+03	4.04E+01	**1.03E+03**	**2.42E+01**	1.35E+03	1.61E+02
25	1.32E+03	1.09E+02	7.59E+02	**3.37E+01**	7.53E+02	5.82E+01	**7.38E+02**	5.52E+01	7.50E+02	7.66E+01
26	1.09E+04	1.57E+03	1.56E+04	5.19E+03	7.82E+03	1.94E+03	**4.62E+03**	**2.11E+02**	8.69E+03	8.85E+02
27	1.17E+03	1.47E+02	1.34E+03	9.82E+01	8.12E+02	4.02E+01	7.38E+02	3.51E+01	**5.00E+02**	**2.90E−04**
28	1.48E+03	2.70E+02	5.87E+02	1.49E+01	5.51E+02	2.74E+01	**4.83E+02**	9.38E+01	5.00E+02	**2.05E+00**
29	3.44E+03	4.97E+02	3.73E+03	2.98E+02	3.40E+03	4.69E+02	**2.19E+03**	**2.66E+02**	3.08E+03	5.95E+02
30	6.18E+04	6.36E+04	7.17E+03	1.17E+03	4.77E+03	1.73E+03	3.37E+03	1.72E+03	**4.46E+02**	**1.40E+02**

Table 10
Rankings obtained through Wilcoxon test (10D).

Dimension	RLBSO vs	R^+	R^-	Z	p-value	$\alpha = 0.05$	$\alpha = 0.1$
10D	MSBSO	171.50	179.5	−1.02E−01	9.19E−01	no	no
	ALBSO	19.00	387.00	−4.19E+00	2.79E−05	yes	yes
	CBSO	17.00	389.00	−4.24E+00	2.28E−05	yes	yes
	ASBSO	25.00	410.00	−4.16E+00	3.15E−05	yes	yes
	MDBSO	0.00	435.00	−4.70E+00	2.56E−06	yes	yes
	MIIBSO	75.00	360.00	−3.08E+00	2.06E−03	yes	yes
	GBSO	0.00	435.00	−4.70E+00	2.56E−06	yes	yes
	OLBSO	20.00	415.00	−4.27E+00	1.95E−05	yes	yes

Table 11
Rankings obtained through Wilcoxon test (30D).

Dimension	RLBSO vs	R^+	R^-	Z	p-value	$\alpha = 0.05$	$\alpha = 0.1$
30D	MSBSO	164.50	270.50	−1.15E+00	2.52E−01	no	no
	ALBSO	4.00	431.00	−4.62E+00	4.00E−06	yes	yes
	CBSO	26.00	409.00	−4.14E+00	3.50E−05	yes	yes
	ASBSO	25.00	410.00	−4.16E+00	3.10E−05	yes	yes
	MDBSO	0.00	0.00	−4.70E+00	3.00E−06	yes	yes
	MIIBSO	64.00	342.00	−3.17E+00	1.55E−03	yes	yes
	GBSO	0.00	406.00	−4.62E+00	4.00E−06	yes	yes
	OLBSO	21.00	414.00	2.10E−05	4.00E−06	yes	yes

Table 12
Rankings obtained through Wilcoxon test (50D).

Dimension	RLBSO vs	R^+	R^-	Z	p-value	$\alpha = 0.05$	$\alpha = 0.1$
50D	MSBSO	61.00	317.00	−3.08E+00	2.10E−03	yes	yes
	ALBSO	23.00	412.00	−4.21E+00	2.60E−05	yes	yes
	CBSO	25.00	410.00	−4.16E+00	3.10E−05	yes	yes
	ASBSO	26.00	409.00	−4.14E+00	3.50E−05	yes	yes
	MDBSO	0.00	406.00	−4.62E+00	4.00E−06	yes	yes
	MIIBSO	35.00	343.00	−3.70E+00	2.16E−04	yes	yes
	GBSO	0.00	378.00	−4.54E+00	6.00E−06	yes	yes
	OLBSO	27.00	408.00	−4.12E+00	3.80E−05	yes	yes

Table 13
Rankings obtained through Wilcoxon test (100D).

Dimension	RLBSO vs	R^+	R^-	Z	p-value	$\alpha = 0.05$	$\alpha = 0.1$
100D	MSBSO	90.00	288.00	−2.38E+00	1.74E−02	yes	yes
	ALBSO	22.00	413.00	−4.23E+00	2.40E−05	yes	yes
	CBSO	23.00	412.00	−4.21E+00	2.60E−05	yes	yes
	ASBSO	23.00	412.00	−4.21E+00	2.60E−05	yes	yes
	MDBSO	0.00	435.00	−4.70E+00	3.00E−06	yes	yes
	MIIBSO	0.00	378.00	−4.54E+00	6.00E−06	yes	yes
	GBSO	0.00	378.00	−4.54E+00	6.00E−06	yes	yes
	OLBSO	24.00	354.00	−3.96E+00	6.00E−06	yes	yes

Table 14
Results of 29 functions.

Fun	RLBSO		RLBSO1		RLBSO2		RLBSO3	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	**0.00E+00**	**0.00E+00**	1.30E−08	4.10E−08	1.74E+10	6.11E+09	7.82E−14	9.22E−14
3	**0.00E+00**	**0.00E+00**	5.68E−15	1.80E−14	7.71E+03	1.87E+03	2.84E−14	3.00E−14
4	**3.13E−01**	1.08E+00	2.66E−06	**8.22E−06**	2.72E+03	1.03E+03	3.99E−01	1.26E+00
5	**9.31E+00**	**4.32E+00**	1.07E+01	4.48E+00	8.43E+01	2.87E+01	2.53E+01	1.13E+01
6	**1.66E−03**	5.41E−03	2.19E−03	**4.01E−03**	4.28E+01	1.57E+01	4.01E+00	3.97E+00
7	**1.93E+01**	**5.05E+00**	1.94E+01	5.82E+00	1.29E+02	4.49E+01	3.30E+01	9.29E+00
8	**8.80E+00**	**3.60E+00**	9.55E+00	5.94E+00	3.37E+01	1.11E+01	2.18E+01	1.03E+01
9	**0.00E+00**	**0.00E+00**	4.54E−02	1.44E−01	7.97E+02	2.95E+02	8.42E+01	8.24E+01
10	5.63E+02	2.43E+02	**3.86E+02**	**9.72E+01**	9.46E+02	2.58E+02	8.82E+02	2.18E+02
11	**2.73E+00**	**2.04E+00**	3.36E+00	2.12E+00	8.78E+03	8.82E+03	2.30E+01	2.15E+01
12	**1.63E+02**	1.30E+02	2.19E+02	**8.56E+01**	2.57E+09	9.35E+08	3.60E+02	2.12E+02
13	**7.41E+00**	**2.66E+00**	8.82E+00	3.87E+00	2.57E+09	9.09E+08	2.02E+01	1.87E+01
14	**1.73E+01**	8.59E+00	2.19E+01	**1.78E+00**	3.58E+08	1.48E+08	2.27E+01	4.47E+00
15	1.91E+00	**1.40E+00**	**1.53E+00**	1.41E+00	2.72E+04	1.04E+04	6.09E+00	3.88E+00
16	7.99E+01	8.11E+01	**7.51E+01**	8.16E+01	3.73E+02	1.45E+02	1.92E+02	**6.82E+01**
17	**1.53E+01**	**1.44E+01**	2.32E+01	1.95E+01	9.22E+01	7.52E+01	3.44E+01	2.42E+01
18	**1.69E+01**	7.50E+00	2.10E+01	**7.70E−01**	3.42E+03	1.73E+03	2.26E+01	5.08E+00
19	**1.08E+00**	**6.09E−01**	1.68E+00	6.14E−01	1.11E+10	3.91E+09	3.11E+00	2.14E+00
20	2.97E+01	4.20E+01	**2.11E+01**	**1.05E+01**	2.43E+02	7.77E+01	9.26E+01	5.76E+01
21	**1.09E+02**	**3.02E+01**	1.91E+02	4.83E+01	1.76E+02	2.86E+01	1.12E+02	3.64E+01
22	1.78E+02	2.38E+02	**1.01E+02**	**3.77E−01**	1.13E+03	3.83E+02	3.25E+02	4.78E+02
23	**3.13E+02**	**4.38E+00**	3.15E+02	5.58E+00	3.55E+02	1.67E+01	3.22E+02	9.36E+00
24	**3.45E+02**	**6.32E+00**	3.45E+02	7.22E+00	5.85E+02	9.05E+01	3.64E+02	8.51E+00
25	**4.20E+02**	2.32E+01	4.20E+02	**3.34E−01**	1.05E+03	2.37E+02	4.17E+02	2.42E+01
26	**3.50E+02**	**1.26E+02**	3.52E+02	1.34E+02	1.52E+03	3.81E+02	4.42E+02	2.12E+02
27	3.80E+02	3.02E+01	3.81E+02	**1.30E+00**	6.51E+02	9.64E+01	**3.76E+02**	3.80E+00
28	**4.64E+02**	3.31E+01	4.73E+02	**1.61E−13**	1.36E+03	2.50E+02	4.84E+02	1.42E+01
29	**2.56E+02**	**1.63E+01**	2.57E+02	1.66E+01	1.68E+03	8.48E+02	3.14E+02	5.77E+01
30	**2.08E+02**	**1.38E+01**	2.09E+02	1.41E+00	4.58E+07	2.94E+07	2.35E+02	4.63E+01

4.5. Result analysis

The results measured in the CEC2017 test suite and analyzed by statistics indicate that the RL significantly improves the performance of BSO. RLBSO is superior to the existing BSO variants and most advanced algorithms in terms of solution accuracy, convergence speed and algorithm robustness in most test cases.

The advantage of RLBSO lies in the introduction of reinforcement learning, which enables the algorithm to choose the appropriate behavior according to the current state of the population and the change of the environment, so as to maximize the return. Q-learning determines the evolutionary direction of the current population by using the behavioral returns of the previous generation of the population, which avoids the blind search of algorithm in the solution space. It can be seen from the mean and standard tables that the performance of RLBSO in solving hybrid functions and composition functions is better than that of single modal and simple multimodal functions. The reason is that RL theoretically manage mutation strategies with different characteristics, and provide excellent global and local search capabilities.

The four mutation strategies play different advantages respectively. The guidance of the global optimal individual enhances the ability of finding the global optimal solution, which improves the convergence speed of the algorithm. Elite individuals are superior individuals in the population, leading individuals to jump from one terrain to another with a smaller span. The buffer effectively prevents RLBSO missing the global optimal solution and falling into local optimum in the process of evolution. The cluster centers lead the individuals to conduct distributed search in the solution space, which improves the solving speed of the algorithm. It can be seen from the convergence graph that some functions, such as f_1, decreases in the later iteration. This is because the historical population effectively increases the diversity of the population, which prevents the algorithm from stagnating in the later iteration, and enables the algorithm to further find a better solution.

Table 15
Rankings obtained through Sign test.

RLBSO vs	R^+	R^-	Z	p-value	$\alpha = 0.05$	$\alpha = 0.1$
RLBSO1	21	6	−2.69E+00	7.05E−03	yes	yes
RLBSO2	29	0	−5.20E+00	2.00E−07	yes	yes
RLBSO3	27	2	−4.46E+00	8.00E−06	yes	yes

The performance of individuals in different dimensions is different, since the fitness landscape of the functions in CEC2017 test suite is rugged. Therefore, utilizing all the dimensions of one individual to update another is not enough. The crossover operation only improves part of the dimensions of an individual, which helps to further improve the ability of the algorithm to solve the variable non-separable problems.

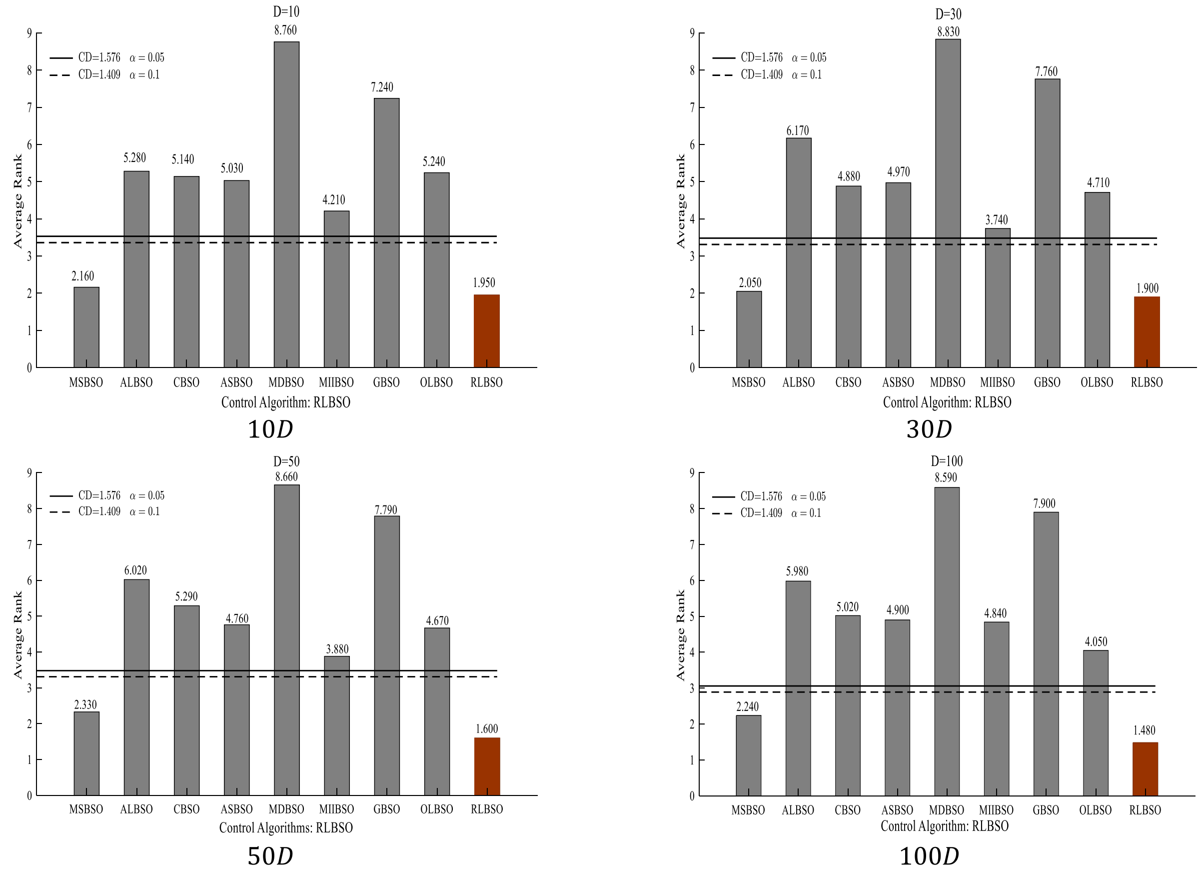


Fig. 12. Rankings obtained through Friedman test.

5. RLBSO for engineering problems

5.1. Notion

The notations and meanings used in this problem are listed below.

i	The generating unit $i.i = \{1, 2, \ldots, N_G\}$
P_{it}	The real power output (in MW) of generator i corresponding to time period t.
N_G	The number of online generating units to be dispatched.
T	The total time period of dispatch.
PD	The total system loads.
PL	The total system losses.
P_i^{min}	The lower bounds for power outputs of the generating unit i in MW.
$P_i^{\max}$	The upper bounds for power outputs of the generating unit i in MW.
P_i^{t-1}	The power generation of unit i at previous hour.
UR_i	The upper ramp rate limits.
DR_i	The lower ramp rate limits.
n	The number of hours.
N	The number of units.

5.2. Problem definition

The Dynamic Economic Dispatch (DED) problem [60] has the characteristics of the hourly scheduling problem, in which the power demand varies with the hour and a 24-hour generation plan needs to be determined. The objective and constraints are listed below.

Minimize

$$F_c = \sum_{k=1}^{T}\sum_{i=1}^{N_G} F_{ih}(P_{ih}) \tag{21}$$

$$F_{ih}(P_{ih}) = a_i P_{it}^2 + b_i P_{it} + c_i + |e_i sin(f_{it}(P_{it}^{\min} - P_{it}))| \tag{22}$$

Subject to

$$\sum_{i=1}^{N_G} P_{it} = P_{Dt} + P_{Lt} \tag{23}$$

$$P_{Lt} = \sum_{i=1}^{N_G}\sum_{j=1}^{N_G} P_{it} B_{ij} P_{jt} \tag{24}$$

$$P_i^{\min} \le P_{it} \le P_i^{\max} \tag{25}$$

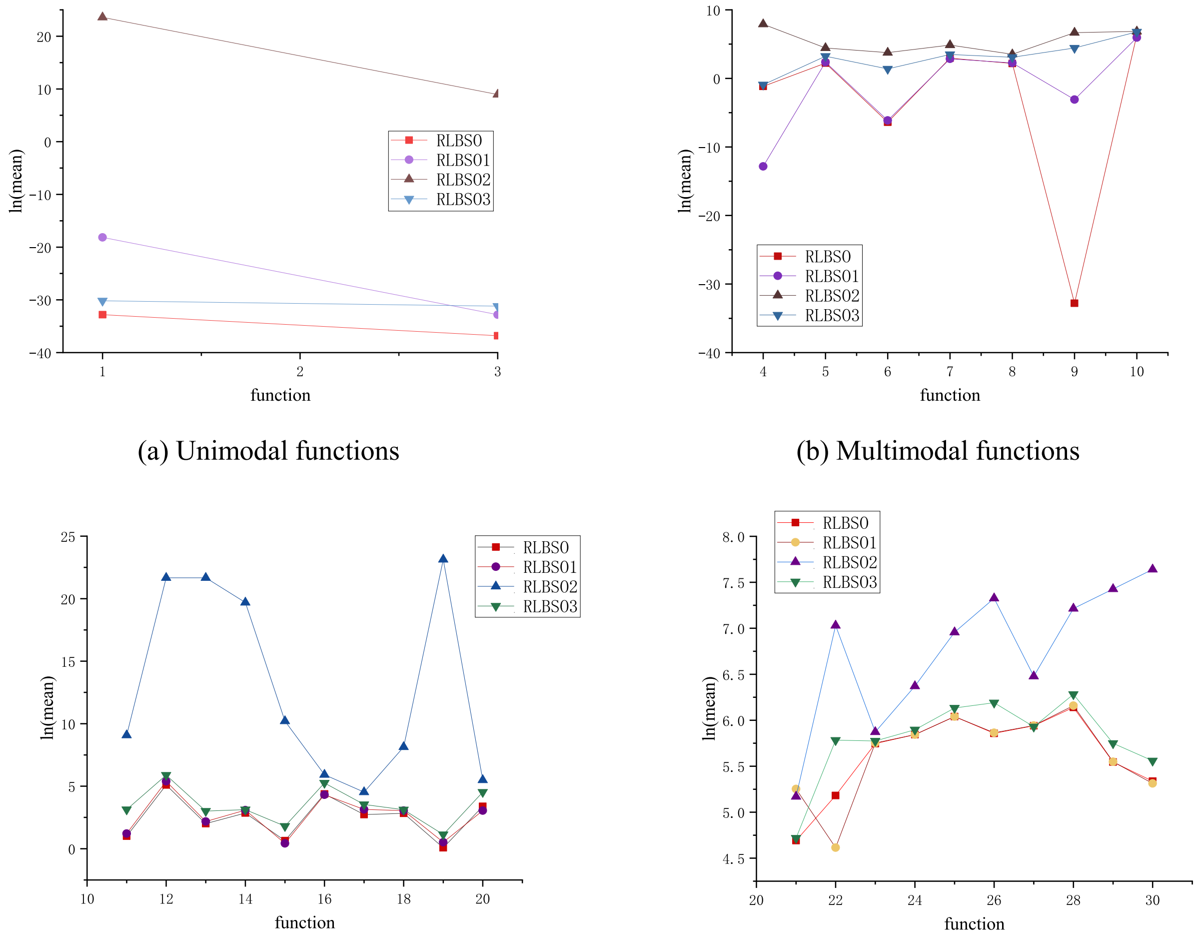
$$max(P_i^{\min}, UR_i - P_i) \le P_i \le min(P_i^{\max}, P_i^{t-1} - DR_i) \tag{26}$$

$$f_k = \sum_{t=1}^{n}\sum_{i=1}^{N} F_i(P_{it}) + \lambda_1\Big(\sum_{t=1}^{n}\sum_{i=1}^{N} P_{it} - P_{Dt}\Big)^2 + \lambda_r\Big(\sum_{t=1}^{n}\sum_{i=1}^{N} P_{it} - P_{r\,\lim}\Big)^2 \tag{27}$$

$$P_{rlim} = \begin{cases} P_{i(t-1)} - DR_i, P_{it} < P_{i(t-1)} - DR_i \\ P_{i(t-1)} + UR_i, P_{it} > P_{i(t-1)} + UR_i \end{cases} \tag{28}$$

Eq. (21) is the objective function corresponding to the production cost, where $F_{ih}(P_{ih})$ is defined in Eq. (22). Eq. (22) is the cost function of the unit with valve point loading effect, where a_i, b_i and c_i are expressions of the cost function corresponding to the cost coefficient of the ith unit, and e_i and f_i are the cost coefficients corresponding to the valve point loading effect.

Because of the valve point loading, the solution may be trapped in the local minimum, which increases the nonlinearity of the



(a) Unimodal functions (b) Multimodal functions

(c) Hybrid functions (d) Composition functions

Fig. 13. Point plot of 29 functions.

system. The limited DEDP problem is subject to a variety of constraints that depend on assumptions and practical implications. Eq. (23) is the power balance constraint, where P_{Lt} is obtained using B-coefficients shown in Eq. (24). The constraint is based on the balance principle between the total power generation of the system and the total load (PD) and total loss (PL) of the system. Eq. (25) represents the constraint of the generator. The output power of each generating unit has a lower bound and an upper bound, and it is between the two boundaries. In many early studies, an unrealistic assumption to simplify the problem was that the adjustment of power output was instantaneous. However, in practice, the climbing rate limit limits the operating range of all in-line units to adjust the operation of the generator between two operating cycles. Power generation can increase or decrease the corresponding up and down ramp rate limits. As a result, units are subject to the rate of climb mentioned below. If power generation increases, $P_{it} - P_i^{t-1} \leq UR_i$. If power generation decreases, $P_i^{t-1} - P_{it} \leq DR_i$. The inclusion of ramp rate limits modifies the generator operation constraints as Eq. (26).

This paper adopts the fitness function model in Eq. (27) for simulation to evaluate the fitness of each individual in the population, minimize the fuel cost, and simultaneously satisfy the unit and system constraints, where λ_1 and λ_r are penalty parameters. The penalty factor modulates the objective function so that the algorithm gives a higher cost value, rather than directly judging that the solution is not feasible. The penalty term reflects the violation of the equality constraint, which imposes a higher cost on the penalty function. $P_{r\,\lim}$ is defined as Eq. (28).

Table 16
Results on DED problem.

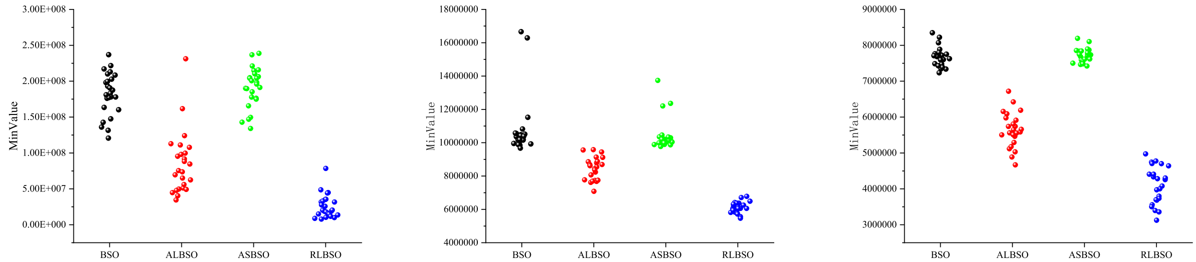
FES		BSO	ALBSO	ASBSO	RLBSO
50000	*Mean*	1.82E+08	8.50E+07	1.91E+08	**2.56E+07**
	Max	2.37E+08	2.31E+08	2.39E+08	**7.85E+07**
	Min	1.21E+08	3.45E+07	1.34E+08	**7.83E+06**
100000	*Mean*	1.10E+07	1.91E+07	1.11E+07	**6.13E+06**
	Max	1.67E+07	1.46E+08	2.40E+07	**6.78E+06**
	Min	9.67E+06	7.08E+06	9.77E+06	**5.46E+06**
150000	*Mean*	7.67E+06	9.65E+06	7.69E+06	**4.13E+06**
	Max	8.35E+06	1.06E+08	8.19E+06	**4.97E+06**
	Min	7.23E+06	4.67E+06	7.42E+06	**3.12E+06**

5.3. Result and discussion

RLBSO are applied to this problem, and compared with ASBSO, ALBSO, and BSO. The four algorithms are run independently for 25 times. The average, best and worst objective function values obtained during these 25 independent runs are calculated after the respective execution of 50000, 100000 and 150000 FEs.

The mean, maximum and minimum values of RLBSO and its comparison algorithm on the three evaluation times are listed in Table 16, and the values of RLBSO are all less than other comparison algorithms.

The three small graphs respectively represent the scatter graphs of the four algorithms in the three evaluation times in Fig. 14. The horizontal axis represents the four algorithms to



$FES = 50000$ $FES = 100000$ $FES = 150000$

Fig. 14. Stability comparison of the four algorithms in solving problem.

solve this problem, and the vertical axis represents the minimum value of the four algorithms to solve this problem. Each small dot represents a running result, and each algorithm consists of 25 points. The more concentrated the distribution of points is, the better the stability of the algorithm is. The lower the point is, the smaller the fitness value of the algorithm is and the better the performance of the algorithm is. It can be concluded from the figure that the stability of RLBSO is better than that of the other three comparison algorithms.

6. Conclusion and future work

A reinforcement learning brain storm optimization algorithm is proposed to improve the ability of brain storm optimization in this paper. The RLBSO is tested on the CEC 2017 benchmark test suite, and the Friedman test and Wilcoxon test are utilized to statistically analyze the experimental results. The results of the tests show that the RLBSO outperforms the advanced BSO algorithm variants and the state-of-the-art algorithms. The boxplots and the convergence curves show that the convergence speed and stability of RLBSO are better than those of the comparison algorithms. The effectiveness of the operation is verified in the form of experiment. The mechanism based on Q-learning and self-learning mechanism are helpful for the algorithm to quickly find the global optimal solution. Four mutation strategies significantly balance the local search and global search capability of the RLBSO. The crossover operation further increases the probability of an individual becoming a superior solution. In addition, RLBSO is applied to solve DED problem in CEC2011. The results show the practicability of RLBSO in practical engineering problems. In general, RLBSO has significant capability in solving high-dimensional problems. In the future, the RLBSO will be applied to applications of high dimension and large-scale.

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

This work was financially supported by the National Natural Science Foundation of China under grant 62063021. It was also supported by the Key Research Programs of Science and Technology Commission Foundation of Gansu Province, China (21YF5WA086), Lanzhou Science Bureau project (2018-rc-98), Project of Zhejiang Natural Science Foundation, China (LGJ19E050001), and Project of Zhejiang Natural Science Foundation, China (LQ20F020011), respectively.

References

[1] F. Zhao, X. He, L. Wang, A two-stage cooperative evolutionary algorithm with problem-specific knowledge for energy-efficient scheduling of no-wait flow-shop problem, IEEE Trans. Cybern. (2020) http://dx.doi.org/10.1109/tcyb.2020.3025662.

[2] F. Zhao, L. Zhao, L. Wang, H. Song, An ensemble discrete differential evolution for the distributed blocking flowshop scheduling with minimizing makespan criterion, Expert Syst. Appl. (2020) http://dx.doi.org/10.1016/j.eswa.2020.113678.

[3] D. Bhati, P. Singh, Branch and bound computational method for multi-objective linear fractional optimization problem, Neural Comput. Appl. 28 (2017) 3341–3351, http://dx.doi.org/10.1007/s00521-016-2243-6.

[4] A.C. Luna, N.L. Diaz, M. Graells, J.C. Vasquez, J.M. Guerrero, Mixed-integer-linear- programming-based energy management system for hybrid PV-wind-battery microgrids: Modeling, design, and experimental verification, IEEE Trans. Power Electron. 32 (2017) 2769–2783, http://dx.doi.org/10.1109/tpel.2016.2581021.

[5] 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.

[6] Y. Cao, H. Zhang, W. Li, M. Zhou, Y. Zhang, W.A. Chaovalitwongse, Comprehensive learning particle swarm optimization algorithm with local search for multimodal functions, IEEE Trans. Evol. Comput. 1 (2018).

[7] G. Gao, Y. Mei, Y.-H. Jia, W.N. Browne, B. Xin, Adaptive coordination ant colony optimization for multipoint dynamic aggregation, IEEE Trans. Cybern. (2021) http://dx.doi.org/10.1109/tcyb.2020.3042511.

[8] Y. Wang, S. Gao, Y. Yu, Z. Cai, Z. Wang, A gravitational search algorithm with hierarchy and distributed framework, Knowl.-Based Syst. (2021) http://dx.doi.org/10.1016/j.knosys.2021.106877.

[9] Y. Wang, S. Gao, M. Zhou, Y. Yu, A multi-layered gravitational search algorithm for function optimization and real-world problems, IEEE/CAA J. Autom. Sin. (2021) http://dx.doi.org/10.1109/JAS.2020.1003462.

[10] S. Turk, M. Deveci, E. Ozcan, F. Canitez, R. John, Interval type-2 fuzzy sets improved by simulated annealing for locating the electric charging stations, Inf. Sci. (Ny). 547 (2021) 641–666, http://dx.doi.org/10.1016/j.ins.2020.08.076.

[11] F. Zhao, L. Zhang, Y. Zhang, W. Ma, C. Zhang, H.B. Song, A hybrid discrete water wave optimization algorithm for the no-idle flow shop scheduling problem with total tardiness criterion, Expert Syst. Appl. 146 (2020) 21, http://dx.doi.org/10.1016/j.eswa.2019.113166.

[12] Z. Zhan, Z. Wang, H. Jin, J. Zhang, Adaptive distributed differential evolution, IEEE Trans. Cybern. (2019) 1–15.

[13] F. Zhao, L. Zhao, L. Wang, H. Song, A collaborative LSHADE algorithm with comprehensive learning mechanism, Appl. Soft Comput. J. (2020) http://dx.doi.org/10.1016/j.asoc.2020.106609.

[14] B. Doerr, M.S. Krejca, Significance-based estimation-of-distribution algorithms, IEEE Trans. Evol. Comput. 24 (2020) 1025–1034, http://dx.doi.org/10.1109/tevc.2019.2956633.

[15] Y. Shi, Brain storm optimization algorithm, in: Y. Tan, Y. Shi, Y. Chai, G. Wang (Eds.), Adv. Swarm Intell. Pt I, 2011, pp. 303–309.

[16] Y. Shi, An optimization algorithm based on brainstorming process, Int. J. Swarm Intell. Res. (2011).

[17] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov, Proximal policy optimization algorithms, 2017.

[18] H. Liu, Z. Zhang, D. Wang, WRFMR: A multi-agent reinforcement learning method for cooperative tasks, IEEE Access. (2020) http://dx.doi.org/10.1109/ACCESS.2020.3040985.

[19] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, I. Mordatch, Multi-agent actor-critic for mixed cooperative-competitive environments, Adv. Neural Inf. Process. Syst. (2017).

[20] A. Seyyedabbasi, R. Aliyev, F. Kiani, M.U. Gulle, H. Basyildiz, M.A. Shah, Hybrid algorithms based on combining reinforcement learning and metaheuristic methods to solve global optimization problems, Knowl.-Based Syst. (2021) http://dx.doi.org/10.1016/j.knosys.2021.107044.
[21] M.I. Radaideh, K. Shirvan, Rule-based reinforcement learning methodology to inform evolutionary algorithms for constrained optimization of engineering applications, Knowl.-Based Syst. (2021) http://dx.doi.org/10.1016/j.knosys.2021.106836.
[22] J. Zhang, A.C. Sanderson, JADE: Adaptive differential evolution with optional external archive, IEEE Trans. Evol. Comput. (2009) http://dx.doi.org/10.1109/TEVC.2009.2014613.
[23] S. Gao, Y. Yu, Y. Wang, J. Wang, M. Zhou, Chaotic local search-based differential evolution algorithms for optimization, IEEE Trans. Cybern. 99 (2019) 1–14.
[24] F. Zhao, J. Zhao, L. Wang, J. Cao, J. Tang, A hierarchical knowledge guided backtracking search algorithm with self-learning strategy, Eng. Appl. Artif. Intell. (2021) http://dx.doi.org/10.1016/j.engappai.2021.104268.
[25] M. Liu, Y. Shen, Y. Shi, A hybrid brain storm optimization algorithm for dynamic vehicle routing problem.
[26] V. Rajinikanth, An approach to extract low-grade tumor from brain mri slice using soft-computing scheme, in: ICCAN 2019, 2019.
[27] G. Wang, G. Hao, S. Cheng, Y. Shi, Z. Cui, An improved brain storm optimization algorithm based on graph theory, in: 2017 IEEE Congr. Evol. Comput., 2017.
[28] D. Azuma, Y. Fukuyama, A. Oi, T. Jintsugawa, H. Fujimoto, Parallel Multi-population Improved Brain Storm Optimization with Differential Evolution strategies for State Estimation in Distribution Systems using Just in Time Modeling and Correntropy, in: 2019 IEEE Symp. Ser. Comput. Intell, 2020.
[29] S. Ogawa, H. Mori, PV output forecasting by deep Boltzmann machines with SS-PPBSO, Electr. Eng. Japan. 213 (2020) 3–12, http://dx.doi.org/10.1002/eej.23274.
[30] Z. Cao, Y. Shi, X. Rong, B. Liu, Y. Bo, Random Grouping Brain Storm Optimization Algorithm with a New Dynamically Changing Step Size, in: Int. Conf. Swarm Intell, 2015.
[31] Z. Zhan, J. Zhang, Y. Shi, H. Liu, A modified brain storm optimization, Evol. Comput. (2012).
[32] Y. Shi, Brain storm optimization algorithm in objective space, in: 2015 IEEE Congr. Evol. Comput, 2015.
[33] D. Oliva, M.A. Elaziz, An improved brainstorm optimization using chaotic opposite-based learning with disruption operator for global optimization and feature selection, Soft Comput. 24 (2020) 14051–14072, http://dx.doi.org/10.1007/s00500-020-04781-3.
[34] J. Luo, R. Zhang, J. Weng, J. Gao, Y. Gao, Brain storm optimization algorithm with estimation of distribution, 2020.
[35] Y. Yu, S. Gao, S. Cheng, Y. Wang, S. Song, CBSO: a memetic brain storm optimization with chaotic local search, Memet. Comput. (2018).
[36] Y. Yu, S. Gao, Y. Wang, Z. Lei, J. Cheng, Y. Todo, A multiple diversity-driven brain storm optimization algorithm with adaptive parameters, IEEE Access. (2019) http://dx.doi.org/10.1109/ACCESS.2019.2939353.
[37] Y. Yu, S. Gao, Y. Wang, J. Cheng, Y. Todo, ASBSO: An improved brain storm optimization with flexible search length and memory-based selection, IEEE Access. (2018) http://dx.doi.org/10.1109/ACCESS.2018.2852640.
[38] M. El-Abd, Global-best brain storm optimization algorithm, Swarm Evol. Comput. (2017) http://dx.doi.org/10.1016/j.swevo.2017.05.001.
[39] Y. Wang, S. Gao, Y. Yu, Z. Xu, The discovery of population interaction with a power law distribution in brain storm optimization, Memet. Comput. (2019) http://dx.doi.org/10.1007/s12293-017-0248-z.
[40] C. Shi, Y. Shi, Q. Qin, T.O. Ting, R. Bai, Maintaining population diversity in brain storm optimization algorithm, Evol. Comput (2014).
[41] M. El-Abd, Brain storm optimization algorithm with re-initialized ideas and adaptive step size, in: IEEE Congr. Evol. Comput, 2016.
[42] F. Pourpanah, R. Wang, X. Wang, Y. Shi, D. Yazdani, mBSO: A multi-population brain storm optimization for multimodal dynamic optimization problems, in: 2019 IEEE Symp. Ser. Comput. Intell, 2020.
[43] J. Liu, H. Peng, Z. Wu, J. Chen, C. Deng, Multi-strategy brain storm optimization algorithm with dynamic parameters adjustment, Appl. Intell. 50 (2020).
[44] Y. Sun, J. Wei, T. Wu, K. Xiao, Y. Jin, Brain storm optimization using a slight relaxation selection and multi-population based creating ideas ensemble, Appl. Intell. (2020).
[45] Y. Shen, J. Yang, S. Cheng, Y. Shi, BSO-AL: Brain storm optimization algorithm with adaptive learning strategy, in: 2020 IEEE Congr. Evol. Comput, 2020.
[46] L. Qu, Q. Duan, J. Yang, S. Cheng, R. Zheng, Y. Shi, BSO-CLS: brain storm optimization algorithm with cooperative learning strategy, in: Nat. Public Heal. Emerg. Collect, pp.12145.
[47] Z. Cao, L. Wang, An active learning brain storm optimization algorithm with a dynamically changing cluster cycle for global optimization, Cluster Comput. (2019).
[48] L. Ma, S. Cheng, Y. Shi, Enhancing learning efficiency of brain storm optimization via orthogonal learning design, IEEE Trans. Syst. Man, Cybern. Syst. (2020) 1–20.
[49] Y. Xu, L.B. Ma, M. Shi, Adaptive brain storm optimization based on learning automata, 2020.
[50] S. Vimal, M. Khari, R.G. Crespo, L. Kalaivani, N. Dey, M. Kaliappan, Energy enhancement using multi objective ant colony optimization with double Q learning algorithm for IoT based cognitive radio networks, Comput. Commun. 154 (2020) 481–490, http://dx.doi.org/10.1016/j.comcom.2020.03.004.
[51] H. Samma, J. Mohamad-Saleh, S.A. Suandi, B. Lahasan, Q-learning-based simulated annealing algorithm for constrained engineering design problems, Neural Comput. Appl. (2019).
[52] X. Deng, D. Han, J. Dezert, Y. Deng, Y. Shyr, Evidence combination from an evolutionary game theory perspective, IEEE Trans. Cybern. 46 (2016) 2070–2082, http://dx.doi.org/10.1109/tcyb.2015.2462352.
[53] Z. Chen, S. Ding, T. Peng, C. Yang, W. Gui, Fault detection for non-Gaussian processes using generalized canonical correlation analysis and randomized algorithms, IEEE Trans. Ind. Electron. 65 (2018) 1559–1567, http://dx.doi.org/10.1109/tie.2017.2733501.
[54] X. Chang, Y. Yang, Semisupervised feature analysis by mining correlations among multiple tasks, IEEE Trans. Neural Networks Learn. Syst. 28 (2017) 2294–2305, http://dx.doi.org/10.1109/tnnls.2016.2582746.
[55] S. Gao, M. Zhou, Y. Wang, J. Cheng, H. Yachi, J. Wang, Dendritic neuron model with effective learning algorithms for classification, approximation, and prediction, IEEE Trans. Neural Networks Learn. Syst. (2019) http://dx.doi.org/10.1109/TNNLS.2018.2846646.
[56] C. Li, Z. Song, J. Fan, Q. Cheng, P.X. Liu, A brain storm optimization with multi-information interactions for global optimization problems, IEEE Access. (2018) http://dx.doi.org/10.1109/ACCESS.2018.2821118.
[57] R. Kommadath, P. Kotecha, Teaching Learning Based Optimization with Focused Learning and Its Performance on CEC2017 Functions, IEEE, 2017.
[58] A. Tangherloni, L. Rundo, M.S. Nobile, Proactive Particles in Swarm Optimization: a Settings-Free Algorithm for Real-Parameter Single Objective Optimization Problems, IEEE, 2017.
[59] N. Lynn, P.N. Suganthan, Ensemble particle swarm optimizer, Appl. Soft Comput. J. (2017) http://dx.doi.org/10.1016/j.asoc.2017.02.007.
[60] S. Das, P.N. Suganthan, Problem definitions and evaluation criteria for CEC 2011 competition on testing evolutionary algorithms on real world optimization problems, Electronics (2011).