



An efficient estimation of distribution algorithm with rank-one modification and population reduction

Yongsheng Liang^a, Zhigang Ren^{a,*}, Miao He^b, Lin Wang^c, Jianfu Cao^d, An Chen^a, Bei Pang^a

^a Department of Automation Science and Technology, School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an, PR China

^b Xi'an Aircraft Certification Center, Airworthiness Certification Center of CAAC, Xi'an, PR China

^c School of Information Science and Technology, Northwest University, Xi'an, PR China

^d Guangdong Xi'an Jiaotong University Academy, PR China

ARTICLE INFO

Keywords:

Estimation of distribution algorithms
Premature convergence
Rank-one modification
population reduction

ABSTRACT

As a type of model-based metaheuristic, estimation of distribution algorithms (EDAs) show certain advantages over other metaheuristics by using statistical learning method to estimate the distribution of promising solutions. However, the commonly-used Gaussian EDAs (GEDAs) usually suffer from premature convergence that severely limits their efficiency. In this paper, we first attempt to enhance the performance of GEDA by improving its model estimation method. The new estimation method shifts the weighted mean of high-quality solutions towards the fitness improvement direction and estimates the covariance matrix by taking the shifted mean as the center. Theoretical analyses show that the new covariance matrix is essentially a rank-one modification (R1M) of the original one. It could effectively adjust both the search scope and the search direction of GEDA, and thus improving the search efficiency. Furthermore, considering the importance of the population size tuning in GEDA, we develop a population reduction (PR) strategy which linearly reduces the population size throughout the evolution. By this means, the exploration and exploitation ability of GEDA could be balanced better in different search stages and a more proper utilization of limited computation resource can be achieved. Combining GEDA with the R1M and PR strategies, a novel EDA variant named EDA-R1M-PR is developed. The performance of EDA-R1M-PR was comprehensively evaluated and compared with that of several state-of-the-art evolutionary algorithms. Experimental results indicate that the R1M and PR strategies effectively enhance the global optimization ability of GEDA and the resultant EDA-R1M-PR significantly outperforms its competitors on a set of benchmark functions.

1. Introduction

In recent decades, a great number of bio-inspired evolutionary algorithms (EAs) have been developed and applied to solve various kinds of optimization problems. Estimation of distribution algorithm (EDA) (Mühlenbein and Paaß, 1996; Larrañaga and Lozano, 2001) can be considered as a special kind of EA that merges a genetic algorithm (GA) with statistical learning method. Different from GA which employs the mutation and crossover operators to generate offspring, EDA explicitly builds a probabilistic model based on some promising solutions and produces offspring by sampling the learned model. Benefiting from the learnable probabilistic model, EDA is endowed with special advantages over other algorithms and is capable of solving a broad array of problems in both discrete and continuous domains (Hauschild and Pelikan, 2011; Chen et al., 2017a; Gao and de Silva, 2018). This study mainly

focuses on EDAs for continuous optimization.

The probabilistic model plays a crucial role in EDA. By far, many kinds of probabilistic models such as Gaussian model (Ren et al., 2018), Cauchy model (Sanyang and Kaban, 2014) and Histogram model (Ding et al., 2008) have been integrated into EDAs, among which the Gaussian model is most widely adopted by continuous EDAs. According to the variable dependencies, Gaussian EDAs (GEDAs) can be further classified into the following three categories. 1) Univariate GEDA which assumes all variables are independent. Examples of this category are univariate marginal distribution algorithm in continuous domain (UMDA) (Larrañaga et al., 1999) and iterated density-estimation evolutionary algorithm (IDEA) (Bosman and Thierens, 2000). 2) Bivariate GEDA which only takes some pairwise variable interactions into account. Representative algorithm of this class is mutual information maximization and input clustering (MIMIC) (De Bonet et al., 1997). 3)

* Corresponding author at: Department of Automation Science and Technology, School of Electronic and Information Engineering, Xi'an Jiaotong University, No.28 Xianning West Road, Xi'an, Shaanxi, 710049, PR China.

E-mail address: renzg@mail.xjtu.edu.cn (Z. Ren).

<https://doi.org/10.1016/j.biosystems.2019.04.001>

Received 31 October 2018; Received in revised form 12 March 2019; Accepted 4 April 2019

Available online 23 April 2019

0303-2647/ © 2019 Elsevier B.V. All rights reserved.

Multivariate GEDA considering interactions among multiple variables. Bayesian optimization algorithm (BOA) (Pelikan et al., 1999) and estimation of multivariate normal density algorithm (EMNA_g) (Larrañaga and Lozano, 2001) belong to this class.

The model-based approach enables EDA to capture the important characteristics of promising solutions and thus facilitate it converging to the optimum. However, the traditional GEDA usually suffers from a tendency to prematurely converge to locally optimal solutions. It was revealed both experimentally and theoretically that a major cause for this drawback is the fast-decreasing of variable variances. To address this issue, Yuan and Gallagher (2005) suggested maintaining all the variances greater than 1. Grahl et al. (2006) proposed an adaptive variance scaling strategy (AVS) which dynamically enlarges or reduces the variances according to the state of the evolution. It is notable that AVS would only be activated when the algorithm is searching on a slope-like solution region and the techniques of correlation triggering rule (Grahl et al., 2006) and standard deviation ratio (SDR) (Bosman et al., 2007) were developed to identify such structure. Besides directly adjusting variances, Handa (2007) introduced a mutation operator into EDA to maintain the population diversity and alleviate premature convergence. Gao and Silva (Gao and de Silva, 2016) designed an extreme elitism selection method to enrich the population diversity by highlighting the roles of some other elite solutions besides the best one.

Since the performance of EDA is highly dependent on its probabilistic model, the model learning method becomes a key element. GEDA usually employs the maximum-likelihood (ML) estimation method to estimate its Gaussian model based on some high-quality solutions. Unfortunately, it was shown that the main search direction of the traditional GEDA tends to become orthogonal to the fitness improvement direction on slope-like regions. This defect was first noted by Cai et al. (2007), and they made the first remedy with a novel variance scaling strategy. To obtain a reasonable variance scaling factor, they tried to measure the divergence between the distribution of the selected solutions in current generation and the distribution of the population in next generation, and then calculate the scaling factor by minimizing the cross entropy between the two distributions. Although this remedy takes effect on some functions, its performance was reported not to be as competitive as AVS (Bosman et al., 2008). Ren et al. (2018) suggested an anisotropic adaptive variance scaling (AAVS) strategy that was shown to be able to deal with the above problems more effectively. Different with other scaling strategies that adjust all the variances with a same factor, AAVS anisotropically tunes the variances along different eigendirections according to the learned structure characteristics. In this way, AAVS is capable of adjusting both the search scope and direction of GEDA simultaneously, and thus greatly enhances its performance. Besides tuning the variances of the Gaussian model learned in traditional way, some other researchers attempted to directly ameliorate the model learning method. CMA-ES (Hansen and Ostermeier, 2001), which can be viewed as a special GEDA, employs the weighted good solutions in current generation and the mean of last generation to update its covariance matrix. The modified estimation method effectively increases the variances in the evolution direction, which endows CMA-ES with better search directions. (Bosman et al., 2008) developed an anticipated mean shift (AMS) strategy which shifts some solutions along the direction of mean shift. After that, these shifted and some other unshifted solutions in current generation as well as the elitist solutions in previous generations are all used to estimate the covariance matrix. Combining AMS with the aforementioned AVS and SDR, they proposed an efficient EDA variant named adapted maximum-likelihood Gaussian model-iterated density-estimation evolutionary algorithm (AMaLGaM) (Bosman et al., 2008). Experimental results indicated that AMaLGaM could obtain better search direction and search scope than traditional GEDA during the evolution and is very competitive with the famous CMA-ES.

In addition to overcoming the shortcomings, many other techniques were also introduced into EDA to improve its performance. Zhou et al.

(2015) combined EDA with cheap and expensive local search methods to improve its exploitation ability. Li et al. (2011) designed a subspace EDA to enhance the model learning and sampling efficiency by implementing the two operations in a selected subspace of all variables at each generation. For accelerating the search process, Chen et al. (2017a) proposed a fast interactive EDA which takes advantage of the domain knowledge of personalized search to reduce the initial search space to a preferred subspace. Yang et al. (2017) combined EDA with clustering strategy to strengthen its performance on multimodal problems. EDA was also scaled up to solve high-dimensional problems by taking advantage of the random matrix theory (Kabán et al., 2016). Furthermore, EDAs have been merged with some other kinds of EAs such as particle swarm optimization (PSO) (Ahn et al., 2012), differential evolution (DE) (Zhao et al., 2016) and artificial bee colony algorithm (ABC) (Xu et al., 2015) to develop hybrid algorithms that perform better than the original ones. There are also a variety of EDAs that were specially developed for different kinds of applications. Shi et al. (2019) proposed an adaptive EDA to derive the optimal plan of multi-policy insurance investment. Wu and Wang (2018) employed a multi-model EDA to deal with the energy efficient scheduling problem in cloud computing system.

Despite so many improvement works on the models of EDAs, fewer theoretical studies were conducted on EDAs. Grahl et al. (2005) studied the convergence behaviour of UMDA_c with truncation selection on monotonous functions, and showed that the algorithm could only travel a limited distance in the solution space and the distance solely relies on the selection pressure. Chen et al. (2010) investigated the computational time complexity of UMDA_c with respect to the population size on two typical unimodal functions. Lima et al. (2011) analyzed the relationship between the probabilistic models learned by BOA and the underlying problem structure to improve the model accuracy. Rastegar (2010) studied the convergence probability of two simple univariate EDAs and showed a sufficient condition for the convergence of the two algorithms on a specific class of functions.

Through the above related works, it can be seen that EDA has received considerable research effort in the past decades and its efficiency has been enhanced significantly from different perspectives. However, along with the advantages of these improved EDAs, there are still some shortcomings. First, some of the EDA variants, such as the famous CMA-ES and AMaLGaM, are accompanied by complicated algorithm framework and many free parameters, which may affect their robustness and limit their applicability. Second, most of the EDAs are improved by using different heuristic strategies and their performances are usually verified through specified experiments, but they generally lack theoretical analyses. The existing theoretical studies mainly focus on analyzing the characteristics of basic EDAs. They are of significance to reveal the properties of EDAs, but may not have strong practicability to improve their search efficiency. Third, most existing EDAs still adopt a fixed population size during the whole evolution, which leaves room for improvement. As a key element, the population size has a fundamental influence on the performance of EDA, especially its convergence rate. A larger population size encourages wider exploration of the solution space but reduces the convergence speed. On the other hand, a smaller population size could quicken the convergence, but may increase the risk of falling into local optimal regions. In different search stages, the population size should be dynamically adjusted to balance the exploration and exploitation ability of the algorithm. To achieve better adjustment of the population sizes of EAs, several adaptive population resizing methods have been successfully developed for algorithms like GA (Lobo and Lima, 2007), CMA-ES (Auger and Hansen, 2005), PSO (de Oca et al., 2011) and DE (Tanabe and Fukunaga, 2014). So it is of significance to develop an adaptive population resizing strategy for EDA to further enhance its performance.

To address the above issues, we proposed a novel GEDA with Rank-One Modification and Population Reduction strategies (EDA-R1M-PR). Theoretical and experimental studies are both conducted to prove and

verify the effectiveness of the proposed strategies and the resultant algorithm. The main contributions of this work are summarized as follows:

- (1) Improve the model estimation method of GEDA with a simple and efficient mean shift strategy. This strategy directly shifts the weighted mean of selected solutions towards more promising solution regions, then estimates the covariance matrix of Gaussian model by taking the shifted mean as the center. The operation not only keeps the simplicity of ML estimation but also significantly enhances the search ability of GEDA.
- (2) Theoretically show that the essence of the improved covariance matrix is rank-one modification (R1M) of the original one. Taking advantage of the analytical properties of Gaussian model, we further prove that the R1M operation could effectively adjust the search scope and main search direction of GEDA simultaneously, and thus greatly improve its performance.
- (3) Design a population reduction strategy for GEDA, which linearly reduces the population size at each generation throughout the evolution. In the earlier search stage, a larger population size is capable of locating promising solution regions more precisely; while in the latter search stage, a relatively smaller population size is beneficial to improve the convergence speed thus facilitating the algorithm finding the optimal solution. In this way, the exploration and exploitation ability of GEDA could be balanced better in different search stages and a more proper utilization of computation resource can be achieved.

The rest of this paper is organized as follows. Section 2 describes the basic knowledge of GEDA. Section 3 presents the proposed EDA-R1M-PR algorithm in detail. Section 4 provides the experimental results and analyses. The conclusions are finally given in Section 5 along with future work.

2. Basic GEDA

First proposed by Mühlenbein and Paaß (1996), EDA inherits the basic framework of GA, but employs a probabilistic model to extract statistical information of good solutions and produces new solutions by sampling the learned model. The general framework of EDA is outlined in Algorithm 1. After initialization, EDA executes an iterative process of evaluation (step 3), selection (step 4), model building (step 5) and sampling (step 6) until a predefined stopping criterion is met.

Algorithm 1: General framework of EDA

- Initialize parameters, set $t = 0$, and generate the initial population P^t ;
- **while** the stopping criterion is not met **do**
- Evaluate population P^t and update the best solution b obtained so far;
- Select promising solutions S^t from P^t according to a selection rule;
- Build a new probability model G^t based on S^t ;
- Generate a new population P^{t+1} by sampling from G^t and update $t = t + 1$;
- **end**
- **return** b .

Most continuous EDAs employ Gaussian model to describe the distribution of good solutions. The probability density function of Gaussian model can be presented by the following notation:

$$G_{(\mu, C)}(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}(\det C)^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\mu)^T C^{-1}(\mathbf{x}-\mu)} \quad (1)$$

where \mathbf{x} is an n -dimensional variable vector, μ and C are the mean and covariance matrix (CM) of \mathbf{x} , respectively. To establish this model, we need to estimate μ and C from some selected solutions. Maximum-likelihood (ML) estimation method is a commonly adopted approach. For a given set of selected solutions S^t , the ML estimates for μ and C are given by the following formulas respectively:

$$\bar{\mu}^t = \frac{1}{|S^t|} \sum_{i=1}^{|S^t|} S_i^t \quad (2)$$

$$\bar{C}^t = \frac{1}{|S^t|} \sum_{i=1}^{|S^t|} (S_i^t - \bar{\mu}^t)(S_i^t - \bar{\mu}^t)^T \quad (3)$$

in which $|\cdot|$ represents the cardinality of a set.

The Gaussian model estimated with Eqs. (2) and (3) uses a full CM and is adopted by many GEDAs like EMNAG. This model keeps all the variable dependencies that it could ensure rotation-invariance and usually performs well on problems with significant dependencies. While a limitation of this model is that it requires a large number of solutions to properly estimate the CM since there are up to $0.5(n^2 + n)$ parameters in it. Existing GEDAs using full CM generally adopt a constant and relatively large population size throughout the evolution, which may limit their search efficiency. This is because, on the one hand, a constant and large population size reduces the number of iterations of GEDA if only limited computation resource is available. On the other hand, the population size has great influence on the convergence behavior of GEDA, a constant value for it means that GEDA cannot well balance the exploration and exploitation ability in different search stages. Thus, the population size setting method in GEDA leaves room for improvement.

Furthermore, as mentioned above that traditional GEDA would generally suffer from two drawbacks: 1) variable variances decrease fast; 2) the main search direction of GEDA tends to become orthogonal to improvement direction of the fitness function. Fig. 1 explains the reasons for the two drawbacks by presenting the evolution of the probability density ellipsoid (PDE) of GEDA in two successive generations on a slope-like region. PDE is determined by the mean and CM of the Gaussian model and could reflect the search center and search range of GEDA. The major axis direction of PDE presents the main search direction of GEDA. As shown in Fig. 1, the new PDE obtained by ML estimation describes the distribution of selected solutions well. However, it is not the actual distribution but the direction of improvement that is of significance in order to find better solutions on slope-like regions. Due to the selection pressure and the property of Gaussian model that it generally produces more solutions near its mean, the new PDE estimated by these selected solutions will generally shrink and its

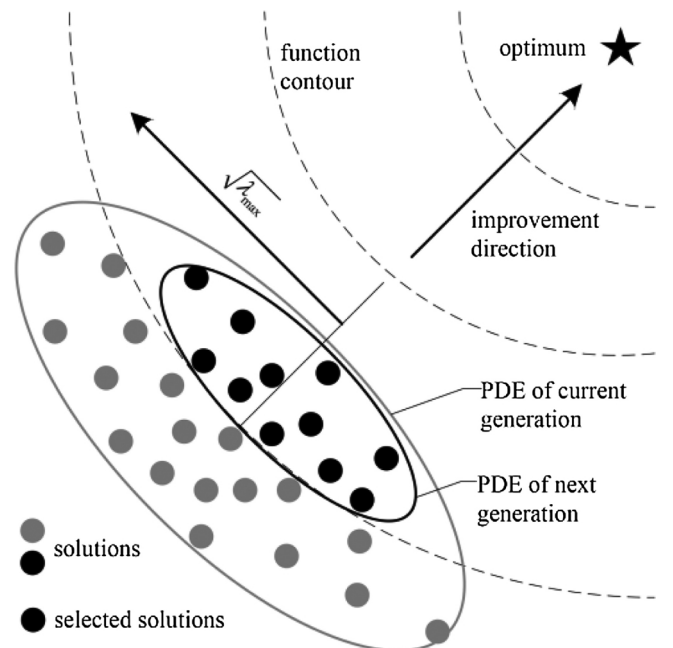


Fig. 1. The evolution of PDE in traditional GEDA.

major axis is inclined to become perpendicular to the fitness improvement direction. As a result, GEDA is unlikely to obtain fine search scope and search direction in this situation.

3. GEDA with rank one modification and population reduction

As shown in Fig. 1, the search center, search scope and main search direction of GEDA are determined by the center, shape and major axis direction of the PDE, respectively. It is hoped that the PDE could be located in promising solution regions, its shape could automatically adapt to the structural characteristics of the solution space, and the direction of its major axis can be parallel to the desired fitness improvement direction. To achieve the above goals, we first tried to improve the estimation method for the mean and CM with a mean shift strategy and showed that the essence of the improved CM is the rank-one modification (R1M) of the original one. Then the effectiveness of the R1M operation was theoretically proved. Moreover, we designed a population reduction (PR) strategy for the proposed GEDA with R1M (EDA-R1M) to further enhance its performance, and the resultant algorithm was finally named as EDA-R1M-PR.

3.1. Mean estimation method

The traditional ML estimator shown in Eq. (2) calculates the mean as the average of selected solutions. However, the goal of estimating the mean is not to calculate the real center of selected solutions, but to obtain a more promising search center for GEDA. Thus, we estimate the mean as the weighted average of selected solutions:

$$\bar{\mu}^t = \frac{\sum_{i=1}^{|S^t|} [\log(|S^t| + 1) - \log(i)] S_{(i)}^t}{\sum_{i=1}^{|S^t|} [\log(|S^t| + 1) - \log(i)]} \quad (4)$$

where $S_{(i)}^t$ denotes the i th best solution in the selected solution set S^t . It can be seen from Eq. (4) that better solutions are endowed with larger weights, then the estimated mean $\bar{\mu}^t$ will be pulled towards relatively high-quality solutions to get a better search center. Our preliminary experiments showed that $\bar{\mu}^t$ usually performs better than the original $\bar{\mu}^t$.

To further accelerate the search process, we try to shift $\bar{\mu}^t$ towards the evolution direction as follows:

$$\tilde{\delta}^t = \bar{\mu}^t - \bar{\mu}^{t-1} \quad (5)$$

$$\hat{\mu}^t = \begin{cases} \bar{\mu}^t + \eta \tilde{\delta}^t, & \text{if } f(\bar{\mu}^t + \eta \tilde{\delta}^t) < f(\bar{\mu}^t) \\ \bar{\mu}^t, & \text{otherwise} \end{cases} \quad (6)$$

where $\hat{\mu}^t$ represents the shifted mean in the t th generation, it is the real mean used for sampling new solutions. $\tilde{\delta}^t$ is the difference between $\bar{\mu}^t$ and $\bar{\mu}^{t-1}$, it reflects the evolution direction. η is a shifting factor which should be greater than 0 and $f(\cdot)$ is the objective function to be minimized. Eq. (6) attempts to update $\bar{\mu}^t$ with $\bar{\mu}^t + \eta \tilde{\delta}^t$ providing that $f(\bar{\mu}^t + \eta \tilde{\delta}^t)$ is smaller than $f(\bar{\mu}^t)$. The purpose of this operation is to exploit the inertia of the evolution direction thus speeding up the search process. A proper value for η could help GEDA finding a more promising search center for the next generation. It is however non-trivial to set the value of η , especially for black box optimization problems. Considering that the purpose of shifting the mean is to get a relatively better search center, rather than finding the optimal value for the mean. In addition, the mean shift operation would also consume some fitness evaluations. Keeping the two principles in mind, we designed a simple line search method to roughly find a proper value for the mean. As shown in Algorithm 2, the line search method tries to shift the mean along the evolution direction step by step until it is unable to find a better value for $\hat{\mu}^t$ or the maximum step size η_{\max} is reached. Note that η_{\max} should not be set too large for sake of saving computation resource and

avoiding disturbing the algorithm too much. According to our preliminary experimental results, we suggest setting $\eta_{\max} = 5$.

Algorithm 2: Line search method

Input: $\bar{\mu}^t$, $\tilde{\delta}^t$ and η_{\max}
Output: $\hat{\mu}^t$
1 $\hat{\mu}^t = \bar{\mu}^t$, $\eta = 0$;
2 **while** $f(\bar{\mu}^t + \tilde{\delta}^t) < f(\bar{\mu}^t)$ && $\eta < \eta_{\max}$ **do**
3 $\hat{\mu}^t = \bar{\mu}^t + \tilde{\delta}^t$;
4 $\eta = \eta + 1$;
5 **end**

3.2. Covariance matrix estimation method

CM of Gaussian model affects the search scope and search direction of GEDA. However, the CM estimated by the traditional ML estimator may result in the PDE aligned in the worst way. Considering that the CM of a random vector equals its second-order central moment, a convenient way to re-align PDE is to change the reference mean. Since we have obtained a more promising mean $\hat{\mu}^t$ through the above mean shift operation, it is reasonable to improve the CM estimation method by replacing the mean $\bar{\mu}^t$ with $\hat{\mu}^t$:

$$\hat{C}^t = \frac{1}{|S^t|} \sum_{i=1}^{|S^t|} (S_i^t - \hat{\mu}^t)(S_i^t - \hat{\mu}^t)^T \quad (7)$$

Compared with the CM obtained with traditional ML estimator, the PDE of \hat{C}^t will be naturally elongated along the direction of $(\hat{\mu}^t - \bar{\mu}^t)$ and the estimated variable variances would also be increased providing that $\hat{\mu}^t \neq \bar{\mu}^t$. In the following, we will show that \hat{C}^t is essentially the rank-one modification of the original \bar{C}^t . Then the properties of \hat{C}^t will be discussed. It will be shown that the new estimation method for CM could adaptively adjust the search scope and search direction of GEDA and thus improve its search efficiency.

For the convenience of derivation, we omit “ t ” in the variables and equations below. For each generation, it can be known from Eqs. (4)–(6) that $\hat{\mu}$ is not worse than $\bar{\mu}$ and $\bar{\mu}$ is usually better than $\bar{\mu}$, then $\hat{\mu}$ would generally be better than $\bar{\mu}$ with high probability. Then, $\tilde{\delta} = \bar{\mu} - \bar{\mu}$ represents a fitness improvement direction. We can transform Eq. (7) as follows:

$$\begin{aligned} \hat{C} &= \frac{1}{|S|} \sum_{i=1}^{|S|} (S_i - \hat{\mu})(S_i - \hat{\mu})^T \\ &= \frac{1}{|S|} \sum_{i=1}^{|S|} (S_i - \bar{\mu} - \tilde{\delta})(S_i - \bar{\mu} - \tilde{\delta})^T \\ &= \frac{1}{|S|} \sum_{i=1}^{|S|} (S_i - \bar{\mu})(S_i - \bar{\mu})^T - \frac{1}{|S|} \sum_{i=1}^{|S|} (S_i - \bar{\mu})\tilde{\delta}^T \\ &\quad - \frac{1}{|S|} \tilde{\delta} \sum_{i=1}^{|S|} (S_i - \bar{\mu})^T + \tilde{\delta}\tilde{\delta}^T \\ &= \bar{C} + \tilde{\delta}\tilde{\delta}^T \end{aligned} \quad (8)$$

Eq. (8) means that \hat{C} is the rank-one modification of \bar{C} (Bunch et al., 1978), that is why we named the improved estimation method as rank-one modification of the original CM. And we can give the following theorems and corollaries

Theorem 1: For a given set of selected solutions, if $\hat{\mu} \neq \bar{\mu}$, then the volume of the PDE determined by \hat{C} is not less than that of the PDE determined by \bar{C} .

Proof: According to the properties of Gaussian distribution, the axis directions of PDE are accordant with the eigendirections of CM, and the lengths of the semiaxes a_d ($d = 1, 2, \dots, n$) are equal to the square roots of corresponding eigenvalues λ_d , i.e. $a_d = \sqrt{\lambda_d}$. Based on the conclusion in (Zhao, 2013), the volume of PDE is proportional to the product of the lengths of its semiaxes. Denote the eigenvalues of \hat{C} and \bar{C} as $\hat{\lambda}_d$ and

$\bar{\lambda}_d (d = 1, 2, \dots, n)$, respectively, then a necessary and sufficient condition for *Theorem 1* is: $\prod_{d=1}^n \hat{\lambda}_d \geq \prod_{d=1}^n \bar{\lambda}_d$ when $\hat{\mu} \neq \bar{\mu}$.

To prove the above condition, we first conduct eigendecomposition on $\bar{\mathbf{C}}$. From Eq. (3), it can be seen that $\bar{\mathbf{C}}$ is a real symmetric matrix and is positive semidefinite. Then $\bar{\mathbf{C}}$ can be decomposed as $\bar{\mathbf{C}} = \bar{\mathbf{Q}}\bar{\mathbf{\Lambda}}\bar{\mathbf{Q}}^T$, in which $\bar{\mathbf{\Lambda}} = \text{diag}(\bar{\lambda}_1, \bar{\lambda}_2, \dots, \bar{\lambda}_n)$, $0 \leq \bar{\lambda}_1 \leq \bar{\lambda}_2 \leq \dots \leq \bar{\lambda}_n$ and $\bar{\mathbf{Q}}$ is a orthogonal matrix composed of the eigenvectors of $\bar{\mathbf{C}}$. Let $\mathbf{p} = \bar{\mathbf{Q}}^T \hat{\delta}$, then we have:

$$\begin{aligned} \bar{\mathbf{Q}}^T \hat{\mathbf{C}} \bar{\mathbf{Q}} &= \bar{\mathbf{Q}}^T (\bar{\mathbf{C}} + \hat{\delta} \hat{\delta}^T) \bar{\mathbf{Q}} \\ &= \bar{\mathbf{Q}}^T \bar{\mathbf{C}} \bar{\mathbf{Q}} + (\bar{\mathbf{Q}}^T \hat{\delta})(\bar{\mathbf{Q}}^T \hat{\delta})^T \\ &= \bar{\mathbf{\Lambda}} + \mathbf{p} \mathbf{p}^T \end{aligned} \quad (9)$$

It is clear that $\bar{\mathbf{\Lambda}} + \mathbf{p} \mathbf{p}^T$ is the rank-one modification of $\bar{\mathbf{\Lambda}}$. Lemma 1.4 in reference Yin (2003) discussed the characteristic polynomial of diagonal matrix with rank-one modification. Based on the result in Lemma 1.4 Yin (2003), we can derive the characteristic polynomial of $\bar{\mathbf{\Lambda}} + \mathbf{p} \mathbf{p}^T$, namely the characteristic polynomial of $\hat{\mathbf{C}}$, as follows:

$$\begin{aligned} F(\lambda) &= \prod_{d=1}^n (\lambda - \hat{\lambda}_d) \\ &= \det(\lambda \mathbf{I} - \bar{\mathbf{\Lambda}} - \mathbf{p} \mathbf{p}^T) \\ &= \det(\lambda \mathbf{I} - \bar{\mathbf{\Lambda}}) \det(\mathbf{I} - (\lambda \mathbf{I} - \bar{\mathbf{\Lambda}})^{-1} \mathbf{p} \mathbf{p}^T) \\ &= \prod_{d=1}^n (\lambda - \bar{\lambda}_d) \left(1 - \sum_{k=1}^n \frac{\mathbf{p}_k^2}{\lambda - \bar{\lambda}_k}\right) \\ &= \prod_{d=1}^n (\lambda - \bar{\lambda}_d) - \sum_{k=1}^n (\mathbf{p}_k^2 \prod_{d=1, d \neq k}^n (\lambda - \bar{\lambda}_d)) \end{aligned} \quad (10)$$

where \mathbf{I} represents the n -dimensional identity matrix. Set $\lambda = 0$ in Eq. (10), then we have:

$$\begin{aligned} \prod_{d=1}^n \hat{\lambda}_d &= \prod_{d=1}^n \bar{\lambda}_d + \sum_{k=1}^n (\mathbf{p}_k^2 \prod_{d=1, d \neq k}^n \bar{\lambda}_d) \\ &\geq \prod_{d=1}^n \bar{\lambda}_d \end{aligned} \quad (11)$$

So *Theorem 1* is proved.

Corollary 1: For a given set of selected solutions \mathbf{S} , if $\hat{\mu} \neq \bar{\mu}$ and $|\mathbf{S}| > n$, then the volume of the PDE determined by $\hat{\mathbf{C}}$ is greater than that of the PDE determined by $\bar{\mathbf{C}}$.

Proof: Considering that $\bar{\mathbf{Q}}^T$ in Eq. (9) is an orthogonal matrix, then $\mathbf{p} = \bar{\mathbf{Q}}^T \hat{\delta} \neq 0$ if $\hat{\delta} = \hat{\mu} - \bar{\mu} \neq 0$. This implies $\mathbf{p}_k \neq 0$ for some $k = 1, 2, \dots, n$. On the other side, all the solutions in \mathbf{S} are independently selected, then $\bar{\mathbf{C}}$ is positively definite with probability 1 if $|\mathbf{S}| > n$ (Xie and Chen, 1990). That is to say, $\bar{\lambda}_d > 0$ for each $d = 1, 2, \dots, n$. Taking the above two results related with Eq. (11) into account, it can be known that $\sum_{k=1}^n (\mathbf{p}_k^2 \prod_{d=1, d \neq k}^n \bar{\lambda}_d) > 0$ and $\prod_{d=1}^n \hat{\lambda}_d > \prod_{d=1}^n \bar{\lambda}_d$.

Corollary 2: For a given set of selected solutions, the greater the Mahalanobis distance between $\hat{\mu}$ and $\bar{\mu}$ ($\hat{\delta}^T \bar{\mathbf{C}}^{-1} \hat{\delta}$) is, the greater the volume of the PDE determined by $\hat{\mathbf{C}}$ will be.

Proof: Let us reconsider Eq. (11):

$$\begin{aligned} \prod_{d=1}^n \hat{\lambda}_d &= \prod_{d=1}^n \bar{\lambda}_d + \sum_{k=1}^n (\mathbf{p}_k^2 \prod_{d=1, d \neq k}^n \bar{\lambda}_d) \\ &= \prod_{d=1}^n \bar{\lambda}_d \left(1 + \sum_{k=1}^n (\mathbf{p}_k^2 / \bar{\lambda}_k)\right) \\ &= \prod_{d=1}^n \bar{\lambda}_d (1 + \mathbf{p}^T \bar{\mathbf{\Lambda}}^{-1} \mathbf{p}) \\ &= \prod_{d=1}^n \bar{\lambda}_d (1 + \hat{\delta}^T \bar{\mathbf{Q}} \bar{\mathbf{\Lambda}}^{-1} \bar{\mathbf{Q}}^T \hat{\delta}) \\ &= \prod_{d=1}^n \bar{\lambda}_d (1 + \hat{\delta}^T \bar{\mathbf{C}}^{-1} \hat{\delta}) \end{aligned} \quad (12)$$

So we can know that $\prod_{d=1}^n \hat{\lambda}_d$ is positively correlated with $\hat{\delta}^T \bar{\mathbf{C}}^{-1} \hat{\delta}$, so does the volume of the PDE determined by $\hat{\mathbf{C}}$.

Remark 1: *Corollary 1* theoretically gives a necessary and sufficient condition for assuring that the volume of the PDE determined by $\hat{\mathbf{C}}$ is greater than the volume of the PDE determined by $\bar{\mathbf{C}}$. According to the above analyses, EDA-R1M adopts $\hat{\mu}$ as the center, which is usually better than and not equal to $\bar{\mu}$. On the other side, the population size of EDA-R1M and traditional multivariate GEDA are generally much greater than the problem dimension, then the number of selected solutions could usually satisfy $|\mathbf{S}| > n$. So *Corollary 1* is generally tenable in EDA-R1M, which means that, for the same set of selected solutions, the exploration ability of EDA-R1M is usually greater than that of the traditional GEDA.

Remark 2: *Corollary 2* shows that if we want to improve the exploration ability of EDA-R1M, we need to increase the Mahalanobis distance between $\hat{\mu}$ and $\bar{\mu}$ ($\hat{\delta}^T \bar{\mathbf{C}}^{-1} \hat{\delta}$), i.e. increase $\mathbf{p}^T \bar{\mathbf{\Lambda}}^{-1} \mathbf{p}$. For a given set of selected solutions, $\bar{\mathbf{C}}$ and $\bar{\mathbf{\Lambda}}$ are both determined. So we could only increase $\mathbf{p}^T \bar{\mathbf{\Lambda}}^{-1} \mathbf{p}$ by increasing the sub-components of \mathbf{p} , i.e. $p_d (d = 1, 2, \dots, n)$, which are the projections of $\hat{\delta}$ on the eigenvectors of $\bar{\mathbf{C}}$. According to Eq. (6), we can alter the position of $\hat{\mu}$ by adjusting the shifting factor η . A larger value for η could move $\hat{\mu}$ to a farther position, and thus increase $\hat{\delta}$ and improve the exploration ability of EDA-R1M.

Theorem 2: For a given set of selected solutions, if $\hat{\mu} \neq \bar{\mu}$, then the angle between $\hat{\delta}$ and the major axis of the PDE determined by $\hat{\mathbf{C}}$ is not greater than the angle between $\hat{\delta}$ and the major axis of the PDE determined by $\bar{\mathbf{C}}$.

Proof: Without loss of generality, it can be assumed that the eigenvalues of $\bar{\mathbf{\Lambda}} + \mathbf{p} \mathbf{p}^T$ satisfy $0 \leq \hat{\lambda}_1 \leq \hat{\lambda}_2 \leq \dots \leq \hat{\lambda}_n$. Denote the major eigenvectors of $\bar{\mathbf{\Lambda}} + \mathbf{p} \mathbf{p}^T$ and $\bar{\mathbf{\Lambda}}$ as $\hat{\mathbf{q}}$ and $\bar{\mathbf{q}}$ respectively, where the major eigenvector means the eigenvector with the greatest eigenvalue. According to Eq. (9), $\bar{\mathbf{\Lambda}} + \mathbf{p} \mathbf{p}^T$ and $\bar{\mathbf{\Lambda}}$ are the orthogonal transformations of $\hat{\mathbf{C}}$ and $\bar{\mathbf{C}}$ by taking $\bar{\mathbf{Q}}^T$ as the transformation matrix, respectively. Then *Theorem 2* is equivalent to proving that if $\bar{\mathbf{Q}}^T \hat{\delta} = \mathbf{p} \neq 0$, then $\angle(\mathbf{p}, \hat{\mathbf{q}}) \leq \angle(\mathbf{p}, \bar{\mathbf{q}})$. Besides, since the major axis of a PDE is undirected, there will be two angles between it and \mathbf{p} . Here we only focus on analyzing the smaller angle between them, so we can assume that $\mathbf{p}^T \hat{\mathbf{q}} \geq 0, \mathbf{p}^T \bar{\mathbf{q}} \geq 0$.

Case 1: If $\mathbf{p}^T \bar{\mathbf{q}} = 0$, then $\angle(\mathbf{p}, \bar{\mathbf{q}}) = 90^\circ$. Since $\angle(\mathbf{p}, \hat{\mathbf{q}}) \leq 90^\circ$, then we can know that $\angle(\mathbf{p}, \hat{\mathbf{q}}) \leq \angle(\mathbf{p}, \bar{\mathbf{q}})$.

Case 2: If $\mathbf{p}^T \bar{\mathbf{q}} \neq 0$, then $\hat{\lambda}_n > \bar{\lambda}_n$ and $\mathbf{p}^T \bar{\mathbf{q}} \neq 0$. The reasons lie in that: First, $\bar{\mathbf{\Lambda}} + \mathbf{p} \mathbf{p}^T$ is the rank-one modification of $\bar{\mathbf{\Lambda}}$, then $\hat{\lambda}_d \geq \bar{\lambda}_d (d = 1, 2, \dots, n)$; Second, the projection of \mathbf{p} on $\bar{\mathbf{q}}$ is not equal to zero as $\mathbf{p}^T \bar{\mathbf{q}} \neq 0$, thus $\hat{\lambda}_n > \bar{\lambda}_n$ (Bunch et al., 1978). Furthermore, consider the definition of eigenvalue:

$$(\bar{\mathbf{\Lambda}} + \mathbf{p} \mathbf{p}^T) \hat{\mathbf{q}} = \hat{\lambda}_n \hat{\mathbf{q}} \quad (13)$$

If $\mathbf{p}^T \bar{\mathbf{q}} = 0$ in Eq. (13), then we have $\bar{\mathbf{\Lambda}} \hat{\mathbf{q}} = \hat{\lambda}_n \hat{\mathbf{q}}$. It means that $\hat{\lambda}_n$ is also the eigenvalue of $\bar{\mathbf{\Lambda}}$, this is contradictory with the former conclusion $\hat{\lambda}_n > \bar{\lambda}_n$. So $\mathbf{p}^T \bar{\mathbf{q}} \neq 0$. Besides, we can also derive from Eq. (13) that $\hat{\mathbf{q}} = (\hat{\lambda}_n \mathbf{I} - \bar{\mathbf{\Lambda}})^{-1} \mathbf{p} \mathbf{p}^T \hat{\mathbf{q}}$.

Set λ in Eq. (10) as $\hat{\lambda}_n$, then

$$\begin{aligned} F(\hat{\lambda}_n) &= \left(1 - \sum_{d=1}^n \frac{\mathbf{p}_d^2}{\hat{\lambda}_n - \bar{\lambda}_d}\right) \prod_{d=1}^n (\hat{\lambda}_n - \bar{\lambda}_d) \\ &= 0 \\ &\Rightarrow \sum_{d=1}^n \frac{\mathbf{p}_d^2}{\hat{\lambda}_n - \bar{\lambda}_d} = 1 \\ &\Rightarrow 1 \geq \frac{\mathbf{p}_n^2}{\hat{\lambda}_n - \bar{\lambda}_n} > 0 \end{aligned} \quad (14)$$

Then:

$$\begin{aligned}
\cos \angle(\mathbf{p}, \hat{\mathbf{q}}) &= \frac{\mathbf{p}^T \hat{\mathbf{q}}}{\|\mathbf{p}\| \cdot \|\hat{\mathbf{q}}\|} = \frac{\mathbf{p}^T \hat{\mathbf{q}}}{\|\mathbf{p}\| \cdot \|(\hat{\lambda}_n I - \bar{\Lambda})^{-1} \mathbf{p}\| \cdot \mathbf{p}^T \hat{\mathbf{q}}} \\
&= \frac{1}{\|\mathbf{p}\| \sqrt{\sum_{d=1}^n \frac{p_d^2}{(\hat{\lambda}_n - \bar{\lambda}_d)^2}}} \\
&\geq \frac{|\mathbf{p}_n|}{\|\mathbf{p}\| \sqrt{(\hat{\lambda}_n - \bar{\lambda}_n) \sum_{d=1}^n \frac{p_d^2}{(\hat{\lambda}_n - \bar{\lambda}_d)^2}}} \geq \frac{|\mathbf{p}_n|}{\|\mathbf{p}\| \sqrt{(\hat{\lambda}_n - \bar{\lambda}_n) \sum_{d=1}^n \frac{p_d^2}{(\hat{\lambda}_n - \bar{\lambda}_n)(\hat{\lambda}_n - \bar{\lambda}_d)}}} \\
&= \frac{|\mathbf{p}_n|}{\|\mathbf{p}\| \sqrt{\sum_{d=1}^n \frac{p_d^2}{\hat{\lambda}_n - \bar{\lambda}_d}}} = \frac{|\mathbf{p}_n|}{\|\mathbf{p}\|} = \frac{\mathbf{p}^T \hat{\mathbf{q}}}{\|\mathbf{p}\| \cdot \|\hat{\mathbf{q}}\|} = \cos \angle(\mathbf{p}, \bar{\mathbf{q}})
\end{aligned} \tag{15}$$

So we can know that $\angle(\mathbf{p}, \hat{\mathbf{q}}) \leq \angle(\mathbf{p}, \bar{\mathbf{q}})$. Combining the results of Case 1 and Case 2, *Theorem 2* is proved.

Remark 3: In Case 1 ($\mathbf{p}^T \bar{\mathbf{q}} = 0$), it can be drawn that $\hat{\lambda}_n$ and $\bar{\mathbf{q}}$ are also the eigenvalue and eigenvector of $\bar{\Lambda} + \mathbf{p}\mathbf{p}^T$, respectively. The reason lies in that when $\|\mathbf{p}\|$ is relatively small, it would only have very slight influence on the eigenvalues. In this circumstance, $\hat{\lambda}_n$ may remain to be the major eigenvalue of $\bar{\Lambda} + \mathbf{p}\mathbf{p}^T$, thus $\bar{\lambda}_n = \hat{\lambda}_n$ and $\angle(\mathbf{p}, \hat{\mathbf{q}}) = \angle(\mathbf{p}, \bar{\mathbf{q}}) = 90^\circ$. With respect to Case 2, $\angle(\mathbf{p}, \hat{\mathbf{q}}) = \angle(\mathbf{p}, \bar{\mathbf{q}}) = 0^\circ$ would happen only when $\mathbf{p}_n^2 = \hat{\lambda}_n - \bar{\lambda}_n$, $\mathbf{p}_d = 0$ ($d = 1, 2, \dots, n-1$), i.e. \mathbf{p} is parallel to $\bar{\mathbf{q}}$. In most other cases, we can have $\angle(\mathbf{p}, \hat{\mathbf{q}}) < \angle(\mathbf{p}, \bar{\mathbf{q}})$. Since the major axis of PDE (the major eigenvector of covariance matrix) determines the main search direction of GEDA, *Theorem 2* implies that the main search direction of GEDA will be more close to the fitness improvement direction when employing $\hat{\mathbf{C}}$ as the covariance matrix.

Through *Theorem 1* and *Theorem 2*, we theoretically show that EDA with R1M (EDA-R1M) has greater search scope and better main search direction than the traditional GEDA. Thus it is reasonable to believe that the search efficiency of EDA-R1M could be greatly improved. In addition, *Corollary 2* and *Remark 2* show that the exploration ability of EDA-R1M could be adjusted by tuning the shifting factor η , so we employ a simple line search method shown in *Algorithm 2* to adaptively get a proper value for η .

3.3. Population reduction strategy

As a type of population-based optimization algorithm, the tuning of population size plays a substantial role in affecting the performance of EDA. A large population size could improve the exploration degree to the solution region dominated by the current probabilistic model, but would slow down the convergence process. In addition, it would also reduce the number of iterations if only limited computation resource is available, which further restricts the convergence ability. On the other hand, a small population size could quicken the convergence, but may increase the risk of premature convergence. Existing EDAs usually adopt a constant population size throughout the evolution and can hardly balance the exploration and exploitation ability in different search stages. Furthermore, GEDAs using full CM generally need a relatively large population size to estimate a proper model, which would consume much computation resource at each generation and thus limit their performance, as discussed in Section 2.

To remedy the above deficiencies, we attempted to develop a dynamic population size tuning strategy for EDA. On the one hand, it is hoped that the strategy could dynamically adjust the population size in different stages. In the earlier search stage, a larger population size could help EDA more accurately capture the characteristics of the solution space and locate promising regions. In the latter search stage, the algorithm may have converged to local solution regions. In this situation, a smaller population size is preferred to accelerate the convergence. On the other hand, the population size tuning strategy should be easy to operate in order to reduce the computation burden. In existing researches, population size tuning methods based on some simple and deterministic rules, such as monotonically increasing or decreasing the population size, have been found to be very effective to enhance the

performance of several other kinds of EAs (de Oca et al., 2011; Tanabe and Fukunaga, 2014). Considering the above reasons, we designed a population reduction strategy for EDA to dynamically change its population size. At each generation, we apply the following equation to calculate the new population size for next generation:

$$PS^{t+1} = \text{Round}[PS_{\max} - (PS_{\max} - PS_{\min}) \frac{FES^t}{FES_{\max}}] \tag{16}$$

where PS^{t+1} denotes the population size in $(t+1)$ th generation, PS_{\max} and PS_{\min} are the defined maximum and minimum population size, respectively. FES^t and FES_{\max} denote the consumed total number of fitness evaluations (FEs) in the first t generations and the allowed maximum number of evaluations, respectively. *Round()* is a rounding function. At the first generation, the initial population size is equal to PS_{\max} . After that, the population size is decreased linearly until it reach the defined minimum value PS_{\min} .

PS_{\max} and PS_{\min} play different roles in affecting the performance of EDA. PS_{\max} directly affects the exploration ability of EDA at the earlier search stage. A larger value for it could encourage the exploration but may reduce the convergence rate. While a smaller value for PS_{\max} would make it lose its effect. We will experimentally analyse the effect of PS_{\max} in Section 4.1. As for PS_{\min} , on the one hand, it should be as small as possible to accelerate the convergence at the latter stage; on the other hand, it need to ensure that EDA could properly estimate the Gaussian model based on the current population. Considering that the full CM of Gaussian model totally has $0.5(n^2 + n)$ parameters need to be estimated, we suggest setting $PS_{\min} = 0.5(n^2 + n)$.

3.4. Procedure of EDA-R1M-PR

Integrating the new model estimation method and the population reduction strategy into GEDA, we can get the procedure of EDA-R1M-PR presented in *Algorithm 3*. In steps 1 and 2, EDA-R1M-PR initializes the parameters and generates the initial population randomly and uniformly in the solution space. In steps 4 and 5, the population are evaluated and a set of promising solutions are selected using the truncation selection rule. Then in steps 6–8, the new mean $\hat{\mu}^t$ and covariance matrix \hat{C}^t are estimated according to the improved estimation method and a new Gaussian model is built. Before sampling new solutions for the next generation, the population size is decreased using the population reduction strategy in step 9. We also employ an elitism strategy by maintaining the best solution in current generation to the next generation in step 11, so only $(PS^{t+1} - 1)$ new solutions are generated in step 10. The above steps are iterated until a stopping criterion is met, such as the FES_{\max} is reached or the algorithm cannot find better solution in a number of consecutive generations.

Compared to the traditional ML estimation method, the improved model estimation method doesn't increase the computation complexity in the model building process. The computation complexity of the linear population reduction strategy is just $O(1)$. So the overall computation complexity of EDA-R1M-PR is the same with that of the traditional multivariate GEDA.

Algorithm 3: Procedure of EDA-R1M-PR

- 1 Initialize selection ratio τ , η_{\max} , PS_{\max} and PS_{\min} ;
 - 2 Set $t = 0$, population size $PS^t = PS_{\max}$, generate the initial population P^t ;
 - 3 **while** the stopping criterion is not met **do**
 - 4 Evaluate population P^t and update the best solution \mathbf{b} obtained so far;
 - 5 Select the best $\lfloor \tau PS^t \rfloor$ solutions from P^t and store them into S^t ;
 - 6 Estimate $\hat{\mu}^t$ according to Eqs. (4)–(6) and *Algorithm 2*;
 - 7 Estimate \hat{C}^t according to Eq. (7);
 - 8 Build a new probabilistic model G^t based on $\hat{\mu}^t$ and \hat{C}^t ;
 - 9 Compute the new population size PS^{t+1} according to Eq. (16);
 - 10 Generate $(PS^{t+1} - 1)$ new solutions by sampling from G^t and store them into A^t ;
 - 11 Set $P^{t+1} = A^t \cup \mathbf{b}$ and update $t = t + 1$;
 - 12 **end**
 - 13 **return** \mathbf{b} .
-

4. Experimental studies

In this section, EDA-R1M-PR was tested on a set of 30 benchmark functions from IEEE CEC 2014 (Liang et al., 2013) to evaluate its effectiveness. These functions are shifted and rotated to ensure the solving difficulty, and they could be classified into four classes, including unimodal functions f_1 – f_3 , multimodal functions f_4 – f_{16} , hybrid functions f_{17} – f_{22} , and composition functions f_{23} – f_{30} . More details of these functions can be found in (Liang et al., 2013).

In our experiments, each algorithm was tested on these functions with 30 dimensions (30D) and 50 dimensions (50D). 25 independent runs were conducted on each function with $FES_{max} = 10,000 \cdot D$ as the stopping criterion for each run. The function error value ($f(\mathbf{b}) - f(\mathbf{x}^*)$) of each run was recorded to evaluate the performance of an algorithm, where \mathbf{b} is the obtained best solution and \mathbf{x}^* is the optimal one. Note that the function error value (FEV) smaller than 10^{-8} was reported as zero.

4.1. Influence of parameters

There are four parameters in EDA-R1M-PR, including the truncation selection ratio τ , shifting factor η , and the maximum population size PS_{max} and minimum population size PS_{min} . The selection ratio was conventionally set as $\tau = 0.35$. The value for the shifting factor η could be adaptively obtained by the line search method presented in **Algorithm 2** in Section 3.1. The value for PS_{min} was discussed in Section 3.3. In this section, we mainly focus on analyzing the influence of PS_{max} . To achieve that, we tested EDA-R1M-PR on a variety of functions with different PS_{max} values. For brevity, only the results on two functions, i.e. unimodal function f_2 and multimodal function f_9 , with 30D and 50D were reported.

Figs. 2 and 3 show the performance of EDA-R1M-PR on the two functions with 30D and 50D, respectively. It can be seen from Figs. 2(a) and 3(a) that EDA-R1M-PR could obtain the optimum of f_2 with a wide range of PS_{max} values and its performance only deteriorates a little on f_2

with 50D when $PS_{max} = 6000$. As for multimodal function f_9 , EDA-R1M-PR could also find the near optimal solution and its performance gradually becomes better with the increase of PS_{max} . From Figs. 2 and 3, we can conclude that EDA-R1M-PR is rather robust to the change of PS_{max} . A larger value for PS_{max} is helpful to enhance the exploration ability, and thus improves the performance on multimodal functions. On the other side, it also slows down the convergence speed on unimodal functions. To balance the performance of EDA-R1M-PR on different classes of functions, we set $PS_{max} = 100 \cdot D$ in the following experiments.

4.2. Effectiveness analysis

In this section, the effectiveness of the rank-one modification and the population reduction strategies is analyzed. For the convenience of analysis, we implemented two variants of EDA-R1M-PR:

- 1) EDA-R1M: the variant of EDA-R1M-PR without employing the population reduction strategy. Instead of using dynamic population size as EDA-R1M-PR, EDA-R1M sets its population size to a constant value, which is 1000 for functions with 30D and 2000 for functions with 50D. The comparison between EDA-R1M and EDA-R1M-PR could reflect the effectiveness of the population reduction strategy.
- 2) quasi-EDA-R1M: a variant strictly accompanying with EDA-R1M. It is developed only for verifying the effectiveness of the new CM estimation method (rank-one modification operation), but not for optimizing the test functions. At each generation, quasi-EDA-R1M estimates its covariance matrix based on the same solutions with EDA-R1M, but employs the traditional ML estimator described in Eqs. (2) and (3) instead of the improved estimation method described in Eqs. (4)–(6). Consequently, the efficiency of the R1M operation could be revealed by comparing the estimated CMs of EDA-R1M and quasi-EDA-R1M since the two algorithms use the same set of selected solutions but different estimation methods to estimate the CM.

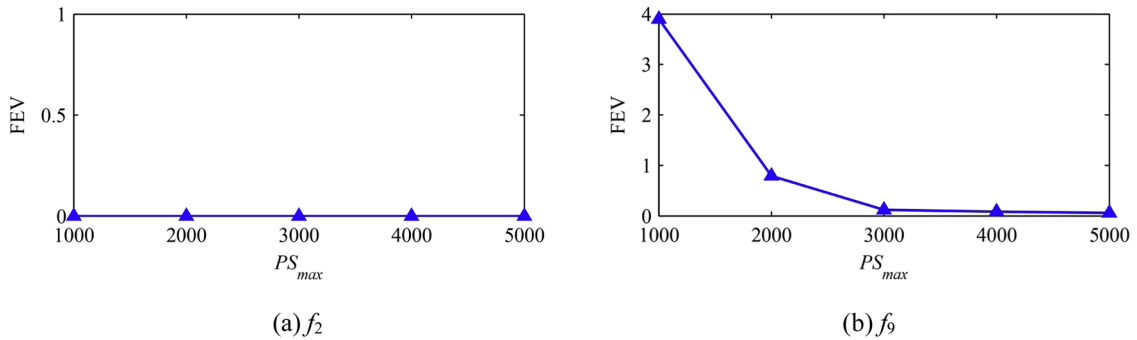


Fig. 2. Performance of EDA-R1M-PR with different PS_{max} values on functions with 30D.

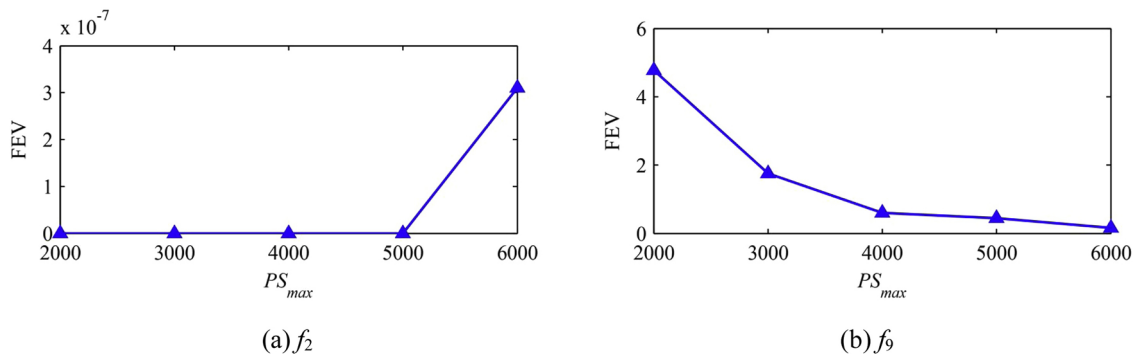


Fig. 3. Performance of EDA-R1M-PR with different PS_{max} values on functions with 50D.

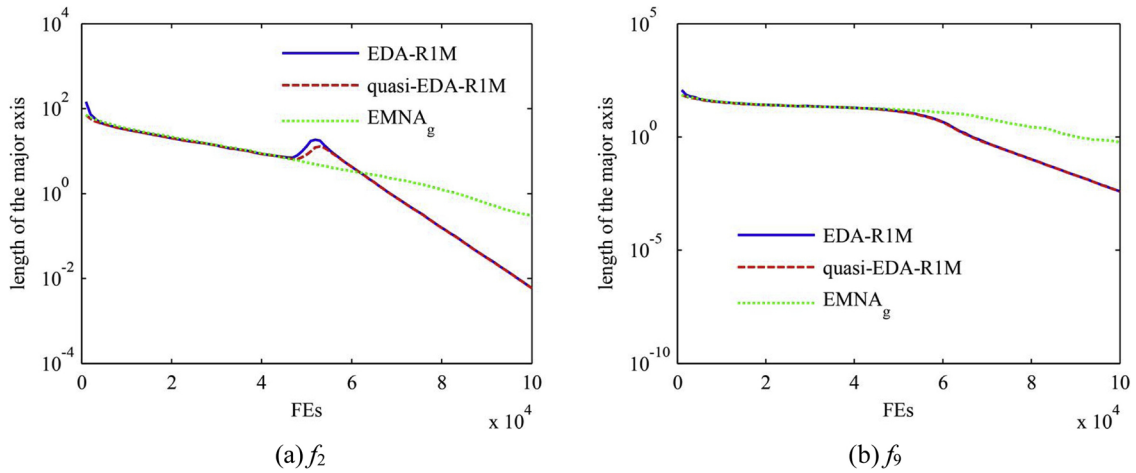


Fig. 4. The variation of the length of PDE's major axis on functions with 30D.

In addition to the above two variants, two other existing algorithms, i.e. EMNA_g (Larrañaga and Lozano, 2001) and AMaLGaM (Bosman et al., 2008) are also included in the comparison. EMNA_g is a representative of the traditional multivariate GEDA. AMaLGaM is an efficient GEDA variant employing the techniques of AVS, SRD and AMS. The population size and selection ratio of EMNA_g and AMaLGaM are all set to the same with that of EDA-R1M to make a fair comparison. AMaLGaM also employs a full covariance matrix and its other parameters are all set to the same values in its original paper (Bosman et al., 2008).

Fig. 4 presents the variation of the length of PDE's major axis on two functions with 30D, respectively. It can be seen from Fig. 4 that the length of the major axis of EDA-R1M is generally not smaller than that of quasi-EDA-R1M during the whole evolution, especially in the initial stage. This conclusion is also valid for other non-major axes, which is consistent with the conclusion in Theorem 1. Fig. 5 shows the variation of the cosine of the angle between the major axis and $\hat{\delta}$ (denoted as $\cos(\theta_1)$). The results in Fig. 5 indicate that the value of $\cos(\theta_1)$ obtained by EDA-R1M is always no less than the value obtained by quasi-EDA-R1M. That is to say, the angle between $\hat{\delta}$ and the major axis of the PDE of EDA-R1M is not greater than that of quasi-EDA-R1M. This phenomenon is in line with the conclusion in Theorem 2. The results in Figs. 4 and 5 demonstrate that the R1M operation could actually enlarge the search scope and help the algorithm find better main search direction.

Fig. 4 also shows the variation of the length of the major axis of EMNA_g. It can be observed that the major axis length of EDA-R1M is

usually larger than that of EMNA_g in the earlier search stage, which is beneficial to explore promising solution regions. While in the latter stage, the major axis length of EDA-R1M quickly decreases and its search effort is focused in a relatively small region. As a result, the exploitation ability of EDA-R1M is improved to refine the quality of promising solutions more efficiently. To further demonstrate the performance difference between EDA-R1M and EMNA_g, Fig. 6 presents the variation of the cosine of the angle between the PDE's major axis and the steepest descent direction (denoted as $\cos(\theta_2)$) of the two algorithms on functions with 30D, where the steepest descent direction is defined as the direction from the current mean to the optimal solution ($\mathbf{x}^* - \hat{\mu}$). As shown in Fig. 6(a), EDA-R1M could always achieve larger cosine value than EMNA_g on f_2 , and it could find the steepest descent direction after only a few generations. As for f_9 , EDA-R1M also obtains better search direction than EMNA_g in the earlier search stage. Fig. 7 presents the variation of the FEV obtained by EDA-R1M and EMNA_g, it is clear that EDA-R1M demonstrates excellent performance on f_2 , and also achieves far better results than EMNA_g on f_9 even though they are both trapped into local optima.

Fig. 7 also shows the evolution of the FEV derived from EDA-R1M-PR and AMaLGaM. The effectiveness of the population reduction strategy could be verified by comparing the performance of EDA-R1M and EDA-R1M-PR. It can be observed from Fig. 7(a) that both EDA-R1M and EDA-R1M-PR obtain the optimal solution of f_2 , with the former keeping a fast improvement tendency and the latter gradually improving its convergence rate by reducing the population size. Although the population reduction strategy slows down the convergence in the

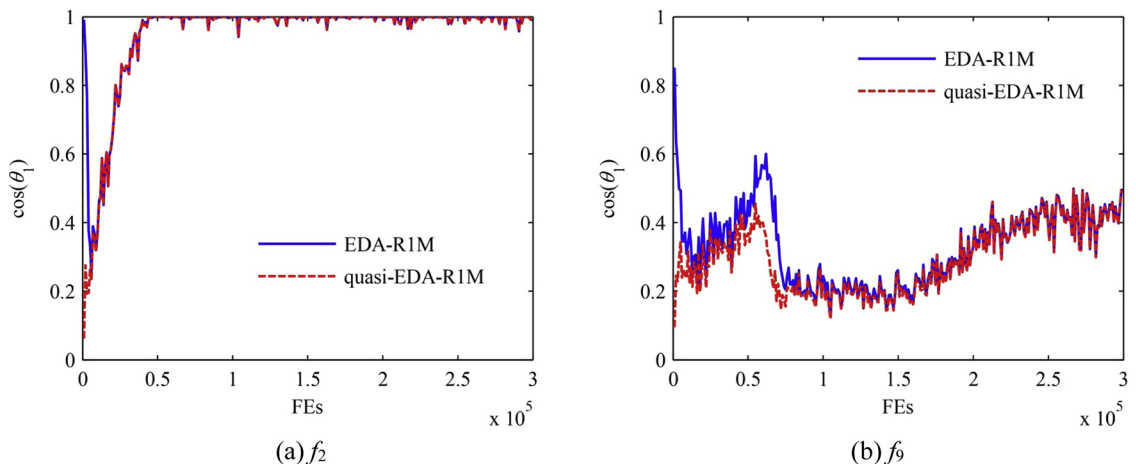


Fig. 5. The variation of the cosine of the angle between the major axis and $\hat{\delta}$ on functions with 30D.

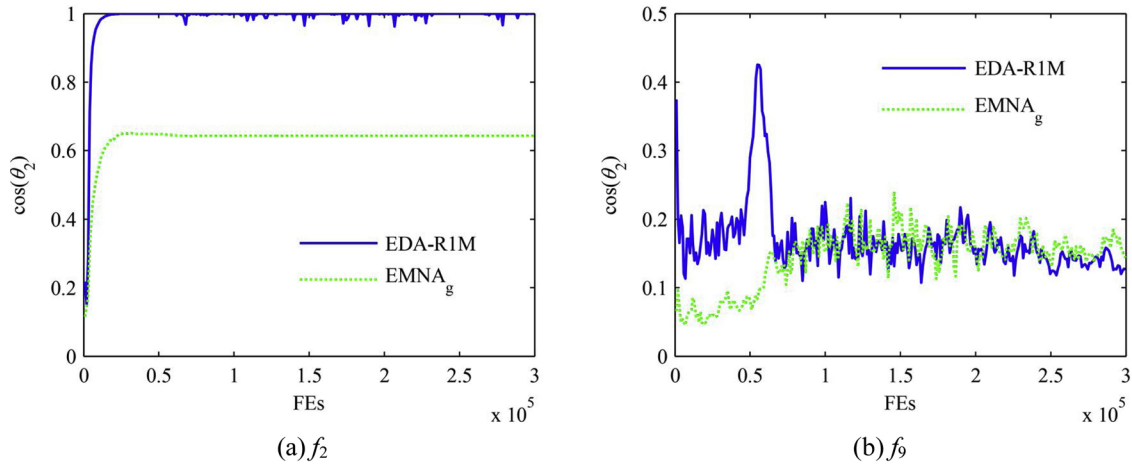


Fig. 6. The variation of the cosine of the angle between the major axis and the steepest descent direction on functions with 30D.

early stage, it is beneficial to explore more promising solution regions and avoid premature convergence. Taking advantage of this property, EDA-R1M-PR could usually achieve better results on relatively complicated multimodal functions like f_9 . As for AMaLGaM, we can see from Fig. 7 that it is outperformed by EDA-R1M and EDA-R1M-PR on both functions.

4.3. Comparison with other state-of-the-art EAs

To further assess the efficiency of EDA-R1M-PR, we compare it with AMaLGaM (Bosman et al., 2008), IPOP-CMA-ES (Auger and Hansen, 2005), BL-PSO-5 (Chen et al., 2017b), L-SHADE (Tanabe and Fukunaga, 2014), MVC_E_S_C (Zhang et al., 2017), as well as EDA-R1M. AMaLGaM is a representative of efficient EDA variant. IPOP-CMA-ES improves the famous CMA-ES by restarting CMA-ES with increasing population size when it gets stuck (Auger and Hansen, 2005). IPOP-CMA-ES was showed to be very competitive to many other efficient EAs developed in recent years (García-Martínez et al., 2017). BL-PSO-5 merges PSO with biogeography-based learning strategy. Numeric simulation indicated that BL-PSO-5 outperforms several well-designed PSO variants and EAs (Chen et al., 2017b). L-SHADE improves the success-history based adaptive DE (SHADE) (Tanabe and Fukunaga, 2013) with a linear population size reduction scheme, it is the winner of the CEC 2014 Competition on Real-parameter Single Objective optimization (Tanabe and Fukunaga, 2014). MVC_E_S_C incorporates three efficient DEs including EPSDE (Mallipeddi et al., 2011), SHADE and CoBiDE (Wang et al., 2014) into a multiple variants coordination (MVC) framework,

which could adaptively select the most suitable optimizer for different search stages and preserve promising solutions for multiple DE optimizers. It was reported that MVC_E_S_C could outperform several up-to-date DEs (Zhang et al., 2017). Besides the above algorithms, EDA-R1M is also included in the comparison to comprehensively investigate the effectiveness of the two components of EDA-R1M-PR, i.e. the rank-one modification operation and population reduction strategy.

We employed the source code of IPOP-CMA-ES provided by the author of (Loshchilov, 2013), and directly took the experimental results of BL-PSO-5, L-SHADE and MVC_E_S_C from their original papers. Cohen's d effect size (Cohen, 1988) was employed to measure the performance difference between EDA-R1M-PR and other algorithms. Cohen's d effect size is a simple and robust measure for quantifying the difference between two groups of data, it relies less on the sample size. Generally, a “small”, “medium”, and “large” effect is considered if the magnitude of effect size falls within $[0.2, 0.3)$, $[0.3, 0.8)$ and $[0.8, +\infty)$, respectively. Tables 1 and 2 summarize the experimental results of the above seven algorithms on the 30 CEC 2014 benchmark functions with 30D and 50D, respectively, where “+”, “−”, and “ \approx ” denote that the result of an algorithm is better than, worse than, and similar to that of EDA-R1M-PR, respectively. Besides, the best results in Tables 1 and 2 are highlighted in bold. From Tables 1 and 2, we can summarize that:

- 1) For unimodal functions f_1 – f_3 , EDA-R1M-PR and EDA-R1M obtain the global optima for the three functions with both 30D and 50D, so does IPOP-CMA-ES. AMaLGaM and L-SHADE achieve fine results on these functions with 30D, but they both show performance

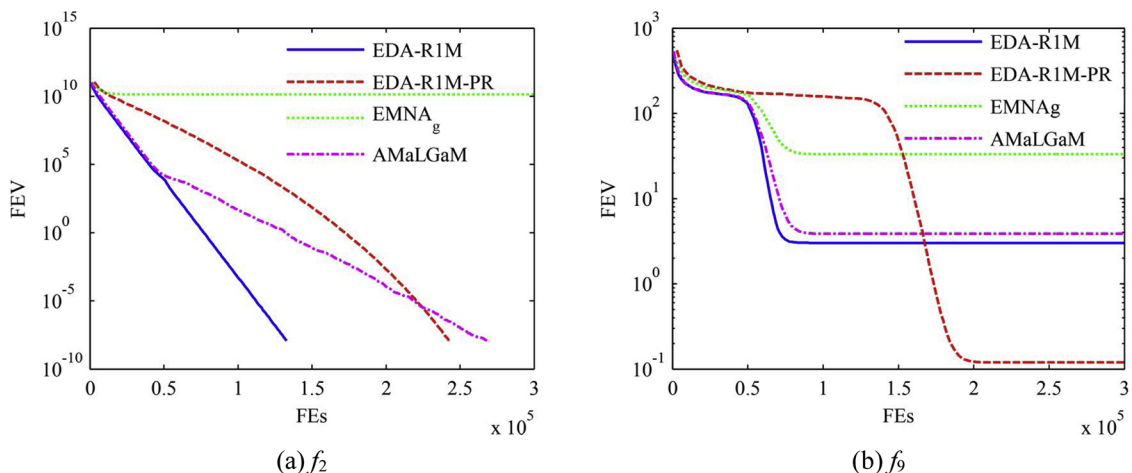


Fig. 7. The variation of the FEV on functions with 30D.

Table 1
the mean and standard deviation (mean \pm standard deviation) of FEVs obtained by 7 algorithms on CEC 2014 benchmark functions with 30D.

| Fun. | AMaL-GaM | IPOP-CMA-ES | BL-PSO-5 | L-SHADE | MVC_E_S_C | EDA-R1M | EDA-R1M-PR |
|-----------------------------|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|-------------------------|
| f_1 | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 2.99E+06 \pm 1.10E+06 – | 0.00E+00 \pm 0.00E+00 \approx | 6.62E+03 \pm 5.93E+03 – | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 |
| f_2 | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 5.09E+03 \pm 4.25E+03 – | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 |
| f_3 | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 3.67E+00 \pm 1.16E+01 – | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 |
| f_4 | 1.05E+01 \pm 2.10E+01 – | 0.00E+00 \pm 0.00E+00 + | 2.68E+01 \pm 3.47E+01 – | 0.00E+00 \pm 0.00E+00 + | 4.95E+00 \pm 6.93E+00 – | 4.27E+00 \pm 2.08E+00 – | 3.88E–02 \pm 7.96E–02 |
| f_5 | 2.09E+01 \pm 2.05E–01 \approx | 2.09E+01 \pm 5.40E–02 \approx | 2.08E+01 \pm 7.01E–02 + | 2.01E+01 \pm 3.68E–02 + | 2.00E+01 \pm 7.78E–05 + | 2.09E+01 \pm 4.19E–02 \approx | 2.09E+01 \pm 5.34E–02 |
| f_6 | 1.83E–02 \pm 9.18E–02 – | 7.54E–02 \pm 3.20E–05 – | 9.37E–06 \pm 3.20E–05 – | 1.38E–07 \pm 9.89E–07 – | 1.30E+00 \pm 1.53E+00 – | 6.04E–02 \pm 2.19E–01 – | 0.00E+00 \pm 0.00E+00 |
| f_7 | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 |
| f_8 | 3.86E+00 \pm 1.77E+00 – | 8.88E–01 \pm 8.89E–01 – | 2.32E–01 \pm 5.65E–01 + | 0.00E+00 \pm 0.00E+00 + | 0.00E+00 \pm 0.00E+00 + | 4.66E+00 \pm 1.57E+00 – | 5.17E–01 \pm 7.66E–01 |
| f_9 | 4.01E+00 \pm 1.63E+00 – | 1.24E+00 \pm 1.68E+00 – | 3.54E+01 \pm 6.93E+01 – | 6.78E+00 \pm 1.48E+00 – | 2.53E+01 \pm 6.83E+00 – | 3.14E+00 \pm 1.81E+00 – | 1.34E–01 \pm 2.18E–01 |
| f_{10} | 4.41E+02 \pm 2.60E+02 – | 4.62E+02 \pm 1.01E+03 – | 8.83E+01 \pm 6.48E+01 – | 1.63E–02 \pm 1.58E–02 + | 3.20E–01 \pm 7.47E–01 + | 2.33E+02 \pm 1.97E+02 – | 8.84E+00 \pm 2.33E+02 |
| f_{11} | 5.17E+02 \pm 2.91E+02 – | 3.55E+02 \pm 3.14E+02 – | 2.08E+03 \pm 3.82E+02 – | 1.23E+03 \pm 1.83E+02 – | 1.61E+03 \pm 5.76E+02 – | 3.60E+02 \pm 2.39E+02 – | 3.12E+01 \pm 6.09E+01 |
| f_{12} | 2.45E+00 \pm 7.98E–01 \approx | 2.40E+00 \pm 3.73E–01 + | 8.83E–01 \pm 1.49E–01 + | 1.61E–01 \pm 2.29E–02 + | 4.24E–02 \pm 1.78E–02 + | 2.43E+00 \pm 2.66E–01 + | 2.49E+00 \pm 2.16E–01 |
| f_{13} | 5.63E–02 \pm 1.12E–02 – | 6.91E–02 \pm 2.13E–02 – | 2.21E–01 \pm 2.85E–02 – | 1.24E–01 \pm 1.75E–02 – | 1.66E–01 \pm 3.83E–02 – | 3.92E–02 \pm 1.07E–02 + | 4.17E–02 \pm 8.54E–03 |
| f_{14} | 4.18E–01 \pm 3.87E–02 – | 3.73E–01 \pm 5.71E–02 \approx | 2.14E–01 \pm 2.88E–02 + | 2.42E–01 \pm 2.98E–02 + | 2.20E–01 \pm 2.84E–02 + | 4.20E–01 \pm 3.98E–02 – | 3.82E–01 \pm 3.36E–02 |
| f_{15} | 3.03E+00 \pm 5.42E–01 – | 2.49E+00 \pm 3.29E–01 + | 7.41E+00 \pm 8.49E–01 – | 2.15E+00 \pm 2.51E–01 + | 2.40E+00 \pm 6.63E–01 + | 2.71E+00 \pm 3.91E–01 – | 2.63E+00 \pm 2.99E–01 |
| f_{16} | 1.08E+01 \pm 4.76E–01 – | 1.04E+01 \pm 1.88E+00 – | 9.67E+00 \pm 4.92E–01 – | 8.50E+00 \pm 4.58E–01 – | 8.62E+00 \pm 6.65E–01 + | 9.57E+00 \pm 5.32E–01 \approx | 9.48E+00 \pm 4.88E–01 |
| f_{17} | 2.31E+02 \pm 1.34E+02 – | 1.93E+03 \pm 3.89E+02 – | 1.86E+05 \pm 1.11E+05 – | 1.88E+02 \pm 7.50E+01 – | 6.49E+02 \pm 3.49E+02 – | 1.25E+02 \pm 6.61E+01 – | 1.85E+01 \pm 2.47E+01 |
| f_{18} | 1.64E+01 \pm 6.54E+00 – | 5.85E+02 \pm 2.12E+02 – | 9.05E+02 \pm 1.20E+03 – | 5.91E+00 \pm 2.86E+00 – | 2.64E+01 \pm 1.55E+01 – | 1.01E+01 \pm 5.85E+00 – | 8.16E–01 \pm 5.97E–01 |
| f_{19} | 3.78E+00 \pm 7.31E–01 – | 7.24E+00 \pm 1.36E+00 – | 3.74E+00 \pm 6.12E–01 – | 3.68E+00 \pm 6.80E–01 – | 2.58E+00 \pm 5.68E–01 – | 3.22E+00 \pm 5.93E–01 – | 2.14E+00 \pm 5.71E–01 |
| f_{20} | 3.05E+00 \pm 1.85E+00 – | 2.42E+02 \pm 2.55E+02 – | 3.12E+02 \pm 3.48E+02 – | 3.08E+00 \pm 1.47E+00 – | 8.19E+00 \pm 3.18E+00 – | 2.07E+00 \pm 1.03E+00 – | 1.31E+00 \pm 2.03E–01 |
| f_{21} | 7.80E+01 \pm 7.07E+01 – | 2.61E+03 \pm 1.69E+03 – | 3.85E+04 \pm 3.19E+04 – | 8.68E+01 \pm 8.99E+01 – | 1.76E+02 \pm 1.14E+02 – | 4.88E+01 \pm 6.11E+01 – | 7.57E+00 \pm 2.54E+01 |
| f_{22} | 9.22E+01 \pm 6.09E+01 – | 2.45E+02 \pm 1.47E+02 – | 1.16E+02 \pm 6.86E+01 – | 2.76E+01 \pm 1.78E+01 + | 1.47E+02 \pm 9.46E+01 – | 7.17E+01 \pm 6.23E+01 – | 5.47E+01 \pm 5.17E+01 |
| f_{23} | 3.15E+02 \pm 0.00E+00 \approx | 3.15E+02 \pm 0.00E+00 \approx | 3.15E+02 \pm 0.00E+00 \approx | 3.15E+02 \pm 0.00E+00 \approx | 3.14E+02 \pm 0.00E+00 + | 3.15E+02 \pm 0.00E+00 \approx | 3.15E+02 \pm 0.00E+00 |
| f_{24} | 2.23E+02 \pm 1.28E+00 – | 2.26E+02 \pm 1.62E+01 – | 2.22E+02 \pm 7.39E–01 – | 2.24E+02 \pm 1.06E+00 – | 2.24E+02 \pm 1.28E+00 – | 2.18E+02 \pm 9.12E+00 – | 2.00E+02 \pm 0.00E+00 |
| f_{25} | 2.03E+02 \pm 3.64E–02 \approx | 2.00E+02 \pm 1.80E–04 + | 2.05E+02 \pm 4.41E–01 – | 2.03E+02 \pm 4.41E–01 \approx | 2.01E+02 \pm 7.13E–01 + | 2.03E+02 \pm 2.86E–02 \approx | 2.03E+02 \pm 1.30E–02 |
| f_{26} | 1.00E+02 \pm 4.40E–02 \approx | 1.00E+02 \pm 2.75E–02 \approx | 1.04E+02 \pm 1.82E+01 – | 1.00E+02 \pm 1.55E–02 \approx | 1.00E+02 \pm 3.27E–02 \approx | 1.00E+02 \pm 1.36E–02 \approx | 1.00E+02 \pm 7.79E–03 |
| f_{27} | 3.02E+02 \pm 1.07E+01 – | 3.16E+02 \pm 4.04E+01 – | 3.08E+02 \pm 2.94E+01 – | 3.00E+02 \pm 0.00E+00 \approx | 3.41E+02 \pm 4.65E+01 – | 3.00E+02 \pm 0.00E+00 \approx | 3.00E+02 \pm 0.00E+00 |
| f_{28} | 9.11E+02 \pm 3.46E+01 – | 3.67E+02 \pm 6.46E–01 + | 7.87E+02 \pm 5.22E+01 + | 8.40E+02 \pm 1.40E+01 – | 3.75E+02 \pm 6.42E+00 + | 8.18E+02 \pm 3.32E+01 – | 8.00E+02 \pm 2.86E+01 |
| f_{29} | 7.32E+02 \pm 9.12E+01 – | 2.07E+02 \pm 1.86E+00 + | 1.39E+03 \pm 1.39E+02 – | 7.17E+02 \pm 5.13E+02 \approx | 2.11E+02 \pm 1.71E+00 + | 7.19E+02 \pm 3.36E+01 \approx | 7.17E+02 \pm 4.32E+00 |
| f_{30} | 1.60E+03 \pm 8.18E+02 + | 5.93E+02 \pm 1.78E+01 + | 1.19E+03 \pm 2.49E+02 + | 1.25E+03 \pm 6.20E+02 + | 5.25E+02 \pm 1.91E+02 + | 2.27E+03 \pm 7.75E+02 – | 2.40E+03 \pm 6.96E+02 |
| No. of + / \approx / – | | 1/9/20 | 7/8/15 | 6/2/22 | 10/9/11 | 12/4/14 | 2/11/17 |

“+”, “–”, and “ \approx ” denote that the result of the corresponding algorithm is better than, worse than, and similar to that of EDA-R1M-PR, respectively.

Table 2
The mean and standard deviation (mean \pm standard deviation) of FEVs obtained by 7 algorithms on CEC 2014 benchmark functions with 50D.

| Fun. | AMalGaM | IPOP-CMAES | BL-PSO-5 | L-SHADE | MVC_E_S_C | EDA-R1M | EDA-R1M-PR |
|----------|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|-------------------------|
| f_1 | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 5.10E+06 \pm 1.28E+06 $-$ | 1.24E+03 \pm 1.52E+03 $-$ | 7.15E+04 \pm 3.02E+04 $-$ | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 |
| f_2 | 1.15E-07 \pm 2.81E-07 $-$ | 0.00E+00 \pm 0.00E+00 \approx | 3.44E+03 \pm 2.35E+03 $-$ | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 |
| f_3 | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 4.23E+01 \pm 8.97E+01 $-$ | 0.00E+00 \pm 0.00E+00 \approx | 7.56E-02 \pm 2.01E-01 $-$ | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 |
| f_4 | 9.15E+01 \pm 6.74E+00 $-$ | 0.00E+00 \pm 0.00E+00 \approx | 8.64E+01 \pm 5.04E+00 $+$ | 5.89E+01 \pm 4.56E+01 $+$ | 3.19E+01 \pm 2.20E+01 $+$ | 8.70E+01 \pm 1.53E+01 $+$ | 8.98E+01 \pm 1.89E+01 |
| f_5 | 2.11E+01 \pm 2.63E-01 \approx | 2.11E+01 \pm 2.55E-02 \approx | 2.09E+01 \pm 5.07E-02 $+$ | 2.02E+01 \pm 4.59E-02 $+$ | 2.00E+01 \pm 4.85E-06 $+$ | 2.11E+01 \pm 3.27E-01 \approx | 2.11E+01 \pm 4.20E-02 |
| f_6 | 2.08E-02 \pm 1.04E-01 \approx | 2.95E-01 \pm 1.15E-01 $-$ | 9.22E-02 \pm 3.21E-01 $-$ | 2.64E-01 \pm 5.23E-01 $-$ | 4.91E+00 \pm 2.34E+00 $-$ | 9.92E-02 \pm 3.15E-01 $-$ | 3.52E-02 \pm 1.32E-01 |
| f_7 | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 \approx | 7.25E-04 \pm 2.57E-03 $-$ | 0.00E+00 \pm 0.00E+00 \approx | 0.00E+00 \pm 0.00E+00 |
| f_8 | 6.09E+00 \pm 2.32E+00 $-$ | 1.55E+00 \pm 1.19E+00 $-$ | 4.97E-01 \pm 8.16E-01 $+$ | 0.00E+00 \pm 0.00E+00 $+$ | 0.00E+00 \pm 0.00E+00 $+$ | 6.05E+00 \pm 2.19E+00 $-$ | 7.96E-01 \pm 8.82E-01 |
| f_9 | 5.49E+00 \pm 1.82E+00 $-$ | 1.79E+00 \pm 1.22E+00 $-$ | 7.10E+01 \pm 9.02E+00 $-$ | 1.14E+01 \pm 2.13E+00 $-$ | 4.45E+01 \pm 9.85E+00 $-$ | 3.70E+00 \pm 1.61E+00 $-$ | 3.32E-01 \pm 6.03E-01 |
| f_{10} | 9.04E+02 \pm 3.18E+02 $-$ | 1.39E+03 \pm 2.54E+03 $-$ | 3.63E+02 \pm 1.81E+02 $-$ | 1.22E-01 \pm 4.13E-02 $+$ | 1.32E-01 \pm 1.62E-01 $+$ | 4.14E+02 \pm 3.17E+02 $-$ | 4.62E+01 \pm 9.82E+01 |
| f_{11} | 9.69E+02 \pm 3.96E+02 $-$ | 1.42E+03 \pm 2.37E+03 $-$ | 4.46E+03 \pm 4.77E+02 $-$ | 3.22E+03 \pm 3.30E+02 $-$ | 3.29E+03 \pm 5.28E+02 $-$ | 4.19E+02 \pm 2.32E+02 $-$ | 2.60E+02 \pm 2.25E+02 |
| f_{12} | 3.08E+00 \pm 9.60E-01 $+$ | 3.37E+00 \pm 2.42E-01 $-$ | 8.77E-01 \pm 1.18E-01 $+$ | 2.19E-01 \pm 2.82E-02 $+$ | 5.44E-02 \pm 1.55E-02 $+$ | 3.31E+00 \pm 3.29E-01 $-$ | 3.23E+00 \pm 3.00E-01 |
| f_{13} | 1.24E-01 \pm 2.35E-02 $+$ | 1.27E-01 \pm 5.18E-02 $+$ | 2.86E-01 \pm 3.72E-02 $-$ | 1.60E-01 \pm 1.83E-02 $+$ | 2.73E-01 \pm 3.67E-02 $-$ | 5.80E-02 \pm 1.08E-02 $+$ | 1.82E-01 \pm 5.47E-02 |
| f_{14} | 4.32E-01 \pm 2.57E-02 $-$ | 3.61E-01 \pm 8.02E-02 $+$ | 2.65E-01 \pm 2.42E-02 $+$ | 2.97E-01 \pm 2.47E-02 $+$ | 2.82E-01 \pm 5.83E-02 $+$ | 4.30E-01 \pm 3.03E-02 $-$ | 3.98E-01 \pm 4.55E-02 |
| f_{15} | 5.33E+00 \pm 5.83E-01 $+$ | 4.56E+00 \pm 8.08E-01 $+$ | 1.48E+01 \pm 1.33E+00 $-$ | 5.15E+00 \pm 5.08E-01 $+$ | 4.87E+00 \pm 1.04E+00 $+$ | 4.74E+00 \pm 5.48E-01 $+$ | 6.36E+00 \pm 2.24E+00 |
| f_{16} | 2.02E+01 \pm 3.89E-01 \approx | 1.80E+01 \pm 2.84E+00 $+$ | 1.82E+01 \pm 4.73E-01 $+$ | 1.69E+01 \pm 4.81E-01 $+$ | 1.69E+01 \pm 7.60E-01 $+$ | 1.88E+01 \pm 4.34E-01 $+$ | 2.02E+01 \pm 4.60E-01 |
| f_{17} | 3.18E+02 \pm 2.24E+02 $-$ | 4.75E+03 \pm 1.05E+03 $-$ | 5.97E+05 \pm 2.10E+05 $-$ | 1.40E+03 \pm 5.13E+02 $-$ | 4.13E+03 \pm 2.05E+03 $-$ | 2.45E+02 \pm 1.26E+02 $-$ | 4.94E+01 \pm 1.70E+01 |
| f_{18} | 2.88E+01 \pm 6.94E+00 $-$ | 1.64E+03 \pm 3.97E+02 $-$ | 3.73E+02 \pm 3.68E+02 $-$ | 9.73E+01 \pm 1.38E+01 $-$ | 1.26E+02 \pm 3.21E+01 $-$ | 2.05E+01 \pm 7.74E+00 $-$ | 1.74E+00 \pm 1.10E+00 |
| f_{19} | 1.21E+01 \pm 6.47E-01 $-$ | 1.51E+01 \pm 2.21E+00 $-$ | 2.16E+01 \pm 9.78E+00 $-$ | 8.30E+00 \pm 1.81E+00 $+$ | 9.07E+00 \pm 2.54E+00 $+$ | 1.11E+01 \pm 5.41E-01 $-$ | 1.03E+01 \pm 1.07E+00 |
| f_{20} | 5.92E+00 \pm 1.82E+00 $-$ | 1.08E+03 \pm 3.91E+02 $-$ | 2.57E+02 \pm 1.41E+02 $-$ | 1.39E+01 \pm 4.56E+00 $-$ | 1.95E+02 \pm 6.68E+01 $-$ | 2.91E+00 \pm 1.30E+00 $-$ | 2.08E+00 \pm 2.81E-01 |
| f_{21} | 2.54E+02 \pm 7.36E+01 $-$ | 4.91E+03 \pm 2.90E+03 $-$ | 3.80E+05 \pm 1.44E+05 $-$ | 5.15E+02 \pm 1.49E+02 $-$ | 1.65E+03 \pm 8.74E+02 $-$ | 2.46E+02 \pm 1.05E+02 $-$ | 1.52E+02 \pm 9.25E+01 |
| f_{22} | 3.74E+01 \pm 3.36E+01 $-$ | 1.30E+03 \pm 5.59E+02 $-$ | 2.64E+02 \pm 1.30E+02 $-$ | 1.14E+02 \pm 7.50E+01 $-$ | 4.45E+02 \pm 1.83E+02 $-$ | 2.90E+01 \pm 2.47E+01 \approx | 3.04E+01 \pm 1.46E+00 |
| f_{23} | 3.44E+02 \pm 0.00E+00 \approx | 3.37E+02 \pm 0.00E+00 \approx | 3.44E+02 \pm 0.00E+00 \approx | 3.44E+02 \pm 0.00E+00 \approx | 3.37E+02 \pm 0.00E+00 $+$ | 3.44E+02 \pm 0.00E+00 \approx | 3.44E+02 \pm 0.00E+00 |
| f_{24} | 2.72E+02 \pm 1.89E+00 $-$ | 2.68E+02 \pm 1.77E+01 \approx | 2.58E+02 \pm 4.07E+00 $+$ | 2.75E+02 \pm 6.61E-01 $-$ | 2.73E+02 \pm 1.91E+00 $-$ | 2.68E+02 \pm 1.51E+00 $-$ | 2.67E+02 \pm 8.28E-01 |
| f_{25} | 2.05E+02 \pm 1.51E-01 \approx | 2.00E+02 \pm 4.89E-03 $+$ | 2.10E+02 \pm 7.36E-01 $-$ | 2.05E+02 \pm 3.65E-01 \approx | 2.11E+02 \pm 9.90E+00 $-$ | 2.05E+02 \pm 1.96E-01 \approx | 2.05E+00 \pm 9.74E-02 |
| f_{26} | 1.01E+02 \pm 2.10E-01 $-$ | 1.00E+02 \pm 4.72E-02 \approx | 1.47E+02 \pm 5.08E+01 $-$ | 1.02E+02 \pm 1.40E+01 $-$ | 1.00E+00 \pm 1.07E-01 \approx | 1.00E+02 \pm 8.68E-02 \approx | 1.00E+02 \pm 5.30E-02 |
| f_{27} | 3.15E+02 \pm 2.17E+01 $+$ | 3.34E+02 \pm 7.35E+01 $-$ | 3.24E+02 \pm 2.85E+01 \approx | 3.33E+02 \pm 3.03E+01 $-$ | 4.23E+02 \pm 5.56E+01 $-$ | 3.20E+02 \pm 2.57E+01 \approx | 3.25E+02 \pm 2.73E+01 |
| f_{28} | 1.26E+03 \pm 7.92E+01 $-$ | 3.74E+02 \pm 1.32E+01 $+$ | 1.14E+03 \pm 4.20E+01 $-$ | 1.11E+03 \pm 2.91E+01 $-$ | 3.67E+02 \pm 9.11E+00 $+$ | 1.12E+03 \pm 3.71E+01 $-$ | 1.10E+03 \pm 3.70E+01 |
| f_{29} | 8.95E+02 \pm 1.39E+02 $-$ | 2.18E+02 \pm 2.54E+00 $+$ | 1.36E+03 \pm 1.82E+02 $-$ | 7.95E+02 \pm 2.40E+01 $+$ | 2.21E+02 \pm 3.49E+00 $+$ | 8.04E+02 \pm 3.42E+01 \approx | 8.08E+02 \pm 4.25E+01 |
| f_{30} | 9.04E+03 \pm 3.95E+02 $-$ | 1.45E+03 \pm 3.80E+02 $+$ | 9.09E+03 \pm 3.05E+02 $-$ | 8.66E+03 \pm 4.13E+02 \approx | 1.10E+03 \pm 2.79E+00 $+$ | 8.72E+03 \pm 4.27E+02 \approx | 8.69E+03 \pm 3.78E+02 |
| No. of | +/- \approx /- | 4/8/18 | 10/7/13 | 7/3/20 | 11/6/13 | 13/2/15 | 4/12/14 |

“+”, “-”, and “ \approx ” denote that the result of the corresponding algorithm is better than, worse than, and similar to that of EDA-R1M-PR, respectively.

deterioration on one function when D is increased to 50. MVC_E_S_C finds the optimal solutions of two functions with 30D, but the corresponding number decreases to one when $D = 50$. BL-PSO-5 performs worst since it fails to find desirable solutions for the three functions. The results indicate that the model with rank-one modification in EDA-R1M could effectively capture the structural characteristics of the optimization problem and thus improve the search efficiency.

- 2) For multimodal functions f_4 – f_{16} , EDA-R1M-PR consistently provides the results of higher quality than other algorithms on four functions, i.e. f_6 , f_7 , f_9 , and f_{11} . When $D = 30$, EDA-R1M-PR performs no worse than AMaLGaM on all the 13 functions, and has an edge over IPOP-CMA-ES, BL-PSO-5, L-SHADE, MVC_E_S_C and EDA-R1M on 7, 8, 4, 5 and 8 functions, respectively. When $D = 50$, EDA-R1M-PR is statistically better than the above six algorithms on 6, 6, 6, 3, 5 and 7 functions, respectively, and is defeated by them on 3, 5, 6, 9, 8 and 4 functions, respectively. These results indicate that EDA-R1M-PR performs slightly worse than L-SHADE and MVC_E_S_C, but better than the other algorithms on these multimodal functions.
- 3) Hybrid functions f_{17} – f_{22} are a kind of partial separable functions, which can more closely approximate the real-world optimization problems (Tanabe and Fukunaga, 2014). Clearly, EDA-R1M-PR demonstrates the best performance on this group of functions since it outperforms almost all its competitors on all the 6 functions, except that it is defeated by L-SHADE on f_{22} with 30D and is outperformed by L-SHADE and MVC_E_S_C on f_{19} with 50D. Moreover, the results of EDA-R1M-PR are usually better than that of other algorithms over one order of magnitude.
- 4) As for composition functions f_{23} – f_{30} , the results obtained by the seven algorithms are far away from the optima. Nevertheless, EDA-R1M-PR still shows certain superiority over other algorithms. It performs no worse than EDA-R1M on all the 8 functions with both 30D and 50D. Compared to AMaLGaM, BL-PSO-5 and L-SHADE, EDA-R1M-PR is only surpassed by them on 1, 2 and 1 functions with 30D, respectively, and on 1, 1 and 1 function with 50D, respectively. However, EDA-R1M-PR fails to defeat IPOP-CMA-ES and MVC_E_S_C on a majority of functions under both cases of 30D and 50D. The success of the two algorithms can be attributed to the capability to escape from local optima through the restart mechanism in IPOP-CMA-ES and the power of MVC framework.

The last rows of Tables 1 and 2 summarize the comparison results, it is evident that EDA-R1M-PR performs significantly better than AMaLGaM, IPOP-CMA-ES, BL-PSO-5 and EDA-R1M, and has an edge over L-SHADE and MVC_E_S_C. To further detect the performance differences of the above algorithms, the Friedman test was carried out to obtain their rankings on this set of functions. As shown in Table 3, EDA-R1M-PR has the best ranking among these algorithms, EDA-R1M also performs surprisingly well and ranks the fourth when $D = 30$ and the second when $D = 50$.

In summary, EDA-R1M-PR performs the best among the seven

algorithms in case of this set of functions with 30D and 50D, which indicates that its two components do play an important role in improving its efficiency. The rank-one modification operation significantly enhances the search ability of GEDA by effectively adjusting the search scope and search direction. The population reduction strategy further improves the performance of EDA-R1M, especially on complicated functions, by taking advantage of its capability to balance the exploration and exploitation with the evolution.

5. Conclusion

In this paper, a novel EDA variant named EDA-R1M-PR is proposed for continuous optimization. EDA-R1M-PR first estimates a more promising search center by shifting the weighted mean of high-quality solutions along the fitness improvement direction through an adaptive line search method. Then it estimates a new covariance matrix with the shifted mean as the center, which is essentially the rank-one modification of the original covariance matrix. Theoretical and experimental studies both demonstrate that the R1M operation could effectively adjust the search scope and search direction of GEDA and thus improving its search efficiency. Moreover, we further extend EDA-R1M with a population reduction strategy that linearly reduces the population size throughout the evolution process. The PR strategy could adaptively balance the exploration and exploitation ability of EDA-R1M in different search stages and thus significantly improves its global optimization ability, especially on complicated multimodal functions. Experiments on a set of benchmark functions with 30D and 50D indicate that EDA-R1M-PR is robust to its parameters and competitive to several efficient EAs.

Our future work will focus on further investigating the properties of the rank-one modification operation and reducing the computational complexity of the model updating and sampling process. It is also of significance to evaluate the performance of EDA-R1M-PR on some challenging real-world optimization problems.

Acknowledgements

This work was supported by the National Natural Science Foundation of China [grant numbers 61873199, 61503300]; the China Post-Doctoral Science Foundation [grant numbers 2014M560784, 2016T90922]; the Fundamental Research Funds for the Central Universities [grant number xzy022019028]; and the Science and Technology Foundation of Foshan, China [grant number 2016AG101813].

References

- Ahn, C.W., An, J., Yoo, J.-C., 2012. Estimation of particle swarm distribution algorithms: combining the benefits of PSO and EDAs. *Inf. Sci. (Nij)* 192 (Supplement C), 109–119.
- Auger, A., Hansen, N., 2005. A restart CMA evolution strategy with increasing population size. 2005 IEEE Congress on Evolutionary Computation. pp. 1769–1776.
- Bosman, P.A.N., Thierens, D., 2000. Expanding from discrete to continuous estimation of distribution algorithms: the IDEA. *International Conference on Parallel Problem Solving from Nature*. pp. 767–776.
- Bosman, P.A.N., Gahl, J., Rothlauf, F., 2007. SDR: a better trigger for adaptive variance scaling in normal EDAs. *Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation*. pp. 492–499.
- Bosman, P.A.N., Gahl, J., Thierens, D., 2008. Enhancing the performance of maximum-likelihood gaussian EDAs using anticipated mean shift. *International Conference on Parallel Problem Solving from Nature*. pp. 133–143.
- Bunch, J.R., Nielsen, C.P., Sorensen, D.C., 1978. Rank-one modification of the symmetric eigenproblem. *Numer. Math.* 31 (1), 31–48.
- Cai, Y., Sun, X., Xu, H., Jia, P., 2007. Cross entropy and adaptive variance scaling in continuous EDA. *Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation*. pp. 609–616.
- Chen, T., Tang, K., Chen, G., Yao, X., 2010. Analysis of computational time of simple estimation of distribution algorithms. *IEEE Trans. Evol. Comput.* 14 (1), 1–22.
- Chen, Y., Sun, X., Gong, D., Zhang, Y., Choi, J., Klasky, S., 2017a. Personalized search inspired fast interactive estimation of distribution algorithm and its application. *IEEE Trans. Evol. Comput.* 21 (4), 588–600.
- Chen, X., Tianfield, H., Mei, C., Du, W., Liu, G., 2017b. Biogeography-based learning

Table 3

Ranking of 7 algorithms on functions with 30D and 50D according to the Friedman test.

| 30D | | 50D | |
|-------------|---------|-------------|---------|
| Algorithms | Ranking | Algorithms | Ranking |
| EDA-R1M-PR | 3.02 | EDA-R1M-PR | 3.33 |
| L-SHADE | 3.18 | EDA-R1M | 3.53 |
| MVC_E_S_C | 3.70 | L-SHADE | 3.63 |
| EDA-R1M | 3.92 | MVC_E_S_C | 3.90 |
| IPOP-CMA-ES | 4.17 | IPOP-CMA-ES | 3.97 |
| AMaLGaM | 4.68 | AMaLGaM | 4.38 |
| BL-PSO-5 | 5.33 | BL-PSO-5 | 5.25 |

- particle swarm optimization. *Soft Comput.* 21 (24), 7519–7541.
- Cohen, J., 1988. *Statistical Power Analysis for the Behavioral Sciences* Lawrence Erlbaum Associates. Hillsdale, NJ, pp. 20–26.
- De Bonet, J.S., Isbell, C.L., Viola, P., 1997. MIMIC: finding optima by estimating probability densities. *Proceedings of the 9th International Conference on Neural Information Processing Systems* 424–431.
- de Oca, M.A.M., Stutzle, T., Van den Eenden, K., Dorigo, M., 2011. Incremental social learning in particle swarms. *IEEE Trans. Syst. Man Cybern. Part B* 41 (2), 368–384.
- Ding, N., Zhou, S.-D., Sun, Z.-Q., 2008. Histogram-based estimation of distribution algorithm: a competent method for continuous optimization. *J. Comput. Sci. Technol.* 23 (1), 35–43.
- Gao, S., de Silva, C.W., 2016. A modified estimation distribution algorithm based on extreme elitism. *Biosystems* 150, 149–166.
- Gao, S., de Silva, C.W., 2018. Estimation distribution algorithms on constrained optimization problems. *Appl. Math. Comput.* 339, 323–345.
- García-Martínez, C., Gutiérrez, P.D., Molina, D., Lozano, M., Herrera, F., 2017. Since CEC 2005 competition on real-parameter optimisation: a decade of research, progress and comparative analysis's weakness. *Soft Comput.* 21 (19), 5573–5583.
- Grahl, J., Minner, S., Rothlauf, F., 2005. Behaviour of UMDAc with truncation selection on monotonous functions. *2005 IEEE Congress on Evolutionary Computation*. pp. 2553–2559.
- Grahl, J., Bosman, P.A.N., Rothlauf, F., 2006. The correlation-triggered adaptive variance scaling IDEA. *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation*. pp. 397–404.
- Handa, H., 2007. The effectiveness of mutation operation in the case of Estimation of Distribution Algorithms. *Biosystems* 87 (no. 2), 243–251.
- Hansen, N., Ostermeier, A., 2001. Completely derandomized self-adaptation in evolution strategies. *Evol. Comput.* 9 (2), 159–195.
- Hauschild, M., Pelikan, M., 2011. An introduction and survey of estimation of distribution algorithms. *Swarm Evol. Comput.* 1 (3), 111–128.
- Kabán, A., Bootkrajang, J., Durrant, R.J., 2016. Toward large-scale continuous EDA: a random matrix theory perspective. *Evol. Comput.* 24 (2), 255–291.
- Larrañaga, P., Lozano, J.A., 2001. *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*, vol. 2 Springer Science & Business Media.
- Larrañaga, P., Etxeberria, R., Lozano, J.A., Peña, J.M., 1999. *Optimization by Learning and Simulation of Bayesian and Gaussian Networks*. Univ. of the Basque Country, Spain Tech. Rep. KZZA-IK-4-99.
- Li, H., Hong, Y., Kwong, S., 2011. Subspace estimation of distribution algorithms: to perturb part of all variables in estimation of distribution algorithms. *Appl. Soft Comput.* 11 (3), 2974–2989.
- Liang, J.J., Qu, B.Y., Suganthan, P.N., 2013. Problem Definitions and Evaluation Criteria for the CEC 2014 Special Session and Competition on Single Objective Real-parameter Numerical Optimization. *Comput. Intell. Lab. Zhengzhou Univ. Zhengzhou, China, Nanyang Technol. Univ. Singapore, Tech. Rep.*
- Lima, C.F., Lobo, F.G., Pelikan, M., Goldberg, D.E., 2011. Model accuracy in the Bayesian optimization algorithm. *Soft Comput.* 15 (7), 1351–1371.
- Lobo, F.G., Lima, C.F., 2007. A review of adaptive population sizing schemes in genetic algorithms. *Proceedings of the 7th Annual Workshop on Genetic and Evolutionary Computation*. pp. 228–234.
- Loshchilov, I., 2013. CMA-ES with restarts for solving CEC 2013 benchmark problems. *2013 IEEE Congress on Evolutionary Computation (CEC)*. pp. 369–376.
- Mallipeddi, R., Suganthan, P.N., Pan, Q.K., Tasgetiren, M.F., 2011. Differential evolution algorithm with ensemble of parameters and mutation strategies. *Appl. Soft Comput.* 11 (2), 1679–1696.
- Mühlenbein, H., Paaß, G., 1996. From recombination of genes to the estimation of distributions I. Binary parameters. *Proceedings of the 4th International Conference on Parallel Problem Solving from Nature* 178–187.
- Pelikan, M., Goldberg, D., Cantú-Paz, E., 1999. BOA: The Bayesian Optimization Algorithm. *Dept. Gen. Eng., Illinois Genetic Algorithms Laboratory (IlligAL), Urbana, IL, USA Tech. Rep.* 99003.
- Rastegar, R., 2010. On the optimal convergence probability of univariate estimation of distribution algorithms. *Evol. Comput.* 19 (August (2)), 225–248.
- Ren, Z., Liang, Y., Wang, L., Zhang, A., Pang, B., Li, B., 2018. Anisotropic adaptive variance scaling for Gaussian estimation of distribution algorithm. *Knowledge-Based Syst.* 146, 142–151.
- Sanyang, M.L., Kaban, A., 2014. Multivariate cauchy EDA optimisation. *International Conference on Intelligent Data Engineering and Automated Learning (IDEAL 2014)*. pp. 449–456.
- Shi, W., Chen, W., Lin, Y., Gu, T., Kwong, S., Zhang, J., 2019. An adaptive estimation of distribution algorithm for multi-policy insurance investment planning. *IEEE Trans. Evol. Comput.* 23 (1), 1–14.
- Tanabe, R., Fukunaga, A., 2013. Success-history based parameter adaptation for differential evolution. *2013 IEEE Congress on Evolutionary Computation*. pp. 71–78.
- Tanabe, R., Fukunaga, A.S., 2014. Improving the search performance of SHADE using linear population size reduction. *2014 IEEE Congress on Evolutionary Computation (CEC)*. pp. 1658–1665.
- Wang, Y., Li, H.-X., Huang, T., Li, L., 2014. Differential evolution based on covariance matrix learning and bimodal distribution parameter setting. *Appl. Soft Comput.* 18, 232–247.
- Wu, C., Wang, L., 2018. A multi-model estimation of distribution algorithm for energy efficient scheduling under cloud computing system. *J. Parallel Distrib. Comput.* 117, 63–72.
- Xie, P., Chen, T., 1990. The positive definiteness of the covariance matrix of continuous sample. *Acta Scientiarum Naturalium Universitatis Sunyatseni* 29 (1), 102–104.
- Xu, H., Jiang, M., Xu, K., 2015. Archimedean copula estimation of distribution algorithm based on artificial bee colony algorithm. *J. Syst. Eng. Electron.* 26 (2), 388–396.
- Yang, Q., Chen, W.N., Li, Y., Chen, C.L.P., Xu, X.M., Zhang, J., 2017. Multimodal estimation of distribution algorithms. *IEEE Trans. Cybern.* 47 (3), 636–650.
- Yin, Q., 2003. The inverse problem of rank-1 modification of real symmetric matrices. *J. Nantong Inst. Technol. (Nat. Sci.)* 2 (4), 11–14.
- Yuan, B., Gallagher, M., 2005. On the importance of diversity maintenance in estimation of distribution algorithms. *Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation*. pp. 719–726.
- Zhang, S.X., Zheng, S.Y., Zheng, L.M., 2017. An efficient multiple variants coordination framework for differential evolution. *IEEE Trans. Cybern.* 47 (9), 2780–2793.
- Zhao, H., 2013. Areas(Volumes) of n-dimension ellipsoid by quadratic curve(Surface) enclosed. *Math. Pract. Theory* 43 (10), 279–282.
- Zhao, F., Shao, Z., Wang, J., Zhang, C., 2016. A hybrid differential evolution and estimation of distribution algorithm based on neighbourhood search for job shop scheduling problems. *Int. J. Prod. Res.* 54 (February (4)), 1039–1060.
- Zhou, A., Sun, J., Zhang, Q., 2015. An estimation of distribution algorithm with cheap and expensive local search methods. *IEEE Trans. Evol. Comput.* 19 (6), 807–822.