Article

An Adaptive Covariance Scaling Estimation of Distribution Algorithm

Qiang Yang [1,*], Yong Li [1], Xu-Dong Gao [1], Yuan-Yuan Ma [2], Zhen-Yu Lu [1], Sang-Woon Jeon [3] and Jun Zhang [3,4]

[1] School of Artificial Intelligence, Nanjing University of Information Science and Technology, Nanjing 210044, China; 20201249539@nuist.edu.cn (Y.L.); 003103@nuist.edu.cn (X.-D.G.); 001114@nuist.edu.cn (Z.-Y.L.)

[2] College of Computer and Information Engineering, Henan Normal University, Xinxiang 453007, China; 121100@htu.edu.cn

[3] Department of Electrical and Electronic Engineering, Hanyang University, Ansan 15588, Korea; sangwoonjeon@hanyang.ac.kr (S.-W.J.); junzhang@ieee.org (J.Z.)

[4] Department of Computer Science and Information Engineering, Chaoyang University of Technology, Taichung 413310, Taiwan

[*] Correspondence: qiang_yang@nuist.edu.cn or mmmyq@126.com

Abstract: Optimization problems are ubiquitous in every field, and they are becoming more and more complex, which greatly challenges the effectiveness of existing optimization methods. To solve the increasingly complicated optimization problems with high effectiveness, this paper proposes an adaptive covariance scaling estimation of distribution algorithm (ACSEDA) based on the Gaussian distribution model. Unlike traditional EDAs, which estimate the covariance and the mean vector, based on the same selected promising individuals, ACSEDA calculates the covariance according to an enlarged number of promising individuals (compared with those for the mean vector). To alleviate the sensitivity of the parameters in promising individual selections, this paper further devises an adaptive promising individual selection strategy for the estimation of the mean vector and an adaptive covariance scaling strategy for the covariance estimation. These two adaptive strategies dynamically adjust the associated numbers of promising individuals as the evolution continues. In addition, we further devise a cross-generation individual selection strategy for the parent population, used to estimate the probability distribution by combing the sampled offspring in the last generation and the one in the current generation. With the above mechanisms, ACSEDA is expected to compromise intensification and diversification of the search process to explore and exploit the solution space and thus could achieve promising performance. To verify the effectiveness of ACSEDA, extensive experiments are conducted on 30 widely used benchmark optimization problems with different dimension sizes. Experimental results demonstrate that the proposed ACSEDA presents significant superiority to several state-of-the-art EDA variants, and it preserves good scalability in solving optimization problems.

Keywords: estimation of distribution algorithm; covariance scaling; gaussian distribution; meta-heuristic algorithm; problem optimization

1. Introduction

Optimization problems are ubiquitous in daily life and industrial engineering [1,2], such as protein structure prediction [3], community detection [4], control of pollutant spreading [5] and multi-compartment electric vehicle routing [6]. These optimization problems often preserve characteristics such as non-convex, discontinuous, and non-differentiable [7–10], which greatly challenge the effectiveness of traditional gradient-based optimization algorithms or even make them infeasible [11]. In particular, in the era of big data and the Internet of Things, optimization problems are becoming more and more complex due to the increase in dimensionality [12–14]. For instance, some unimodal problems become multimodal with many local optima [15], while some multimodal problems

become more complicated with an increasing number of wide and flat local areas [16–18]. Such complicated optimization problems are becoming more and more common nowadays, and thus, it is urgent to develop effective optimization algorithms to solve them, so as to promote the development of related fields.

As a kind of gradient-free meta-heuristic algorithm, estimation of distribution algorithm (EDA) mainly maintains a population of individuals to iteratively search the solution space, with each individual representing a feasible solution [19]. During each generation, it selects a number of promising individuals to estimate the probability distribution of the population, and it then randomly samples a new population of solutions based on the estimated probability distribution [20,21]. Due to the randomness in sampling the offspring, EDA preserves high diversity and strong global search ability [22]. Therefore, a lot of researchers have paid extensive attention to developing effective EDAs, and, consequently, not only have EDAs been applied to solve various optimization problems, such as multimodal optimization problems [23] and multi-objective optimization problems [24], but also they have been employed to solve many real-world problems, such as multi-policy insurance investment planning [25] and multi-source heterogeneous user-generated content-driven interactive [22].

In the literature, most EDAs utilize the Gaussian distribution model to evaluate the probability distribution of the population, which is then adopted to sample new solutions [26,27]. During the estimation of the distribution, based on whether the correlation between variables is considered, the current Gaussian estimations of distribution algorithms (GEDAs) are mainly divided into two categories [20,27,28], namely univariate GEDAs (UGEDAs) [29–31] and multivariate GEDAs (MGEDAs) [32–36].

UGEDAs [30] consider that each variable is independent on each other. Therefore, the probability distribution of each variable is estimated individually. The most advantageous property of UGEDAs is that the computational cost of the distribution estimation and the offspring sampling is low [31]. However, their effectiveness deteriorates drastically when confronted with optimization problems with interacted variables [29].

Different from UGEDAs, MGEDAs take the correlation among variables into consideration [34], which is realized by estimating the covariance among all variables. With the covariance matrix, MGEDAs could capture the structure of the optimization problem and thus implicitly offer useful information to direct the search of the population [35]. Due to this advantage, MGEDAs achieve much better performance than UGEDAs, especially on problems with many interactive variables [20]. As a result, MGEDAs have been extensively researched in the literature [37–39]. However, such superiority of MGEDAs is at the sacrifice of efficiency, as calculating the covariance among all variables is very time-consuming [32].

In MGEDAs, the parameters (i.e., the mean vector and the covariance matrix) of the probability distribution are usually estimated based on a certain number of promising individuals [28]. Specifically, in the probability distribution, the mean vector plays a key role in controlling the center of the offspring to be sampled, while the covariance takes charge of the range of the offspring around the center. In other words, the mean vector makes a crucial influence on the convergence of the population to the optimal areas, while the covariance affects the population diversity [40]. Therefore, to maintain high search diversity for EDAs, many researchers have designed variance or covariance scaling methods [37,41–43] to enlarge the sampling range of the estimated probability distribution. However, on the one side, the basic covariance (variance) is still estimated on the same promising individuals selected for the estimation of the mean vector in most existing GEDAs. Therefore, after the covariance (variance) scaling, implemented by multiplying a scaling factor on the estimated covariance (variance) in most covariance scaling methods, the learned structure of the optimization problem, by them, remains unchanged; on the other side, most existing covariance (variance) scaling methods enlarge the estimated covariance (variance) the same degree in different directions.

To remedy the above shortcomings, this paper devises an adaptive covariance scaling method for MGEDAs, leading to an adaptive covariance scaling estimation of distribution algorithm (ACSEDA). Specifically, for the estimation of the mean vector in the probability distribution, it is the same as existing MGEDAs, namely estimating it based on a certain number of promising individuals. However, for the estimation of the covariance in the probability distribution, different from existing MGEDAs, ACSEDA first adaptively enlarges the number of promising individuals, and then, it calculates the covariance on the basis of the scaled promising individuals. In this way, the sampling range of the estimated probability distribution could be enlarged, which is helpful for sampling diversified offspring. As a result, the search diversity of EDA could be amplified, and thus, the chance of falling into local areas could be declined.

As a whole, the main contributions of this paper are summarized as follows:

(1) An adaptive covariance scaling method is proposed to adaptively enlarge the sampling range of the estimated probability distribution. Different from most existing covariance scaling methods, we scale the covariance by calculating the covariance based on an amplified number of promising individuals. As a result, not only could the learned structure of the optimization problem captured by the algorithm be improved but the covariance in different directions is also scaled differently. In this way, it is expected that the sampled offspring are not only of high quality, but they are also diversified in different areas.

(2) An adaptive selection of promising individuals for the estimation of the mean vector is further designed by adaptively decreasing the selection ratio, which is the number of the selected promising individuals out of the whole population. In this way, the estimated mean vector, namely the center of the offspring to be sampled, is gradually close to the promising areas that the current population covers. Therefore, the search process is gradually biased toward exploiting the solution space to refine the solution accuracy. However, it should be mentioned that such a bias is not greedy and not at the serious sacrifice of the population diversity because of the aforementioned covariance scaling technique.

(3) A cross-generation individual selection scheme, for the parent population to estimate the probability distribution, is devised by combining the sampled offspring in the last generation and the one in the current generation to select parent individuals for the next generation. Instead of directly utilizing the sampled offspring as the parent population for the next generation in most existing MGEDAs, the proposed ACSEDA combines the sampled offspring in the last generation and the one in the current generation to select the top half best individuals to form the parent population for the next generation. In this way, the parent population formed is neither too crowded nor too scattered, and thus, the estimated probability distribution is of high quality to sample slightly diversified offspring to approach the optimal areas.

(4) With the above mechanisms, the proposed ACSEDA is expected to compromise intensification and diversification of the search process well to explore and exploit the solution space and could thus achieve promising performance.

To verify the effectiveness of the proposed ACSEDA, this paper conducts extensive experiments on the widely used CEC2014 [44] benchmark optimization problems, with different dimension sizes, by comparing ACSEDA to 7 state-of-the-art GEDAs. In addition, deep investigations on the components of ACSEDA are also taken to observe what contributes to its promising performance in solving optimization problems.

The remainder of this study is organized as follows: Section 2 reviews related works on GEDAs; then, the proposed ACSEDA is elucidated in detail in Section 3; in Section 4, extensive experiments are conducted to verify the effectiveness of the developed ACSEDA; last, in Section 5, conclusions are presented.

2. Related Work

2.1. Basic GEDA

The overall framework of a general GEDA is outlined in Algorithm 1. As a whole, the basic principle of GEDA is to iteratively build a Gaussian probability distribution model, based on a certain number of promising individuals selected from the current population, and then sample new individuals based on the built probability model for the next generation [21].

Algorithm 1: The Procedure of GEDA.

Input: population size *PS*, selection ratio *sr*;
1: Set $g = 0$, and randomly initialize the population P^g;
2: Obtain the global best solution *Gbest*;
3: **Repeat**
4:   Select $\lceil sr * PS \rceil$ promising solutions S^g from P^g;
5:   Build a Gaussian probability distribution model G^g based on S^g;
6:   Randomly generate a new population P^{g+1} by sampling from G^g;
7:   Update the global best solution *Gbest*;
8:   $g = g + 1$;
9: **Until** *the stopping criterion is met*.
Output: the global best solution *Gbest*;

Specifically, as shown in Algorithm 1, given that the population size is *PS* and the selection ratio is *sr* (which is the number of selected promising individuals out of the whole population), the number of selected promising individuals in each generation is $s = \lceil sr * PS \rceil$. After *PS* individuals are initialized randomly and evaluated accordingly, as shown in Line 1, the global best solution found so far is obtained (as shown in Line 2). Subsequently, it comes to the main iteration of the algorithm. First, a set (*S*) of *s* promising individuals are selected from the current population (Line 4). Then, a Gaussian probability distribution model is estimated based on the selected individuals (Line 5). After that, *PS* new individuals are randomly sampled based on the estimated Gaussian distribution to form a new population (Line 6). Subsequently, the newly generated individuals are evaluated, and the global best solution is updated. The above process proceeds repeatedly until the termination condition is met. At last, the found global best solution is output.

In GEDAs, the key component is the way to estimate the probability distribution. Different manners of probability distribution estimation result in different kinds of GEDAs. In the literature, based on whether the linkage between variables is considered, existing GEDAs are mainly classified into two categories [20,27,28], namely univariate GEDAs (UGEDAs) [29–31,45,46] and multivariate GEDAs [27,32–36].

(1)   *UGEDAs*: In UGEDAs [29,45,46], each variable is considered to be separable and independent on each other. As a result, the probability distribution of *D* variables can be estimated separately, and the joint probability distribution of *D* variables is computed as follows:

$$P(x^1, x^2, \ldots, x^D) = \prod_{i=1}^{D} P(x^i). \tag{1}$$

where $P(x^i)$ is the probability distribution of the *i*th variable, which is estimated as:

$$P(x^i) = \frac{1}{\sigma^i \sqrt{2\pi}} e^{-\frac{(x^i - \mu^i)^2}{2\sigma^i}} \tag{2}$$

where u^i and σ^i are the mean value and the variance of the *i*th variable respectively, which are calculated as follows:

$$\mu^i = \frac{1}{s} \sum_{j=1}^{s} S_j{}^i \tag{3}$$

$$\sigma^i = \sqrt{\left(\frac{1}{s-1}\sum_{j=1}^{s}(S_j{}^i - \mu^i)^2\right)} \tag{4}$$

where S is the set of the selected promising individuals, $S_j{}^i$ is the ith dimension of the jth promising individual in S, and D denotes the dimension size of the optimization problem. Based on the estimated probability distribution of each variable, a new solution can be constructed by randomly sampling a new value of each variable separately, based on the associated probability distribution.

(2) *MGEDAs*: In MGEDAs [33–36], the correlations between variables are taken into consideration to estimate the probability distribution. Consequently, different from UGEDAs, the probability distribution of D variables in MGEDAs is estimated together, and the joint probability distribution of D variables is computed as follows:

$$P(x^1, x^2, \ldots, x^D) = \frac{1}{\sqrt{(2\pi)^D |C|}} e^{(-\frac{1}{2}(X-\mu)^T C^{-1}(X-\mu))} \tag{5}$$

where u is the mean vector of the multivariate Gaussian distribution, which is calculated by Equation (3). C is the covariance matrix, which is calculated as follows:

$$C = \frac{1}{s-1}(S - \mu)(S - \mu)^T \tag{6}$$

Based on the estimated joint probability distribution, a new solution is constructed by jointly sampling values for all variables, randomly, from the multivariate Gaussian distribution model. In general, to make the sampling of new solutions simple, a modified version presented below is usually utilized to generate the offspring in most MGEDAs [32,35]:

$$\begin{aligned} X &= \mu + A\Lambda Z \\ Z &\sim N(0,1) \end{aligned} \tag{7}$$

$$C = A\Lambda^2 A^T \tag{8}$$

where A is the eigenvector matrix of C, and Λ is the diagonal matrix whose entries are the square root of the eigenvalues of C. Z is a real number vector, each value of which is randomly sampled from a standard normal distribution separately.

With respect to the computational cost, UGEDAs are less time-consuming, while MGEDAs take more computational cost due to the calculation of the covariance matrix [33–35]. However, in terms of the optimization performance, MGEDAs show much better performance, especially on problems with many interacted variables, while UGEDAs only present promising performance on separable optimization problems [29,30,47]. This is because MGEDAs could capture the interaction between variables and thus evolve the population more effectively than UGEDAs [27,37,48].

2.2. Recent Advance of GEDAs

During the optimization, one crucial challenge that most existing GEDAs encounter is the rapid shrinkage of the variance (or the covariance) [20,42,43], which leads to the quickly narrowed sampling range of the probability distribution. This may lead to a quick loss of the search diversity and thus may result in premature convergence and falling into local areas. To remedy this shortcoming, researchers have devoted plenty of attention to designing novel mechanisms to improve the quality of the probability distribution in GEDAs [38,49–51].

In [51], the authors demonstrated empirically that high diversity maintenance is very crucial for EDAs to achieve satisfactory performance. Then, based on the findings, they further developed a novel three-step method by combining clustering methods with EDAs to search for the optimal areas with high diversity. To prevent premature convergence

in EDAs, Pošík [52] directly multiplied a constant factor on the estimated variance of the Gaussian distribution in each generation to enlarge the sampling range. In [43], Grahl et al. proposed a correlation-triggered adaptive variance scaling strategy to reduce the risk of premature convergence and then embedded it into the iterated density–estimation evolutionary algorithm (IDEA). Specifically, similar to [52], the proposed method multiplies a factor to the estimated variance. The difference lies in such a factor not being constant but dynamically adjusted during the evolution, based on whether the global best solution is improved or not. In addition, such adjustment is triggered based on the correlation between the ranks of the normal density and the fitness of the selected solutions. To further trigger the dynamic adjustment of the scaling factor properly, in [42], Bosman et al. proposed a novel indicator, named Standard–Deviation Ratio (SDR), to trigger the adjustment adaptively. Specifically, based on this indicator, the variance scaling is triggered only when the improvements are found to be far away from the mean vector. In [53], a cross-entropy based adaptive variance scaling method was proposed. In this method, the difference between the sampled population and the prediction of the probabilistic model is first measured, and the scaling factor on the variance is then computed by minimizing the cross-entropy between the two distributions.

Different from the above variance scaling methods that directly multiply a scaling factor on the estimated variance, in [49], the authors proposed a novel probability density estimator based on the new mean vector obtained by the anticipated mean shift strategy. Then, once the new mean vector gets better, the variance estimator adaptively enlarges the variance without using an explicit factor, but rather, by using the new better mean vector to calculate the variance. Furthermore, they also developed a reflecting sampling strategy to further improve the search efficiency of GEDA. Accompanied with these two schemes, a new GEDA variant named EDA_{VERS} [49] was developed. Subsequently, a novel anisotropic adaptive variance scaling (AAVS) method was proposed in [41], and a new GEDA named AAVS-EDA was designed. Specifically, in this algorithm, a topology-based detection method was devised to detect the landscape characteristics of the optimization problems, and then, based on the captured characteristics, the variances along different eigen-directions are anisotropically scaled. In this way, the variances and the main search direction of GEDA could be simultaneously adjusted. Recently, Liang et al. proposed a new GEDA variant, named EDA2 [37], to improve the optimization performance of EDA. Specifically, instead of only utilizing promising individuals in the current generation to estimate the Gaussian model, this algorithm stores historical high-quality individuals generated in the previous generations into an archive and adopts these individuals to collaboratively estimate the covariance of the Gaussian model. In this manner, valuable historical evolution information could be integrated into the estimated model.

The above mentioned MGEDAs usually adopt the full rank covariance matrix to estimate the covariance. Since the calculation of the full-rank covariance matrix is very time-consuming, the computational complexity of most MGEDAs is usually high. To alleviate this shortcoming, researchers turn to seeking efficient covariance matrix adaption (CMA) techniques for EDA [54]. As for the covariance matrix, a direct and simple method to speed up its computation is to reduce the degrees of freedom. To this end, Ros and Hansen [55] proposed to only update the elements in the diagonal of the covariance matrix, leading to a de-randomized evolution strategy, named sep-CMA-ES. This method reduces the updating time and space complexity of the covariance matrix from quadratic to linear. In [56], the authors devised an adaptive diagonal decoding scheme to accelerate covariance matrix adaptation. Further, ref. [57] developed a matrix-free CMA strategy by employing combinations of difference vectors between archived individuals and random vectors, generated by the univariate Gaussian distribution along directions of the past shifts of the mean vector. In [58], Beyer and Sendhoff proposed a matrix adaptation evolution strategy (MA-ES) by removing one evolution path in the calculation of the covariance matrix, leading to that the covariance update is no longer needed. In [59], Li and Zhang first designed a rank one evolution strategy by using a single principal search direction,

which is of linear complexity. Then, they developed a rank-m evolution strategy by employing multiple search directions. In particular, these two evolution strategies mainly adopt principal search directions to seek for the optimal low rank approximation to the covariance matrix. In [60], He et al. put forward a search direction adaptation evolution strategy (SDA-ES) with linear time and space complexity. Specifically, this algorithm first models the covariance matrix with an identity matrix along with multiple search directions. Then, it uses a heuristic to update the search directions such as the principal component analysis.

Besides the advance of GEDAs in covariance (or variance) scaling and adaption, some researchers have also attempted to design new ways to shift the mean vector of the Gaussian distribution model. For instance, in [61], Bosman et al. proposed an anticipated mean shift to update the mean vector and then used the updated mean vector to calculate the variance. In addition, some researchers have also attempted to adopt other distribution models, instead of the Gaussian distribution model, to estimate the probability distribution. For example, in [62], a probabilistic graphical model was designed to consider the dependencies between multivariate variables. Specifically, in this algorithm, a parallel of a certain number of subgraphs, with a smaller number of variables, is estimated separately to capture the dependencies among variables in each subgraph. Then, each estimated graph model associated with the subgraph samples new values for the associated variables separately. In [39], the authors utilized the Boltzmann distribution to build the probability distribution model in EDA, leading to BUMDA. In particular, the distribution parameters are derived from the analytical minimization of the Kullback–Leibler divergence. In [50], the authors devised a novel multiple sub-models maintenance technique for EDA, leading to a new EDA variant, named maintaining and processing sub-models (MAPS). Specifically, this algorithm maintains multiple sub-models to detect promising areas.

Since EDAs utilize the estimated probability distribution model to sample new solutions, they generally lack subtle refinement to improve the solution accuracy [63]. To fill this gap, local search methods are commonly accompanied with EDAs to refine the found promising solutions [38,49,50]. For instance, in [64], simulated annealing (SA) based local search operator was incorporated into EDA to balance the exploration and exploitation to search the solution space properly. Specifically, the SA-based local search is probabilistically executed on some good solutions to improve their accuracy. To improve the solution accuracy, Zhou et al. [38] developed cheap and expensive local search methods for EDA, leading to a new EDA variant named EDA/LS. In particular, this EDA variant adopts a modified univariate histogram probabilistic model to sample a part of individuals, and it then utilizes a cheap local search method to sample the rest of the individuals. Besides, it also employs an expensive local search method to refine the found promising solutions. Along this direction, an extension of EDA/LS, named EDA/LS-MS, was developed in [65] by introducing a mean shift strategy to replace the cheap local search method in EDA/LS to refine some good parent solutions.

Though a lot of remarkable GEDA variants have emerged and shown promising performance in solving optimization problems, they still encounter limitations, such as falling into local areas and premature convergence. In particular, it is found that most existing GEDAs estimate the variance (or covariance) based on the same selected promising individuals used for the estimation of the mean vector. Although various variance (or covariance) scaling methods [37,43,49,53,66] and covariance matrix adaption methods [56–60] have been proposed to improve the sampling range of the estimated probability distribution model, on the one hand, the structure of the optimization problem captured by most existing GEDA variants remains unchanged after the scaling; on the other hand, most existing variance (covariance) scaling methods scale the estimated variance (covariance) equally in different directions. This is actually not beneficial to effectively sample new promising individuals.

To alleviate the above concern, this paper devises an adaptive covariance scaling EDA by adaptively enlarging the number of promising individuals (as compared with those

for the mean vector estimation) to estimate the covariance. In this way, not only does the structure of the optimization problem captured by the algorithm become better, but also the covariance is scaled differently in different directions.

3. Proposed ACSEDA

To improve the effectiveness of EDA in solving optimization problems, this paper proposes an adaptive covariance scaling EDA (ACSEDA) by introducing more promising individuals to calculate the covariance. Furthermore, to alleviate the sensitivity of the proposed ACSEDA to parameters, this paper further devises two adaptive strategies for the two key parameters in ACSEDA. The components of ACSEDA are elucidated as follows.

3.1. Adaptive Covariance Scaling

In traditional GEDAs [27], both the mean vector and the covariance of the multivariate Gaussian distribution model are estimated based on the selected promising individuals. Then, on the basis of the estimated probability distribution model, the offspring are sampled randomly. In particular, we can see that the mean vector has a great influence on the convergence speed of GEDAs to promising areas, while the covariance mainly takes charge of the sampling range of the distribution model, which plays a significant role in high diversity maintenance.

During the evolution, the population gradually approaches the promising areas and the selected promising individuals used for probability distribution estimation are gradually aggregated together as well. In this situation, the estimated covariance would become smaller and smaller. Once the estimated mean vector falls into local areas, the sampled offspring could hardly escape from local areas. As a consequence, the population falls into local areas, and premature convergence occurs. Such a predicament is encountered by many existing GEDAs [20].

To alleviate this issue, this paper proposes a covariance scaling strategy to enlarge the covariance by introducing more promising individuals on the basis of the selected individuals for the estimation of the mean vector. Specifically, given the population size is PS, $s = \lceil sr * PS \rceil$ promising individuals are first selected from the population to estimate the mean vector of the probability distribution, where sr is the selection ratio, defined as the number of selected individuals out of the population. Then, different from most existing GEDAs [67], which estimate the covariance based on the $sc = \lceil cs * PS \rceil$ individuals as well, this paper selects sc promising individuals to estimate the covariance, where cs is the covariance scaling parameter, which is the number of the promising individuals out of the population and is usually larger than sr. In this way, more promising individuals are selected to participate in the estimation of the covariance and thus, the covariance is enlarged.

As shown in Figure 1, after the population is sorted from the best to the worst with respect to the fitness, s best individuals are selected to form the promising individual set S, and then, the mean vector u is estimated based on S according to Equation (3). Subsequently, different from most existing GEDAs, the proposed covariance scaling method selects sc best individuals to form the promising individual set S_C to estimate the covariance. It should be mentioned that cs is usually larger than sr, which also indicates that S_C is larger than S. In this way, S is a subset of S_C. Subsequently, instead of using S to calculate the covariance according to Equation (6), the proposed method utilizes S_C, namely an enlarged individual set, to estimate the covariance as follows:

$$C = \frac{1}{sc - 1}(S_C - \mu)(S_C - \mu)^T \tag{9}$$
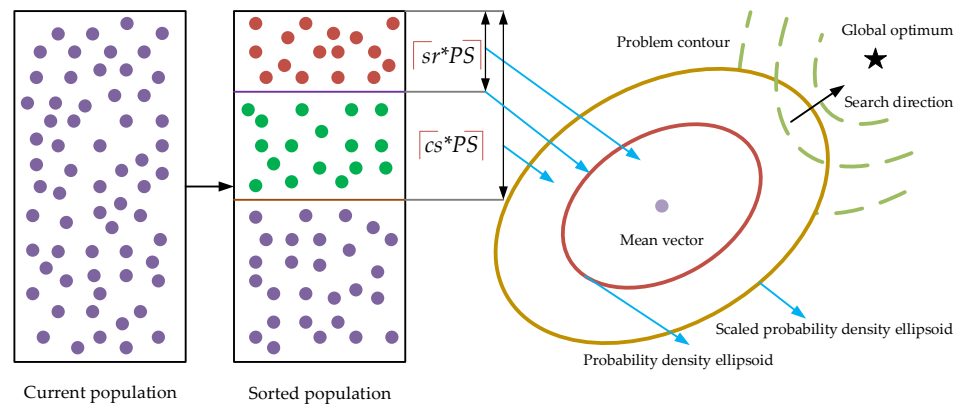
Figure 1. The visual structure of the covariance scaling strategy.

As shown in Figure 1, since more promising individuals participate in the estimation of the covariance, the sampling range of the estimated probability distribution model is enlarged. On the one hand, the sampled offspring based on this model are more diversified, which is very beneficial for the population to avoid falling into local areas. On the other hand, it might also be likely to generate more promising individuals close to the promising areas, and thus, the convergence could also be strengthened to some extent.

Remark 1. *It deserves attention that different from existing covariance scaling methods, which scales the covariance directly with a fixed scalar, the proposed covariance scaling method estimates the covariance based on an enlarged number of promising individuals as compared to those for the estimation of the mean vector. This brings the following two benefits for the estimated probability distribution model:*

(1) By introducing more promising individuals, the proposed scaling method enlarges the covariance differently in different directions between variables and thus it implicitly takes the difference between variables into consideration. However, existing scaling methods [37] enlarge the covariance with a same scalar and hence they do not consider the difference between variables.

(2) The proposed scaling method is likely to better capture the structure of the optimization problem with respect to the correlations between variables by introducing more promising individuals. Nevertheless, the structure captured by existing scaling methods remains unchanged after the scaling.

Taking a deep investigation on the parameter cs, in the proposed covariance scaling method, we find that neither a too large cs, nor a too small cs are proper to aid EDA to achieve promising performance. On the one side, a too-large cs may lead to a too large sampling range of the probability distribution. This may result in too diversified offspring sampled from the distribution model. In particular, it is found that, in the early stage of the evolution, a large cs may be beneficial to maintain a large sampling range and thus sample diversified offspring. This is helpful for EDA to explore the solution space in very different directions, whereas, in the late stage, such a setting of cs is not appropriate because it is not beneficial for the population to extensively exploit the found promising areas to refine the solution accuracy. On the other side, a too-small cs may bring in a too-small sampling range of the distribution model, which may sample concentrated offspring. Though it is desirable in the late stage of the evolution, it is not suitable during the whole evolution, because it may increase the risk of EDA in falling into local areas. Consequently, based on the above analysis, it is found that cs should not be fixed, but dynamically adjusted during the evolution process.

To the above end, this paper further designs an adaptive strategy for cs as follows:

$$cs = 1 - (1 - sr_{\min})\left(\frac{FEs}{FEs_{\max}}\right)^2 \tag{10}$$

where sr_{min} denotes the lower bound of the selection ratio sr used for the estimation of the mean vector of the multivariate Gaussian distribution model, FEs_{max} is the maximum number of fitness evaluations, while FEs denotes the used number of fitness evaluations up to the current generation.

From Equation (10), we can see that cs decreases from 1.0 to sr_{min} as the evolution continues. Specifically, it is found that, in the early stage, most individuals in the population are used to estimate the covariance. This brings two benefits for EDA: (1) the sampling range of the probability distribution model is large and thus the sampled offspring are diversified and scatter dispersedly to explore the solution space. It is not only beneficial for the population to find more promising areas, but it is also very profitable for the population to avoid falling into local areas. (2) The captured structure of the optimization problem tends to be global and accurate with a large number of promising individuals. In the early stage, the individuals are usually scattered diversely in the solution space. In this situation, the captured structure of the optimization problem is usually global. Therefore, to accurately capture the correlations between variables globally, a large number of promising individuals are usually needed. Consequently, in the early stage, it is helpful to capture an accurate structure of the optimization problem when cs is large.

Conversely, in the late stage, from Equation (10), it is found that cs becomes smaller and smaller. This leads to a narrow sampling range of the probability distribution. Therefore, the sampled offspring are concentrated and surrounded around the mean vector. In this situation, the population exploits the found promising areas, and thus, the accuracy of the solution can be improved.

To summarize, with the above adaptive covariance scaling scheme, the proposed EDA variant is expected to obtain a promising balance between diversification and intensification of the population. Therefore, the algorithm could explore and exploit the complicated solution space properly to obtain promising performance in solving complicated optimization problems.

3.2. Adaptive Promising Individuals Selection

In GEDAs, the number ($s = \lceil sr * PS \rceil$) of selected promising individuals, for the estimation of the mean vector, makes a significant influence on the convergence speed of EDAs. As shown in Figure 1, the mean vector mainly takes control of the center of the sampled offspring. A too-large sr may lead to that a large number of promising individuals being used to estimate the mean vector. As a result, the estimated mean vector may be too far away from the promising areas. In the early stage of the evolution, this is beneficial for EDAs to maintain high search diversity. Nevertheless, in the late stage of the evolution, a large sr may slow down the convergence of the population to find high-quality solutions. On the contrary, a too-small sr may result in the estimated mean vector being too close to the promising areas. This case is suitable, in the late stage of the evolution, to exploit the found promising areas. However, it may lead to premature convergence if we keep sr small during the whole evolution, especially when the selected promising individuals all fall into local areas.

Based on the above analysis, it might as well dynamically adjust sr during the evolution. To this end, this paper devises a simple adaptive strategy for sr as follows:

$$sr = sr_{max} - (sr_{max} - sr_{min}) \left(\frac{FEs}{FEs_{max}} \right)^{0.1} \tag{11}$$

where sr_{max} and sr_{min} represent the maximum selection ratio and the minimum selection ratio, which accordingly determine the maximum number ($s_{max} = \lceil sr_{max} * PS \rceil$) and the minimum number ($s_{min} = \lceil sr_{min} * PS \rceil$) of the selected promising individuals. In this paper, we set them as 0.35 and 0.05, respectively.

From Equation (11), we can see that in the early stage, sr is large, and then, it decreases gradually as the evolution goes. This indicates that during the evolution, the mean vector

of the estimated probability distribution is becoming closer and closer to the promising areas. In this way, the population gradually tends to exploit the found promising areas.

Remark 2. *In particular, compared Equation (11) with Equation (10), as shown in Figure* 2, *the following findings can be obtained:*

(1)　sr decreases dramatically in the early stage, and mildly in the late stage, while cs decreases mildly in the early stage, and dramatically in the late stage. This actually matches the expectation that the proposed ACSEDA should explore the solution space in the early stage without serious loss of convergence, while it should exploit the search space in the late stage without serious sacrifice of search diversity. For one thing, in the early stage, sr decreases rapidly and thus the estimated mean vector is close to the promising areas that the current population lies. However, it should be mentioned that in such a situation, the sampling diversity of the estimated probability distribution is not declined, because the estimated covariance is large due to the large cs. On the contrary, in this situation, the sampling quality of the estimated probability distribution could be improved due to the high-quality mean vector and thus the population could effectively explore the search space to find promising areas in the early stage. In the late stage, sr decreases mildly, while cs descends quickly. In this situation, the quality of the mean vector is gradually promoted by approaching the promising areas closer and closer. At the same time, the sampling range of the estimated distribution gradually shrinks due to the covariance estimated on the reduced number of promising individuals. Therefore, in the late stage, ACSEDA gradually biases to exploiting the found promising areas to improve the solution quality. However, it should be mentioned that such a bias is not at the serious sacrifice of the search diversity because of the proposed covariance scaling technique.

(2)　cs is always larger than sr during the evolution and the gap between cs and sr gradually shrinks as the evolution goes. This indicates that during the evolution, compared with traditional GEDAs, the covariance is always amplified, so that the estimated probability distribution could sample diversified offspring around the estimated mean vector with high quality. In addition, the gradually narrowed difference between sr and cs indicates that the scaling of the covariance is gradually declined. This implies that the proposed ACSEDA gradually concentrates on exploiting the solution space to refine the solution accuracy.
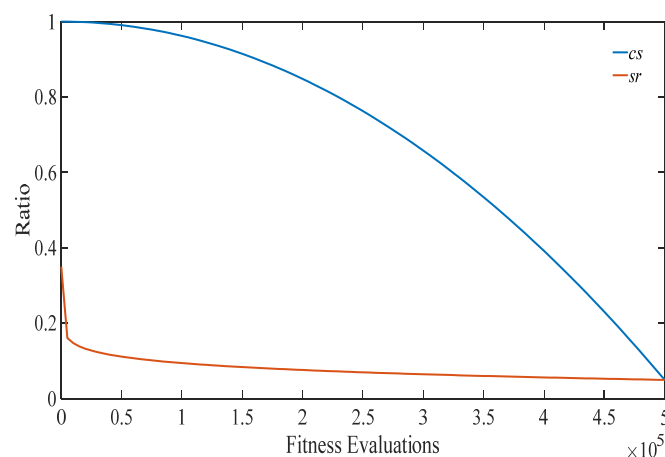
Figure 2. The change curves of *cs* and *sr* with the proposed two adaptive strategies.

3.3. Cross-Generation Individual Selection for Parent Population

In traditional EDAs [28], the offspring is directly utilized as the parent population for the next generation to estimate the probability distribution model. Since the quality of the sampled offspring is uncertain, the quality of the estimated probability distribution model may not be improved or even degrade compared with the estimated probability distribution in the last generation. This may slow down the convergence of the population

to promising areas. Therefore, to remedy this shortcoming, some EDA variants [68] combine the offspring and the parent population together and then select the best *PS* individuals as the parent population for the next generation. However, such selection is too greedy and thus may lead to premature convergence and falling into local areas.

To alleviate the above predicament, and to further make a promising compromise between convergence and diversity, this paper further devises a cross-generation individual selection strategy for the parent population.

As Figure 3 shows, this paper combines the sampled offspring in the last generation and the sampled offspring in the current generation and then selects the best *PS* individuals as the parent population for the next generation to estimate the probability distribution model. In this way, the historical information in the last generation can be utilized to build the probability distribution model.
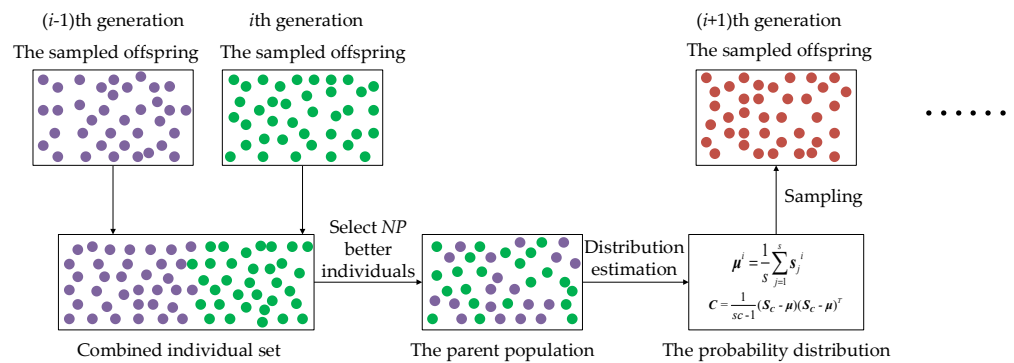
Figure 3. The process of the cross-generation individual selection strategy.

Remark 3. *Different from existing individual selection methods [26,48,69], the proposed cross-generation individual selection strategy takes advantage of the sampled offspring in the two consecutive generations to select individuals for the parent population in the next generation. Such a selection strategy brings the following benefits to ACSEDA:*

(1) Individuals in the parent population are diversified. The sampled offspring in the last generation usually have big difference with the offspring in the current generation. Combining them together to select individuals is less likely to generate crowded individuals for the parent population. As a result, the estimated probability distribution model is less likely to fall into local areas and at the same time, has a wide sampling range to generate diversified offspring. In this way, ACSEDA could preserve high search diversity during the evolution.

(2) With this strategy, the latest historical promising individuals in the last generation could be integrated with those in the current offspring. As a consequence, individuals in the parent population are not only diversified, but also of high quality. Hence, the estimated probability distribution is of high quality to generate more promising offspring. By this means, the convergence of ACSEDA could be guaranteed.

(3) With this selection strategy, ACSEDA is further expected to preserve a good compromise between exploration and exploitation to search the solution space effectively. Experiments conducted in Section 4 will demonstrate the effectiveness of the proposed cross-generation individual selection strategy.

3.4. Overall Procedure of ACSEDA

Combining the above three schemes together, the proposed ACSEDA is outlined in Algorithm 2. Specifically, after the initialization of the population (Line 2), the algorithm goes to the main iteration loop for evolution (Lines 5–16). In the main loop, it first executes the proposed adaptive promising individual selection strategy for the estimation of the mean vector (Lines 6 and 7). Then, it comes to the proposed adaptive covariance scaling strategy to estimate the covariance of the probability distribution model (Lines 8 and 9).

Subsequently, the offspring are randomly sampled based on the estimated probability distribution model (Line 10). Hereafter, it arrives at the proposed cross-generation individual selection strategy to select individuals for the parent population in the next generation (Line 12). At last, a local search method is conducted on the global best solution to refine its accuracy (Line 14).

Algorithm 2: The Procedure of ACSEDA.

Input: population size *PS*;
1: Set *FEs* = 0;
2: Initialize *PS* individuals randomly and evaluate their fitness;
3: *FEs* = *FEs* + *PS*;
4: Obtain the global best solution ***Gbest*** and store the current population;
5: **While** (*FEs* < *FEs*$_{\max}$)
6:      Calculate the selection ratio *sr* according to Equation (11);
7:      Select $\lceil sr * PS \rceil$ promising solutions from the population and calculate the mean value ***μ*** using Equation (3);
8:      Calculate the covariance scaling parameter *cs* according to Equation (10);
9:      Estimate the covariance matrix ***C*** according to Equation (9);
10:      Randomly sample *PS* new individuals based on the estimated multivariate Gaussian model, evaluate their fitness and store them;
11:      *FEs* = *FEs* + *PS*;
12:      Combine the offspring in the last generation and the offspring in the current generation to select *PS* better individuals to form the parent population for the next generation;
13:      Update the global best solution ***Gbest***;
14:      Execute local search *2* times on ***Gbest***;
15:      *FEs* = *FEs* +2;
16:      **End While**
Output: the global best solution ***Gbest***;

In Algorithm 2, it should be noticed that a local search strategy is additionally added to improve the solution accuracy of the global best solution. This is because EDAs are probability distribution model based optimization algorithms, and as a consequence, EDAs usually lack strong local exploitation [26,38,70]. Therefore, in the literature [23,38,70], local search methods are generally accompanied by EDAs to improve the solution quality. Hence, the same as most existing EDA variants [23,26,38,70,71], ACSEDA also adopts a local search method to refine the global best solution, as shown in Line 14.

For simplicity and keeping consistent with the probability distribution model in ACSEDA, this paper applies the univariate Gaussian distribution with a small variance to execute the local search on the global best solution. In this paper, the small variance is set as 1.0×10^{-4}. In addition, for saving computational resources, we execute the local search method on the global best solution only two times, as shown in Line 14.

As a whole, with the proposed three main techniques and the local search method, ACSEDA is expected to explore and exploit the solution space properly to locate the optima of optimization problems.

4. Experimental Studies

This section mainly conducts extensive experiments to verify the effectiveness of the proposed ACSEDA. Specifically, the commonly used CEC 2014 benchmark problem set [44] is adopted in this paper. This benchmark set contains 30 various complicated optimization problems, such as unimodal problems, multimodal problems, hybrid problems, and composition problems. For detailed information on this benchmark set, please refer to [44].

4.1. Experimental Settings

First, to comprehensively demonstrate the effectiveness of the proposed ACSEDA, we select several state-of-the-art EDA variants to make comparisons in solving the complicated CEC 2014 benchmark problems. Specifically, the selected state-of-the-art EDA variants are

EDA2 [37], EDA$_{\text{VERS}}$ [49], EDA/LS [38], EDA/LS-MS [65], MA-ES [58], and BUMDA [39]. In addition, as a baseline method, the traditional multivariate Gaussian model based EDA [28] is also utilized as a compared method. To tell it apart from the others, we denote it as TRA-EDA in the experiments.

Second, to make comprehensive comparisons between the proposed ACSEDA and the above compared EDA variants, we compare their optimization performance in solving the CEC 2014 problems with three different dimension sizes, namely 30-*D*, 50-*D*, and 100-*D*. For fairness, the maximum number of fitness evaluations ($FEs_{\max}$) is set as $10,000 * D$ for all algorithms.

Third, for fair comparisons, the key parameter settings of the compared algorithms are set, as recommended, in the associated papers. For the population size, we tune the settings of all algorithms on the CEC 2014 benchmark set with different dimension sizes. Specifically, after preliminary experiments, the parameter settings of all algorithms are shown in Table 1.

Table 1. Parameter settings of ACSEDA and the compared algorithms.

Algorithm	ACSEDA	EDA/LS					EDA2		EDA/LS-MS					EDA$_{\text{VERS}}$		BUMDA	TRA-EDA		MA-ES	
Parameter	*PS*	*PS*	*M*	*P_b*	*P_c*	*θ*	*PS*	*l*	*PS*	*M*	*P_a*	*P_b*	*θ*	*PS*	*sr*	*PS*	*PS*	*sr*	*PS*	*mu*
D 30	1300	150					100		150					500		900	2500		$4 + \lfloor 3 * \ln D \rfloor$	$\lfloor PS/2 \rfloor$
50	1800	150	15	0.2	0.2	0.1	200	20	1000	15	0.2	0.2	0.1	600	0.35	1100	4700	0.2		
100	3200	150					300		2000					700		1500	6000			

Fourth, to comprehensively evaluate the optimization performance of each algorithm, we execute each algorithm independently for 30 runs and utilize the median, mean, and standard deviation values over the 30 independent runs to evaluate its optimization performance. Furthermore, to tell the statistical significance, we conduct the Wilcoxon rank-sum test, at the significance level of $\alpha = 0.05$, to compare the proposed ACSEDA with each associated EDA variant. In addition, to compare the overall optimization performance of all algorithms on the whole CEC 2014 benchmark set, we further conduct the Friedman test at the significance level of $\alpha = 0.05$ by taking advantage of the mean value of each algorithm on each function in the benchmark set.

At last, it deserves attention that all algorithms are programmed under MATLAB R2018a, and they are run on the same computer with Intel(R) Core(TM) i7-10700T CPU @ 2.90 GHz 2.90 GHz and 8 G RAM.

4.2. Comparison with State-of-the-Art EDAs

Table 2, Table 3, and Table 4 display the comparison results between ACSEDA and the compared EDA variants on the 30-*D*, 50-*D*, and 100-*D* CEC 2014 benchmark problems, respectively. In these tables, the symbols "+", "−" and "=" represent that ACSEDA is significantly better than, significantly worse than, and equivalent to the associated compared algorithms on the associated problems, respectively. Besides, "*w/t/l*" denotes the numbers of the problems where ACSEDA achieves significantly better performance, equivalent performance, and significantly worse performance than the compare algorithms, respectively. Actually, "*w/t/l*" is equal to the numbers of "+", "=" and "−", respectively. Additionally, in the last rows of this table, the averaged rank of each algorithm obtained from the Friedman test is presented.

Table 2. Comparison between ACSEDA and the compared state-of-the-art EDA variants on the 30-*D* CEC2014 benchmark problems. The bold results indicate that ACSEDA is significantly better than the compared methods.

F	Category	Quality	ACSEDA	EDA2	EDA$_{VERS}$	EDA/LS	EDA/LS-MS	TRA-EDA	BUMDA	MA-ES
F_1		Median	0.00×10^0	0.00×10^0	6.48×10^4	1.90×10^{-9}	1.83×10^{-9}	6.94×10^7	1.15×10^8	1.42×10^{-14}
		Mean	0.00×10^0	0.00×10^0	7.76×10^4	9.09×10^1	1.98×10^1	6.80×10^7	1.16×10^8	1.37×10^{-14}
		Std	0.00×10^0	0.00×10^0	4.85×10^4	4.89×10^2	1.07×10^2	1.17×10^7	2.12×10^7	2.55×10^{-15}
		p-value	-	NaN$^=$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.17 \times 10^{-13+}}$
F_2	Unimodal Problems	Median	0.00×10^0	0.00×10^0	2.54×10^3	1.48×10^{-11}	4.59×10^{-11}	6.61×10^9	1.80×10^4	2.84×10^{-14}
		Mean	0.00×10^0	0.00×10^0	2.36×10^3	2.52×10^{-10}	1.45×10^{-10}	6.58×10^9	5.15×10^5	2.65×10^{-14}
		Std	0.00×10^0	0.00×10^0	8.94×10^2	5.04×10^{-10}	2.43×10^{-10}	1.05×10^9	2.45×10^6	7.09×10^{-15}
		p-value	-	NaN$^=$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{7.15 \times 10^{-13+}}$
F_3		Median	0.00×10^0	0.00×10^0	1.41×10^3	8.42×10^{-9}	8.85×10^{-10}	9.83×10^3	6.49×10^3	5.68×10^{-14}
		Mean	0.00×10^0	0.00×10^0	1.44×10^3	3.10×10^{-9}	2.34×10^{-9}	1.00×10^4	6.52×10^3	5.68×10^{-14}
		Std	0.00×10^0	0.00×10^0	5.54×10^2	4.33×10^{-9}	4.21×10^{-9}	1.97×10^3	1.85×10^3	5.05×10^{-29}
		p-value	-	NaN$^=$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.69 \times 10^{-14+}}$
F_{1-3}	*w/t/l*		-	0/3/0	3/0/0	3/0/0	3/0/0	3/0/0	3/0/0	3/0/0
F_4		Median	3.25×10^0	0.00×10^0	6.39×10^{-1}	1.02×10^{-9}	4.75×10^{-9}	7.93×10^2	9.74×10^1	5.68×10^{-14}
		Mean	3.15×10^0	0.00×10^0	6.14×10^{-1}	1.33×10^{-1}	2.66×10^{-1}	7.72×10^2	9.88×10^1	2.66×10^{-1}
		Std	1.02×10^0	0.00×10^0	2.76×10^{-1}	7.16×10^{-1}	9.94×10^{-1}	1.27×10^2	2.69×10^1	9.94×10^{-1}
		p-value	-	$1.21 \times 10^{-12-}$	$3.82 \times 10^{-10-}$	$2.15 \times 10^{-10-}$	$1.41 \times 10^{-9-}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$1.69 \times 10^{-10-}$
F_5		Median	2.09×10^1	2.10×10^1	2.10×10^1	2.00×10^1	2.00×10^1	2.09×10^1	2.10×10^1	2.00×10^1
		Mean	2.09×10^1	2.09×10^1	2.09×10^1	2.00×10^1	2.00×10^1	2.09×10^1	2.10×10^1	2.04×10^1
		Std	4.96×10^{-2}	4.86×10^{-2}	7.13×10^{-2}	5.47×10^{-6}	3.43×10^{-2}	5.47×10^{-2}	5.00×10^{-2}	6.93×10^{-1}
		p-value	-	$7.01 \times 10^{-2=}$	$7.48 \times 10^{-2=}$	$1.17 \times 10^{-11-}$	$9.37 \times 10^{-12-}$	$8.77 \times 10^{-1=}$	$\mathbf{2.32 \times 10^{-2+}}$	$3.96 \times 10^{-4-}$
F_6	Simple Multimodal Problems	Median	1.99×10^{-6}	0.00×10^0	1.77×10^{-5}	4.09×10^1	3.81×10^1	4.58×10^{-1}	1.50×10^0	2.25×10^1
		Mean	2.07×10^{-6}	2.09×10^{-1}	4.76×10^{-1}	4.05×10^1	3.84×10^1	5.16×10^{-1}	1.61×10^0	2.24×10^1
		Std	8.29×10^{-7}	5.63×10^{-1}	8.19×10^{-1}	2.14×10^0	2.75×10^0	4.43×10^{-1}	1.12×10^0	4.08×10^0
		p-value	-	$\mathbf{3.82 \times 10^{-5+}}$	$6.61 \times 10^{-1=}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{1.07 \times 10^{-7+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_7		Median	0.00×10^0	0.00×10^0	1.14×10^{-13}	2.21×10^{-2}	3.69×10^{-2}	9.12×10^1	3.24×10^{-2}	1.14×10^{-13}
		Mean	0.00×10^0	0.00×10^0	1.06×10^{-13}	3.52×10^{-2}	4.84×10^{-2}	9.01×10^1	9.35×10^{-2}	2.79×10^{-3}
		Std	0.00×10^0	0.00×10^0	2.84×10^{-14}	4.56×10^{-2}	6.23×10^{-2}	1.25×10^1	1.41×10^{-1}	4.40×10^{-3}
		p-value	-	NaN$^=$	$\mathbf{7.15 \times 10^{-13+}}$	$\mathbf{1.20 \times 10^{-12+}}$	$\mathbf{1.19 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{3.85 \times 10^{-13+}}$
F_8		Median	9.95×10^{-1}	6.47×10^0	1.52×10^2	2.27×10^2	2.06×10^2	1.49×10^1	0.00×10^0	1.68×10^2
		Mean	1.16×10^0	6.60×10^0	1.32×10^2	2.38×10^2	2.13×10^2	1.54×10^1	0.00×10^0	1.68×10^2
		Std	8.93×10^{-1}	1.81×10^0	5.09×10^1	6.69×10^1	6.08×10^1	3.39×10^0	0.00×10^0	3.98×10^0
		p-value	-	$\mathbf{1.92 \times 10^{-11+}}$	$\mathbf{3.86 \times 10^{-11+}}$	$\mathbf{2.13 \times 10^{-11+}}$	$\mathbf{2.13 \times 10^{-11+}}$	$\mathbf{2.13 \times 10^{-11+}}$	$4.26 \times 10^{-9-}$	$\mathbf{2.02 \times 10^{-11+}}$

Table 2. *Cont.*

F	Category	Quality	ACSEDA	EDA2	EDA$_{VERS}$	EDA/LS	EDA/LS-MS	TRA-EDA	BUMDA	MA-ES
F_9		Median	9.95×10^{-1}	5.47×10^0	1.48×10^2	3.33×10^2	3.20×10^2	1.10×10^1	9.95×10^{-1}	1.87×10^2
		Mean	1.16×10^0	5.80×10^0	1.31×10^2	3.28×10^2	3.18×10^2	1.13×10^1	1.41×10^0	1.88×10^2
		Std	9.29×10^{-1}	1.69×10^0	4.96×10^1	6.66×10^1	4.01×10^1	2.68×10^0	1.05×10^0	5.42×10^0
		p-value	-	$\mathbf{2.85 \times 10^{-11+}}$	$\mathbf{1.13 \times 10^{-11+}}$	$\mathbf{2.40 \times 10^{-11+}}$	$\mathbf{2.40 \times 10^{-11+}}$	$\mathbf{2.40 \times 10^{-11+}}$	$\mathbf{2.03 \times 10^{-2+}}$	$\mathbf{2.10 \times 10^{-11+}}$
F_{10}		Median	3.59×10^0	6.33×10^0	5.69×10^3	4.76×10^3	4.04×10^3	2.39×10^2	2.84×10^1	3.88×10^3
		Mean	1.13×10^1	5.28×10^1	5.62×10^3	4.76×10^3	4.11×10^3	2.86×10^2	3.81×10^1	3.90×10^3
		Std	2.89×10^1	8.32×10^1	3.90×10^2	1.24×10^3	8.86×10^2	1.98×10^2	5.51×10^1	3.24×10^2
		p-value	-	$\mathbf{2.10 \times 10^{-3+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{1.09 \times 10^{-10+}}$	$\mathbf{8.77 \times 10^{-2=}}$	$\mathbf{3.01 \times 10^{-11+}}$
F_{11}		Median	8.36×10^0	1.52×10^1	6.55×10^3	4.64×10^3	4.69×10^3	2.63×10^2	2.26×10^2	4.31×10^3
		Mean	2.80×10^1	7.94×10^1	6.53×10^3	5.04×10^3	4.99×10^3	2.58×10^2	2.43×10^2	4.31×10^3
		Std	4.53×10^1	1.19×10^2	2.54×10^2	1.36×10^3	1.07×10^3	1.53×10^2	2.10×10^2	2.21×10^2
		p-value	-	$\mathbf{2.87 \times 10^{-2+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{2.44 \times 10^{-9+}}$	$\mathbf{8.84 \times 10^{-7+}}$	$\mathbf{2.98 \times 10^{-11+}}$
F_{12}		Median	2.35×10^0	2.46×10^0	2.56×10^0	6.89×10^{-1}	6.67×10^{-1}	2.47×10^0	2.51×10^0	3.34×10^{-2}
		Mean	2.32×10^0	2.49×10^0	2.56×10^0	9.21×10^{-1}	7.84×10^{-1}	2.39×10^0	2.47×10^0	3.68×10^{-2}
		Std	2.41×10^{-1}	2.71×10^{-1}	1.98×10^{-1}	8.73×10^{-1}	4.11×10^{-1}	3.49×10^{-1}	2.56×10^{-1}	2.29×10^{-2}
		p-value	-	$\mathbf{1.76 \times 10^{-2+}}$	$\mathbf{1.89 \times 10^{-4+}}$	$\mathbf{5.57 \times 10^{-10-}}$	$\mathbf{4.50 \times 10^{-11-}}$	$\mathbf{1.45 \times 10^{-1=}}$	$\mathbf{1.70 \times 10^{-2+}}$	$\mathbf{3.02 \times 10^{-11-}}$
F_{13}		Median	9.34×10^{-2}	4.40×10^{-2}	1.85×10^{-1}	5.86×10^{-1}	7.04×10^{-1}	3.24×10^0	4.27×10^{-2}	3.16×10^{-1}
		Mean	9.35×10^{-2}	4.39×10^{-2}	1.84×10^{-1}	1.85×10^0	2.57×10^0	3.20×10^0	4.25×10^{-2}	3.40×10^{-1}
		Std	1.50×10^{-2}	1.08×10^{-2}	2.68×10^{-2}	2.53×10^0	2.87×10^0	2.03×10^{-1}	8.97×10^{-3}	9.18×10^{-3}
		p-value	-	$\mathbf{6.07 \times 10^{-11-}}$	$\mathbf{3.34 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.69 \times 10^{-11-}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{14}		Median	2.59×10^{-1}	4.10×10^{-1}	3.10×10^{-1}	3.18×10^{-1}	3.27×10^{-1}	4.63×10^1	3.89×10^{-1}	3.39×10^{-1}
		Mean	2.62×10^{-1}	4.06×10^{-1}	3.11×10^{-1}	3.36×10^{-1}	3.42×10^{-1}	4.68×10^1	3.89×10^{-1}	3.81×10^{-1}
		Std	5.24×10^{-2}	3.87×10^{-2}	3.61×10^{-2}	9.64×10^{-2}	9.18×10^{-2}	5.15×10^0	3.91×10^{-2}	1.37×10^{-1}
		p-value	-	$\mathbf{1.78 \times 10^{-10+}}$	$\mathbf{3.01 \times 10^{-4+}}$	$\mathbf{7.66 \times 10^{-5+}}$	$\mathbf{9.51 \times 10^{-6+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{5.07 \times 10^{-11+}}$	$\mathbf{1.61 \times 10^{-6+}}$
F_{15}		Median	4.16×10^0	4.24×10^0	1.30×10^1	9.26×10^1	8.31×10^1	3.60×10^0	3.46×10^0	3.24×10^0
		Mean	4.21×10^0	4.26×10^0	1.30×10^1	9.75×10^1	8.43×10^1	4.69×10^0	4.68×10^0	3.40×10^0
		Std	1.43×10^0	1.08×10^0	1.04×10^0	3.23×10^1	2.67×10^1	3.53×10^0	5.52×10^0	9.00×10^{-1}
		p-value	-	$\mathbf{7.39 \times 10^{-1=}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{9.82 \times 10^{-1=}}$	$\mathbf{3.11 \times 10^{-1=}}$	$\mathbf{3.51 \times 10^{-2-}}$
F_{16}		Median	8.81×10^0	1.24×10^1	1.11×10^1	1.36×10^1	1.34×10^1	1.06×10^1	1.09×10^1	1.37×10^1
		Mean	8.74×10^0	1.23×10^1	1.11×10^1	1.35×10^1	1.34×10^1	1.05×10^1	1.09×10^1	1.37×10^1
		Std	5.75×10^{-1}	1.92×10^{-1}	3.95×10^{-1}	3.73×10^{-1}	2.65×10^{-1}	3.39×10^{-1}	4.89×10^{-1}	9.40×10^{-2}
		p-value	-	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.34 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{4.08 \times 10^{-11+}}$	$\mathbf{3.34 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$

Table 2. *Cont.*

F	Category	Quality	ACSEDA	EDA2	EDA$_{\text{VERS}}$	EDA/LS	EDA/LS-MS	TRA-EDA	BUMDA	MA-ES
F_{4-16}	*w/t/l*		-	8/3/2	10/2/1	10/0/3	10/0/3	10/3/0	9/2/2	9/0/4
F_{17}		Median	1.45×10^1	1.16×10^1	3.19×10^4	2.13×10^3	6.76×10^4	9.97×10^4	3.05×10^7	1.76×10^3
		Mean	2.17×10^1	2.00×10^1	3.58×10^4	4.10×10^7	2.38×10^7	1.36×10^5	2.98×10^7	1.77×10^3
		Std	2.98×10^1	2.80×10^1	1.97×10^4	7.15×10^7	5.06×10^7	1.05×10^5	7.53×10^6	4.44×10^2
		p-value	-	$4.16 \times 10^{-1=}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{18}		Median	4.71×10^{-1}	1.33×10^0	6.85×10^1	1.61×10^2	1.56×10^2	2.28×10^1	1.38×10^2	8.24×10^1
		Mean	8.77×10^{-1}	1.14×10^0	1.13×10^2	9.57×10^8	5.05×10^8	2.57×10^1	1.79×10^2	8.36×10^1
		Std	6.39×10^{-1}	7.04×10^{-1}	9.31×10^1	1.82×10^9	1.32×10^9	9.48×10^0	1.36×10^2	3.01×10^1
		p-value	-	$6.79 \times 10^{-2=}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{19}	Hybrid Problems	Median	2.78×10^0	3.66×10^0	4.86×10^0	1.56×10^1	1.44×10^1	4.28×10^1	2.10×10^1	9.97×10^0
		Mean	2.78×10^0	3.72×10^0	4.89×10^0	6.83×10^1	5.28×10^1	4.36×10^1	2.42×10^1	9.86×10^0
		Std	5.02×10^{-1}	4.02×10^{-1}	4.92×10^{-1}	1.85×10^2	1.33×10^2	7.25×10^{-1}	1.00×10^1	1.52×10^0
		p-value	-	$\mathbf{7.12 \times 10^{-9+}}$	$\mathbf{3.34 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{2.87 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{20}		Median	1.25×10^0	1.57×10^0	3.95×10^0	1.52×10^4	1.62×10^4	7.85×10^2	3.29×10^4	2.31×10^2
		Mean	1.33×10^0	1.76×10^0	4.14×10^0	2.10×10^4	2.61×10^4	1.03×10^3	3.18×10^4	2.72×10^2
		Std	1.98×10^{-1}	5.96×10^{-1}	1.26×10^0	1.89×10^4	2.55×10^4	6.34×10^2	5.47×10^3	1.43×10^2
		p-value	-	$\mathbf{7.70 \times 10^{-4+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{21}		Median	1.92×10^0	1.85×10^0	2.65×10^4	2.23×10^4	3.49×10^4	2.29×10^2	3.40×10^6	8.32×10^2
		Mean	1.45×10^1	3.35×10^1	2.93×10^4	5.96×10^6	1.05×10^7	2.86×10^2	3.38×10^6	8.98×10^2
		Std	3.40×10^1	5.26×10^1	9.29×10^3	2.53×10^7	2.38×10^7	1.94×10^2	1.58×10^6	3.58×10^2
		p-value	-	$8.88 \times 10^{-1=}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{22}		Median	2.67×10^1	2.81×10^1	2.46×10^2	1.15×10^3	1.24×10^3	3.67×10^1	1.53×10^2	4.15×10^2
		Mean	3.46×10^1	3.63×10^1	2.54×10^2	1.28×10^3	1.41×10^3	8.88×10^1	1.57×10^2	4.95×10^2
		Std	2.97×10^1	3.17×10^1	4.84×10^1	6.04×10^2	7.89×10^2	6.13×10^1	1.16×10^2	1.63×10^2
		p-value	-	$\mathbf{2.77 \times 10^{-5+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{6.72 \times 10^{-10+}}$	$\mathbf{9.76 \times 10^{-10+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{17-22}	*w/t/l*		-	3/3/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0
F_{23}	Composition Problems	*p*-value	3.15×10^2	3.15×10^2	3.15×10^2	3.20×10^2	3.20×10^2	3.37×10^2	3.19×10^2	2.00×10^2
		Mean	3.15×10^2	3.15×10^2	3.15×10^2	4.20×10^2	4.87×10^2	3.38×10^2	3.20×10^2	2.00×10^2
		Std	5.68×10^{-14}	5.68×10^{-14}	5.68×10^{-14}	2.59×10^2	2.96×10^2	3.78×10^0	1.92×10^0	0.00×10^0
		p-value	-	NaN$^=$	NaN$^=$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.69 \times 10^{-14-}}$

Table 2. *Cont.*

F	Category	Quality	ACSEDA	EDA2	EDA$_{VERS}$	EDA/LS	EDA/LS-MS	TRA-EDA	BUMDA	MA-ES
F_{24}		Median	2.00×10^2	2.00×10^2	2.23×10^2	2.63×10^2	2.56×10^2	2.27×10^2	2.30×10^2	2.00×10^2
		Mean	2.04×10^2	2.00×10^2	2.23×10^2	2.80×10^2	2.65×10^2	2.27×10^2	2.30×10^2	2.00×10^2
		Std	8.66×10^0	0.00×10^0	7.69×10^{-1}	5.55×10^1	3.19×10^1	1.50×10^0	2.92×10^0	1.31×10^{-4}
		p-value	-	$1.10 \times 10^{-2-}$	$\mathbf{9.93 \times 10^{-12+}}$	$6.48 \times 10^{-12+}$	$6.48 \times 10^{-12+}$	$6.48 \times 10^{-12+}$	$6.48 \times 10^{-12+}$	$2.03 \times 10^{-1=}$
F_{25}		Median	2.03×10^2	2.03×10^2	2.00×10^2	3.06×10^2	2.82×10^2	2.07×10^2	2.06×10^2	2.00×10^2
		Mean	2.03×10^2	2.03×10^2	2.00×10^2	2.97×10^2	2.78×10^2	2.07×10^2	2.06×10^2	2.00×10^2
		Std	1.33×10^2	2.63×10^{-2}	0.00×10^0	4.49×10^2	2.76×10^1	3.90×10^{-1}	1.24×10^0	0.00×10^0
		p-value	-	$8.27 \times 10^{-1=}$	$5.91 \times 10^{-13-}$	$1.74 \times 10^{-11+}$	$1.74 \times 10^{-11+}$	$1.74 \times 10^{-11+}$	$1.74 \times 10^{-11+}$	$5.91 \times 10^{-13-}$
F_{26}		Median	1.00×10^2	1.00×10^2	1.00×10^2	1.01×10^2	1.00×10^2	1.00×10^2	1.00×10^2	2.00×10^2
		Mean	1.00×10^2	1.00×10^2	1.00×10^2	1.05×10^2	1.04×10^2	1.01×10^2	1.01×10^2	2.00×10^2
		Std	1.84×10^{-2}	1.25×10^{-2}	1.06×10^2	2.38×10^2	1.82×10^2	1.05×10^0	9.10×10^{-1}	0.00×10^0
		p-value	-	$2.37 \times 10^{-1-}$	$\mathbf{3.34 \times 10^{-11+}}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$
F_{27}		Median	3.00×10^2	3.41×10^2	3.00×10^2	1.58×10^3	1.46×10^3	4.24×10^2	3.00×10^2	2.00×10^2
		Mean	3.00×10^2	3.38×10^2	3.00×10^2	1.49×10^3	1.21×10^3	4.14×10^2	3.17×10^2	2.00×10^2
		Std	0.00×10^0	4.02×10^1	0.00×10^0	2.92×10^2	4.53×10^2	3.09×10^2	2.56×10^2	0.00×10^0
		p-value	-	$\mathbf{2.20 \times 10^{-6+}}$	NaN$^{=}$	$1.21 \times 10^{-12+}$	$1.21 \times 10^{-12+}$	$1.21 \times 10^{-12+}$	$1.21 \times 10^{-12+}$	$1.69 \times 10^{-14-}$
F_{28}		Median	8.41×10^2	8.19×10^2	8.16×10^2	5.49×10^3	4.98×10^3	7.10×10^2	4.78×10^2	2.00×10^2
		Mean	8.40×10^2	8.21×10^2	8.22×10^2	5.56×10^3	5.11×10^3	7.03×10^2	4.76×10^2	2.00×10^2
		Std	2.05×10^1	4.03×10^1	2.02×10^1	1.59×10^3	1.06×10^3	6.79×10^1	5.60×10^0	0.00×10^0
		p-value	-	$2.15 \times 10^{-2-}$	$3.34 \times 10^{-3-}$	$3.01 \times 10^{-11+}$	$3.01 \times 10^{-11+}$	$7.09 \times 10^{-9-}$	$3.01 \times 10^{-11-}$	$1.20 \times 10^{-12-}$
F_{29}		Median	7.21×10^2	7.15×10^2	1.24×10^3	4.17×10^8	4.37×10^8	1.00×10^3	5.88×10^2	2.00×10^2
		Mean	7.28×10^2	1.40×10^6	1.28×10^3	4.14×10^8	4.40×10^8	1.04×10^3	8.95×10^2	2.00×10^2
		Std	2.71×10^1	3.59×10^6	1.39×10^2	1.18×10^8	1.53×10^8	1.60×10^2	8.34×10^2	0.00×10^0
		p-value	-	$\mathbf{1.56 \times 10^{-2+}}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.34 \times 10^{-11+}$	$6.57 \times 10^{-2=}$	$1.21 \times 10^{-12-}$
F_{30}		Median	1.35×10^3	8.73×10^2	2.13×10^3	3.40×10^6	3.33×10^6	9.55×10^2	4.69×10^2	2.00×10^2
		Mean	1.57×10^3	1.00×10^3	2.03×10^3	3.60×10^6	3.41×10^6	1.02×10^3	4.82×10^2	2.00×10^2
		Std	5.78×10^2	4.70×10^2	5.37×10^2	1.45×10^6	1.62×10^6	2.69×10^2	2.16×10^2	0.00×10^0
		p-value	-	$1.43 \times 10^{-5-}$	$\mathbf{5.56 \times 10^{-4+}}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$5.46 \times 10^{-6-}$	$3.02 \times 10^{-11-}$	$1.21 \times 10^{-12-}$
F_{23-30}		$w/t/l$	-	2/2/4	4/2/2	8/0/0	8/0/0	6/0/2	5/1/2	1/1/6
		$w/t/l$	-	13/11/6	23/4/3	27/0/3	27/0/3	25/3/2	23/3/4	19/1/10
		Rank	**2.25**	2.97	4.77	6.43	6.10	5.03	4.83	3.62

Table 3. Comparison between ACSEDA and the compared state-of-the-art EDA variants on the 50-*D* CEC2014 benchmark problems. The bold results indicate that ACSEDA is significantly better than the compared methods.

F	Category	Quality	ACSEDA	EDA2	EDA$_{VERS}$	EDA/LS	EDA/LS-MS	TRA-EDA	BUMDA	MA-ES
F_1		Median	1.42×10^{-14}	7.11×10^{-14}	4.09×10^5	6.84×10^{-9}	1.28×10^{-8}	3.42×10^8	1.61×10^7	1.42×10^{-14}
		Mean	1.14×10^{-14}	7.06×10^{-14}	4.13×10^5	6.41×10^2	9.33×10^7	3.50×10^8	1.63×10^7	1.80×10^{-14}
		Std	6.77×10^{-15}	1.00×10^{-14}	7.25×10^4	2.32×10^3	5.02×10^8	5.08×10^7	3.06×10^6	6.28×10^{-15}
		p-value	-	$\mathbf{5.50 \times 10^{-12+}}$	$\mathbf{9.04 \times 10^{-12+}}$	$\mathbf{9.04 \times 10^{-12+}}$	$\mathbf{9.04 \times 10^{-12+}}$	$\mathbf{9.04 \times 10^{-12+}}$	$\mathbf{9.04 \times 10^{-12+}}$	$\mathbf{5.37 \times 10^{-4+}}$
F_2	Unimodal Problems	Median	5.97×10^{-13}	1.23×10^{-11}	3.30×10^3	1.77×10^{-11}	3.67×10^{-11}	3.27×10^{10}	3.27×10^3	2.84×10^{-14}
		Mean	7.40×10^{-13}	1.23×10^{-11}	3.49×10^3	7.56×10^{-10}	1.59×10^{-9}	3.32×10^{10}	1.56×10^4	3.60×10^{-14}
		Std	3.83×10^{-13}	1.92×10^{-12}	1.76×10^3	2.98×10^{-9}	3.81×10^{-9}	1.79×10^9	2.60×10^4	1.26×10^{-14}
		p-value	-	$\mathbf{2.98 \times 10^{-12+}}$	$\mathbf{2.98 \times 10^{-12+}}$	$\mathbf{8.43 \times 10^{-7+}}$	$\mathbf{1.48 \times 10^{-6+}}$	$\mathbf{2.98 \times 10^{-11+}}$	$\mathbf{2.98 \times 10^{-11+}}$	$8.74 \times 10^{-12-}$
F_3		Median	0.00×10^0	0.00×10^0	4.85×10^3	1.10×10^{-8}	1.11×10^{-8}	2.93×10^4	1.25×10^4	1.14×10^{-13}
		Mean	0.00×10^0	0.00×10^0	5.10×10^3	1.74×10^1	4.35×10^{-5}	2.87×10^4	1.35×10^4	1.00×10^{-13}
		Std	0.00×10^0	0.00×10^0	1.03×10^3	9.36×10^1	1.72×10^{-4}	2.52×10^3	5.05×10^3	2.40×10^{-14}
		p-value	-	NaN$^=$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.97 \times 10^{-13+}}$
F_{1-3}	*w/t/l*		-	2/1/0	3/0/0	3/0/0	3/0/0	3/0/0	3/0/0	2/0/1
F_4		Median	9.81×10^1	5.35×10^0	8.16×10^1	5.71×10^{-9}	1.29×10^{-8}	4.47×10^3	1.25×10^2	1.14×10^{-13}
		Mean	9.25×10^1	3.93×10^1	7.80×10^1	1.04×10^3	3.86×10^3	4.46×10^3	1.28×10^2	3.99×10^{-1}
		Std	7.64×10^0	4.51×10^1	2.13×10^1	5.11×10^3	1.27×10^4	4.24×10^2	1.57×10^1	1.20×10^0
		p-value	-	$1.93 \times 10^{-3-}$	$4.80 \times 10^{-4-}$	$\mathbf{5.29 \times 10^{-9+}}$	$\mathbf{7.12 \times 10^{-8+}}$	$\mathbf{1.62 \times 10^{-11+}}$	$\mathbf{1.10 \times 10^{-10+}}$	$8.76 \times 10^{-12-}$
F_5		Median	2.11×10^1	2.11×10^1	2.11×10^1	2.00×10^1	2.00×10^1	2.11×10^1	2.11×10^1	2.00×10^1
		Mean	2.11×10^1	2.11×10^1	2.11×10^1	2.00×10^1	2.00×10^1	2.11×10^1	2.11×10^1	2.05×10^1
		Std	4.14×10^{-2}	2.54×10^{-2}	3.32×10^{-2}	0.00×10^0	2.21×10^{-2}	4.83×10^{-2}	3.07×10^{-2}	7.59×10^{-1}
		p-value	-	$8.30 \times 10^{-1=}$	$2.51 \times 10^{-2-}$	$1.21 \times 10^{-12-}$	$4.10 \times 10^{-12-}$	$1.45 \times 10^{-1=}$	$1.19 \times 10^{-1=}$	$1.95 \times 10^{-3-}$
F_6	Simple Multimodal Problems	Median	5.22×10^{-5}	1.39×10^{-4}	5.51×10^{-1}	7.19×10^1	6.91×10^1	6.96×10^0	3.64×10^{-1}	4.21×10^1
		Mean	1.74×10^{-2}	2.52×10^{-1}	8.59×10^{-1}	7.17×10^1	6.93×10^1	7.18×10^0	9.50×10^{-1}	4.15×10^1
		Std	9.31×10^{-2}	5.81×10^{-1}	9.72×10^{-1}	3.34×10^0	3.41×10^0	2.00×10^0	1.23×10^0	6.17×10^0
		p-value	-	$\mathbf{3.16 \times 10^{-10+}}$	$\mathbf{1.09 \times 10^{-10+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{1.41 \times 10^{-9+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_7		Median	0.00×10^0	0.00×10^0	3.41×10^{-13}	1.72×10^{-2}	1.97×10^{-2}	2.98×10^2	5.73×10^{-3}	2.27×10^{-13}
		Mean	0.00×10^0	0.00×10^0	9.86×10^{-4}	2.39×10^{-2}	2.18×10^{-2}	2.99×10^2	2.20×10^{-2}	2.47×10^{-4}
		Std	0.00×10^0	0.00×10^0	2.96×10^{-3}	2.69×10^{-2}	2.33×10^{-2}	1.73×10^1	4.80×10^{-2}	1.33×10^{-3}
		p-value	-	NaN$^=$	$\mathbf{6.50 \times 10^{-13+}}$	$\mathbf{1.20 \times 10^{-12+}}$	$\mathbf{1.20 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{5.02 \times 10^{-13+}}$
F_8		Median	2.98×10^0	7.96×10^0	3.09×10^2	4.69×10^2	4.35×10^2	4.93×10^1	0.00×10^0	3.10×10^2
		Mean	2.98×10^0	7.89×10^0	2.96×10^2	4.83×10^2	4.57×10^2	4.82×10^1	3.32×10^{-2}	3.10×10^2
		Std	1.23×10^0	2.56×10^0	5.42×10^1	1.07×10^2	9.68×10^1	6.28×10^0	1.79×10^{-1}	5.00×10^0
		p-value	-	$\mathbf{7.55 \times 10^{-10+}}$	$\mathbf{2.51 \times 10^{-11+}}$	$\mathbf{2.51 \times 10^{-11+}}$	$\mathbf{2.51 \times 10^{-11+}}$	$\mathbf{2.51 \times 10^{-11+}}$	$1.69 \times 10^{-12-}$	$\mathbf{2.47 \times 10^{-11+}}$

Table 3. *Cont.*

F	Category	Quality	ACSEDA	EDA2	EDA$_\text{VERS}$	EDA/LS	EDA/LS-MS	TRA-EDA	BUMDA	MA-ES
F_9		Median	2.49×10^0	6.96×10^0	3.11×10^2	7.00×10^2	6.23×10^2	3.80×10^1	1.99×10^0	4.37×10^2
		Mean	2.49×10^0	6.96×10^0	2.82×10^2	6.95×10^2	6.54×10^2	3.75×10^1	2.14×10^0	4.35×10^2
		Std	1.35×10^0	1.87×10^0	8.86×10^1	1.20×10^2	9.46×10^1	4.75×10^0	2.74×10^0	9.41×10^0
		p-value	-	$\mathbf{1.59 \times 10^{-10+}}$	$\mathbf{2.61 \times 10^{-11+}}$	$\mathbf{2.61 \times 10^{-11+}}$	$\mathbf{2.61 \times 10^{-11+}}$	$\mathbf{2.61 \times 10^{-11+}}$	$2.95 \times 10^{-1=}$	$\mathbf{2.59 \times 10^{-11+}}$
F_{10}		Median	8.44×10^0	1.30×10^2	1.18×10^4	7.57×10^3	7.42×10^3	1.06×10^3	5.79×10^1	7.87×10^3
		Mean	1.07×10^2	1.99×10^2	1.18×10^4	8.30×10^3	8.24×10^3	1.04×10^3	6.25×10^1	7.84×10^3
		Std	1.65×10^2	1.80×10^2	4.28×10^2	2.40×10^3	2.07×10^3	4.09×10^2	4.11×10^1	3.26×10^2
		p-value	-	$\mathbf{5.44 \times 10^{-3+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{9.91 \times 10^{-11+}}$	$6.95 \times 10^{-1=}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{11}		Median	1.36×10^2	1.42×10^2	1.25×10^4	8.39×10^3	8.52×10^3	2.57×10^3	4.52×10^2	8.44×10^3
		Mean	1.18×10^2	1.63×10^2	1.25×10^4	9.46×10^3	9.43×10^3	2.52×10^3	4.58×10^2	8.47×10^3
		Std	9.29×10^0	1.65×10^2	3.67×10^2	2.78×10^3	3.02×10^3	9.47×10^2	2.82×10^2	2.77×10^2
		p-value	-	$2.84 \times 10^{-1=}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.65 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{12}		Median	3.35×10^0	3.30×10^0	3.20×10^0	7.43×10^{-1}	6.52×10^{-1}	3.27×10^0	3.38×10^0	2.85×10^{-2}
		Mean	3.35×10^0	3.27×10^0	3.25×10^0	8.15×10^{-1}	6.83×10^{-1}	3.26×10^0	3.35×10^0	3.26×10^{-2}
		Std	2.52×10^{-1}	2.71×10^{-1}	3.05×10^{-1}	3.04×10^{-1}	2.57×10^{-1}	2.56×10^{-1}	1.99×10^{-1}	1.70×10^{-2}
		p-value	-	$2.84 \times 10^{-1=}$	$1.96 \times 10^{-1=}$	$3.02 \times 10^{-1-}$	$3.02 \times 10^{-11-}$	$2.17 \times 10^{-1=}$	$8.77 \times 10^{-1=}$	$3.02 \times 10^{-11-}$
F_{13}		Median	1.40×10^{-1}	7.28×10^{-2}	2.46×10^{-1}	4.98×10^{-1}	4.50×10^{-1}	3.71×10^0	6.67×10^{-2}	3.63×10^{-1}
		Mean	1.40×10^{-1}	7.48×10^{-2}	2.44×10^{-1}	1.24×10^0	4.97×10^{-1}	3.72×10^0	6.38×10^{-2}	3.63×10^{-1}
		Std	1.89×10^{-2}	1.03×10^{-2}	2.22×10^{-2}	2.15×10^0	1.24×10^{-1}	1.31×10^{-1}	1.03×10^{-2}	6.78×10^{-2}
		p-value	-	$3.02 \times 10^{-11-}$	$\mathbf{4.08 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$3.02 \times 10^{-11-}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{14}		Median	2.57×10^{-1}	4.01×10^{-1}	3.73×10^{-1}	3.82×10^{-1}	3.97×10^{-1}	6.42×10^1	4.36×10^{-1}	3.37×10^{-1}
		Mean	2.40×10^{-1}	3.93×10^{-1}	3.76×10^{-1}	4.58×10^{-1}	4.95×10^{-1}	6.40×10^1	4.27×10^{-1}	4.23×10^{-1}
		Std	6.75×10^{-2}	4.80×10^{-2}	2.87×10^{-2}	1.86×10^{-1}	1.86×10^{-1}	3.95×10^0	3.49×10^{-2}	2.25×10^{-1}
		p-value	-	$\mathbf{5.07 \times 10^{-10+}}$	$\mathbf{1.61 \times 10^{-10+}}$	$\mathbf{2.87 \times 10^{-10+}}$	$\mathbf{3.16 \times 10^{-10+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{4.08 \times 10^{-11+}}$	$\mathbf{1.87 \times 10^{-7+}}$
F_{15}		Median	4.87×10^0	1.23×10^1	2.74×10^1	1.84×10^2	1.73×10^2	2.39×10^3	5.30×10^0	6.61×10^0
		Mean	4.81×10^0	1.19×10^1	2.64×10^1	1.84×10^2	1.82×10^2	2.45×10^2	5.27×10^0	6.61×10^0
		Std	4.93×10^{-1}	1.79×10^0	4.02×10^0	4.72×10^1	5.73×10^1	9.48×10^2	6.27×10^{-1}	1.37×10^0
		p-value	-	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.34 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{2.38 \times 10^{-3+}}$	$\mathbf{7.69 \times 10^{-8+}}$
F_{16}		Median	1.80×10^1	2.18×10^1	2.07×10^1	2.28×10^1	2.25×10^1	2.01×10^1	1.98×10^1	2.24×10^1
		Mean	1.80×10^1	2.18×10^1	2.07×10^1	2.28×10^1	2.26×10^1	2.00×10^1	1.97×10^1	2.24×10^1
		Std	6.67×10^{-1}	2.58×10^{-1}	3.75×10^{-1}	5.14×10^{-1}	4.27×10^{-1}	2.92×10^{-1}	4.87×10^{-1}	1.40×10^{-1}
		p-value	-	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{1.09 \times 10^{-10+}}$	$\mathbf{3.02 \times 10^{-11+}}$

Table 3. *Cont.*

F	Category	Quality	ACSEDA	EDA2	EDA$_{\text{VERS}}$	EDA/LS	EDA/LS-MS	TRA-EDA	BUMDA	MA-ES
F_{4-16}	*w/t/l*		-	7/4/2	11/1/1	11/0/2	11/0/2	11/2/0	07/4/2	10/0/3
F_{17}		Median	7.79×10^1	3.16×10^1	3.45×10^4	5.74×10^7	1.49×10^4	1.96×10^7	1.69×10^7	2.45×10^3
		Mean	1.16×10^2	3.77×10^1	3.62×10^4	3.42×10^8	1.04×10^8	2.02×10^7	1.76×10^7	2.49×10^3
		Std	7.34×10^1	2.47×10^1	1.57×10^4	4.09×10^8	1.77×10^8	4.76×10^6	3.45×10^6	5.06×10^2
		p-value	-	$1.10 \times 10^{-11-}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{18}		Median	7.84×10^0	1.40×10^0	1.58×10^2	2.88×10^2	2.98×10^2	5.85×10^8	1.02×10^3	1.66×10^2
		Mean	8.47×10^0	1.35×10^0	2.13×10^2	3.49×10^9	1.34×10^9	6.42×10^8	1.08×10^3	1.65×10^2
		Std	3.23×10^0	6.37×10^{-1}	1.59×10^2	7.03×10^9	3.62×10^9	1.93×10^8	5.06×10^2	4.18×10^1
		p-value	-	$3.02 \times 10^{-11-}$	$\mathbf{4.98 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{19}	Hybrid Problems	Median	1.09×10^1	7.36×10^0	3.50×10^1	2.53×10^1	2.55×10^1	5.99×10^1	6.30×10^1	1.85×10^1
		Mean	1.08×10^1	7.46×10^0	3.48×10^1	2.59×10^1	9.34×10^1	5.81×10^1	6.07×10^1	1.84×10^1
		Std	7.71×10^{-1}	7.65×10^{-1}	4.99×10^0	2.77×10^0	3.64×10^2	9.80×10^0	9.75×10^0	2.52×10^0
		p-value	-	$4.50 \times 10^{-11-}$	$\mathbf{4.50 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{4.08 \times 10^{-11+}}$
F_{20}		Median	1.96×10^0	2.52×10^0	2.33×10^3	6.41×10^4	8.45×10^4	5.01×10^2	3.41×10^4	3.01×10^2
		Mean	1.95×10^0	2.50×10^0	2.37×10^3	9.01×10^4	7.81×10^4	5.08×10^2	3.35×10^4	3.27×10^2
		Std	2.27×10^{-1}	3.48×10^{-1}	7.57×10^2	1.47×10^5	4.91×10^4	2.60×10^2	6.10×10^3	1.35×10^2
		p-value	-	$\mathbf{1.16 \times 10^{-7+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{21}		Median	2.47×10^2	1.23×10^2	7.13×10^4	2.35×10^7	1.23×10^6	1.64×10^3	8.45×10^6	1.75×10^3
		Mean	1.99×10^2	9.74×10^1	7.27×10^4	8.60×10^7	1.83×10^7	1.98×10^3	8.53×10^6	1.76×10^3
		Std	8.11×10^1	7.43×10^1	1.94×10^4	1.08×10^8	3.03×10^7	1.49×10^3	2.01×10^6	4.26×10^2
		p-value	-	$8.15 \times 10^{-5-}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.34 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{22}		Median	2.72×10^1	3.02×10^1	9.51×10^2	1.89×10^3	2.11×10^3	9.87×10^1	4.69×10^1	7.94×10^2
		Mean	3.54×10^1	3.02×10^1	9.06×10^2	9.23×10^3	3.33×10^3	1.18×10^2	1.37×10^2	7.58×10^2
		Std	3.02×10^1	6.70×10^{-1}	1.90×10^2	2.74×10^4	5.36×10^3	6.15×10^1	1.46×10^2	2.84×10^2
		p-value	-	$1.36 \times 10^{-7-}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{2.23 \times 10^{-9+}}$	$\mathbf{1.17 \times 10^{-9+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{17-22}	*w/t/l*		-	1/0/5	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0	6/0/0
F_{23}	Composition Problems	Median	3.44×10^2	3.44×10^2	3.44×10^2	2.48×10^3	1.70×10^3	4.17×10^2	3.66×10^2	2.00×10^2
		Mean	3.44×10^2	3.44×10^2	3.44×10^2	2.20×10^3	1.52×10^3	4.21×10^2	3.66×10^2	2.00×10^2
		Std	1.71×10^{-13}	1.71×10^{-13}	1.71×10^{-13}	9.29×10^2	6.07×10^2	1.44×10^1	2.74×10^0	0.00×10^0
		p-value	-	NaN$^=$	NaN$^=$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$1.69 \times 10^{-14-}$
F_{24}		Median	2.68×10^2	2.68×10^2	2.67×10^2	4.09×10^2	3.60×10^2	2.72×10^2	2.78×10^2	2.00×10^2
		Mean	2.68×10^2	2.68×10^2	2.64×10^2	4.78×10^2	4.44×10^2	2.72×10^2	2.78×10^2	2.00×10^2
		Std	1.33×10^0	1.10×10^0	5.66×10^0	1.69×10^2	1.40×10^2	8.52×10^{-1}	1.11×10^0	0.00×10^0
		p-value	-	$9.17 \times 10^{-1=}$	$3.00 \times 10^{-1=}$	$\mathbf{2.88 \times 10^{-11+}}$	$\mathbf{2.88 \times 10^{-11+}}$	$\mathbf{5.25 \times 10^{-11+}}$	$\mathbf{2.88 \times 10^{-11+}}$	$1.14 \times 10^{-12-}$

Table 3. *Cont.*

F	Category	Quality	ACSEDA	EDA2	EDA$_{\text{VERS}}$	EDA/LS	EDA/LS-MS	TRA-EDA	BUMDA	MA-ES
F_{25}		Median	2.05×10^2	2.05×10^2	2.00×10^2	5.77×10^2	4.32×10^2	2.17×10^2	2.14×10^2	2.00×10^2
		Mean	2.05×10^2	2.05×10^2	2.00×10^2	5.57×10^2	4.29×10^2	2.17×10^2	2.15×10^2	2.00×10^2
		Std	1.49×10^{-1}	1.58×10^{-1}	0.00×10^0	8.87×10^1	5.87×10^1	1.56×10^0	5.84×10^0	0.00×10^0
		p-value	-	$4.12 \times 10^{-1=}$	$1.19 \times 10^{-12-}$	$\mathbf{2.98 \times 10^{-11+}}$	$\mathbf{2.98 \times 10^{-11+}}$	$\mathbf{2.98 \times 10^{-11+}}$	$\mathbf{2.98 \times 10^{-11+}}$	$1.19 \times 10^{-12-}$
F_{26}		Median	1.00×10^2	1.00×10^2	2.00×10^2	1.01×10^2	1.01×10^2	1.06×10^2	1.74×10^2	2.00×10^2
		Mean	1.00×10^2	1.00×10^2	1.96×10^2	2.90×10^2	1.72×10^2	1.06×10^2	1.58×10^2	2.00×10^2
		Std	4.78×10^{-2}	1.35×10^{-2}	1.64×10^1	2.27×10^2	1.18×10^2	1.35×10^0	4.00×10^1	0.00×10^0
		p-value	-	$8.98 \times 10^{-11-}$	$\mathbf{2.80 \times 10^{-11+}}$	$6.06 \times 10^{-11+}$	$1.09 \times 10^{-10+}$	$\mathbf{2.98 \times 10^{-11+}}$	$\mathbf{2.98 \times 10^{-11+}}$	$\mathbf{1.21 \times 10^{-11+}}$
F_{27}		Median	3.00×10^2	4.19×10^2	3.52×10^2	2.68×10^3	2.51×10^3	7.00×10^2	3.68×10^2	2.00×10^2
		Mean	3.17×10^2	4.23×10^2	3.48×10^2	2.67×10^3	2.49×10^3	7.00×10^2	3.74×10^2	2.00×10^2
		Std	2.20×10^1	4.98×10^1	4.35×10^1	1.79×10^2	1.16×10^2	5.13×10^1	2.72×10^1	0.00×10^0
		p-value	-	$\mathbf{5.37 \times 10^{-11+}}$	$\mathbf{5.79 \times 10^{-6+}}$	$\mathbf{2.95 \times 10^{-11+}}$	$\mathbf{2.95 \times 10^{-11+}}$	$\mathbf{2.95 \times 10^{-11+}}$	$\mathbf{4.11 \times 10^{-9+}}$	$1.18 \times 10^{-12-}$
F_{28}		Median	1.16×10^3	1.16×10^3	1.15×10^3	1.15×10^4	1.02×10^4	1.92×10^3	4.29×10^2	2.00×10^2
		Mean	1.16×10^3	1.19×10^3	1.15×10^3	1.15×10^4	9.91×10^3	2.01×10^3	4.31×10^2	2.00×10^2
		Std	4.37×10^1	1.68×10^2	3.88×10^1	2.14×10^3	2.17×10^3	4.59×10^2	6.89×10^0	0.00×10^0
		p-value	-	$9.35 \times 10^{-1=}$	$1.81 \times 10^{-1=}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{1.33 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$1.21 \times 10^{-12-}$
F_{29}		Median	8.14×10^2	7.39×10^2	1.66×10^3	1.57×10^9	1.05×10^9	1.23×10^3	7.29×10^2	2.00×10^2
		Mean	8.34×10^2	4.24×10^6	1.67×10^3	1.57×10^9	1.06×10^9	2.65×10^3	9.82×10^2	2.00×10^2
		Std	5.15×10^1	1.63×10^7	1.50×10^2	3.30×10^8	2.06×10^8	2.81×10^3	7.50×10^2	0.00×10^0
		p-value	-	$\mathbf{6.77 \times 10^{-5+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{9.92 \times 10^{-11+}}$	$1.33 \times 10^{-1=}$	$1.21 \times 10^{-12-}$
F_{30}		Median	8.56×10^3	9.37×10^3	9.02×10^3	2.49×10^7	1.83×10^7	1.74×10^5	8.19×10^2	2.00×10^2
		Mean	8.57×10^3	9.50×10^3	9.20×10^3	2.66×10^7	1.92×10^7	1.67×10^5	8.17×10^2	2.00×10^2
		Std	3.37×10^2	7.30×10^2	3.67×10^2	1.14×10^7	6.42×10^6	2.95×10^4	1.82×10^2	0.00×10^0
		p-value	-	$\mathbf{1.07 \times 10^{-7+}}$	$\mathbf{7.77 \times 10^{-9+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$1.21 \times 10^{-12-}$
F_{23-30}	*w/t/l*		-	3/4/1	4/3/1	8/0/0	8/0/0	8/0/0	5/1/2	1/0/7
	w/t/l		-	13/9/8	24/4/2	28/0/2	28/0/2	28/2/0	21/5/4	19/0/11
	Rank		**2.33**	2.93	4.52	6.80	6.27	5.70	4.30	3.15

Table 4. Comparison between ACSEDA and the compared state-of-the-art EDA variants on the 100-*D* CEC2014 benchmark problems. The bold results indicate that ACSEDA is significantly better than the compared methods.

F	Category	Quality	ACSEDA	EDA2	EDA$_{VERS}$	EDA/LS	EDA/LS-MS	TRA-EDA	BUMDA	MA-ES
F_1		Median	1.80×10^{-9}	6.82×10^{-13}	1.25×10^{6}	1.87×10^{6}	1.39×10^{3}	4.74×10^{8}	8.33×10^{7}	4.26×10^{-14}
		Mean	1.90×10^{-9}	7.24×10^{-13}	1.22×10^{6}	8.24×10^{9}	4.41×10^{8}	4.62×10^{8}	8.48×10^{7}	1.46×10^{-13}
		Std	7.43×10^{-9}	1.70×10^{-13}	1.68×10^{5}	8.95×10^{9}	1.03×10^{9}	6.07×10^{7}	8.05×10^{6}	5.46×10^{-13}
		p-value	-	$2.99 \times 10^{-11-}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$1.20 \times 10^{-11-}$
F_2	Unimodal Problems	Median	3.84×10^{-7}	9.36×10^{-11}	9.84×10^{3}	1.88×10^{-10}	1.14×10^{-10}	9.69×10^{10}	4.27×10^{4}	8.53×10^{-14}
		Mean	3.94×10^{-7}	9.67×10^{-11}	1.12×10^{4}	1.54×10^{-9}	3.87×10^{1}	9.74×10^{10}	7.15×10^{4}	7.48×10^{-14}
		Std	2.02×10^{-7}	2.78×10^{-11}	5.42×10^{3}	3.04×10^{-9}	2.08×10^{2}	3.46×10^{9}	9.73×10^{4}	1.55×10^{-14}
		p-value	-	$\mathbf{3.02 \times 10^{-11-}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{5.57 \times 10^{-10+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$1.48 \times 10^{-11-}$
F_3		Median	5.40×10^{-13}	0.00×10^{0}	6.11×10^{3}	2.13×10^{-8}	2.57×10^{-8}	1.11×10^{5}	2.39×10^{4}	1.71×10^{-13}
		Mean	7.09×10^{-13}	0.00×10^{0}	6.25×10^{3}	9.02×10^{-1}	4.18×10^{2}	1.10×10^{5}	2.34×10^{4}	2.01×10^{-13}
		Std	4.71×10^{-13}	0.00×10^{0}	1.48×10^{3}	4.31×10^{0}	1.13×10^{3}	5.11×10^{3}	3.80×10^{3}	4.08×10^{-14}
		p-value	-	$1.18 \times 10^{-12-}$	$\mathbf{2.96 \times 10^{-11+}}$	$\mathbf{2.96 \times 10^{-11+}}$	$\mathbf{2.96 \times 10^{-11+}}$	$\mathbf{2.96 \times 10^{-11+}}$	$\mathbf{2.96 \times 10^{-11+}}$	$4.68 \times 10^{-9-}$
F_{1-3}	*w/t/l*		-	0/0/3	3/0/0	2/0/1	3/0/0	3/0/0	3/0/0	0/0/3
F_4		Median	1.85×10^{2}	2.01×10^{2}	1.43×10^{2}	3.99×10^{0}	1.37×10^{-7}	1.27×10^{4}	1.91×10^{2}	1.71×10^{-13}
		Mean	1.81×10^{2}	1.92×10^{2}	1.40×10^{2}	1.04×10^{4}	4.09×10^{3}	1.27×10^{4}	1.82×10^{2}	1.20×10^{0}
		Std	2.93×10^{1}	2.69×10^{1}	7.29×10^{0}	3.91×10^{4}	1.31×10^{4}	7.87×10^{2}	2.44×10^{1}	1.83×10^{0}
		p-value	-	$3.76 \times 10^{-1=}$	$1.83 \times 10^{-9-}$	$\mathbf{3.51 \times 10^{-9+}}$	$\mathbf{1.45 \times 10^{-7+}}$	$\mathbf{2.61 \times 10^{-11+}}$	$9.70 \times 10^{-1=}$	$1.14 \times 10^{-11-}$
F_5		Median	2.13×10^{1}	2.13×10^{1}	2.13×10^{1}	2.00×10^{1}	2.00×10^{1}	2.13×10^{1}	2.13×10^{1}	2.00×10^{1}
		Mean	2.13×10^{1}	2.13×10^{1}	2.13×10^{1}	2.00×10^{1}	2.00×10^{1}	2.13×10^{1}	2.13×10^{1}	2.03×10^{1}
		Std	2.74×10^{-2}	2.56×10^{-2}	2.46×10^{-2}	1.47×10^{-2}	0.00×10^{0}	2.28×10^{-2}	3.62×10^{-2}	6.25×10^{-1}
		p-value	-	$1.70 \times 10^{-2-}$	$1.86 \times 10^{-1=}$	$3.16 \times 10^{-12-}$	$1.21 \times 10^{-12-}$	$1.69 \times 10^{-1=}$	$5.44 \times 10^{-1=}$	$9.19 \times 10^{-6-}$
F_6	Simple Multimodal Problems	Median	4.10×10^{-3}	1.94×10^{0}	1.01×10^{1}	1.55×10^{2}	6.88×10^{1}	3.37×10^{1}	4.93×10^{0}	1.19×10^{2}
		Mean	3.83×10^{-1}	2.43×10^{0}	1.06×10^{1}	1.56×10^{2}	6.96×10^{1}	3.39×10^{1}	5.05×10^{0}	1.20×10^{2}
		Std	5.70×10^{-1}	1.85×10^{0}	3.51×10^{0}	5.33×10^{0}	2.75×10^{0}	3.54×10^{0}	1.58×10^{0}	7.18×10^{0}
		p-value	-	$\mathbf{9.51 \times 10^{-6+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.34 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_7		Median	2.27×10^{-13}	0.00×10^{0}	1.02×10^{-12}	1.90×10^{-8}	1.72×10^{-2}	1.10×10^{3}	5.50×10^{-2}	3.41×10^{-13}
		Mean	2.31×10^{-13}	0.00×10^{0}	1.00×10^{-12}	9.27×10^{-3}	2.65×10^{-2}	1.10×10^{3}	7.16×10^{-2}	4.93×10^{-4}
		Std	1.19×10^{-13}	0.00×10^{0}	1.84×10^{-13}	2.21×10^{-2}	2.82×10^{-2}	3.23×10^{1}	5.06×10^{-2}	1.84×10^{-3}
		p-value	-	$8.27 \times 10^{-13-}$	$\mathbf{1.98 \times 10^{-11+}}$	$\mathbf{2.22 \times 10^{-11+}}$	$\mathbf{2.25 \times 10^{-11+}}$	$\mathbf{2.25 \times 10^{-11+}}$	$\mathbf{2.25 \times 10^{-11+}}$	$\mathbf{4.97 \times 10^{-6+}}$
F_8		Median	9.45×10^{0}	2.39×10^{1}	7.33×10^{2}	1.56×10^{3}	4.64×10^{2}	2.71×10^{2}	0.00×10^{0}	6.15×10^{2}
		Mean	9.22×10^{0}	2.37×10^{1}	5.79×10^{2}	1.48×10^{3}	4.93×10^{2}	2.75×10^{2}	1.33×10^{-1}	6.13×10^{2}
		Std	2.52×10^{0}	4.46×10^{0}	2.94×10^{2}	3.00×10^{2}	1.31×10^{2}	1.62×10^{1}	3.38×10^{-1}	1.08×10^{1}
		p-value	-	$\mathbf{3.11 \times 10^{-11+}}$	$\mathbf{2.88 \times 10^{-11+}}$	$\mathbf{2.88 \times 10^{-11+}}$	$\mathbf{2.88 \times 10^{-11+}}$	$\mathbf{2.88 \times 10^{-11+}}$	$3.86 \times 10^{-12-}$	$\mathbf{2.86 \times 10^{-11+}}$

Table 4. *Cont.*

F	Category	Quality	ACSEDA	EDA2	EDA$_{\text{VERS}}$	EDA/LS	EDA/LS-MS	TRA-EDA	BUMDA	MA-ES
F_9		Median	7.96×10^0	2.14×10^1	7.43×10^2	1.68×10^3	5.92×10^2	2.77×10^2	2.98×10^0	8.09×10^2
		Mean	8.06×10^0	2.08×10^1	6.50×10^2	1.68×10^3	6.27×10^2	2.78×10^2	3.26×10^0	8.06×10^2
		Std	2.29×10^0	4.72×10^0	2.40×10^2	2.62×10^2	1.31×10^2	1.83×10^1	1.46×10^0	1.52×10^1
		p-value	-	$\mathbf{4.34 \times 10^{-11+}}$	$\mathbf{2.78 \times 10^{-11+}}$	$\mathbf{2.78 \times 10^{-11+}}$	$\mathbf{2.78 \times 10^{-11+}}$	$\mathbf{2.78 \times 10^{-11+}}$	$2.22 \times 10^{-9-}$	$\mathbf{2.77 \times 10^{-11+}}$
F_{10}		Median	6.11×10^2	1.63×10^3	2.79×10^4	1.78×10^4	7.72×10^3	6.51×10^3	1.01×10^2	1.64×10^4
		Mean	6.60×10^2	1.79×10^3	2.78×10^4	2.20×10^4	8.58×10^3	6.55×10^3	1.17×10^2	1.65×10^4
		Std	3.85×10^2	7.29×10^2	7.77×10^2	7.13×10^3	2.27×10^3	7.46×10^2	1.16×10^2	4.17×10^2
		p-value	-	$\mathbf{1.55 \times 10^{-9+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$1.20 \times 10^{-8-}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{11}		Median	1.43×10^3	1.81×10^3	2.89×10^4	2.17×10^4	7.99×10^3	5.73×10^3	1.25×10^3	1.31×10^4
		Mean	1.44×10^3	1.85×10^3	2.87×10^4	2.29×10^4	8.34×10^3	5.66×10^3	1.23×10^3	1.31×10^4
		Std	3.41×10^2	4.59×10^2	9.63×10^2	6.21×10^3	1.97×10^3	4.87×10^2	5.13×10^2	4.55×10^2
		p-value	-	$\mathbf{1.24 \times 10^{-3+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$6.57 \times 10^{-2=}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{12}		Median	3.80×10^0	3.97×10^0	3.99×10^0	1.04×10^0	6.78×10^{-1}	4.00×10^0	3.93×10^0	2.03×10^{-2}
		Mean	3.85×10^0	3.99×10^0	3.97×10^0	1.19×10^0	7.28×10^{-1}	3.93×10^0	3.96×10^0	2.03×10^{-2}
		Std	2.16×10^{-1}	2.21×10^{-1}	2.00×10^{-1}	5.49×10^{-1}	2.81×10^{-1}	3.04×10^{-1}	2.10×10^{-1}	6.72×10^{-3}
		p-value	-	$\mathbf{3.15 \times 10^{-2+}}$	$\mathbf{3.92 \times 10^{-2+}}$	$3.02 \times 10^{-11-}$	$3.02 \times 10^{-11-}$	$5.75 \times 10^{-2=}$	$7.98 \times 10^{-2=}$	$3.02 \times 10^{-11-}$
F_{13}		Median	2.38×10^{-1}	1.33×10^{-1}	3.15×10^{-1}	4.35×10^{-1}	4.99×10^{-1}	5.54×10^0	5.77×10^{-2}	5.49×10^{-1}
		Mean	2.36×10^{-1}	1.35×10^{-1}	3.23×10^{-1}	2.39×10^0	8.72×10^{-1}	5.54×10^0	5.85×10^{-2}	5.57×10^{-1}
		Std	2.01×10^{-2}	1.85×10^{-2}	2.55×10^{-2}	3.94×10^0	1.17×10^0	7.78×10^{-2}	9.94×10^{-3}	7.13×10^{-2}
		p-value	-	$3.02 \times 10^{-11-}$	$\mathbf{3.69 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{2.15 \times 10^{-10+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$3.02 \times 10^{-11-}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{14}		Median	2.74×10^{-1}	4.00×10^{-1}	3.59×10^{-1}	1.44×10^2	3.77×10^{-1}	3.29×10^2	4.56×10^{-1}	2.92×10^{-1}
		Mean	2.70×10^{-1}	3.86×10^{-1}	3.63×10^{-1}	2.40×10^2	4.26×10^{-1}	3.27×10^2	4.50×10^{-1}	3.66×10^{-1}
		Std	4.12×10^{-2}	4.86×10^{-2}	1.83×10^{-2}	2.63×10^2	1.50×10^{-1}	7.41×10^0	2.05×10^{-2}	2.36×10^{-1}
		p-value	-	$\mathbf{3.82 \times 10^{-9+}}$	$\mathbf{4.98 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{1.29 \times 10^{-9+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$
F_{15}		Median	1.04×10^1	1.04×10^1	6.55×10^1	5.57×10^2	1.93×10^2	1.90×10^5	1.20×10^1	1.38×10^1
		Mean	1.06×10^1	1.03×10^1	6.01×10^1	2.48×10^7	2.48×10^5	1.90×10^5	1.22×10^1	1.45×10^1
		Std	8.45×10^{-1}	8.26×10^{-1}	1.66×10^1	7.66×10^7	1.33×10^6	3.06×10^4	1.07×10^0	2.62×10^0
		p-value	-	$3.33 \times 10^{-1=}$	$\mathbf{1.29 \times 10^{-9+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{1.29 \times 10^{-6+}}$	$\mathbf{1.69 \times 10^{-9+}}$
F_{16}		Median	4.12×10^1	4.60×10^1	4.53×10^1	4.65×10^1	2.26×10^1	4.33×10^1	4.26×10^1	4.59×10^1
		Mean	4.12×10^1	4.59×10^1	4.53×10^1	4.64×10^1	2.26×10^1	4.33×10^1	4.26×10^1	4.58×10^1
		Std	5.74×10^{-1}	2.25×10^{-1}	3.31×10^{-1}	9.07×10^{-1}	4.77×10^{-1}	4.56×10^{-1}	5.66×10^{-1}	1.42×10^{-1}
		p-value	-	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$3.02 \times 10^{-11-}$	$\mathbf{3.69 \times 10^{-11+}}$	$\mathbf{1.07 \times 10^{-9+}}$	$\mathbf{3.02 \times 10^{-11+}}$

Table 4. *Cont.*

F	Category	Quality	ACSEDA	EDA2	EDA$_{VERS}$	EDA/LS	EDA/LS-MS	TRA-EDA	BUMDA	MA-ES
F_{4-16}	$w/t/l$		-	8/2/3	11/1/1	11/0/2	10/0/3	11/2/0	5/4/4	10/0/3
F_{17}		Median	7.10×10^2	8.16×10^2	2.78×10^5	1.29×10^9	1.33×10^7	9.57×10^7	3.87×10^7	5.48×10^3
		Mean	7.27×10^2	7.72×10^2	2.74×10^5	1.32×10^9	1.19×10^8	9.54×10^7	3.85×10^7	5.59×10^3
		Std	1.93×10^2	2.45×10^2	5.52×10^4	9.08×10^8	1.52×10^8	1.15×10^7	3.48×10^6	6.64×10^2
		p-value	-	$3.26 \times 10^{-1=}$	$\mathbf{3.02 \times 10^{-11+}}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$
F_{18}		Median	5.80×10^1	2.62×10^1	2.02×10^2	5.64×10^8	2.47×10^2	3.04×10^9	1.36×10^2	3.63×10^2
		Mean	5.89×10^1	2.60×10^1	2.54×10^2	1.97×10^{10}	1.41×10^9	3.01×10^9	1.73×10^2	3.63×10^2
		Std	1.37×10^1	7.82×10^0	1.87×10^2	2.58×10^{10}	3.71×10^9	4.29×10^8	1.43×10^2	6.79×10^1
		p-value	-	$7.39 \times 10^{-11-}$	$\mathbf{7.12 \times 10^{-9+}}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$\mathbf{2.50 \times 10^{-3+}}$	$3.02 \times 10^{-11+}$
F_{19}		Median	9.09×10^1	9.40×10^1	9.44×10^1	5.32×10^1	2.61×10^1	5.03×10^2	6.97×10^1	6.03×10^1
	Hybrid Problems	Mean	9.10×10^1	9.40×10^1	9.44×10^1	2.43×10^3	2.57×10^1	5.14×10^2	7.21×10^1	6.28×10^1
		Std	1.21×10^0	1.38×10^0	1.41×10^0	4.59×10^3	2.12×10^0	6.67×10^1	1.71×10^1	1.41×10^1
		p-value	-	$\mathbf{2.67 \times 10^{-9+}}$	$\mathbf{3.16 \times 10^{-10+}}$	$1.86 \times 10^{-1=}$	$3.02 \times 10^{-11-}$	$3.02 \times 10^{-11+}$	$9.51 \times 10^{-6-}$	$8.48 \times 10^{-9-}$
F_{20}		Median	6.20×10^0	9.31×10^0	8.31×10^3	2.43×10^5	6.28×10^4	2.68×10^4	1.08×10^5	6.01×10^2
		Mean	6.68×10^0	9.56×10^0	8.43×10^3	2.33×10^6	6.75×10^4	2.60×10^4	1.08×10^5	6.36×10^2
		Std	2.34×10^0	2.40×10^0	2.00×10^3	5.76×10^6	3.67×10^4	2.96×10^3	1.89×10^4	1.92×10^2
		p-value	-	$\mathbf{8.88 \times 10^{-6+}}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$
F_{21}		Median	2.23×10^2	1.83×10^2	1.59×10^5	9.21×10^8	5.22×10^6	3.03×10^6	2.42×10^7	3.58×10^3
		Mean	2.51×10^2	1.98×10^2	1.51×10^5	8.09×10^8	2.02×10^7	3.18×10^6	2.40×10^7	3.51×10^3
		Std	8.04×10^1	6.34×10^1	3.04×10^4	5.30×10^8	3.12×10^7	1.09×10^6	2.95×10^6	5.75×10^2
		p-value	-	$1.08 \times 10^{-2-}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$
F_{22}		Median	3.89×10^1	3.75×10^1	3.54×10^3	4.09×10^3	1.86×10^3	7.75×10^1	2.58×10^2	1.20×10^3
		Mean	4.62×10^1	6.56×10^1	3.49×10^3	1.07×10^5	2.43×10^3	9.21×10^1	2.90×10^2	1.26×10^3
		Std	3.11×10^1	5.21×10^1	2.48×10^2	2.59×10^5	1.73×10^3	4.21×10^1	2.09×10^2	3.12×10^2
		p-value	-	$6.00 \times 10^{-1=}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$\mathbf{5.00 \times 10^{-9+}}$	$\mathbf{1.43 \times 10^{-8+}}$	$3.02 \times 10^{-11+}$
F_{17-22}	$w/t/l$		-	2/2/2	6/0/0	5/1/0	5/0/1	6/0/0	5/0/1	5/0/1
F_{23}		Median	3.48×10^2	3.48×10^2	3.48×10^2	6.57×10^3	1.64×10^3	4.43×10^2	3.59×10^2	2.00×10^2
		Mean	3.48×10^2	3.48×10^2	3.48×10^2	6.72×10^3	1.44×10^3	4.41×10^2	3.59×10^2	2.00×10^2
		Std	0.00×10^0	0.00×10^0	0.00×10^0	1.60×10^3	6.17×10^2	1.26×10^1	2.91×10^0	0.00×10^0
	Composition Problems	p-value	-	NaN$^=$	NaN$^=$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{3.36 \times 10^{-11+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$\mathbf{1.21 \times 10^{-12+}}$	$1.69 \times 10^{-14-}$
F_{24}		Median	3.72×10^2	3.72×10^2	3.62×10^2	5.05×10^2	3.39×10^2	3.82×10^2	3.82×10^2	2.00×10^2
		Mean	3.73×10^2	3.73×10^2	3.61×10^2	5.07×10^2	4.29×10^2	3.82×10^2	3.82×10^2	2.00×10^2
		Std	2.22×10^0	1.90×10^0	4.63×10^0	2.74×10^1	1.32×10^2	1.41×10^0	1.59×10^0	0.00×10^0
		p-value	-	$7.73 \times 10^{-1=}$	$4.08 \times 10^{-11-}$	$\mathbf{3.02 \times 10^{-11+}}$	$1.91 \times 10^{-1=}$	$\mathbf{3.02 \times 10^{-11+}}$	$3.02 \times 10^{-11+}$	$1.21 \times 10^{-12-}$

Table 4. *Cont.*

F	Category	Quality	ACSEDA	EDA2	EDA$_{\mathrm{VERS}}$	EDA/LS	EDA/LS-MS	TRA-EDA	BUMDA	MA-ES
F_{25}		Median	2.15×10^2	2.16×10^2	2.00×10^2	1.35×10^3	4.27×10^2	2.42×10^2	2.22×10^2	2.00×10^2
		Mean	2.15×10^2	2.16×10^2	2.00×10^2	1.37×10^3	4.20×10^2	2.42×10^2	2.23×10^2	2.00×10^2
		Std	3.90×10^{-1}	8.62×10^{-1}	0.00×10^0	2.04×10^2	5.42×10^1	3.06×10^0	9.36×10^0	0.00×10^0
		p-value	-	$\mathbf{1.11 \times 10^{-4+}}$	$1.21 \times 10^{-12-}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$\mathbf{2.39 \times 10^{-4+}}$	$1.21 \times 10^{-12-}$
F_{26}		Median	2.00×10^2	2.00×10^2	2.00×10^2	1.07×10^3	1.01×10^2	2.00×10^2	2.00×10^2	2.00×10^2
		Mean	2.00×10^2	2.00×10^2	2.00×10^2	8.64×10^2	1.25×10^2	2.00×10^2	2.00×10^2	2.00×10^2
		Std	2.50×10^{-4}	3.97×10^{-4}	7.67×10^{-3}	4.77×10^2	6.80×10^1	2.31×10^{-3}	4.88×10^{-2}	0.00×10^0
		p-value	-	$3.16 \times 10^{-1=}$	$\mathbf{1.96 \times 10^{-12+}}$	$9.65 \times 10^{-8+}$	$1.02 \times 10^{-6-}$	$\mathbf{9.17 \times 10^{-6+}}$	$\mathbf{2.59 \times 10^{-11+}}$	$9.91 \times 10^{-13-}$
F_{27}		Median	3.00×10^2	4.65×10^2	3.98×10^2	5.45×10^3	2.42×10^3	1.78×10^3	3.13×10^2	2.00×10^2
		Mean	3.04×10^2	4.96×10^2	4.15×10^2	5.50×10^3	2.43×10^3	1.78×10^3	3.23×10^2	2.00×10^2
		Std	7.91×10^0	1.13×10^2	5.12×10^1	3.00×10^2	1.17×10^2	1.05×10^2	2.48×10^1	0.00×10^0
		p-value	-	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{3.02 \times 10^{-11+}}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$\mathbf{3.02 \times 10^{-11+}}$	$\mathbf{7.60 \times 10^{-7+}}$	$1.21 \times 10^{-12-}$
F_{28}		Median	2.31×10^3	2.79×10^3	2.17×10^3	2.99×10^4	1.07×10^4	3.99×10^3	8.14×10^2	2.00×10^2
		Mean	2.31×10^3	2.74×10^3	2.17×10^3	2.98×10^4	1.05×10^4	4.45×10^3	8.14×10^2	2.00×10^2
		Std	1.10×10^2	5.59×10^2	5.72×10^1	5.35×10^3	1.89×10^3	1.10×10^3	1.85×10^1	0.00×10^0
		p-value	-	$\mathbf{1.17 \times 10^{-2+}}$	$2.38 \times 10^{-7-}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11-}$	$1.21 \times 10^{-12-}$
F_{29}		Median	1.13×10^3	7.44×10^2	1.79×10^3	5.47×10^9	9.15×10^8	1.49×10^5	3.32×10^2	2.00×10^2
		Mean	1.26×10^3	7.85×10^2	1.80×10^3	5.60×10^9	9.62×10^8	1.69×10^5	4.54×10^2	2.00×10^2
		Std	2.79×10^2	1.55×10^2	4.30×10^1	1.23×10^9	2.57×10^8	1.26×10^5	3.73×10^2	0.00×10^0
		p-value	-	$3.16 \times 10^{-10-}$	$\mathbf{3.82 \times 10^{-10+}}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$5.00 \times 10^{-9-}$	$1.21 \times 10^{-12-}$
F_{30}		Median	9.17×10^3	4.37×10^3	8.66×10^3	3.02×10^8	1.31×10^7	7.14×10^4	1.92×10^3	2.00×10^2
		Mean	8.91×10^3	4.71×10^3	8.82×10^3	3.11×10^8	1.38×10^7	8.01×10^4	1.88×10^3	2.00×10^2
		Std	1.03×10^3	1.21×10^3	6.54×10^2	1.03×10^8	5.63×10^6	2.63×10^4	4.86×10^2	0.00×10^0
		p-value	-	$9.92 \times 10^{-11-}$	$3.87 \times 10^{-1=}$	$3.02 \times 10^{-11+}$	$3.02 \times 10^{-11+}$	$3.90 \times 10^{-10+}$	$3.02 \times 10^{-11-}$	$1.21 \times 10^{-12-}$
F_{23-30}	*w/t/l*		-	3/3/2	3/2/3	8/0/0	6/1/1	8/0/0	5/0/3	0/0/8
	w/t/l		-	13/7/10	23/3/4	26/1/3	24/1/5	28/2/0	18/4/8	15/0/15
	Rank		2.70	3.37	4.58	7.07	5.27	5.83	4.03	3.15

From Table 2, the comparison results between ACSEDA and the compared state-of-the-art EDAs on the 30-D CEC 2014 benchmark problems can be summarized as follows:

(1) As shown in the last row of Table 2, in view of the Friedman test, it is found that the proposed ACSEDA obtains the smallest rank and this rank value is much smaller than those of the other algorithms. This indicates that ACSEDA achieves the best overall performance on the 30-D CEC 2014 benchmark set and obtains significant superiority to the compared algorithms.

(2) As shown in the second-to-last row of Table 2, from the perspective of the Wilcoxon rank-sum test, we can see that ACSEDA achieves significantly better performance than the compared algorithms on at least 23 problems, except for EDA2 and MA-ES. Compared with EDA2, ACSEDA shows significant superiority on 13 problems, and only presents inferiority on 6 problems. Competing with MA-ES, ACSEDA presents significant dominance on 19 problems, and it loses the competition on 10 problems.

(3) With respect to the optimization performance on different kinds of problems, on the three unimodal problems, both ACSEDA and EDA2 achieve the true global optima of these three problems and thus show significantly better performance than the other 6 EDA variants. On the 13 simple multimodal problems, ACSEDA is significantly superior to EDA$_{VESR}$, EDA/LS, EDA/LS-MS, and TRA-EDA on 10 problems, and it also beats EDA2, BUMDA, and MA-ES down on 8, 9, and 9 problems, respectively. In terms of the six hybrid problems, the optimization performance of ACSEDA is significantly better than the compared EDA variants on all these functions, except for EDA2. In comparison with EDA2, ACSEDA shows great superiority on three problems and achieves equivalent performance with EDA2 on three problems. In particular, on these six hybrid problems, ACSEDA shows no inferiority to all the compared EDA variants. As for the eight composition problems, it is observed that ACSEDA is significantly better than EDA/LS and EDA/LS-MS on all these problems. In comparison with TRA-EDA and BUMDA, it outperforms them on six and five problems, respectively. Particularly, on this kind of problem, ACSEDA is a litter inferior to EDA2 and MA-ES.

(4) To sum up, it is observed that ACSEDA shows very competitive, or even significantly better, performance in solving the 30-D CEC 2014 benchmark problems, as compared with the selected state-of-the-art EDA variants. In particular, encountered with complicated optimization problems, such as multimodal problems, hybrid problems, and composition problems, the proposed ACSEDA shows great superiority to the compared algorithms, which indicates that it is very promising for complicated problems.

Subsequently, from Table 3, we can get the following findings, with respect to the comparison results between ACSEDA and the compared state-of-the-art EDA variants, on the 50-D CEC 2014 benchmark problems:

(1) In terms of the Friedman test, as shown in the last row of Table 3, it is found that ACSEDA still achieves the lowest rank among all algorithms. This verifies that ACSEDA still obtains the best overall performance on the 50-D CEC 2014 problems.

(2) With respect to the Wilcoxon rank sum test, as shown in the second last row of Table 3, ACSEDA outperforms EDA$_{VERS}$, EDA/LS, EDA/LS-MS, TRA-EDA, and BUMDA significantly on 24, 28, 28, 28, and 21 problems, respectively. Compared with EDA2, ACSEDA attains much better performance on 13 problems and equivalent performance on 9 problems. Competing with MA-ES, ACSEDA significantly outperforms it on 19 problems and only shows inferiority on 11 problems.

(3) Regarding the performance on different kinds of optimization problems, on the three unimodal problems, except for EDA2 and MA-ES, ACSEDA still presents great superiority to the other compared EDA variants on all the three problems. In particular, both ACSEDA and EDA2 locate the true global optimum of F_3, while ACSEDA displays great dominance over EDA2 on the other two problems. Compared with MA-ES, ACSEDA is much better on two problems, and it obtains worse performance on only

one problem. On the 13 simple multimodal functions, ACSEDA significantly outperforms EDA_{VERS}, EDA/LS, EDA/LS-MS, and TRA-EDA on 11 problems, performs significantly better than MA-ES on 10 problems, and wins the competition on 7 problems, as competed with both EDA2 and BUMDA. On the 6 hybrid problems, ACSEDA is significantly superior to the compared EDA variants on all the six problems, except for EDA2. In competition with EDA2, ACSEDA loses the competition on five problems. On the 8 composition problems, ACSEDA is better than EDA/LS, EDA/LS-MS, TRA-EDA on all eight problems. At the same time, it achieves equivalent or even much better performance than EDA2, EDA_{VERS}, and BUMDA on at least six problems. However, ACSEDA shows inferior performance to MA-ES on seven problems.

(4) In summary, encountered with the 50-*D* CEC 2014 problems, ACSEDA still exhibits significantly better performance than the compared EDA variants. This further demonstrates that ACSEDA is promising for both simple optimization problems, such as unimodal problems, and complicated optimization problems, such as hybrid problems and composition problems.

At last, from Table 4, the following observations can be achieved from the comparison results between ACSEDA and the compared state-of-the-art EDA variants on the 100-*D* CEC 2014 benchmark problems:

(1) From the averaged rank obtained from the Friedman test, it is observed that ACSEDA still obtains the smallest rank value among all algorithms. This means that ACSEDA consistently achieves the best overall optimization performance on the 100-*D* CEC 2014 benchmark set.

(2) According to the results of the Wilcoxon rank sum test, ACSEDA presents great dominance to EDA_{VERS}, EDA/LS, EDA/LS-MS, and TRA-EDA on 23, 26, 24, and 28 problems, respectively. In comparison with EDA2 and BUMDA, ACSEDA obtains competitive or even better performance on 20 and 22 problems, respectively. Compared with MA-ES, ACSEDA achieves much better performance on 15 problems and presents inferiority on 15 problems as well. This indicates that ACSEDA is very competitive to MA-ES on the 100-*D* CEC2014 benchmark problems.

(3) Concerning the optimization performance on different kinds of optimization problems, on the three unimodal problems, ACSEDA outperforms EDA_{VERS}, EDA/LS-MS, TRA-EDA, and BUMDA on all these three problems, and it performs much better than EDA/LS on two problems. However, it loses the competition on these three problems to both EDA2 and MA-ES. When it comes to the 13 simple multimodal functions, ACSEDA shows significantly better performance than EDA_{VERS}, EDA/LS, EDA/LS-MS, TRA-EDA, and MA-ES on at least 10 problems, and presents great superiority to EDA2 on 8 problems. On these 13 problems, ACSEDA and BUMDA achieve very similar performance. Encountered with the six hybrid problems, ACSEDA exhibits much better performance than the compared EDA variants on at least five problems, except for EDA2. Faced with the eight composition problems, ACSEDA is better than EDA/LS, EDA/LS-MS, TRA-EDA, and BUMDA on at least five problems, presents very competitive performance with EDA2 and EDA_{VERS}, and is only inferior to MA-ES.

(4) To conclude, encountered with the 100-*D* CEC 2014 problems, ACSEDA still shows great superiority to the compared state-of-the-art EDA variants in solving such high-dimensional problems. In particular, on the complicated problems with such high dimensionality, such as the hybrid problems and the composition problems, ACSEDA still presents significant dominance to most of the compared algorithms. This further demonstrates that the proposed ACSEDA is promising for optimization problems.

Comprehensively speaking, from the above comparisons, we can see that ACSEDA consistently exhibits great superiority to the compared state-of-the-art EDA variants on the CEC 2014 benchmark problem set with different dimension sizes. This demonstrates that ACSEDA is promising for both simple unimodal problems and complicated multimodal problems. Besides, it also preserves good scalability in solving optimization problems.

The above demonstrated superiority of the proposed ACSEDA mainly benefits from the proposed three techniques. With the cohesive cooperation of them, ACSEDA could strike a promising balance between exploration and exploitation to search the complicated solution space properly.

4.3. Deep Investegation on ACSEDA

From the above comparison experiments, we can see that ACSEDA shows great dominance over the compared state-of-the-art EDA variants. In this section, we take a deep observation on ACSEDA to investigate the influence of each component, so it is clear to see what contributes to the promising performance of ACSEDA.

4.3.1. Effectiveness of the Covariance Scaling Strategy

First, we conduct experiments to verify the effectiveness of the proposed covariance scaling strategy, which is realized by setting a larger cs (to estimate the covariance) than the selection ratio sr (to estimate the mean vector). To this end, first, we fix different sr. Then, based on each fixed sr, we set different cs, each of which is larger than the associated sr. Subsequently, based on the above settings of sr and cs, we conduct experiments on the 50-D CEC 2014 benchmark problems. Table 5 shows the comparison results among ACSEDA with different settings of the selection ratio (sr) and different settings of the covariance scaling (cs) parameter on the 50-D CEC 2014 benchmark problems. In this table, the best results are highlighted in bold in each part associated with each fixed sr. In addition, the averaged ranks of each cs, in each part obtained from the Friedman test, are listed in the last row of the table.

From Table 5, we can get the following findings:

(1)   With respect to the comparison results of each part, it is found that a larger cs than sr helps ACSEDA achieve much better performance than the one with $cs = sr$. In particular, it is interesting to find that the superiority of the ACSEDA with a larger cs (than sr) to the one without the covariance scaling ($cs = sr$) is particularly significant in solving complicated problems such as F_{20}-F_{30}. This demonstrates the covariance scaling technique is helpful for ACSEDA to obtain promising performance in solving optimization problems, especially on complicated problems.

(2)   It is also interesting to find that neither a too small cs, nor a too large cs (compared with sr) are suitable for ACSEDA to achieve promising performance. For instance, when $sr = 0.1$, we find that, though ACSEDA with a too-small cs ($cs \leq 0.4$) achieves much better performance than the one with $cs = sr$ (namely without covariance scaling), its performance is much worse than the ones with a larger cs ($0.4 < cs < 0.8$). On the contrary, when ACSEDA has a too large cs ($cs \geq 0.8$), its performance degrades dramatically, as compared with the ones with a proper cs. A similar situation occurs to ACSEDA with the other settings of sr when cs is either too large or too small. Such experimental results verify the analysis presented in Section 3.1.

To sum up, it is found that the proposed covariance scaling strategy is effective to help EDA achieve promising performance in solving optimization problems, especially complicated problems.

Table 5. Comparison among ACSEDA, with different settings of the selection ratio (*sr*) and different settings of the covariance scaling (*cs*) parameter, on the 50-*D* CEC 2014 benchmark problems. The best results in each part are highlighted in bold in this table.

F	sr=0.1 cs=0.1 (cs/sr=1.0)	cs=0.2 (2.0)	cs=0.3 (3.0)	cs=0.4 (4.0)	cs=0.5 (5.0)	cs=0.6 (6.0)	cs=0.7 (7.0)	cs=0.8 (8.0)	cs=0.9 (9.0)	cs=1.0 (10.0)	sr=0.2 cs=0.2 (1.0)	cs=0.4 (2.0)	cs=0.6 (3.0)	cs=0.8 (4.0)	cs=1.0 (5.0)	sr=0.3 cs=0.3 (1.0)	cs=0.6 (2.0)	cs=0.9 (3.0)	cs=1.0 (3.3)	sr=0.4 cs=0.4 (1.0)	cs=0.8 (2.0)	cs=1.0 (2.5)	sr=0.5 cs=0.5 (1.0)	cs=1.0 (2.0)
F_1	2.32×10^{8}	2.33×10^{7}	1.44×10^{5}	3.43×10^{3}	**5.87×10^{-14}**	5.74×10^{-10}	4.28×10^{-6}	2.57×10^{-2}	5.70×10^{2}	1.03×10^{8}	4.00×10^{8}	3.98×10^{6}	4.67×10^{1}	**1.87×10^{-2}**	9.68×10^{7}	6.93×10^{8}	9.64×10^{4}	**2.52×10^{3}**	1.02×10^{8}	1.05×10^{9}	**3.46×10^{3}**	1.26×10^{8}	1.44×10^{9}	**1.47×10^{8}**
F_2	3.39×10^{10}	1.41×10^{10}	5.17×10^{8}	9.41×10^{1}	**9.35×10^{-12}**	1.08×10^{-7}	7.97×10^{-4}	5.14×10^{0}	4.17×10^{4}	2.97×10^{9}	3.94×10^{10}	7.22×10^{8}	**1.87×10^{-2}**	4.16×10^{0}	2.34×10^{9}	4.71×10^{10}	3.71×10^{8}	**2.33×10^{4}**	2.15×10^{9}	5.63×10^{10}	**9.39×10^{2}**	2.09×10^{9}	6.48×10^{10}	**2.01×10^{9}**
F_3	3.47×10^{4}	1.30×10^{4}	4.03×10^{2}	**0.00×10^{0}**	**0.00×10^{0}**	2.80×10^{-13}	2.45×10^{-9}	1.77×10^{-5}	1.75×10^{-1}	1.13×10^{5}	3.72×10^{4}	5.53×10^{3}	**1.71×10^{-13}**	1.02×10^{-5}	1.13×10^{5}	4.33×10^{4}	2.81×10^{2}	**8.26×10^{-2}**	1.17×10^{5}	4.81×10^{4}	**9.73×10^{-6}**	1.17×10^{5}	5.47×10^{4}	1.17×10^{5}
F_4	4.69×10^{3}	1.46×10^{3}	1.36×10^{2}	9.02×10^{1}	**8.96×10^{1}**	9.26×10^{1}	9.42×10^{1}	9.65×10^{1}	9.73×10^{1}	3.38×10^{2}	5.58×10^{3}	6.39×10^{2}	**9.31×10^{1}**	9.53×10^{1}	3.31×10^{2}	6.81×10^{3}	1.33×10^{2}	**9.55×10^{1}**	3.52×10^{2}	8.82×10^{3}	**9.32×10^{1}**	3.76×10^{2}	1.06×10^{4}	**4.20×10^{2}**
F_5	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**	**2.11×10^{1}**
F_6	1.91×10^{1}	8.04×10^{0}	1.36×10^{0}	1.65×10^{-1}	6.99×10^{-2}	**1.89×10^{-2}**	3.50×10^{-2}	3.66×10^{-1}	1.80×10^{0}	3.38×10^{1}	1.39×10^{1}	2.32×10^{0}	**1.21×10^{-1}**	1.46×10^{-1}	3.12×10^{1}	1.44×10^{1}	**1.06×10^{-1}**	1.51×10^{0}	3.06×10^{1}	1.58×10^{1}	**1.50×10^{-1}**	3.06×10^{1}	1.76×10^{1}	3.13×10^{1}
F_7	3.19×10^{2}	1.29×10^{2}	6.81×10^{0}	**0.00×10^{0}**	**0.00×10^{0}**	1.17×10^{-13}	1.03×10^{-9}	6.92×10^{-6}	6.71×10^{-2}	3.39×10^{1}	3.68×10^{2}	6.40×10^{1}	**1.14×10^{-13}**	4.14×10^{-6}	2.64×10^{1}	4.33×10^{2}	7.17×10^{0}	**3.47×10^{-2}**	2.46×10^{1}	5.12×10^{2}	**4.55×10^{-6}**	2.45×10^{1}	5.92×10^{2}	**2.44×10^{1}**
F_8	1.61×10^{2}	8.34×10^{1}	3.44×10^{1}	1.63×10^{1}	8.86×10^{0}	6.00×10^{0}	4.88×10^{0}	**3.39×10^{0}**	1.71×10^{1}	3.02×10^{2}	1.26×10^{2}	4.67×10^{1}	1.19×10^{1}	**4.02×10^{0}**	2.97×10^{2}	1.14×10^{2}	2.81×10^{1}	**1.09×10^{1}**	2.96×10^{2}	1.15×10^{2}	**1.56×10^{1}**	2.96×10^{2}	1.29×10^{2}	2.96×10^{2}
F_9	1.60×10^{2}	7.09×10^{1}	2.67×10^{1}	9.24×10^{0}	6.53×10^{0}	4.11×10^{0}	**3.68×10^{0}**	3.79×10^{0}	1.15×10^{2}	3.32×10^{2}	1.09×10^{2}	3.34×10^{1}	6.37×10^{0}	**2.70×10^{0}**	3.27×10^{2}	1.02×10^{2}	**1.93×10^{1}**	3.32×10^{1}	3.26×10^{2}	1.07×10^{2}	**9.42×10^{0}**	3.21×10^{2}	1.23×10^{2}	3.26×10^{2}
F_{10}	4.63×10^{3}	2.46×10^{3}	1.25×10^{3}	8.25×10^{2}	5.36×10^{2}	3.14×10^{2}	2.76×10^{2}	**1.30×10^{2}**	3.55×10^{2}	8.99×10^{3}	3.26×10^{3}	1.37×10^{3}	7.34×10^{2}	**2.96×10^{2}**	8.83×10^{3}	2.79×10^{3}	1.05×10^{3}	**9.53×10^{2}**	9.21×10^{3}	2.70×10^{3}	**1.23×10^{3}**	9.28×10^{3}	2.86×10^{3}	1.01×10^{4}
F_{11}	4.85×10^{3}	2.46×10^{3}	1.31×10^{3}	8.47×10^{2}	6.18×10^{2}	4.83×10^{2}	3.77×10^{2}	**3.00×10^{2}**	3.22×10^{3}	1.05×10^{4}	3.24×10^{3}	1.33×10^{3}	7.81×10^{2}	**5.04×10^{2}**	1.05×10^{4}	2.87×10^{3}	**1.26×10^{3}**	3.65×10^{3}	1.05×10^{4}	2.73×10^{3}	**1.23×10^{3}**	1.08×10^{4}	2.66×10^{3}	1.10×10^{4}
F_{12}	**1.14×10^{0}**	3.17×10^{0}	3.21×10^{0}	3.20×10^{0}	3.15×10^{0}	3.28×10^{0}	3.23×10^{0}	3.18×10^{0}	3.23×10^{0}	3.42×10^{0}	3.21×10^{0}	3.28×10^{0}	**3.20×10^{0}**	3.30×10^{0}	3.20×10^{0}	3.27×10^{0}	3.28×10^{0}	**3.22×10^{0}**	3.35×10^{0}	3.26×10^{0}	3.28×10^{0}	**3.25×10^{0}**	**3.25×10^{0}**	3.33×10^{0}
F_{13}	3.81×10^{0}	1.53×10^{0}	1.78×10^{-1}	1.43×10^{-1}	1.40×10^{-1}	1.30×10^{-1}	1.08×10^{-1}	8.13×10^{-2}	**6.16×10^{-2}**	4.44×10^{-1}	4.01×10^{0}	3.50×10^{-1}	1.33×10^{-1}	**9.14×10^{-2}**	3.78×10^{-1}	4.38×10^{0}	1.63×10^{-1}	**6.22×10^{-2}**	4.16×10^{-1}	4.74×10^{0}	**1.22×10^{-1}**	4.24×10^{-1}	5.13×10^{0}	**4.27×10^{-1}**
F_{14}	6.72×10^{1}	2.05×10^{1}	2.85×10^{-1}	3.37×10^{-1}	3.30×10^{-1}	3.15×10^{-1}	2.93×10^{-1}	**2.61×10^{-1}**	2.69×10^{-1}	8.46×10^{0}	8.23×10^{1}	8.36×10^{1}	3.20×10^{-1}	**2.98×10^{-1}**	5.38×10^{0}	9.72×10^{1}	**3.02×10^{-1}**	3.06×10^{-1}	3.74×10^{0}	1.14×10^{2}	**3.26×10^{-1}**	4.16×10^{0}	1.37×10^{2}	**3.64×10^{0}**
F_{15}	5.07×10^{3}	5.31×10^{1}	5.65×10^{0}	**4.75×10^{0}**	4.94×10^{0}	4.77×10^{0}	7.47×10^{0}	1.38×10^{1}	2.54×10^{1}	8.19×10^{1}	7.77×10^{3}	6.69×10^{0}	**5.02×10^{0}**	1.30×10^{1}	6.93×10^{1}	1.20×10^{4}	**5.41×10^{0}**	2.34×10^{1}	6.93×10^{1}	2.26×10^{4}	**1.27×10^{1}**	7.29×10^{1}	4.24×10^{4}	**8.86×10^{1}**
F_{16}	1.72×10^{1}	**1.58×10^{1}**	1.84×10^{1}	1.91×10^{1}	2.03×10^{1}	2.02×10^{1}	2.02×10^{1}	2.01×10^{1}	2.02×10^{1}	2.03×10^{1}	**1.74×10^{1}**	1.97×10^{1}	2.07×10^{1}	2.08×10^{1}	2.05×10^{1}	**2.00×10^{1}**	2.13×10^{1}	2.12×10^{1}	2.10×10^{1}	**2.12×10^{1}**	2.20×10^{1}	2.16×10^{1}	2.24×10^{1}	**2.24×10^{1}**
F_{17}	1.18×10^{7}	6.54×10^{2}	3.78×10^{2}	2.87×10^{2}	2.51×10^{2}	2.42×10^{2}	**2.30×10^{2}**	3.50×10^{2}	7.20×10^{2}	5.30×10^{6}	2.58×10^{7}	3.90×10^{2}	**3.20×10^{2}**	4.51×10^{2}	6.01×10^{6}	4.60×10^{7}	**4.82×10^{2}**	9.18×10^{2}	6.59×10^{6}	8.43×10^{7}	**6.54×10^{2}**	7.59×10^{6}	1.23×10^{8}	**9.22×10^{6}**
F_{18}	2.44×10^{8}	1.30×10^{2}	9.18×10^{1}	4.83×10^{1}	2.76×10^{1}	2.44×10^{1}	**2.42×10^{1}**	2.63×10^{1}	1.63×10^{2}	1.72×10^{8}	8.31×10^{8}	8.80×10^{1}	**3.95×10^{1}**	4.11×10^{1}	1.40×10^{8}	1.77×10^{9}	**7.36×10^{1}**	1.66×10^{2}	1.32×10^{8}	3.13×10^{9}	**7.62×10^{1}**	1.27×10^{8}	4.56×10^{9}	**1.22×10^{8}**
F_{19}	8.07×10^{1}	3.42×10^{1}	2.69×10^{1}	1.45×10^{1}	1.23×10^{1}	**1.19×10^{1}**	1.22×10^{1}	1.23×10^{1}	1.31×10^{1}	8.38×10^{1}	9.25×10^{1}	3.61×10^{1}	1.44×10^{1}	**1.23×10^{1}**	8.69×10^{1}	1.34×10^{2}	3.42×10^{1}	**1.30×10^{1}**	8.90×10^{1}	2.11×10^{2}	**1.62×10^{1}**	9.02×10^{1}	3.22×10^{2}	**9.37×10^{1}**
F_{20}	7.88×10^{2}	9.08×10^{1}	3.64×10^{1}	9.82×10^{0}	5.40×10^{0}	3.19×10^{0}	**2.71×10^{0}**	4.25×10^{0}	3.90×10^{1}	2.17×10^{4}	1.21×10^{3}	4.07×10^{1}	7.49×10^{0}	**5.36×10^{0}**	2.97×10^{4}	1.69×10^{3}	**2.34×10^{1}**	3.38×10^{1}	3.55×10^{4}	2.51×10^{3}	**2.10×10^{1}**	3.88×10^{4}	**3.84×10^{3}**	3.82×10^{4}
F_{21}	1.22×10^{4}	9.05×10^{2}	4.28×10^{2}	3.15×10^{2}	**2.61×10^{2}**	2.62×10^{2}	2.84×10^{2}	3.24×10^{2}	5.40×10^{2}	3.39×10^{6}	1.45×10^{4}	4.69×10^{2}	**3.00×10^{2}**	3.99×10^{2}	4.06×10^{6}	2.85×10^{4}	**4.04×10^{2}**	7.46×10^{2}	4.15×10^{6}	5.22×10^{4}	**5.38×10^{2}**	4.96×10^{6}	1.19×10^{5}	5.55×10^{6}
F_{22}	1.48×10^{2}	6.22×10^{1}	5.03×10^{1}	**3.77×10^{1}**	3.87×10^{1}	5.43×10^{1}	5.87×10^{1}	6.81×10^{1}	8.39×10^{1}	8.07×10^{2}	1.43×10^{2}	7.32×10^{1}	**6.23×10^{1}**	7.91×10^{1}	8.02×10^{1}	3.79×10^{2}	**4.38×10^{1}**	1.20×10^{2}	8.64×10^{1}	1.12×10^{3}	**8.50×10^{1}**	8.52×10^{1}	3.10×10^{3}	8.66×10^{2}
F_{23}	4.62×10^{2}	3.93×10^{2}	3.52×10^{2}	**3.44×10^{2}**	**3.44×10^{2}**	**3.44×10^{2}**	**3.44×10^{2}**	**3.44×10^{2}**	**3.44×10^{2}**	3.94×10^{2}	4.53×10^{2}	3.75×10^{2}	**3.44×10^{2}**	**3.44×10^{2}**	4.02×10^{2}	4.62×10^{2}	3.52×10^{2}	**3.44×10^{2}**	4.14×10^{2}	4.94×10^{2}	**3.44×10^{2}**	4.24×10^{2}	5.24×10^{2}	4.44×10^{2}
F_{24}	2.80×10^{2}	2.74×10^{2}	2.73×10^{2}	2.72×10^{2}	2.72×10^{2}	2.70×10^{2}	2.70×10^{2}	2.69×10^{2}	2.70×10^{2}	3.27×10^{2}	2.76×10^{2}	2.73×10^{2}	2.73×10^{2}	**2.71×10^{2}**	3.22×10^{2}	2.75×10^{2}	**2.73×10^{2}**	2.73×10^{2}	3.21×10^{2}	**2.74×10^{2}**	**2.74×10^{2}**	3.21×10^{2}	2.72×10^{2}	3.22×10^{2}
F_{25}	2.22×10^{2}	2.15×10^{2}	2.09×10^{2}	**2.05×10^{2}**	**2.05×10^{2}**	**2.05×10^{2}**	**2.05×10^{2}**	**2.05×10^{2}**	**2.05×10^{2}**	2.36×10^{2}	2.20×10^{2}	2.13×10^{2}	**2.05×10^{2}**	**2.05×10^{2}**	2.47×10^{2}	2.21×10^{2}	2.09×10^{2}	**2.05×10^{2}**	2.51×10^{2}	2.20×10^{2}	**2.05×10^{2}**	2.51×10^{2}	2.19×10^{2}	2.51×10^{2}
F_{26}	1.27×10^{2}	1.11×10^{2}	1.04×10^{2}	1.01×10^{2}	1.03×10^{2}	1.00×10^{2}	**1.00×10^{2}**	**1.00×10^{2}**	**1.00×10^{2}**	1.62×10^{2}	1.33×10^{2}	1.17×10^{2}	**1.05×10^{2}**	**1.05×10^{2}**	1.85×10^{2}	1.38×10^{2}	1.19×10^{2}	**1.06×10^{2}**	1.91×10^{2}	1.64×10^{2}	**1.08×10^{2}**	1.99×10^{2}	**1.71×10^{2}**	1.80×10^{2}
F_{27}	1.05×10^{3}	6.94×10^{2}	4.08×10^{2}	3.37×10^{2}	**3.29×10^{2}**	3.36×10^{2}	3.30×10^{2}	3.34×10^{2}	3.81×10^{2}	1.08×10^{3}	9.37×10^{2}	5.39×10^{2}	**3.22×10^{2}**	3.29×10^{2}	1.03×10^{3}	9.48×10^{2}	3.93×10^{2}	**3.62×10^{2}**	1.02×10^{3}	1.02×10^{3}	**3.24×10^{2}**	1.02×10^{3}	1.10×10^{3}	**1.04×10^{3}**
F_{28}	2.40×10^{3}	1.73×10^{3}	1.51×10^{3}	**1.16×10^{3}**	1.17×10^{3}	1.18×10^{3}	1.21×10^{3}	1.23×10^{3}	1.28×10^{3}	1.47×10^{3}	2.34×10^{3}	1.47×10^{3}	**1.16×10^{3}**	1.25×10^{3}	1.51×10^{3}	2.43×10^{3}	1.48×10^{3}	**1.30×10^{3}**	1.56×10^{3}	2.15×10^{3}	**1.23×10^{3}**	1.56×10^{3}	2.33×10^{3}	**1.56×10^{3}**
F_{29}	1.24×10^{4}	2.66×10^{3}	1.97×10^{3}	1.20×10^{3}	**8.88×10^{2}**	9.91×10^{2}	1.29×10^{3}	1.71×10^{3}	4.11×10^{3}	1.07×10^{6}	1.16×10^{3}	1.64×10^{3}	**9.69×10^{2}**	1.62×10^{3}	1.70×10^{6}	3.79×10^{4}	**1.59×10^{3}**	4.49×10^{3}	5.00×10^{6}	4.45×10^{4}	**1.24×10^{3}**	4.60×10^{6}	**1.15×10^{3}**	7.17×10^{6}
F_{30}	1.68×10^{4}	3.69×10^{4}	1.66×10^{4}	1.06×10^{4}	9.03×10^{3}	**8.77×10^{3}**	9.08×10^{3}	9.47×10^{3}	9.29×10^{3}	3.70×10^{4}	2.36×10^{5}	2.54×10^{4}	9.83×10^{3}	**8.95×10^{3}**	5.48×10^{4}	3.11×10^{5}	1.73×10^{4}	**9.12×10^{3}**	5.35×10^{4}	5.07×10^{5}	**9.99×10^{3}**	6.20×10^{4}	6.99×10^{5}	**7.81×10^{4}**
Rank	8.83	7.43	6.03	4.53	3.25	3.25	**3.22**	3.65	5.73	9.07	4.20	3.00	**1.62**	1.98	4.20	3.30	1.70	**1.60**	3.40	2.43	**1.20**	2.37	1.57	**1.43**

4.3.2. Effectiveness of the Proposed Adaptive Covariance Scaling Strategy

Then, we conduct experiments to verify the effectiveness of the proposed adaptive covariance scaling method, which is realized by dynamically adjusting cs according to Equation (10). Since the proposed adaptive covariance scaling method is related to sr, as can be seen from Equation (10), we first fix sr as 0.1 and 0.2 to investigate the effectiveness of the proposed adaptive method. These two settings of sr are utilized because, from the experimental result in the last subsection as shown in Table 5, when sr is larger than 0.3, ACSEDA achieves much worse performance and thus, it is meaningless to investigate the effectiveness of the proposed adaptive cs when sr is larger than 0.3. Subsequently, for each set of sr, we set different fixed cs for ACSEDA and then compare them with the ACSEDA with the adaptive cs strategy.

Table 6 presents the comparison between ACSEDA with the adaptive covariance scaling method and the ones with different fixed settings of cs on the 50-D CEC 2014 benchmark problems. From this table, we attain the following observations:

(1)   As a whole, no matter if it is from the perspective of the averaged rank obtained from the Friedman test or from the perspective of the number of problems where the algorithm achieves the best results, the ACSEDA with the proposed adaptive cs obtains much better performance than those with different fixed cs. This verifies that the proposed adaptive cs strategy is effective to help ACSEDA achieve promising performance.

(2)   Under the same set of sr, we find that the optimal fixed cs for ACSEDA to achieve the best performance is different on different problems. This indicates that the optimal setting of cs is not consistent for all problems. With the proposed adaptive strategy, we can see that not only is the sensitivity of ACSEDA to cs alleviated, but its optimization performance is also largely promoted.

In summary, based on the above experiments, we can see that the proposed covariance scaling strategy is very beneficial for ACSEDA to achieve promising performance. This is mainly because the proposed adaptive strategy helps ACSEDA bias to explore the solution space in the early stage and, gradually, bias to exploit the found promising areas without serious loss of diversity as the evolution iterates. As a result, with this adaptive strategy, ACSEDA could explore and exploit the solution space properly to find the optima of optimization problems.

4.3.3. Effectiveness of the Proposed Adaptive Promising Selection Strategy

Subsequently, we conduct experiments to verify the effectiveness of the proposed promising individual selection strategy, which is realized by dynamically adjusting the parameter sr, based on Equation (11). Since the setting of sr influences the covariance scaling parameter cs, we first fix cs as 0.6 and then accordingly set different fixed sr. It should be noticed that $cs = 0.6$ is adopted here because, in the last subsection, as shown in Table 6, ACSEDA obtains the best overall performance when $cs = 0.6$ under the two settings of sr. Then, we compare the ACSEDA with the proposed adaptive sr and the ones with different fixed sr under the same set of cs (namely $cs = 0.6$).

Table 6. Comparison between ACSEDA, with and without the adaptive covariance scaling method, on the 50-*D* CEC 2014 benchmark problems. The best results in each part are highlighted in bold in this table.

F	sr = 0.1											sr = 0.2					
	cs = 0.1	cs = 0.2	cs = 0.3	cs = 0.4	cs = 0.5	cs = 0.6	cs = 0.7	cs = 0.8	cs = 0.9	cs = 1.0	Adaptive − cs	cs = 0.2	cs = 0.4	cs = 0.6	cs = 0.8	cs = 1.0	Adaptive − cs
F_1	2.32×10^8	2.33×10^7	1.44×10^5	3.43×10^3	5.87×10^{-14}	5.74×10^{-10}	4.28×10^{-6}	2.57×10^{-2}	5.70×10^2	1.03×10^8	$\mathbf{1.47 \times 10^{-14}}$	4.00×10^8	3.98×10^6	4.67×10^1	1.87×10^{-2}	9.68×10^7	$\mathbf{2.53 \times 10^{-13}}$
F_2	3.39×10^8	1.41×10^{10}	5.17×10^8	9.41×10^1	9.35×10^{-12}	1.08×10^{-7}	7.97×10^{-4}	5.14×10^0	4.17×10^4	2.97×10^9	$\mathbf{2.04 \times 10^{-12}}$	3.94×10^{10}	7.22×10^9	1.87×10^{-2}	4.16×10^0	2.34×10^9	$\mathbf{3.83 \times 10^{-11}}$
F_3	3.47×10^4	1.30×10^4	4.03×10^2	$\mathbf{0.00 \times 10^0}$	$\mathbf{0.00 \times 10^0}$	2.80×10^{-13}	2.45×10^{-9}	1.77×10^{-5}	1.75×10^{-1}	1.13×10^5	$\mathbf{0.00 \times 10^0}$	3.72×10^4	5.53×10^3	1.71×10^{-14}	1.02×10^{-5}	1.13×10^5	$\mathbf{0.00 \times 10^0}$
F_4	4.69×10^3	1.46×10^3	1.36×10^2	9.02×10^1	$\mathbf{8.96 \times 10^1}$	9.26×10^1	9.42×10^1	9.65×10^1	9.73×10^1	3.38×10^2	9.31×10^1	5.58×10^3	6.39×10^2	$\mathbf{9.31 \times 10^1}$	9.53×10^1	3.31×10^2	9.35×10^1
F_5	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$
F_6	1.91×10^1	8.04×10^0	1.36×10^0	1.65×10^{-1}	6.99×10^{-2}	1.89×10^{-2}	3.50×10^{-2}	3.66×10^{-1}	1.80×10^0	3.38×10^1	$\mathbf{1.74 \times 10^{-2}}$	1.39×10^1	2.32×10^0	1.21×10^{-1}	1.46×10^{-1}	3.12×10^1	$\mathbf{2.20 \times 10^{-4}}$
F_7	3.19×10^2	1.29×10^2	6.81×10^0	$\mathbf{0.00 \times 10^0}$	$\mathbf{0.00 \times 10^0}$	1.17×10^{-13}	1.03×10^{-9}	6.92×10^{-6}	6.71×10^{-2}	3.39×10^1	$\mathbf{0.00 \times 10^0}$	3.68×10^2	6.40×10^1	1.14×10^{-13}	4.14×10^{-6}	2.64×10^1	$\mathbf{0.00 \times 10^0}$
F_8	1.61×10^2	8.34×10^1	3.44×10^1	1.63×10^1	8.86×10^0	6.00×10^0	4.88×10^0	3.39×10^0	1.71×10^1	3.02×10^2	$\mathbf{3.05 \times 10^0}$	1.26×10^2	4.67×10^1	1.19×10^1	4.02×10^0	2.97×10^2	$\mathbf{2.35 \times 10^0}$
F_9	1.60×10^2	7.09×10^1	2.67×10^1	9.24×10^0	6.53×10^0	4.11×10^0	3.68×10^0	3.79×10^0	1.15×10^2	3.32×10^2	$\mathbf{2.06 \times 10^0}$	1.09×10^2	3.34×10^1	6.37×10^0	2.70×10^0	3.27×10^2	$\mathbf{1.86 \times 10^0}$
F_{10}	4.63×10^3	2.46×10^3	1.25×10^3	8.25×10^2	5.36×10^2	3.14×10^2	2.76×10^2	1.30×10^2	3.55×10^2	8.99×10^3	$\mathbf{7.00 \times 10^1}$	3.26×10^3	1.37×10^3	7.34×10^2	2.96×10^2	8.83×10^3	$\mathbf{1.12 \times 10^2}$
F_{11}	4.85×10^3	2.46×10^3	1.31×10^3	8.47×10^2	6.18×10^2	4.83×10^2	3.77×10^2	3.00×10^2	3.22×10^3	1.05×10^4	$\mathbf{1.22 \times 10^2}$	3.24×10^3	1.33×10^3	7.81×10^2	5.04×10^2	1.05×10^4	$\mathbf{1.72 \times 10^2}$
F_{12}	$\mathbf{1.14 \times 10^0}$	3.17×10^0	3.21×10^0	3.20×10^0	3.15×10^0	3.28×10^0	3.23×10^0	3.18×10^0	3.23×10^0	3.42×10^0	3.27×10^0	3.21×10^0	3.28×10^0	3.20×10^0	3.30×10^0	3.20×10^0	$\mathbf{3.16 \times 10^0}$
F_{13}	3.81×10^0	1.53×10^0	1.78×10^{-1}	1.43×10^{-1}	1.40×10^{-1}	1.30×10^{-1}	1.08×10^{-1}	8.13×10^{-2}	$\mathbf{6.16 \times 10^{-2}}$	4.44×10^{-1}	1.47×10^{-1}	4.01×10^0	3.50×10^{-1}	1.33×10^{-1}	$\mathbf{9.14 \times 10^{-2}}$	3.78×10^{-1}	1.67×10^{-1}
F_{14}	6.72×10^1	2.05×10^1	2.85×10^{-1}	3.37×10^{-1}	3.30×10^{-1}	3.15×10^{-1}	2.93×10^{-1}	$\mathbf{2.61 \times 10^{-1}}$	2.69×10^{-1}	8.46×10^0	2.81×10^{-1}	8.23×10^1	8.36×10^{-1}	3.20×10^{-1}	2.98×10^{-1}	5.38×10^0	$\mathbf{2.96 \times 10^{-1}}$
F_{15}	5.07×10^1	5.31×10^1	5.65×10^0	$\mathbf{4.75 \times 10^0}$	4.94×10^0	4.77×10^0	7.47×10^0	1.38×10^1	2.54×10^1	8.19×10^1	4.78×10^0	7.77×10^3	6.69×10^0	$\mathbf{5.02 \times 10^0}$	1.30×10^1	6.93×10^1	5.04×10^0
F_{16}	1.72×10^1	$\mathbf{1.58 \times 10^1}$	1.84×10^1	1.91×10^1	2.03×10^1	2.02×10^1	2.02×10^1	2.01×10^1	2.02×10^1	2.03×10^1	1.80×10^1	$\mathbf{1.74 \times 10^1}$	1.97×10^1	2.07×10^1	2.08×10^1	2.05×10^1	1.86×10^1
F_{17}	1.18×10^7	6.54×10^2	3.78×10^2	2.87×10^2	2.51×10^2	2.42×10^2	2.30×10^2	3.50×10^2	7.20×10^2	5.30×10^6	$\mathbf{1.28 \times 10^2}$	2.58×10^7	3.90×10^2	3.20×10^2	4.51×10^2	6.01×10^6	$\mathbf{1.40 \times 10^2}$
F_{18}	2.44×10^8	1.30×10^2	9.18×10^1	4.83×10^1	2.76×10^1	2.44×10^1	2.42×10^1	2.63×10^1	1.63×10^2	1.72×10^8	$\mathbf{7.69 \times 10^0}$	8.31×10^8	8.80×10^1	3.95×10^1	4.11×10^1	1.40×10^8	$\mathbf{1.40 \times 10^1}$
F_{19}	8.07×10^1	3.42×10^1	2.69×10^1	1.45×10^1	1.23×10^1	1.19×10^1	1.22×10^1	1.23×10^1	1.31×10^1	8.38×10^1	$\mathbf{1.11 \times 10^1}$	9.25×10^1	3.61×10^1	1.44×10^1	1.23×10^1	8.69×10^1	$\mathbf{1.12 \times 10^1}$
F_{20}	7.88×10^2	9.08×10^1	3.64×10^1	9.82×10^0	5.40×10^0	3.19×10^0	2.71×10^0	4.25×10^0	3.90×10^0	2.17×10^4	$\mathbf{2.04 \times 10^0}$	1.21×10^2	4.07×10^1	7.49×10^0	5.36×10^0	2.97×10^4	$\mathbf{2.17 \times 10^0}$
F_{21}	1.22×10^4	9.05×10^2	4.28×10^2	3.15×10^2	2.61×10^2	2.62×10^2	2.84×10^2	3.24×10^2	5.40×10^2	3.39×10^6	$\mathbf{2.12 \times 10^2}$	1.45×10^4	4.69×10^2	3.00×10^2	3.99×10^2	4.06×10^6	$\mathbf{2.13 \times 10^2}$
F_{22}	1.48×10^2	6.22×10^1	5.03×10^1	3.77×10^1	3.87×10^1	5.43×10^1	5.87×10^1	6.81×10^1	8.39×10^1	8.07×10^2	$\mathbf{3.70 \times 10^1}$	1.43×10^2	7.32×10^1	6.23×10^1	7.91×10^1	8.02×10^2	$\mathbf{3.70 \times 10^1}$
F_{23}	4.62×10^2	3.93×10^2	3.52×10^2	$\mathbf{3.44 \times 10^2}$	$\mathbf{3.44 \times 10^2}$	$\mathbf{3.44 \times 10^2}$	$\mathbf{3.44 \times 10^2}$	$\mathbf{3.44 \times 10^2}$	$\mathbf{3.44 \times 10^2}$	3.94×10^2	$\mathbf{3.44 \times 10^2}$	4.53×10^2	3.75×10^2	$\mathbf{3.44 \times 10^2}$	$\mathbf{3.44 \times 10^2}$	3.22×10^2	$\mathbf{3.44 \times 10^2}$
F_{24}	2.80×10^2	2.74×10^2	2.73×10^2	2.72×10^2	2.72×10^2	2.70×10^2	2.70×10^2	2.69×10^2	2.70×10^2	3.27×10^2	$\mathbf{2.68 \times 10^2}$	2.76×10^2	2.73×10^2	2.73×10^2	2.71×10^2	3.22×10^2	$\mathbf{2.69 \times 10^2}$
F_{25}	2.22×10^2	2.15×10^2	2.09×10^2	$\mathbf{2.05 \times 10^2}$	$\mathbf{2.05 \times 10^2}$	$\mathbf{2.05 \times 10^2}$	$\mathbf{2.05 \times 10^2}$	$\mathbf{2.05 \times 10^2}$	$\mathbf{2.05 \times 10^2}$	2.36×10^2	$\mathbf{2.05 \times 10^2}$	2.20×10^2	2.13×10^2	$\mathbf{2.05 \times 10^2}$	$\mathbf{2.05 \times 10^2}$	2.47×10^2	$\mathbf{2.05 \times 10^2}$
F_{26}	1.27×10^2	1.11×10^2	1.04×10^2	1.01×10^2	1.03×10^2	$\mathbf{1.00 \times 10^2}$	$\mathbf{1.00 \times 10^2}$	$\mathbf{1.00 \times 10^2}$	$\mathbf{1.00 \times 10^2}$	1.62×10^2	$\mathbf{1.00 \times 10^2}$	1.33×10^2	1.17×10^2	1.05×10^2	1.05×10^2	1.85×10^2	$\mathbf{1.00 \times 10^2}$
F_{27}	1.05×10^3	6.94×10^2	4.08×10^2	3.37×10^2	$\mathbf{3.29 \times 10^2}$	3.36×10^2	3.30×10^2	3.34×10^2	3.81×10^2	1.08×10^3	3.31×10^2	9.37×10^2	5.39×10^2	3.22×10^2	3.29×10^2	1.03×10^3	$\mathbf{3.21 \times 10^2}$
F_{28}	2.40×10^3	1.73×10^3	1.51×10^3	$\mathbf{1.16 \times 10^3}$	1.17×10^3	1.18×10^3	1.21×10^3	1.23×10^3	1.28×10^3	1.47×10^3	1.18×10^3	2.34×10^3	1.47×10^3	1.16×10^3	1.25×10^3	1.51×10^3	$\mathbf{1.14 \times 10^3}$
F_{29}	1.24×10^4	2.66×10^3	1.97×10^3	1.20×10^3	8.88×10^2	9.91×10^2	1.29×10^3	1.71×10^3	4.11×10^3	1.07×10^6	$\mathbf{8.77 \times 10^2}$	1.16×10^4	1.64×10^3	9.69×10^2	1.62×10^3	1.70×10^6	$\mathbf{9.02 \times 10^2}$
F_{30}	1.68×10^5	3.69×10^4	1.66×10^4	1.06×10^4	9.03×10^3	8.77×10^3	9.08×10^3	9.47×10^3	9.29×10^3	3.70×10^4	$\mathbf{8.60 \times 10^3}$	2.36×10^5	2.54×10^4	9.83×10^3	8.95×10^3	5.48×10^4	$\mathbf{8.51 \times 10^3}$
Rank	9.70	8.40	7.13	4.98	4.05	3.95	4.08	4.55	6.55	10.18	**2.42**	5.22	3.97	2.47	2.83	5.20	**1.32**

Table 7 shows the comparison results between the ACSEDA with the proposed adaptive *sr* and the ones with different fixed settings of *sr* on the 50-*D* CEC 2014 benchmark problems. From this table, the following findings can be attained:

(1) In view of the averaged rank obtained from the Friedman test, the ACSEDA with the adaptive *sr* achieves the best overall performance than the ones with different fixed *sr*. This verifies the effectiveness of the proposed adaptive promising individual selection strategy for the estimation of the mean vector.

(2) For different problems, the optimal *sr* is different for ACSEDA to achieve the best performance. In particular, we find that a small *sr* tends to help ACSEDA obtain better performance than a large *sr*. The proposed adaptive strategy, based on Equation (11), matches this observation that *sr* is dramatically decreased to a small value in the early stage, and then, it mildly declines as the evolution goes as stated in Section 3.2.

Table 7. Comparison between ACSEDA, with and without the adaptive promising individual selection method, on the 50-*D* CEC 2014 benchmark problems. The best results are highlighted in bold in this table.

				$cs = 0.6$				
F	$sr = 0.05$	$sr = 0.10$	$sr = 0.15$	$sr = 0.20$	$sr = 0.25$	$sr = 0.30$	$sr = 0.35$	**Adaptive** $- sr$
F_1	1.61×10^{-14}	1.42×10^{-14}	$\mathbf{1.28 \times 10^{-14}}$	7.55×10^1	2.52×10^4	4.67×10^5	1.75×10^7	1.42×10^{-14}
F_2	2.91×10^{-12}	1.08×10^{-12}	$\mathbf{9.56 \times 10^{-13}}$	2.02×10^0	1.65×10^7	2.91×10^9	1.20×10^{10}	2.22×10^{-12}
F_3	$\mathbf{0.00 \times 10^0}$	$\mathbf{0.00 \times 10^0}$	$\mathbf{0.00 \times 10^0}$	$\mathbf{0.00 \times 10^0}$	2.28×10^0	1.93×10^3	8.44×10^3	$\mathbf{0.00 \times 10^0}$
F_4	9.07×10^1	8.88×10^1	9.14×10^1	9.26×10^1	1.04×10^2	3.45×10^2	1.14×10^3	$\mathbf{8.67 \times 10^1}$
F_5	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$	$\mathbf{2.11 \times 10^1}$
F_6	1.23×10^{-4}	7.35×10^{-5}	5.20×10^{-2}	$\mathbf{6.25 \times 10^{-5}}$	1.74×10^{-2}	1.64×10^{-2}	2.24×10^{-1}	9.75×10^{-5}
F_7	$\mathbf{0.00 \times 10^0}$	$\mathbf{0.00 \times 10^0}$	$\mathbf{0.00 \times 10^0}$	$\mathbf{0.00 \times 10^0}$	6.22×10^{-1}	3.07×10^1	1.13×10^2	$\mathbf{0.00 \times 10^0}$
F_8	7.36×10^0	$\mathbf{6.80 \times 10^0}$	8.66×10^0	1.43×10^1	1.95×10^1	2.63×10^1	3.48×10^1	9.78×10^0
F_9	6.20×10^0	$\mathbf{4.91 \times 10^0}$	6.10×10^0	7.16×10^0	1.20×10^1	1.71×10^1	2.40×10^1	6.70×10^0
F_{10}	$\mathbf{2.52 \times 10^2}$	3.05×10^2	4.49×10^2	5.11×10^2	7.11×10^2	8.08×10^2	7.93×10^2	3.22×10^2
F_{11}	$\mathbf{2.30 \times 10^2}$	3.36×10^2	4.32×10^2	4.96×10^2	5.80×10^2	7.30×10^2	6.84×10^2	3.57×10^2
F_{12}	3.27×10^0	3.27×10^0	3.21×10^0	3.20×10^0	$\mathbf{3.20 \times 10^0}$	3.30×10^0	3.31×10^0	3.25×10^0
F_{13}	$\mathbf{1.39 \times 10^{-1}}$	1.55×10^{-1}	1.60×10^{-1}	1.57×10^{-1}	1.53×10^{-1}	2.45×10^{-1}	8.50×10^{-1}	1.51×10^{-1}
F_{14}	2.85×10^{-1}	2.95×10^{-1}	3.17×10^{-1}	3.11×10^{-1}	3.16×10^{-1}	$\mathbf{2.77 \times 10^{-1}}$	1.25×10^1	3.09×10^{-1}
F_{15}	4.82×10^0	4.78×10^0	$\mathbf{4.77 \times 10^0}$	5.04×10^0	5.07×10^0	5.10×10^0	8.61×10^0	4.77×10^0
F_{16}	1.86×10^1	1.85×10^1	$\mathbf{1.84 \times 10^1}$	1.87×10^1	1.90×10^1	1.91×10^1	1.93×10^1	1.87×10^1
F_{17}	1.80×10^2	1.75×10^2	1.62×10^2	1.97×10^2	1.70×10^2	2.37×10^2	2.28×10^2	$\mathbf{1.44 \times 10^2}$
F_{18}	$\mathbf{1.35 \times 10^1}$	1.85×10^1	2.08×10^1	2.98×10^1	3.97×10^1	5.17×10^1	5.58×10^1	1.61×10^1
F_{19}	1.17×10^1	$\mathbf{1.15 \times 10^1}$	1.22×10^1	1.33×10^1	2.17×10^1	2.81×10^1	2.23×10^1	1.17×10^1
F_{20}	$\mathbf{2.58 \times 10^0}$	2.71×10^0	3.76×10^0	6.29×10^0	9.09×10^0	1.50×10^1	2.34×10^1	3.14×10^0
F_{21}	2.29×10^2	2.11×10^2	2.28×10^2	2.47×10^2	2.72×10^2	3.10×10^2	3.15×10^2	$\mathbf{1.98 \times 10^2}$
F_{22}	4.16×10^1	$\mathbf{3.13 \times 10^1}$	4.45×10^1	3.27×10^1	4.57×10^1	5.78×10^1	5.63×10^1	3.18×10^1
F_{23}	$\mathbf{3.44 \times 10^2}$	$\mathbf{3.44 \times 10^2}$	$\mathbf{3.44 \times 10^2}$	$\mathbf{3.44 \times 10^2}$	3.50×10^2	3.64×10^2	3.80×10^2	$\mathbf{3.44 \times 10^2}$
F_{24}	$\mathbf{2.69 \times 10^2}$	2.71×10^2	2.72×10^2	2.72×10^2	2.72×10^2	2.72×10^2	2.71×10^2	2.70×10^2
F_{25}	2.05×10^2	2.05×10^2	2.05×10^2	2.05×10^2	2.08×10^2	2.10×10^2	2.12×10^2	$\mathbf{2.05 \times 10^2}$
F_{26}	$\mathbf{1.00 \times 10^2}$	$\mathbf{1.00 \times 10^2}$	$\mathbf{1.00 \times 10^2}$	1.02×10^2	1.04×10^2	1.06×10^2	1.09×10^2	$\mathbf{1.00 \times 10^2}$
F_{27}	3.33×10^2	3.23×10^2	$\mathbf{3.11 \times 10^2}$	3.12×10^2	3.17×10^2	4.09×10^2	5.12×10^2	3.13×10^2
F_{28}	1.15×10^3	1.13×10^3	1.15×10^3	1.20×10^3	1.46×10^3	1.40×10^3	1.40×10^3	$\mathbf{1.11 \times 10^3}$
F_{29}	8.43×10^2	8.32×10^2	8.30×10^2	1.07×10^3	1.50×10^3	1.40×10^3	1.34×10^3	$\mathbf{8.24 \times 10^2}$
F_{30}	$\mathbf{8.92 \times 10^3}$	9.05×10^3	9.18×10^3	1.14×10^4	1.54×10^4	2.05×10^4	3.29×10^4	9.20×10^3
Rank	2.98	2.70	3.25	4.53	5.80	6.67	7.43	2.63

Based on the above experiments, it is demonstrated that the proposed adaptive promising individual selection strategy, for the estimation of the mean vector, is very useful for ACSEDA to not only achieve promising performance but also alleviate the sensitivity to the parameter *sr*.

4.3.4. Effectiveness of the Proposed Cross-Generation Individual Selection Strategy

At last, we conduct experiments to verify the usefulness of the proposed cross-generation individual selection strategy for the parent population. To this end, we first develop three other ACSEDA variants by using some existing typical selection strategies for the parent population. The first is to directly utilize the generated offspring as the parent population, such as in some traditional EDAs [20,21,28,30]. This variant of ACSEDA

is denoted as "ACSEDA-O". The second is to combine the parent population in the last generation and the generated offspring and then, select the best half of the combined population as the parent population for the next generation, as in some EDA variants [51,68]. This variant of ACSEDA is represented as "ACSEDA-OP". The last one is to maintain an archive l in some EDA variants [37] to store the historical useful individuals and then they are combined with the generated offspring to select the best half of the combined population as the parent population. This ACSEDA variant is denoted as "ACSEDA-OA".

After the preparation of the compared methods, we conduct experiments on the 50-*D* CEC 2015 benchmark problems to compare the ACSEDA with the proposed cross-generation individual selection strategy and the ones with the above mentioned three compared strategies. Table 8 presents the comparison results among these different variants of ACSEDA.

Table 8. Comparison among ACSEDA with different selection strategies for the parent population on the 50-*D* CEC 2014 benchmark problems. The best results are highlighted in bold in this table.

F	ACSEDA	ACSEDA-O	ACSEDA-OP	ACSEDA-OA
F_1	1.14×10^{-14}	9.80×10^{1}	5.68×10^{-15}	$\mathbf{5.21 \times 10^{-15}}$
F_2	7.40×10^{-2}	6.04×10^{2}	$\mathbf{1.04 \times 10^{-13}}$	8.62×10^{-13}
F_3	$\mathbf{0.00 \times 10^{0}}$	4.29×10^{-11}	$\mathbf{0.00 \times 10^{0}}$	$\mathbf{0.00 \times 10^{0}}$
F_4	9.25×10^{1}	9.59×10^{1}	$\mathbf{9.12 \times 10^{1}}$	9.68×10^{1}
F_5	$\mathbf{2.11 \times 10^{1}}$	$\mathbf{2.11 \times 10^{1}}$	$\mathbf{2.11 \times 10^{1}}$	$\mathbf{2.11 \times 10^{1}}$
F_6	1.74×10^{-2}	$\mathbf{4.15 \times 10^{-3}}$	1.74×10^{-2}	3.47×10^{-2}
F_7	$\mathbf{0.00 \times 10^{0}}$	8.03×10^{-12}	1.14×10^{-14}	$\mathbf{0.00 \times 10^{0}}$
F_8	$\mathbf{2.98 \times 10^{0}}$	4.84×10^{0}	3.13×10^{2}	5.14×10^{1}
F_9	$\mathbf{2.49 \times 10^{0}}$	4.88×10^{0}	3.16×10^{2}	1.23×10^{2}
F_{10}	1.07×10^{2}	$\mathbf{4.99 \times 10^{1}}$	1.22×10^{4}	6.17×10^{3}
F_{11}	$\mathbf{1.18 \times 10^{2}}$	2.84×10^{2}	1.27×10^{4}	9.74×10^{3}
F_{12}	3.35×10^{0}	$\mathbf{3.19 \times 10^{0}}$	3.23×10^{0}	3.20×10^{0}
F_{13}	1.40×10^{-1}	$\mathbf{8.34 \times 10^{-2}}$	2.66×10^{-1}	2.21×10^{-1}
F_{14}	2.40×10^{-1}	2.91×10^{-1}	$\mathbf{2.28 \times 10^{-1}}$	2.35×10^{-1}
F_{15}	$\mathbf{4.81 \times 10^{0}}$	5.42×10^{0}	2.81×10^{1}	2.43×10^{1}
F_{16}	$\mathbf{1.80 \times 10^{1}}$	1.88×10^{1}	2.15×10^{1}	1.96×10^{1}
F_{17}	$\mathbf{1.16 \times 10^{2}}$	7.10×10^{2}	1.54×10^{3}	1.42×10^{2}
F_{18}	$\mathbf{8.47 \times 10^{0}}$	7.40×10^{1}	9.10×10^{1}	1.15×10^{1}
F_{19}	$\mathbf{1.08 \times 10^{1}}$	1.22×10^{1}	1.15×10^{1}	1.14×10^{1}
F_{20}	$\mathbf{1.95 \times 10^{0}}$	1.54×10^{1}	6.69×10^{1}	4.34×10^{0}
F_{21}	$\mathbf{1.99 \times 10^{2}}$	4.93×10^{2}	9.70×10^{2}	2.59×10^{2}
F_{22}	$\mathbf{3.54 \times 10^{1}}$	8.56×10^{1}	9.26×10^{2}	5.28×10^{1}
F_{23}	$\mathbf{3.44 \times 10^{2}}$	$\mathbf{3.44 \times 10^{2}}$	$\mathbf{3.44 \times 10^{2}}$	$\mathbf{3.44 \times 10^{2}}$
F_{24}	2.68×10^{2}	2.72×10^{2}	2.68×10^{2}	$\mathbf{2.67 \times 10^{2}}$
F_{25}	$\mathbf{2.05 \times 10^{2}}$	$\mathbf{2.05 \times 10^{2}}$	$\mathbf{2.05 \times 10^{2}}$	$\mathbf{2.05 \times 10^{2}}$
F_{26}	$\mathbf{1.00 \times 10^{2}}$	1.01×10^{2}	$\mathbf{1.00 \times 10^{2}}$	$\mathbf{1.00 \times 10^{2}}$
F_{27}	$\mathbf{3.17 \times 10^{2}}$	3.44×10^{2}	3.25×10^{2}	3.30×10^{2}
F_{28}	1.16×10^{3}	1.36×10^{3}	$\mathbf{1.14 \times 10^{3}}$	1.19×10^{3}
F_{29}	8.34×10^{2}	1.81×10^{4}	$\mathbf{8.17 \times 10^{2}}$	9.04×10^{2}
F_{30}	$\mathbf{8.57 \times 10^{3}}$	1.07×10^{4}	8.60×10^{3}	8.74×10^{3}
Rank	**1.77**	2.92	2.82	2.50

From Table 8, we can see that, from the perspective of the averaged rank obtained from the Friedman test and the number of problems where the algorithm achieves the best results, the ACSEDA with the proposed cross-generation individual selection strategy obtains the best overall performance. In particular, not only is the averaged rank is much smaller than those of the compared methods but the number of problems where the proposed ACSEDA achieves the best results is also much larger than those of the compared methods.

The above observations demonstrate that the proposed cross-generation individual selection strategy for the parent population is very helpful for ACSEDA to obtain promising

performance. This is because, by combining the generated offspring in the last generation and in the current generation, this strategy is less likely to generate crowded individuals for the parent population in the next generation and could, thus, aid ACSEDA to preserve high search diversity during the evolution, as analyzed in Section 3.3.

5. Conclusions

This paper has proposed an adaptive covariance scaling estimation of distribution algorithm (ACSEDA) to solve optimization problems. First, instead of estimating the mean vector and the covariance, based on the same selected promising individuals like traditional EDAs, the proposed ACSEDA estimates the covariance based on an enlarged number of promising individuals. In this way, the sampling range of the estimated probability distribution model is enlarged and thus, the estimated model could generate more diversified offspring, which is helpful to avoid falling into local areas. To alleviate the sensitivity of the associated parameter, we further devise an adaptive covariance scaling method to dynamically adjust the covariance scaling parameter during the evolution. Second, to further help ACSEDA to explore and exploit the solution space properly, this paper further devises an adaptive promising individual selection strategy for the estimation of the mean vector. By dynamically adjusting the selection ratio parameter related to the estimation of the mean vector, the proposed ACSEDA gradually biases to exploit the found promising areas without serious loss of diversity as the evolution goes. At last, to further promote the diversity of the proposed ACSEDA, we develop a cross-generation individual selection strategy for the parent population. Different from existing selection methods, the proposed selection method combines the randomly sampled offspring in the last generation and the one in the current generation, together, and then selects the best half of the combined population as the parent population to estimate the probability distribution model. With the cohesive collaboration among the three devised techniques, the proposed ACSEDA is expected to explore and exploit the solution space appropriately and thus, is likely to achieve promising performance.

Extensive comparison experiments have been conducted on the widely used CEC 2014 benchmark problem set with different dimension sizes (30-*D*, 50-*D*, and 100-*D*). Experimental results have demonstrated that the proposed ACSEDA achieves very competitive, or even much better, performance than several state-of-the-art EDA variants. The comparison results also show that ACSEDA preserves good scalability to solve higher-dimensional optimization problems. In addition, deep investigations on the effectiveness of the three proposed techniques have also been performed. The investigation results have demonstrated that the three proposed mechanisms make great contributions to helping ACSEDA to achieve promising performance.

Author Contributions: Q.Y.: Conceptualization, supervision, methodology, formal analysis, and writing—original draft preparation. Y.L.: Implementation, formal analysis, and writing—original draft preparation. X.-D.G.: Methodology, and writing—review and editing. Y.-Y.M.: Writing—review and editing. Z.-Y.L.: Writing—review and editing, and funding acquisition. S.-W.J.: Writing—review and editing. J.Z.: Conceptualization and writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hasan, M.Z.; Al-Rizzo, H. Optimization of Sensor Deployment for Industrial Internet of Things Using a Multiswarm Algorithm. *IEEE Internet Things J.* **2019**, *6*, 10344–10362. [CrossRef]
2. Li, H.; Yu, J.; Yang, M.; Kong, F. Secure Outsourcing of Large-scale Convex Optimization Problem in Internet of Things. *IEEE Internet Things J.* **2021**, 1. [CrossRef]
3. Zhou, X.G.; Peng, C.X.; Liu, J.; Zhang, Y.; Zhang, G.J. Underestimation-Assisted Global-Local Cooperative Differential Evolution and the Application to Protein Structure Prediction. *IEEE Trans. Evol. Comput.* **2020**, *24*, 536–550. [CrossRef]
4. Zeng, X.; Wang, W.; Chen, C.; Yen, G.G. A Consensus Community-Based Particle Swarm Optimization for Dynamic Community Detection. *IEEE Trans. Cybern.* **2020**, *50*, 2502–2513. [CrossRef]
5. Chen, W.N.; Tan, D.Z.; Yang, Q.; Gu, T.; Zhang, J. Ant Colony Optimization for the Control of Pollutant Spreading on Social Networks. *IEEE Trans. Cybern.* **2020**, *50*, 4053–4065. [CrossRef]
6. Shen, Y.; Li, W.; Li, J. An Improved Estimation of Distribution Algorithm for Multi-compartment Electric Vehicle Routing Problem. *J. Syst. Eng. Electron.* **2021**, *32*, 365–379.
7. Li, X.; Tang, K.; Omidvar, M.N.; Yang, Z.; Qin, K.; China, H. Benchmark Functions for the CEC 2013 Special Session and Competition on Large-scale Global Optimization. 2013. Available online: https://www.tflsgo.org/assets/cec2018/cec2013-lsgo-benchmark-tech-report.pdf (accessed on 6 December 2021).
8. Wu, G.; Mallipeddi, R.; Suganthan, P.N. Problem Definitions and Evaluation Criteria for the CEC 2017 Competition on Constrained Real-parameter Optimization. 2017. Available online: https://moam.info/problem-definitions-and-evaluation-criteria-for-the-_5bad2530097c479e798b46a8.html (accessed on 6 December 2021).
9. Wei, F.F.; Chen, W.N.; Yang, Q.; Deng, J.; Luo, X.N.; Jin, H.; Zhang, J. A Classifier-Assisted Level-Based Learning Swarm Optimizer for Expensive Optimization. *IEEE Trans. Evol. Comput.* **2021**, *25*, 219–233. [CrossRef]
10. Yang, Q.; Chen, W.N.; Gu, T.; Jin, H.; Mao, W.; Zhang, J. An Adaptive Stochastic Dominant Learning Swarm Optimizer for High-Dimensional Optimization. *IEEE Trans. Cybern.* **2020**, 1–17. [CrossRef]
11. Tanweer, M.R.; Suresh, S.; Sundararajan, N. Dynamic Mentoring and Self-regulation Based Particle Swarm Optimization Algorithm for Solving Complex Real-world Optimization Problems. *Inf. Sci.* **2016**, *326*, 1–24. [CrossRef]
12. Wu, X.; Zhao, J.; Tong, Y. Big Data Analysis and Scheduling Optimization System Oriented Assembly Process for Complex Equipment. *IEEE Access* **2018**, *6*, 36479–36486. [CrossRef]
13. Yang, Q.; Chen, W.; Deng, J.D.; Li, Y.; Gu, T.; Zhang, J. A Level-Based Learning Swarm Optimizer for Large-Scale Optimization. *IEEE Trans. Evol. Comput.* **2018**, *22*, 578–594. [CrossRef]
14. Yang, Q.; Chen, W.; Gu, T.; Zhang, H.; Deng, J.D.; Li, Y.; Zhang, J. Segment-Based Predominant Learning Swarm Optimizer for Large-Scale Optimization. *IEEE Trans. Cybern.* **2017**, *47*, 2896–2910. [CrossRef]
15. Yang, Q.; Chen, W.; Yu, Z.; Gu, T.; Li, Y.; Zhang, H.; Zhang, J. Adaptive Multimodal Continuous Ant Colony Optimization. *IEEE Trans. Evol. Comput.* **2017**, *21*, 191–205. [CrossRef]
16. Tanabe, R.; Ishibuchi, H. A Review of Evolutionary Multimodal Multiobjective Optimization. *IEEE Trans. Evol. Comput.* **2020**, *24*, 193–200. [CrossRef]
17. Yang, Q.; Chen, W.; Zhang, J. Evolution Consistency Based Decomposition for Cooperative Coevolution. *IEEE Access* **2018**, *6*, 51084–51097. [CrossRef]
18. Yang, Q.; Chen, W.N.; Gu, T.; Zhang, H.; Yuan, H.; Kwong, S.; Zhang, J. A Distributed Swarm Optimizer with Adaptive Communication for Large-Scale Optimization. *IEEE Trans. Cybern.* **2020**, *50*, 3393–3408. [CrossRef]
19. Doerr, B.; Krejca, M.S. Significance-Based Estimation-of-Distribution Algorithms. *IEEE Trans. Evol. Comput.* **2020**, *24*, 1025–1034. [CrossRef]
20. Hauschild, M.; Pelikan, M. An Introduction and Survey of Estimation of Distribution Algorithms. *Swarm Evol. Comput.* **2011**, *1*, 111–128. [CrossRef]
21. Larrañaga, P.; Lozano, J.A. *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*; Springer: Berlin/Heidelberg, Germany, 2001.
22. Bao, L.; Sun, X.; Gong, D.; Zhang, Y. Multi-source Heterogeneous User Generated Contents-driven Interactive Estimation of Distribution Algorithms for Personalized Search. *IEEE Trans. Evol. Comput.* **2021**, 1. [CrossRef]
23. Yang, Q.; Chen, W.; Li, Y.; Chen, C.L.P.; Xu, X.; Zhang, J. Multimodal Estimation of Distribution Algorithms. *IEEE Trans. Cybern.* **2017**, *47*, 636–650. [CrossRef] [PubMed]
24. Shao, W.; Pi, D.; Shao, Z. A Pareto-Based Estimation of Distribution Algorithm for Solving Multiobjective Distributed No-Wait Flow-Shop Scheduling Problem with Sequence-Dependent Setup Time. *IEEE Trans. Autom. Sci. Eng.* **2019**, *16*, 1344–1360. [CrossRef]
25. Shi, W.; Chen, W.; Lin, Y.; Gu, T.; Kwong, S.; Zhang, J. An Adaptive Estimation of Distribution Algorithm for Multipolicy Insurance Investment Planning. *IEEE Trans. Evol. Comput.* **2019**, *23*, 1–14. [CrossRef]
26. Wang, X.; Han, T.; Zhao, H. An Estimation of Distribution Algorithm with Multi-Leader Search. *IEEE Access* **2020**, *8*, 37383–37405. [CrossRef]
27. Krejca, M.S.; Witt, C. Theory of Estimation-of-distribution Algorithms. In *Theory of Evolutionary Computation*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 405–442.

28.   Pelikan, M.; Hauschild, M.W.; Lobo, F.G. Estimation of Distribution Algorithms. In *Springer Handbook of Computational Intelligence*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 899–928.

29.   Montaño, O.D.L.; Gómez-Castro, F.I.; Gutierrez-Antonio, C. Design and Optimization of a Shell-and-tube Heat Exchanger Using the Univariate Marginal Distribution Algorithm. In *Computer Aided Chemical Engineering*; Elsevier: Amsterdam, The Netherlands, 2021; Volume 50, pp. 43–49.

30.   Muelas, S.; Mendiburu, A.; LaTorre, A.; Peña, J.-M. Distributed Estimation of Distribution Algorithms for Continuous Optimization: How Does the Exchanged Information Influence Their Behavior? *Inf. Sci.* **2014**, *268*, 231–254. [CrossRef]

31.   Zhang, Q. On Stability of Fixed Points of Limit Models of Univariate Marginal Distribution Algorithm and Factorized Distribution Algorithm. *IEEE Trans. Evol. Comput.* **2004**, *8*, 80–93. [CrossRef]

32.   Dong, W.; Yao, X. Unified Eigen Analysis on Multivariate Gaussian Based Estimation of Distribution Algorithms. *Inf. Sci.* **2008**, *178*, 3000–3023. [CrossRef]

33.   Gao, B.; Wood, I. TAM-EDA: Multivariate T Distribution, Archive and Mutation Based Estimation of Distribution Algorithm. *Anziam J.* **2012**, *54*, C720–C746. [CrossRef]

34.   Gao, K.; Harrison, J.P. Multivariate Distribution Model for Stress Variability Characterisation. *Int. J. Rock Mech. Min. Sci.* **2018**, *102*, 144–154. [CrossRef]

35.   Gao, Y.; Hu, X.; Liu, H. Estimation of Distribution Algorithm Based on Multivariate Gaussian Copulas. In Proceedings of the IEEE International Conference on Progress in Informatics and Computing, Shanghai, China, 10–12 December 2010; pp. 254–257.

36.   Yang, G.; Li, H.; Yang, W.; Fu, K.; Sun, Y.; Emery, W.J. Unsupervised Change Detection of SAR Images Based on Variational Multivariate Gaussian Mixture Model and Shannon Entropy. *IEEE Geosci. Remote. Sens. Lett.* **2019**, *16*, 826–830. [CrossRef]

37.   Liang, Y.; Ren, Z.; Yao, X.; Feng, Z.; Chen, A.; Guo, W. Enhancing Gaussian Estimation of Distribution Algorithm by Exploiting Evolution Direction with Archive. *IEEE Trans. Cybern.* **2020**, *50*, 140–152. [CrossRef]

38.   Zhou, A.; Sun, J.; Zhang, Q. An Estimation of Distribution Algorithm with Cheap and Expensive Local Search Methods. *IEEE Trans. Evol. Comput.* **2015**, *19*, 807–822. [CrossRef]

39.   Valdez, S.I.; Hernández, A.; Botello, S. A Boltzmann Based Estimation of Distribution Algorithm. *Inf. Sci.* **2013**, *236*, 126–137. [CrossRef]

40.   Ceberio, J.; Irurozki, E.; Mendiburu, A.; Lozano, J.A. A review on estimation of distribution algorithms in permutation-based combinatorial optimization problems. *Prog. Artif. Intell.* **2012**, *1*, 103–117. [CrossRef]

41.   Ren, Z.; Liang, Y.; Wang, L.; Zhang, A.; Pang, B.; Li, B. Anisotropic Adaptive Variance Scaling for Gaussian Estimation of Distribution Algorithm. *Knowl.-Based Syst.* **2018**, *146*, 142–151. [CrossRef]

42.   Bosman, P.A.; Grahl, J.; Rothlauf, F. SDR: A Better Trigger for Adaptive Variance Scaling in Normal EDAs. In Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, New York, NY, USA, 7–11 July 2007; pp. 492–499. [CrossRef]

43.   Grahl, J.; Bosman, P.A.; Rothlauf, F. The Correlation-Triggered Adaptive Variance Scaling IDEA. In Proceedings of the Annual Conference on Genetic and Evolutionary Computation, Seattle, WA, USA, 8–12 July 2006; pp. 397–404.

44.   Liang, J.J.; Qu, B.Y.; Suganthan, P.N. Problem Definitions and Evaluation Criteria for the CEC 2014 Special Session and Competition on Single Objective Real-parameter Numerical Optimization. In *Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report*; Nanyang Technological University: Singapore, 2013.

45.   Bronevich, A.G.; De Oliveira, J.V. On the Model Updating Operators in Univariate Estimation of Distribution Algorithms. *Nat. Comput.* **2016**, *15*, 335–354. [CrossRef]

46.   Rastegar, R. On the Optimal Convergence Probability of Univariate Estimation of Distribution Algorithms. *Evol. Comput.* **2011**, *19*, 225–248. [CrossRef]

47.   Krejca, M.S. *Theoretical Analyses of Univariate Estimation-of-Distribution Algorithms*; Universität Potsdam: Potsdam, Germany, 2019.

48.   Wang, X.; Zhao, H.; Han, T.; Wei, Z.; Liang, Y.; Li, Y. A Gaussian Estimation of Distribution Algorithm with Random Walk Strategies and Its Application in Optimal Missile Guidance Handover for Multi-UCAV in Over-the-Horizon Air Combat. *IEEE Access* **2019**, *7*, 43298–43317. [CrossRef]

49.   Ren, Z.; He, C.; Zhong, D.; Huang, S.; Liang, Y. Enhance Continuous Estimation of Distribution Algorithm by Variance Enlargement and Reflecting Sampling. In Proceedings of the IEEE Congress on Evolutionary Computation, Vancouver, BC, Canada, 24–29 July 2016; pp. 3441–3447.

50.   Yang, P.; Tang, K.; Lu, X. Improving Estimation of Distribution Algorithm on Multimodal Problems by Detecting Promising Areas. *IEEE Trans. Cybern.* **2015**, *45*, 1438–1449. [CrossRef]

51.   Yuan, B.; Gallagher, M. On the Importance of Diversity Maintenance in Estimation of Distribution Algorithms. In Proceedings of the Annual Conference on Genetic and Evolutionary Computation, Washington, DC, USA, 25–29 June 2005; pp. 719–726.

52.   Pošík, P. Preventing Premature Convergence in A Simple EDA Via Global Step Size Setting. In Proceedings of the International Conference on Parallel Problem Solving from Nature, Lecture Notes in Computer Science, Technische Universität, Dortmund, Germany, 13–17 September 2008; pp. 549–558.

53.   Cai, Y.; Sun, X.; Xu, H.; Jia, P. Cross Entropy and Adaptive Variance Scaling in Continuous EDA. In Proceedings of the Annual Conference on Genetic and Evolutionary Computation, New York, NY, USA, 7–11 July 2007; pp. 609–616.

54.   Emmerich, M.; Shir, O.M.; Wang, H. Evolution Strategies. In *Handbook of Heuristics*; Martí, R., Panos, P., Resende, M.G.C., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 1–31.

55.   Ros, R.; Hansen, N. A Simple Modification in CMA-ES Achieving Linear Time and Space Complexity. In Proceedings of the Parallel Problem Solving from Nature–PPSN X, Berlin/Heidelberg, Germany, 13–17 September 2008; pp. 296–305.

56.   Akimoto, Y.; Hansen, N. Diagonal Acceleration for Covariance Matrix Adaptation Evolution Strategies. *Evol. Comput.* **2020**, *28*, 405–435. [CrossRef]

57.   Arabas, J.; Jagodziński, D. Toward a Matrix-Free Covariance Matrix Adaptation Evolution Strategy. *IEEE Trans. Evol. Comput.* **2020**, *24*, 84–98. [CrossRef]

58.   Beyer, H.; Sendhoff, B. Simplify Your Covariance Matrix Adaptation Evolution Strategy. *IEEE Trans. Evol. Comput.* **2017**, *21*, 746–759. [CrossRef]

59.   Li, Z.; Zhang, Q. A Simple Yet Efficient Evolution Strategy for Large-Scale Black-Box Optimization. *IEEE Trans. Evol. Comput.* **2018**, *22*, 637–646. [CrossRef]

60.   He, X.; Zhou, Y.; Chen, Z.; Zhang, J.; Chen, W.N. Large-Scale Evolution Strategy Based on Search Direction Adaptation. *IEEE Trans. Cybern.* **2021**, *51*, 1651–1665. [CrossRef]

61.   Bosman, P.A.; Grahl, J.; Thierens, D. Enhancing the Performance of Maximum–likelihood Gaussian EDAs Using Anticipated Mean Shift. In Proceedings of the International Conference on Parallel Problem Solving from Nature, Technische Universität, Dortmund, Germany, 13–17 September 2008; pp. 133–143.

62.   PourMohammadBagher, L.; Ebadzadeh, M.M.; Safabakhsh, R. Graphical Model Based Continuous Estimation of Distribution Algorithm. *Appl. Soft Comput.* **2017**, *58*, 388–400. [CrossRef]

63.   Yang, Q.; Chen, W.-N.; Zhang, J. Probabilistic Multimodal Optimization. In *Metaheuristics for Finding Multiple Solutions*; Preuss, M., Epitropakis, M.G., Li, X., Fieldsend, J.E., Eds.; Springer International Publishing: Cham, Switzerland, 2021; pp. 191–228.

64.   Huang, X.; Jia, P.; Liu, B. Controlling Chaos by an Improved Estimation of Distribution Algorithm. *Math. Comput. Appl.* **2010**, *15*, 866–871. [CrossRef]

65.   Fang, H.; Zhou, A.; Zhang, G. An Estimation of Distribution Algorithm Guided by Mean Shift. In Proceedings of the IEEE Congress on Evolutionary Computation, Vancouver, BC, Canada, 24–29 July 2016; pp. 3268–3275.

66.   Liu, J.; Wang, Y.; Teng, H. Variance Analysis and Adaptive Control in Intelligent System Based on Gaussian Model. *Int. J. Model. Identif. Control* **2013**, *18*, 26–33. [CrossRef]

67.   Santana, R.; Larranaga, P.; Lozano, J.A. Adaptive Estimation of Distribution Algorithms. In *Adaptive and Multilevel Metaheuristics*; Springer: Berlin/Heidelberg, Germany, 2008; Volume 136, pp. 177–197.

68.   Dong, W.; Chen, T.; Tiňo, P.; Yao, X. Scaling Up Estimation of Distribution Algorithms for Continuous Optimization. *IEEE Trans. Evol. Comput.* **2013**, *17*, 797–822. [CrossRef]

69.   Hansen, N. Towards a New Evolutionary Computation. *Stud. Fuzziness Soft Comput.* **2006**, *192*, 75–102.

70.   Hedar, A.-R.; Allam, A.A.; Fahim, A. Estimation of Distribution Algorithms with Fuzzy Sampling for Stochastic Programming Problems. *Appl. Sci.* **2020**, *10*, 6937. [CrossRef]

71.   Maza, S.; Touahria, M. Feature Selection for Intrusion Detection Using New Multi-objective Estimation of Distribution Algorithms. *Appl. Intell.* **2019**, *49*, 4237–4257. [CrossRef]