Information Sciences 608 (2022) 1157–1182

Contents lists available at ScienceDirect

Information Sciences

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

An efficient mixture sampling model for gaussian estimation of distribution algorithm

Qianlong Dang [a], Weifeng Gao [a,*], Maoguo Gong [b]

[a] School of Mathematics and Statistics, Xidian University, Xi'an 710126, China
[b] Key Laboratory of Intelligent Perception and Image Understanding, International Research Center for Intelligent Perception and Computation, Ministry of Education, Xidian University, Xi'an 710071, China

ARTICLE INFO

Article history:
Received 22 March 2021
Received in revised form 23 February 2022
Accepted 2 July 2022
Available online 06 July 2022

Keyword:
Gaussian estimation of distribution algorithm
Evolutionary algorithm
Premature convergence
Efficient mixture sampling model

ABSTRACT

Estimation of distribution algorithm (EDA) is a stochastic optimization algorithm based on probability distribution model and has been widely applied in global optimization. However, the random sampling of Gaussian EDA (GEDA) usually suffers from the poor diversity and the premature convergence, which severely limits its performance. This paper analyzes the shortcomings of the random sampling and develops an efficient mixture sampling model (EMSM). EMSM can explore more promising regions and utilize the unsuccessful mutation vectors, which achieves a good tradeoff between the diversity and the convergence. Moreover, the feasibility analysis of EMSM is studied. A new GEDA variant named EMSM-EDA is developed, which combines EMSM with enhancing Gaussian estimation of distribution algorithm (EDA^2). The experimental results on IEEE CEC2013 and IEEE CEC2014 test suites demonstrate that EMSM-EDA is efficient and competitive.

1. Introduction

Many engineering and scientific problems can be transformed into black-box optimization problems, and these problems are important and difficult. Since the analytical expression of black-box optimization problems is unknown, the traditional gradient-based algorithms are difficult to optimize. In recent years, evolutionary algorithms (EAs) [1–5] have been found to be effective in solving these types of optimization problems. Estimation of distributed algorithm (EDA) [6–10] is a special branch of EAs. Compared with crossover and mutation operators of other EAs, EDA generates candidate solutions by sampling from the probability distribution. Specifically, EDA utilizes the statistical information of good solutions to establish the corresponding probability distribution. Then, new candidate solutions are generated by utilizing the established probabilistic distribution. Over the last decades, EDA has been applied to many discrete optimization problems [11–15], continuous optimization problems [16–19], and mixed-variable optimization problems [20,21].

The traditional Gaussian sampling in EDA has the drawbacks of the poor diversity and the premature convergence. Thus, some efforts have been made to solve these issues and improve the performance of EDA. Zhou et al. [22] utilized Gaussian distribution and Cauchy distribution to propose a sampling algorithm for EDA, which makes use of the good exploration ability of Cauchy distribution. Velasco et al. [23] presented an adaptive Gibbs sampling, which combines with a local search strategy to generate new promising solutions. Ren et al. [24] constructed a novel probability density estimator to improve

[*] Corresponding author.

https://doi.org/10.1016/j.ins.2022.07.016
0020-0255/© 2022 Elsevier Inc. All rights reserved.

the sampling quality. Lin et al. [25] proposed a mutual information EDA with hybrid sampling mechanism to relieve the premature convergence. Although the above work has been done in sampling phase of EDA, these methods still utilize the random sampling to generate the mutation vectors and suffer from the poor diversity and the premature convergence in some degree.

The general EDA randomly generates the mutation vectors by the multivariate Gaussian distribution. Nevertheless, this method often suffers from a biased sample that does not uniformly cover the sampling space. Therefore, the Gaussian sampling is unable to explore some promising regions and has the poor diversity, which results in the premature convergence. From this point of view, this paper introduces the idea of the uniform design sampling for improving the diversity and the exploration ability of the mutation vectors in GEDA. Moreover, the orthogonal method orthogonalizes the mutation vectors and ensures that more dimension directions are explored. Thus, the diversity of the mutation vectors can be further enhanced. In addition, GEDA only selects a part of the successful mutation vectors in each generation to construct the Gaussian distribution of the next generation. Most of the unsuccessful mutation vectors do not contribute to the search of the algorithm. It is considered that the reverse direction of the unsuccessful mutation vectors may be the promising direction. Therefore, the mirrored sampling is used to extract the useful information of the unsuccessful mutation vectors.

In this paper, an EMSM method is proposed for improving the performance of GEDA. The main contributions are listed as follows:

1. The random sampling suffers from the poor diversity and the premature convergence. To solve these issues, this paper designs an EMSM method that can maintain a good tradeoff between the diversity and the convergence by the uniform method, the orthogonal method, and the mirrored sampling.
2. The theoretical analysis of EMSM is given. Firstly, the diversity of the uniform method and the orthogonal method is studied by the volume of the probability density ellipsoid. Then, the convergence and the diversity of the mirrored sampling are analyzed by the progress rate and the volume of the probability density ellipsoid. Finally, it can be concluded that EMSM can improve the exploration ability and the convergence rate.
3. EMSM is applied to EDA^2 [26] that is an excellent variant of GEDA, named EMSM-EDA. Compared with existing EAs, the experimental results demonstrate that EMSM-EDA is a global optimization algorithm with excellent performance.

The remainder of this paper is arranged as follows. Section 2 reviews the related work on basic GEDA and a series of the sampling methods. Section 3 presents the research motivation and the details of an efficient mixture sampling model. Section 4 gives the theoretical analysis of the efficient mixture sampling model. The experimental results are presented in Section 5. Finally, Section 6 summarizes this paper and prospects the potential future research directions.

2. Background and Related Work

2.1. Basic GEDA

Basic GEDA [27] utilizes the probability distribution to generate the population. The details of basic GEDA are shown in **Algorithm 1**.

Algorithm 1: Basic GEDA

```
Input: initial population and relevant parameters.
Output: optimal solution.
While (termination conditions are not satisfied) do
    Evaluate the quality of each solution in the current population;
    Select the excellent sample set by the selection rule;
    Estimate the Gaussian model based on the excellent sample set;
    Sample according to the Gaussian model, and build a new population;
    Update the obtained optimal solution;
End While
```

GEDA adopts Gaussian probability model (GPM) to control the distribution of better solutions. For an n-dimensional random vector $\mathbf{x}$, the joint probability density function of GPM can be expressed as:

$$G_{(\mu,C)}(\mathbf{x}) = \frac{(2\pi)^{-n/2}}{(\det C)^{1/2}} \exp\left(-(\mathbf{x}-\mu)^T (C)^{-1} (\mathbf{x}-\mu)/2\right) \tag{1}$$

where μ is the mean and C is the covariance matrix. In each generation, GEDA usually selects excellent sample set S^t based on the truncation selection rule, and estimates the new μ and C with maximum likelihood estimation:

$$\overline{\mu}^{t+1} = \frac{1}{|S^t|}\sum_{i=1}^{|S^t|} S_i^t \tag{2}$$

$$\overline{C}^{t+1} = \frac{1}{|S^t|}\sum_{i=1}^{|S^t|} \left(S_i^t - \overline{\mu}^{t+1}\right)\left(S_i^t - \overline{\mu}^{t+1}\right)^T \tag{3}$$

where $|\cdot|$ denotes the cardinality of a set.

Fig. 1 is a schematic that describes the search process of basic GEDA. This search process includes populations of two successive generations, and the Gaussian model of population can be defined by the probability density ellipsoid (PDE). As shown in Fig. 1, the blue ellipse and the green ellipse represent the PDEs of the t-th generation and $(t+1)$-th generation, respectively. Specifically, the center and shape of the PDE are determined by $\overline{\mu}$ and $\overline{C}$, respectively.

2.2. Uniform design sampling and orthogonal method

The uniform design sampling (UDS) [28] is a method to generate uniform sampling points in the sampling space, which has better sample representativeness than the random sampling. According to the uniformity of UDS, Zhou et al. [29] redesigned the sampling point operation in EAs. In order to find suitable parameters, Huang et al. [30] applied UDS method to support vector machines. In addition, the idea of USD can also be found in a novel artificial bee colony algorithm [31] and orthogonal learning particle swarm optimization [32]. The details of UDS are presented in **Algorithm 2.**

Algorithm 2: Uniform Design Sampling

```
Input: nonnegative integers $n, t$.
Output: uniform vectors $\{\mathbf{x}_1, \dots, \mathbf{x}_t\}$.
Select $n$ nonnegative integers $h_1, h_2, \dots, h_n$;
$\eta_1, \eta_2, \dots, \eta_n$ are independent identically distributed samples of multinomial distribution;
$x_{kj}$ is defined as $x_{kj} = \left\{\frac{kh_j+\eta_j-\frac{1}{2}}{n}\right\} + \frac{u_{kj}}{n}$, where $k = 1, \dots, t, j = 1, \dots, n, u_{kj}$ and $\eta_i$ are sampled uniformly in $\left[-\frac{1}{2}, \frac{1}{2}\right]$;
The $k$-th uniform vector is $\mathbf{x}_k = (x_{k1}, \dots, x_{kn})$;
The uniform vectors are $\{\mathbf{x}_1, \dots, \mathbf{x}_t\}$.
```

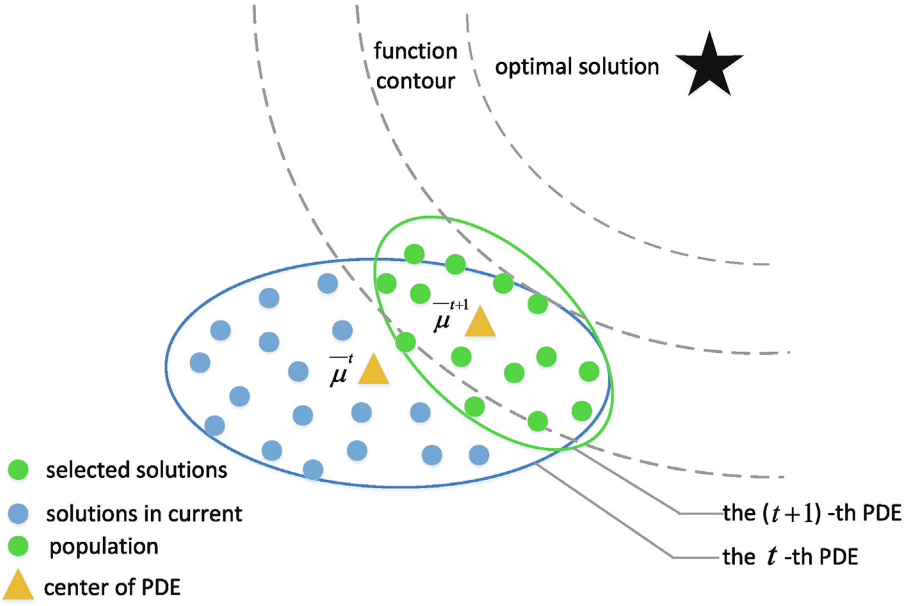


Fig. 1. Schematic to explain the search process of basic GEDA.

The orthogonal method can be found in adaptive coordinate descent [33] and enhance the diversity of the mutation vectors. In general, the algorithm generates a set of random vectors $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ by the random sampling. Then, the Gram-Schmidt orthogonal method is used to orthogonalize the randomly vectors. Finally, a set of orthogonal vectors $\{\overline{\mathbf{x}}_1, \overline{\mathbf{x}}_2, \ldots, \overline{\mathbf{x}}_n\}$ are generated, which can obtain high quality sample.

2.3. Mirrored sampling

Brockhoff et al. [34] first proposed the mirrored sampling method that generates two sample points. The advantage of the mirrored sampling is that there is always a good one of the two mirrored directions. Therefore, the mirrored sampling can accelerate the convergence of the algorithm and has been widely used. Auger et al. [35] applied the mirrored sampling to evolution strategy with weighted multi-recombination. Wang et al. [36] presented a mirrored orthogonal sampling method and applied it to covariance matrix adaptation evolution strategy. Auger et al. [37] made a refined single parent evolution strategy by the mirrored sampling. The details of the mirrored sampling are given in **Algorithm 3**.

```
Algorithm 3: Mirrored Sampling
Input: distribution mean $\mu$, mutation strength $\sigma$.
Output: mirrored vectors $\mathbf{x}_{2i-1}$ and $\mathbf{x}_{2i}$.
1: Generate a random vector $\mathbf{z}_{2i-1}$ by the random sampling method;
2: Generate two mirrored vectors $\mathbf{x}_{2i-1}$ and $\mathbf{x}_{2i}$ by $\mu, \sigma$, and $\mathbf{z}_{2i-1}$; where $\mathbf{x}_{2i-1} = \mu + \sigma\mathbf{z}_{2i-1}, \mathbf{x}_{2i} = \mu - \sigma\mathbf{z}_{2i-1}$.
```

3. Proposed Approach

This section gives the research motivation by analyzing the biased mutation samples and the unsuccessful mutation samples, and proposes EMSM method in detail. Moreover, EMSM-EDA is developed, which combines EMSM with EDA^2.

3.1. Motivation

The mutation vectors are important factors affecting the search of GEDA, which are generated by a simple random sampling from the multivariate Gaussian distribution. In the Gaussian distribution $\mathbf{x}_i \sim \mathcal{N}\left(\overline{\mu}, \overline{C}\right), \overline{\mu}$ is the mean of the current population and $\overline{C}$ is covariance matrix. However, the random sampling has two disadvantages. Firstly, the vectors generated by the random sampling are not uniform and cannot effectively explore the local function landscape. Secondly, most of the unsuccessful mutation vectors do not contribute to the search of GEDA.

Fig. 2 shows an example of biased mutation samples in which six mutation vectors are generated from the random sampling. The black solid star denotes the optimal solution of the function. The grey dotted line denotes the function contour.

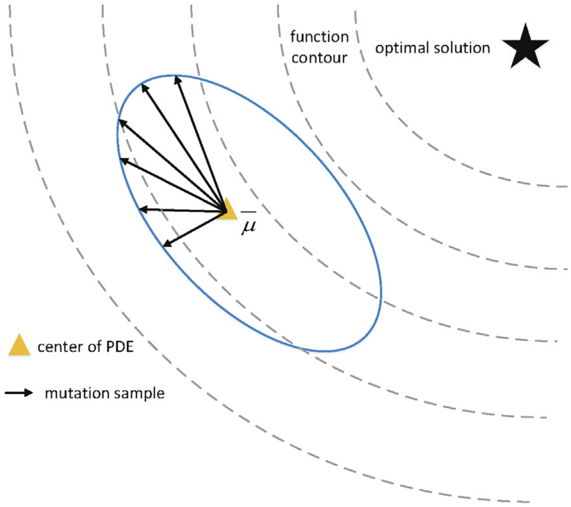


Fig. 2. Example of a set of biased mutation samples.

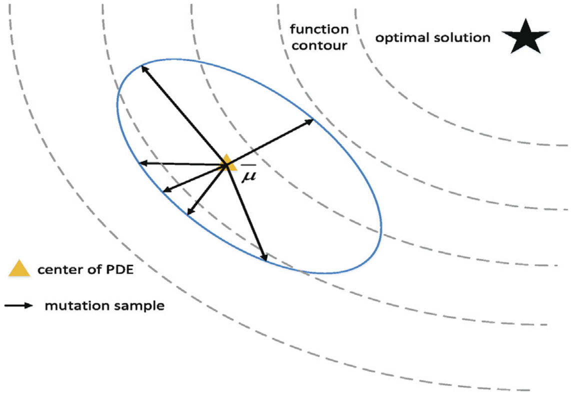


Fig. 3. Example of a set of unsuccessful mutation samples.

The blue ellipsoid denotes the sample space. It is clear that six mutation vectors are concentrated in a small region. Therefore, this sampling method has poor diversity. It is hard to find the promising descent directions, which results in the premature convergence.

As illustrated by Fig. 3, a set of unsuccessful mutation samples are given. Although the distribution of the mutation vectors are relatively uniform, most of the mutation vectors are unsuccessful, which makes the population move toward the opposite direction to the optimal solution and affects the convergence of the algorithm. Therefore, how to utilize the unsuccessful mutation vectors for improving the convergence is worth studying.

3.2. The proposed EMSM method

As discussed in the above subsection, the random sampling of GEDA has the poor diversity and suffers from the premature convergence. To solve these problems, this paper achieves a good tradeoff between the diversity and the convergence by the uniform method, the orthogonal method, and the mirrored sampling.

Algorithm4: A Simple Uniform Method

```
Input: vector dimension $n$, population size $p$, distance threshold $\overline{d}$.
Output: uniform vectors $\{\mathbf{x}_1, \ldots, \mathbf{x}_p\}$.
1: While (number of the mutation vectors is no more than $p$) do
2:     Generate one vector $\mathbf{x}_1$ by using the random sampling;
3: The normalized mutation vector $\widehat{\mathbf{x}}_1$ is generated by $\mathbf{x}_1$;
4:     For $i = 2$ to $p$
5:         Generate one vector $\mathbf{x}_i$ by using the random sampling;
6:         The normalized mutation vector $\widehat{\mathbf{x}}_i$ is generated by $\mathbf{x}_i$;
7:         For $j = 1$ to $i - 1$
8:             Calculate the distance $d(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_j)$ of the vector $\widehat{\mathbf{x}}_i$ and the vector $\widehat{\mathbf{x}}_j$ according to the Eq. (5);
9:         End For
10:         Calculate the diversity index $\mathrm{DI}(\mathbf{x}_i)$ of $\mathbf{x}_i$ according to the Eq. (4);
11:         If $\mathrm{DI}(\mathbf{x}_i) > \overline{d}$ then
12:             Retain $\mathbf{x}_i$;
13:         Else
14:             Delete $\mathbf{x}_i$ and go to step 4;
15:         End If
16:     End For
17: End While
```

A simple uniform method is proposed for maintaining the diversity of the mutation vectors. Firstly, the diversity index is defined by distance metric. Then, the diversity index of each mutation vector is calculated and compared to the distance threshold. Finally, a set of uniform mutation vectors are generated. The detailed process of the uniform method is given in **Algorithm 4**.

The definition process of the diversity index is as follows: Firstly, the normalized mutation vectors $\{\widehat{\mathbf{x}}_1, \widehat{\mathbf{x}}_2, \ldots, \widehat{\mathbf{x}}_p\}$ are obtained from the mutation vectors $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p\}$. Secondly, the diversity index $\mathrm{DI}(\mathbf{x}_i)$ of $\mathbf{x}_i$ is calculated as follows:

$$\mathrm{DI}(\mathbf{x}_i) = \min\{d(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_1), \ldots, d(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_j), \ldots, d(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_p)\} \tag{4}$$

where d is the distance metric and $\widehat{\mathbf{x}}_i \neq \widehat{\mathbf{x}}_j$. The distance metric $d(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_j)$ of $\widehat{\mathbf{x}}_i$ and $\widehat{\mathbf{x}}_j$ is computed as

$$d(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_j) = \sum_{t=1}^{n} |\widehat{x}_{it} - \widehat{x}_{jt}| \tag{5}$$

where $\widehat{x}_{it}$ and $\widehat{x}_{jt}$ are the t-th components of $\widehat{\mathbf{x}}_i$ and $\widehat{\mathbf{x}}_j$, respectively. Finally, the uniform mutation vectors are generated by comparing the diversity index $\mathrm{DI}(\mathbf{x}_i)$ of $\mathbf{x}_i$ with the distance threshold $\overline{d}$. The setting of the distance threshold $\overline{d}$ will be discussed in Section 5.2.

Fig. 4 is an example that explains the advantage of the uniform method. The red circle represents the origin point, the blue circle represents the mutation point, and the arrow represents the mutation vector. As shown in Fig. 4, the mutation vectors are uniformly distributed in the sampling space and have the rich diversity.

However, the uniform method may lose some directions. Thus, the orthogonal method is used to explore more promising directions, which can further enhance the diversity. In addition, the mirrored sampling is used to extract the useful information of the unsuccessful mutation vectors and accelerate the convergence of the algorithm.

Based on the above discussion, an efficient mixture sampling model is proposed in **Algorithm 5**. Firstly, the algorithm generates a set of uniform vectors by the uniform method. Secondly, the uniform mutation vectors are orthogonalized by the Gram-Schmidt orthogonal method. Thirdly, the mirrored vectors are generated by the mirrored sampling. Therefore, the algorithm can maintain a good tradeoff between the diversity and the convergence. Compared with the random sampling, EMSM has the following three advantages.

```
Algorithm 5: An Efficient Mixture Sampling Model
Input: vector dimension $n$, distribution mean $\mu$, covariance matrix $C$, populationsize $p$.
Output: uniform orthogonal vectors.
1: Utilize Algorithm 4 to generate a set of simple uniform vectors $\{\mathbf{x}_1, \ldots, \mathbf{x}_p\}$;
2: If $n \geqslant p$ then
3:     Get the uniform orthogonal vectors $\{\overline{\mathbf{x}}_1, \ldots, \overline{\mathbf{x}}_p\}$ by orthogonalizing the uniform vectors $\{\mathbf{x}_1, \ldots, \mathbf{x}_p\}$ based on the Gram-Schmidt orthogonal method;
4: Else
5:     Get the uniform orthogonal vectors $\{\overline{\mathbf{x}}_1, \ldots, \overline{\mathbf{x}}_n\}$ by orthogonalizing the uniform vectors $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ based on the Gram-Schmidt orthogonal method; Then, merge sets $\{\overline{\mathbf{x}}_1, \ldots, \overline{\mathbf{x}}_n\}$and $\{\mathbf{x}_{n+1}, \ldots, \mathbf{x}_p\}$;
6: End If
7: Select half of the uniform orthogonal vectors to generate the mirrored vectors by Algorithm 3.
```

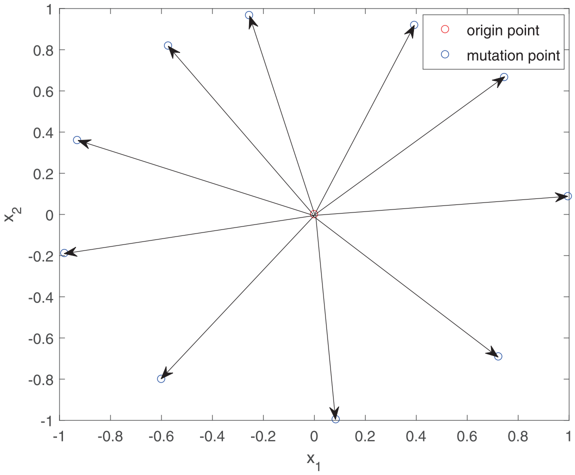


Fig. 4. Example to explain the advantage of a simple uniform method to the rich diversity of the mutation vectors.

1. EMSM adopts the uniform method and the orthogonal method to generate the uniform mutation vectors. Compared with the random sampling, the mutation vectors generated by EMSM have the rich diversity. In the black-box optimization problems, there are a lot of local optimal solutions. With the rich diversity, EMSM can skip the local optimal solutions more easily. Finally, EMSM can explore more promising regions and avoid the premature convergence.
2. EMSM can utilize the unsuccessful mutation vectors to generate the successful mutation vectors by the mirrored sampling, which accelerates the convergence of the algorithm. In addition, the successful mutation vectors produce some unsuccessful mutation vectors by the mirrored sampling. In GEDA, only part of the successful mutation vectors are selected to construct the Gaussian distribution model. Therefore, these unsuccessful mutation vectors are discarded and do not affect the convergence.
3. By introducing the distance threshold $\overline{d}$, EMSM can flexibly control the uniform degree of the mutation vectors. In the low-dimensional space, a small distance threshold $\overline{d}$ can be adopted. Moreover, a large distance threshold $\overline{d}$ can be used for the high-dimensional space.

3.3. Applying EMSM to EDA^2

EMSM is an excellent sampling method and can be easily applied to various GEDA variants. In this part, EMSM-EDA is presented, which combines EMSM with EDA^2. The details of EMSM-EDA are shown in **Algorithm 6**. Firstly, the algorithm initializes the relevant parameters. Secondly, EMSM is utilized to generate the mutation vectors in steps 1 and 13. Finally, EMSM-EDA adopts the archiving mechanism of EDA^2 to construct the mean and the covariance matrix to guide the search.

Algorithm 6: EMSM-EDA

```
Input: population size $p$, selection ratio $\tau$, archive length $l$.
Output: optimal solution $b^t$.
1: Generate the initial population $P^t$ by Algorithm 5;
2: Set $A^t = \varnothing, t = 0$, and $j = 0$;
3: While (termination conditions are not satisfied) do
4:     Evaluate all solutions in population $P^t$ and obtain the best solution $b^t$;
5:     Generate elite population $S^t$ by selecting the best $\lfloor \tau p \rfloor$solutions from $P^t$;
6:     Compute the mean $\overline{\mu}^{t+1}$ with $S^t$ according to the Eq. (2);
7:     Compute the covariance matrix $\overline{C}^{t+1}$ with $S^t$ and $A^t$ based on the Eq. (3);
8:     If $j < l$ then
9:         $A^{t+1} = A^t \cup S^t$;
10:    Else
11:        $A^{t+1} = A^t \cup S^t \setminus S^{t-l}$;
12:    End If
13:    Generate $p - 1$ solutions $M^{t+1}$ by Algorithm 5, $\overline{\mu}^{t+1}$, and $\overline{C}^{t+1}$;
14:    Set $P^{t+1} = M^{t+1} \cup b^t$;
15: End While
```

3.4. Complexity Analysis

The computational cost of EMSM is mainly involved in: 1) the uniform method; 2) the orthogonal method; and 3) the mirrored sampling. Firstly, the computational complexity of uniform method is $O(n^2)$, where n is the dimension of the mutation vector. Secondly, the computational cost of orthogonal method is spent in calling the Gram-Schmidt orthogonal process, which is $O\left(nk^2\right), k = \min\{p, n\}$, where p is population size. According to the parameter setting in Section 5.2, $p > n$. Thus, $O\left(nk^2\right) = O(n^3)$. Thirdly, the computational complexity of mirrored sampling is $O(n)$. Overall, the computational complexity of EMSM is $O(n^3)$. In [26], the computational complexity of EDA^2 is $O(n^3)$. Since EMSM-EDA is proposed by combining EMSM with EDA^2, its computational complexity is $O(n^3)$. Therefore, EMSM-EDA can effectively improve the performance of EDA^2 without increasing the computational complexity.

4. Theoretical Analysis

In this section, this paper firstly analyzes the diversity of the uniform method and the orthogonal method, and then studies the convergence and the diversity of the mirrored sampling.

4.1. Uniform and orthogonal method

This part mainly compares the diversity of the uniform method and the orthogonal method with the random sampling. The diversity of two methods is described by the volume of PDE. As shown in Fig. 5, it is a 3-dimensional example that the volume of the cube represents the volume of PDE in the random sampling and EMSM. For EMSM and the random sampling, the volumes of their PDEs are V_{EMSM} and V_{RS}, respectively. The volume of PDE for the multivariate Gaussian distribution is V_{Gauss}. Compared with V_{Gauss}, the volume errors of two sampling methods are E_{EMSM} and E_{RS}, respectively, where $E_{\text{EMSM}} = V_{\text{Gauss}} - V_{\text{EMSM}}$ and $E_{\text{RS}} = V_{\text{Gauss}} - V_{\text{RS}}$. Thus, if a sampling method has smaller volume error, it has the richer diversity.

Property 1. *The volume error of EMSM is not greater than the random sampling.*

Analysis. Let the axis length of PDE of multivariate Gaussian distribution be composed of a set of n orthogonal vectors $d_1, d_2, \dots, d_n$. Therefore, the volume of PDE is

$$V_{\text{Gauss}} = \prod_{i=1}^{n} |d_i|$$

In GEDA, population size p can be larger than or smaller than dimension n. The following two cases are discussed.

Case 1, $n \geqslant p$.

The volume of this PDE in EMSM is

$$V_{\text{EMSM}} = \prod_{i=1}^{p} |d_i|$$

The volume of this PDE in the random sampling is

$$V_{\text{RS}} = \prod_{i=1}^{p} |\mathbf{x}_i| \cdot \prod_{i=1}^{p} |\cos\theta_i|$$

where $\cos\theta_i$ is the cosine of the smallest angle between the mutation vector $\mathbf{x}_i$ and other mutation vectors.

The volume errors of two sampling methods are

$$E_{\text{EMSM}} = V_{\text{Gauss}} - V_{\text{EMSM}}$$

$$E_{\text{RS}} = V_{\text{Gauss}} - V_{\text{RS}}$$

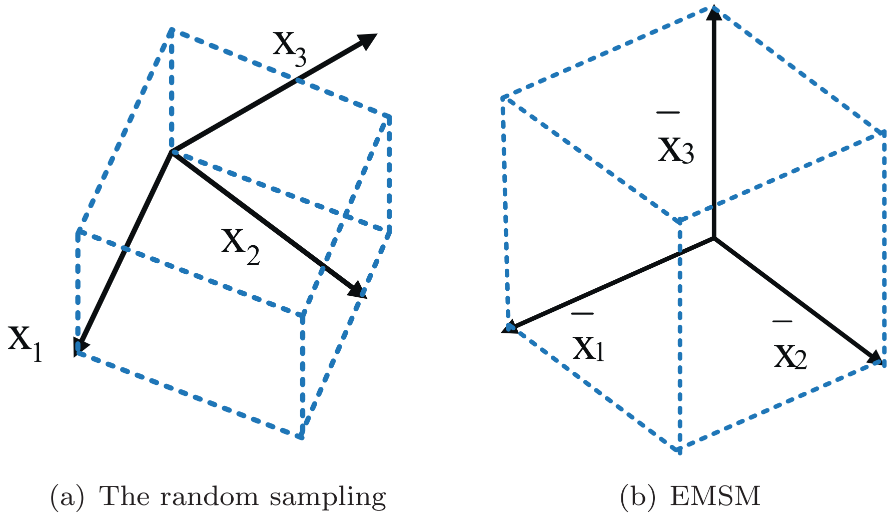
Obviously, $\prod_{i=1}^{p} |d_i| \geqslant \prod_{i=1}^{p} |\mathbf{x}_i|, 0 \leqslant \prod_{i=1}^{p} |\cos\theta_i| \leqslant 1$. Then, $V_{\text{EMSM}} \geqslant V_{\text{RS}}$. Therefore, $E_{\text{EMSM}} \leqslant E_{\text{RS}}$.

Case 2, $n < p$.

The volume of this PDE in EMSM is

$$V_{\text{EMSM}} = \prod_{i=1}^{n} |d_i|$$

The volume of this PDE in the random sampling is



(a) The random sampling (b) EMSM

Fig. 5. Example to explain the volumes of PDEs in the random sampling and EMSM.

$$V_{\mathrm{RS}} = \prod_{i=1}^{n} |\mathbf{x}_i| \cdot \prod_{i=1}^{n} |\cos\theta_i|$$

From the conclusion of case 1, it is easy to obtain $\prod_{i=1}^{n}|d_i| \geqslant \prod_{i=1}^{n}|\mathbf{x}_i| \cdot \prod_{i=1}^{n}|\cos\theta_i|$. Thus, $V_{\mathrm{EMSM}} \geqslant V_{\mathrm{RS}}$ and $E_{\mathrm{EMSM}} \leqslant E_{\mathrm{RS}}$.

By the above analysis, the final conclusion is that the volume error of EMSM is not greater than the random sampling. Therefore, EMSM has the richer diversity.

4.2. Mirrored sampling method

This part firstly utilizes the progress rate to research the convergence of the mirrored sampling and the random sampling. Specifically, the progress rate is the distance that the overall probability distribution moves in the descent direction. Then, the diversity of the mirrored sampling is analyzed by the volume of PDE.

Property 2. *The mirrored sampling has the greater progress rate than the random sampling.*

Analysis. The progress rate is defined based on the method in references [38–40]. As shown in Fig. 6, PDE is a sphere. **C** is the center of PDE, and **O** is the optimal solution to the sphere function. The distance between **C** and **O** is $\|\mathbf{CO}\| = L$. $\mathbf{x}$ is the mutation sample. $-\mathbf{x}$ is the mirrored sample. The vector z is the projection of the mutation sample $\mathbf{x}$ onto the line **CO**.

The progress rate can be calculated by the projection of the vector z onto **CO**. Let $\hat{z}$ be the largest projection of the mutation vectors $\{\mathbf{x}_1, \ldots, \mathbf{x}_p\}$ onto **CO**. Based on reference [39], the progress rate ϕ can be expressed as

$$\phi = L \cdot \mathbb{E}\left\{1 - \left(1 + \frac{n}{2L^2} - \frac{\hat{z}}{L}\right)\right\}$$

where $\mathbb{E}$ is the expectation. The cumulative probability distribution of the largest order statistic for EMSM can be computed as

$$\begin{aligned} P_{\mathrm{EMSM}} &= (Z \leqslant z) = [Pr(-z < Z \leqslant z)]^{p/2} \\ &= [\Phi(z) - \Phi(-z)]^{p/2} \\ &= [2\Phi(z) - 1]^{p/2} \end{aligned}$$

where $\Phi(\cdot)$ stands for a standard Gaussian random variable. Let define $\forall z \geqslant 0$. The probability density function of EMSM is

$$f_{\mathrm{EMSM}} = pf(z)[2\Phi(z) - 1]^{p/2-1}$$

where $f(\cdot)$ is the probability density function of a standard Gaussian distribution. Based on reference [38], the probability density function of the random sampling is

$$f_{\mathrm{RS}} = pf(z)\Phi(z)^{p-1}$$

From the conclusions of references [38,39], the progress coefficients of EMSM and the random sampling are

$$c_{\mathrm{EMSM}} = p\int_{-\infty}^{\infty} zf(z)[2\Phi(z) - 1]^{p/2-1}dz$$

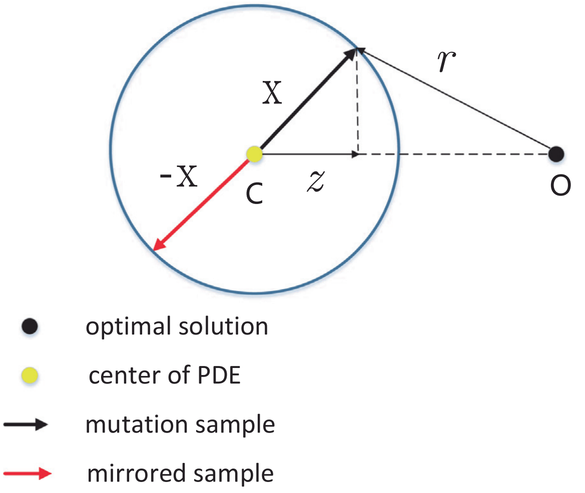


Fig. 6. The progress rate on the sphere function.

$$c_{\mathrm{RS}} = p\int_{-\infty}^{\infty} zf(z)\Phi(z)^{p-1}dz$$

Moreover, the normalized progress rates of EMSM and the random sampling are

$$\phi^*_{\mathrm{EMSM}} = \sigma^* c_{\mathrm{EMSM}} - \frac{(\sigma^*)^2}{2}$$

$$\phi^*_{\mathrm{RS}} = \sigma^* c_{\mathrm{RS}} - \frac{(\sigma^*)^2}{2}$$

where $\phi^* = \phi\frac{n}{L}, \sigma^* = \frac{n}{L}$. By doing a simple calculation, it is easy to get $c_{\mathrm{EMSM}} \geqslant c_{\mathrm{RS}}$ and $\phi^*_{\mathrm{EMSM}} \geqslant \phi^*_{\mathrm{RS}}$. Compared with the random sampling, the mirrored sampling has the greater progress rate and the faster convergence.

Property 3. *The greater progress rate of the mirrored sampling can obtain the larger volume of PDE.*

Analysis. The volume of PDE in the mirrored sampling is studied in reference [41]. For different selection methods of solutions, the algorithm can get different progress rates. Therefore, the means $\widehat{\mu}_{\mathrm{EMSM}}$ and $\overline{\mu}_{\mathrm{EMSM}}$ can be obtained for two different progress rates, where $\widehat{\mu}_{\mathrm{EMSM}} = \overline{\mu}_{\mathrm{EMSM}} + \delta$ and $\delta \geqslant 0$. The corresponding covariance matrices are $\widehat{C}_{\mathrm{EMSM}}$ and $\overline{C}_{\mathrm{EMSM}}$, respectively. The eigenvalues of the covariance matrices $\widehat{C}_{\mathrm{EMSM}}$ and $\overline{C}_{\mathrm{EMSM}}$ are $\left\{\widehat{\lambda}_1, \ldots, \widehat{\lambda}_n\right\}$ and $\{\overline{\lambda}_1, \ldots, \overline{\lambda}_n\}$, respectively. For the two progress rates, the volumes of PDEs are $\prod_{i=1}^n \widehat{\lambda}_i$ and $\prod_{i=1}^n \overline{\lambda}_i$, respectively. The covariance matrix $\widehat{C}_{\mathrm{EMSM}}$ can be computed as follows:

$$\begin{aligned}
&\widehat{C}_{\mathrm{EMSM}} = \tfrac{1}{|S^t|}\sum_{i=1}^{|S^t|}\left(S_i^t - \widehat{\mu}_{\mathrm{EMSM}}\right)\left(S_i^t - \widehat{\mu}_{\mathrm{EMSM}}\right)^T \\
&= \tfrac{1}{|S^t|}\sum_{i=1}^{|S^t|}\left(S_i^t - \overline{\mu}_{\mathrm{EMSM}} - \delta\right)\left(S_i^t - \overline{\mu}_{\mathrm{EMSM}} - \delta\right)^T \\
&= \tfrac{1}{|S^t|}\sum_{i=1}^{|S^t|}\left(S_i^t - \overline{\mu}_{\mathrm{EMSM}}\right)\left(S_i^t - \overline{\mu}_{\mathrm{EMSM}}\right)^T + \delta\delta^T \\
&- \tfrac{1}{|S^t|}\sum_{i=1}^{|S^t|}\left(S_i^t - \overline{\mu}_{\mathrm{EMSM}}\right)^T\delta - \tfrac{1}{|S^t|}\sum_{i=1}^{|S^t|}\left(S_i^t - \overline{\mu}_{\mathrm{EMSM}}\right)\delta^T \\
&= \overline{C}_{\mathrm{EMSM}} + \delta\delta^T
\end{aligned}$$

$\widehat{C}_{\mathrm{EMSM}}$ and $\overline{C}_{\mathrm{EMSM}}$ are semi-positive definite matrices. Then, the matrix $\widehat{C}_{\mathrm{EMSM}}$ is decomposed into

$$\overline{C}_{\mathrm{EMSM}} = Q\Lambda Q^T$$

where Q is an orthogonal matrix and Λ is a diagonal matrix. The orthogonal transformation of the matrix $\widehat{C}_{\mathrm{EMSM}}$ is obtained

$$\begin{aligned}
&Q\widehat{C}_{\mathrm{EMSM}}Q^T = Q\overline{C}_{\mathrm{EMSM}}Q^T + (Q\delta)(Q\delta)^T \\
&= \Lambda + (Q\delta)(Q\delta)^T
\end{aligned}$$

Obviously, $\Lambda + (Q\delta)(Q\delta)^T$ is the rank-one modification of Λ. According to the conclusion in reference [42], the characteristic polynomial of $\widehat{C}_{\mathrm{EMSM}}$ is

$$F(\lambda) = \prod_{i=1}^n(\lambda - \lambda_i) - \sum_{k=1}^n\left((Q\delta)^2 \cdot \prod_{i=1, i\neq k}^n(\lambda - \lambda_i)\right)$$

Let $\lambda = 0, F(\lambda)$ is transformed into

$$F(\lambda) = \prod_{i=1}^n \lambda_i + \sum_{k=1}^n (Q\delta)^2 \cdot \prod_{i=1, i\neq k}^n \lambda_i$$

Based on the definition of the characteristic polynomial, the characteristic polynomial of $\widehat{C}_{\mathrm{EMSM}}$ is

$$F(\lambda) = \prod_{i=1}^n \widehat{\lambda}_i$$

From the above two formulas, the following relation can be obtained

$$\prod_{i=1}^{n}\widehat{\lambda}_i = \prod_{i=1}^{n}\overline{\lambda}_i + \sum_{k=1}^{n}(Q\delta)^2 \cdot \prod_{i=1,i\neq k}^{n}\overline{\lambda}_i$$

where $\sum_{k=1}^{n}(Q\delta)^2 \cdot \prod_{i=1,i\neq k}^{n}\overline{\lambda}_i \geqslant 0$. Then, $\prod_{i=1}^{n}\widehat{\lambda}_i \geqslant \prod_{i=1}^{n}\overline{\lambda}_i$. Therefore, the greater progress rate of the mirrored sampling can obtain the larger volume of PDE and the richer diversity.

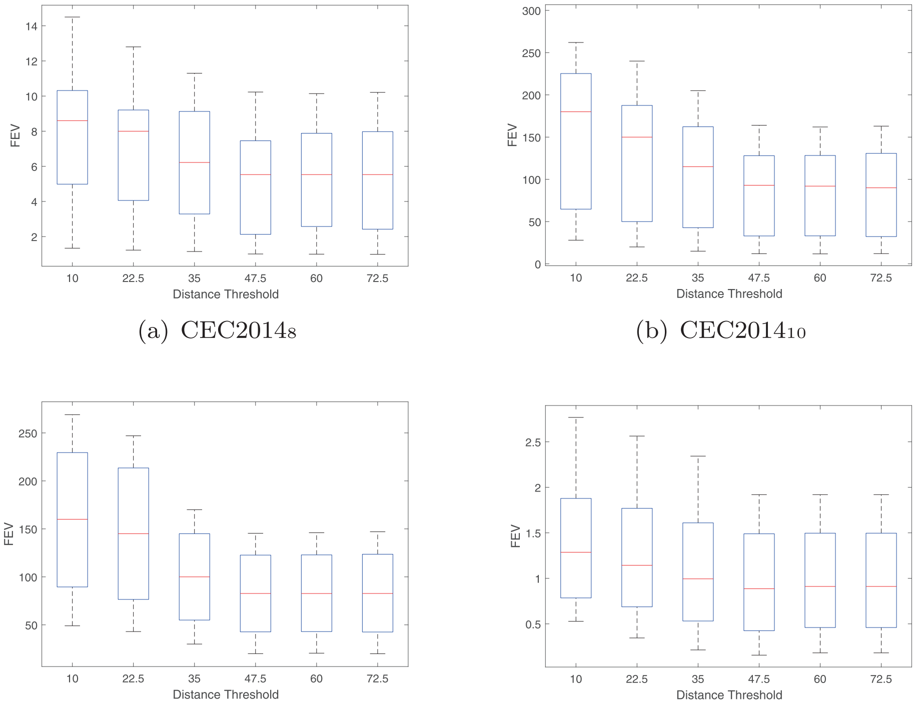
5. Experimental Studies

5.1. Experimental settings

The experiments are carried out on 58 test functions with 30 dimensions (30D), 50 dimensions (50D), and 100 dimensions (100D) of IEEE CEC2013 [43] and IEEE CEC2014 [44] test suites, respectively. For the convenience of description, we denote 28 functions for IEEE CEC2013 as CEC2013$_1$-CEC2013$_{28}$ and 30 functions for IEEE CEC2014 as CEC2014$_1$-CEC2014$_{30}$. According to their characteristics, 58 test functions can be categorized in Table 1. Specifically, the unimodal functions have only global optimum and their structure is smooth. The simple multimodal functions have a large number of local optimum, which can reflect the characteristics of real-world problems as well. For the hybrid functions, the variables are

Table 1
Test functions.

IEEE CEC2013	
CEC2013$_1$-CEC2013$_5$	unimodal functions
CEC2013$_6$-CEC2013$_{20}$	simple multimodal functions
CEC2013$_{21}$-CEC2013$_{28}$	composition functions
IEEE CEC2014	
CEC2014$_1$-CEC2014$_3$	unimodal functions
CEC2014$_4$-CEC2014$_{16}$	simple multimodal functions
CEC2014$_{17}$-CEC2014$_{22}$	hybrid functions
CEC2014$_{23}$-CEC2014$_{30}$	composition functions



(a) CEC2014$_8$

(b) CEC2014$_{10}$

(c) CEC2014$_{11}$

(d) CEC2014$_{18}$

Fig. 7. Box plots of the FEVs provided by EMSM-EDA with different distance thresholds on 4 functions of CEC2014 benchmark functions with 50D.

Table 2
The means and the standard deviations of FEVs obtained by the two algorithms over 25 independent runs on 28 CEC2013 benchmark functions with 30D, 50D, and 100D.

Functions	EMSM-EDA(30D)	EDA^2(30D)	EMSM-EDA(50D)	EDA^2(50D)	EMSM-EDA(100D)	EDA^2(100D)
CEC2013$_{1}$	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≈
CEC2013$_{2}$	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	1.55E+04±1.44E+04-
CEC2013$_{3}$	0.00E+00±0.00E+00	0.00E+00±0.00E+00≈	5.00E-04±2.23E-04	1.07E-06±3.22E-07+	1.68E+02±1.79E+02	4.41E+07±4.81E+07-
CEC2013$_{4}$	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≈
CEC2013$_{5}$	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	3.67E-04±4.25E-04-
CEC2013$_{6}$	1.06E+00±5.28E+00	1.06E+00±5.28E+00≈	4.34E+01±0.00E+00	4.34E+01±0.00E+00≈	2.10E+02±4.51E+01	2.15E+02±3.00E+01≈
CEC2013$_{7}$	5.93E-06±1.04E-05	1.81E-05±3.26E-05-	2.01E-04±3.57E-05	3.10E-05±1.06E-05+	4.50E-03±1.67E-03	4.68E+00±1.90E+00-
CEC2013$_{8}$	2.09E+01±3.99E-02	2.09E+01±5.49E-02≈	2.11E+01±4.12E-02	2.11E+01±4.30E-02≈	2.12E+01±5.19E-02	2.12E+01±4.39E-02≈
CEC2013$_{9}$	3.44E+00±1.17E+00	4.19E+00±1.75E+00-	4.40E+00±2.57E+00	5.78E+00±3.52E+00-	1.10E+01±2.69E+00	2.49E+01±2.73E+00-
CEC2013$_{10}$	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≈
CEC2013$_{11}$	3.94E+00±1.94E+00	5.21E+00±2.23E+00-	3.74E+00±1.80E+00	6.09E+00±2.67E+00-	1.69E+01±3.83E+00	7.46E+01±1.20E+01-
CEC2013$_{12}$	2.79E+00±1.75E+00	5.29E+00±2.20E+00-	3.24E+00±1.79E+00	5.89E+00±2.83E+00-	1.62E+01±3.87E+00	8.53E+01±1.92E+01-
CEC2013$_{13}$	2.99E+00±1.66E+00	6.18E+00±6.37E+00-	2.82E+00±1.37E+00	5.17E+00±2.46E+00-	1.59E+01±5.00E+00	1.36E+02±3.22E+01-
CEC2013$_{14}$	5.49E+02±5.36E+02	1.74E+03±1.68E+03-	1.48E+02±1.20E+02	3.96E+03±3.09E+03-	1.56E+03±4.97E+02	4.54E+03±6.89E+02-
CEC2013$_{15}$	3.88E+02±7.24E+02	9.06E+02±1.41E+03-	4.26E+02±1.80E+02	7.51E+02±5.74E+02-	1.95E+03±5.00E+02	4.64E+03±7.15E+02-
CEC2013$_{16}$	2.44E+00±2.82E-01	2.39E+00±2.35E-01+	3.24E+00±3.24E-01	3.22E+00±3.35E-01+	4.00E+00±2.95E-01	3.95E+00±3.04E-01≈
CEC2013$_{17}$	1.69E+02±7.79E+00	1.71E+02±1.13E+01-	3.27E+02±1.69E+01	3.46E+02±1.16E+01-	7.34E+02±4.82E+01	3.98E+02±2.10E+02+
CEC2013$_{18}$	1.73E+02±8.83E+00	1.75E+02±8.00E+00-	3.36E+02±1.52E+01	3.44E+02±1.03E+01-	7.97E+02±1.92E+01	8.27E+02±2.12E+01-
CEC2013$_{19}$	4.09E+00±1.71E+00	4.55E+00±1.14E+00-	9.79E+00±2.47E+00	1.05E+01±2.47E+00-	9.84E+00±9.22E-01	1.15E+01±1.15E+01-
CEC2013$_{20}$	1.16E+01±3.12E-01	1.19E+01±2.57E-01-	2.12E+01±3.40E-01	2.15E+01±2.87E-01-	4.99E+01±6.54E-01	5.00E+01±4.32E-06≈
CEC2013$_{21}$	2.78E+02±5.71E+01	2.93E+02±7.09E+01-	6.80E+02±4.70E+02	5.46E+02±4.39E+02+	3.28E+02±4.58E+01	3.48E+02±5.10E+01-
CEC2013$_{22}$	1.95E+02±7.65E+01	2.26E+02±6.67E+01-	1.97E+02±8.99E+01	3.37E+02±6.32E+02-	1.63E+03±5.13E+02	4.83E+03±7.71E+02-
CEC2013$_{23}$	2.85E+02±1.29E+02	3.20E+02±2.86E+02-	4.98E+02±2.08E+02	6.72E+02±1.81E+02-	2.40E+03±5.12E+02	5.74E+03±9.92E+02-
CEC2013$_{24}$	2.18E+02±1.45E+01	2.18E+02±1.41E+01≈	2.22E+02±2.17E+01	2.24E+02±2.13E+01-	2.50E+02±4.98E+01	2.80E+02±4.51E+01-
CEC2013$_{25}$	2.44E+02±5.55E+00	2.45E+02±7.66E+00-	2.77E+02±7.73E+00	2.81E+02±9.21E+00-	3.72E+02±2.32E+01	4.17E+02±1.91E+01-
CEC2013$_{26}$	2.78E+02±3.69E+01	2.72E+02±4.18E+01+	3.04E+02±1.19E+01	3.04E+02±8.76E+00≈	3.18E+02±1.76E+01	3.53E+02±1.60E+01-
CEC2013$_{27}$	3.57E+02±8.25E+01	3.64E+02±9.00E+01-	4.84E+02±1.47E+02	5.66E+02±1.22E+02-	8.98E+02±1.72E+02	1.22E+03±1.37E+02-
CEC2013$_{28}$	3.00E+02±0.00E+00	3.00E+02±0.00E+00≈	4.00E+02±2.82E-08	7.47E+02±9.58E+02-	4.00E+03±8.97E+02	3.71E+03±1.04E+03+
No.of+/≈/-	**NA**	**2/10/16**	**NA**	**4/8/16**	**NA**	**2/7/19**

“+”, “-”, and “≈” indicate that the performance of EDA^2 algorithm is better than, worse than, and similar to that of EMSM-EDA, respectively.

randomly divided into some subcomponents and then different basic functions are used for different subcomponents. Then, the different subcomponents of the variables may have different properties. With hybrid functions as the basic functions, the composition function can have different properties for different variables subcomponents and keep continuous around the global/local optima. Moreover, these test functions are all minimization problems with a single objective and their detailed description can be found in [43,44].

In the comparison experiments, each algorithm independently runs 25 times, and 10000*D total fitness evaluation is used as the termination condition. All the experiments are implemented on a PC with Intel Core i5-8400 CPU @ 4.00 GHz and 64-bit Windows 10 operating system. The function error value (FEV) is the difference between the best solution of the algorithm and the true optimal solution. When the FEV is less than 1.0E-8, it is taken as 0.

5.2. Influence of parameters

EMSM-EDA has four parameters: archive length l, selection ratio τ, population size p, and distance threshold $\overline{d}$. Specifically, l and τ are set to 20 and 0.35, respectively. p is set to 100, 200, and 200 on test functions with 30D, 50D, and 100D, respectively. In addition, $\overline{d}$ is the parameter that controls the diversity of the mutation vectors. A large $\overline{d}$ can increase the diversity, but it requires a large amount of computation. Moreover, a small $\overline{d}$ has little effect on the diversity. Based on the experimental analysis, when $\overline{d}$ is set to 28.5, 47.5, and 95 with 30D, 50D, and 100D, EMSM-EDA can obtain a good test result and have an acceptable amount of computation.

Fig. 7 shows the results provided by EMSM-EDA with different $\overline{d}$ on CEC2014$_8$, CEC2014$_{10}$, CEC2014$_{11}$, and CEC2014$_{18}$ with 50D. When $\overline{d}$ increases from 10 to 47.5, FEV decreases rapidly. However, as $\overline{d}$ continues to increase, FEV is almost constant. In particular, the algorithm needs to consume a large amount of computation with the increase of $\overline{d}$. Therefore, $\overline{d}$ is set to 47.5. In the case of 30D and 100D, the variations of $\overline{d}$ and FEV are similar to 50D.

5.3. Effectiveness of EMSM

To verify the effectiveness of EMSM, this paper firstly compares EMSM-EDA with EDA^2 on two test suites with 30D, 50D, and 100D, respectively. Then, the evolution curves of EMSM-EDA and EDA^2 on 8 test functions with 100D are compared.

Table 2 summarizes the test results of EMSM-EDA and EDA^2 on IEEE CEC2013 test suite. For the benchmark functions with 30D, 50D, and 100D, EMSM-EDA outperforms EDA^2 on 16, 16, and 19 functions, respectively. In the case of 100D, EMSM-EDA improves EDA^2 on CEC2013$_3$ and CEC2013$_7$ functions by three order of magnitude. Meanwhile, EMSM-EDA also improves EDA^2 on CEC2013$_{13}$ and CEC2013^{27} functions by one order of magnitude. Moreover, EMSM-EDA provides the optimal solutions with 100% successful rate on the functions CEC2013$_2$ and CEC2013$_5$. Furthermore, we conduct Wilcoxon's test [45] on two algorithms based on the total 28 test functions with 30D, 50D, and 100D. Table 3 presents the test results that EMSM-EDA has higher R^+ values than R^- values and corresponding p values are much less than 0.05. The results show that EMSM-EDA is superior to EDA^2 on the total 28 test functions.

The results of EMSM-EDA and EDA^2 are summarized in Table 4 on IEEE CEC2014 test suite with 30D, 50D, and 100D, respectively. It is clear that EMSM-EDA outperforms EDA^2 on 20, 19, and 24 functions, respectively. Specially, for the case of 100D, EMSM-EDA improves EDA^2 on CEC2014$_{20}$ and CEC2014$_{29}$ functions by one order of magnitude. Moreover, EMSM-EDA improves EDA^2 on CEC2014$_{17}$ and CEC2014$_{18}$ functions by two order of magnitude. In addition, EMSM-EDA improves EDA^2 on CEC2014$_1$ and CEC2014$_2$ functions by nine order of magnitude. Furthermore, Wilcoxon's test is conducted on IEEE CEC2014 test functions with 30D, 50D, and 100D. The test results in Table 5 show that EMSM-EDA performs better than EDA^2.

Fig. 8 and Fig. 9 present the evolution curves of FEVs on CEC2013$_2$, CEC2013$_3$, CEC2013$_5$, CEC2013$_7$, CEC2014$_1$, CEC2014$_2$, CEC2014$_{17}$, and CEC 2014$_{18}$ with 100D. For all 8 test functions, EDA^2 suffers from the premature convergence. However, EMSM-EDA can obtain better solutions. In addition, EMSM-EDA finds the optimal solutions on CEC2013$_2$ and CEC2013$_5$.

All the above investigations verify that EMSM is effective, which can accomplish a good tradeoff between the diversity and the convergence, and thus obtain better solutions and faster convergence speed in EMSM-EDA. Moreover, the simple structure of EMSM is easily applied to various GEDA variants.

Table 3
Results obtained by the Wilcoxon's test on CEC2013 Benchmark functions with 30D, 50D, and 100D.

VS	R^+	R^-	p-value
EDA^2(30D)	344.5	61.5	0.001184
EDA^2(50D)	335.0	71.0	0.002551
EDA^2(100D)	353.0	53.0	0.000588

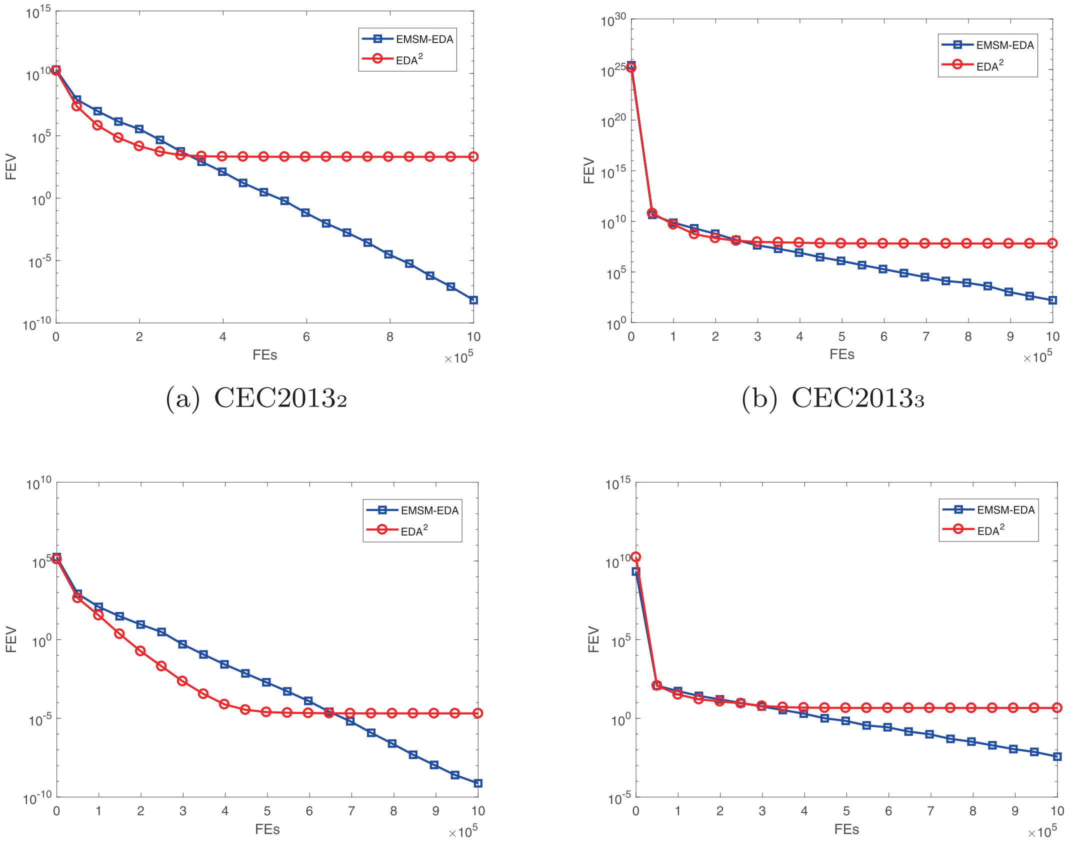
Table 4
The means and the standard deviations of FEVs obtained by the two algorithms over 25 independent runs on 30 CEC2014 benchmark functions with 30D, 50D, and 100D.

Functions	EMSM-EDA(30D)	EDA^2(30D)	EMSM-EDA(50D)	EDA^2(50D)	EMSM-EDA(100D)	EDA^2(100D)
CEC2014$_1$	0.00E+00±0.00E+00	0.00E+00±0.00E+00≈	0.00E+00±0.00E+00	0.00E+00±0.00E+00≈	1.26E-08±1.18E-08	1.51E+04±1.34E+04-
CEC2014$_2$	0.00E+00±0.00E+00	0.00E+00±0.00E+00≈	0.00E+00±0.00E+00	0.00E+00±0.00E+00≈	1.37E-06±1.00E-06	4.14E+03±1.05E+04-
CEC2014$_3$	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≈
CEC2014$_4$	0.00E+00±0.00E+00	0.00E+00±0.00E+00≈	4.77E+01±4.95E+01	4.96E+01±4.75E+01-	1.79E+02±2.95E+01	1.92E+02±2.57E+01-
CEC2014$_5$	2.09E+01±5.11E-02	2.10E+01±3.40E-02-	2.11E+01±4.30E-02	2.11E+01±3.87E-02≈	2.13E+01±2.55E-02	2.13E+01±1.92E-02≈
CEC2014$_6$	1.60E-01±4.09E-01	1.89E-01±4.75E-01-	2.55E-01±4.49E-01	2.63E-01±5.20E-01-	2.47E+00±1.68E+00	4.48E+00±2.51E+00-
CEC2014$_7$	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≈
CEC2014$_8$	4.42E+00±2.28E+00	6.13E+00±2.47E+00-	5.53E+00±1.89E+00	8.04E+00±1.97E+00-	2.37E+01±5.27E+00	5.77E+01±1.03E+01-
CEC2014$_9$	3.98E+00±1.84E+00	6.49E+00±2.26E+00-	5.25E+00±2.25E+00	7.84E+00±2.19E+00-	2.09E+01±4.08E+00	8.19E+01±1.47E+01-
CEC2014$_{10}$	1.88E+01±3.87E+01	2.85E+01±5.60E+01-	9.33E+01±1.48E+02	1.76E+02±1.93E+02-	1.14E+03±5.31E+02	3.35E+03±6.77E+02-
CEC2014$_{11}$	9.42E+01±7.83E+01	1.24E+02±1.01E+02-	8.27E+01±1.12E+02	1.86E+02±1.48E+02-	1.32E+03±4.58E+02	3.73E+03±7.75E+02-
CEC2014$_{12}$	2.45E+00±2.71E-01	2.48E+00±2.41E-01-	3.38E+00±3.59E-01	3.43E+00±1.93E-01≈	4.03E+00±1.90E-01	3.91E+00±2.56E-01≈
CEC2014$_{13}$	3.62E-02±1.01E-02	4.52E-02±1.02E-02-	6.31E-02±9.91E-03	7.22E-02±9.66E-03-	1.27E-01±2.04E-02	1.62E-01±1.98E-02-
CEC2014$_{14}$	3.62E-01±5.66E-02	4.08E-01±3.53E-02-	3.57E-01±5.11E-02	3.96E-01±4.63E-02-	3.76E-01±4.66E-02	3.97E-01±4.19E-02-
CEC2014$_{15}$	4.11E+00±1.37E+00	4.09E+00±1.27E+00+	1.07E+01±2.25E+00	1.12E+01±1.60E+00-	1.00E+01±8.77E-01	1.17E+01±1.07E+00-
CEC2014$_{16}$	1.20E+01±3.30E-01	1.22E+01±1.85E-01-	2.15E+01±2.55E-01	2.19E+01±1.75E-01-	4.57E+01±4.03E-01	4.60E+01±3.10E-01-
CEC2014$_{17}$	5.35E+00±5.19E+00	1.40E+01±8.26E+00-	2.39E+01±1.51E+01	2.92E+01±1.62E+01-	8.44E+01±2.69E+01	1.89E+03±5.60E+02-
CEC2014$_{18}$	5.56E-01±2.64E-01	8.74E-01±6.12E-01-	1.08E+00±7.96E-01	1.44E+00±1.04E+00-	3.65E+00±1.66E+00	1.50E+02±3.75E+01-
CEC2014$_{19}$	3.15E+00±2.98E-01	3.88E+00±6.63E-01-	6.51E+00±5.27E-01	7.38E+00±7.50E-01-	9.17E+01±1.14E+00	9.52E+01±1.59E+00-
CEC2014$_{20}$	1.45E+00±3.48E-01	1.48E+00±2.80E-01-	2.24E+00±3.59E-01	2.64E+00±5.44E-01-	4.32E+00±1.16E+00	4.31E+01±1.31E+01-
CEC2014$_{21}$	1.97E+00±4.06E+00	6.28E+00±2.37E+01-	1.48E+02±7.65E+01	1.24E+02±9.92E+01+	1.43E+02±7.01E+01	4.97E+02±1.98E+02-
CEC2014$_{22}$	4.70E+01±4.46E+01	4.73E+01±4.54E+01-	2.95E+01±6.27E-01	2.98E+01±5.88E-01-	4.83E+01±5.36E+01	7.17E+01±6.14E+01-
CEC2014$_{23}$	3.15E+02±0.00E+00	3.15E+02±0.00E+00≈	3.44E+02±0.00E+00	3.44E+02±0.00E+00≈	3.48E+02±0.00E+00	3.48E+02±6.33E-04≈
CEC2014$_{24}$	2.00E+02±2.91E-08	2.00E+02±0.00E+00≈	2.66E+02±2.49E+00	2.68E+02±1.57E+00-	3.65E+02±7.59E-01	3.78E+02±3.04E+00-
CEC2014$_{25}$	2.03E+02±2.12E-02	2.03E+02±3.11E-02≈	2.04E+02±1.55E-01	2.04E+02±1.16E-01≈	2.14E+02±2.83E-01	2.18E+02±1.42E+00-
CEC2014$_{26}$	1.00E+02±7.73E-03	1.00E+02±1.10E-02≈	1.00E+02±1.24E-02	1.00E+02±1.60E-02≈	2.00E+02±9.51E-07	2.00E+02±1.56E-02≈
CEC2014$_{27}$	3.29E+02±4.22E+01	3.49E+02±4.41E+01-	4.00E+02±5.23E+01	4.09E+02±5.16E+01-	4.37E+02±9.29E+01	6.50E+02±1.60E+02-
CEC2014$_{28}$	8.06E+02±4.32E+01	8.12E+02±4.36E+01-	1.18E+03±1.83E+02	1.21E+03±1.91E+02-	2.50E+03±3.80E+02	3.02E+03±5.24E+02-
CEC2014$_{29}$	6.21E+02±2.22E+02	7.65E+05±2.67E+06-	7.36E+02±1.27E+02	7.08E+02±1.20E+02+	4.75E+06±2.37E+07	1.35E+07±4.81E+07-
CEC2014$_{30}$	9.04E+02±4.30E+02	1.05E+03±4.44E+02-	9.14E+03±6.14E+02	9.37E+03±8.11E+02-	3.23E+03±6.92E+02	4.93E+03±1.27E+03-
No.of+/≈/-	**NA**	**1/9/20**	**NA**	**2/9/19**	**NA**	**0/6/24**

"+", "-", and "≈" indicate that the performance of EDA^2 algorithm is better than, worse than, and similar to that of EMSM-EDA, respectively.

Table 5
Results obtained by the Wilcoxon's test on CEC2014 Benchmark functions with 30D, 50D, and 100D.

VS	R^+	R^-	p-value
EDA^2(30D)	407.0	28.0	0.000040
EDA^2(50D)	396.0	69.0	0.000743
EDA^2(100D)	423.0	12.0	0.000008



(a) CEC2013$_2$

(b) CEC2013$_3$

(c) CEC2013$_5$

(d) CEC2013$_7$

Fig. 8. Evolution of FEVs derived from EMSM-EDA and EDA^2 on 4 functions of CEC2013 benchmark functions with 100D.

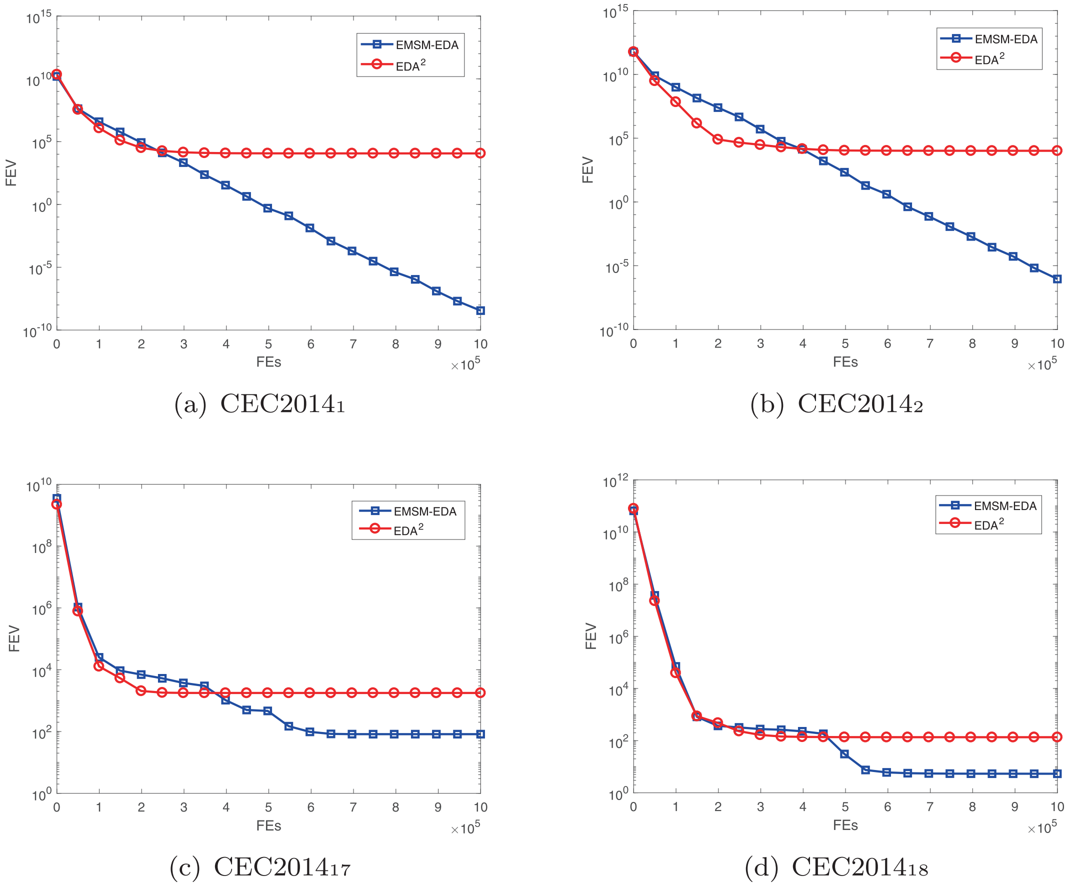
5.4. The effectiveness of EMSM's three components

The aim of this subsection is to identify the effectiveness of three components of EMSM: the simple uniform method, the orthogonal method, and the mirrored sampling. For this reason, the above three components are applied to EDA^2, and these three algorithms are named as U-EDA^2, O-EDA^2, and M-EDA^2, respectively. Moreover, EDA^2 and EMSM-EDA are chosen as a basic comparison.

Table 6 summarizes the test results of EDA^2, U-EDA^2, O-EDA^2, M-EDA^2, and EMSM-EDA on IEEE CEC2014 test suite with 30D. From Table 6, EMSM-EDA and EDA^2 have the best and the worst performance among the five comparison algorithms, respectively. Moreover, U-EDA^2, O-EDA^2, and M-EDA^2 surpass EDA^2 on 16, 18, and 16 test functions, which means the three components of EMSM play an important role in improving the performance of EDA^2. In addition, U-EDA^2, O-EDA^2, and M-EDA^2 is surpassed by EMSM-EDA on 18, 21, and 21 test functions, respectively. It means that the three components jointly improve the performance of EDA^2.

5.5. Comparisons with the other EAs on IEEE CEC2013 test suite

This part compares EMSM-EDA with EMNA$_g$ [8], AMaLGaM [6], JADE [46], and SaDE [47] on IEEE CEC2013 test suite. The parameter settings of EMNA$_g$, JADE, and SaDE are based on the above references. The test results of AMaLGaM are directly obtained from reference [6]. To analyze the comparison results more clearly, this paper utilizes the method in [48], where "



(a) CEC2014$_1$

(b) CEC2014$_2$

(c) CEC2014$_{17}$

(d) CEC2014$_{18}$

Fig. 9. Evolution of FEVs derived from EMSM-EDA and EDA^2 on 4 functions of CEC2014 benchmark functions with 100D.

+", "-", and "≈" denote that the performance of the comparison algorithm is better than, worse than, or similar to EMSM-EDA, respectively. In addition, the best results are bolded. Tables 7 and 8 summarize the test results obtained by the five algorithms on test suite with 30D and 50D. The comments can be obtained.

1. CEC2013$_1$-CEC2013$_5$ are unimodal functions, on which EMSM-EDA shows the best performance among the five algorithms with 30D and 50D. For the case of 30D, EMSM-EDA can find the optimal solutions for all 5 test functions, while AMaLGaM fails on CEC2013$_3$. Moreover, JADE and SaDE only obtain the optimal solutions of CEC2013$_1$ and CEC2013$_5$. In particular, EMNA$_g$ cannot find the optimal solutions for all 5 test functions. When the test functions are 50D, EMSM-EDA finds the optimal solutions for 4 test functions and still has the best performance.
2. For basic multimodal functions CEC2013$_6$-CEC2013$_{20}$ with 30D, EMSM-EDA and JADE have better performance which can obtain 8 best solutions for all 15 test functions. AMaLGAM obtains 2 best solutions. Moreover, there is only 1 best solution for EMNA$_g$ and SaDE. In the case of 50D, EMSM-EDA is surpassed by JADE on 9 functions, which indicates the performance of JADE is superior to EMSM-EDA. EMSM-EDA performs much better than EMNA$_g$, AMaLGAM, and SaDE.
3. For complicated composition functions CEC2013$_{21}$- CEC2013$_{28}$ with 30D, EMSM-EDA finds 4 best solutions for all 8 test functions and has the best performance. As a comparison, AMaLGAM and JADE can obtain 3 best solutions. Moreover, SaDE only finds 1 best solution. As a classical GEDA, EMNA$_g$ cannot obtain best solutions. For test functions with 50D, although EMSM-EDA defeats EMNA$_g$, JADE, and SaDE, it is surpassed by AMaLGAM on 4 test functions, which indicates AMaLGAM performs better than EMSM-EDA.

The overall results of five algorithms are summarized in the last rows of Tables 7 and 8. It is clear that EMSM-EDA outperforms EMNA$_g$, AMaLGAM, JADE, and SaDE on IEEE CEC2013 test suite. To vividly describe the advantage of EMSM-EDA, Fig. 10 and Fig. 11 present the evolution curves of five comparison algorithms on 4 test functions with 30D and 50D, respectively. The evolution curves indicate that EMSM-EDA has the fastest convergence speed and excellent performance.

Table 6
The means and the standard deviations of FEVs obtained by the five algorithms over 25 independent runs on 30 CEC2014 benchmark functions with 30D.

Functions	EDA^2	U-EDA^2	O-EDA^2	M-EDA^2	EMSM-EDA
CEC2014$_1$	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
CEC2014$_2$	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
CEC2014$_3$	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
CEC2014$_4$	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
CEC2014$_5$	2.10E+01±3.40E-02-	2.09E+01±4.23E-02≈	2.10E+01±4.57E-02-	2.10E+01±7.42E-02-	2.09E+01±5.11E-02
CEC2014$_6$	1.89E-01±4.75E-01-	1.72E-01±5.13E-01-	1.80E-01±2.53E-01-	1.78E-01±3.92E-01-	1.60E-01±4.09E-01
CEC2014$_7$	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
CEC2014$_8$	6.13E+00±2.47E+00-	5.43E+00±3.01E+00-	4.97E+00±2.72E+00-	4.81E+00±2.31E+00-	4.42E+00±2.28E+00
CEC2014$_9$	6.49E+00±2.26E+00-	4.98E+00±1.74E+00-	5.36E+00±2.43E+00-	6.51E+00±4.92E+00-	3.98E+00±1.84E+00
CEC2014$_{10}$	2.85E+01±5.60E+01-	2.84E+01±6.02E+01-	2.53E+01±4.78E+01-	2.02E+01±4.05E+01-	1.88E+01±3.87E+01
CEC2014$_{11}$	1.24E+02±1.01E+02-	1.02E+02±9.84E+01-	1.20E+02±1.00E+02-	9.87E+01±8.99E+01-	9.42E+01±9.42E+01
CEC2014$_{12}$	2.48E+00±2.41E-01-	2.49E+00±3.42E-01-	2.48E+00±2.51E-01-	2.46E+00±2.85E-01-	2.45E+00±2.71E-01
CEC2014$_{13}$	4.52E-02±1.02E-02-	4.47E-02±1.32E-02-	3.84E-02±4.36E-02-	4.01E-02±3.47E-02-	3.62E-02±3.62E-02
CEC2014$_{14}$	4.08E-01±3.53E-02-	3.84E-01±4.52E-01-	3.97E-01±5.13E-01-	4.01E-01±6.42E-01-	3.62E-01±5.66E-02
CEC2014$_{15}$	4.09E+00±1.27E+00+	4.31E+00±1.25E+00-	4.24E+00±1.52E+00-	4.19E+00±1.69E+00-	4.11E+00±1.37E+00
CEC2014$_{16}$	1.22E+01±1.85E-01-	1.20E+01±2.41E-01≈	1.21E+01±3.15E-01-	1.23E+01±5.23E-01-	1.20E+01±3.30E-01
CEC2014$_{17}$	1.40E+01±8.26E+00-	6.73E+00±7.12E+00-	1.23E+01±1.06E+01-	1.37E+01±1.14E+01-	5.35E+00±5.19E+00
CEC2014$_{18}$	8.74E-01±6.12E-01-	7.72E-01±5.48E-01-	6.13E-01±5.62E-01-	8.02E-01±6.04E-01-	5.56E-01±2.64E-01
CEC2014$_{19}$	3.88E+00±6.63E-01-	3.54E+00±6.52E-01-	3.21E+00±3.14E-01-	3.19E+00±3.04E-01-	3.15E+00±2.98E-01
CEC2014$_{20}$	1.48E+00±2.80E-01-	1.48E+00±3.05E-01-	1.47E+00±3.42E-01-	1.47E+00±3.42E-01-	1.45E+00±3.48E-01
CEC2014$_{21}$	6.28E+00±2.37E+01-	3.01E+00±2.76E+00-	4.72E+00±4.31E+00-	2.03E+00±5.13E+00-	1.97E+00±4.06E+00
CEC2014$_{22}$	4.73E+01±4.54E+01-	4.62E+00±3.55E+00+	4.72E+00±4.51E+00-	4.74E+00±6.27E+00-	4.70E+01±4.46E+01
CEC2014$_{23}$	3.15E+02±0.00E+00≈	3.15E+02±0.00E+00≈	3.15E+02±0.00E+00≈	3.15E+02±0.00E+00≈	3.15E+02±0.00E+00
CEC2014$_{24}$	2.00E+02±0.00E+00≈	2.00E+02±0.00E+00≈	2.00E+02±0.00E+00≈	2.00E+02±0.00E+00≈	2.00E+02±0.00E+00
CEC2014$_{25}$	2.03E+02±3.11E-02≈	2.03E+02±3.11E-02≈	2.03E+02±3.11E-02≈	2.03E+02±3.11E-02≈	2.03E+02±2.12E-02
CEC2014$_{26}$	1.00E+02±1.10E-02≈	1.00E+02±1.52E-02≈	1.00E+02±2.31E-02≈	1.00E+02±1.21E-02≈	1.00E+02±7.73E-03
CEC2014$_{27}$	3.49E+02±4.41E+01-	3.45E+02±4.36E+01-	3.37E+02±5.01E+01-	3.31E+02±4.36E+01-	3.29E+02±4.22E+01
CEC2014$_{28}$	8.12E+02±4.36E+01-	8.12E+00±3.48E+01-	8.08E+00±5.42E+01-	8.09E+00±5.41E+01-	8.06E+02±4.32E+01
CEC2014$_{29}$	7.65E+05±2.67E+06-	9.73E+02±4.36E+02-	7.21E+02±3.61E+02-	1.04E+03±7.35E+02-	6.21E+02±2.22E+02
CEC2014$_{30}$	1.05E+03±4.44E+02-	1.02E+03±3.42E+02-	9.77E+02±4.52E+00-	1.04E+03±5.62E+02-	9.04E+02±4.30E+02
No.of+/≈/-	**1/9/20**	**1/11/18**	**0/9/21**	**0/9/21**	**NA**

"+", "-", and "≈" indicate that the performance of the corresponding algorithm is better than, worse than, and similar to that of EMSM-EDA, respectively.

5.6. Comparisons with the other EAs on IEEE CEC2014 test suite

To further evaluate the performance of EMSM-EDA, this paper compares it with five algorithms on IEEE CEC2014 test suite with 30D and 50D. Besides EMNA$_g$, AMaLGaM, JADE, and SaDE, one new competitor is added to the experiment, i.e., BL-PSO-5 [49]. The parameter settings of EMNA$_g$, AMaLGaM, JADE, and SaDE are the same as IEEE CEC2013 test experiment. The test results of BL-PSO-5 are directly obtained from reference [49]. Tables 9 and 10 summarize the test results of the six algorithms. The comments can be obtained.

1. For the unimodal functions CEC2014$_1$-CEC2014$_3$ with 30D and 50D, EMSM-EDA can find the optimal solutions for all 3 test functions and achieve perfect performance. AMaLGaM fails on CEC2014$_2$ with 50D. EMNA$_g$ and BL-PSO-5 cannot find the optimal solutions. For JADE, it obtains the optimal solutions on CEC2014$_2$ with 30D and 50D. SaDE only obtains the optimal solution on CEC2014$_2$ with 30D.
2. CEC2014$_4$-CEC2014$_{16}$ are simple multimodal functions, on which EMSM-EDA finds 5 best solutions and demonstrates the best performance among the six algorithms with 30D. AMaLGaM, BL-PSO-5, JADE, and SaDE find 2, 3, 4, and 1 best solution for all 13 test functions, respectively. EMNA$_g$ cannot find the best solutions for all 13 test functions with 30D. Nevertheless, EMSM-EDA is surpassed by JADE on 5 out of total 13 functions with 50D, which means JADE performs better than EMSM-EDA. AMaLGAM and BL-PSO-5 can obtain 3 best solutions for all 13 test functions. For EMNA$_g$ and SaDE, they cannot obtain the best solutions.
3. Regarding to the hybrid functions CEC2014$_{17}$-CEC2014$_{22}$ with 30D and 50D, EMSM-EDA finds the best solutions for all 6 functions. Specifically, EMSM-EDA improves EMNA$_g$, AMaLGaM, BL-PSO-5, JADE, and SaDE on CEC2014$_{17}$ and CEC2014$_{18}$ functions with 30D by two order of magnitude. Meanwhile, EMSM-EDA also improves EMNA$_g$, AMaLGaM, BL-PSO-5, JADE, and SaDE on CEC2014$_{17}$, CEC2014$_{18}$, and CEC2014$_{19}$ functions with 50D by one order of magnitude. To summarise, EMSM-EDA is the champion on the hybrid functions.
4. EMSM-EDA has the excellent performance on composition functions CEC 2014$_{23}$-CEC2014$_{30}$. For the functions with 30D, EMSM-EDA outperforms five comparison algorithms on 8, 4, 5, 5, and 7 out of all 8 functions, respectively. In addition, EMSM-EDA and AMaLGaM find 4 best solutions for 8 test functions with 50D and perform better than the other four algorithms. Specially, EMNA$_g$, BL-PSO-5, JADE, and SaDE find 0, 2, 3, and 1 best solution for 8 test functions with 50D, respectively.

Table 7
The means and the standard deviations of FEVs obtained by the five algorithms over 25 independent runs on 28 CEC2013 benchmark functions with 30D.

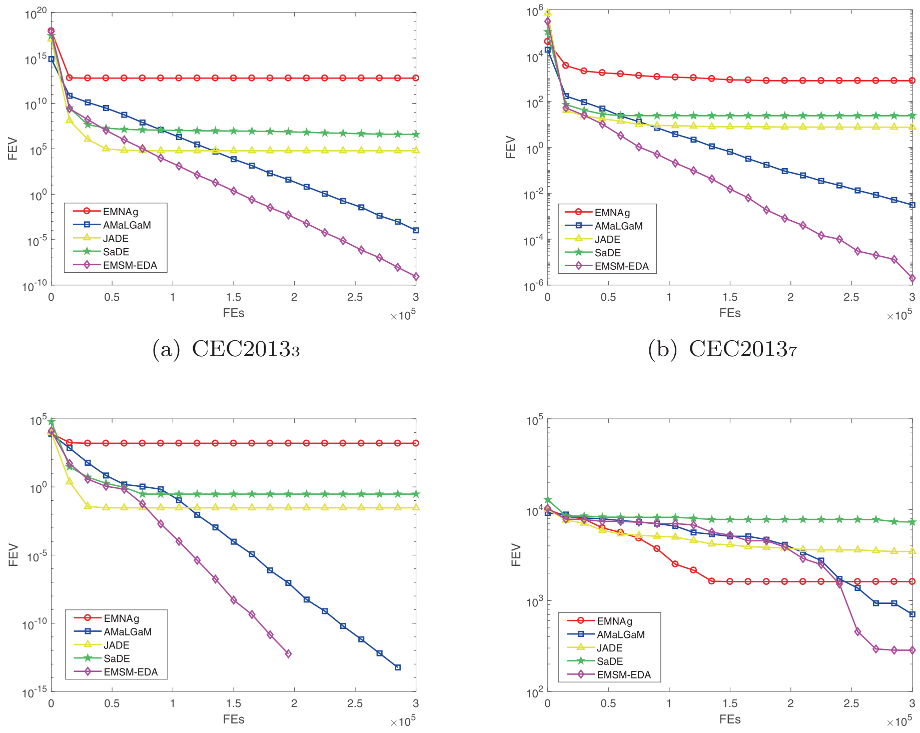
Functions	EMNA$_g$	AMaLGaM	JADE	SaDE	EMSM-EDA
CEC2013$_1$	1.29E+04±1.12E+03-	**0.00E+00±0.00E+00**≈	**0.00E+00±0.00E+00**≈	**0.00E+00±0.00E+00**≈	**0.00E+00±0.00E+00**
CEC2013$_2$	6.49E+07±2.05E+07-	**0.00E+00±0.00E+00**≈	5.70E+03±4.20E+03-	4.06E+05±1.94E+05-	**0.00E+00±0.00E+00**
CEC2013$_3$	1.06E+13±1.28E+13-	5.95E-02±2.48E-01-	6.48E+05±3.18E+06-	1.66E+07±1.82E+07-	**0.00E+00±0.00E+00**
CEC2013$_4$	2.52E+04±2.56E+03-	**0.00E+00±0.00E+00**≈	4.37E+03±1.21E+04-	3.90E+03±1.90E+03-	**0.00E+00±0.00E+00**
CEC2013$_5$	1.11E+03±3.38E+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**
CEC2013$_6$	2.01E+03±2.33E+02-	1.04E+01±3.75E+00-	2.11E+00±7.31E+00-	3.43E+01±2.76E+01-	**1.06E+00±5.28E+00**
CEC2013$_7$	1.10E+03±1.03E+03-	6.60E-01±9.65E-01-	4.65E+00±4.99E+00-	2.60E+01±1.32E+01-	**5.93E-06±1.04E-05**
CEC2013$_8$	**2.09E+01±4.51E-02**≈	**2.09E+01±5.58E-02**≈	**2.09E+01±6.10E-02**≈	**2.09E+01±3.52E-02**≈	**2.09E+01±3.99E-02**
CEC2013$_9$	1.21E+01±1.79E+00-	6.25E+00±1.72E+00-	2.67E+01±1.33E+00-	1.76E+01±3.16E+00-	**3.44E+00±1.17E+00**
CEC2013$_{10}$	1.94E+03±2.88E+02-	**0.00E+00±0.00E+00**≈	4.08E-02±2.76E-02-	3.23E-01±1.95E-01-	**0.00E+00±0.00E+00**
CEC2013$_{11}$	9.72E+01±1.56E+01-	4.01E+00±1.78E+00-	**0.00E+00±0.00E+00**+	1.99E-01±4.06E-01+	3.94E+00±1.94E+00
CEC2013$_{12}$	1.07E+02±1.59E+01-	3.10E+00±1.75E+00-	2.36E+01±4.40E+00-	4.59E+01±1.33E+01-	**2.79E+00±1.75E+00**
CEC2013$_{13}$	1.88E+02±2.56E+01-	3.08E+00±1.82E+00-	4.67E+01±9.63E+00-	1.07E+02±2.81E+01-	**2.99E+00±1.66E+00**
CEC2013$_{14}$	1.76E+03±4.63E+02-	7.61E+02±4.89E+02-	**2.08E-02±2.17E-02**+	6.89E-01±1.03E+00+	5.49E+02±5.36E+02
CEC2013$_{15}$	1.32E+03±4.08E+02-	7.18E+02±2.22E+02-	3.25E+03±2.61E+02-	4.50E+03±1.04E+03-	**3.88E+02±7.24E+02**
CEC2013$_{16}$	2.36E+00±2.89E-01+	2.35E+00±1.07E+00+	**1.90E+00±6.98E-01**+	2.25E+00±2.07E-01+	2.44E+00±2.82E-01
CEC2013$_{17}$	4.84E+01±6.96E+00+	3.26E+01±7.30E-01+	**3.04E+01±0.00E+00**+	3.05E+01±4.38E-02+	1.69E+02±7.79E+00
CEC2013$_{18}$	1.13E+02±1.15E+01+	9.94E+01±7.05E+01+	**7.60E+01±6.65E+00**+	1.22E+02±4.09E+01+	1.73E+02±8.83E+00
CEC2013$_{19}$	1.60E+03±6.07E+02-	3.02E+00±3.52E-01+	**1.41E+00±1.39E-01**+	4.04E+00±8.57E-01+	4.09E+00±1.71E+00
CEC2013$_{20}$	1.29E+01±1.19E+00-	1.19E+01±1.15E+00-	**1.05E+01±6.34E-01**+	1.08E+01±6.34E-01+	1.16E+01±3.12E-01
CEC2013$_{21}$	8.30E+02±2.49E+02-	3.01E+02±6.51E+01-	3.16E+02±9.08E+01-	3.18E+02±8.04E+01-	**2.78E+02±5.71E+01**
CEC2013$_{22}$	1.43E+03±2.69E+02-	7.27E+02±2.47E+02-	**8.78E+01±3.79E+01**+	1.13E+02±3.75E+00+	1.95E+02±7.65E+01
CEC2013$_{23}$	1.48E+03±4.99E+02-	7.55E+02±2.74E+02-	3.51E+03±3.68E+02-	4.73E+03±1.00E+03-	**2.85E+02±1.29E+02**
CEC2013$_{24}$	2.80E+02±8.13E+00-	**2.00E+02±1.51E-02**+	2.14E+02±1.41E+01+	2.29E+02±6.95E+00-	2.18E+02±1.45E+01
CEC2013$_{25}$	2.77E+02±8.28E+00-	2.45E+02±6.19E+00-	2.78E+02±1.15E+01-	2.65E+02±1.73E+01-	**2.44E+02±5.55E+00**
CEC2013$_{26}$	2.79E+02±2.38E+01-	2.87E+02±1.86E+01-	**2.06E+02±3.07E+01**+	2.10E+02±3.51E+01+	2.78E+02±3.69E+01
CEC2013$_{27}$	6.62E+02±3.24E+01-	**3.00E+02±3.17E-01**+	6.69E+02±2.15E+02-	6.00E+02±5.79E+01-	3.57E+02±8.25E+01
CEC2013$_{28}$	1.57E+03±4.84E+02-	**3.00E+02±0.00E+00**≈	**3.00E+02±0.00E+00**≈	**3.00E+02±0.00E+00**≈	**3.00E+02±0.00E+00**
No.of+/≈/-	**3/1/24**	**6/7/15**	**10/4/14**	**9/4/15**	**NA**

"+", "-", and "≈" indicate that the performance of the corresponding algorithm is better than, worse than, and similar to that of EMSM-EDA, respectively.

Table 8
The means and the standard deviations of FEVs obtained by the five algorithms over 25 independent runs on 28 CEC2013 benchmark functions with 50D.

Functions	EMNA$_g$	AMaLGaM	JADE	SaDE	EMSM-EDA
CEC2013$_1$	1.97E+04±1.22E+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**
CEC2013$_2$	2.37E+08±4.77E+07-	**0.00E+00±0.00E+00**≈	2.51E+04±1.58E+04-	7.99E+05±3.24E+05-	**0.00E+00±0.00E+00**
CEC2013$_3$	3.62E+10±8.14E+09-	3.91E+00±1.95E+01-	3.24E+06±5.72E+06-	6.29E+07±5.95E+07-	**5.00E-04±2.23E-04**
CEC2013$_4$	3.64E+04±3.04E+03-	3.13E-08±1.18E-07-	6.01E+03±1.67E+04-	4.42E+03±1.65E+03-	**0.00E+00±0.00E+00**
CEC2013$_5$	1.12E+03±2.09E+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**
CEC2013$_6$	1.00E+03±1.30E+02-	4.38E+01±5.77E-01-	**4.34E+01±0.00E+00**≈	4.89E+01±1.76E+01-	**4.34E+01±0.00E+00**
CEC2013$_7$	4.03E+01±4.99E+00-	6.84E-01±1.20E+00-	2.09E+01±1.17E+01-	4.96E+01±1.01E+01-	**2.01E-04±3.57E-05**
CEC2013$_8$	**2.11E+01±4.49E-02**≈	**2.11E+01±3.12E-02**≈	**2.11E+01±2.76E-02**≈	**2.11E+01±3.71E-02**≈	**2.11E+01±4.12E-02**
CEC2013$_9$	2.15E+01±2.29E+00-	9.00E+00±3.55E+00-	5.40E+01±2.24E+00-	3.91E+01±4.13E+00-	**4.40E+00±2.57E+00**
CEC2013$_{10}$	2.56E+03±2.27E+02-	**0.00E+00±0.00E+00**≈	4.68E-02±3.29E-02-	2.68E-01±1.26E-01-	**0.00E+00±0.00E+00**
CEC2013$_{11}$	1.35E+02±1.47E+01-	6.48E+00±2.28E+00-	**0.00E+00±0.00E+00**+	2.91E+00±2.91E+00+	3.74E+00±1.80E+00
CEC2013$_{12}$	1.92E+02±1.89E+01-	5.37E+00±1.94E+00-	5.65E+01±9.80E+00-	1.18E+02±1.94E+01-	**3.24E+00±1.79E+00**
CEC2013$_{13}$	3.41E+02±2.69E+01-	5.39E+00±2.17E+00-	1.33E+02±2.82E+01-	2.66E+02±4.74E+01-	**2.82E+00±1.37E+00**
CEC2013$_{14}$	3.13E+03±6.13E+02-	1.37E+03±4.34E+02-	**5.05E-02±2.24E-02**+	6.17E+00±4.36E+00+	1.48E+02±1.20E+02
CEC2013$_{15}$	2.83E+03±4.12E+02-	1.51E+03±4.21E+02-	6.87E+03±5.26E+02-	8.93E+03±2.23E+03-	**4.26E+02±1.80E+02**
CEC2013$_{16}$	3.28E+00±3.36E-01-	3.17E+00±9.85E-01+	**2.03E+00±7.42E-01**+	3.00E+00±2.96E-01+	3.24E+00±3.24E-01
CEC2013$_{17}$	9.10E+01±1.23E+01+	5.76E+01±3.04E+00+	**5.08E+01±3.41E-14**+	5.14E+01±3.11E-01+	3.27E+02±1.69E+01
CEC2013$_{18}$	3.26E+02±3.15E+01+	2.56E+02±1.09E+02+	**1.43E+02±1.18E+01**+	1.67E+02±8.28E+01+	3.36E+02±1.52E+01
CEC2013$_{19}$	5.96E+03±2.42E+03-	5.20E+00±4.74E-01+	**2.78E+00±1.93E-01**+	1.25E+01±2.48E+00-	9.79E+00±2.47E+00
CEC2013$_{20}$	2.09E+01±4.22E-01+	2.06E+01±4.74E-01+	**1.97E+01±7.69E-01**+	2.02E+01±8.48E-01+	2.12E+01±3.40E-01
CEC2013$_{21}$	1.14E+03±3.10E+02-	**3.47E+02±3.45E+02**+	8.04E+02±4.30E+02-	8.72E+02±3.27E+02-	6.80E+02±4.70E+02
CEC2013$_{22}$	3.29E+03±5.72E+02-	1.17E+03±5.02E+02-	**1.38E+01±1.14E+01**+	2.39E+01±6.01E+00+	1.97E+02±8.99E+01
CEC2013$_{23}$	2.84E+03±6.97E+02-	1.45E+03±4.69E+02-	7.09E+03±4.78E+02-	8.85E+03±2.04E+03-	**4.98E+02±2.08E+02**
CEC2013$_{24}$	4.54E+02±2.43E+01-	**2.00E+02±2.88E-01**+	2.52E+02±2.12E+01-	2.79E+02±1.05E+01-	2.22E+02±2.17E+01
CEC2013$_{25}$	3.72E+02±1.27E+01-	2.82E+02±6.15E+00-	3.53E+02±1.79E+01-	3.46E+02±1.03E+01-	**2.77E+02±7.73E+00**
CEC2013$_{26}$	3.42E+02±1.97E+01-	3.00E+02±3.65E+00+	3.35E+02±1.05E+02-	**2.57E+02±8.51E+01**+	3.04E+02±1.19E+01
CEC2013$_{27}$	1.31E+03±6.48E+01-	**3.33E+02±1.12E+02**+	1.43E+03±3.20E+02-	1.20E+03±1.18E+02-	4.84E+02±1.47E+02
CEC2013$_{28}$	2.30E+03±1.98E+02-	**4.00E+02±0.00E+00**≈	8.75E+02±1.11E+03-	**4.00E+02±0.00E+00**≈	**4.00E+02±2.82E-08**
No.of+/≈/-	**3/1/24**	**9/6/13**	**8/4/16**	**8/4/16**	**NA**

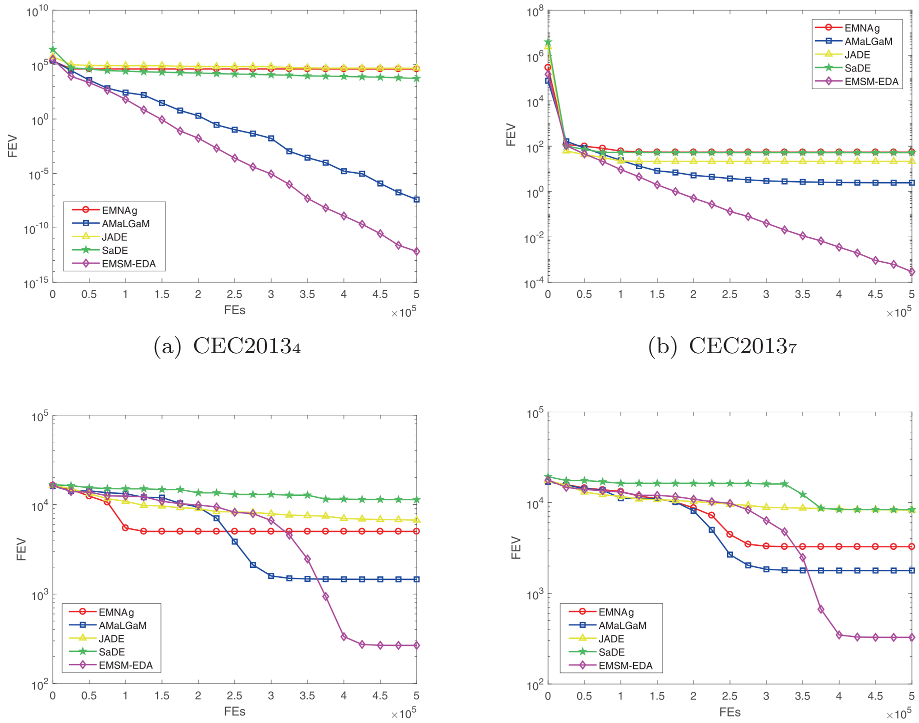
"+", "-", and "≈" indicate that the performance of the corresponding algorithm is better than, worse than, and similar to that of EMSM-EDA, respectively.



(a) CEC2013$_3$ (b) CEC2013$_7$

(c) CEC2013$_{10}$ (d) CEC2013$_{23}$

Fig. 10. Evolution of FEVs derived from EMNA$_g$, AMaLGaM, JADE, SaDE, and EMSM-EDA on 4 functions of CEC2013 benchmark functions with 30D.



(a) CEC2013$_4$ (b) CEC2013$_7$

(c) CEC2013$_{15}$ (d) CEC2013$_{23}$

Fig. 11. Evolution of FEVs derived from EMNA$_g$, AMaLGaM, JADE, SaDE, and EMSM-EDA on 4 functions of CEC2013 benchmark functions with 50D.

Table 9
The means and the standard deviations of FEVs obtained by the six algorithms over 25 independent runs on 30 CEC2014 benchmark functions with 30D.

Functions	EMNA$_g$	AMaLGaM	BL-PSO-5	JADE	SaDE	EMSM-EDA
CEC2014$_{1}$	1.54E+08±3.53E+07-	**0.00E+00±0.00E+00**≈	2.99E+06±1.10E+06-	4.40E+02±9.75E+02-	3.62E+05±2.65E+05-	**0.00E+00±0.00E+00**
CEC2014$_{2}$	1.39E+10±1.67E+09-	**0.00E+00±0.00E+00**≈	5.09E+03±4.25E+03-	**0.00E+00±0.00E+00**≈	**0.00E+00±0.00E+00**≈	**0.00E+00±0.00E+00**
CEC2014$_{3}$	1.80E+04±2.89E+03-	**0.00E+00±0.00E+00**≈	3.67E+00±1.16E+01-	1.74E-03±6.43E-03-	1.78E+00±3.31E+00-	**0.00E+00±0.00E+00**
CEC2014$_{4}$	1.64E+03±3.15E+02-	1.05E+01±2.10E+01-	2.68E+01±3.47E+01-	2.54E+00±1.27E+01-	3.95E+01±3.89E+01-	**0.00E+00±0.00E+00**
CEC2014$_{5}$	2.09E+01±5.05E-02≈	2.09E+01±2.05E-01≈	2.08E+01±7.01E-02≈	**2.03E+01±3.40E-02**+	2.05E+01±4.80E-02+	2.09E+01±5.11E-02
CEC2014$_{6}$	4.69E+00±1.22E+00-	1.83E-02±9.18E-02+	**9.37E-06±3.20E-05**+	9.47E+00±1.81E+00-	5.38E+00±2.25E+00-	1.60E-01±4.09E-01
CEC2014$_{7}$	1.62E+02±2.29E+01-	**0.00E+00±0.00E+00**≈	**0.00E+00±0.00E+00**≈	2.96E-04±1.48E-03-	7.08E-03±1.43E-02-	**0.00E+00±0.00E+00**
CEC2014$_{8}$	3.70E+01±7.20E+00-	3.86E+00±1.77E+00+	2.32E-01±5.65E-01+	0.00E+00±0.00E+00+	**3.98E-02±1.99E-01**+	4.42E+00±2.28E+00
CEC2014$_{9}$	3.31E+01±6.95E+00-	4.01E+00±1.63E+00-	3.54E+01±6.93E+01-	2.61E+01±3.70E+00-	3.80E+01±8.85E+00-	**3.98E+00±1.84E+00**
CEC2014$_{10}$	9.73E+02±3.34E+02-	4.41E+02±2.60E+02-	8.83E+01±6.48E+01-	**7.49E-03±1.33E-02**+	2.17E-01±5.00E-01+	1.88E+01±3.87E+01
CEC2014$_{11}$	1.26E+03±4.52E+02-	5.17E+02±2.91E+02-	2.08E+03±3.82E+02-	1.65E+03±2.20E+02-	3.15E+03±7.14E+02-	**9.42E+01±9.42E+01**
CEC2014$_{12}$	2.43E+00±2.71E-01+	2.45E+00±7.98E-01≈	8.83E-01±1.49E-01+	**2.65E-01±3.52E-02**+	7.91E-01±1.09E-01+	2.45E+00±2.71E-01
CEC2014$_{13}$	4.07E+00±1.73E-01-	5.63E-02±1.12E-02-	2.21E-01±2.85E-02-	2.17E-01±4.26E-02-	2.58E-01±3.89E-02-	**3.62E-02±3.62E-02**
CEC2014$_{14}$	7.92E+01±9.51E+00-	4.18E-01±3.87E-02-	**2.14E-01±2.88E-02**+	2.32E-01±3.23E-02+	2.37E-01±3.55E-02+	3.62E-01±5.66E-02
CEC2014$_{15}$	2.69E+01±2.53E+01-	**3.03E+00±5.42E-01**+	7.41E+00±8.49E-01-	3.16E+00±3.60E-01+	4.11E+00±1.28E+00≈	4.11E+00±1.37E+00
CEC2014$_{16}$	1.10E+01±5.32E-01+	1.08E+01±4.76E-01+	9.67E+00±4.92E-01+	**9.38E+00±2.93E-01**+	1.10E+01±2.85E-01+	1.20E+01±3.30E-01
CEC2014$_{17}$	1.30E+06±8.43E+05-	2.31E+02±1.34E+02-	1.86E+05±1.11E+05-	1.31E+03±4.29E+02-	1.00E+04±8.27E+03-	**5.35E+00±5.19E+00**
CEC2014$_{18}$	6.47E+01±1.26E+01-	1.64E+01±6.54E+00-	9.05E+02±1.20E+03-	8.92E+01±3.85E+01-	5.19E+02±6.10E+02-	**5.56E-01±2.64E-01**
CEC2014$_{19}$	1.52E+01±6.36E+00-	3.78E+00±7.31E-01-	3.74E+00±6.12E-01-	4.35E+00±7.03E-01-	3.85E+00±8.99E-01-	**3.15E+00±2.98E-01**
CEC2014$_{20}$	2.79E+03±1.13E+03-	3.05E+00±1.85E+00-	3.12E+02±3.48E+02-	3.41E+03±3.07E+03-	9.28E+01±7.61E+01-	**1.45E+00±3.48E-01**
CEC2014$_{21}$	2.76E+04±3.95E+04-	7.80E+01±7.07E+01-	3.85E+04±3.19E+04-	2.98E+02±1.17E+02-	2.07E+03±2.00E+03-	**1.97E+00±4.06E+00**
CEC2014$_{22}$	1.11E+02±6.32E+01-	9.22E+01±6.09E+01-	1.16E+02±6.86E+01-	1.76E+02±8.59E+01-	1.56E+02±9.64E+01-	**4.70E+01±4.46E+01**
CEC2014$_{23}$	3.63E+02±8.45E+00-	**3.15E+02±0.00E+00**≈	**3.15E+02±0.00E+00**≈	**3.15E+02±0.00E+00**≈	**3.15E+02±0.00E+00**≈	**3.15E+02±0.00E+00**
CEC2014$_{24}$	2.31E+02±2.45E+00-	2.23E+02±1.28E+00-	2.22E+02±7.39E-01-	2.25E+02±3.46E+00-	2.25E+02±1.56E+00-	**2.00E+02±2.91E-08**
CEC2014$_{25}$	2.09E+02±4.11E-01-	**2.02E+02±3.64E-02**+	2.05E+02±4.41E-01-	2.04E+02±9.97E-01-	2.09E+02±1.76E+00-	2.03E+02±2.12E-02
CEC2014$_{26}$	1.04E+02±9.07E-01-	**1.00E+02±4.40E-02**≈	1.04E+02±1.82E+01-	**1.00E+02±4.53E-02**≈	1.08E+02±2.76E+01-	**1.00E+02±7.73E-03**
CEC2014$_{27}$	5.65E+02±2.73E+01-	**3.02E+02±1.07E+01**+	3.08E+02±2.94E+01+	3.31E+02±4.54E+01-	4.25E+02±4.15E+01-	3.29E+02±4.22E+01
CEC2014$_{28}$	9.59E+02±1.11E+02-	9.11E+02±3.46E+01-	**7.87E+02±5.22E+01**+	7.90E+02±3.91E+01+	8.88E+02±3.13E+01-	8.06E+02±4.32E+01
CEC2014$_{29}$	1.78E+03±1.45E+03-	7.32E+02±9.12E+01-	1.39E+03±1.39E+02-	7.31E+02±1.68E+01-	1.05E+03±2.07E+02-	**6.21E+02±2.22E+02**
CEC2014$_{30}$	3.81E+03±1.16E+03-	1.60E+03±8.18E+02-	1.19E+03±2.49E+02-	1.66E+03±6.78E+02-	1.85E+03±6.41E+02-	**9.04E+02±4.30E+02**
No.of+/≈/-	**2/1/27**	**6/8/16**	**7/3/20**	**8/3/19**	**6/3/21**	**NA**

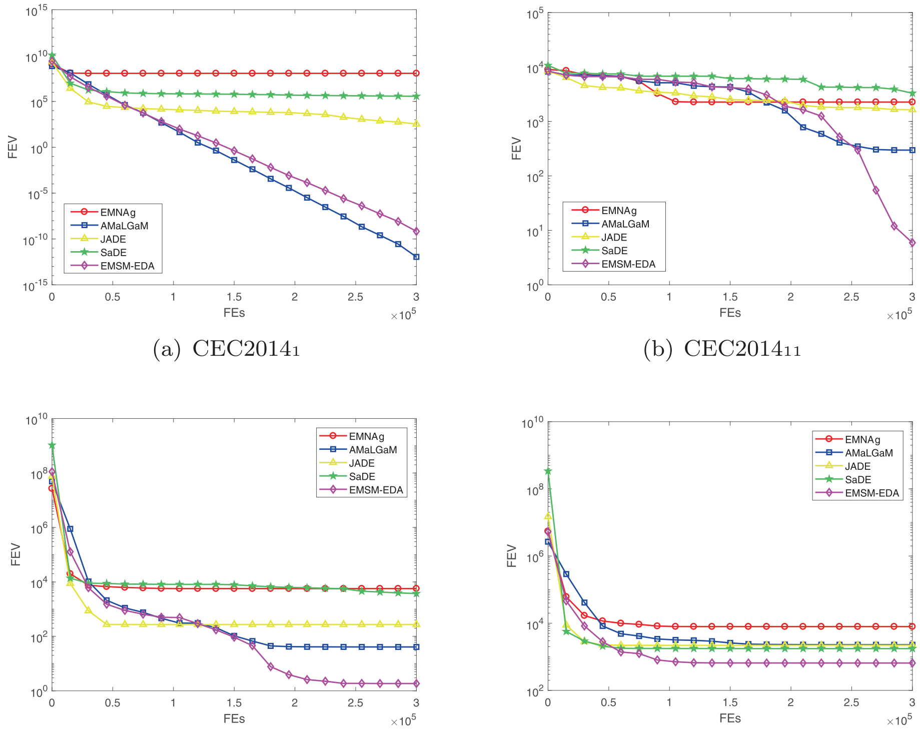
"+", "-", and "≈" indicate that the performance of the corresponding algorithm is better than, worse than, and similar to that of EMSM-EDA, respectively.

Table 10

The means and the standard deviations of FEVs obtained by the six algorithms over 25 independent runs on 30 CEC2014 benchmark functions with 50D.

Functions	EMNA$_g$	AMaLGaM	BL-PSO-5	JADE	SaDE	EMSM-EDA
CEC2014$_1$	8.66E+08±1.16E+08-	**0.00E+00**±**0.00E+00**≈	5.10E+06±1.28E+06-	1.25E+04±8.85E+03-	9.43E+05±2.70E+05-	**0.00E+00**±**0.00E+00**
CEC2014$_2$	5.39E+10±3.83E+09-	2.77E-07±5.83E-07-	3.44E+03±2.35E+03-	**0.00E+00**±**0.00E+00**≈	4.14E+03±4.22E+03-	**0.00E+00**±**0.00E+00**
CEC2014$_3$	5.13E+04±5.76E+03-	**0.00E+00**±**0.00E+00**≈	4.23E+01±8.97E+01-	3.28E+03±2.56E+03-	4.28E+03±2.45E+03-	**0.00E+00**±**0.00E+00**
CEC2014$_4$	8.37E+03±1.02E+03-	9.46E+01±6.19E+00-	**8.64E+00**±**5.04E+00**+	1.21E+01±3.24E+01+	7.09E+01±4.08E+01-	4.77E+01±4.95E+01
CEC2014$_5$	2.11E+01±3.61E-02≈	2.11E+01±4.26E-02≈	2.09E+01±5.07E-02+	**2.04E+01**±**3.29E-02**+	2.07E+01±2.89E-02+	2.11E+01±4.30E-02
CEC2014$_6$	2.08E+01±2.08E+00-	**0.00E+00**±**0.00E+00**+	9.22E-02±3.21E-01+	1.82E+01±5.61E+00-	1.88E+01±3.73E+00-	2.55E-01±4.49E-01
CEC2014$_7$	5.04E+02±3.00E+01-	**0.00E+00**±**0.00E+00**≈	**0.00E+00**±**0.00E+00**≈	1.77E-03±4.63E-03-	1.39E-02±1.42E-02-	**0.00E+00**±**0.00E+00**
CEC2014$_8$	1.80E+02±1.85E+01-	5.45E+00±2.20E+00≈	4.97E-01±8.16E-01+	**0.00E+00**±**0.00E+00**+	1.67E+00±1.34E+00+	5.53E+00±1.89E+00
CEC2014$_9$	1.77E+02±1.89E+01-	5.37E+00±2.24E+00-	7.10E+01±9.02E+00-	5.18E+01±8.61E+00-	8.90E+01±1.35E+01-	**5.25E+00**±**2.25E+00**
CEC2014$_{10}$	4.87E+03±7.16E+02-	8.91E+02±3.46E+02-	3.63E+02±1.81E+02-	**1.35E-02**±**1.57E-02**+	1.28E+00±1.10E+00+	9.33E+01±1.48E+02
CEC2014$_{11}$	4.98E+03±5.43E+02-	1.07E+03±4.05E+02-	4.46E+03±4.77E+02-	3.85E+03±3.50E+02-	6.61E+03±1.61E+03-	**8.27E+01**±**1.12E+02**
CEC2014$_{12}$	1.40E+00±1.19E+00+	3.04E+00±1.17E+00+	8.77E-01±1.18E-01+	**2.62E-01**±**3.40E-02**+	1.04E+00±9.30E-02+	3.38E+00±3.59E-01
CEC2014$_{13}$	4.69E+00±1.78E-01-	1.27E-01±2.24E-02-	2.86E-01±3.72E-02-	3.26E-01±4.98E-02-	4.39E-01±5.68E-02-	**6.31E-02**±**9.91E-03**
CEC2014$_{14}$	1.21E+02±1.07E+01-	4.23E-01±3.68E-02-	**2.65E-01**±**2.42E-02**+	2.87E-01±3.01E-02+	2.99E-01±3.57E-02+	3.57E-01±5.11E-02
CEC2014$_{15}$	2.59E+04±1.29E+04-	**5.33E+00**±**7.37E-01**+	1.48E+01±1.33E+00-	7.20E+00±8.22E-01+	1.69E+01±4.07E+00-	1.07E+01±2.25E+00
CEC2014$_{16}$	2.01E+01±5.42E-01+	2.01E+01±6.94E-01+	1.82E+01±4.73E-01+	**1.78E+01**±**3.92E-01**+	2.02E+01±3.17E-01+	2.15E+01±2.55E-01
CEC2014$_{17}$	6.84E+07±1.92E+07-	3.61E+02±1.82E+02-	5.97E+05±2.10E+05-	2.52E+03±5.52E+02-	7.60E+04±5.80E+04-	**2.39E+01**±**1.51E+01**
CEC2014$_{18}$	2.27E+09±5.30E+08-	2.71E+01±8.65E+00-	3.73E+02±3.68E+02-	1.76E+02±4.96E+01-	6.41E+02±4.98E+02-	**1.08E+00**±**7.96E-01**
CEC2014$_{19}$	1.90E+02±4.03E+01-	1.22E+01±1.24E+00-	2.16E+01±9.78E+00-	1.15E+01±3.26E+00-	1.54E+01±8.07E+00-	**6.51E+00**±**5.27E-01**
CEC2014$_{20}$	2.81E+03±1.56E+03-	5.72E+00±1.63E+00-	2.57E+02±1.41E+02-	6.73E+03±7.41E+03-	1.01E+03±1.01E+03-	**2.24E+00**±**3.59E-01**
CEC2014$_{21}$	7.73E+04±5.85E+04-	2.70E+02±1.14E+02-	3.80E+05±1.44E+05-	1.43E+03±3.76E+02-	5.57E+04±3.38E+04-	**1.48E+02**±**7.65E+01**
CEC2014$_{22}$	8.17E+02±3.63E+02-	6.74E+01±5.78E+01-	2.64E+02±1.30E+00-	4.74E+02±1.79E+02-	5.39E+02±1.95E+02-	**2.95E+01**±**6.27E-01**
CEC2014$_{23}$	5.42E+02±2.36E+01-	**3.44E+02**±**0.00E+00**≈	**3.44E+02**±**0.00E+00**≈	**3.44E+02**±**0.00E+00**≈	**3.44E+02**±**0.00E+00**≈	**3.44E+02**±**0.00E+00**
CEC2014$_{24}$	2.78E+02±1.76E+00-	2.72E+02±1.85E+00-	**2.58E+02**±**4.07E+00**+	2.74E+02±2.06E+00-	2.75E+02±4.80E+00-	2.66E+02±2.49E+00
CEC2014$_{25}$	2.20E+02±1.68E+00-	**2.04E+02**±**1.51E-01**≈	2.10E+02±7.36E-01-	2.16E+02±6.50E+00-	2.16E+02±9.81E+00-	**2.04E+02**±**1.55E-01**
CEC2014$_{26}$	1.75E+02±3.63E+01-	1.02E+02±1.19E+01-	1.47E+02±8.00E-01-	**1.00E+02**±**1.14E-01**≈	1.84E+02±3.73E+01-	**1.00E+02**±**1.24E-02**
CEC2014$_{27}$	1.19E+03±7.05E+01-	**3.12E+02**±**2.11E+01**+	3.24E+02±2.85E+01+	4.48E+02±7.00E+01-	7.67E+02±9.06E+01-	4.00E+02±5.23E+01
CEC2014$_{28}$	2.81E+03±5.55E+02-	1.23E+03±5.73E+01-	1.14E+03±4.20E+01+	**1.13E+03**±**5.48E+01**+	1.40E+03±9.63E+01-	1.18E+03±1.83E+02
CEC2014$_{29}$	1.42E+05±1.15E+05-	8.38E+02±1.06E+02-	1.36E+03±1.82E+02-	8.93E+02±7.49E+01-	1.61E+03±4.45E+02-	**7.36E+02**±**1.27E+07**
CEC2014$_{30}$	4.48E+05±2.03E+05-	**8.96E+03**±**6.14E+02**+	9.09E+03±3.05E+02+	9.59E+03±7.94E+02-	1.22E+04±1.91E+03-	9.14E+03±6.14E+02
No.of+/≈/-	**2/1/27**	**6/7/17**	**11/2/17**	**9/3/18**	**6/1/23**	**NA**

“+”, “-”, and “≈” indicate that the performance of the corresponding algorithm is better than, worse than, and similar to that of EMSM-EDA, respectively.



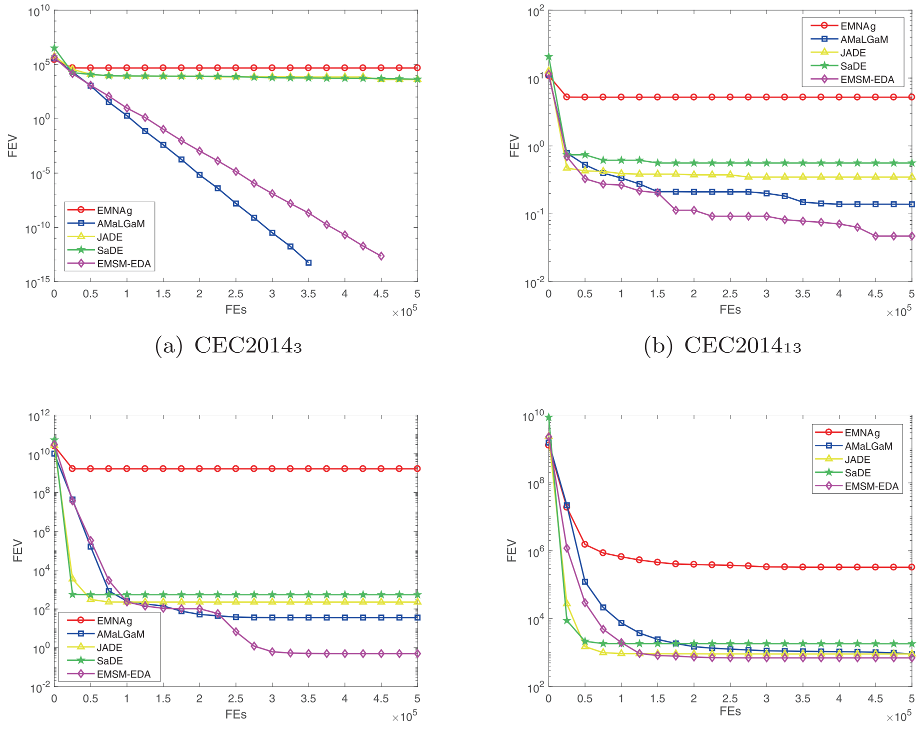
(a) CEC2014₁

(b) CEC2014₁₁

(c) CEC2014₂₁

(d) CEC2014₃₀

Fig. 12. Evolution of FEVs derived from EMNA$_g$, AMaLGaM, JADE, SaDE, and EMSM-EDA on 4 functions of CEC2014 benchmark functions with 30D.



(a) CEC2014₃

(b) CEC2014₁₃

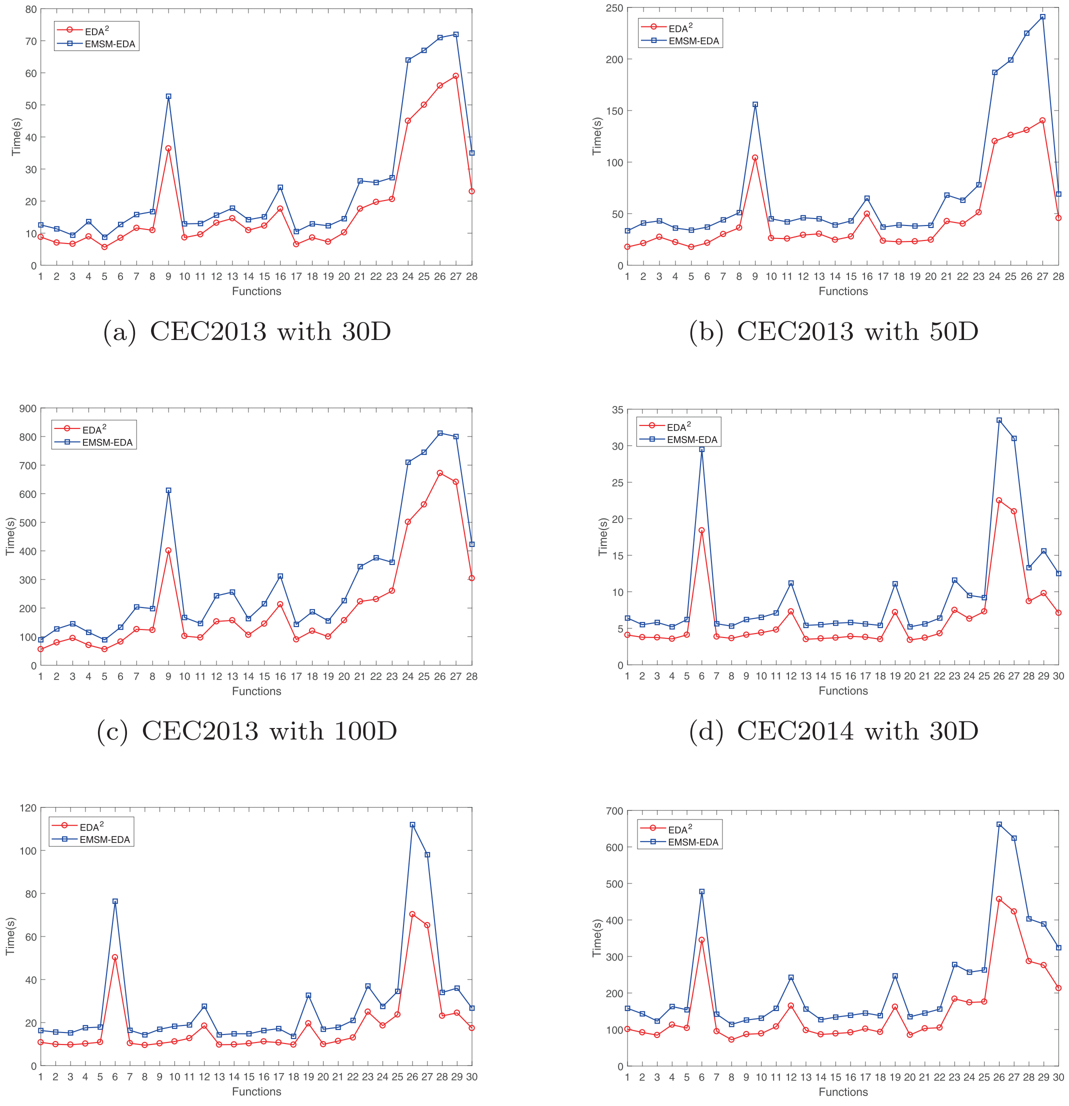
(c) CEC2014₁₈

(d) CEC2014₂₉

Fig. 13. Evolution of FEVs derived from EMNA$_g$, AMaLGaM, JADE, SaDE, and EMSM-EDA on 4 functions of CEC2014 benchmark functions with 50D.

Table 11
Rankings of EMSM-EDA, AMaLGaM, JADE, and SaDE on 58 test functions based on Friedman's test.

Algorithms	Ranking	
	30D	50D
EMSM-EDA	**2.0258**	**2.0172**
AMaLGaM	2.3276	2.2845
JADE	2.4397	2.4310
SaDE	3.2069	3.2672



(a) CEC2013 with 30D

(b) CEC2013 with 50D

(c) CEC2013 with 100D

(d) CEC2014 with 30D

(e) CEC2014 with 50D

(f) CEC2014 with 100D

Fig. 14. Runtime of EDA^2 and EMSM-EDA on 28 CEC2013 benchmark functions and 30 CEC2014 benchmark functions with 30D, 50D, and 100D.

From the last rows of Tables 9 and 10, it is obvious that EMSM-EDA outperforms EMNA$_g$, AMaLGAM, BL-PSO-5, JADE, and SaDE on IEEE CEC2014 test suite with 30D and 50D. To vividly depict the excellent performance of EMSM-EDA, Fig. 12 and Fig. 13 present the evolution curves of five comparison algorithms on 4 test functions with 30D and 50D, respectively. Compared with EMNA$_g$, AMaLGaM, JADE, and SaDE, EMSM-EDA has the better solutions and faster convergence speed.

Tables 7–10 summarize all the results that EMSM-EDA, AMaLGaM, JADE, and SaDE are competitive on IEEE CEC2013 and IEEE CEC2014 test suites. Furthermore, this paper conducts Friedman's test [45] on four excellent algorithms based on two test suites with 30D and 50D. Table 11 presents the test results that EMSM-EDA has the best ranking with 30D and 50D, sequentially followed by AMaLGaM, JADE, and SaDE.

In summary, EMSM-EDA performs better than five excellent EAs on two test suites with 30D and 50D. Specifically, EMSM-EDA is the champion on unimodal functions and hybrid functions. For the simple multimodal functions and the hybrid functions, the performance of EMSM-EDA is competitive.

5.7. Computational Complexity of EMSM-EDA

EMSM is designed to improve the performance of EDA^2, which causes extra computational burden. Thus, we study the computational cost of EMSM through the runtime of EDA^2 and EMSM-EDA on 28 CEC2013 benchmark functions and 30 CEC2014 benchmark functions with 30D, 50D, and 100D. Fig. 14 presents the runtime of EDA^2 and EMSM-EDA on CEC2013 and CEC2014 benchmark functions with 30D, 50D, and 100D, respectively. As shown in Fig. 14, the extra runtime of EMSM is the same order of magnitude as the runtime of corresponding EDA^2. Therefore, EMSM can significantly improve the performance of EDA^2 without too much additional computing burden.

5.8. Real-World Optimization Problems

Seven real-world engineering optimization problems in different fields such as radar system, power system, economic scheduling, and spacecraft trajectory optimization are selected to evaluate the performance of EMSM-EDA. These seven real-world optimization problems (called P_1-P_7 in this paper) are collected at the 2011 IEEE Congress on Evolutionary Computation [50], which show different engineering characteristics and are very difficult to solve. Specifically, the details of these seven real-world optimization problems are summarized in Table 12. In this experiment, EMNA$_g$, AMaLGaM, JADE, and SaDE are used as comparison algorithms to compare the performance of EMSM-EDA. In addition, the parameter settings of the five comparison algorithms are the same as IEEE CEC2013 test experiment. Finally, each algorithm is run 25 times, and 150,000 total fitness evaluation is used as the termination condition.

Table 13 summarizes the test results of EMNA$_g$, AMaLGaM, JADE, SaDE, and EMSM-EDA on seven real-world optimization problems. As shown in Table 13, EMSM-EDA surpasses EMNA$_g$, AMaLGaM, JADE, and SaDE on 7, 5, 6, and 6 problems, respectively. As a contrast, EMSM-EDA is defeated by EMNA$_g$, AMaLGaM, JADE, and SaDE on 0, 1, 0, and 0 problems, respectively. This indicates that EMSM-EDA can achieve the excellent performance on these real-world optimization problems. Furthermore, this paper conducts Friedman's test on five comparison algorithms based on seven real-world engineering optimization problems. Table 14 shows that EMSM-EDA has the best ranking, sequentially followed by AMaLGaM, SaDE, JADE, and EMNA$_g$. Therefore, the above results demonstrate the potential of EMSM-EDA on the real-world optimization problems.

Table 12
Seven real-world optimization problems.

Problems	Name of the problems
P_1	parameter estimation for frequency-modulated sound waves
P_2	spread spectrum radar polly phase code design
P_3	large scale transmission pricing problem
P_4	dynamic economic dispatch problem
P_5	static economic load dispatch problem
P_6	hydrothermal scheduling problem
P_7	messenger: spacecraft trajectory optimization problem

Table 13
The means and the standard deviations of FEVs obtained by the five algorithms over 25 independent runs on seven real-world optimization problems.

Problems	EMNA$_g$	AMaLGaM	JADE	SaDE	EMSM-EDA
P_1	7.83E-01±6.52E-01-	4.27E-01±7.69E-01-	4.63E-01±8.04E-01-	**0.00E+00±0.00E+00**≈	**0.00E+00±0.00E+00**
P_2	1.68E+00±2.75E-01-	1.14E+00±1.36E-01-	1.17E+00±1.00E-01-	1.09E+00±2.39E-01-	**4.78E-01±1.02E-01**
P_3	7.93E+03±6.82E+03-	3.48E+03±5.72E+02-	2.04E+03±4.89E+02-	6.48E+03±7.69E+03-	**1.68E+03±3.20E+02**
P_4	5.93E+04±5.68E+02-	5.13E+04±3.76E+02≈	5.14E+04±4.92E+02≈	5.61E+04±1.51E+04-	**5.11E+04±3.84E+02**
P_5	1.36E+05±6.56E+03-	1.35E+05±3.48E+03-	1.32E+05±5.09E+03-	1.32E+05±1.66E+03-	**1.24E+05±1.36E+03**
P_6	9.87E+05±1.38E+05-	9.36E+05±2.67E+03-	9.40E+05±3.47E+03-	9.76E+05±1.26E+05-	**9.14E+05±2.18E+03**
P_7	2.98E+01±6.73E+00-	**1.39E+01±1.25E+00**+	1.74E+01±2.12E+00-	1.56E+01±1.85E+00-	1.49E+01±1.63E+00
No.of+/≈/-	**0/0/7**	**1/1/5**	**0/1/6**	**0/1/6**	**NA**

"+", "-", and "≈" indicate that the performance of the corresponding algorithm is better than, worse than, and similar to that of EMSM-EDA, respectively.

Table 14
Rankings of EMSM-EDA, AMaLGaM, SaDE, JADE, and EMNA$_g$ on seven real-world optimization problems based on Friedman's test.

Algorithms	Ranking
EMSM-EDA	**1.2143**
AMaLGaM	2.5714
SaDE	3.0000
JADE	3.2143
EMNA$_g$	5.0000

6. Conclusion

In this paper, an efficient mixture sampling model named EMSM is presented. Different from the random sampling, EMSM maintains a good tradeoff between the diversity and the convergence by the uniform method, the orthogonal method, and the mirrored sampling. Moreover, EMSM-EDA is developed, which combines EMSM with EDA^2. In addition, the diversity and convergence of EMSM are analyzed. Finally, the experimental results indicate that EMSM-EDA is a global optimization algorithm with excellent performance.

Although EMSM-EDA has achieved competitive performance, there are still several issues worth investigating in the future. Firstly, we will further study new sampling methods to explore promising regions. For example, the Markov Chain Monte Carlo method can be used to adaptively select the promising mutation vectors according to the historical performance, so as to obtain better solutions. Secondly, the computational complexity of EMSM-EDA is $O(n^3)$, which can lead to longer runtime in solving large-scale problems. The calculation cost of EMSM-EDA mainly depends on the calculation and decomposition of covariance matrix. It is very interesting to try to use the new matrix decomposition method to improve the computational efficiency. Finally, it is meaningful to extend EMSM-EDA to multimodal optimization problems, dynamic optimization problems, constrained optimization problems, and large-scale optimization problems.

CRediT authorship contribution statement

Qianlong Dang: Conceptualization, Methodology, Software, Writing - original draft, Visualization. **Weifeng Gao:** Methodology, Supervision, Funding acquisition. **Maoguo Gong:** Validation, Writing - review & editing, Software.

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.

Acknowledgment

This work was supported in part by the National Nature Science Foundation of China under Grant 61772391 and Grant 61966030, in part by the Natural Science Basic Research Plan in Shaanxi Province of China under Grant 2022JQ-670, in part by the Fundamental Research Funds for the Central Universities under Grant YJS2215.

References

[1] L. Luo, X. Hou, J. Zhong, W. Cai, J. Ma, Sampling-based adaptive bounding evolutionary algorithm for continuous optimization problems, Information Sciences 382 (2017) 216–233.

[2] L.B. Deng, L.L. Zhang, N. Fu, H.L. Sun, L.Y. Qiao, ERG-DE: An elites regeneration framework for differential evolution, Information Sciences 593 (2020) 81–103.

[3] M. Dorigo, V. Maniezzo, A. Colorni, Ant system: optimization by a colony of cooperating agents, IEEE Transactions on Systems, Man, and Cybernetics 26 (1) (1996) 29–41.

[4] R. Storn, K. Price, Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces, Journal of Global Optimization 11 (4) (1997) 341–359.

[5] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: Proceedings of the Sixth International Symposium on Micro Machine and Human Science, IEEE, 1995, pp. 39–43.

[6] P. Bosman, J. Grahl, D. Thierens, Enhancing the performance of maximum-likelihood Gaussian edas using anticipated mean shift, in: Parallel Problem Solving from Nature, Springer, 2008, pp. 133–143.

[7] M. Jiang, L. Qiu, Z. Huang, G.G. Yen, Dynamic multi-objective estimation of distribution algorithm based on domain adaptation and nonparametric estimation, Information Sciences 435 (2018) 203–223.

[8] P. Larranaga, J.A. Lozano, Estimation of distribution algorithms: a new tool for evolutionary computation, Boston, MA, Kluwer Academic, USA, 2001.

[9] K. Liu, D. Gong, F. Meng, H. Chen, G.G. Wang, Gesture segmentation based on a two-phase estimation of distribution algorithm, Information Sciences 394 (2017) 88–105.

[10] S. Muelas, A. Mendiburu, A. LaTorre, J.M. Pena, Distributed estimation of distribution algorithms for continuous optimization: how does the exchanged information influence their behavior?, Information Sciences 268 (2014) 231–254

[11] J.A. Lozano, P. Larranga, I. Inza, E. Bengoetxea, Towards a new evolutionary computation: advances in estimation of distribution algorithms, Springer, 2006.

[12] B. Jarboui, M. Eddaly, P. Siarry, An estimation of distribution algorithm for minimizing the total flowtime in permutation flowshop scheduling problems, Computers and Operations Research 36 (9) (2009) 2638–2646.

[13] T. Friedrich, F. Neumann, Maximizing submodular functions under matroid constraints by evolutionary algorithms, Evolutionary Computation 23 (4) (2015) 543–558.

[14] J. Ceberio, E. Irurozqui, A. Mendiburu, J.A. Lozano, A distance-based ranking model estimation of distribution algorithm for the flow-shop scheduling problem, IEEE Transactions on Evolutionary Computation 18 (2) (2014) 286–300.

[15] X. Liang, H. Chen, J.A. Lozano, A boltzmann-based estimation of distribution algorithm for a general resource scheduling model, IEEE Transactions on Evolutionary Computation 19 (6) (2015) 793–806.

[16] M. Pelikan, D.E. Goldberg, F.G. Lobo, A survey of optimization by building and using probabilistic models, Computational Optimization and Applications 21 (1) (2002) 5–20.

[17] M. Pelikan, Scalable optimization via probabilistic modeling: from algorithms to applications, Studies, Computational Intelligence (2006).

[18] S. Zhou, Z. Sun, A survey on estimation of distribution algorithms, Acta Automatica Sinica 32 (2) (2007) 113–124.

[19] A. Zhou, J. Sun, Q. Zhang, An estimation of distribution algorithm with cheap and expensive local search methods, IEEE Transactions on Evolutionary Computation 19 (6) (2015) 807–822.

[20] W. Shi, W.N. Chen, Y. Lin, T. Gu, S. Kwong, J. Zhang, An adaptive estimation of distribution algorithm for multi-policy insurance investment planning, IEEE Transactions on Evolutionary Computation 23 (1) (2019) 1–14.

[21] F. Wang, Y. Li, A. Zhou, K. Tang, An estimation of distribution algorithm for mixed-variable newsvendor problems, IEEE Transactions on Evolutionary Computation (2019).

[22] N. Luo, F. Qian, Estimation of distribution algorithm sampling under gaussian and cauchy distribution in continuous domain, in: in: International Conference on Control and Automation, IEEE, 2010, pp. 1716–1720.

[23] J. Velasco, M.A. Saucedo-Espinosa, H.J. Escalante, An estimation of distribution algorithm with adaptive Gibbs sampling for unconstrained global optimization, Mathematics (2011).

[24] Z. Ren, C. He, D. Zhong, S. Huang, Y. Liang, Enhance continuous estimation of distribution algorithm by variance enlargement and reflecting sampling, in: Congress on Evolutionary Computation, 2016, pp. 3441–3447.

[25] Z.Y. Lin, G.B. Xie, L.J. Wang, Mutual information estimation of distribution algorithm with hybrid sampling mechanism, Control and Decision 32 (5) (2017) 829–835.

[26] Y. Liang, Z. Ren, X. Yao, Z. Feng, A. Chen, W. Guo, Enhancing Gaussian estimation of distribution algorithm by exploiting evolution direction with archive, IEEE Transactions on Cybernetics 50 (1) (2018) 140–152.

[27] H. Muhlenbein, G. Paab, From recombination of genes to the estimation of distributions I. Binary parameters, in: Parallel Problem Solving from Nature, Berlin, Germany, 1996, pp. 178–187.

[28] R.C. Zhang, Z.J. Wang, Uniform design sampling and its fine properties, Chinese Journal of Applied Probability and Statistics 12 (4) (1996) 133–143.

[29] B.D. Zhou, H.L. Yao, M.H. Shi, Q. Yue, H. Wang, An new immune genetic algorithm based on uniform design sampling, Knowledge and Information Systems 31 (2) (2012) 389–403.

[30] C. Huang, Y. Lee, D. Lin, S. Huang, Model selection for support vector machines via uniform design, Computational Statistics and Data Analysis 52 (1) (2007) 335–346.

[31] W. Gao, S. Liu, L. Huang, A novel artificial bee colony algorithm based on modified search equation and orthogonal learning, IEEE Transactions on Cybernetics 43 (3) (2013) 1011–1024.

[32] Z. Zhan, J. Zhang, Y. Li, Y. Shi, Orthogonal learning particle swarm optimization, IEEE Transactions on Evolutionary Computation 15 (6) (2011) 832–847.

[33] I. Loshchilov, M. Schoenauer, M. Sebag, Adaptive coordinate descent, Genetic and Evolutionary Computation Conference (2011) 885–892.

[34] D. Brockhoff, A. Auger, N. Hansen, D.V. Arnold, T. Hohm, Mirrored sampling and sequential selection for evolution strategies, in: International Conference on Parallel Problem Solving from Nature: Part I, Springer, Berlin, Heidelberg, 2010, pp. 11–21.

[35] A. Auger, D. Brockhoff, N. Hansen, Mirrored sampling in evolution strategies with weighted recombination, in: Conference on Genetic and Evolutionary Computation, ACM, 2011.

[36] H. Wang, M. Emmerich, T. Back, Mirrored orthogonal sampling with pairwise selection in evolution strategies, in: Acm Symposium on Applied Computing, 2014, pp. 154–156.

[37] A. Auger, D. Brockhoff, N. Hansen, Analyzing the impact of mirrored sampling and sequential selection in elitist evolution strategies, in: Foundations of Genetic Algorithms, 11th International Workshop, ACM, Schwarzenberg, Austria, 2011, pp. 861–868.

[38] H.G. Beyer, The theory of evolution strategies, Natural Computing Series (2001).

[39] H. Wang, M. Emmerich, T. back, Mirrored orthogonal sampling for covariance matrix adaptation evolution strategies, Evolutionary Computation 27 (4) (2019) 699–725.

[40] T. Back, Order statistics for convergence velocity analysis of simplified evolutionary algorithms, Foundations of Genetic Algorithms (1995) 91–102.

[41] Z.G. Ren, Y.S. Liang, A.M. Zhang, B. Pang, A Gaussian estimation of distribution algorithm using general second-order mixed moment, Acta Automatica Sinica 44 (4) (2018) 635–645.

[42] Q.X. Yin, The inverse problem of rank-1 modification of real symmetric matricesA, Journal of Nantong Institute of Technology 2 (4) (2003) 11–14.

[43] J. Liang, B. Qu, P. Suganthan, A. Hernandez-Daz, Problem definitions and evaluation criteria for the cec 2013 special session on real-parameter optimization, Compututatinal. Intelligence. Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore, 2013.

[44] J. Liang, B. Qu, P. Suganthan, Problem definitions and evaluation criteria for the cec 2014 special session and competition on single objective real-parameter numerical optimization, Compututatinal. Intelligence. Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore, 2013.

[45] J. Alcal-Fdez, L. Snchez, S. Garca, M.J.D. Jesus, S. Ventura, J.M. Garrellz, J. Otero, C. Romero, J. Bacardit, V.M. Rivas, KEEL: a software tool to assess evolutionary algorithms for data mining problems, Soft Computing 13 (3) (2009) 307–318.

[46] J. Zhang, A.C. Sanderson, JADE: Adaptive differential evolution with optional external archive, IEEE Transactions on Evolutionary Computation 13 (5) (2009) 945–958.

[47] A.K. Qin, V.L. Huang, P.N. Suganthan, Differential evolution algorithm with strategy adaptation for global numerical optimization, IEEE Transactions on Evolutionary Computation 13 (2) (2009) 398–417.

[48] J. Cohen, Statistical Power Analysis for the Behavioral Sciences, Lawrence Erlbaum Assoc, Hillsdale, NJ, USA, 1988, pp. 20–26.

[49] X. Chen, H. Tianfield, C. Mei, W. Du, G. Liu, Biogeography-based learning particle swarm optimization, Soft Computing 21 (24) (2017) 7519–7541.

[50] S. Das, P.N. Suganthan, Problem definitions and evaluation criteria for CEC, competition on testing evolutionary algorithms on real world optimization problems, Jadavpur University, Nanyang Technological University, Kolkata 2010 (2011) 341–359.