



A co-evolutionary migrating birds optimization algorithm based on online learning policy gradient

Fuqing Zhao^{a,*}, Tao Jiang^a, Tianpeng Xu^a, Ningning Zhu^a, Jonrinaldi^b

^a School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China

^b Department of Industrial Engineering, Universitas Andalas, Padang 25163, Indonesia



ARTICLE INFO

Keywords:

Migrating birds optimization
Co-evolution
Long short-term memory
Policy gradient
Markov decision process

ABSTRACT

A co-evolutionary migrating birds optimization algorithm based on online learning policy gradient (CMBO-PG) is proposed to address complex continuous real-parameter optimization problems. In CMBO-PG, a Gaussian estimation of distribution algorithm (GEDA), which enhances the exploitation tendency, is utilized to generate the solutions of the leading flock. The neighborhood solutions of the following flock are produced by a multi-strategy learning mechanism to promote exploration capability. The co-evolution of the leading flock and following flock is realized by the information-sharing mechanism and the operation of destruction and construction to keep the balance of exploration and exploitation. The nonlinear selection of mutation strategies is laborious due to the differences in the ability to address optimization problems. In the mechanism of multi-strategy learning, a long short-term memory (LSTM) is adopted as a selector of mutation strategies to predict the selection probability of three mutation strategies. The evolutionary procedure of the following flock is modeled as a Markov decision process (MDP). The policy gradient (PG) is employed as a model optimizer to control the parameters of LSTM based on the historical feedback information. The performance of CMBO-PG is testified on the CEC 2017 benchmark test suite. The experimental results show that CMBO-PG is superior to the 12 comparison algorithms, including state-of-art algorithms.

1. Introduction

Optimization is perceived as a problem to search for an optimal solution under a given function, prescribed decision variables, and certain problem constraints (Huang, Li, & Yao, 2020). The optimization problem widely exists in engineering design (Nekooi, Acosta, & Ollero, 2022), economic models (Yousefnejad & Monfared, 2022), and scheduling (Qiao, Wu, He, Li, & Chen, 2022). Complex continuous optimization problems are usually defined as the minimization of the continuous objective function (Zhao, Zhu, et al., 2022). In complex continuous real-parameter optimization problems, the existence of a large number of variables considerably expands the scale of the problem. The problems of higher dimensionality and huge decision-making space require a lot of time and computing resources to obtain the objective solution. The increase in the spatial dimension leads to the shift of the characteristics, which results in an exponential increase in regional extreme points. The complex of nonlinear and nondifferentiable characteristics considerably raises the difficulty of addressing continuous real-parameter optimization (Wang, Gong, Liao, & Wang, xxxx).

The traditional method of solving optimization problems is deterministic mathematical methods based on gradients, including the Newton method (Altinoz & Yilmaz, 2019), the conjugate gradient method (Yang, 2022), etc. A certain point along the gradient direction of the problem is iterated for refinement to obtain the optimal value via analyzing the structure and the characteristics of the problem. However, the characteristic of large-scale practical engineering problems with the nonlinear, nondifferentiable, multimodal, black box, and the impact of noise increases the difficulty of solving problems with traditional methods (Yang, Zhou, Li, & Yao, 2021). Due to the deficiencies of mathematical methods in solving practical problems, meta-heuristic algorithms (Chen et al., 2022) are commonly utilized to solve large-scale optimization problems.

Meta-heuristic algorithms tend to search for an optimal approximate solution with minimal error compared with mathematical methods which obtain precise solutions based on the problem characteristics in the optimization problem (Huang et al., 2020). Therefore, the meta-heuristic algorithm markedly reduces the overhead of time and computational resources when addressing NP-hard problems and

* Corresponding author.

E-mail addresses: zhaofq@lut.edu.cn (F. Zhao), 18755525159@163.com (T. Jiang).

complex continuous problems potentially within an acceptable time (Zhao et al., 2022). The universal meta-heuristic algorithm constructs the mechanism independent of the problem characteristics, which overcomes the black box and nonlinear problems in complex continuous problems. Meta-heuristic algorithms are divided into evolutionary algorithms, physics-based algorithms, and swarm intelligence algorithms in this paper (Zhao, Zhao, Wang, Cao, & Tang, 2021). Evolutionary algorithms follow the principle of survival of the fittest including genetic algorithm (GA) (Qiao et al., 2022), differential evolution (DE) (Civicioglu & Besdok, 2022), estimation of distribution algorithm (EDA) (Wang et al., 2021), etc. Physic-based algorithms draw inspiration from physical phenomena including simulated annealing (SA) (Kirkpatrick, Gelatt, & Vecchi, 1983), biogeography-based optimization (BBO) (Simon, 2008), etc. The swarm intelligence algorithms get inspiration via imitating the social behavior of natural animals including particle swarm optimization (PSO) (Xia et al., 2020), ant colony optimization (ACO) (Hx et al., 2021), etc.

Although the meta-heuristic algorithm has achieved satisfactory results in solving continuous optimization and combinatorial optimization problems, the performance of the meta-heuristic algorithm has different degrees of deviation due to different mutation strategies. It is noteworthy that no mutation strategy algorithm performs best in all the cases as stated by the no-free lunch theorem (Wolpert & Macready, 1997). Recently, the strategy pool, which includes multiple mutation strategies suitable for a certain problem, is proposed to adjust the diversity of the population (Li, Wang, Yang, Chen, & Yang, 2023; Zhao, Zhu, et al., 2022). The strategy selection is characterized by complexity and nonlinearity, and how to select the appropriated mutation strategy during the algorithm evolution is the priority and difficulty problem. Since the learning mechanism adjusts parameters adaptively by the feedback of historical information, the research works about integrating the learning mechanism into the meta-heuristic algorithm is the focus of current research (Zhao, Wang, & Wang, 2023).

In this paper, a co-evolutionary migrating birds optimization algorithm based on online learning policy gradient (CMBO-PG) is proposed to address continuous optimization problems. The leading flock is defined as a subpopulation containing certain elite individuals, and the following flock is defined as a subpopulation containing all individuals except certain elite individuals. In CMBO-PG, a Gaussian estimation of distribution algorithm (GEDA) is utilized to generate the neighborhood solutions of the leading flock. The fast convergence of GEDA enhances the exploitation of the CMBO-PG. A multi-strategy learning mechanism, which generates solutions for the following flock, is designed to enhance the exploration of CMBO-PG. For the multi-strategy learning mechanism, three different mutation strategies including *MBO/current-to-pbest/1*, *MBO/rand/1*, and *MBO/current-to-rand/1* are constructed into the candidate pool of mutation strategies. In order to select the appropriate mutation strategy for the specific problem, a neural network called long short-term memory (LSTM) is utilized as a strategy selector to predict the strategy selection probability via constructing a nonlinear function (Sun, Liu, Back, & Xu, 2021). The evolutionary procedure of the following flock is modeled as a finite-horizon Markov decision process (MDP). The parameters of LSTM are controlled via policy gradient (PG) to update the predicted results of the LSTM in this work. The main contributions of the paper are as follows.

- 1) A meta-heuristic algorithm named CMBO-PG for co-evolution of the leading flock and the following is proposed to address complex continuous real-parameter optimization problems.
- 2) For the nonlinear selection of mutation strategy, a selector based on LSTM is designed to fit the selection probabilities online.
- 3) A model optimizer based on PG is proposed to learn the parameters of the LSTM. The selector of mutation strategies is improved dynamically according to the historical feedback information.

The procedure details of CMBO-PG are introduced in Section 2.

Experiments and analysis are described in Section 3. Conclusions and future work are shown in Section 4.

2. Related works

2.1. Migrating birds optimization

Migrating birds optimization (MBO) (Duman, Uysal, & Alkaya, 2012), which is a population-based algorithm inspired by the flight of V-shaped structures, was proposed to address the quadratic assignment problem (QAP). Individuals of MBO share information through benefit mechanisms and leader supersession to discover the optimal solution. In MBO, the useful knowledge obtained by the elite solution (leading bird) is incorporated within the other solutions (the following birds) to guide the meta-heuristic toward better performance in terms of solution quality, convergence rate, and robustness. The mechanism of leader supersession disrupted the ordering of the population to redefine the elite solution, which enhances the diversity of MBO. Therefore, MBO keeps a good balance between the exploration and the exploitation stage.

Recently, the improved MBO is widely employed in solving combinatorial optimization problems and continuous real-parameter optimization problems due to its outstanding performance. For the scheduling problems, an improved MBO was utilized to solve the hybrid flow-shop scheduling problem by combining the heuristic for specific problems with the local search process (Pan & Dong, 2014). Gao and Pan (2016) proposed a shuffled multiswarm micro-migrating birds optimization (SM²-MBO) to address a multiple resource-constrained flexible job-shop scheduling problem. In SM²-MBO, the population is divided into multiple micro-populations, and each micro-population executes the corresponding MBO independently. A random shuffled process is designed for the co-evolution of multiple micro-populations. Meng, Pan, Li, and Sang (2018) proposed an improved migrating birds optimization (IMMBO) to solve the lot-streaming flow-shop scheduling problem, which designed a scheme based on harmony search (HS) to construct the neighborhood of the solution. Tongur et al (Tongur & Ülker, 2018) designed a PSO-based multi-flocks migrating birds optimization (IMFMBO) to solve the traveling salesman problem. In IMFMBO, the multi-flocks are employed to avoid local minimum, and the interaction of these flocks promotes the algorithm achieving better solutions. Han et al (Han, Tang, Zhang, & Li, 2020) improved the MBO and applied it to steel-making and the continuous-casting scheduling problem. Deng, Xu, Zhang, Jiang, and Su (2021) proposed migrating birds optimization with a diversified mechanism (dMBO) considering the machine energy consumption in blocking flow-shop. Furthermore, MBO performs well on assembly line balance problems (Xiao, Guo, & Li, xxxx), facility layout problems (Niroomand, Hadi-Vencheh, Sahin, & Vizvári, 2015), and other problems. Segredo et al. (2018) proposed an enhanced MBO (EMBO) algorithm to address large-scale continuous problems. EMBO hybrid the mutation strategy of DE and the MBO framework given the low efficiency and premature convergence of MBO. The results of the comparison experiment indicate the effectiveness of the hybrid mechanism in dealing with large-scale continuous problems.

Through the review of related literature, the variants of MBO have been widely studied in combinatorial optimization problems such as scheduling. There are very few works employing the algorithm in complex continuous problems. The reason is that the original MBO neighborhood search operator has insufficient efficiency and poor diversity for continuous optimization problems. Segredo et al. designed a new neighborhood operator and achieved excellent performance in solving continuous optimization problems, which shows that MBO has great potential in solving complex continuous optimization problems. The core problem is to select a suitable mutation strategy for MBO to act as the search operator.

2.2. Strategy selection based on learning mechanism

The no-free lunch theorem states that no mutation strategy performs superior and steady performance in all functions. It is worth studying to adaptive select the appropriate mutation strategy for solving different function problems. The learning mechanism adjusts the parameters and strategies adaptively using the feedback of historical information, including the parameter adaptive mechanism, the ensemble learning mechanism, and the reinforcement learning (RL) mechanism, etc.

[Qin, Huang, and Suganthan \(2009\)](#) introduced a differential evolution algorithm with strategy adaptation (SaDE). The values of successful individuals were preserved in the memory bank. The selection probabilities of different mutation strategies were self-adaptively modified by learning from the previous experience of a promising solution. [Wu, Mallipeddi, Suganthan, Wang, and Chen \(2016\)](#) proposed a differential evolution with a multi-population-based ensemble of mutation strategies (MPEDE). The reward subpopulation is assigned to the best-performing indicator subpopulation for ensemble learning in MPEDE. [Wu et al. \(2018\)](#) proposed a framework based on multi-population to integrate multiple mutation strategies. The mutation strategies with different merits support one another to solve optimization problems cooperatively.

In RL, the MDP model is constructed and the reward value is given to different actions in each step. the agent selects the next action according to the feedback on reward value. RL is widely used in strategy selection due to the capability of strong decision-making. [Zhao et al. \(2022\)](#) integrated a method of reinforcement learning named Q-learning to select mutation strategies. The mutation strategy based on the best Q-value is selected to generate new solutions. The Q-value recorded in the Q-table is calculated based on the feedback reward function value. [Cai, Lei, Wang, and Wang \(2022\)](#) designed a shuffled frog-learning algorithm with Q-learning (QSFLA) to address the distributed hybrid flow shop scheduling problem. In QSFLA, the Q-learning process is embedded to select a search strategy dynamically for memeplex search. [Gölcük and Ozsoydan \(2021\)](#) proposed a hyper-heuristic architecture based on Q-learning to address optimization problem. Four meta-heuristic algorithms are designed as low-level optimizers, and Q-learning is used as the high-level rule to select the optimizer automatically in the optimization process. [Wang and Wang \(2022\)](#) design an RL-based policy agent to select appropriate operators through offline learning. [Song, Chen, Li, and Cao \(2023\)](#) employ deep reinforcement learning to select the priority dispatching rules for solving flexible job shop scheduling problems. Due to the strong fitting ability of neural networks for nonlinear problems, some research works ([Chen, Xing, Xiao, Xu, & Tao, 2021](#); [Song et al., 2023](#); [Xiao et al., 2021](#)) combine the neural networks and RL to fit the selection probability of different strategies.

From the literature review, the related research about embedding learning mechanisms, especially RL mechanism, into meta-heuristic algorithms has gained some achievement. Under the learning mechanism, the parameters are controlled adaptively, and the mutation strategy is selected reasonably. It shows that the selection of mutation strategy guided by RL has a foundation in solving optimization problems. Few works consider the sequential process of the meta-heuristic algorithm in the selection of mutation strategy. It is worthwhile using neural networks with sequential processes (such as LSTM) to fit the selection probability of different strategies. Second, there are few related research works on using RL to guide meta-heuristic algorithms by online learning. Compared with offline learning, the online learning mechanism dynamically select the mutation strategy to guide the evolution of the algorithm in each iteration process according to the current state. Therefore, this paper designs a selector based on LSTM to fit the selection probabilities online and proposes a model optimizer based on PG to improve LSTM dynamically.

3. Co-evolutionary migrating birds optimization algorithm based on online learning policy gradient

3.1. The basic migrating birds optimization

MBO, inspired by the V-shaped flight structure of migratory birds, was first proposed and applied to quadratic assignment problems ([Duman et al., 2012](#)). The bird at the forefront of the structure is called the leading bird, and the remaining birds are called the following birds. The schematic diagram is shown in [Fig. 1](#) and [Fig. 2](#).

From [Fig. 1](#), the leading bird generates k neighborhood solutions which are sorted by fitness value. The leader is improved by the optimum of neighborhood solutions. x suboptimal solutions are shared with the following birds. The following bird generates $k - x$ neighborhood solutions which merged with the shared solutions into k neighborhood solutions. The k neighborhood solutions are utilized to evolve the follower and to share with the next bird. As shown in [Fig. 2](#), the leader, who is exhausted after m tours, moves to the back of the team, and the following bird moves to the front position in turn. The pseudo-code of the basic MBO is shown in Algorithm 1.

Algorithm 1: The Basic MBO Algorithm

1	Initialize n birds;
2	Place birds on a hypothetical V formation;
3	While Termination Occurs is not satisfied
4	For $i = 1$ to m
5	Evolve the leader bird;
6	Evolve the followers arranged on the left;
7	Evolve the followers arranged on the right;
8	End for
9	Update with the optimal solution;
10	Change the leader bird;
11	End while

The benefit mechanism and leader supersession are the main mechanisms of MBO. The leading bird guides the generation of new solutions to the following bird via sharing information. The following bird transmits information to the leader via leader supersession. MBO co-evolution through information interaction between the leading bird and the following bird.

3.2. The framework of CMBO-PG

[Section 3.2](#) introduces the overall framework of CMBO-PG. The co-evolution mechanism of the leading flock and the following flock is described in [section 3.2.1](#). An evolutional mechanism based on GEDA is proposed to generate neighborhood solutions for the leading flock in [section 3.2.2](#). A multi-strategy learning mechanism is introduced as the method of neighborhood solutions for the following flock in [section 3.2.3](#). The schematic diagram of CMBO-PG is shown in [Fig. 3](#) and [Fig. 4](#). The pseudo-code of CMBO-PG is shown in Algorithm 2.

In CMBO-PG, n individuals which are initialized randomly are divided into two populations. $L1$ individuals with better fitness value are allocated into the leading flock, and the rest individuals are divided into the following flock. The GEDA evolutional mechanism is employed to generate neighborhood solutions for the leading flock. The $L1$ best-performance neighborhood solutions are utilized to evolve the leading flock, and the x sub-optimal solutions are shared to guide the following flock. In this paper, a multi-strategy learning mechanism is designed to generate the neighborhood of the following flock. As shown in [Fig. 3](#), the learning mechanism integrates three mutation strategies with different characteristics into a strategy pool. The LSTM network is constructed to fit the selection probabilities ($p1, p2$, and $p3$) of mutation strategies in the strategy pool. According to the selection probabilities of mutation strategies, the following flock is divided into three sub-populations, namely $pop1, pop2$, and $pop3$. Each sub-population generates the neighborhood solutions by the corresponding mutation strategy. The fitness

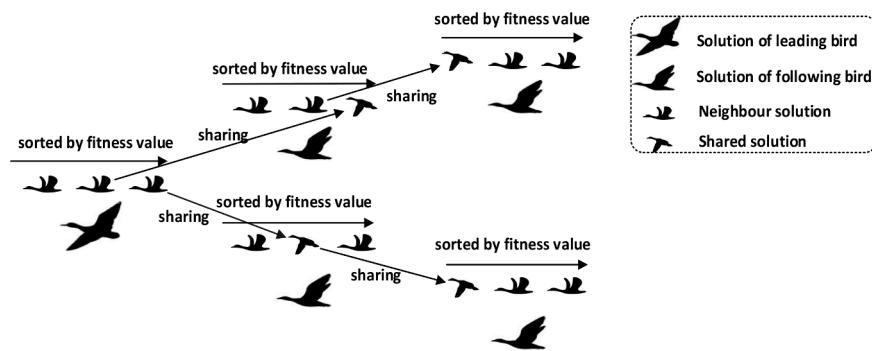
Fig. 1. Flock evolution ($k = 3, x = 1$).

Fig. 2. Leader supersession.

value of the neighborhood solution in $pop1, pop2$, and $pop3$ are stored in the memory bank. The length of the memory bank is LP , and the earliest information will be replaced by new information when the memory bank is full. The evolutionary process of the following flock is modeled as MDP, and the reward value is calculated according to the stored information of the memory bank. The parameters of the LSTM network are updated with the PG method. The leading flock and the following flock cooperatively evolve the population with the operation of destruction and construction when the population iterated for m times.

Algorithm 2: CMBO-PG

Input: The number of individuals in the flock n , the number of flaps m , the number of sharing solutions x , the length of the memory bank LP , the number of the leading flock $L1$, the number of the following flock $L2$; The optimal value $optimal$;
Output: Initialize a flock of n individual birds randomly;
1 Divide the flock into the leading flock of $L1$ individuals and the following flock of $L2$ individuals;
2 $t = 1$;
3 While termination conditions are not met
4 For $i = 1$ to m
5 The leading flock generates n neighborhood solutions according to Algorithm 3;
6 Sort the neighborhood solutions according to fitness value;
7 Evolution of leading flock with the optimal $L1$ solutions;
8 Share x suboptimal solutions with the following flock randomly;
9 Construct an LSTM neural network and fit selection probabilities of mutation strategies $p1, p2$, and $p3$ by (9)-(14);
10 Divide $L2$ individuals into three sub-populations $pop1, pop2$, and $pop3$ according to the selection probabilities;

(continued on next page)

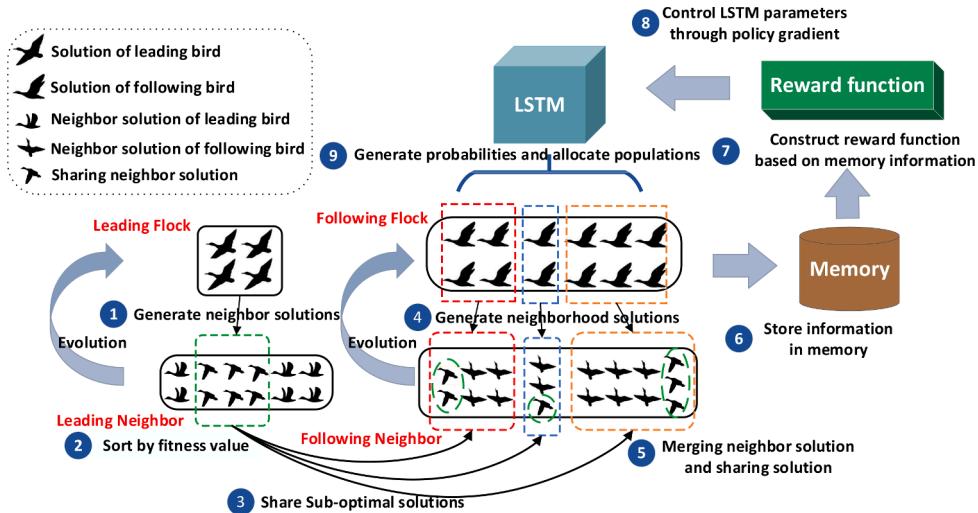


Fig. 3. CMBO-PG schematic diagram.

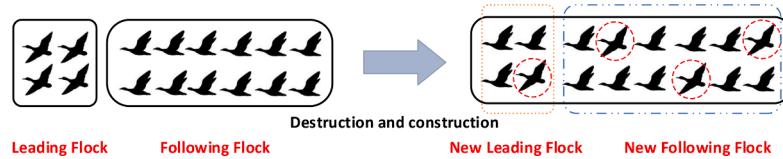


Fig. 4. The operation of destruction and construction.

(continued)

```

12      Generate L2 neighborhood solutions according to different
       strategies, as shown in Algorithm 4;
13      Combine L2 neighborhood solutions with  $x$  shared solutions and
       evolve the following flock with the merged solutions;
14      If  $t < LP$ 
15          Store the fitness value as historical information of  $pop1, pop2$ , and
        $pop3$  in the memory;
16      Else
17          Store fitness information in the memory bank and replace the
       earliest information;
18      End if
19       $t = t + 1$ ;
20      Model the evolution process of the following flock into an MDP;
21      Control LSTM parameters with PG through (19)-(23);
22      Record the optimum of the flock  $optimal$ ;
23  End for
24  Redivide the leading flock and the following flock through the
   mechanism of destruction and construction;
25 End while

```

The notations of all relevant mathematical symbols in CMBO-PG are given in Table 1 to understand and reimplement the proposed algorithm.

3.2.1. Co-evolution mechanism of the leading flock and the following flock

In this paper, a co-evolution mechanism including the information-sharing mechanism and the operation of destruction and construction is proposed to keep the balance of exploration and exploitation. In CMBO-PG, the population is divided into the leading flock and the following flock according to the emphasis. The leading flock performs superior in accelerating the convergence of the algorithm, while the following flock pays more attention to the diversity of the algorithm. As

Table 1
Notations of CMBO-PG.

Notation	Description
n	The number of individuals in the flock
m	The number of flaps
x	The number of sharing solutions
LP	The length of the memory bank
$L1$	The number of the leading flock
$L2$	The number of the following flock
τ	The truncated selection probability
ζ	The shifting factor
ω	The weighted mean solution
μ	The mean solution of the overall flock
C	The covariance matrix
$leaF$	The solutions of the leading flock
D	The number of dimensions
$folF$	The solutions of the following flock
Fc	The scaling factor satisfies the Cauchy distribution
μF	The mean value of Fc
$elite$	The elite individuals of the following flock
Fol_neigh	The neighborhood solutions of the following flock
f_t, i_t, o_t	The forget gate, input gate, and output gate at time t
x	The input information.
\hat{C}_t	The input extrusion unit at time t .
C_t	The memory cell stores long-term historical information.
W	The parameters between two layer
h	The output information of the memory cell
b	The biases
σ	The sigmoid activation function
p	The final prediction probability
$Agent$	The agent in the MDP
E	The environment in the MDP
S	The state-space of the agent
TP	The transition probability
R	The reward function
θ	The parameter matrix of the LSTM.
α	The learning factor
γ	The discount factor
Q	The state-action value function
V	The cumulative reward value function

shown in Fig. 3, the co-evolution mechanism interacts with information between the two populations for obtaining satisfactory candidate solutions. The specific operation process is as follows.

First, the information-sharing mechanism is that the leading flock share x sub-optimal solutions of neighborhood solutions with the following flock. The shared solutions and the neighborhood solutions are noticed to evolve the individuals in the following flock. Even though the performance of the neighborhood solutions generated from the following flock is not good, the individuals evolve and converge rapidly due to the shared solutions. Secondly, the operation of destruction and construction is that overall individuals are shuffled and redistributed into the leading flock and the following flock. As shown in Fig. 4, the individuals from the original following flock are mixed in the new leading flock, which enhances the diversity of the leading flock. The individuals from the original leading flock are mixed in the new following flock, which accelerates the convergence ability of the following flock. The co-evolution mechanism integrates the information of the leading flock and the following flock and promotes the evolution of the population in a method of cooperative.

3.2.2. EDA mechanism to the leading flock

The neighborhood solutions of the leading flock are produced by the mechanism of EDA (Liang et al., 2020). The Gaussian distribution model, which is constructed by calculating the mean and covariance matrix, is adopted as the probability model. EDA has obvious advantages in convergence speed compared with other meta-heuristic algorithms. The reason for rapid convergence is the ellipsoid constructed by the Gaussian distribution model. In CMBO-PG, the leading flock generates excellent solutions to guide the following flock in the early stage due to the advantage of EDA convergence speed. The pseudo-code of the neighborhood operator for the leading flock is shown in Algorithm 3.

The values of τ and ζ are consistent with (Liang et al., 2020). S is the elite flock namely the leading flock, which consists of top $L1$ individuals, and S_i is the i th individual in the S . f_i is the fitness value of the individual i . The mean value of the Gaussian distribution model is produced by (1)–(4).

$$\omega_i = \frac{\text{lea}F_i \times L1 \times f_{L1-i}}{\sum_{i=1}^{L1} f_i} \quad (1)$$

$$\mu = \frac{1}{n} \times \sum_{i=1}^n Flock_i \quad (2)$$

$$\Delta = \omega - \mu \quad (3)$$

$$\omega = \begin{cases} \omega + \Delta \times \zeta, & f(\omega + \Delta \times \zeta) < f(\omega) \\ \omega, & \text{otherwise} \end{cases} \quad (4)$$

$$C = \frac{1}{L1} \times (\text{lea}F - \omega) \times (\text{lea}F - \omega)^T \quad (5)$$

Where ω is the weighted mean solution of the leading flock, and ω_i is the i th weighted mean solution. From (1), the solution with a lower fitness value occupies a higher weight in the ω . μ is the mean solution of the overall flock. Δ is the difference vector between the ω and μ . C is the covariance matrix, and the formula is shown in (5). n new solutions are produced by the Gaussian distribution model according to the mean vector ω and the covariance matrix C .

Algorithm 3: Neighborhood operator for the leading flock

- Input:** The solutions of all individuals $Flock$, the number of individuals in the flock n , the truncated selection probability τ , the shifting factor ζ ;
Output: Neighborhood solutions of the leading flock $lead_neigh$, the number of the leading flock $L1$
- 1 $L1 = \tau \times n;$
 - 2 Sort $Flock$ by fitness value;
 - 3 $leaF = \text{Top } L1 \text{ individuals};$
 - 4 Generate ω, μ , and Δ according to (1)–(3).

(continued on next page)

(continued)

```

5   For  $j = 1$  to  $\zeta$ 
6     If  $f(\omega + \Delta \times \zeta) < f(\bar{\mu})$ 
7        $\omega = \omega + \Delta \times \zeta;$ 
8     End if
9   End for
10  Construct the covariance matrix  $C$  according to (5);
11  Generate  $lead\_neigh$  through Gaussian distribution;

```

3.2.3. Multi-strategy learning mechanism to the following flock

A multi-strategy learning mechanism is proposed to produce the neighborhood solutions of the following flock in CMBO-PG. The purpose of the multi-strategy learning mechanism is to generate solutions with diverse and robust. The co-evolution between the leading flock and the following flock effectively is designed to avoid the algorithm from premature convergence. The pseudo-code of the neighborhood operator for the following flock is shown in Algorithm 4.

Algorithm 4: Neighborhood operator for the following flock

Input:	The number of dimensions D , the solutions of the following flock $folF$, the number of the following flock $L2$, the mean of Fc μ_F , the elite individuals of the following flock $elite$;
Output:	All new solutions generated by the following flock Fol_neigh ;

```

1  Generate  $Fc = cauchy(\mu_F, 0.1);$ 
2  For  $i = 1$  to  $L2$ 
3    If  $pop_i \in pop1$ 
4       $v_i = folF_i + Fc \times (elite - folF_i + folF_{r1} - folF_{r2});$ 
5    Else if  $pop_i \in pop2$ 
6       $v_i = folF_i + Fc \times (folF_{r1} - folF_{r2});$ 
7    Else
8       $v_i = folF_i + Fc \times (folF_{r1} - folF_{r2}) + rand \times (folF_{r3} - folF_i);$ 
9    End if
10    $v_i = boundConstraint(v_i);$ 
11   Randomly generate a logical matrix  $map$ ;
12   For  $j = 1$  to  $D$ 
13     If  $map_{ij} = 1$ 
14        $Fol\_neigh_{ij} = v_{ij};$ 
15     Else
16        $Fol\_neigh_{ij} = folF_{ij};$ 
17     End if
18   End for
19 End for

```

The prediction results of LSTM are applied as the selection probabilities of three different mutation strategies which include $MBO/current_to_pbest/1$, $MBO/rand/1$, and $MBO/current_to_rand/1$ in this paper. The following flock is divided into three different sub-populations including $pop1$, $pop2$, and $pop3$. The size of the three sub-populations varies with the prediction probability of output. $MBO/current_to_pbest/1$ is applied to $pop1$, the formula is shown in (6).

$$v_i = folF_i + Fc \times (elite - folF_i + folF_{r1} - folF_{r2}) \quad (6)$$

Where v_i represents the mutation vector generated by the i th individual of the following flock through the mutation strategy. $folF_i$ is the i th individual of the following flock. $elite$ is one of the top 10% performance elites in the following flock, $r1$ and $r2$ are the subscript values respectively of different individuals in the following flock. Fc is a Cauchy distribution value with μ_F as the mean value and σ as the standard deviation. The value of the μ_F is 0.5, and σ is 0.1, which is the same as (Wu et al., 2016). $MBO/rand/1$ is applied to $pop2$, the formula is shown in (7). $MBO/current_to_rand/1$ is applied to $pop2$, the formula is shown in (8).

$$v_i = folF_i + Fc \times (folF_{r1} - folF_{r2}) \quad (7)$$

$$v_i = folF_i + Fc \times (folF_{r1} - folF_{r2}) + rand \times (folF_{r3} - folF_i) \quad (8)$$

Where $r3$ is a subscript different from $r1$ and $r2$, and the $rand$ is a random value in $[0, 1]$. $MBO/current_to_rand/1$ performs advantages in the

problems with rotation characteristics due to the rotation invariance.

3.3. The selection of mutation strategies through LSTM

3.3.1. Deep learning

Deep learning (DL), which is a branch of machine learning, is derived from artificial neural networks (ANN). ANN, which is a universal function approximator connected by a large number of neurons, is divided into the input layer, hidden layer, and output layer. Recurrent neural networks (RNN) with feedback connections establish the dependency relationship of the data at different periods by memorizing the state of the past data. LSTM (an improved RNN) memorizes the results of multiple generations compared with RNN which memorizes the data state of one generation. The selection probability information of each generation influences the performance of the algorithm. The closer to the current generation, the more influential the performance of the algorithm. Therefore, LSTM is selected as the neural network model for predicting probability in this work.

3.3.1.1. Generating selection probability through LSTM. The memory module of LSTM is composed of forget gate, input gate, output gate, and memory cell. The memory cell retains certain historical information, which influences the output result at the next moment. The calculation formula of the simplest LSTM structure is shown in (9)-(14).

$$f_t = \sigma(W_{xf} \times x_t + W_{hf} \times h_{t-1} + b_f) \quad (9)$$

$$i_t = \sigma(W_{xi} \times x_t + W_{hi} \times h_{t-1} + b_i) \quad (10)$$

$$\hat{C}_t = \tanh(W_{xc} \times x_t + W_{hc} \times h_{t-1} + b_c) \quad (11)$$

$$C_t = f_t \otimes C_{t-1} + i_t \otimes \hat{C}_t \quad (12)$$

$$o_t = \sigma(W_{xo} \times x_t + W_{ho} \times h_{t-1} + b_o) \quad (13)$$

$$h_t = o_t \otimes \tanh(C_t) \quad (14)$$

Where f_t , i_t , and o_t are the forget gate, input gate, and output gate at time t respectively, the gate units control the proportion of corresponding information transmission. x_t is the input information at time t . \hat{C}_t is the input extrusion unit at time t . C_t is a memory cell that stored long-term historical information. The long-term memory at time t is formed by the interaction of the memory cell at time $t-1$ with the input information, as shown in (12). The memory cell at time $t-1$ discards part of the historical information when flowing through the forget gate. h_t is short-term memory. h_t is the output information of the memory cell under the control of the output gate at time t . W_{xf} , W_{xi} , W_{xc} , and W_{xo} represent the parameter matrices between x_t and the forget gate, input gate, input extrusion unit, and output gate. W_{hf} , W_{hi} , W_{hc} , and W_{ho} represent the parameter matrices between h_t and the forget gate, input gate, input extrusion unit, and output gate. b_f , b_i , b_c , and b_o are the biases. σ represents the sigmoid activation function, the formula is shown in (15). \tanh represents the hyperbolic tangent activation function, the formula is shown in (16).

$$\sigma(z) = \frac{1}{1 + e^{-z}} \quad (15)$$

$$\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \quad (16)$$

Neural networks are employed to fit functions with nonlinear relationships as a function approximator with abundant neurons. The input information of the neural network is regarded as time-series data following the number of iterations. The input value of the initial LSTM is generated randomly, and the normal distribution value of the predicted probability in the previous generation is utilized as the subsequent input

value. The output value of LSTM is normalized as the selection probability in a new round of iteration. The formula of the normalized process is shown in (17).

$$p_i^t = \frac{h_i^t}{\sum_{i=1}^k h_i^t} \quad (17)$$

Where p_i^t is the final prediction probability of the i th strategy at time t , h_i^t is the output information of i th strategy at time t , k is the number of the strategy.

The traditional LSTM is guided by the correct solution to fit a function that outputs the predicted value closest to the accuracy solution through offline training. However, the correct solution is unknown during the optimization. The offline training method is not suitable to update the LSTM model under certain evaluation times due to the waste of evaluation time for the offline training process. Therefore, an RL method is adopted as online training to control the LSTM model dynamically.

3.4. Learning LSTM parameters through policy gradient

3.4.1. Reinforcement learning

RL learns in a trial-and-error manner based on the experience in the optimization process to guide the next behavior. RL procedures are normally described by the Markov decision process (MDP) including agent, environment, state, action, reward, and transition probability. Agent $Agent$ is in the environment E . The state space of the agent is S including S_1, S_2, \dots, S_t , and each state S_t is the perception of $Agent$ to the E . S_t is the state of $Agent$ at time t . The action space A is composed of all the actions taken by the agent. Each action a acts on the state S_t , the state S_t transfers to the next state S_{t+1} with a certain probability called transition probability (TP). The E rewards for different actions of A according to the reward function R . a_{t+1} is re-selected for A according to the reward value R_{t+1} at time $t + 1$. RL is divided into Value-based and Policy-based according to the evaluation method. The value of the action is the output for the method of Value-based, and the most valuable action will be selected. The classic methods of Value-based are Q-learning and Sarsa. For the Policy-based method, the probability of the next action is output, and the action will be selected according to the probability. As the classic methods of Policy-based, PG is utilized to select the action for online learning.

3.4.2. Standardize the flock evolution as an MDP

The evolution procedure of the following flock is modeled as an MDP in this paper. The following flock is considered as an $Agent$ in the optimization process, and 29 different complex optimization functions are utilized as the E . S_t is the fitness value for the following flock in generation t . a_k^t is the k th probability value of strategy selection by the following flock at time t , $k \in \{1, 2, 3\}$. The three mutation strategies including $MBO/current_to_pbest/1$, $MBO/rand/1$, and $MBO/current_to_rand/1$ for the following flock are shown in (6)-(8). The parameters of LSTM are updated to produce new predictions according to the result of historical information feedback in the evolution procedure. The output of LSTM is considered as the action a of the RL. The construction of the reward function is shown in (18).

$$R_k = \frac{M_k^{t-1} - M_k^{t-LP}}{M_k^{t-1}} + \frac{M_k^{t-1} - \text{mean}(M^{t-1})}{M_k^{t-1}} \quad (18)$$

Where M is a memory bank, which stores state information from $t-LP$ to $t-1$ generation. The construction of R is based on the degree of the change from S_k^{t-1} to S_k^{t-LP} and the contemporary states of different strategies. TP is described as the transition probability of A which adopts a_k^t from S_k^{t-1} to S_k^t . The fitness value of the population keeps changing as the number of iterations increases in evolutionary computing. Online learning, which is trained based on existing data and information, guides

the next stage of behavior according to the current training results, different from offline learning with a large amount of training data. In CMBO-PG, M is regarded as the trajectory of the RL, and LP is the length of the trajectory. The exclusive trajectory is present in MDP, the trajectory changes continuously with the iteration process.

3.4.3. Learning LSTM parameters through policy gradient

PG [Zhang, Ong, Wang, and Xue \(2021\)](#) is applied to control the parameters of LSTM by maximizing the value of the reward function. The probability distributions obtained by LSTM are utilized to perform actions. The following flock is divided into three different subpopulations including $pop1$, $pop2$, and $pop3$ according to the probability distribution. The fitness values of $pop1$, $pop2$, and $pop3$ are considered to be the new state S_1^t , S_2^t and S_3^t of the MDP. The memory bank is updated to generate a new reward function R_{t+1} that is utilized to control the parameters of LSTM. The PG constructs a reward function according to the (18) as feedback information based on historical information. The changeable degree and the excellence of fitness values for different subpopulations are considered as the criterion to evaluate probability distribution due to the unknown of the optimal solution in the process of algorithm optimization. The subpopulation will be assigned to a higher selection probability if the fitness of the subpopulation with more improvements. The detailed description of updating the LSTM through the reward function is as follows.

The gradient ascent method is utilized to find the maximum reward function by updating the parameter of LSTM. The formula of the control parameter is shown in (19).

$$\theta_{t+1} = \theta_t + \alpha \nabla_{\theta} V(\theta_t) \quad (19)$$

Where θ_t is the parameter matrix of the LSTM. α is the learning factor. The value of the α is 0.05, which is the same as [\(Sun et al., 2021\)](#). $\nabla_{\theta} V(\theta_t)$ is the gradient of V to θ_t . The key of PG is to construct a state value function V . The construction formula of the state value function is shown in (20).

$$V_{\pi}(s_t) = E_A[Q_{\pi}(s_t, A)] = \sum_a \pi(a|s_t)^* Q_{\pi}(s_t, a) \quad (20)$$

Where V_{π} is the cumulative reward value of all actions of s_t using π . Q is the state-action value function and the formula of Q_{π} is shown in (21). The policy π is utilized to define the decision-making process of the agent in RL [\(Nguyen, Nguyen, & Nahavandi, 2020\)](#). π is a mapping function from the current state s_t to the action a . The policy is deterministic when $p(a|s) = 1$, and the policy is stochastic when $p(a|s) < 1$. In this work, the output result of the neural network regarded as the policy $\pi(a|s)$ is described as a conditional probability of performing a under the state of s , as shown in (22).

$$Q_{\pi}(s_t, a_t) = E_A[U_t | S_t = s_t, A_t = a_t] \quad (21)$$

$$\pi(a|s) = p(a|s) \quad (22)$$

Where E_A stands for expectation, U_t is the return value on the trajectory t . The formula of U_t is shown in (23).

$$U_t = R_t + \gamma \times R_{t-1} + \gamma^2 \times R_{t-2} + \gamma^3 \times R_{t-3} + \dots + \gamma^{LP-1} \times R_{t-LP+1} \quad (23)$$

Where γ is the discount factor, and R_t is the rewarding result at time t . The mechanism is utilized to control the parameter of LSTM without the process of extensive training in this paper.

In the evolution process of the following flock, the mechanism of multi-strategy learning is utilized as an evolution method to generate new solutions. LSTM is utilized as a strategy selector to output the selection probabilities of different mutation strategies. PG, which is a model optimizer learning the parameters of the LSTM online, improves the selector of mutation strategies dynamically.

4. Experimental study and discussion

The CEC2017 benchmark test suite (CEC'17) is utilized to verify the pros and cons of different algorithms for real-valued continuous optimization problems. The CEC'17 contains four dimensions of information namely $10D$, $30D$, $50D$, $100D$ (D is the dimension). According to the property of function, thirty functions are divided into unimodal functions (f_1-f_3), multimodal functions (f_4-f_{10}), hybrid functions ($f_{11}-f_{20}$), and composition functions ($f_{21}-f_{30}$). f_2 was removed in the CEC'17 due to the instability in higher dimensional and the significant differences in the results achieved by different programming languages. The absolute value of the error between the experimental value and the optimal value is recorded to evaluate the performance of the algorithm. The experiment value is the optimal value when the error is zero. All the experiments are implemented on MATLAB R2016b. All the algorithms are executed on an Intel(R) Xeon(R) Gold 5218 CPU @ 2.30 GHz with 64 GB RAM in the Windows Server 2019 Standard 64-bit Operating System.

4.1. Parameter experiment

The experimental results are extensively affected by the selection of the parameters. In this paper, the design of the experiment (DOE) (Arnouts, Goos, & Jones, 2010) is utilized to calibrate the parameters of CMBO-PG. The parameters τ and ζ of EDA have been proved by a large number of experiments in (Liang et al., 2020) that the optimal values are $\tau = 0.35$, $\zeta = 5$. The parameters to be calibrated are the population size N , the number of flaps m , and the number of shared solutions x . The levels of parameter intervals are set as $N = \{20 \times D, 30 \times D, 40 \times D, 50 \times D, 60 \times D\}$, $m = \{5, 10, 20, 30\}$, $x = \{0.05 \times N, 0.15 \times N, 0.25 \times N, 0.35 \times N\}$. A total of $5 \times 4 \times 4 = 80$ parameter combinations are verified in the experiment, and each combination is run 20 times. The analysis of variance (ANOVA) of orthogonal experiments about N , m , and x are shown in Table 2.

The P -values of N and m are smaller than 0.05 in Table 2, which shows that both N and m are key parameters. The adjustment of N and m resulted in a significant change for the CMBO-PG with a 95% confidence level (Zhao, Zhu, et al., 2022). The F -ratio value of N and m are the higher than other parameters, which means N and m perform the most influence on the CMBO-PG. The main effects plot of parameters for CMBO-PG is shown in Fig. 5. The mean values of the algorithm performing on 29 functions are the minimum when $N = 30 \times D, m = 20, x = 0.05 \times N$. The results of Fig. 5 indicate the optimal combination of the parameter is $N = 30 \times D, m = 20, x = 0.05 \times N$. The p -value of $N \times m$ is less than 0.05 according to Table 2, which means the interaction relationship between N and m needs to further study through the interaction diagram (Zhao, Ding, Wang, Cao, & Tang, 2021). The interaction plot of $N \times m$ is shown in Fig. 6. The optimal parameter combination of $N \times m$ is $N = 30 \times D, m = 20$, which is consistent with the main effect plot. Thus, the parameter combination of CMBO-PG is $N = 30 \times D, m = 20, x = 0.05 \times N$.

Table 2
ANOVA results of parameters for CMBO-PG.

Source	Sum of Squares	Degrees of freedom	Mean Square.	F-ratio	P-value
N	4750.6	4	1187.65	470.81	0
m	4365.4	3	1455.13	576.84	0
x	5.4	3	1.81	0.72	0.548
$N \times m$	3111	12	259.25	102.77	0
$N \times x$	12.9	12	1.07	0.43	0.9426
$m \times x$	14.5	9	1.61	0.64	0.7556
Error	90.8	36	2.52		
Total	12350.7	79			

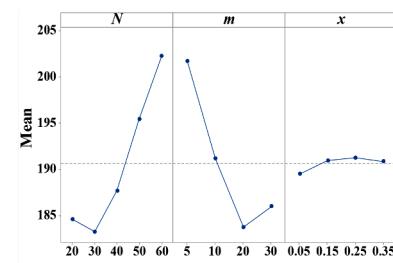


Fig. 5. Main effects plot of parameters for CMBO-PG.

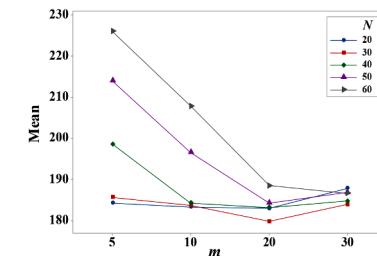


Fig. 6. The interaction plots of $N \times m$ for CMBO-PG.

4.2. Comparison experiment with different algorithms

CMBO-PG is compared with 12 different algorithms on the CEC'17. Each function runs 51 times independently with the same evaluation times ($10,000 \times D$) to ensure the fairness of the experiment. The mean values (Mean) and standard deviation (Std) of each function obtained for four dimensions ($10D$, $30D$, $50D$, $100D$) are recorded in Tables 4–7. The optimal value on each function is expressed in bold. The comparison algorithms are MBO (Duman et al., 2012), IMFMBO (Tongur & Ülker, 2018), EMBO (Segredo et al., 2018), AAVS (Ren et al., 2018), AGSK (Mohamed, Hadi, Mohamed, Awad, & Ieee, 2020), jSO (Brest, Maucec, Boskovic, & Ieee, 2017), MPEDE (Wu et al., 2016), CJADE (Gao et al., 2021), RLBSO (Zhao et al., 2022), TAPSO (Xia et al., 2020), XPSO (Xia et al., 2020), and OLBSO (Ma, Cheng, & Shi, 2021), and the parameters setting of all comparison algorithms are shown in Table 3. In addition to the basic MBO, two variants of MBO are selected as the comparison algorithm. IMFMBO is utilized as a comparison algorithm due to the idea of dividing the flock. EMBO is proven to exhibit good performance on large-scale continuous problems. AAVS is an excellent EDA algorithm, which is similar to the generation strategy of the leading flock. AGSK, which is a state-of-art algorithm, is the second-ranked algorithm in the CEC2021 benchmark test suite. As one of the best variants of DE, jSO, which ranks second on the CEC'17, performs well in different dimensions. RLBSO and OLBSO are the latest meta-heuristics integrating the mechanism of machine learning. Other state-of-art algorithms including CJADE, MPEDE, TAPSO, and XPSO are selected for comparative study. The error value is regarded as zero when it is smaller than 10^{-8} .

From Tables 4–7, it can be seen that CMBO-PG obtains the optimal solution of f_9 in different dimensions. For unimodal functions, the performance of CMBO-PG is not advantage to the comparison algorithms. The reason is that the complex mechanisms cost too much evaluation times. For multimode functions in all dimensions, CMBO-PG has an absolute performance advantage over the comparison algorithms. Due to the characteristics of the problem, the comparison algorithm is deceived by the extreme points to jump out of the local optimum in the multimode functions. The experimental results show that the co-evolution mechanism of CMBO-PG keeps the balance of exploration and exploitation. For hybrid functions and composition functions, the advantage of CMBO-PG performs is not obvious in the lower dimensions,

Table 3

The parameters setting of comparison algorithms.

Algorithm name	Year	Parameter	Parameter setting
MBO (Duman et al., 2012)	2012	N : population size k : neighbor size m : the number of flaps x : the number of shared neighbors	$N = 51$; $k = 3$; $m = 10$; $x = 1$;
IMFMBO (Tongur & Ülker, 2018)	2018	P : population size k : neighbor size m : the number of flaps x : the number of shared neighbors z : flock size c_1 : learning rate c_2 : learning rate	$P = 71$; $k = 3$; $m = 30$; $x = 1$; $z = 15$; $c_1 = 1$; $c_2 = 1$;
EMBO (Segredo et al., 2018)	2018	N : population size k : neighbor size m : the number of flaps x : the number of shared neighbors F : scaling factor CR : crossover rate	$N = 150$; $k = 9$; $m = 5$; $x = 2$; $F = 0.5$; $CR = 0.5$;
AAVS (Ren et al., 2018)	2018	p : population size τ : selection ratio α : the enlarging factor	$p = 1000$; $\tau = 0.35$; $\alpha = 1.7$;
AGSK (Mohamed et al., 2020)	2020	NP : population size p : the suitable rate of NP K_{wp} : the probability parameter c : learning rate	$NP = 20 \times D$; $p = 0.05$; $K_{wp} = [0.85, 0.05, 0.05, 0.05]$; $c = 0.05$;
jSO (Brest et al., 2017)	2017	N : population size p_{max} : max linearly decreases rate p_{min} : min linearly decreases rate H : historical memory size M_F : historical memory values	$N = 25 \times \log(D) \sqrt{(D)}$; $p_{max} = 0.25$; $p_{min} = p_{max}/2$; $H = 5$; $M_F = 0.3$;
MPEDE (Wu et al., 2016)	2016	NP : population size ng : the numbers of generation	$NP = 250$; $ng = 20$;
CJADE (Gao et al., 2021)	2021	N : population size CR_m : crossover rate F_m : scaling factor	$N = 100$; $CR_m = 0.5$; $F_m = 0.5$;
RLBSO (Zhao et al., 2022)	2022	N : population size E : elite individual $CP1$: crossover rate $CP2$: descending factor	$N = 100$; $E = 10$; $CP1 = 0.9$; $CP2 = 0.2$;
TAPSO (Xia et al., 2020)	2020	ω : control parameter p_c : crossover rate p_m : mutation probability M : the size of the archives	$\omega = 0.7298$; $p_c = 0.5$; $p_m = 0.02$; $M = N/4$;
XPSO (Xia, et al., 2020)	2020	η : the degree parameter $Stagmax$: the frequency of reselecting neighbors p : control parameter	$\eta = 0.2$; $Stagmax = 5$; $p = 5$;
OLBSO (Ma et al., 2021)	2021	N : population size M : the number of clusters $L-Pg$: the probability of activating exploration OD engine $L-T$: the probability of activating exploitation OD engine	$N = 100$; $m = 5$; $L-Pg = 4$; $L-T = 15$;
CMBO-PG		N : population size m : the number of flaps x : the number of shared neighbors τ : selection ratio ζ : shifting factor	$NP = 20 \times D$; $m = 20$; $x = 0.05 \times N$; $\tau = 0.35$; $\zeta = 5$;

but significant in the higher dimensions. In general, the experimental results of the proposed algorithm are better than comparison algorithms, especially in high-dimensional problems. High-dimensional problems contain a large number of decision variables, which result in an exponential increase in complexity. The multi-strategy learning mechanism of CMBO-PG includes the LSTM network and PG to adjust the selection probability of different strategies dynamically in each generation of algorithm evolution. Although the characteristic of high-dimensional increase the problem complexity, PG and LSTM based on online learning adjust the algorithm adaptively to obtain higher performance solutions.

4.3. Convergence and stability experiments

The convergence and stability of the algorithm are significant evaluation factors for judging the pros and cons of the algorithm. $f1$, $f8$, $f16$, $f29$ are the represents selected from different types of functions to draw convergence diagrams. The convergence diagrams of $10D$, $30D$, $50D$, and $100D$ are shown in Figs. 7–10. Fig. 7 illustrates that AAVS, AGSK, and jSO are forceful competitors although CMBO-PG performs fast convergence speed and high convergence accuracy. The reason for the phenomenon is that the problem is not complex enough in the lower dimensions. The role of exquisite design, particularly the co-evolution of multi-flock and the mechanism of multi-strategy learning, is underestimated although CMBO-PG still performs well. CMBO-PG has obvious advantages in convergence accuracy from Figs. 8–10 specifically the appearance of the quadratic convergence process in multimodal and hybrid problems. Multiple local attraction basins are present on the multimodal and hybrid functions, which deceive most algorithms into falling into the local optimum in the optimization search process. The quadratic convergence embodies that CMBO-PG stepped out of the local attraction basin to find a better solution due to the guidance of the mechanism of multi-strategy learning.

The box plots drawn by the experiment results of 51 independent runs of the same function are utilized to observe the stability of the algorithm. The experimental results of each run are distinct although with the same algorithm due to the randomness of the optimization process. The stability of the algorithm is reflected according to the fluctuation of experiment results in 51 independent runs. The box plots are shown in Figs. 11–14. The flat box plot means that the algorithm is stable due to the error between the upper quartile and the lower quartile being tiny. The red line in the middle is the median of 51 results. The smaller the median value, the better the performance of the algorithm. The box plots of CMBO-PG are flat in all dimensions of the functions for unimodal, multimodal, hybrid, and composition functions from Figs. 11–14. It shows that the CMBO-PG is stable compared to the comparison algorithm.

4.4. Comparison using the Friedman test and Wilcoxon test

A nonparametric experiment called the rank-sum test is utilized to analyze the significant difference between the control algorithm and the comparison algorithms. The rank-sum test among multiple samples is a statistical analysis method utilizing the sample rank instead of the sample value (Zhao et al., 2022). In this section, the Friedman test and Wilcoxon test are introduced to evaluate the significant difference between CMBO-PG and the comparison algorithms. The Friedman test is to test the significant difference between multiple algorithms, whereas the Wilcoxon test is between the control algorithm and each comparison algorithm. The algorithm results are verified by both statistical analyses within the 95% and 90% confidence interval, where CMBO-PG is the control algorithm. The mean rank of CMBO-PG is smaller than the comparison algorithms (except for $10D$), which means the performance of CMBO-PG is better than other algorithms. The disadvantage of the Friedman test is that the degree of significance is still unknown although obtained the difference between various algorithms. In this regard, a critical difference (CD) is introduced via a post-hoc test called the

Table 4

Mean and Standard Deviation of CMBO-PG and Comparison Algorithm on CEC 17 Benchmark Test Suite (10D).

<i>f</i>		MBO	IMFMBO	EMBO	AAVS	AGSK	jSO	MPEDE	CJADE	RLBSO	TAPSO	XPSO	OLBSO	CMBO-PG
<i>f</i> 1	<i>Mean</i>	1.11E+09	3.83E+07	2.58E+01	3.80E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	1.99E+03	1.94E+03	1.36E+03	0.00E+00	
	<i>Std</i>	4.71E+08	1.46E+07	1.84E+02	1.52E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	2.46E+03	2.17E+03	1.88E+03	0.00E+00	
<i>f</i> 3	<i>Mean</i>	1.61E+04	1.10E+04	4.15E+01	3.43E-05	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	1.73E+03	0.00E+00
	<i>Std</i>	6.91E+03	3.37E+03	1.42E+02	1.36E-05	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	1.59E+03	0.00E+00
<i>f</i> 4	<i>Mean</i>	8.24E+01	9.79E+00	6.38E+00	1.62E-01	0.00E+00	0.00E+00	0.00E+00	0.00E+00	3.13E-01	1.12E-05	2.11E-01	1.25E+01	8.80E-02
	<i>Std</i>	3.01E+01	1.63E+00	1.63E+00	2.67E-01	0.00E+00	0.00E+00	0.00E+00	0.00E+00	1.08E+00	1.17E-05	1.98E-01	1.79E+01	1.38E-01
<i>f</i> 5	<i>Mean</i>	4.33E+01	1.82E+01	9.84E+00	2.20E+00	6.91E+00	2.50E+00	6.31E+00	3.08E+00	9.31E+00	1.22E+01	6.87E+00	3.60E+01	4.05E-01
	<i>Std</i>	8.25E+00	3.67E+00	5.14E+00	1.17E+00	2.20E+00	1.01E+00	1.65E+00	7.91E-01	4.32E+00	5.45E+00	3.11E+00	1.45E+01	7.19E-01
<i>f</i> 6	<i>Mean</i>	1.94E+01	3.92E+00	0.00E+00	7.89E-04	6.65E-06	0.00E+00	2.22E-05	0.00E+00	1.66E-03	6.15E-04	5.91E-07	7.57E+00	0.00E+00
	<i>Std</i>	3.86E+00	8.14E-01	0.00E+00	3.28E-04	2.07E-05	0.00E+00	6.15E-06	0.00E+00	5.41E-03	1.46E-03	1.96E-06	5.33E+00	0.00E+00
<i>f</i> 7	<i>Mean</i>	7.12E+01	4.35E+01	2.28E+01	1.27E+01	1.81E+01	1.31E+01	1.71E+01	1.37E+01	1.93E+01	1.94E+01	1.77E+01	5.71E+01	1.07E+01
	<i>Std</i>	1.25E+01	4.69E+00	7.29E+00	2.94E+00	2.20E+00	7.45E-01	1.59E+00	1.11E+00	5.05E+00	4.32E+00	3.59E+00	1.96E+01	3.52E-01
<i>f</i> 8	<i>Mean</i>	4.04E+01	1.95E+01	1.04E+01	1.66E-01	8.34E+00	2.65E+00	6.28E+00	3.40E+00	8.80E+00	1.19E+01	6.67E+00	3.14E+01	1.58E-01
	<i>Std</i>	7.21E+00	3.16E+00	7.97E+00	3.92E-01	2.78E+00	1.10E+00	1.62E+00	8.28E-01	3.60E+00	5.24E+00	2.60E+00	1.21E+01	3.62E-01
<i>f</i> 9	<i>Mean</i>	2.21E+02	2.55E+01	1.07E-07	7.13E-05	7.02E-03	0.00E+00	0.00E+00	0.00E+00	1.16E-01	0.00E+00	1.99E+02	0.00E+00	
	<i>Std</i>	8.33E+01	1.09E+01	4.86E-07	3.64E-05	2.43E-02	0.00E+00	0.00E+00	0.00E+00	3.48E-01	0.00E+00	1.55E+02	0.00E+00	
<i>f</i> 10	<i>Mean</i>	1.07E+03	5.32E+02	7.76E+02	5.35E+01	3.10E+02	1.75E+02	2.69E+02	8.52E+01	5.63E+02	4.14E+02	2.85E+02	9.92E+02	1.18E+01
	<i>Std</i>	1.78E+02	1.44E+02	3.56E+02	1.03E+02	7.80E+01	1.26E+02	1.17E+02	6.21E+01	2.43E+02	1.86E+02	1.54E+02	3.44E+02	2.90E+01
<i>f</i> 11	<i>Mean</i>	5.13E+02	6.39E+01	2.94E+00	6.84E-01	1.25E-01	0.00E+00	2.23E+00	2.39E+00	2.73E+00	9.71E+00	2.47E+00	5.50E+01	9.62E-02
	<i>Std</i>	4.48E+02	7.63E+01	1.22E+00	6.16E-01	3.35E-01	0.00E+00	6.59E-01	7.31E-01	2.04E+00	6.94E+00	1.81E+00	4.83E+01	3.01E-01
<i>f</i> 12	<i>Mean</i>	2.61E+07	1.54E+06	3.95E+03	6.56E+01	4.56E+01	1.00E+01	1.01E+01	9.02E+01	1.63E+02	9.60E+03	1.23E+04	1.30E+06	6.77E+01
	<i>Std</i>	4.48E+02	7.63E+01	1.22E+00	6.16E-01	3.35E-01	0.00E+00	6.59E-01	7.31E-01	2.04E+00	6.94E+00	1.81E+00	4.83E+01	3.01E-01
<i>f</i> 13	<i>Mean</i>	8.90E+06	2.50E+04	1.04E+01	1.83E+00	3.33E+00	3.77E+00	4.94E+00	3.85E+00	7.41E+00	4.63E+03	6.64E+03	9.29E+03	3.78E+00
	<i>Std</i>	1.01E+07	1.93E+04	3.57E+00	2.98E+00	2.17E+00	2.19E+00	1.98E+00	2.57E+00	2.66E+00	3.84E+03	5.68E+03	7.57E+03	3.07E+00
<i>f</i> 14	<i>Mean</i>	1.01E+05	6.68E+02	1.33E+01	9.57E-01	4.32E-02	2.78E-01	5.18E+00	5.02E-01	1.73E+01	1.41E+03	3.91E+01	2.23E+03	3.44E+00
	<i>Std</i>	1.63E+05	5.16E+02	9.31E+00	2.87E+00	1.96E-01	1.38E+00	1.40E+00	4.09E-01	8.59E+00	1.14E+03	1.58E+01	3.07E+03	7.29E+00
<i>f</i> 15	<i>Mean</i>	2.60E+05	1.87E+03	1.40E+00	1.01E+00	2.44E-02	2.75E-01	6.91E-01	4.17E-01	1.91E+00	2.08E+03	4.64E+01	3.34E+03	6.06E-01
	<i>Std</i>	4.51E+05	2.72E+03	1.91E+00	6.18E-01	3.18E-02	2.07E-01	1.96E-01	1.27E-01	1.40E+00	3.02E+03	7.88E+01	4.44E+03	6.34E-01
<i>f</i> 16	<i>Mean</i>	4.42E+02	5.73E+01	4.12E+00	1.64E+00	6.66E-01	8.94E-01	2.46E+00	1.08E+00	7.99E+01	1.47E+02	7.97E+01	2.52E+02	1.23E+00
	<i>Std</i>	1.00E+02	5.46E+01	1.74E+01	1.73E+00	3.22E-01	5.05E-01	9.44E-01	3.41E-01	8.41E+01	1.12E+02	9.39E+01	1.37E+02	6.31E-01
<i>f</i> 17	<i>Mean</i>	1.31E+02	2.86E+01	7.92E+00	2.66E+01	1.36E+00	8.61E+00	7.66E+00	4.06E-01	1.53E+01	3.14E+01	2.43E+01	6.76E+01	2.10E+01
	<i>Std</i>	7.73E+01	1.17E+01	1.11E+01	9.67E+00	1.15E+00	8.25E+00	2.03E+00	2.99E-01	1.44E+01	3.21E+01	1.51E+01	4.48E+01	8.10E+00
<i>f</i> 18	<i>Mean</i>	9.20E+06	2.40E+04	1.43E+01	6.59E-01	1.20E-01	2.41E-01	3.00E+00	3.09E-01	1.69E+01	7.67E+03	7.40E+03	1.84E+04	5.62E+00
	<i>Std</i>	9.27E+06	1.81E+04	8.22E+00	3.62E-01	1.53E-01	1.95E-01	1.29E+00	4.00E-01	7.49E+00	1.03E+04	6.61E+03	1.73E+04	8.79E+00
<i>f</i> 19	<i>Mean</i>	7.57E+05	3.37E+03	2.41E-01	6.31E-01	5.68E-02	1.54E-02	5.86E-01	3.58E-02	1.09E+00	5.12E+03	5.15E+01	4.81E+03	3.07E-01
	<i>Std</i>	1.81E+06	4.36E+03	5.96E-01	3.92E-01	4.12E-02	1.93E-02	1.89E-01	1.65E-02	8.09E-01	6.92E+03	8.01E+01	6.53E+03	3.59E-01
<i>f</i> 20	<i>Mean</i>	1.65E+02	2.52E+01	2.00E+00	7.42E+00	6.12E-02	7.43E-01	9.36E-01	2.72E-10	2.97E+01	7.10E+00	3.63E+01	7.57E+01	4.90E+00
	<i>Std</i>	4.39E+01	5.74E+00	5.45E+00	9.14E+00	1.40E-01	2.80E+00	5.35E-01	3.83E-10	4.18E+01	8.12E+00	4.81E+01	4.85E+01	7.75E+00
<i>f</i> 21	<i>Mean</i>	1.79E+02	1.34E+02	1.90E+02	1.99E+02	1.02E+02	1.28E+02	1.11E+02	1.52E+02	1.09E+02	1.96E+02	1.83E+02	1.50E+02	1.99E+02
	<i>Std</i>	4.79E+01	4.08E+01	4.73E+01	7.01E+00	1.56E+01	4.66E+01	3.27E+01	4.76E+01	3.02E+01	4.19E+01	4.68E+01	5.90E+01	1.13E+01
<i>f</i> 22	<i>Mean</i>	1.87E+02	9.74E+01	9.88E+01	1.00E+02	8.20E+01	1.00E+02	9.07E+01	9.83E+01	1.78E+02	1.01E+02	1.02E+02	1.06E+02	1.00E+02
	<i>Std</i>	4.44E+01	2.95E+01	1.25E+01	7.66E-04	3.59E+01	4.06E-02	2.87E+01	1.24E+01	2.38E+02	1.41E+00	8.39E-01	4.25E+00	0.00E+00
<i>f</i> 23	<i>Mean</i>	3.43E+02	3.22E+02	3.17E+02	3.01E+02	3.03E+02	3.02E+02	3.06E+02	3.05E+02	3.13E+02	3.17E+02	3.08E+02	3.48E+02	3.02E+02
	<i>Std</i>	9.12E+00	4.20E+00	9.26E+00	2.39E+00	4.41E+01	1.74E+00	1.91E+00	1.30E+00	4.38E+00	6.02E+00	4.85E+00	1.82E+01	1.94E+00
<i>f</i> 24	<i>Mean</i>	3.04E+02	2.61E+02	3.36E+02	3.13E+02	1.03E+02	2.81E+02	2.64E+02	2.65E+02	3.45E+02	3.40E+02	3.08E+02	3.19E+02	3.27E+02
	<i>Std</i>	8.61E+01	9.90E+01	4.35E+01	1.26E+01	6.06E+01	9.57E+01	1.07E+02	9.71E+01	6.32E+00	4.95E+01	7.69E+01	1.15E+02	8.55E+00
<i>f</i> 25	<i>Mean</i>	4.88E+02	4.18E+02	4.26E+02	4.03E+02	3.86E+02	4.10E+02	4.04E+02	4.17E+02	4.20E+02	4.30E+02	4.21E+02	4.31E+02	4.13E+02
	<i>Std</i>	1.76E+01	1.54E+01	2.26E+01	1.33E+01	5.84E+01	2.00E+01	1.60E+01	2.29E+01	2.32E+01	2.28E+01	2.35E+01	2.06E+01	2.10E+01
<i>f</i> 26	<i>Mean</i>	5.44E+02	3.29E+02	3.48E+02	3.00E+02	2.59E+02	3.00E+02	3.00E+02	3.00E+02	3.50E+02	3.86E+02	2.97E+02	4.89E+02	3.00E+02
	<i>Std</i>	1.01E+02	6.92E+01	1.13E+02	8.92E-01	1.04E+02	0.00E+00	0.00E+00	0.00E+00	1.26E+02	2.55E+02	1.43E+02	3.07E+02	0.00E+00
<i>f</i> 27	<i>Mean</i>	4.17E+02	3.79E+02	3.72E+02	3.94E+02	3.89E+02	3.89E+02	3.89E+02	3.89E+02	3.80E+02	3.96E+02	3.98E+02	4.17E+02	3.93E+02
	<i>Std</i>	5.80E+00	2.50E+00	1.10E+00	1.59E+00	7.83E-01	1.88E-01	6.01E-01	7.04E-01	3.02E+01	1.30E+01	1.39E+01	6.84E+01	2.58E+00
<i>f</i> 28	<i>Mean</i>	4.97E+02	3.73E+02	4.54E+02	3.00E+02	2.08E+02	3.31E+02	3.12E+02	3.79E+02	4.64E+02	4.82E+02	4.87E+02	4.96E+02	5.97E+02
	<i>Std</i>	7.58E+01	3.32E+01	6.04E+01	5.19E-03	1.38E+02	8.94E+01	5.84E+01	1.32E+02	3.31E+01	1.32E+02	1.40E+02	1.16E+02	1.35E+01
<i>f</i>														

Table 5

Mean and Standard Deviation of CMBO-PG and Comparison Algorithm on CEC'17 Benchmark Test Suite (30D).

<i>f</i>		MBO	IMFMBO	EMBO	AAVS	AGSK	jSO	MPEDE	CJADE	RLBSO	TAPSO	XPSO	OLBSO	CMBO-PG
<i>f</i> 1	<i>Mean</i>	1.71E+09	2.34E+09	1.96E+07	6.13E+01	0.00E+00	0.00E+00	0.00E+00	0.00E+00	1.63E−05	3.09E+03	3.09E+03	4.70E+04	0.00E+00
	<i>Std</i>	6.38E+08	4.07E+08	4.61E+07	3.69E+02	0.00E+00	0.00E+00	0.00E+00	0.00E+00	1.16E−04	3.25E+03	3.74E+03	3.22E+04	0.00E+00
<i>f</i> 3	<i>Mean</i>	1.43E+05	1.36E+05	1.52E+05	2.97E+02	2.02E+03	0.00E+00	0.00E+00	4.52E+03	3.35E+03	4.63E−07	9.28E−03	4.82E+03	0.00E+00
	<i>Std</i>	2.57E+04	2.16E+04	3.37E+04	8.02E+02	1.16E+03	0.00E+00	0.00E+00	1.26E+04	5.51E+03	3.30E−06	2.87E−02	8.85E+03	0.00E+00
<i>f</i> 4	<i>Mean</i>	2.39E+02	3.53E+02	7.81E+01	2.73E+00	5.86E+01	5.86E+01	4.93E+01	3.96E+01	2.44E+01	1.95E+01	1.17E+02	7.22E+01	5.97E+01
	<i>Std</i>	5.37E+01	5.83E+01	2.98E+01	3.11E+00	0.00E+00	0.00E+00	2.25E+01	2.90E+01	2.41E+01	2.77E+01	2.58E+01	2.09E+01	3.94E+00
<i>f</i> 5	<i>Mean</i>	1.30E+02	1.64E+02	1.18E+02	1.64E+01	7.20E+01	1.74E+01	2.79E+01	2.59E+01	5.63E+01	5.40E+01	4.28E+01	1.85E+02	1.54E+00
	<i>Std</i>	1.54E+01	1.55E+01	6.24E+01	5.13E+00	5.60E+00	1.28E+01	6.81E+00	3.85E+00	1.60E+01	1.33E+01	9.16E+00	4.73E+01	1.11E+00
<i>f</i> 6	<i>Mean</i>	1.31E+01	1.60E+01	5.97E−03	8.05E−03	3.83E−02	1.85E−08	0.00E+00	0.00E+00	1.16E−01	1.41E−02	4.05E−02	1.31E+01	0.00E+00
	<i>Std</i>	2.04E+00	1.59E+00	2.27E−02	2.06E−02	2.84E−02	5.78E−08	0.00E+00	0.00E+00	1.65E−01	8.44E−03	7.47E−02	6.37E+00	0.00E+00
<i>f</i> 7	<i>Mean</i>	2.23E+02	3.14E+02	1.90E+02	3.40E+01	1.16E+02	6.98E+01	5.62E+01	5.49E+01	8.30E+01	8.86E+01	8.25E+01	2.28E+02	3.25E+01
	<i>Std</i>	2.20E+01	2.46E+01	6.00E+01	6.96E+00	7.98E+00	1.11E+01	6.89E+00	4.05E+00	1.44E+01	1.75E+01	1.54E+01	3.80E+01	4.54E−01
<i>f</i> 8	<i>Mean</i>	1.18E+02	1.61E+02	1.26E+02	1.41E+01	8.53E+01	1.56E+01	2.91E+01	2.60E+01	5.33E+01	5.76E+01	4.30E+01	1.73E+02	1.44E+00
	<i>Std</i>	1.55E+01	1.64E+01	6.45E+01	4.26E+00	6.28E+00	1.24E+01	8.33E+00	3.78E+00	1.68E+01	1.37E+01	1.39E+01	4.39E+01	1.39E+00
<i>f</i> 9	<i>Mean</i>	8.55E+02	1.85E+03	6.89E+00	0.00E+00	5.67E+00	0.00E+00	7.02E−03	1.76E−03	1.32E+01	1.10E+02	5.40E+00	5.16E+03	0.00E+00
	<i>Std</i>	4.51E+02	4.63E+02	9.80E+00	0.00E+00	6.40E+00	0.00E+00	2.43E−02	1.25E−02	1.59E+01	1.56E+02	4.38E+00	2.21E+03	0.00E+00
<i>f</i> 10	<i>Mean</i>	3.49E+03	4.49E+03	6.88E+03	1.29E+03	3.68E+03	4.05E+03	2.70E+03	1.92E+03	2.97E+03	2.65E+03	2.74E+03	4.73E+03	1.28E+02
	<i>Std</i>	3.91E+02	3.09E+02	8.31E+02	2.94E+02	2.82E+02	3.79E+02	3.64E+02	2.54E+02	5.53E+02	5.52E+02	5.96E+02	7.02E+02	1.30E+02
<i>f</i> 11	<i>Mean</i>	4.56E+03	2.57E+03	7.61E+01	1.89E+01	2.49E+01	8.57E+00	2.12E+01	3.16E+01	2.18E+01	8.53E+01	8.45E+01	1.77E+02	1.10E+01
	<i>Std</i>	3.45E+03	1.50E+03	4.17E+01	7.09E+00	2.15E+01	1.75E+01	8.12E+00	2.57E+01	1.18E+01	3.70E+01	3.30E+01	6.21E+01	1.91E+01
<i>f</i> 12	<i>Mean</i>	2.72E+08	1.29E+08	1.18E+06	9.81E+06	9.74E+03	1.94E+02	9.94E+02	1.37E+03	3.05E+04	3.19E+04	1.39E+05	3.59E+06	2.42E+02
	<i>Std</i>	1.18E+08	4.01E+07	1.24E+06	3.97E+07	5.90E+03	1.23E+02	3.25E+02	9.42E+02	2.96E+04	5.09E+04	3.03E+05	2.85E+06	1.44E+02
<i>f</i> 13	<i>Mean</i>	3.20E+08	6.93E+07	1.47E+04	2.02E+01	5.64E+01	1.66E+01	2.05E+01	4.80E+01	2.20E+02	1.54E+04	1.45E+04	2.40E+04	1.18E+01
	<i>Std</i>	1.73E+08	2.76E+07	1.09E+04	1.49E+01	2.19E+01	6.58E+00	8.79E+00	3.27E+01	2.76E+02	1.55E+04	1.34E+04	1.68E+04	9.69E+00
<i>f</i> 14	<i>Mean</i>	2.28E+06	5.69E+05	1.77E+03	2.74E+01	3.07E+01	2.22E+01	1.27E+01	2.73E+03	4.50E+01	3.20E+03	5.79E+03	2.51E+04	2.05E+01
	<i>Std</i>	1.95E+06	4.28E+05	6.47E+03	4.72E+00	2.18E+00	1.66E+00	9.59E+00	5.19E+03	1.43E+01	4.44E+03	5.65E+03	3.67E+04	4.48E+00
<i>f</i> 15	<i>Mean</i>	8.51E+07	1.25E+07	3.38E+03	1.40E+01	1.37E+01	1.34E+00	7.55E+00	1.79E+02	7.30E+01	5.52E+03	4.71E+03	8.89E+03	4.64E+00
	<i>Std</i>	7.68E+07	8.96E+06	5.49E+03	8.16E+00	1.74E+00	9.57E−01	2.52E+00	1.02E+03	2.19E+02	7.13E+03	5.58E+03	7.39E+03	2.68E+00
<i>f</i> 16	<i>Mean</i>	1.53E+03	1.28E+03	1.46E+03	1.89E+02	4.64E+02	1.85E+02	3.70E+02	4.57E+02	9.17E+02	8.29E+02	6.01E+02	1.48E+03	1.22E+01
	<i>Std</i>	2.64E+02	2.29E+02	6.39E+02	1.39E+02	7.34E+01	1.03E+02	1.82E+02	1.59E+02	2.92E+02	2.95E+02	2.30E+02	3.08E+02	4.18E+00
<i>f</i> 17	<i>Mean</i>	7.29E+02	5.28E+02	1.69E+02	4.17E+01	1.25E+02	8.15E+01	5.28E+01	7.42E+01	4.33E+02	3.37E+02	1.64E+02	7.29E+02	2.49E+01
	<i>Std</i>	1.72E+02	1.61E+02	1.19E+02	7.16E+00	2.12E+01	1.30E+01	1.56E+01	2.67E+01	2.15E+02	1.71E+02	7.63E+01	2.51E+02	5.00E+00
<i>f</i> 18	<i>Mean</i>	5.31E+06	2.09E+06	1.61E+05	2.67E+01	3.62E+01	2.10E+01	2.36E+01	6.72E+03	9.50E+03	3.58E+04	1.37E+05	2.70E+05	2.30E+01
	<i>Std</i>	3.74E+06	1.65E+06	3.01E+05	4.30E+00	6.79E+00	5.35E−01	5.35E+00	3.52E+04	8.67E+03	1.84E+04	7.37E+04	2.07E+05	1.47E+00
<i>f</i> 19	<i>Mean</i>	1.04E+08	1.38E+07	2.14E+03	1.33E+01	1.33E+01	5.88E+00	6.88E+00	3.05E+02	1.59E+01	4.13E+03	6.40E+03	2.98E+04	5.69E+00
	<i>Std</i>	6.80E+07	8.08E+06	3.05E+03	3.90E+00	2.21E+00	2.27E+00	1.91E+00	2.02E+03	6.05E+00	6.40E+03	6.96E+03	5.22E+04	1.43E+00
<i>f</i> 20	<i>Mean</i>	7.30E+02	5.52E+02	2.33E+02	7.72E+01	2.01E+02	9.90E+01	5.46E+01	1.14E+02	4.43E+02	3.03E+02	1.73E+02	5.73E+02	2.72E+01
	<i>Std</i>	1.42E+02	1.43E+02	1.85E+02	6.52E+01	4.45E+01	2.86E+01	4.33E+01	5.43E+01	1.86E+02	1.29E+02	5.08E+01	2.21E+02	2.14E+01
<i>f</i> 21	<i>Mean</i>	3.24E+02	3.62E+02	3.30E+02	2.11E+02	2.69E+02	2.12E+02	2.28E+02	2.26E+02	2.59E+02	2.61E+02	2.46E+02	4.06E+02	2.03E+02
	<i>Std</i>	1.46E+01	1.37E+01	6.38E+01	5.60E+00	1.11E+01	6.52E+00	9.06E+00	3.97E+00	1.55E+01	1.38E+01	1.49E+01	6.11E+01	8.60E−01
<i>f</i> 22	<i>Mean</i>	8.36E+02	1.38E+03	2.31E+03	1.00E+02	1.00E+02	1.00E+02	1.00E+02	1.00E+02	3.38E+03	7.70E+02	4.19E+02	3.03E+03	1.00E+02
	<i>Std</i>	1.17E+03	3.07E+03	3.73E−05	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	6.06E+02	1.30E+03	9.12E+02	2.56E+03	0.00E+00
<i>f</i> 23	<i>Mean</i>	4.82E+02	5.26E+02	4.85E+02	3.42E+02	4.00E+02	3.56E+02	3.80E+02	3.73E+02	4.12E+02	4.38E+02	3.98E+02	6.10E+02	3.55E+02
	<i>Std</i>	1.80E+01	2.20E+01	6.93E+01	1.26E+01	4.21E+00	5.37E+00	1.05E+01	5.24E+00	1.56E+01	2.18E+01	1.85E+01	6.73E+01	6.45E+00
<i>f</i> 24	<i>Mean</i>	6.02E+02	6.50E+02	5.89E+02	3.31E+02	4.62E+02	4.30E+02	4.43E+02	4.42E+02	4.83E+02	5.17E+02	4.73E+02	6.78E+02	4.26E+02
	<i>Std</i>	2.06E+01	1.78E+01	7.08E+01	3.50E+01	4.09E+00	2.66E+00	6.69E+00	4.76E+00	1.38E+01	2.72E+01	3.37E+01	6.89E+01	2.96E+00
<i>f</i> 25	<i>Mean</i>	4.87E+02	5.58E+02	3.87E+02	3.89E+02	3.87E+02	3.87E+02	3.87E+02	3.87E+02	3.79E+02	3.94E+02	3.99E+02	4.07E+02	3.87E+02
	<i>Std</i>	3.35E+01	3.69E+01	1.55E+01	3.97E+00	1.76E−02	8.96E−03	6.05E−02	5.35E−01	7.39E−01	1.55E+01	1.49E+01	2.26E+01	4.10E−03
<i>f</i> 26	<i>Mean</i>	1.81E+03	2.34E+03	1.92E+03	3.43E+02	1.60E+03	9.73E+01	1.24E+03	1.20E+03	1.42E+03	1.50E+03	9.07E+02	3.38E+03	8.68E+02
	<i>Std</i>	7.98E+02	8.38E+02	6.48E+02	1.92E+02	5.75E+01	5.79E+01	1.13E+02	8.20E+01	2.55E+02	7.44E+02	6.33E+02	1.27E+03	5.44E+01
<i>f</i> 27	<i>Mean</i>	5.50E+02	5.01E+02	4.84E+02	4.83E+02	4.99E+02	5.00E+02	5.02E+02	5.04E+02	5.00E+02	5.20E+02	5.36E+02	4.96E+02	5.08E+02
	<i>Std</i>	1.15E+01	1.59E+01	1.74E+01	6.62E+00	7.22E+00	6.56E+00	7.26E+00	8.10E+00	2.10E−04	1.61E+01	1.78E+01	7.63E+00	4.55E+00
<i>f</i> 28	<i>Mean</i>	5.51E+02	6.60E+02	4.50E+02	3.00E+02	3.14E+02	3.11E+02	3.34E+02	3.34E+02	4.99E+02	3.44E+02			

Table 6

Mean and Standard Deviation of CMBO-PG and Comparison Algorithm on CEC'17 Benchmark Test Suite (50D).

<i>f</i>		MBO	IMFMBO	EMBO	AAVS	AGSK	jSO	MPEDE	CJADE	RLBSO	TAPSO	XPSO	OLBSO	CMBO-PG
<i>f</i> 1	<i>Mean</i>	1.62E+09	1.23E+10	3.08E+08	2.19E+07	9.21E-01	0.00E+00	0.00E+00	0.00E+00	6.77E-01	3.94E+03	3.66E+03	2.82E+05	0.00E+00
	<i>Std</i>	6.38E+08	1.62E+09	4.65E+08	1.31E+08	2.12E+00	0.00E+00	0.00E+00	0.00E+00	4.50E+00	4.37E+03	4.66E+03	1.72E+05	0.00E+00
<i>f</i> 3	<i>Mean</i>	2.84E+05	2.79E+05	3.24E+05	7.53E+03	5.99E+04	0.00E+00	3.13E-04	2.35E+04	1.37E+05	2.23E+02	2.70E+02	2.32E+01	0.00E+00
	<i>Std</i>	3.80E+04	3.57E+04	5.26E+04	4.05E+03	1.33E+04	0.00E+00	1.52E-03	3.91E+04	5.95E+04	4.32E+02	1.81E+02	8.06E+00	0.00E+00
<i>f</i> 4	<i>Mean</i>	3.40E+02	1.46E+03	1.47E+02	9.55E+01	1.02E+02	4.51E+01	5.73E+01	4.34E+01	3.92E+01	4.64E+01	2.38E+02	1.51E+02	4.19E+01
	<i>Std</i>	6.42E+01	2.82E+02	5.14E+01	5.16E+01	5.76E+01	4.23E+01	4.79E+01	4.75E+01	2.16E+01	4.54E+01	4.73E+01	4.72E+01	3.63E+01
<i>f</i> 5	<i>Mean</i>	2.03E+02	3.78E+02	2.10E+02	4.34E+01	2.10E+02	1.75E+01	5.27E+01	5.32E+01	1.20E+02	1.29E+02	9.13E+01	3.95E+02	2.58E+00
	<i>Std</i>	2.08E+01	2.33E+01	1.39E+02	1.08E+01	9.04E+00	1.33E+01	1.08E+01	7.71E+00	3.28E+01	2.63E+01	2.21E+01	1.03E+02	1.51E+00
<i>f</i> 6	<i>Mean</i>	1.01E+01	2.68E+01	1.50E-01	4.58E-01	1.95E-01	1.02E-06	1.68E-03	0.00E+00	1.66E+00	2.21E-02	8.23E-01	1.24E+01	2.39E-07
	<i>Std</i>	1.47E+00	2.44E+00	1.55E-01	6.78E-01	9.29E-02	1.47E-06	4.97E-03	0.00E+00	2.10E+00	1.47E-02	7.05E-01	4.57E+00	3.13E-08
<i>f</i> 7	<i>Mean</i>	3.52E+02	7.91E+02	3.93E+02	5.92E+01	2.97E+02	1.60E+02	1.07E+02	9.98E+01	1.67E+02	1.70E+02	1.76E+02	4.36E+02	1.14E+02
	<i>Std</i>	2.59E+01	5.01E+01	1.23E+02	3.90E+00	1.44E+01	3.13E+01	1.07E+01	7.28E+00	3.12E+01	2.65E+01	3.22E+01	9.00E+01	8.50E+01
<i>f</i> 8	<i>Mean</i>	1.99E+02	3.72E+02	2.48E+02	4.45E+01	2.05E+02	1.74E+01	5.67E+01	5.34E+01	1.13E+02	1.25E+02	9.24E+01	3.36E+02	3.14E+00
	<i>Std</i>	2.30E+01	2.48E+01	1.42E+02	1.19E+01	1.00E+01	5.75E+00	1.14E+01	6.22E+00	2.88E+01	2.49E+01	2.20E+01	9.26E+01	1.68E+00
<i>f</i> 9	<i>Mean</i>	2.28E+03	9.88E+03	8.46E+02	2.16E+00	1.22E+01	0.00E+00	9.97E-01	1.17E+00	3.83E+02	6.79E+02	1.24E+02	1.28E+04	0.00E+00
	<i>Std</i>	1.22E+03	2.36E+03	9.80E+02	1.58E+00	8.20E+00	0.00E+00	1.02E+00	1.22E+00	5.04E+02	4.20E+02	1.11E+02	4.78E+03	0.00E+00
<i>f</i> 10	<i>Mean</i>	5.45E+03	8.83E+03	1.30E+04	2.90E+03	8.66E+03	8.79E+03	4.95E+03	3.77E+03	5.86E+03	4.35E+03	5.16E+03	7.76E+03	1.75E+02
	<i>Std</i>	4.90E+02	3.68E+02	1.09E+03	6.19E+02	3.44E+02	4.52E+02	6.16E+02	2.39E+02	7.90E+02	7.17E+02	8.34E+02	8.50E+02	9.19E+01
<i>f</i> 11	<i>Mean</i>	1.41E+04	1.22E+04	3.52E+02	8.95E+01	4.87E+01	2.81E+01	1.02E+02	1.39E+02	5.44E+01	1.42E+02	1.79E+02	3.10E+02	2.28E+01
	<i>Std</i>	7.31E+03	4.81E+03	2.01E+02	1.91E+01	2.52E+00	3.79E+00	2.88E+01	3.46E+01	2.22E+01	4.23E+01	4.54E+01	8.46E+01	2.71E+00
<i>f</i> 12	<i>Mean</i>	7.52E+08	2.14E+09	1.92E+07	4.49E+08	7.38E+04	1.74E+03	1.06E+04	6.02E+03	2.96E+05	7.22E+04	2.77E+06	1.24E+07	1.07E+03
	<i>Std</i>	2.74E+08	5.45E+08	4.03E+07	7.55E+08	4.99E+04	4.30E+02	7.51E+03	3.22E+03	2.47E+05	5.06E+04	6.52E+06	5.88E+06	2.76E+02
<i>f</i> 13	<i>Mean</i>	4.20E+08	7.56E+08	4.89E+03	8.62E+04	2.81E+02	3.40E+01	8.34E+01	3.93E+02	2.33E+03	4.26E+03	4.92E+03	3.99E+04	4.53E+01
	<i>Std</i>	1.89E+08	2.77E+08	5.64E+03	6.10E+05	7.47E+01	2.04E+01	4.91E+01	3.53E+02	2.89E+03	6.79E+03	6.01E+03	2.49E+04	3.21E+01
<i>f</i> 14	<i>Mean</i>	8.16E+06	4.81E+06	7.07E+04	4.26E+04	6.50E+01	2.44E+01	6.27E+01	6.75E+03	1.34E+04	1.35E+04	4.97E+04	8.54E+04	2.60E+01
	<i>Std</i>	7.41E+06	3.20E+06	8.86E+04	2.24E+05	6.10E+00	1.63E+00	1.48E+01	2.71E+04	3.12E+01	1.94E+04	1.12E+05	7.16E+04	3.46E+00
<i>f</i> 15	<i>Mean</i>	1.82E+08	2.49E+08	1.33E+04	1.58E+02	6.57E+01	2.39E+01	7.29E+01	3.01E+02	5.97E+02	6.48E+03	4.29E+03	1.24E+04	2.62E+01
	<i>Std</i>	1.06E+08	1.43E+08	1.56E+04	5.16E+01	1.17E+01	2.79E+00	3.52E+01	1.51E+02	1.72E+03	6.62E+03	4.56E+03	1.04E+04	7.33E+00
<i>f</i> 16	<i>Mean</i>	2.37E+03	2.69E+03	2.98E+03	4.25E+02	1.10E+03	6.90E+02	9.03E+02	8.38E+02	1.71E+03	1.28E+03	9.39E+02	2.42E+03	1.31E+02
	<i>Std</i>	2.94E+02	3.30E+02	7.78E+02	1.24E+02	7.74E+01	2.01E+02	2.84E+02	2.07E+02	3.43E+02	3.98E+02	2.89E+02	5.35E+02	1.24E+01
<i>f</i> 17	<i>Mean</i>	1.85E+03	2.03E+03	1.73E+03	3.74E+02	8.06E+02	5.15E+02	5.75E+02	6.16E+02	1.21E+03	9.75E+02	8.12E+02	1.94E+03	9.76E+01
	<i>Std</i>	2.40E+02	2.85E+02	6.00E+02	1.55E+02	5.37E+01	1.13E+02	2.14E+02	1.17E+02	3.12E+02	2.88E+02	2.31E+02	3.78E+02	7.11E+01
<i>f</i> 18	<i>Mean</i>	1.40E+07	1.52E+07	9.26E+05	5.97E+01	1.04E+03	2.50E+01	1.17E+02	1.31E+04	1.07E+05	3.50E+04	3.39E+05	8.30E+05	2.53E+01
	<i>Std</i>	9.91E+06	9.09E+06	1.09E+06	2.35E+01	8.08E+02	2.31E+00	9.03E+01	9.27E+04	7.74E+04	2.37E+04	5.63E+05	6.53E+05	1.98E+00
<i>f</i> 19	<i>Mean</i>	7.35E+07	6.68E+07	1.57E+04	3.34E+01	3.75E+01	1.46E+01	4.10E+01	1.41E+02	7.20E+02	1.39E+04	1.38E+04	2.09E+04	1.57E+01
	<i>Std</i>	3.62E+07	3.21E+07	1.11E+04	9.04E+00	4.89E+00	3.45E+00	1.78E+01	4.95E+01	4.06E+03	8.13E+03	8.71E+03	2.14E+04	2.85E+00
<i>f</i> 20	<i>Mean</i>	1.48E+03	1.53E+03	1.71E+03	9.68E+01	6.45E+02	3.87E+02	4.01E+02	4.44E+02	9.07E+02	5.87E+02	5.19E+02	1.41E+03	2.58E+01
	<i>Std</i>	3.03E+02	2.21E+02	5.68E+02	4.21E+01	6.62E+01	8.90E+01	1.85E+02	1.16E+02	2.50E+02	2.93E+02	2.46E+02	3.85E+02	3.59E+00
<i>f</i> 21	<i>Mean</i>	4.10E+02	5.82E+02	4.73E+02	2.39E+02	3.98E+02	2.18E+02	2.52E+02	2.57E+02	3.23E+02	3.28E+02	2.90E+02	5.35E+02	2.07E+02
	<i>Std</i>	1.98E+01	2.47E+01	1.36E+02	8.94E+00	9.11E+00	5.25E+00	1.18E+01	8.54E+00	3.03E+01	2.89E+01	1.87E+01	1.03E+02	2.03E+00
<i>f</i> 22	<i>Mean</i>	6.11E+03	9.03E+03	1.36E+04	1.00E+02	4.69E+03	3.87E+03	3.15E+03	3.58E+03	6.87E+03	5.19E+03	3.95E+03	8.43E+03	1.00E+02
	<i>Std</i>	5.27E+02	1.83E+03	1.17E+03	2.55E-03	4.55E+03	4.39E+03	2.78E+03	1.52E+03	1.00E+03	1.51E+03	2.90E+03	1.47E+03	1.20E-08
<i>f</i> 23	<i>Mean</i>	6.67E+02	8.61E+02	7.12E+02	4.35E+02	5.80E+02	4.40E+02	4.80E+02	4.80E+02	5.39E+02	6.04E+02	5.35E+02	9.42E+02	4.28E+02
	<i>Std</i>	2.26E+01	2.57E+01	1.41E+02	2.07E+01	6.96E+00	9.26E+00	1.27E+01	1.00E+01	4.75E+01	3.57E+01	5.10E+01	1.13E+02	5.73E+00
<i>f</i> 24	<i>Mean</i>	8.45E+02	1.05E+03	8.62E+02	4.53E+02	6.12E+02	5.12E+02	5.43E+02	5.42E+02	6.29E+02	7.05E+02	6.55E+02	9.77E+02	5.07E+02
	<i>Std</i>	4.11E+01	3.57E+01	1.30E+02	5.13E+01	5.98E+00	4.45E+00	1.23E+01	1.08E+01	3.52E+01	4.77E+01	9.21E+01	1.06E+02	5.00E+00
<i>f</i> 25	<i>Mean</i>	7.10E+02	1.61E+03	5.10E+02	4.97E+02	4.82E+02	4.81E+02	5.25E+02	5.23E+02	4.42E+02	5.42E+02	5.98E+02	4.93E+02	4.80E+02
	<i>Std</i>	5.40E+01	2.13E+02	5.48E+01	4.26E+01	4.24E+00	4.76E+00	2.65E+01	3.53E+01	1.98E+01	3.72E+01	3.01E+01	4.08E+01	5.05E-03
<i>f</i> 26	<i>Mean</i>	3.36E+03	5.52E+03	3.85E+03	4.44E+02	2.68E+03	1.19E+03	1.60E+03	1.64E+03	2.44E+03	2.49E+03	1.18E+03	5.27E+03	8.63E+02
	<i>Std</i>	8.19E+02	7.00E+02	1.44E+03	4.05E+02	7.17E+01	7.24E+01	1.69E+02	1.19E+02	4.82E+02	1.04E+03	9.54E+02	2.22E+03	1.05E+02
<i>f</i> 27	<i>Mean</i>	7.01E+02	7.20E+02	5.02E+02	5.17E+02	5.89E+02	5.13E+02	5.47E+02	5.53E+02	5.00E+02	6.24E+02	7.16E+02	5.05E+02	5.43E+02
	<i>Std</i>	4.67E+01	2.56E+02	1.55E+01	2.60E+01	4.30E+01	9.31E+00	2.62E+01	2.79E+01	1.92E-04	4.74E+01	9.21E+01	2.03E+01	1.26E+01
<i>f</i> 28	<i>Mean</i>	6.57E+02	1.86E+03	4.87E+02	4.77E+02	4.61E+02	4.63E+02	4.92E+02	4.93E+02	4.93E+02	4.95E+02	5.44E+02	4.90E+02	4.59E+02
	<i>Std</i>	6.72E+01	3.47E+02											

Table 7

Mean and Standard Deviation of CMBO-PG and Comparison Algorithm on CEC'17 Benchmark Test Suite (100D).

<i>f</i>		MBO	IMFMBO	EMBO	AAVS	AGSK	jSO	MPEDE	CJADE	RLBSO	TAPSO	XPSO	OLBSO	CMBO-PG
<i>f</i> 1	<i>Mean</i>	1.71E+09	8.38E+10	1.94E+09	9.55E+09	5.31E+00	0.00E+00	0.00E+00	0.00E+00	6.93E+02	6.59E+03	5.95E+04	1.26E+06	5.66E-05
	<i>Std</i>	6.93E+08	9.05E+09	1.73E+09	4.92E+09	1.26E+01	0.00E+00	0.00E+00	0.00E+00	2.19E+03	6.90E+03	3.66E+05	4.77E+05	4.48E-05
<i>f</i> 3	<i>Mean</i>	6.76E+05	6.84E+05	8.13E+05	5.71E+04	2.75E+05	4.90E-08	4.69E+01	1.42E+05	5.16E+05	3.40E+04	2.65E+04	3.60E+02	9.68E+00
	<i>Std</i>	5.30E+04	5.60E+04	1.28E+05	1.22E+04	3.02E+04	8.00E-08	1.64E+02	1.66E+05	1.20E+05	1.26E+04	6.80E+03	6.87E+01	6.91E+01
<i>f</i> 4	<i>Mean</i>	4.99E+02	1.17E+04	3.61E+02	3.68E+02	2.25E+02	1.98E+02	7.09E+01	2.49E+01	1.51E+02	1.79E+02	4.95E+02	2.40E+02	2.08E+02
	<i>Std</i>	6.49E+01	1.96E+03	8.30E+01	6.17E+01	3.91E+00	1.29E+01	6.64E+01	4.65E+01	4.76E+01	5.15E+01	5.63E+01	5.16E+01	7.47E+00
<i>f</i> 5	<i>Mean</i>	4.55E+02	1.12E+03	6.47E+02	2.05E+02	5.91E+02	3.23E+01	1.51E+02	1.53E+02	3.67E+02	3.62E+02	2.34E+02	9.57E+02	2.13E+00
	<i>Std</i>	3.48E+01	4.26E+01	3.33E+02	3.78E+01	1.36E+01	5.47E+00	2.06E+01	2.18E+01	8.81E+01	4.90E+01	4.28E+01	2.37E+02	3.67E+00
<i>f</i> 6	<i>Mean</i>	6.69E+00	5.12E+01	5.18E+00	1.23E+01	1.16E+00	2.05E-04	1.65E-01	7.67E-04	1.54E+01	7.63E-02	5.88E+00	1.57E+01	2.77E-05
	<i>Std</i>	8.58E-01	2.47E+00	2.34E+00	3.76E+00	1.02E+00	7.45E-04	1.23E-01	2.39E-03	5.97E+00	2.65E-02	2.14E+00	6.02E+00	1.33E-06
<i>f</i> 7	<i>Mean</i>	7.55E+02	2.97E+03	1.13E+03	2.80E+02	8.09E+02	1.61E+02	2.96E+02	2.77E+02	5.31E+02	5.49E+02	4.68E+02	9.42E+02	5.80E+02
	<i>Std</i>	5.38E+01	1.66E+02	3.81E+02	4.16E+01	1.65E+01	7.65E+01	3.50E+01	2.23E+01	1.20E+02	7.09E+01	8.03E+01	1.90E+02	1.50E+02
<i>f</i> 8	<i>Mean</i>	4.69E+02	1.14E+03	5.39E+02	2.20E+02	5.80E+02	3.17E+01	1.51E+02	1.45E+02	3.76E+02	3.79E+02	2.18E+02	9.20E+02	2.68E+00
	<i>Std</i>	4.16E+01	4.84E+01	3.12E+02	4.49E+01	1.54E+01	5.83E+00	2.80E+01	1.86E+01	7.84E+01	6.48E+01	4.19E+01	2.67E+02	3.67E+00
<i>f</i> 9	<i>Mean</i>	1.14E+04	5.51E+04	1.92E+04	1.34E+03	1.53E+01	4.59E-02	3.17E+01	1.03E+02	7.06E+03	6.26E+03	7.74E+02	2.81E+04	0.00E+00
	<i>Std</i>	5.05E+03	7.16E+03	5.19E+03	7.31E+02	1.16E+01	1.03E-01	1.74E+01	7.05E+01	3.84E+03	2.35E+03	3.10E+02	9.66E+03	0.00E+00
<i>f</i> 10	<i>Mean</i>	1.19E+04	2.24E+04	3.01E+04	9.35E+03	2.47E+04	2.29E+04	1.08E+04	1.02E+04	1.46E+04	1.20E+04	1.25E+04	1.61E+04	3.39E+02
	<i>Std</i>	8.01E+02	6.46E+02	1.72E+03	1.36E+03	5.57E+02	6.77E+02	1.10E+03	5.45E+02	1.69E+03	1.17E+03	1.17E+03	1.43E+03	2.42E+02
<i>f</i> 11	<i>Mean</i>	1.25E+05	1.38E+05	2.61E+05	9.20E+02	5.42E+02	1.05E+02	8.18E+02	2.99E+03	5.89E+02	3.06E+02	1.30E+03	1.45E+03	2.65E+01
	<i>Std</i>	2.50E+04	3.11E+04	1.30E+05	1.56E+02	6.49E+01	3.38E+01	2.40E+02	3.44E+03	2.57E+02	8.81E+01	2.44E+02	1.94E+02	3.78E+01
<i>f</i> 12	<i>Mean</i>	8.81E+08	2.37E+10	4.41E+08	6.12E+08	2.28E+05	1.98E+04	3.11E+04	1.85E+04	1.38E+06	4.26E+05	3.19E+07	2.70E+07	3.56E+03
	<i>Std</i>	2.86E+08	3.81E+09	5.62E+08	1.33E+09	1.11E+05	1.01E+04	1.56E+04	7.42E+03	9.48E+05	3.25E+05	2.72E+07	8.18E+06	4.20E+02
<i>f</i> 13	<i>Mean</i>	2.59E+08	3.84E+09	1.74E+07	3.34E+04	2.28E+03	1.51E+02	8.03E+02	4.27E+03	4.37E+03	4.28E+03	6.61E+03	3.07E+04	5.93E+01
	<i>Std</i>	1.17E+08	8.55E+08	4.57E+07	4.62E+04	1.21E+03	4.03E+01	9.63E+02	4.66E+03	6.92E+03	5.14E+03	6.28E+03	1.07E+04	4.68E+01
<i>f</i> 14	<i>Mean</i>	2.03E+07	4.11E+07	1.55E+06	2.00E+02	4.75E+02	6.10E+01	4.85E+02	5.53E+02	7.25E+04	3.68E+04	4.66E+05	4.49E+05	3.47E+01
	<i>Std</i>	1.14E+07	1.67E+07	8.48E+05	3.16E+01	8.34E+01	1.17E+01	1.29E+02	2.49E+02	4.49E+04	3.60E+04	7.76E+05	2.18E+05	5.15E+00
<i>f</i> 15	<i>Mean</i>	1.31E+08	1.43E+09	4.23E+05	8.09E+02	3.72E+02	1.70E+02	3.89E+02	3.59E+02	3.53E+03	1.68E+03	4.04E+03	1.12E+04	4.81E+01
	<i>Std</i>	7.62E+07	4.43E+08	2.97E+06	1.87E+02	1.23E+02	4.45E+01	2.80E+02	8.53E+01	6.39E+03	2.30E+03	6.34E+03	4.27E+03	2.56E+01
<i>f</i> 16	<i>Mean</i>	4.74E+03	8.11E+03	7.34E+03	2.03E+03	4.27E+03	2.78E+03	2.84E+03	2.58E+03	3.93E+03	3.28E+03	2.94E+03	4.98E+03	6.53E+01
	<i>Std</i>	5.62E+02	5.43E+02	1.96E+03	4.55E+02	1.83E+02	5.60E+02	5.72E+02	3.38E+02	6.38E+02	6.56E+02	6.78E+02	7.67E+02	5.82E+01
<i>f</i> 17	<i>Mean</i>	3.80E+03	1.11E+04	4.83E+03	1.30E+03	2.58E+03	1.84E+03	1.99E+03	1.90E+03	2.97E+03	2.55E+03	2.52E+03	4.02E+03	9.40E+01
	<i>Std</i>	4.19E+02	6.66E+03	1.18E+03	3.77E+02	1.16E+02	2.48E+02	4.11E+02	2.55E+02	5.82E+02	5.67E+02	5.10E+02	6.34E+02	7.85E+01
<i>f</i> 18	<i>Mean</i>	1.60E+07	3.28E+07	2.26E+06	3.19E+02	1.33E+05	1.89E+02	9.14E+02	1.41E+03	4.48E+05	1.05E+05	3.46E+05	7.19E+05	4.01E+01
	<i>Std</i>	6.67E+06	1.21E+07	1.14E+06	5.19E+01	4.30E+04	3.98E+01	8.84E+02	7.34E+02	2.57E+05	5.83E+04	2.01E+05	2.81E+05	8.97E+00
<i>f</i> 19	<i>Mean</i>	1.01E+08	1.46E+09	8.32E+04	2.27E+02	2.06E+02	1.15E+02	2.46E+02	1.48E+03	3.83E+03	2.32E+03	4.16E+03	1.18E+05	4.21E+01
	<i>Std</i>	4.33E+07	4.17E+08	4.18E+05	4.95E+01	3.98E+01	2.10E+01	6.30E+01	2.18E+03	5.61E+03	2.84E+03	4.82E+03	1.21E+05	1.12E+01
<i>f</i> 20	<i>Mean</i>	3.51E+03	4.55E+03	5.11E+03	9.00E+02	2.75E+03	2.15E+03	1.95E+03	1.88E+03	2.73E+03	2.25E+03	2.26E+03	4.03E+03	1.68E+02
	<i>Std</i>	4.10E+02	3.91E+02	6.74E+02	2.25E+02	1.00E+02	2.06E+02	4.58E+02	2.83E+02	6.11E+02	5.45E+02	5.30E+02	6.09E+02	3.10E+01
<i>f</i> 21	<i>Mean</i>	6.92E+02	1.38E+03	7.95E+02	4.00E+02	8.00E+02	3.88E+02	3.67E+02	3.63E+02	6.64E+02	6.11E+02	4.66E+02	1.21E+03	2.25E+02
	<i>Std</i>	4.27E+01	5.04E+01	3.28E+02	4.68E+01	1.38E+01	1.98E+01	2.52E+01	1.64E+01	8.96E+01	5.70E+01	4.60E+01	1.78E+02	4.54E+00
<i>f</i> 22	<i>Mean</i>	1.30E+04	2.36E+04	3.11E+04	1.16E+03	2.56E+04	1.10E+04	1.21E+04	1.11E+04	1.60E+04	1.29E+04	1.20E+04	1.77E+04	2.63E+02
	<i>Std</i>	7.73E+02	7.10E+02	1.63E+03	3.50E+03	6.44E+02	5.05E+02	9.43E+02	2.27E+03	1.47E+03	1.27E+03	5.37E+03	1.34E+03	4.10E+02
<i>f</i> 23	<i>Mean</i>	8.75E+02	1.69E+03	1.14E+03	6.90E+02	9.65E+02	7.56E+02	6.96E+02	6.51E+02	9.83E+02	8.19E+02	8.29E+02	1.53E+03	5.57E+02
	<i>Std</i>	3.12E+01	5.23E+01	3.12E+02	4.15E+01	1.36E+01	4.70E+01	2.73E+01	1.46E+01	8.64E+01	4.42E+01	5.95E+01	2.18E+02	1.04E+01
<i>f</i> 24	<i>Mean</i>	1.39E+03	2.40E+03	1.59E+03	1.03E+03	1.29E+03	1.11E+03	1.02E+03	1.03E+03	1.36E+03	1.34E+03	1.26E+03	1.98E+03	8.90E+02
	<i>Std</i>	4.36E+01	7.46E+01	2.96E+02	1.24E+02	1.35E+01	5.80E+01	2.92E+01	2.42E+01	1.22E+02	6.06E+01	9.70E+01	2.25E+02	9.90E+00
<i>f</i> 25	<i>Mean</i>	1.07E+03	9.96E+03	9.58E+02	9.16E+02	7.13E+02	7.37E+02	7.47E+02	7.38E+02	7.50E+02	7.64E+02	1.08E+03	7.20E+02	6.37E+02
	<i>Std</i>	8.91E+01	1.23E+03	1.23E+02	7.42E+01	4.48E+01	5.28E+01	4.44E+01	5.52E+01	7.66E+01	7.26E+01	7.91E+01	5.61E+01	4.81E-02
<i>f</i> 26	<i>Mean</i>	8.37E+03	1.84E+04	1.08E+04	4.04E+02	7.84E+03	5.42E+02	4.58E+03	4.62E+03	8.69E+03	8.07E+03	4.63E+03	1.38E+04	2.76E+03
	<i>Std</i>	3.50E+02	7.51E+02	2.86E+03	5.46E+02	1.84E+02	4.11E+02	3.09E+02	2.11E+02	8.85E+02	9.32E+02	2.29E+03	4.25E+03	2.08E+02
<i>f</i> 27	<i>Mean</i>	7.67E+02	1.76E+03	6.27E+02	6.75E+02	6.63E+02	7.13E+02	7.01E+02	7.38E+02	5.00E+02	8.31E+02	8.71E+02	5.00E+02	6.44E+02
	<i>Std</i>	3.93E+01	9.75E+01	1.17E+02	3.72E+01	2.27E+01	4.17E+01	3.54E+01	3.51E+01	2.90E-04	8.08E+01	6.34E+01	2.19E-04	3.67E+01
<i>f</i> 28	<i>Mean</i>	8.19E+02	1.22E+04	7.55E+02	7.17E+02	5.36E+02	5.30E+02	5.32E+02	4.83E+02	4.83E+02	5.52E+02	8.28E+02	5.00E+02	5.75E+02
	<i>Std</i>	6.29E+01	1.88E+03	1.70E+02	5.85E+01</									

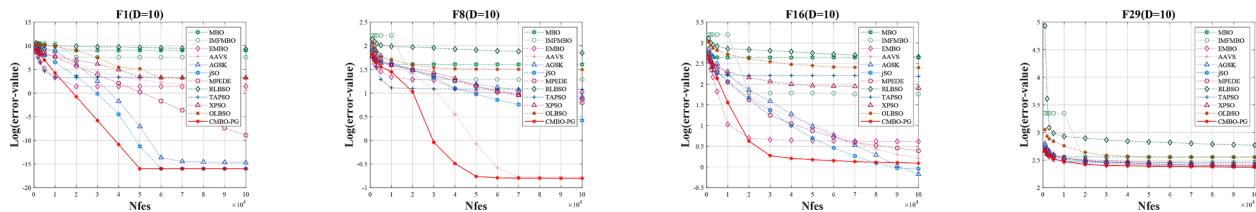


Fig. 7. The convergence graphs of CMBO-PG and comparison algorithm (10D).

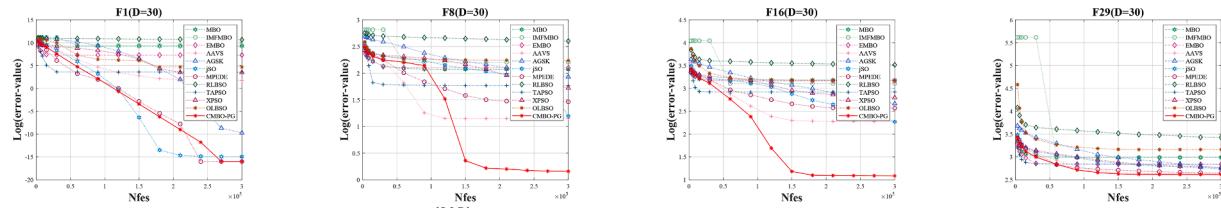


Fig. 8. The convergence graphs of CMBO-PG and comparison algorithm (30D).

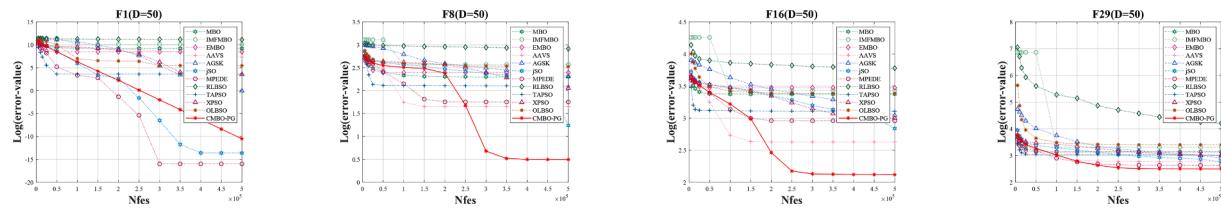


Fig. 9. The convergence graphs of CMBO-PG and comparison algorithm (50D).

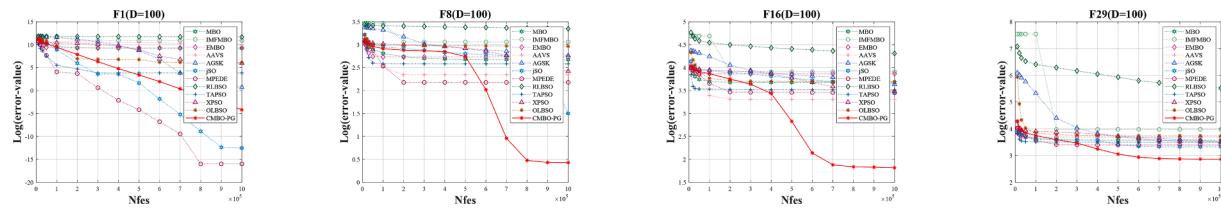


Fig. 10. The convergence graphs of CMBO-PG and comparison algorithm (100D).

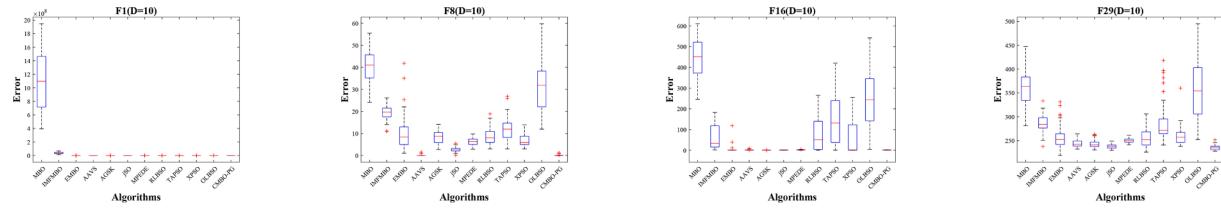


Fig. 11. The box plots of CMBO-PG and comparison algorithm (10D).

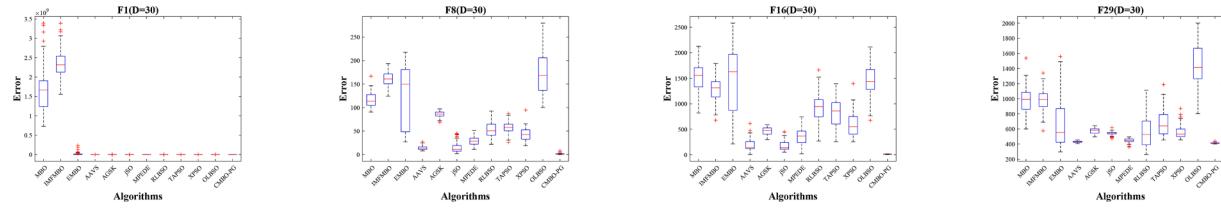


Fig. 12. The box plots of CMBO-PG and comparison algorithm (30D).

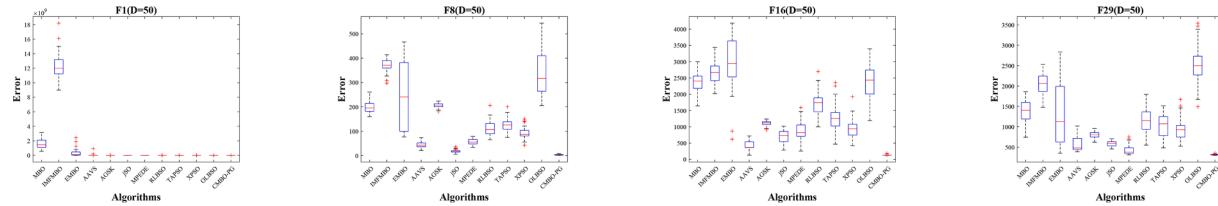


Fig. 13. The box plots of CMBO-PG and comparison algorithm (50D).

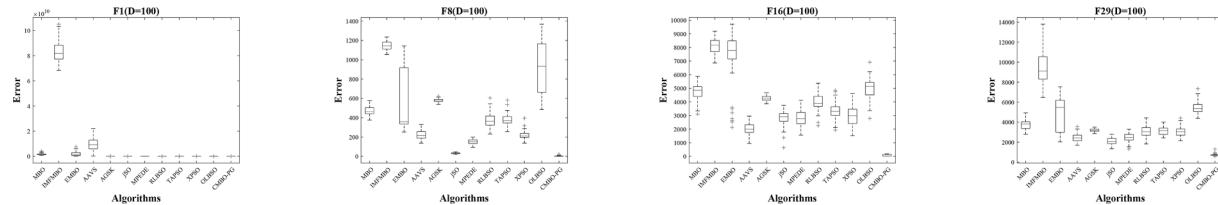


Fig. 14. The box plots of CMBO-PG and comparison algorithm (100D).

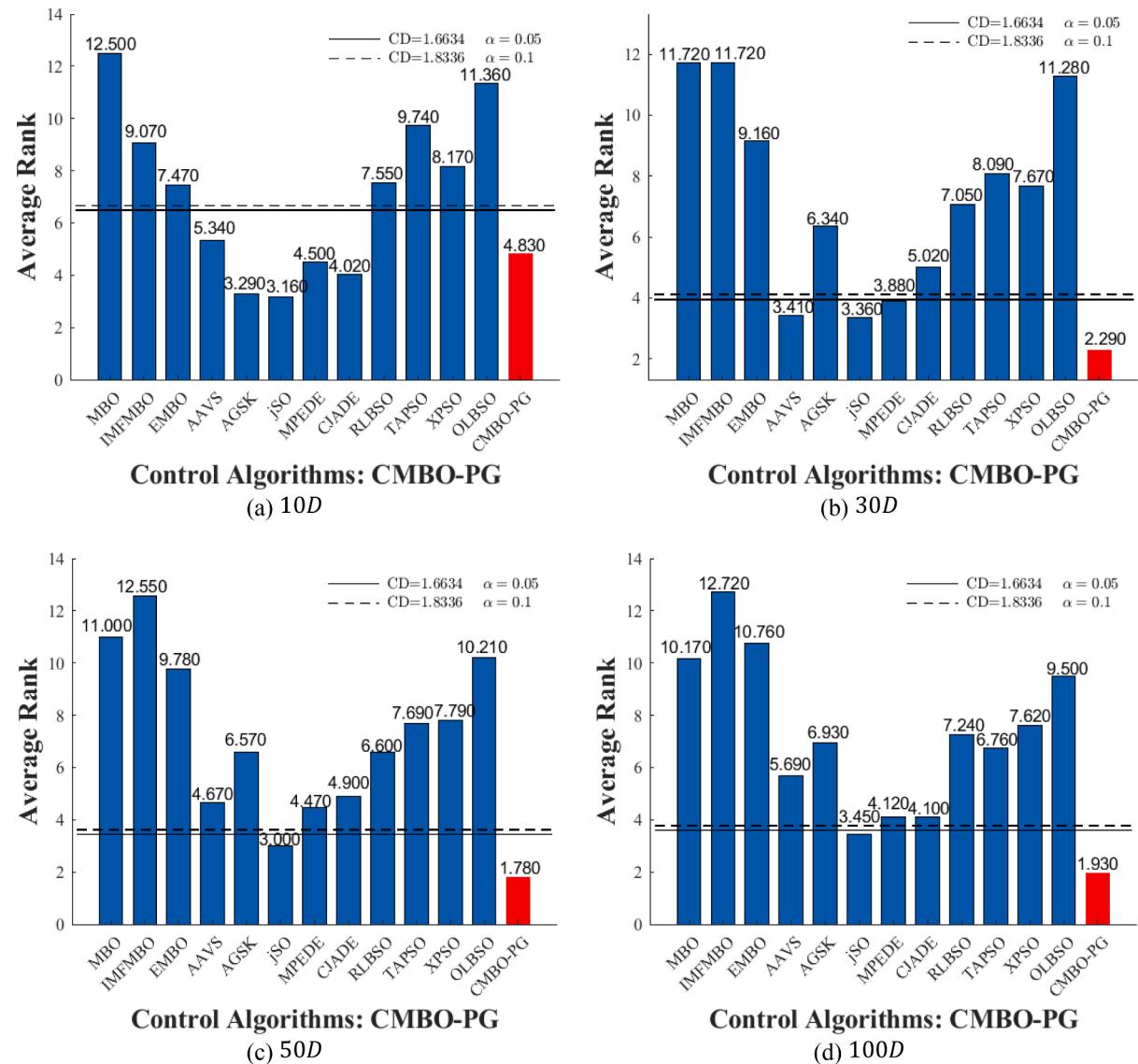


Fig. 15. Graphical representation of the Nemenyi test.

Nemenyi test which is based on the Friedman test. The solid line and dotted line are the CD between the mean rank of each algorithm and the CMBO-PG. If the difference is greater than the CD, a significant difference is present between the two algorithms and vice versa. The results of the Nemenyi test are recorded in Fig. 15(a)–(d). The mean rank of CMBO-PG is the smallest among all comparison algorithms in higher dimensions, especially in 50D and 100D. CMBO-PG is significantly better than almost comparison algorithms (except for jSO). The outstanding performance indicates that the proposed algorithm is effective and suitable for real-valued optimization problems in higher dimensions.

The Wilcoxon test (Demiar & Schuurmans, 2006) is utilized to further validate the significant difference by comparing the two algorithms. The results of the Wilcoxon test in different dimensions are recorded in Table 8. Where R^+ represents the sum of ranks, which means CMBO-PG is better than the comparison algorithm, and R^- represents the sum of ranks for the opposite. + is the number of functions that perform better than the comparison algorithm, – is the worse number of functions, and \approx is the similar number of functions. A significant difference is present between the control algorithm and the comparison algorithm when Z is less than -1.96 . Another criterion is the p –value. A

significant difference is present between CMBO-PG and the comparison algorithm when the p –value is less than 0.05, which is represented by yes. Otherwise, the difference is recorded by no. The results of Wilcoxon test indicate CMBO-PG is substantially superior to other algorithms in a pairwise comparison from Table 8, especially in 30D, 50D, and 100D. The result of the rank-sum test shows that the proposed algorithm is effective in solving continuous real-valued optimization problems.

4.5. Effectiveness analysis of strategy

This paper compares the performance of the CMBO-PG algorithm with that of eliminating the leading flock (WITHOUT LEADER) and that of eliminating the following flock (WITHOUT FOLLOWER), to verify the effectiveness of the co-evolution mechanism in CMBO-PG. In addition, the effectiveness of the multi-strategy learning mechanism by comparing the proposed algorithm with the CMBO-PG using the single strategy (WITH STRATEGY1, WITH STRATEGY2, and WITH STRATEGY3). Finally, the performance of CMBO-PG was compared with that of CMBO-PG using the random selection mechanism (RANDOM SELECTION) to further study the effectiveness of the multi-strategy learning mechanism. All compared algorithms were run 51 times in 10D, 30D,

Table 8
Results of CMBO-PG and comparison algorithm through Wilcoxon test.

Dimension	CMBO-PG vs	+	–	\approx	R^+	R^-	Z	p –value	$\alpha = 0.05$	$\alpha = 0.1$
10D	MBO	26	3	0	418.00	17.00	-4.314	1.60E-05	yes	yes
	IMFMBO	23	6	0	347.00	88.00	-2.800	5.11E-03	yes	yes
	EMBO	21	8	0	309.00	126.00	-1.979	4.79E-02	yes	yes
	AAVS	17	9	3	185.00	250.00	-0.703	4.82E-01	no	no
	AGSK	10	18	1	112.00	294.00	-2.072	3.83E-02	yes	yes
	jSO	5	18	6	59.50	216.50	-2.388	1.92E-02	yes	yes
	MPEDE	14	12	3	142.50	208.50	-0.838	4.01E-01	no	no
	CJADE	12	13	4	156.50	168.50	-0.161	8.72E-01	no	no
	RLBSO	23	5	1	303.00	103.00	-0.277	2.22E-02	yes	yes
	TAPSO	26	3	0	405.50	29.50	-4.065	4.79E-05	yes	yes
30D	XPSO	25	4	0	374.00	61.00	-3.384	7.14E-04	yes	yes
	OLBSO	26	3	0	405.00	30.00	-4.054	5.02E-05	yes	yes
	MBO	29	0	0	435.00	0.00	-4.703	2.56E-06	yes	yes
	IMFMBO	28	1	0	434.00	1.00	-4.682	2.85E-06	yes	yes
	EMBO	27	1	1	402.00	4.00	-4.532	5.86E-06	yes	yes
	AAVS	22	6	1	291.00	115.00	-2.004	4.51E-02	yes	yes
	AGSK	24	3	2	364.00	14.00	-4.204	2.62E-05	yes	yes
	jSO	19	7	3	271.00	80.00	-2.426	1.52E-02	yes	yes
	MPEDE	21	4	4	301.00	24.00	-3.727	1.93E-04	yes	yes
	CJADE	24	3	2	364.00	14.00	-4.204	2.61E-05	yes	yes
50D	RLBSO	25	4	0	395.00	40.00	-3.838	1.23E-04	yes	yes
	TAPSO	28	1	0	429.00	6.00	-4.573	4.80E-06	yes	yes
	XPSO	29	0	0	435.00	0.00	-4.703	2.56E-06	yes	yes
	OLBSO	28	1	0	434.00	1.00	-4.681	2.84E-06	yes	yes
	MBO	29	0	0	435.00	0.00	-4.703	2.56E-06	yes	yes
	IMFMBO	29	0	0	435.00	0.00	-4.703	2.56E-06	yes	yes
	EMBO	27	2	0	405.00	30.00	-4.054	5.03E-05	yes	yes
	AAVS	24	4	1	351.00	55.00	-3.370	7.51E-04	yes	yes
	AGSK	29	0	0	435.00	0.00	-4.703	2.56E-06	yes	yes
	jSO	22	7	0	369.00	66.00	-3.276	1.05E-03	yes	yes
100D	MPEDE	27	2	0	428.00	7.00	-4.552	5.31E-06	yes	yes
	CJADE	26	3	0	426.00	9.00	-4.508	6.53E-06	yes	yes
	RLBSO	26	3	0	429.00	16.00	-4.357	2.56E-05	yes	yes
	TAPSO	29	0	0	435.00	0.00	-4.703	2.56E-06	yes	yes
	XPSO	29	0	0	435.00	0.00	-4.703	2.56E-06	yes	yes
	OLBSO	28	1	0	430.00	5.00	-4.595	4.32E-06	yes	yes
	MBO	29	0	0	435.00	0.00	-4.703	2.56E-06	yes	yes
	IMFMBO	29	0	0	435.00	0.00	-4.703	2.56E-06	yes	yes
	EMBO	28	1	0	433.00	2.00	-4.660	3.16E-06	yes	yes
	AAVS	27	2	0	398.00	37.00	-3.903	9.50E-05	yes	yes
100D	AGSK	28	1	0	429.00	6.00	-4.573	4.80E-06	yes	yes
	jSO	25	4	0	396.00	39.00	-3.860	1.13E-05	yes	yes
	MPEDE	25	4	0	405.00	30.00	-4.054	5.02E-05	yes	yes
	CJADE	25	4	0	406.00	29.00	-4.076	4.57E-05	yes	yes
	RLBSO	25	4	0	420.00	15.00	-4.379	1.19E-05	yes	yes
	TAPSO	26	3	0	426.00	9.00	-4.508	6.53E-06	yes	yes
	XPSO	28	1	0	433.00	2.00	-4.660	3.16E-06	yes	yes
	OLBSO	27	2	0	427.00	8.00	-4.530	5.89E-06	yes	yes

Table 9

Results of strategy validity analysis for CMBO-PG (10D).

f	WITHOUT LEADER		WITHOUT FOLLOWER		WITH STRATEGY1		WITH STRATEGY2		WITH STRATEGY3		RANDOM SELECTION		CMBO-PG	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
f1	5.80E+04	2.53E+04	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
f3	1.11E+01	5.66E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
f4	2.33E+00	6.62E-01	5.65E-02	1.09E-01	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	8.80E-02	1.38E-01
f5	1.30E+01	2.18E+00	1.21E+00	1.17E+00	9.36E-01	1.00E+00	7.41E-01	7.67E-01	1.21E+00	1.04E+00	8.19E-01	8.83E-01	4.05E-01	7.19E-01
f6	2.32E-01	6.19E-02	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
f7	2.68E+01	2.89E+00	1.15E+01	6.88E-01	1.10E+01	4.62E-01	1.12E+01	2.11E+00	1.10E+01	4.67E-01	1.10E+01	4.66E-01	1.07E+01	3.52E-01
f8	1.37E+01	2.37E+00	9.56E-01	7.70E-01	7.80E-01	8.75E-01	7.02E-01	7.26E-01	6.63E-01	7.36E-01	9.17E-01	8.17E-01	1.58E-01	3.62E-01
f9	9.07E-02	2.80E-02	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
f10	5.85E+02	1.29E+02	8.92E+01	1.35E+02	1.93E+01	3.89E+01	2.78E+01	5.14E+01	2.93E+01	7.39E+01	3.20E+01	5.61E+01	1.18E+01	2.90E+01
f11	4.81E+00	6.89E-01	6.65E-01	8.68E-01	3.51E-01	5.57E-01	2.29E-01	6.02E-01	2.54E-01	4.81E-01	4.29E-01	6.68E-01	9.62E-02	3.01E-01
f12	7.51E+03	4.44E+03	1.17E+02	5.24E+01	1.17E+02	8.00E+01	1.06E+02	7.23E+01	1.00E+02	7.99E+01	1.27E+02	7.96E+01	6.77E+01	5.87E+01
f13	3.00E+01	9.45E+00	5.16E+00	2.84E+00	4.49E+00	3.01E+00	5.10E+00	2.43E+00	4.67E+00	2.68E+00	4.65E+00	2.95E+00	3.78E+00	3.07E+00
f14	1.18E+01	2.67E+00	2.60E+00	5.69E+00	4.91E+00	8.40E+00	5.02E+00	8.51E+00	4.85E+00	8.21E+00	4.20E+00	8.11E+00	3.44E+00	7.29E+00
f15	3.58E+00	9.37E-01	1.07E+00	1.10E+00	1.01E+00	8.13E-01	9.90E-01	8.44E-01	8.19E-01	6.17E-01	8.51E-01	6.07E-01	6.06E-01	6.34E-01
f16	3.01E+00	1.21E+00	1.43E+00	4.79E-01	1.42E+00	4.99E-01	3.71E+00	1.66E+01	1.27E+00	5.28E-01	3.64E+00	1.68E+01	1.23E+00	6.31E-01
f17	1.97E+01	5.14E+00	2.05E+01	7.54E+00	1.83E+01	9.51E+00	1.98E+01	8.14E+00	2.10E+01	8.52E+00	1.84E+01	8.94E+00	2.10E+01	8.10E+00
f18	2.73E+01	5.05E+00	1.11E+01	9.84E+00	9.43E+00	9.84E+00	6.78E+00	9.20E+00	8.96E+00	9.91E+00	7.07E+00	9.29E+00	5.62E+00	8.79E+00
f19	2.91E+00	6.75E-01	1.00E+00	7.92E-01	4.36E-01	4.84E-01	2.83E-01	4.32E-01	5.23E-01	6.31E-01	4.33E-01	5.67E-01	3.07E-01	3.59E-01
f20	9.61E+00	3.81E+00	1.42E+01	9.78E+00	9.85E+00	1.01E+01	8.29E+00	9.53E+00	1.07E+01	9.74E+00	9.47E+00	9.86E+00	4.90E+00	7.75E+00
f21	9.85E+01	1.31E+01	2.02E+02	5.39E+00	2.02E+02	1.01E+01	1.98E+02	1.67E+01	2.00E+02	1.20E+01	2.02E+02	5.47E+00	1.99E+02	1.13E+01
f22	7.82E+00	1.97E+01	1.00E+02	0.00E+00	1.00E+02	0.00E+00	1.00E+02	0.00E+00	1.00E+02	0.00E+00	1.00E+02	0.00E+00	1.00E+02	0.00E+00
f23	3.12E+02	1.19E+01	3.05E+02	2.79E+00	3.05E+02	3.61E+00	3.04E+02	2.76E+00	3.04E+02	3.22E+00	3.05E+02	3.02E+00	3.02E+02	1.94E+00
f24	1.02E+02	8.15E-01	3.30E+02	1.78E+00	3.29E+02	1.19E+01	3.30E+02	2.11E+00	3.28E+02	1.52E+01	3.29E+02	8.07E+00	3.27E+02	8.55E+00
f25	3.98E+02	2.73E-01	4.28E+02	2.21E+01	4.30E+02	2.10E+01	4.30E+02	2.08E+01	4.35E+02	1.75E+01	4.30E+02	2.09E+01	4.13E+02	2.10E+01
f26	2.90E+02	5.20E+01	3.15E+02	2.67E+01	3.10E+02	2.35E+01	3.10E+02	2.38E+01	3.08E+02	2.26E+01	3.06E+02	1.84E+01	3.00E+02	0.00E+00
f27	3.90E+02	5.90E-01	3.96E+02	2.73E+00	3.96E+02	2.16E+00	3.96E+02	2.89E+00	3.96E+02	2.80E+00	3.96E+02	2.81E+00	3.93E+02	2.58E+00
f28	2.98E+02	2.67E+01	6.02E+02	4.40E+01	6.05E+02	1.20E+01	6.06E+02	1.10E+01	6.06E+02	1.08E+01	6.06E+02	1.32E+01	5.97E+02	1.35E+01
f29	2.61E+02	7.11E+00	2.42E+02	9.83E+00	2.36E+02	5.48E+00	2.36E+02	6.14E+00	2.38E+02	9.14E+00	2.36E+02	6.37E+00	2.35E+02	5.19E+00
f30	1.99E+03	8.76E+02	3.11E+05	4.31E+05	2.89E+05	3.94E+05	2.91E+05	3.93E+05	2.31E+05	3.96E+05	3.77E+05	4.24E+05	6.09E+04	2.06E+05

Table 10

Results of strategy validity analysis for CMBO-PG (30D).

f	WITHOUT LEADER		WITHOUT FOLLOWER		WITH STRATEGY1		WITH STRATEGY2		WITH STRATEGY3		RANDOM SELECTION		CMBO-PG	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
f1	5.26E+07	1.16E+07	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
f3	2.24E+04	4.11E+03	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
f4	1.02E+02	1.25E+01	8.87E+01	9.09E-01	8.39E+01	5.82E+00	7.98E+01	9.78E+00	8.31E+01	7.46E+00	8.29E+01	7.28E+00	5.97E+01	3.94E+00
f5	1.48E+02	1.01E+01	5.95E+00	2.87E+00	3.53E+00	1.94E+00	2.50E+00	1.80E+00	3.41E+00	2.04E+00	3.08E+00	1.80E+00	1.54E+00	1.11E+00
f6	4.76E+00	7.69E-01	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
f7	2.02E+02	8.05E+00	3.38E+01	9.07E-01	3.28E+01	6.86E-01	5.27E+01	4.04E+01	3.27E+01	5.84E-01	3.27E+01	5.26E-01	3.25E+01	4.54E-01
f8	1.50E+02	8.45E+00	5.33E+00	2.61E+00	3.20E+00	2.00E+00	2.63E+00	1.63E+00	3.24E+00	1.99E+00	2.65E+00	1.65E+00	1.44E+00	1.39E+00
f9	4.82E+01	1.25E+01	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
f10	5.77E+03	2.41E+02	5.04E+02	2.62E+02	3.11E+02	1.91E+02	1.93E+02	1.64E+02	2.97E+02	1.89E+02	2.51E+02	1.65E+02	1.28E+02	1.30E+02
f11	9.84E+01	1.73E+01	2.39E+01	2.54E+01	3.58E+01	2.86E+01	2.68E+01	2.77E+01	2.74E+01	2.80E+01	2.69E+01	2.79E+01	1.10E+01	1.91E+01
f12	4.54E+06	1.36E+06	5.52E+02	2.84E+02	6.18E+02	2.58E+02	6.00E+02	3.08E+02	5.49E+02	2.57E+02	5.64E+02	2.76E+02	2.42E+02	1.44E+02
f13	2.07E+05	8.88E+04	3.04E+01	1.40E+01	2.39E+01	1.70E+01	2.23E+01	7.68E+00	2.13E+01	6.94E+00	1.84E+01	7.35E+00	1.18E+01	9.69E+00
f14	2.49E+02	4.89E+01	2.45E+01	2.97E+00	2.11E+01	6.53E+00	2.05E+01	6.07E+00	2.23E+01	3.84E+00	1.98E+01	6.33E+00	2.05E+01	4.48E+00
f15	9.70E+03	5.01E+03	1.02E+01	4.68E+00	7.06E+00	3.44E+00	6.55E+00	2.85E+00	6.84E+00	3.81E+00	7.45E+00	4.46E+00	4.64E+00	2.68E+00
f16	9.15E+02	1.57E+02	1.94E+01	2.98E+01	2.49E+01	3.59E+01	1.98E+01	3.02E+01	1.94E+01	3.82E+01	2.50E+01	4.62E+01	1.22E+01	4.18E+00
f17	2.12E+02	4.71E+01	3.21E+01	4.71E+00	2.80E+01	3.48E+00	2.82E+01	1.94E+00	2.70E+01	4.68E+00	2.67E+01	6.09E+00	2.49E+01	5.00E+00
f18	3.88E+04	1.25E+04	2.49E+01	3.44E+00	2.43E+01	2.28E+00	2.53E+01	3.97E+00	2.51E+01	4.73E+00	2.42E+01	2.88E+00	2.30E+01	1.47E+00
f19	8.11E+03	4.47E+03	7.28E+00	1.92E+00	6.60E+00	1.71E+00	5.86E+00	1.78E+00	6.10E+00	1.65E+00	6.32E+00	1.69E+00	5.69E+00	1.43E+00
f20	2.44E+02	8.18E+01	3.50E+01	2.93E+01	4.99E+00	2.59E+01	1.73E+01	3.18E+01	2.94E+01	3.33E+01	3.39E+01	2.72E+01	2.14E+01	
f21	3.43E+02	9.00E+00	2.08E+02	3.44E+00	2.05E+02	2.04E+00	2.04E+02	1.79E+00	2.05E+02	2.27E+00	2.05E+02	2.43E+00	2.03E+02	8.60E-01
f22	1.36E+02	3.96E+00	1.00E+02	0.00E+00	1.00E+02	0.00E+00	1.00E+02	0.00E+00	1.00E+02	0.00E+00	1.00E+02	0.00E+00	1.00E+02	0.00E+00
f23	4.96E+02	9.91E+00	3.63E+02	7.12E+00	3.64E+02	6.69E+00	3.63E+02	5.60E+00	3.62E+02	8.22E+00	3.62E+02	6.82E+00	3.55E+02	6.45E+00
f24	5.69E+02	8.86E+00	4.31E+02	4.74E+00	4.29E+02	3.38E+00	4.28E+02	3.18E+00	4.30E+02	5.11E+00	4.28E+02	3.81E+00	4.26E+02	2.96E+00
f25	3.93E+02	4.14E+00	3.87E+02	2.95E-02	3.87E+02	1.49E-02	3.87E+02	1.14E-02	3.87E+02	1.01E-02	3.87E+02	1.13E-02	3.87E+02	4.10E-03
f26	4.44E+02	9.73E+01	9.99E+02	9.71E+01	9.75E+02	9.80E+01	9.62E+02	8.53E+01	9.71E+02	9.67E+01	9.80E+02	8.77E+01	8.68E+02	5.44E+01
f27	5.21E+02	7.54E+00	5.24E+02	1.24E+01	5.22E+02	1.14E+01	5.23E+02	1.17E+01	5.23E+02	1.18E+01	5.22E+02	1.28E+01	5.08E+02	4.55E+00
f28	4.32E+02	8.46E+00	4.13E+02	3.21E+01	4.10E+02	3.92E+01	4.04E+02	4.38E+01	4.14E+02	3.64E+01	3.95E+02	4.80E+01	3.17E+02	3.49E+01
f29	8.26E+02	6.32E+01	4.30E+02	1.42E+01	4.21E+02	1.05E+01	4.23E+02	2.29E+01	4.23E+02	1.18E+01	4.18E+02	1.41E+01	4.13E+02	7.66E+00
f30	8.43E+04	2.65E+04	2.17E+03	2.08E+02	2.08E+03	1.09E+02	2.10E+03	1.33E+02	2.09E+03	1.34E+02	2.08E+03	1.14E+02	1.97E+03	2.72E+01

Table 11

Results of strategy validity analysis for CMBO-PG (50D).

f	WITHOUT LEADER		WITHOUT FOLLOWER		WITH STRATEGY1		WITH STRATEGY2		WITH STRATEGY3		RANDOM SELECTION		CMBO-PG		
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	
f1	3.94E+08	1.13E+08	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	
f3	1.09E+05	1.82E+04	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	
f4	1.97E+02	4.97E+01	1.48E+02	2.77E+01	1.40E+02	4.06E+01	1.26E+02	4.68E+01	1.26E+02	4.60E+01	1.26E+02	4.74E+01	4.19E+01	3.63E+01	
f5	3.25E+02	1.52E+01	9.46E+00	4.04E+00	6.07E+00	3.19E+00	3.90E+00	2.05E+00	5.50E+00	2.58E+00	5.09E+00	2.02E+00	2.58E+00	1.51E+00	
f6	9.22E+00	1.46E+00	3.54E-08	2.05E-07	2.23E-07	2.24E-08	3.43E-07	1.98E-08	2.58E-07	2.51E-08	2.72E-07	2.44E-08	2.39E-07	3.13E-08	
f7	4.17E+02	1.45E+01	5.69E+01	1.28E+00	6.95E+01	4.67E+01	3.03E+02	5.42E+01	8.43E+01	6.34E+01	1.41E+02	9.44E+01	1.14E+02	8.50E+01	
f8	3.27E+02	1.29E+01	9.93E+00	3.81E+00	6.20E+00	2.99E+00	4.39E+00	2.26E+00	5.48E+00	2.43E+00	5.54E+00	2.50E+00	3.14E+00	1.68E+00	
f9	3.56E+02	1.52E+02	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	
f10	1.15E+04	3.12E+02	8.63E+02	3.37E+02	3.67E+02	2.30E+02	2.86E+02	1.76E+02	3.78E+02	2.14E+02	3.56E+02	2.06E+02	1.75E+02	9.19E+01	
f11	3.18E+02	4.81E+01	3.59E+01	8.46E+00	3.11E+01	5.70E+00	3.05E+01	5.44E+00	3.22E+01	7.80E+00	3.08E+01	5.17E+00	2.28E+01	2.71E+00	
f12	7.19E+07	1.61E+07	1.52E+03	4.15E+02	1.38E+03	3.94E+02	1.44E+03	3.03E+02	1.42E+03	4.23E+02	1.44E+03	4.72E+02	1.07E+03	2.76E+02	
f13	4.68E-06	1.71E+06	1.13E+02	4.03E+01	9.90E+01	5.53E+01	8.92E+01	4.17E+01	8.49E+01	4.00E+01	8.29E+01	3.75E+01	4.53E+01	3.21E+01	
f14	1.08E+04	4.61E+03	2.77E+01	4.12E+00	2.69E+01	2.95E+00	2.71E+01	4.08E+00	2.67E+01	2.89E+00	2.78E+01	3.61E+00	2.60E+01	3.46E+00	
f15	2.81E+05	1.20E+05	4.75E+01	1.42E+01	4.15E+01	1.38E+01	3.92E+01	1.19E+01	4.30E+01	1.36E+01	4.20E+01	1.28E+01	2.62E+01	7.33E+00	
f16	2.35E+03	2.09E+02	1.40E+02	3.28E+01	1.40E+02	3.34E+01	1.41E+02	3.68E+01	1.42E+02	3.63E+01	1.32E+02	1.75E+01	1.31E+02	1.24E+01	
f17	1.44E+03	1.60E+02	1.53E+02	6.19E+01	1.28E+02	6.53E+01	1.33E+02	6.91E+01	1.36E+02	5.90E+01	1.16E+02	7.22E+01	9.76E+01	7.11E+01	
f18	3.51E+05	1.15E+05	2.74E+01	3.95E+00	2.76E+01	4.20E+00	2.98E+01	5.78E+00	2.89E+01	6.70E+00	2.75E+01	4.56E+00	2.53E+01	1.98E+00	
f19	1.65E+05	5.83E+04	1.95E+01	4.99E+00	1.77E+01	3.85E+00	1.67E+01	4.49E+00	1.80E+01	5.29E+00	1.74E+01	4.00E+00	1.57E+01	2.85E+00	
f20	1.20E+03	1.87E+02	4.54E+01	1.26E+01	3.49E+01	1.63E+01	2.63E+01	3.59E+00	3.52E+01	2.33E+01	2.88E+01	4.85E+00	2.58E+01	3.59E+00	
f21	5.32E+02	1.29E+01	2.14E+02	3.75E+00	2.11E+02	2.77E+00	2.08E+02	2.51E+00	2.10E+02	2.54E+00	2.10E+02	3.23E+00	2.07E+02	2.03E+00	
f22	2.50E+02	6.09E+01	2.51E+02	3.77E+02	2.04E+02	2.97E+02	1.33E+02	1.10E+02	2.01E+02	3.08E+02	1.45E+02	1.90E+02	1.00E+02	1.20E-08	
f23	7.59E+02	1.71E+01	4.49E+02	1.17E+01	4.40E+02	9.18E+00	4.41E+02	7.02E+00	4.43E+02	7.38E+00	4.43E+02	8.21E+00	4.28E+02	5.73E+00	
f24	8.25E+02	1.51E+01	5.22E+02	6.93E+00	5.18E+02	7.09E+00	5.16E+02	8.09E+00	5.16E+02	7.76E+00	5.15E+02	6.87E+00	5.07E+02	5.00E+00	
f25	5.92E+02	2.99E+01	4.86E+02	1.61E+01	4.80E+02	1.94E+00	4.82E+02	1.16E+01	4.81E+02	2.80E+00	4.81E+02	3.33E+00	4.80E+02	5.05E-03	
f26	4.33E+03	1.38E+02	1.17E+03	9.89E+01	1.11E+03	1.12E+02	1.10E+03	1.00E+02	1.09E+03	9.69E+01	1.08E+03	1.26E+02	8.63E+02	1.05E+02	
f27	6.06E+02	6.91E+01	6.35E+02	4.83E+01	6.53E+02	4.80E+01	6.64E+02	6.74E+01	6.55E+02	4.47E+01	6.73E+02	4.94E+01	5.43E+02	1.26E+01	
f28	5.14E+02	4.27E+01	4.82E+02	1.66E+01	4.74E+02	1.78E+01	4.77E+02	1.95E+01	4.72E+02	1.41E+01	4.70E+02	1.23E+01	4.59E+02	5.95E-01	
f29	1.49E+03	1.64E+02	3.72E+02	4.60E+01	3.48E+02	4.18E+01	3.36E+02	2.42E+01	3.47E+02	3.17E+01	3.44E+02	3.35E+01	3.16E+02	1.12E+01	
f30	1.29E+07	3.90E+06	6.58E+05	7.71E+04	6.83E+05	7.54E+04	6.85E+05	8.76E+04	6.77E+05	8.57E+04	6.54E+05	6.84E+04	5.82E+05	1.19E+04	
f	WITHOUT LEADER		WITHOUT FOLLOWER		WITH STRATEGY1		WITH STRATEGY2		WITH STRATEGY3		RANDOM SELECTION		CMBO-PG		
Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
f1	3.94E+08	1.13E+08	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
f3	1.09E+05	1.82E+04	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
f4	1.97E+02	4.97E+01	1.48E+02	2.77E+01	1.40E+02	4.06E+01	1.26E+02	4.68E+01	1.26E+02	4.60E+01	1.26E+02	4.74E+01	4.19E+01	3.63E+01	
f5	3.25E+02	1.52E+01	9.46E+00	4.04E+00	6.07E+00	3.19E+00	3.90E+00	2.05E+00	5.50E+00	2.58E+00	5.09E+00	2.02E+00	2.58E+00	1.51E+00	
f6	9.22E+00	1.46E+00	3.54E-08	2.05E-07	2.23E-07	2.24E-08	3.43E-07	1.98E-08	2.58E-07	2.51E-08	2.72E-07	2.44E-08	2.39E-07	3.13E-08	
f7	4.17E+02	1.45E+01	5.69E+01	1.28E+00	6.95E+01	4.67E+01	3.03E+02	5.42E+01	8.43E+01	6.34E+01	1.41E+02	9.44E+01	1.14E+02	8.50E+01	
f8	3.27E+02	1.29E+01	9.93E+00	3.81E+00	6.20E+00	2.99E+00	4.39E+00	2.26E+00	5.48E+00	2.43E+00	5.54E+00	2.50E+00	3.14E+00	1.68E+00	
f9	3.56E+02	1.52E+02	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
f10	1.15E+04	3.12E+02	8.63E+02	3.37E+02	3.67E+02	2.30E+02	2.86E+02	1.76E+02	3.78E+02	2.14E+02	3.56E+02	2.06E+02	1.75E+02	9.19E+01	
f11	3.18E+02	4.81E+01	3.59E+01	8.46E+00	3.11E+01	5.70E+00	3.05E+01	5.44E+00	3.22E+01	7.80E+00	3.08E+01	5.17E+00	2.28E+01	2.71E+00	
f12	7.19E+07	1.61E+07	1.52E+03	4.15E+02	1.38E+03	3.94E+02	1.44E+03	3.03E+02	1.42E+03	4.23E+02	1.44E+03	4.72E+02	1.07E+03	2.76E+02	
f13	4.68E-06	1.71E+06	1.13E+02	4.03E+01	9.90E+01	5.53E+01	8.92E+01	4.17E+01	8.49E+01	4.00E+01	8.29E+01	3.75E+01	4.53E+01	3.21E+01	
f14	1.08E+04	4.61E+03	2.77E+01	4.12E+00	2.69E+01	2.95E+00	2.71E+01	4.08E+00	2.67E+01	2.89E+00	2.78E+01	3.61E+00	2.60E+01	3.46E+00	
f15	2.81E+05	1.20E+05	4.75E+01	1.42E+01	4.15E+01	1.38E+01	3.92E+01	1.19E+01	4.30E+01	1.36E+01	4.20E+01	1.28E+01	2.62E+01	7.33E+00	
f16	2.35E+03	2.09E+02	1.40E+02	3.28E+01	1.40E+02	3.34E+01	1.41E+02	3.68E+01	1.42E+02	3.63E+01	1.32E+02	1.75E+01	1.31E+02	1.24E+01	
f17	1.44E+03	1.60E+02	1.53E+02	6.19E+01	1.28E+02	6.53E+01	1.33E+02	6.91E+01	1.36E+02	5.90E+01	1.16E+02	7.22E+01	9.76E+01	7.11E+01	
f18	3.51E+05	1.15E+05	2.74E+01	3.95E+00	2.76E+01	4.20E+00	2.98E+01	5.78E+00	2.89E+01	6.70E+00	2.75E+01	4.56E+00	2.53E+01	1.98E+00	
f19	1.65E+05	5.83E+04	1.95E+01	4.99E+00	1.77E+01	3.85E+00	1.67E+01	4.49E+00	1.80E+01	5.29E+00	1.74E+01	4.00E+00	1.57E+01	2.85E+00	
f20	1.20E+03	1.87E+02	4.54E+01	1.26E+01	3.49E+01	1.63E+01	2.63E+01	3.59E+00	3.52E+01	2.33E+01	2.88E+01	4.85E+00	2.58E+01	3.59E+00	
f21	5.32E+02	1.29E+01	2.14E+02	3.75E+00	2.11E+02	2.77E+00	2.08E+02	2.51E+00	2.10E+02	2.54E+00	2.10E+02	3.23E+00	2.07E+02	2.03E+00	
f22	2.50E+02	6.09E+01	2.51E+02	3.77E+02	2.04E+02	2.97E+02	1.33E+02	1.10E+02	2.01E+02	3.08E+02	1.45E+02	1.90E+02	1.00E+02	1.20E-08	
f23	7.59E+02	1.71E+01	4.49E+02	1.17E+01	4.40E+02	9.18E+00	4.41E+02	7.02E+00	4.43E+02	7.38E+00	4.43E+02	8.21E+00	4.28E+02	5.73E+00	
f24	8.25E+02	1.51E+01	5.22E+02	6.93E+00	5.18E+02	7.09E+00	5.16E+02	8.09E+00	5.16E+02	7.76E+00	5.15E+02	6.87E+00	5.07E+02	5.00E+00	
f25	5.92E+02	2.99E+01	4.86E+02	1.61E+01	4.80E+02	1.94E+00	4.82E+02	1.16E+01	4.81E+02	2.80E+00	4.81E+02	3.33E+00	4.80E+02	5.	

and 50D, and the mean and standard deviation obtained by the experiment results are recorded in Tables 9–11. The optimal value of each function is expressed in bold. Less than 10^{-8} is considered 0. CMBO-PG outperforms the comparison algorithms on most functions by comparison experiment. From Tables 9–11, WITHOUT FOLLOWER performs worst among the various strategies in all dimensions which means the leading bird flock plays a key role in the algorithm. An expression method is proposed to calculate the gain ratio of the algorithm is proposed to further verify the validity of the multi-strategy learning strategy. The formula is shown in (26).

$$GR = \frac{F_{\text{variant}} - F_{\text{original}}}{F_{\text{original}}} \quad (26)$$

Where GR is the gain ratio, F_{variant} is the fitness value of the comparison mutation strategies, and the F_{original} is the fitness value of the CMBO-PG.

A positive gain ratio indicates that the strategy improves the result of the algorithm on a function, whereas a negative ratio performs the opposite. The 29 functions are represented by the x-axis, and the gain rates are represented by the y-axis. In this paper, the value of GR is utilized to measure the effectiveness of the multi-strategy learning mechanism in CMBO-PG. The total number of combinations of 29 functions with 3 dimensions including 10D, 30D, and 50D is $29 \times 3 = 87$. Compared with the single strategy (WITHOUT STRATEGY1), the CMBO-PG has 50 cases where the gain rate is positive and 4 cases where the gain rate is negative in Fig. 16. Compared with the single strategy (WITHOUT STRATEGY2), the proposed algorithm has 54 positive cases and 4 negative cases from Fig. 17. Compared with the single strategy (WITHOUT STRATEGY3), the multi-strategy learning mechanism has 56 positive cases, 2 negative cases from Fig. 18. Compared with the random selection mechanism (RANDOM SELECTION), the multi-strategy learning mechanism has 49 positive cases, and 3 negative cases in Fig. 19. In contrast with the mechanism of the single strategy, the gain ratio is positive on most functions for the multi-strategy learning mechanism, especially on multimodal and hybrid functions. This shows that the multi-strategy learning mechanism promotes the algorithm to jump out of the locally optimal well by selecting different strategies for different functions. The effect of the multi-strategy learning mechanism is better than the random selection mechanism from Fig. 19. It indicates except for a few functions, the gain is positive in most functions, especially in multimodal functions and hybrid functions, which confirms the effect of jumping out of the local optimum in the following flock. Therefore, the multi-strategy learning mechanism plays a significant role in improving the performance of the algorithm.

In this paper, the distribution of individual fitness values for WITHOUT LEADER, WITHOUT FOLLOWER, and CMBO-PG on the functions f1, f3, f7, f8, f25, f28 are shown in Figs. 20–25. The

effectiveness of the population-based co-evolution mechanism in CMBO-PG is verified by balancing the exploration and the exploitation. The two-dimensional function model diagram is fitted to observe the properties of the function by the thermodynamic diagram. The areas with optimal fitness values are indicated in red in the thermodynamic diagram. The areas with excellent fitness values are indicated in red and the global optimum is marked with yellow stars in the thermodynamic diagram. The remaining four images are individual distribution diagrams viewed from a vertical-view perspective. 2000 individuals (blue dots) iterate 10 times depending on the maximum evaluation times ($2 \times 10000 = 20000$). Individual distribution diagrams of the algorithm include $0.1 \times \text{max_nfees}$, $0.3 \times \text{max_nfees}$, $0.5 \times \text{max_nfees}$, and max_nfees are recorded to observe the balance of the exploration and the exploitation.

As shown in Figs. 20–25, the individual distribution of the iterative process for WITHOUT LEADER is more diffused than WITHOUT FOLLOWER and CMBO-PG. The individual distribution of WITHOUT FOLLOWER is more concentrated than other algorithms, and CMBO-PG is between the two. The results of the individual distribution indicate that CMBO-PG takes an effective balance of exploration and exploitation. It is worth mentioning that WITHOUT LEADER generates new individuals who are concentrated in the valley bottom area from Figs. 20(a)–25(a), which means individuals automatically cluster according to the fitness value. WITHOUT FOLLOWER generates new individuals based on the mean value from Figs. 20(b)–25(b), which is distributed in an ellipsoid area. For the unimodal functions, WITHOUT FOLLOWER converges to the optimal area quickly from Figs. 20(a)–21(a). However, the individuals fail to continuously converge to a valley bottom area in multimodal functions from Figs. 22(a)–23(a). The reason is that the mean value is deceived out of diversity due to the aggregation of multiple local optimal. The automatic clustering capacity of WITHOUT LEADER and the fast convergence property of WITHOUT FOLLOWER are combined for CMBO-PG, which converges individuals to a small number of valley areas for locating the optimal solution. This is the reason why CMBO-PG performs well on multimodal functions. As shown in Figs. 24(a)–25(a), the individuals generated by WITHOUT LEADER discover all valley bottom areas, but with poor convergence rate. The individuals generated by WITHOUT FOLLOWER are deceived to a regional extreme point ignoring the valley area on the other side from Figs. 24(b)–25(b). Hence, the individuals fall into the local optimum leading to premature convergence. As shown in Figs. 24(c)–25(c) the individuals generated by CMBO-PG discover multiple valley areas and quickly converge. The individuals are distributed in various valley regions due to the multi-strategy learning mechanism, and the individuals converge rapidly to search for the global optimum.

Through the analysis above, the conclusion is that CMBO-PG

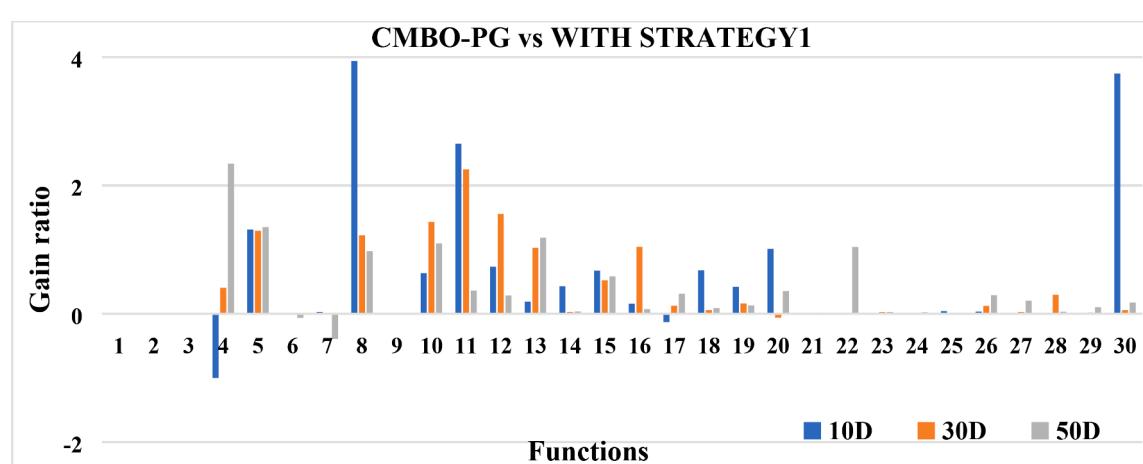


Fig. 16. The gain ratio diagram for the CMBO-PG to WITH STRATEGY1.

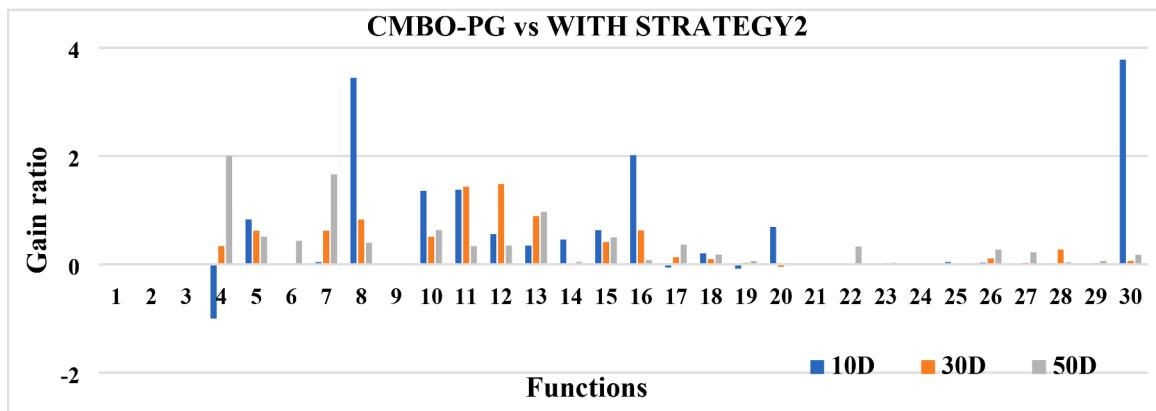


Fig. 17. The gain ratio diagram for the CMBO-PG to WITH STRATEGY2.

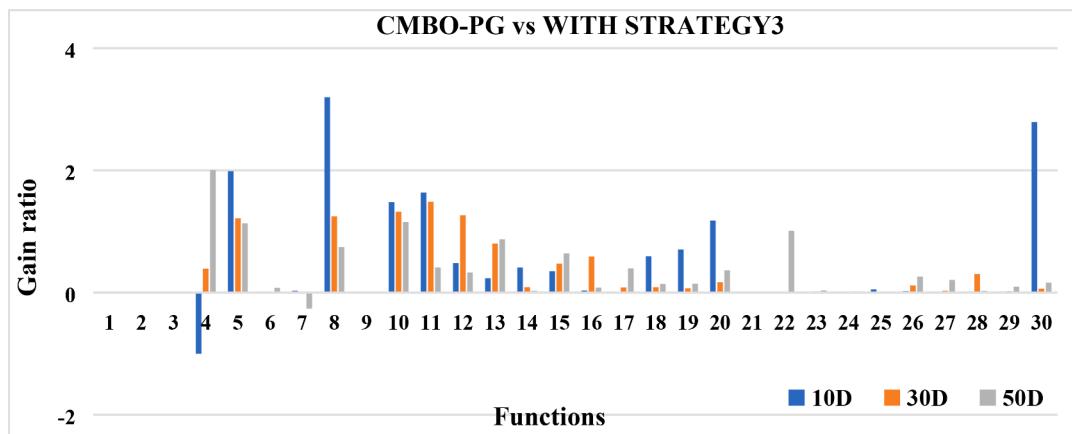


Fig. 18. The gain ratio diagram for the CMBO-PG to WITH STRATEGY3.

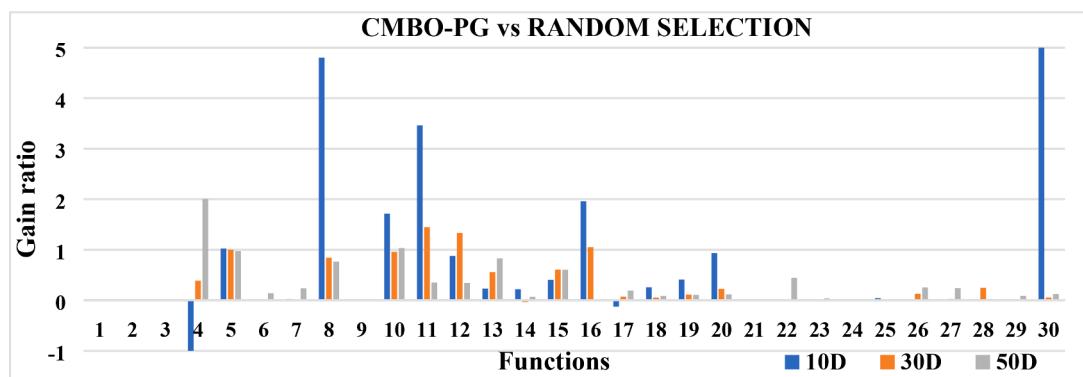


Fig. 19. The gain ratio diagram for the CMBO-PG to RANDOM SELECTION.

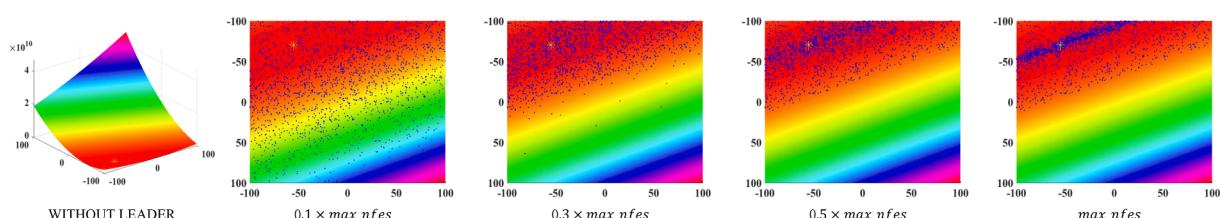


Fig. 20a. Distribution of individual fitness values on the f1 function (WITHOUT LEADER).

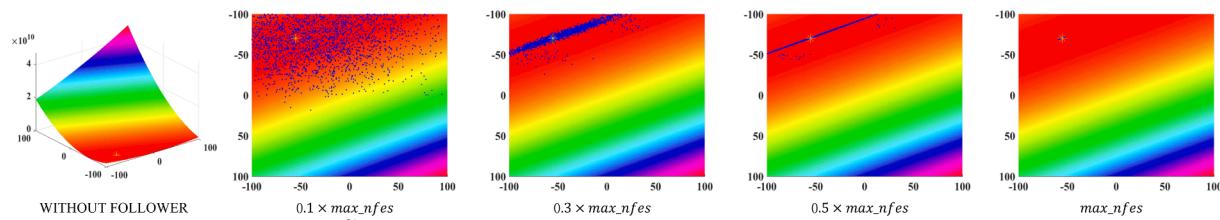


Fig. 20b. Distribution of individual fitness values on the f_1 function (WITHOUT FOLLOWER).

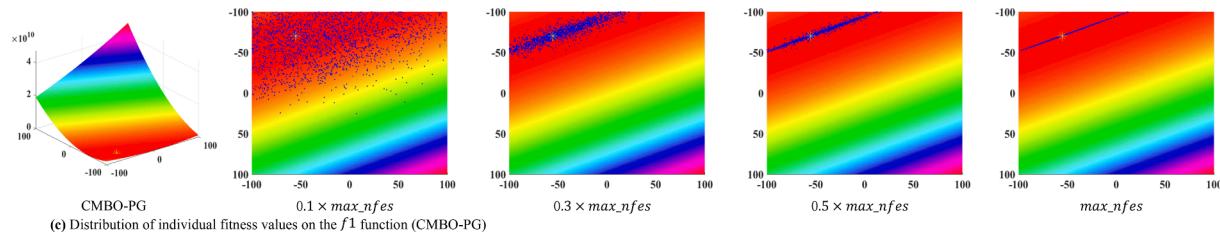


Fig. 20c. Distribution of individual fitness values on the f_1 function (CMBO-PG).

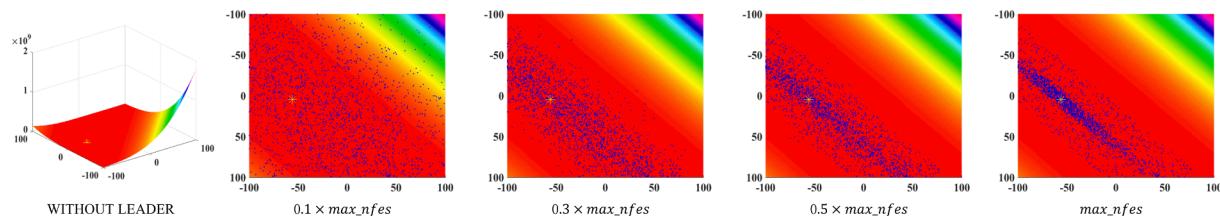


Fig. 21a. Distribution of individual fitness values on the f_3 function (WITHOUT LEADER).

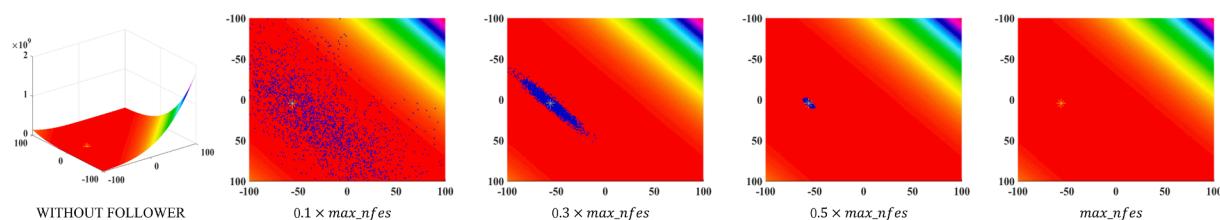


Fig. 21b. Distribution of individual fitness values on the f_3 function (WITHOUT FOLLOWER).

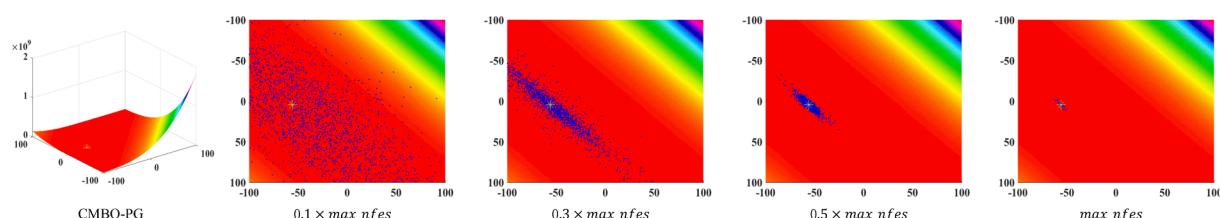


Fig. 21c. Distribution of individual fitness values on the f_3 function (CMBO-PG).

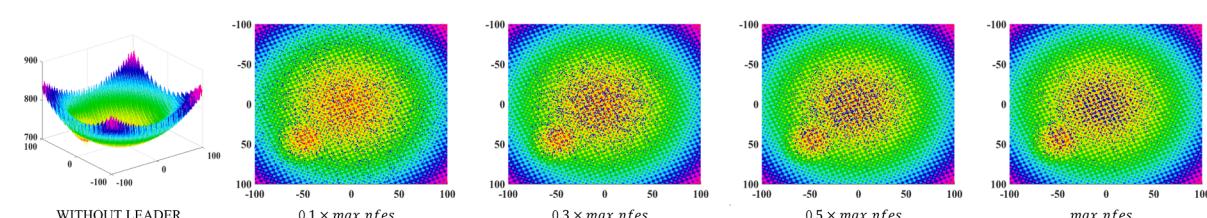


Fig. 22a. Distribution of individual fitness values on the f_7 function (WITHOUT LEADER).

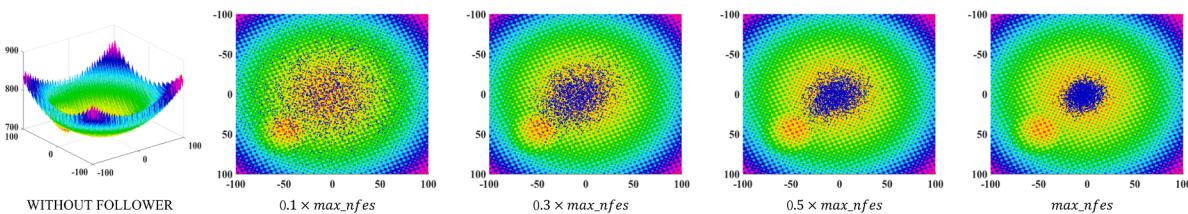


Fig. 22b. Distribution of individual fitness values on the f_7 function (WITHOUT FOLLOWER).

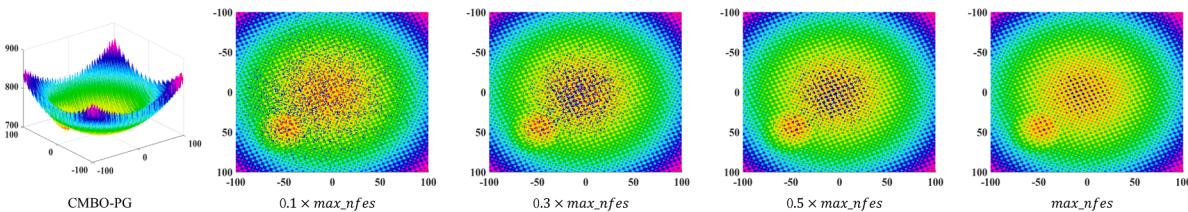


Fig. 22c. Distribution of individual fitness values on the f_7 function (CMBO-PG).

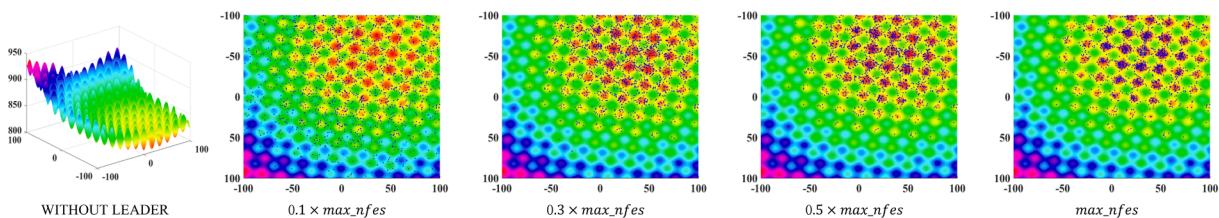


Fig. 23a. Distribution of individual fitness values on the f_8 function (WITHOUT LEADER).

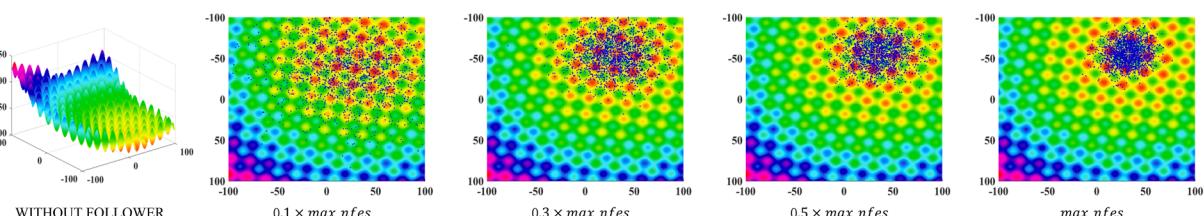


Fig. 23b. Distribution of individual fitness values on the f_8 function (WITHOUT FOLLOWER).

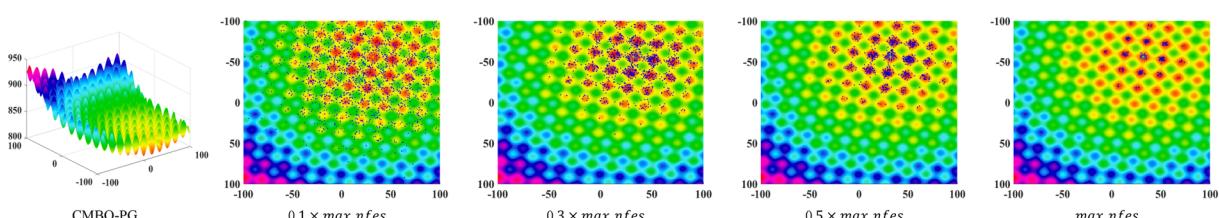


Fig. 23c. Distribution of individual fitness values on the f_8 function (CMBO-PG).

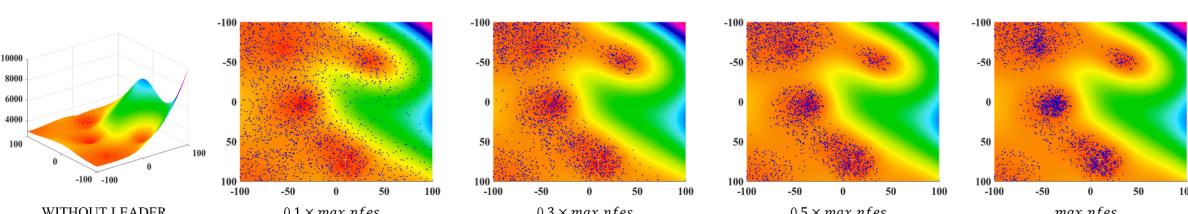


Fig. 24a. Distribution of individual fitness values on the f_{25} function (WITHOUT LEADER).

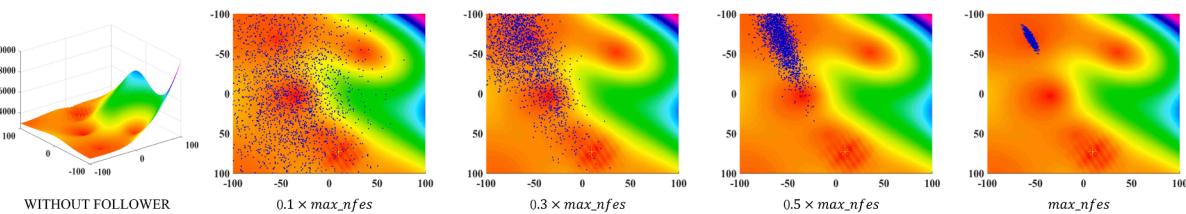


Fig. 24b. Distribution of individual fitness values on the f_{25} function (WITHOUT FOLLOWER).

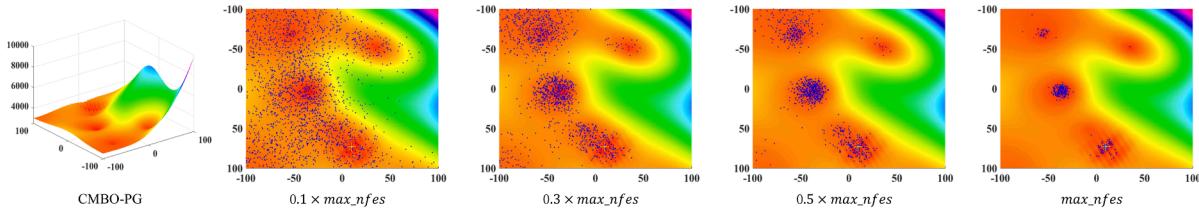


Fig. 24c. Distribution of individual fitness values on the f_{25} function (CMBO-PG).

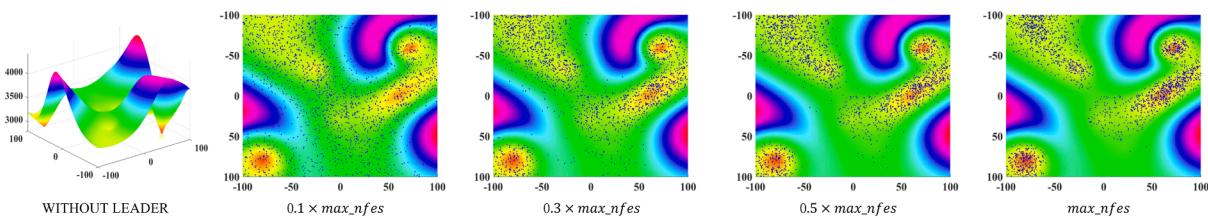


Fig. 25a. Distribution of individual fitness values on the f_{28} function (WITHOUT LEADER).

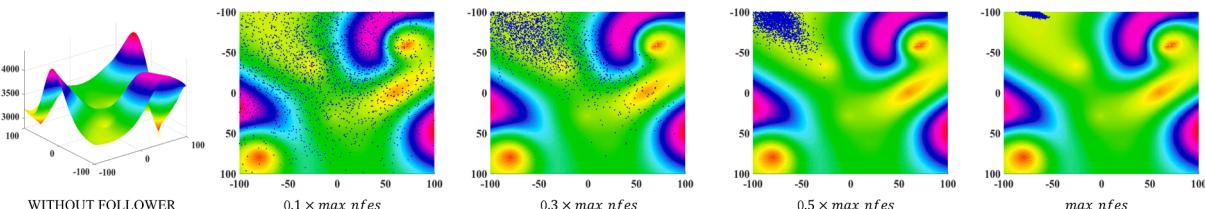


Fig. 25b. Distribution of individual fitness values on the f_{28} function (WITHOUT FOLLOWER).

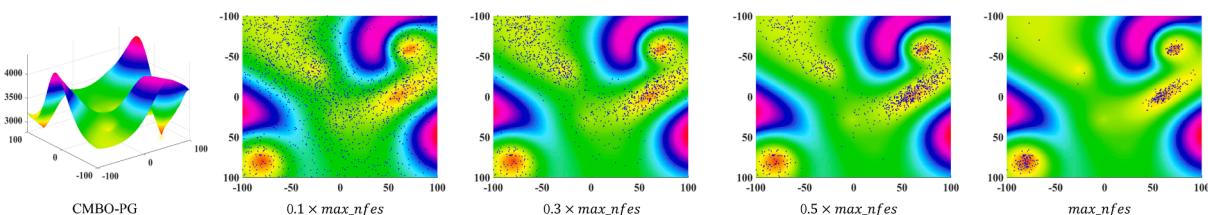


Fig. 25c. Distribution of individual fitness values on the f_{28} function (CMBO-PG).

balances the exploration and exploitation well by the co-evolutionary method of the leading flock and the following flock.

4.6. The analysis of algorithm complexity

This section discusses the algorithm complexity of CMBO-PG from time complexity and computational complexity.

4.6.1. The analysis of time complexity

The proposed algorithm is tested on the CEC'17 benchmark which includes four dimensions ($10D, 30D, 50D, 100D$). The time complexity of

CMBO-PG is determined by the population size, the dimension of the problem, the sorting process of the neighborhood solutions, the weight of LSTM, and the operation of PG. The specific time complexity of CMBO-PG is analyzed as follows:

- (1) The population size is N , and the time complexity of the initial population method is $O(N)$.
- (2) The evaluation of fitness value is related to the dimension of the problem, and the time complexity of the initial evaluation is $O(D^*N)$.

- (3) There are N neighborhood solutions generated by the leading flock, the generation of neighborhood solutions costs $O(N)$, and the evaluation of neighbors costs $O(D^*N)$.
- (4) The quicksort process of neighborhood solutions costs $O(N^*\log N)$, and costs $O(N^2)$ under the worst condition.
- (5) A total of L_2 neighborhood solutions are generated by the following flock. The complexity of the neighborhood solutions is $O(L_2)$, and the evaluation of neighborhood solutions for the following flock costs $O(D^*L_2)$.
- (6) In the LSTM network, the derivatives need to be stored and updated, and the time complexity is determined by the number of weights W . The LSTM network cost $O(W)$.
- (7) In PG, the historical information stored in the memory bank is regarded as the trajectory, and the gradient information is calculated by the trajectory. The time complexity of PG is $O(1)$.

Throughout the analysis above, the time complexity of CMBO-PG is $O(N) + O(D^*N) + O(N) + O(D^*N) + O(N^2) + O(D^*L_2) + O(W) + O(1)$. The conclusion from the analysis of time complexity is that time complexity is still acceptable although certain mechanisms are added to improve the performance of the proposed algorithm.

4.6.2. The analysis of computational complexity

The determination method of computational complexity is introduced by the technical documentation of the CEC'17 benchmark (Awad, Ali, Suganthan, Liang, & Qu, 2017). It defined T0, T1, and T2, where T0 is the cost time running a specific program, T1 is the calculation time evaluating a certain fitness function, and T2 is the calculation time for the tested algorithm under certain evaluation times. The computational complexity of the algorithm is evaluated by $(T_2-T_1)/T_0$.

In this paper, we evaluate six algorithms including EMBO (the variant of MBO), AAVS (the variant of EDA with fast convergence), TSPSO (state-of-arts algorithm), RLBSO (a meta-heuristic algorithm with RL), jSO (the winner algorithm of CEC'17), and CMBO-PG in the dimensions of 10D, 30D, 50D, and 100D. The evaluation function is f29, and the number of evaluations is 200,000 (Sallam, Abdel-Basset, El-Abd, & Wagdy, 2022). Each algorithm runs 20 times to ensure the reliability of experimental results, and the mean value of results is recorded as T2. The calculation complexity and fitness values of the six algorithms are shown in Table 12. The optimal values of computational complexity and

fitness in all algorithms are shown in bold.

From Table 12, the computational complexity of the CMBO-PG algorithm is only inferior to AAVS and TAPSO in 10D, 30D, and 50D. In terms of performance, CMBO-PG is superior to all comparison algorithms. This shows that CMBO-PG spends less time to get the best results in 10D, 30D, and 50D. The time cost of this algorithm is high in 100D due to the time-consuming operation including derivative in high-dimensional data and update of neuron parameters in the LSTM network. The performance of CMBO-PG is only inferior to the AAVS. From Fig. 10, the performance of CMBO-PG is superior to AAVS under higher evaluation times (1,000,000) which shows that CMBO-PG has a good balance between the exploration and development capabilities of the algorithm. Therefore, the CMBO-PG consumed slightly higher resources and obtain competent performance.

5. Conclusion and future work

The CMBO-PG is proposed to solve complex continuous problems in this paper. A multi-strategy learning mechanism, which uses LSTM to make better decisions, is proposed to guide the evolution process of meta-heuristic algorithms. The method of PG incorporates the knowledge extracted from historical data within the search process to optimize the LSTM. Under the consistent control condition, we compared the performance of CMBO-PG with a multi-strategy learning mechanism, single-strategy mechanism, and randomly selecting mechanism. The experimental results show that the direction of the meta-heuristic method guided by LSTM and PG is effective and promising. We designed a co-evolutional mechanism to balance the exploitation and exploration stages. The visualization of the execution elaborated on the distribution of individuals during the evolution process. The results indicate the characteristic of fast convergence of the leading flock and the diversity of the following flock. The efficient balance for exploitation and exploration when the co-evolutional mechanism is adopted. CMBO-PG is compared with 12 comparison algorithms including certain state-of-the-art algorithms on the CEC2017 benchmark test suite. Experimental results show that CMBO-PG has significant advantages in solving complex continuous real-parameter optimization problems. The Friedman test and Wilcoxon test demonstrate that the proposed algorithm is a statistically critical difference from the comparison algorithm.

The conclusion of the summary above is that: (1) Given the nonlinear

Table 12
The computational complexity of CMBO-PG and comparison algorithms.

Dimensionality	Indicator	EMBO	AAVS	TAPSO	RLBSO	jSO	CMBO-PG
10D	T0	6.95E-02	6.95E-02	6.95E-02	6.95E-02	6.95E-02	6.95E-02
	T1	1.29E+00	1.29E+00	1.29E+00	1.29E+00	1.29E+00	1.29E+00
	T2	8.12E+01	4.76E+00	3.39E+00	1.07E+02	1.27E+01	7.48E+00
	(T2-T1)/T0	1.15E+03	4.99E+01	3.02E+01	1.52E+03	1.64E+02	8.90E+01
	Fitness	2.57E+02	2.38E+02	2.87E+02	2.80E+02	2.32E+02	2.30E+02
30D	T0	6.95E-02	6.95E-02	6.95E-02	6.95E-02	6.95E-02	6.95E-02
	T1	4.00E+00	4.00E+00	4.00E+00	4.00E+00	4.00E+00	4.00E+00
	T2	1.14E+02	8.22E+00	1.20E+01	1.47E+02	1.99E+01	1.16E+01
	(T2-T1)/T0	1.59E+03	6.06E+01	1.15E+02	2.05E+03	2.29E+02	1.09E+02
	Fitness	6.70E+02	4.37E+02	7.87E+02	3.70E+02	4.49E+02	4.08E+02
50D	T0	6.95E-02	6.95E-02	6.95E-02	6.95E-02	6.95E-02	6.95E-02
	T1	7.94E+00	7.94E+00	7.94E+00	7.94E+00	7.94E+00	7.94E+00
	T2	1.48E+02	1.29E+01	1.15E+01	1.34E+02	2.43E+01	2.00E+01
	(T2-T1)/T0	2.02E+03	7.17E+01	5.15E+01	1.82E+03	2.35E+02	1.74E+02
	Fitness	1.11E+03	5.15E+02	1.28E+03	1.16E+03	5.61E+02	5.06E+02
100D	T0	6.95E-02	6.95E-02	6.95E-02	6.95E-02	6.95E-02	6.95E-02
	T1	2.47E+01	2.47E+01	2.47E+01	2.47E+01	2.47E+01	2.47E+01
	T2	2.33E+02	3.49E+01	3.83E+01	1.60E+02	3.61E+01	2.87E+02
	(T2-T1)/T0	3.00E+03	1.47E+02	1.96E+02	1.94E+03	1.64E+02	3.78E+03
	Fitness	4.98E+03	2.50E+03	3.33E+03	3.43E+03	3.17E+03	2.55E+03

and non-differentiable characteristics of the strategy selection in the meta-heuristic method, it is a worthy reference to use the neural network model, which is guided by RL, fitting the distribution probability of different strategies. (2) It is an effective attempt to guide the meta-heuristic by the RL method in the way of online learning. the proposed algorithm achieved good results in performance, although it is time-consuming to solve large-scale problems. (3) The co-evolutional mechanism is considered an effective method to balance the convergence and the diversity of meta-heuristic algorithms.

In the future, we will further the current research from the following aspects: (1) CMBO-PG is still a time-consuming algorithm although we have improved the RL method and LSTM model to obtain better performance. In the following work, we will further study the combinatorial methods of RL and meta-heuristic algorithms to reduce the consumption of time. In addition, we will concern with the model optimization of LSTM. (2) We will apply the proposed algorithm to address realistic industrial application problems, including production scheduling problems, mechanical processing problems, and chemical production problems. The focus of future research is the application of the algorithm guided by extracting knowledge from specific problems in solving large-scale complex problems.

CRediT authorship contribution statement

Fuqing Zhao: Funding acquisition, Investigation, Supervision. **Tao Jiang:** Investigation, Software, Methodology, Resources, Validation, Visualization, Writing - original draft, Writing - review & editing. **Tianpeng Xu:** Project administration. **Ningning Zhu:** Conceptualization, Formal analysis. **Jonrinaldi:** Visualization.

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.

Data availability

Data will be made available on request.

Acknowledgement

This work was financially supported by the National Natural Science Foundation of China under grant 62063021. It was also supported by the High-level Foreign Experts Project of Gansu Province under Grant 22JR10KA007, the Key Research Programs of Science and Technology Commission Foundation of Gansu Province (21YF5WA086), Lanzhou Science Bureau project (2018-rc-98), and Project of Gansu Natural Science Foundation (21JR7RA204), respectively.

References

- Altinoz, O. T., & Yilmaz, A. E. (2019). Multiobjective Hooke-Jeeves algorithm with a stochastic Newton-Raphson-like step-size method. *Expert Systems with Applications*, *117*, 166–175.
- Arnouts, H., Goos, P., & Jones, B. (2010). Design and analysis of industrial strip-plot experiments. *Quality & Reliability Engineering International*, *26*, 127–136.
- Awad, N. H., Ali, M. Z., Suganthan, P. N., Liang, J. J., & Qu, B. Y. (2017). Problem Definitions and Evaluation Criteria for the CEC 2017.
- Brest, J., Maucec, M. S., Boskovic, B., & Ieee. (2017). Single Objective Real-Parameter Optimization: Algorithm jSO. In *2017 IEEE CONGRESS ON EVOLUTIONARY COMPUTATION (CEC)* (pp. 1311-1318).
- Cai, J., Lei, D., Wang, J., & Wang, L. (2022). A novel shuffled frog-leaping algorithm with reinforcement learning for distributed assembly hybrid flow shop scheduling. *International Journal Of Production. Research*.
- Chen, D. B., Ge, Y. Y., Wan, Y. J., Deng, Y., Chen, Y., & Zou, F. (2022). Poplar optimization algorithm: A new meta-heuristic optimization technique for numerical optimization and image segmentation. *Expert Systems with Applications*, *200*.
- Chen, J., Xing, H., Xiao, Z., Xu, L., & Tao, T. (2021). A DRL Agent for Jointly Optimizing Computation Offloading and Resource Allocation in MEC. *IEEE Internet of Things Journal*, *8*, 17508–17524.
- Civicioglu, P., & Besdok, E. (2022). Contrast stretching based pansharpening by using weighted differential evolution algorithm. *Expert Systems with Applications*, *208*.
- Demiray, J., & Schuurmans, D. (2006). Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research*, *7*, 1–30.
- Deng, G., Xu, M., Zhang, S., Jiang, T., & Su, Q. (2021). Migrating birds optimization with a diversified mechanism for blocking flow shops to minimize idle and blocking time. *Applied Soft Computing*.
- Duman, E., Uysal, M., & Alkaya, A. F. (2012). Migrating Birds Optimization: A new metaheuristic approach and its performance on quadratic assignment problem. *Information Sciences*, *217*, 65–77.
- Gao, L., & Pan, Q. K. (2016). A shuffled multi-swarm micro-migrating birds optimizer for a multi-resource-constrained flexible job shop scheduling problem. *Information Sciences*, *372*, 655–676.
- Gao, S. C., Yu, Y., Wang, Y. R., Wang, J. H., Cheng, J. J., & Zhou, M. C. (2021). Chaotic Local Search-Based Differential Evolution Algorithms for Optimization. *IEEE TRANSACTIONS ON SYSTEMS MAN CYBERNETICS-SYSTEMS*, *51*, 3954–3967.
- Gölcük, İ., & Ozsoydan, F. B. (2021). Q-learning and hyper-heuristic based algorithm recommendation for changing environments. *Engineering Applications of Artificial Intelligence*, *102*.
- Han, D., Tang, Q., Zhang, Z., & Li, Z. (2020). An Improved Migrating Birds Optimization Algorithm for a Hybrid Flow Shop Scheduling within Steel Plants. *Mathematics*, *8*.
- Huang, C., Li, Y., & Yao, X. (2020). A Survey of Automatic Parameter Tuning Methods for Metaheuristics. *IEEE Transactions on Evolutionary Computation*, *24*, 201–216.
- Hx, A., Jing, Z. A., Rong, Q. B., Pd, A., Sl, A., & Mai, A. (2021). An ACO for energy-efficient and traffic-aware virtual machine placement in cloud computing. *Swarm And Evolutionary Computation*, *68*.
- Kirkpatrick, S., Gelatt, C. D., Jr., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, *220*, 671–680.
- Li, Y., Wang, S., Yang, H., Chen, H., & Yang, B. (2023). Enhancing differential evolution algorithm using leader-adjoint populations. *Information Sciences*, *622*, 235–268.
- Liang, Y., Ren, Z., Yao, X., Feng, Z., Chen, A., & Guo, W. (2020). Enhancing Gaussian Estimation of Distribution Algorithm by Exploiting Evolution Direction With Archive. *IEEE Trans Cybern*, *50*, 140–152.
- Ma, L. B., Cheng, S., & Shi, Y. H. (2021). Enhancing Learning Efficiency of Brain Storm Optimization via Orthogonal Learning Design. *IEEE TRANSACTIONS ON SYSTEMS MAN CYBERNETICS-SYSTEMS*, *51*, 6723–6742.
- Meng, T., Pan, Q. K., Li, J. Q., & Sang, H. Y. (2018). An improved migrating birds optimization for an integrated lot-streaming flow shop scheduling problem. *Swarm And Evolutionary Computation*, *38*, 64–78.
- Mohamed, A. W., Hadi, A. A., Mohamed, A. K., Awad, N. H., & Ieee.. (2020). Evaluating the Performance of Adaptive Gaining-Sharing Knowledge Based Algorithm on CEC 2020 Benchmark Problems. *IEEE Congress on Evolutionary Computation (CEC) as part of the IEEE World Congress on Computational Intelligence (IEEE WCCI)*. Electr Network.
- Nekoo, S. R., Acosta, J. A., & Ollero, A. (2022). A search algorithm for constrained engineering optimization and tuning the gains of controllers. *Expert Systems with Applications*, *206*.
- Nguyen, T. T., Nguyen, N. D., & Nahavandi, S. (2020). Deep Reinforcement Learning for Multiagent Systems: A Review of Challenges, Solutions, and Applications. *IEEE Trans Cybern*, *50*, 3826–3839.
- Niroomand, S., Hadi-Vencheh, A., Şahin, R., & Vizvári, B. (2015). Modified migrating birds optimization algorithm for closed loop layout with exact distances in flexible manufacturing systems. *Expert Systems with Applications*, *42*, 6586–6597.
- Pan, Q.-K., & Dong, Y. (2014). An improved migrating birds optimisation for a hybrid flowshop scheduling with total flowtime minimisation. *Information Sciences*, *277*, 643–655.
- Qiao, Y., Wu, N. Q., He, Y. F., Li, Z. W., & Chen, T. (2022). Adaptive genetic algorithm for two-stage hybrid flow-shop scheduling with sequence-independent setup time and no-interruption requirement. *Expert Systems with Applications*, *208*.
- Qin, A. K., Huang, V. L., & Suganthan, P. N. (2009). Differential Evolution Algorithm With Strategy Adaptation for Global Numerical Optimization. *IEEE Transactions on Evolutionary Computation*, *13*, 398–417.
- Ren, Z., Liang, Y., Wang, L., Zhang, A., Pang, B., & Li, B. (2018). Anisotropic adaptive variance scaling for Gaussian estimation of distribution algorithm. *Knowledge-Based Systems*, *146*, 142–151.
- Sallam, K. M., Abdel-Basset, M., El-Abd, M., & Wagdy, A. (2022). IMODEII: An Improved IMODE algorithm based on the Reinforcement Learning. In *In 2022 IEEE Congress on Evolutionary Computation (CEC)* (pp. 1–8).
- Segredo, E., Lalla-Ruiz, H., Emma, V., & Stefan.. (2018). On the performance of the hybridisation between migrating birds optimisation variants and differential evolution for large scale continuous problems. *Expert Systems with Application*.
- Simon, D. (2008). Biogeography-Based Optimization. *IEEE Transactions on Evolutionary Computation*, *12*, 702–713.
- Song, W., Chen, X., Li, Q., & Cao, Z. (2023). Flexible Job-Shop Scheduling via Graph Neural Network and Deep Reinforcement Learning. *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS*, *19*, 1600–1610.
- Sun, J., Liu, X., Back, T., & Xu, Z. (2021). Learning Adaptive Differential Evolution Algorithm From Optimization Experiences by Policy Gradient. *IEEE Transactions on Evolutionary Computation*, *25*, 666–680.
- Tongur, V., & Ülker, E. (2018). PSO-based improved multi-flocks migrating birds optimization (IMFMB) algorithm for solution of discrete problems. *Soft Computing*, *23*, 5469–5484.

- Wang, J.-J., & Wang, L. (2022). A Cooperative Memetic Algorithm with Learning-Based Agent for Energy-Aware Distributed Hybrid Flow-Shop Scheduling. *IEEE Transactions on Evolutionary Computation*, *26*, 461–475.
- Wang, K., Gong, W. Y., Liao, Z. W., & Wang, L. Hybrid Niching-Based Differential Evolution With Two Archives for Nonlinear Equation System. *IEEE TRANSACTIONS ON SYSTEMS MAN CYBERNETICS-SYSTEMS*.
- Wang, X., Xing, H., Zhan, D., Luo, S., Dai, P., & Iqbal, M. A. (2021). A two-stage approach for multicast-oriented virtual network function placement. *Applied Soft Computing*, *112*.
- Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, *1*, 67–82.
- Wu, G., Mallipeddi, R., Suganthan, P. N., Wang, R., & Chen, H. (2016). Differential evolution with multi-population based ensemble of mutation strategies. *Information Sciences*, *329*, 329–345.
- Wu, G., Shen, X., Li, H., Chen, H., Lin, A., & Suganthan, P. N. (2018). Ensemble of differential evolution variants. *Information Sciences*, *423*, 172–186.
- Xia, X., Gui, L., He, G., Wei, B., Zhang, Y., Yu, F., ... Zhan, Z.-H. (2020). An expanded particle swarm optimization based on multi-exemplar and forgetting ability. *Information Sciences*, *508*, 105–120.
- Xia, X., Gui, L., Yu, F., Wu, H., Wei, B., Zhang, Y. L., & Zhan, Z. H. (2020). Triple Archives Particle Swarm Optimization. *IEEE Trans Cybern*, *50*, 4862–4875.
- Xiao, Q., Guo, X., & Li, D. Partial disassembly line balancing under uncertainty: robust optimisation models and an improved migrating birds optimisation algorithm.
- Xiao, Z., Xu, X., Xing, H., Song, F., Wang, X., & Zhao, B. (2021). A federated learning system with enhanced feature extraction for human activity recognition. *Knowledge-Based Systems*, *229*.
- Yang, M., Zhou, A. M., Li, C. H., & Yao, X. (2021). An Efficient Recursive Differential Grouping for Large-Scale Continuous Problems. *IEEE Transactions on Evolutionary Computation*, *25*, 159–171.
- Yang, Z. (2022). Adaptive stochastic conjugate gradient for machine learning. *Expert Systems with Applications*, *206*.
- Yousefnejad, H., & Monfared, M. S. (2022). A control algorithm for a non-stationary batch service production system using Kalman filter. *Expert Systems with Applications*, *207*.
- Zhang, Z., Ong, Y. S., Wang, D. Q., & Xue, B. Q. (2021). A Collaborative Multiagent Reinforcement Learning Method Based on Policy Gradient Potential. *IEEE Transactions on Cybernetics*, *51*, 1015–1027.
- Zhao, F., Ding, R., Wang, L., Cao, J., & Tang, J. (2021). A hierarchical guidance strategy assisted fruit fly optimization algorithm with cooperative learning mechanism. *Expert Systems with Applications*, *183*.
- Zhao, F., Hu, X., Wang, L., Zhao, J., Tang, J., & Jonrinaldi.. (2022). A reinforcement learning brain storm optimization algorithm (BSO) with learning mechanism. *Knowledge-Based Systems*, *235*.
- Zhao, F., Wang, Q., & Wang, L. (2023). An inverse reinforcement learning framework with the Q-learning mechanism for the metaheuristic algorithm. *Knowledge-Based Systems*, *265*.
- Zhao, F., Zhao, J., Wang, L., Cao, J., & Tang, J. (2021). A hierarchical knowledge guided backtracking search algorithm with self-learning strategy. *Engineering Applications of Artificial Intelligence*, *102*.
- Zhao, F., Zhou, G., Wang, L., Xu, T., Zhu, N., & Jonrinaldi.. (2022). A two-stage cooperative scatter search algorithm with multi-population hierarchical learning mechanism. *Expert Systems with Applications*, *203*.
- Zhao, F., Zhu, B., Wang, L., Xu, T., Zhu, N., & Jonrinaldi, J. (2022). An offline learning co-evolutionary algorithm with problem-specific knowledge. *Swarm And Evolutionary Computation*, *75*.
- Zhao, F. Q., Zhang, H., Wang, L., Ma, R., Xu, T. P., Zhu, N. N., & Jonrinaldi.. (2022). A surrogate-assisted Jaya algorithm based on optimal directional guidance and historical learning mechanism. *Engineering Applications of Artificial Intelligence*, *111*.