Applied Soft Computing 144 (2023) 110492

Contents lists available at ScienceDirect

Applied Soft Computing

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

# Differential Evolution with wavelet basis function based parameter control and dimensional interchange for diversity enhancement

Zhenghao Song [a], Zhenyu Meng [a,b,*]

[a] Institute of Artificial Intelligence, Fujian University of Technology, Fuzhou, China
[b] Fujian Provincial Key Laboratory of Big Data Mining and Applications, Fujian University of Technology, Fuzhou, China

ARTICLE INFO

*Article history:*
Received 13 August 2022
Received in revised form 25 May 2023
Accepted 26 May 2023
Available online 8 June 2023

*Keywords:*
Dimensional interchange
Diversity enhancement
Numerical optimization
Parameter control
Wavelet basis function

ABSTRACT

Differential Evolution (DE) is a potent population-based global optimization algorithm which has already proven efficient for optimization demands in engineering applications. However, even the state-of-the-art DE variants still suffer premature convergence and lack of diversity. In order to overcome the above mentioned weaknesses, this paper proposes a brand-new DE variant, namely zDE algorithm, for single-objective numerical optimization. The main contributions can be summarized as follows: First, an improved wavelet basis function is incorporated into the generation of the scale factor $F$ and a Minkowski Distance based adaptation scheme is proposed for the adaptation of it. Second, a new trial vector generation strategy is first proposed as a supplementary of the existing strategies, and t-distribution is incorporated into this strategy. Third, a novel diversity enhancement technique is firstly proposed by changing the dimensional parameters of the individuals in the population. The zDE algorithm is validated under 88 benchmarks from the CEC2013, CEC2014, and CEC2017 test suites for real-parameter single-objective optimization, and the results show the superiority of our algorithm in comparison with the recent state-of-the-art DE variants.



## 1. Introduction

Differential Evolution (DE), proposed by Storn and Price in 1995 [1] as a branch of the evolutionary algorithm family, was initially proposed to solve the Chebyshev inequality problem [2–4], which made it a newcomer in the field of evolutionary computation. As one of the population-based stochastic optimization algorithm that employs real-valued encoding [5–7], DE relies on the differences between individuals to generate offspring and guide the evolution of the population. Through mutation operators based on individual differences and "greedy" selection scheme, DE updates the population to obtain optimal solutions upon satisfying the termination criterion. At the First International Contest on Evolutionary Optimization, DE has demonstrated its highly competitive optimization capability, thus garnering attention from researchers in computational intelligence and other fields. Subsequent research revealed that DE can effectively tackle complex optimization problems, leading to its increasing influence and application scope. Recently, DE has found widespread applications in diverse domains, including data mining, feature extraction, image processing, production scheduling, multi-objective optimization owing to its simple structure, fast convergence, and strong robustness [8–10]. However, DE still exhibits three deficiencies when compared to other stochastic optimization evolutionary algorithms. Firstly, DE overly relies on parameters, and different optimization problems necessitate appropriate parameter tuning for achieving optimal results. Secondly, DE also suffers from inadequate local search capability, making it susceptible to local optima and premature convergence. Moreover, DE's optimization accuracy tends to be relatively low when handling high-dimensional optimization problems [11,12]. Thus, in-depth analysis and research on DE's limitations and potential improvements hold significant academic and practical value [13–15].

Generally, the effectiveness of DE is heavily dependent on the balance between exploration and exploitation capabilities [16–18]. This can be accomplished through the development of more efficient mutation strategies and parameter adaptation schemes. To this end, over the past few decades, various methods have been developed to improve the search capability of DE. Meng et al. [19] proposed a cooperative strategy pool that leverages two mutation strategies to generate trial vectors, enabling individuals to adapt to these strategies over the course of evolution. Besides, a new parameter adaptation scheme is introduced, where all three control parameters including scale factor, crossover rate and population size are automatically adjusted. Zheng et al. [20] developed a collective information-powered DE (CIPDE), where

* Corresponding author at: Institute of Artificial Intelligence, Fujian University of Technology, Fuzhou, China.
*E-mail address:* mzy1314@gmail.com (Z. Meng).

https://doi.org/10.1016/j.asoc.2023.110492
1568-4946/

information of $m$ best vectors is combined to generate the difference vector in a linear manner. Liao et al. [21] proposed a DE variant with cellular direction information, which is utilized to determine a neighborhood for each individual. The new mutation strategy is based on the difference vector in the neighborhood. Cai et al. [22] suggested a neighborhood-based mutation operator by constructing multiple neighborhood relationships and adaptively choosing a neighborhood topology from an index-based topology pool. Sun et al. [23] designed a topology-dependent mutation strategy by assigning each individual an appropriate candidate topology. Qin et al. [24] proposed a self-adapted DE, where both mutation strategies and related control parameters are automatically updated. Tian et al. [25] proposed a novel DE variant, where two mutation operators based on neighborhood and a selection probability based on individual are introduced to improve the search ability of each individual.

Besides the novel mutation strategies, parameter adaptation scheme also plays a key role in enhancing searching ability [26, 27]. Numerous methods have been proposed for parameter control of DE, taking into account diverse viewpoints such as the incorporation of feedback data from the population and individual information.t perspectives, including the utilization of feedback data from the population and the information of individuals [28–30]. In 2005, Liu et al. [31] introduced a fuzzy adaptive DE for parameter control. The update process of parameters is conducted by a fuzzy logic controller in the proposed algorithm, even though the performance improvement is marginal. Brest et al. [32] proposed a DE variant known as jDE, where control parameters are encoded to each individual, and adapted as the evolution proceeds. The fundamental concept behind jDE is that individuals exhibiting superior fitness may possess more effective control parameters, which are more likely to be inherited by their offspring, thus speeding up the convergence. The idea of encoding control parameters into each individual was further improved and implemented in JADE [33], which became the winner of large scale optimization competition held in 2008. In JADE, both scale factor $F$ and crossover rate $CR$ are adapted based on their historical records of successful individuals. By doing so, a better balance between exploration and exploitation abilities can be achieved. On the basis of parameter adaptation of JADE, Tanabe and Fukunaga proposed LSHADE, in which a fitness-dependent weighting method is introduced for updating $F$ and $CR$. Tang et al. [34] developed an individual-dependent parameter set to assign suitable parameters to each individual for fully utilizing fitness differences between individuals. Yu et al. [29] proposed a DE variant with two-level parameter adaptation, where the population-level parameters $F$ and $CR$ for the whole population are adapted based on different optimization states and individual-level parameters for each individual are produced using the population-level parameters. Zhou et al. [35] proposed a variant of JADE with adaptive parameter control, named JADE-sort, where individuals with higher fitness will be assigned to a low crossover rate so as to keep producing promising solution in the following generations.

In addition to the mutation strategy and parameter control scheme, population diversity is crucial to the search effectiveness of DE, particularly in the later stages of evolution. For most DE variants for single objective optimization, individuals tend to converge into several local optimum or global optimum before the end of evolution, and population diversity will decrease inevitably. However, if population diversity is too low, then some individuals may become stagnant and fail to jump out of local optimum, thus resulting in low optimization accuracy. Recently, researchers have developed several methods to improve local exploitation ability by improving population diversity. For example, He et al. [36] proposed the fuzzy neighborhood-based directed differential evolution algorithm (FNODE), which selects suitable individuals to form a neighborhood based on fuzzy rules and individual distribution, and then incorporates the directional information of neighborhood individual migration into the variance to enhance population diversity. The fitness landscape features reflect the optimal solution distribution, number and local single-peaked topology of the optimization problem from different perspectives. Li et al. [37] combined the fitness landscape with the DE to quantitatively analyze the information related to the fitness distance and evaluate the difficulty of optimization problem. The proposed algorithm can avoid the algorithm from falling into local optimum and improve the solution accuracy and convergence performance. Yang et al. [38] designed an auto-enhanced population diversity mechanism to resolve the problems of premature convergence and stagnation by measuring the distribution of population in each dimension. In CS-DE [19], a population diversity enhancement was proposed, where a restart mechanism is performed when individuals are detected by stagnation indicators as stagnant individuals. Here in this paper, we advanced a new DE variant, and the main highlights are summarized as follows:

1. An improved wavelet basis function is employed in the generation of scale factor $F$ instead of employing the Cauchy distribution, and the Minkowski distance based adaptation is proposed for parameter control.
2. A novel mutation strategy with t-distribution based perturbation scheme is developed, and a new crossover method is proposed as well aiming at maintaining a better diversity of the population.
3. A novel stagnation indicator is proposed and the corresponding dimension interchange for the diversity enhancement is launched when the population is in stagnation status.
4. Our zDE algorithm is validated on 88 benchmarks from the CEC2013, CEC2014, and CEC2017 test suites for numerical optimization, and the experiment results reveal that our algorithm is quite competitive with several recently proposed state-of-the-art DE variants.

The rest of the paper is arranged as follows: A review of several state-of-art DE variants is presented in Section 2. Section 3 provides details of our algorithm. Section 4 presents experiment results in comparison with DE variants under three test suites. Section 5 concludes the paper.

## 2. Several powerful DE variants

In this section, we briefly review some powerful DE variants which are published recently, including LSHADE [13], jSO [39], Hip-DE [40], CS-DE [19], and TDE [41], because our zDE algorithm is developed on the basis of these variants.

### 2.1. LSHADE

Tanabe and Fukunaga [13] proposed the LSHADE algorithm in 2014. It is an improved version of SHADE. The parameter control based on historical memory was firstly introduced in the SHADE algorithm, where control parameters for subsequent generations are guided by this mechanism. In LSHADE, the parameter pool is initially set to $F = CR = 0.5$, and $\mu_F$ and $\mu_{CR}$ store $H$ historical memories. The scale factor $F$ and crossover rate $CR$ in LSHADE obey the Cauchy distribution and normal standard distribution, respectively. Each individual employs a random entry chosen from the pool in generating $F$ and $CR$. The $F$ and $CR$ of successful individuals are used to update $\mu_F$ and $\mu_{CR}$, respectively. The parameter adaptation methods for $F$ and $CR$ are shown in Eqs. (1) and (2).

$$F = \begin{cases} randc(\mu_{F,H}, 0.1), \ while\ F \le 0 \\ 1.\ if\ F > 1 \end{cases} \tag{1}$$

$$CR = \begin{cases} 0, \ if\ \mu_{CR} = 0 \\ randn_i(\mu_{CR,H}, 0.1), \ otherwise \end{cases} \tag{2}$$

$$\begin{cases} w_k = \dfrac{\Delta f_j}{\sum_{k=1}^{|S_F|} \Delta f_j} \\ \Delta f_j = f(X_{j,G}) - f(U_{j,G}) \\ \text{mean}_{WL}(S_F) = \dfrac{\sum_{k=1}^{|S_F|} w_k \cdot S_{F,k}^2}{\sum_{k=1}^{|S_F|} w_k \cdot S_{F,k}^2} \\ \mu_{F,k,G+1} = \begin{cases} \text{mean}_{WL}(S_F), & \text{if } S_F \ne \varnothing \\ \mu_{F,k,G}, & \text{otherwise} \end{cases} \end{cases} \tag{3}$$

$$\begin{cases} w_k = \dfrac{\Delta f_j}{\sum_{k=1}^{|S_{CR}|} \Delta f_j} \\ \Delta f_j = f(X_{j,G}) - f(U_{j,G}) \\ \text{mean}_{WL}(S_F) = \dfrac{\sum_{k=1}^{|S_{CR}|} w_k \cdot S_{CR,k}^2}{\sum_{k=1}^{|S_{CR}|} w_k \cdot S_{CR,k}^2} \\ \mu_{CR,k,G+1} = \begin{cases} \text{mean}_{WL}(S_{CR}), & \text{if } S_{CR} \ne \varnothing \\ \mu_{CR,k,G}, & \text{otherwise} \end{cases} \end{cases} \tag{4}$$

where $S_F$ and $S_{CR}$ are corresponding $F$ and $CR$ of trial vector whose fitness value is better than that of its target vector. The symbol $\Delta f_j$ denotes the fitness difference between trial vector and its corresponding target vector. $mean_{WL}$ denotes the weighted Lehmer mean. In addition, a population size reduction mechanism is also employed in LSHADE, which is presented in Eq. (5).

$$PS_{G+1} = round\left[\left(\frac{PS_{min} - PS_{init}}{nfe_{max}}\right) \cdot nfe + PS_{init}\right] \tag{5}$$

where $PS_{G+1}$ represents the population size of the next generation, $PS_{init}$ denotes the initial population size, and $PS_{min}$ denotes the minimum population size defined by LSHADE. By the way, as the population size decreases, the storage of external archive A should be reduced according to Eq. (6).

$$|A| = r^{arc} \cdot PS_G \tag{6}$$

where $|A|$ denotes the size of external archive whose ratio to population size is $r^{arc}$.

### 2.2. jSO

The jSO algorithm was proposed in 2017 [39], and it is an improved version of LSHADE by modifying the mutation strategy. The weighted mutation strategy is called "DE/current-to-$p$best-w/1", which is shown as follows.

$$V_{i,G} = X_{i,G} + F_w \cdot F \cdot (X_{best,G}^p - X_{i,G}) + F \cdot (X_{r_1,G} - \widetilde{X}_{r_2,G}) \tag{7}$$

$$F_w = \begin{cases} 0.7, & if\ nfe < 0.2 \cdot nfe_{max} \\ 0.8, & if\ 0.2 \le nfe < 0.4 \cdot nfe_{max} \\ 1.2, & otherwise \end{cases} \tag{8}$$

where same symbols have the similar meaning as ones in LSHADE. $F_w$ denotes a inertia weight for controlling the difference vector $X_{best,G}^p - X_{i,G}$, and it is adapted according to Eq. (6). In addition, the scale factor $F$ and crossover rate $CR$ of each individual are generated according to Eq. (9) and Eq. (10), respectively.

$$F_{i,G} = \begin{cases} min(F_{i,G}, 0.7), & if\ nfe < 0.6nfe_{max} \\ 0.7 & otherwise \end{cases} \tag{9}$$

$$CR_{i,G} = \begin{cases} max(CR_{i,G}, 0.7), & if\ nfe < 0.25nfe_{max} \\ max(CR_{i,G}, 0.6), & if\ 0.25nfe_{max} \le nfe < 0.5nfe_{max} \\ CR_{i,G}, & otherwise \end{cases} \tag{10}$$

### 2.3. Hip-DE

Mutation strategy and corresponding parameter control are two main directions of improving the performance of DE and its variants. Although many state-of-art DE variants have secured excellent performance by modifying these two components, there still exist weaknesses such as insufficient utilization of population knowledge during the evolution and interlacement between scale factor and crossover rate in parameter adaptation. Based on considerations above, a new DE variant named as Hip-DE was proposed in 2021 [40]. A mutation strategy based on historical population is introduced to strike a better balance between the exploration and exploitation abilities. The details of the strategy is shown in Eq. (11):

$$V_{i,G} = X_{i,G} + F \cdot (X_{best,G}^p - X_{i,G}) + F_1 \cdot (X_{r_1,G} - \widehat{X}_{r_2,G}) \tag{11}$$

where $\widehat{X}_{r_2,G}$ denotes a vector randomly chosen from the union $\mathbf{P} \cup \mathbf{H}$, in which $\mathbf{H}$ stores historical populations of past generations. The reason using the historical populations lies in that the relationship between current population and the historical population is able to reveal the landscape of objective function and information obtained from the relationship is constructive to steer the evolution toward a better direction.

The core idea behind many parameter adaptation schemes is that better control parameters tend to generation better offspring, which incline to survive and pass on these parameters to the next generations. However, a better scale factor $F$ and a worse crossover rate $CR$ (or vice verse) may produce a better offspring whose worse parameters will be used in the parameter adaptation. On basis of this observation, a grouping strategy is incorporated into the parameter adaptation for separating $F$ and $CR$ into different group and updating them independently. In Hip-DE, the population are divided into $K$ groups according to selection probability. The initial probability of each group is equal $\{p(k)|p(1) = p(2) = \cdots = p(k) = \cdots = p(K) = \frac{1}{K}, k \in \{1, 2, \ldots, K\}\}$, and the selection probability is renewed as follows:

$$\begin{cases} r_k = \begin{cases} \dfrac{ns_k^2}{ns \cdot (ns_k + nf_k)}, & if\ ns_k > 0 \\ \varepsilon, \ otherwise \end{cases} \\ p(k) = \dfrac{r_k}{\sum_{k=1}^{K}(r_k)}, \end{cases} \tag{12}$$

In Hip-DE, the scale factor $F$ and crossover rate $CR$ of each individual are generated according to Eqs. (14) and (15):

$$\begin{cases} F_{tmp} = randc(\mu_{F_k}, 0.1) \\ F_{tmp} = \begin{cases} randc(\mu_{F,h}, 0.1), \ while\ F \le 0 \\ 1, \ if\ F > 1 \end{cases} \end{cases} \tag{13}$$

$$F_{i,G+1} = \begin{cases} F_{tmp}, \ if\ rand_1 < \tau_1 \\ F_{i,G}, \ otherwise \end{cases} \tag{14}$$

$$CR_{i,G+1} = \begin{cases} randn(\mu_{CR_k}, 0.1), \ if\ rand_2 < \tau_2 \& \mu_{CR} \ne 0 \\ 0, \ if\ rand_2 < \tau_2 \& \mu_{CR} = 0 \\ CR_{i,G}, \ otherwise \end{cases} \tag{15}$$

where $rand_1$ and $rand_2$ are two numbers randomly generated within [0, 1]; $\tau_1$ and $\tau_2$ are two fixed ratio both equaling to 0.9.

### 2.4. CS-DE

The CS-DE [19] algorithm was proposed in 2021. The key innovation is a collaborative strategy pool with two similar mutation strategies, each of which has a shared parameter control. The mutation strategy ensemble consists of two mutation strategies which can be chosen by individuals to produce trial vectors. The two similar mutation strategies are presented in Eq. (16).

$$\begin{cases} V_{i,G} = X_{i,G} + F_b \cdot (X^p_{best,G} - X_{i,G}) + F \cdot (X_{r_1,G} - \widehat{X}_{r_2,G}) \\ V_{i,G} = X_{i,G} + F_b \cdot (X^p_{best,G} - X_{i,G}) + F \cdot (X_{r_1,G} - \widetilde{X}_{r_2,G}) \end{cases} \tag{16}$$

where $\widehat{X}_{r_2,G}$ and $\widetilde{X}_{r_2,G}$ denote individual randomly selected from the union of **P**∪**A** and **P**∪**B**, respectively. **P**∪**A** denotes the union of individuals in the current population and external archive storing inferior individuals, and **P**∪**B** denotes the union of current individuals and external archive storing historical individuals in the past generations. Furthermore, the parameter $F_b$ is similar to the one in jSO, and the selection of $F_b$ is shown as follows.

$$F_b = \begin{cases} 0.7 \cdot F, & if\ nfe < 0.2 \cdot nfe_{\max} \\ 0.8 \cdot F, & if\ 0.2 \cdot nfe_{\max} \leq nfe < 0.4 \cdot nfe_{\max} \\ F, & otherwise \end{cases} \tag{17}$$

After generating the initial population, all individuals perform the mutation and crossover operation, thus producing trial vectors. Following fitness evaluation of trial vectors, a success sign "s" is labeled on the individual with better offspring, otherwise a failure sign "f" is labeled. After that, control parameter $\mu_F$ and $\mu_{CR}$ are updated according to Eqs. (18) and (19).

$$\begin{cases} \Delta f_i = f(U_{i,G}) - f(X_{i,G}) \\ w_s = \dfrac{\Delta f_i}{\sum_{s=1}^{|S|} \Delta f_s} \\ mean_{WL}(S_F) = \dfrac{\sum_{s=1}^{|S|} w_s \cdot S_F^2(s)}{\sum_{s=1}^{|S|} w_s \cdot S_F(s)} \\ c = \dfrac{ns}{PS} \\ \mu_F = \begin{cases} c \cdot \mu_F + (1-c) \cdot (mean_{WL}(S_F)), & if\ S \neq \emptyset \\ \mu_F & otherwise \end{cases} \end{cases} \tag{18}$$

$$\begin{cases} \Delta f_i = f(U_{i,G}) - f(X_{i,G}) \\ w_s = \dfrac{\Delta f_i}{\sum_{s=1}^{|S|} \Delta f_i} \\ mean_{WL}(S_{CR}) = \dfrac{\sum_{s=1}^{|S|} w_s \cdot S_{CR}^2(s)}{\sum_{s=1}^{|S|} w_s \cdot S_{CR}(s)} \\ \mu_{CR_{idx,G+1}} = \begin{cases} mean_{WL}(S_{CR}), & if\ S \neq \emptyset \\ \mu_{CR_{idx,G}}, & otherwise \end{cases} \end{cases} \tag{19}$$

where $S_F$ denotes the set of scale factor $F$ of "s" individuals, and $S_{CR}$ denotes the set of crossover rate $CR$ of 's"individuals; "s" denotes the index of $S_F$ and $S_{CR}$. Besides parameter adaptation of two main control parameters $F$ and $CR$, the population size $PS$ is also adapted in CS-DE. One reason to adopt a new mechanism for reducing population size is that if the size decreases too rapidly in the early stages of evolution, it can negatively impact the overall ability to explore and improve solutions as the process continues. To address this, a hybrid approach is proposed that combines linear and elliptic reduction methods. The details are shown in Eq. (20).

$$PS = \begin{cases} \left\lceil \sqrt{PS_{ini}^2 - \dfrac{PS_{ini}^2 - y^2}{x^2} \cdot nfe^2} \right\rceil, & if\ nfe < x \\ \left\lceil \dfrac{PS_{\min} - y}{nfe_{\max} - x} \cdot (nfe - x) + y \right\rceil, & otherwise \end{cases} \tag{20}$$

where $Pivot = (x, y)$ denotes the point connecting the ellipse and segment.

### 2.5. TDE

A two-stage differential evolution (TDE) was proposed in 2022 [41], where the evolution is subdivided into two stages using a unique mutation strategy. According to the "No free lunch theorem", each mutation strategy has its inherent strengths and weaknesses. Therefore, TDE with two distinct mutation strategy is expected to improve the performance with respect to solution accuracy. The first mutation strategy is a mutation strategy based on historical solutions and the second one is based on inferior solutions. The two strategies are presented as follows.

$$\begin{cases} V_{i,G} = X_{i,G} + F \cdot (X^p_{best,G} - X_{i,G}) + F_w \cdot (X_{r_1,G} - \widetilde{X}_{r_2,G}), if\ nfe < \rho \\ V_{i,G} = X_{i,G} + F \cdot (X^p_{best,G} - X_{i,G}) + F \cdot (X_{r_1,G} - \widehat{X}_{r_2,G}), otherwise \end{cases} \tag{21}$$

where $\widetilde{X}_{r_2,G}$ denotes a individual randomly selected from the union **P**∪**A**, in which **P** is the current population and **A** is the set of archive storing the inferior individuals; $\widehat{X}_{r_2,G})$ denotes a individual randomly chosen from the union **P** ∪ **A**, in which **B** is the set of archive recording the historical individuals; $F_w$ denote scale factor constricting to 0.9; the parameter $\rho$ acts as the threshold between two stages.

For parameter adaptation, instead of fitness-difference based adaptation schemes used in the winner DE variants in recent CEC competitions, a fitness-independent parameter control is proposed to adjust these control parameters. In the TDE algorithm, all individuals are divided into $K$ groups. The scale factor $F$ of each individual obeys Cauchy distribution, $\mu_{F_k}, 0.1)$. Two parameters are associated with individuals from each group: the Gaussian distribution mean value $\mu_{CR}$ and selection possibility $P(\cdot)$. $\{p(k)|p(1) = p(2) = \cdots = p(k) = \cdots = p(K) = \frac{1}{K}, k \in \{1, 2, \ldots, K\}\}$ of each group. The initial values of $\mu_{F_1} = \mu_{F_2} = \cdots = \mu_{F_k} = \cdots = \mu_{F_K} = \mu_F$ and $\mu_{CR_1} = \mu_{CR_2} = \cdots = \mu_{CR_k} = \cdots \mu_{CR_K} = \mu_{CR}$ = 0.8.

The selection probability of the $k$th group is updated in accordance with Eq. (12) at the end of each generation when the trial vector generation and selection procedures are finished.

$$\begin{cases} r_k = \begin{cases} \dfrac{ns_k^2}{ns \cdot (ns_k + nf_k)}, & if\ ns_k > 0 \\ \varepsilon, & otherwise \end{cases} \\ p(k) = \dfrac{r_k}{\sum_{k=1}^{K}(r_k)}, \end{cases} \tag{22}$$

where $nf_k$ and $ns_k$ are the number of failed and successful trial vectors in the $k$th group respectively, and ns denotes the total number of successful trial vectors in the current generation. The update formulas of $\mu_F$ and $\mu_{CR}$ are presented as follows.

$$\begin{cases} w_s = \dfrac{std(\Delta loc_i)}{\sum_{s=1}^{|S_F|} std(\Delta loc_i)} \\ std(\Delta loc_i) = loc(U_{i,G} - X_{i,G}) \\ mean_{WL}(S_F) = \dfrac{\sum_{s=1}^{|S_F|} w_s \cdot S_F^2(s)}{\sum_{s=1}^{|S_F|} w_s \cdot S_F(s)} \\ \mu_{F_{idx,G+1}} = \begin{cases} mean_{WL}(S_F), & if\ S \neq \emptyset \\ \mu_F, & otherwise \end{cases} \end{cases} \tag{23}$$

where $loc(U_{i,G} - X_{i,G})$ denotes locating the effective dimensional changes of the displacement, $(U_{i,G} - X_{i,G})$. $std(\cdot)$ denotes the standard deviation of the $D-$dimensional changes.

A two-stage population reduction mechanism is proposed in TDE to achieve a balance between the exploration and exploitation abilities. In the first stage, a fixed population size is used for recognizing the objective and a linearly reduced population size is employed for better exploitation at the second stage. The scheme of this mechanism is provided as follows.

$$PS_{G+1} = \begin{cases} PS, & if\ nfe \leq p_s \cdot nfe_{max} \\ \lfloor \frac{PS_{min} - PS_{ini}}{(1-PS) \cdot nfe_{max}} + PS_{ini} \rfloor, & otherwise \end{cases} \tag{24}$$

where $p_s$ denotes the ratio of the fixed population evolution to the whole population, and is set to 0.05 in the TDE.

## 3. The proposed zDE algorithm

In recent years, numerous techniques are developed to enhance DE's performance. However, there still exist some problems to be resolved. For example, (1) how to use the scale factor in the mutation strategy to steer a proper direction for the population; (2) how to select an appropriate donor vector during crossover operation to generate a promising trial vector; (3) how to update the stagnant individuals which are trapped into local optimum. To solve the aforementioned problems, a novel zDE DE variant is proposed, and the whole algorithm can be subdivided into three parts: the first part introduces the bi-stage parameter control with a distance-based weighting strategy; the second part presents the new trial vector generation strategy by incorporating perturbation strategy; and the last one describes the population diversity mechanism.

### 3.1. Parameter adaptation

The performance of DE algorithm largely depends on the adjustment of control parameters. Therefore, A bi-stage parameter control for scale factor $F$ is proposed, in which a wavelet basis function is used to generate $F$ at the first stage of evolution and the Cauchy distribution is used at the second stage. In addition, crossover rate $CR$ obeys the normal distribution as does in LSHADE. The wavelet basis function is a sequence of functions obtained by extension and translation in both time and frequency domains. The adaptation of $F$ and $CR$ is presented as follows: Eqs. (26)–(28).

$$\varepsilon = 0.1 \cdot sin(\pi \cdot rand_i - 0.8) \tag{25}$$

$$F_i = \begin{cases} \sqrt{2} \cdot \pi^{\frac{-1}{3}} \cdot (1 - \mu_{F,r_i}^2) \cdot e^{-\mu_{F,r_i}^2} + \varepsilon, & if\ nfe < \perp \\ randc_i(\mu_{F,r_i}, 0.1), & otherwise \end{cases} \tag{26}$$

$$CR_i = \begin{cases} 0, & if\ \mu_{CR,r_i} = \emptyset \\ randn_i(\mu_{CR,r_i}, 0.1), & otherwise \end{cases} \tag{27}$$

$$CR_i = \begin{cases} max(CR_i, 0.6), min(CR_i, 1) & if\ nfe < \perp \\ max(CR_i, 0), min(CR_i, 1), & otherwise \end{cases} \tag{28}$$

where initial values of $\mu_F$ and $\mu_{CR}$ are 0.5 and 0.8. The $\mu_F$ and $\mu_{CR}$ are updated through the memory update mechanism. During the first stage of evolution, $F$ is maintained in the range of [0.4, 0.6] by using the wavelet basis function. Therefore, we can make full use of the characteristics of the wavelet basis function to adapt $F$ and avoid the premature convergence tendency in "DE/current-to-$p$best".

How to perform more efficient parameter control is another issue that has been explored by many researchers. The update of $\mu_F$ and $\mu_{CR}$ is based on weighted Lehmer mean, in which the weight $w$ is calculated using the fitness difference between trial vector and its corresponding target vector. Although the fitness difference-based weight is easy to calculate, there is a risk of premature convergence, especially in high-dimensional optimization. Based on considerations above, this paper proposes a Minkowski distance-based weighting strategy for updating control parameters.

Minkowski distance takes the individual's position within the search space into account, so the complexity will slightly increase. In the original weighting strategy, the difference between fitness of trial vector and its corresponding target vector can be much large, especially in high-dimensional optimization problem, thus increasing the overall time complexity. The calculation of Minkowski distance is according to Eqs. (29) and (30), and the adaptation mechanisms of $\mu_F$ and $\mu_{CR}$ are presented in Eqs. (31) and (32).

$$w_k = \frac{(\sum_{j=1}^{D} |u_{k,j,G} - x_{k,j,G}|^m)^{\frac{1}{m}}}{\sum_{m=1}^{|S_{CR}|}(\sum_{j=1}^{D} |u_{k,j,G} - x_{k,j,G}|^m)^{\frac{1}{m}}} \tag{29}$$

$$m = round[(\frac{m_{max} - m_{min}}{nfe_{\max}}) \cdot nfe + m_{min}] \tag{30}$$

$$\begin{cases} mean_{WL}(S_F) = \dfrac{\sum_{k=1}^{|S_F|} w_k \cdot S_F^2(k)}{\sum_{k=1}^{|S_F|} w_k \cdot S_F(k)} \\ \mu_{F,k,G+1} = \begin{cases} \frac{(mean_{WL}(S_F)+\mu_{F,k,G})}{2}, & if\ S_F \neq \emptyset \\ \mu_{F,k,G}, & otherwise \end{cases} \end{cases} \tag{31}$$

$$\begin{cases} mean_{WL}(S_{CR}) = \dfrac{\sum_{k=1}^{|S_{CR}|} w_k \cdot S_{CR}^2(k)}{\sum_{k=1}^{|S_{CR}|} w_k \cdot S_{CR}(k)} \\ \mu_{CR,k,G+1} = \begin{cases} mean_{WL}(S_{CR}), & if\ S_{CR} \neq \emptyset \\ \mu_{CR,k,G}, & otherwise \end{cases} \end{cases} \tag{32}$$

where $mean_{WL}(S_F)$ and $mean_{WL}(S_{CR})$ are the weighted Lehmer mean of successful $F$ set and successful $CR$ set respectively, and the weight $w_k$ is the Minkowski distance employed in the proposed algorithm; $m_{min}$ is 1 and $m_{max}$ is 4.

### 3.2. Donor vector perturbation strategy

It is well known that classical DE may suffer from premature convergence during evolution, which can reduce its search capability. In this case, the newly generated trial vectors may fail to escape from local optimum. Perturbation strategy is a common method to assist algorithms in jumping out of local optimum. By implementing a crossover operation between donor vector perturbed by the t-distribution probability density function and target vector, a newly generated trial vector will have higher possibility of escaping from local optima. The pseudo code of crossover operation combining with perturbation strategy is presented in Algorithm 1.

In Algorithm 1, *rand* represents a random number uniformly selected from [0,1], $\tau_1$ and $\tau_2$ are set to 0.005 and 0.5 respectively. $j_{rand}$ is a integer uniformly randomly selected from [1, $D$]. $X_{gbest,G}$ denotes the individual with best fitness at current generation. $tpdf(G, \gamma)$ denotes the probability density function value for each element in $G$ under the t-distribution indicated by degree of freedom $\gamma$. $G$ is the number of iterations currently being performed, and $\gamma$ is the degree of freedom parameter, which adjusts the curve of the t-distribution probability density function. The larger the $\gamma$ is, the faster the curve will fall. In this case, $\gamma$ is set to 0.8, which ensures that the curve descends smoothly and maintains a significant disturbance level in the later stages, which is useful for escaping out of the local optima.

The crossover operation is improved by adopting a perturbation mechanism. The method can improve the local exploitation ability, the convergence speed and robustness of the algorithm.

**Algorithm 1** Donor vector perturbation strategy

**for** $i = 1$ **to** $PS$ **do**
  $V_{i,G} = X_{i,G} + F \cdot (X^{p}_{best,G} - X_{i,G}) + F \cdot (X_{r_1,G} - \widetilde{X}_{r_2,G})$;
  Generate $j_{rand}$ = randint $(1, D)$;
  **for** $j = 1$ **to** $D$ **do**
    **if** $j = j_{rand}$ or $rand(0, 1) < CR_i$ **then**
      $U_{j,i,G} = V_{j,i,G}$;
    **else**
      **if** $rand > \tau_1$ **then**
        $U_{j,i,G} = X_{j,i,G}$;
      **else**
        **if** $rand > \tau_2$ **then**
          $U_{j,i,G} = X_{j,i,G} + rand \cdot$
          $(X_{gbest,G} \cdot (tpdf(G, \gamma) + 1))$;
        **else**
          $U_{j,i,G} = X_{j,i,G} + rand \cdot$
          $mean(X_{j,i,G}) \cdot (tpdf(G, \gamma) + 1))$;
        **end if**
      **end if**
    **end if**
  **end for**
**end for**

### 3.3. Population diversity mechanism

Like other computational intelligence, DE may stagnate during evolution, especially on multimodal optimization problems. The lack of population diversity in the later phase of evolution is the primary cause of this issue. To tackle the problem, the mechanism for enhancing population diversity is proposed. The mechanism can be divided into two parts: one is the detection of stagnation individuals and the other is the population diversity enhancement.

Based on our observation, after a certain number of iterations, the evolution process becomes stable and some individuals fail to improve during a certain number of generations, which can be referred to as the stagnation [25,42]. Therefore, we propose a method to calculate the distribution of individuals in the population by using surrogate hypervolumes [43]. The formula of boundary range is used to calculate the corresponding volume search space. In our proposed method, we utilize the ratio of the volume associated with the boundary and the volume of the search space as a diversity metric. This metric is computed using the following formula:

$$V_{lim} = \sqrt{\prod_{i=1}^{D} |u_i - l_i|} \tag{33}$$

where $u_i$ and $l_i$ denote the upper boundary and lower boundary, respectively. The formula for the diversity measurement in Eq. (34):

$$VOL = \sqrt{V_{pop}/V_{lim}} \tag{34}$$

The detailed calculation of $V_{pop}$ is presented in Algorithm 2:

**Algorithm 2** population feedback mechanism

**for** $j = 1$ **to** $D$ **do**
  $V_{num} = (max(X_{i,j}) - min(X_{i,j}))/2$;
**end for**
$V_{pop} = \sqrt{V_{pop} \cdot V_{num}}$;

Since individuals coalesce into the optimal solution as the algorithm proceeds, population may gather around the local optimum, thus reducing the exploration ability of the algorithm. Tian et al. [25,42] proposed a fitness-based population stagnation mechanism. A counter is used to record the individuals' stagnation state during the evolution. More details about calculation of *counter* can refer to Algorithm 3.

**Algorithm 3** Calculate *counter*

**for** $i = 1$ **to** $PS$ **do**
  **if** $f(X_{i,G}) > f(U_{i,G})$ **then**
    $counter(i) = counter(i) + 1$
  **else**
    $counter(i) = 0$;
  **end if**
**end for**

Once the predefined threshold is satisfied, the individuals are considered to be in the stagnant state. The horizontal crossover and vertical crossover, inspired from the genetic algorithm's crossover operation, are combined to form a straightforward competition mechanism.

Under the premise of not affecting the convergence speed of the algorithm, the dimension of the stagnant individuals is adjusted. Eqs. (35) and (36) denote the horizontal crossover for perturbing the stagnant individuals, and Eq. (37) denotes the vertical crossover. The reason for adopting the above two methods lies in that (1) horizontal crossover is able to split the solution space of the multidimensional problem into semi-group hypercubes, and perform edge search on the space to improve the algorithm's global search capability while reducing blind spots. Meanwhile, individuals of the same dimension are randomly prescreened firstly, and then all prescreened individuals are randomly paired for updating (which can be repeated). (2) vertical crossover enables individuals in a population to perform crossover operation in different dimensions. Individuals in the population may be caught in one dimension of the local optimum due to prematureness. Therefore, it requires to deal with the one dimension that is trapped into the local optimum without affecting other dimensions. The population diversity is then improved by a simple competitive mechanism between the two methods. The horizontal crossover represents the arithmetic crossover between two individuals randomly selected from current population. All individuals should be paired prior to the horizontal crossover. The detailed description of horizontal crossover between the two vector $X_{i_1,D}$ and $X_{i_2,D}$ is presented as follows.

$$NX_{i_1,D} = rand \cdot X_{i_1,G} + (1 - rand) \cdot X_{i_2,G} + rands \cdot X_{i_1,G} - X_{i_2,G}) \tag{35}$$

$$NX_{i_2,D} = rand \cdot X_{i_2,G} + (1 - rand) \cdot X_{i_1,G} + rands \cdot (X_{i_2,G} - X_{i_1,G}) \tag{36}$$

where *rands* denotes the number randomly distributed within $[-1, 1]$; *rand* denotes the number randomly distributed within $[0, 1]$; $X_{i_1}$ and $X_{i_2,G}$ denote the two individual randomly selected from current population, and $NX_{i_1}$ and $NX_{i_2}$ denote the two offspring generated by arithmetic crossover between two current individuals.

Different from horizontal crossover, the vertical crossover is to perform arithmetic crossover between different dimensions of a certain stagnant individual. The issue of local optimum of dimensions can be prevented by using the vertical crossover. The vertical crossover between $D_1$ and $D_1$ of a stagnant individual is performed according to Eq. (37).

$$NX_{i,D_1} = rand \cdot X(i, D_1) + (1 - rand) \cdot X(i, D_2) \tag{37}$$

where $D_1$ and $D_2$ denote two different dimensions randomly selected from [1,D]. $NX_{i,D1}$ is the offspring generated by $D_1$ and $D_2$ using vertical crossover. By this method, the individuals will not lose information about outstanding dimensions, and make full

**Table 1**
Recommended parameter settings of all algorithms.

| Algorithm | Year | Parameter settings |
|---|---|---|
| Our algorithm | – | $F = 0.5$, $CR = 0.8$, $p = 0.11$, $r^{rac} = 1.4$, $PS = 25 \cdot lnD \cdot \sqrt{D} \sim 4$, $H = 4$, $nfe_{\max} = D \cdot 10000$, $G = 1$, $n = 2 \cdot D$, $m = 1 \sim 3$ |
| LSHADE [13] | 2014 | $\mu_F = 0.5$, $\mu_{CR} = 0.5$, $F$ and $CR$ same as JADE, $PS = 18 \cdot D \sim 4$, $r^{rac} = 2.6$, $p = 0.11$, $H = 6$ |
| jSO [39] | 2017 | $F$ and $CR$, $r^{rac}$ same as iLSHADE, $\mu_F = 0.3$, $\mu_{CR} = 0.8$, $PS = 25 \cdot lnD \cdot \sqrt{D} \sim 4$, $p = 0.25 \sim 0.125$, $H = 5$ |
| Hip-DE [40] | 2021 | $\mu_F = 0.3$, $\mu_{CR} = 0.8$, $F$ and $CR$ same as LSHADE, $p = 0.11$, $PS = 25 \cdot lnD \cdot \sqrt{D} \sim 4$, $r^{rac,A} = 1.6$, $r^{rac,B} = 3$, $k = 4$ |
| CS-DE [19] | 2021 | $\mu_F = 0.6$, $\mu_{CR} = 0.8$, $F$ and $CR$ same as LSHADE, $p = 0.25 \sim 0.05$, $r^{rac,A} = 1.6$, $r^{rac,B} = 5$, $T_0 = \frac{Gen_{max}}{2}$, $PS = 25 \cdot lnD \cdot \sqrt{D} \sim 6$ |
| TDE [41] | 2022 | $\mu_F = 0.3$, $\mu_{CR} = 0.5$, $F$ and $CR$ same as JADE, $PS = 25 \cdot lnD \cdot \sqrt{D} \sim 4$, $n = 2 \cdot D$ |

use of the useful information in each dimension and thus have the opportunity to jump out of the local optimum. Therefore, the population diversity will also be enhanced, thus improving the overall optimization accuracy.

Some new individuals, generated by the strategy above, will be prescreened based on probability distribution and ranked according to their fitness value. Through the individual screening strategy, the individuals with poor fitness values have a higher probability of being eliminated, by which the local search ability is improved. The new individuals are generated according to Eq. (38).

$$X_i^{G+1} = \begin{cases} X_i^G - rand \cdot (X_{i_1}^G - X_{best}^G), & if\ \frac{rank_i}{PS} > 0.5 \\ X_i^G + rand \cdot (X_{i_1}^G - X_{t_2}^G), & otherwise \end{cases} \tag{38}$$

Algorithm 4 presents the population diversity mechanism, and Algorithm 5 contains our algorithm's pseudocode.

**Algorithm 4** Population diversity mechanism

**for** $i = 1$ **to** $PS$ **do**
  **if** $counter(i) > 2 \cdot D$ && $VOL < \xi$ **then**
    **if** $rand > 0.8$ **then**
      Calculate $X_{i_1,D}$ and $X_{i_2,D}$ according to Eq. (35);
    **else**
      Calculate $X_{i,D_1}$ according to Eq. (36);
    **end if**
    **if** $\frac{rank_i}{PS} > 0.5$ **then**
      Calculate $X_i^{G+1}$ according to Eq. (37);
    **end if**
  **end if**
**end for**

**Algorithm 5** Pseudo code of our zDE algorithm

**for** $i = 1$ **to** $PS$ **do**
  $X_{i,G} = X_i$; Calculate the fitness value $f(X_{i,G})$;
**end for**
Find the global best $X_{gbest,G}$ and fitness value $f(X_{gbest,G})$;
**while** $nfe < nfe_{\max}$ **do**
  $S_{CR} = \emptyset$, $S_F = \emptyset$;
  **for** $i = 1$ **to** $PS$ **do**
    $r_i$ = Select from $[1, H]$ randomly;
    **if** $\mu_{CR,r_i} = \perp$, $CR_{i,G} = 0$. Otherwise **then**
    **end if**
    Adjust parameters $F$ and $CR$;
    (Eq. (26) and Eq. (27) and Eq. (28));
    Execute Algorithm 1;
  **end for**
  **for** $i = 1$ **to** $PS$ **do**
    **if** $f(U_{i,G} \le f(X_{i,G})$ **then**
      $X_{i,G+1} = U_{i,G}$;
    **else**
      $X_{i,G+1} = X_{i,G}$;
    **end if**
    **if** $f(U_{i,G} < f(X_{i,G})$ **then**
      $X_{i,G} \to A$;
      $CR_{i,G} \to S_{CR}$, $F_{i,G} \to S_F$;
    **end if**
  **end for**
  It will be randomly removed from the archive,
  making the population size save $A$;
  Update memories $\mu_{CR}$ Eq. (31) and $\mu_F$ Eq. (32);
  Execute Algorithm 3;
  Calculate Eq. (34);
  Execute Algorithm 2;
  Execute Algorithm 4;
  It will be randomly removed from the archive,
  making the population size save $A$;
  Calculate $PS_{G+1}$ according to Eq. (5);
  $G = G + 1$;
**end while**

## 4. Experiment analysis

In this section, the performance of our algorithm is analyzed by comparing with several state-of-art DE variants and non-DE variants. All experiments are performed using on a PC with Inter Core i7-4570 CPU @3.20 GHz on Win10 and all algorithms are run in Matlab 2021a version. The details of the parameter settings of these algorithms taken into comparison are given in Table 1, and we use 88 benchmark functions from CEC2013, CEC2014, and CEC2017 test suites for real-parameter single-objective optimization in order to avoiding the over-fitting problems occurred in using only one test suite. The maximum number of function evaluations ($nfe_{\max}$) for all algorithms is set to $10,000 \cdot D$, and each problem is run independently 51 times. Wilcoxon signed rank test with the significant level $\alpha = 0.05$ is used in the performance evaluation for each benchmark, and the comparison results are listed behind "Mean/Std"(mean and standard deviation) of the total 51 runs.

### 4.1. Optimization accuracy

The paper aimed at improving the overall performance of DE on real-parameter single-objective optimization, therefore we focus more attention on the state-of-the-art DE variants comparison. There are total six DE variants including the LSHADE algorithm [13], the jSO algorithm [39], the Hip-DE algorithm [40], the CS-DE algorithm [19], the TDE algorithm [41] and our zDE algorithm. The LSHADE algorithm is the winner in CEC2014 competition, the jSO algorithm is the winner of the DE branches in CEC2017 competition. The Hip-DE algorithm, the CS-DE algorithm and the TDE algorithm are three recently published state-of-the-art DE variants that are superior to LSHADE and jSO. The results

**Table 2**
Comparison between LSHADE, jSO, Hip-DE, CS-DE, TDE and our algorithm under CEC2013 on 10D optimization.

| DE Variants NO. | LSHADE Mean/Std | jSO Mean/Std | Hip-DE Mean/Std | CS-DE Mean/Std | TDE Mean/Std | our algorithm Mean/Std |
|---|---|---|---|---|---|---|
| $f_{a_1}$ | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00 |
| $f_{a_2}$ | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00 |
| $f_{a_3}$ | 6.9958E−03/2.1430e−02(≈) | 1.3992E−03/9.9920e−03(>) | 1.3992E−03/9.9920e−03(>) | 2.7983E−03/1.3989e−02(>) | 2.7983E−03/1.3989e−02(>) | 6.9958E−03/2.1430e−02 |
| $f_{a_4}$ | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00 |
| $f_{a_5}$ | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00 |
| $f_{a_6}$ | 4.2328E+00/4.9081e+00(<) | 1.9240E−01/1.3740e+00(>) | 3.8480E+00/4.8384e+00(<) | 3.8480E−01/1.9236e+00(>) | **0.0000E+00/0.0000e+00(>)** | 2.2724E+00/3.9679e+00 |
| $f_{a_7}$ | 5.1012E−06/1.1493e−05(<) | 1.2811E−05/3.5711e−05(<) | 5.8196E−06/1.0527e−05(<) | 1.6965E−05/3.5103e−05(<) | 2.2774E−05/4.5620e−05(<) | **3.7784E−06/1.1114e−05** |
| $f_{a_8}$ | 2.0234E+01/1.6500e−01(<) | 2.0338E+01/6.9219e−02(<) | 2.0190E+01/1.6379e−01(<) | 2.0105E+01/1.2731e−01(>) | **2.0102E+01/1.3018e−01(>)** | 2.0188E+01/1.4134e−01 |
| $f_{a_9}$ | 2.6146E+00/1.4096e+00(<) | 5.4580E−01/8.0453e−01(<) | 5.2625E−01/7.5240e−01(<) | 2.6945E+00/1.4676e+00(<) | **3.8490E−01/5.2213e−01(>)** | 4.0425E−01/5.8873e−01 |
| $f_{a_{10}}$ | 3.1887E−03/5.4471e−03(>) | 9.6684E−04/2.9617e−03(>) | 1.9800E−03/5.9716e−03(>) | **1.4503E−04/1.0357e−03(>)** | 8.1592E−03/1.3726e−02(>) | 2.1240E−02/1.6709e−02 |
| $f_{a_{11}}$ | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00 |
| $f_{a_{12}}$ | 2.3714E+00/8.8618e−01(<) | 2.2631E+00/8.6924e−01(<) | **1.6059E+00/7.1994e−01(>)** | 2.2319E+00/9.0336e−01(<) | 3.3434E+00/2.0305e+00(<) | 2.0946E+00/1.1780e+00 |
| $f_{a_{13}}$ | 1.6942E+00/1.1617e+00(<) | 2.1161E+00/1.0569e+00(<) | 1.4554E+00/6.0909e−01(<) | 2.0630E+00/8.7898e−01(<) | 3.0129E+00/2.9641e+00(<) | **1.3347E+00/5.9243e−01** |
| $f_{a_{14}}$ | 2.2043E−02/3.7103e−02(<) | 4.0944E−02/5.5231e−02(<) | 7.3476E−03/2.0322e−02(>) | **2.4492E−03/1.2244e−02(>)** | 8.5778E−03/2.7983e−02(>) | 1.5920E−02/3.0194e−02 |
| $f_{a_{15}}$ | 3.0936E+02/1.0250e+02(<) | 2.7873E+02/9.6577e+01(<) | 3.1918E+02/1.5903e+02(<) | 3.1562E+02/1.1318e+02(<) | 4.6118E+02/2.0257e+02(<) | **2.6668E+02/8.8805e+01** |
| $f_{a_{16}}$ | 2.4399E−01/1.3480e−01(<) | 1.0699E+00/1.9135e−01(<) | 1.1344E−01/1.2949e−01(>) | 1.8332E−01/1.2907e−01(<) | **9.5421E−02/1.0864e−01(>)** | 1.6481E−01/1.4488e−01 |
| $f_{a_{17}}$ | 1.0122E+01/1.5461e−14(≈) | 1.0124E+01/2.9996e−03(<) | 1.0122E+01/2.3613e−14(≈) | 1.0122E+01/1.7940e−15(≈) | 1.0122E+01/7.9877e−15(≈) | 1.0122E+01/1.1223e−14 |
| $f_{a_{18}}$ | **1.3780E+01/9.3860e−01(>)** | 1.3904E+01/1.3513e+00(>) | 1.3952E+01/1.4635e+00(>) | 1.3901E+01/1.0517e+00(>) | 1.5021E+01/1.8920e+00(<) | 1.4249E+01/2.0562e+00 |
| $f_{a_{19}}$ | **2.1888E−01/3.5891e−02(>)** | 2.5004E−01/5.4417e−02(<) | 2.2487E−01/3.7121e−02(>) | 2.2655E−01/4.1659e−02(>) | 2.3191E−01/5.4409e−02(>) | 2.3223E−01/3.5999e−02 |
| $f_{a_{20}}$ | 1.9590E+00/3.3882e−01(<) | 1.6501E+00/2.9398e−01(<) | 1.4847E+00/3.3121e−01(<) | 1.8988E+00/3.9468e−01(<) | 1.6887E+00/4.1582e−01(<) | **1.3120E+00/3.6342e−01** |
| $f_{a_{21}}$ | 4.0019E+02/0.0000e+00(≈) | 3.9627E+02/2.8033e+01(>) | 4.0019E+02/0.0000e+00(≈) | **3.9234E+02/3.9246e+01(>)** | 4.0019E+02/0.0000e+00(≈) | 4.0019E+02/0.0000e+00 |
| $f_{a_{22}}$ | 7.3504E+00/1.3629e+01(<) | 8.0761E+00/1.3948e+01(<) | 3.8246E+00/1.4344e+01(<) | **1.6485E+00/2.5176e+00(>)** | 3.8934E+00/1.4266e+01(<) | 3.8159E+00/4.3983e+00 |
| $f_{a_{23}}$ | 2.6995E+02/1.4681e+02(<) | 1.9197E+02/1.1895e+02(<) | 2.2225E+02/1.3975e+02(<) | 3.6090E+02/1.2551e+02(<) | 3.6089E+02/1.8125e+02(<) | **1.7690E+02/1.0997e+02** |
| $f_{a_{24}}$ | 2.0312E+02/3.3689e+00(<) | 2.0065E+02/1.9811e+00(<) | 2.0000E+02/6.9040e−06(≈) | 1.9840E+02/1.1419e+01(>) | **1.9810E+02/1.3548e+01(>)** | 2.0000E+02/0.0000e+00 |
| $f_{a_{25}}$ | 2.0103E+02/2.0273e+00(<) | 2.0009E+02/6.3574e−01(<) | 2.0000E+02/1.1191e−04(≈) | **1.9881E+02/8.5160e+00(>)** | 2.0000E+02/7.8830e−06(≈) | 2.0000E+02/0.0000e+00 |
| $f_{a_{26}}$ | 1.3171E+02/4.4867e+01(<) | **1.0189E+02/8.9557e−01(>)** | 1.0535E+02/1.9328e+01(>) | 1.0264E+02/9.9097e−01(>) | 1.0748E+02/1.9021e+01(<) | 1.0590E+02/1.9252e+01 |
| $f_{a_{27}}$ | 3.0000E+02/0.0000e+00(≈) | 3.0000E+02/0.0000e+00(≈) | 3.0000E+02/0.0000e+00(≈) | 3.0000E+02/4.5475e−14(≈) | 3.0000E+02/1.7747e−11(≈) | 3.0000E+02/0.0000e+00 |
| $f_{a_{28}}$ | 3.0000E+02/0.0000e+00(≈) | 3.0000E+02/0.0000e+00(≈) | 3.0000E+02/0.0000e+00(≈) | 3.0000E+02/3.2155e−14(≈) | 3.0000E+02/7.1902e−14(≈) | 3.0000E+02/0.0000e+00 |
| >/≈/< | 3/10/15 | 6/7/15 | 8/11/9 | 12/8/8 | 9/10/9 | –/–/– |

**Table 3**
Comparison between LSHADE, jSO, Hip-DE, CS-DE, TDE and our algorithm under CEC2013 on 30D optimization.

| DE Variants NO. | LSHADE Mean/Std | jSO Mean/Std | Hip-DE Mean/Std | CS-DE Mean/Std | TDE Mean/Std | our algorithm Mean/Std |
|---|---|---|---|---|---|---|
| $f_{a_1}$ | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00(≈) | 0.0000E+00/0.0000e+00 |
| $f_{a_2}$ | 3.1654E−13/2.5756e−13(<) | 1.3390E−10/2.5586e−10(<) | 6.6429E−13/2.4539e−12(<) | 1.8725E−13/1.0863e−13(<) | 2.5858E−13/1.3657e−13(<) | **0.0000E+00/0.0000e+00** |
| $f_{a_3}$ | 1.2898E−01/6.9024e−01(<) | **8.3236E−09/5.4693e−08(>)** | 5.9494E−01/2.5762e+00(<) | 1.1415E−02/7.1155e−02(<) | 1.2661E−06/9.0418e−06(>) | 1.0684E−05/7.6246e−05 |
| $f_{a_4}$ | 5.7958E−14/1.0008e−13(<) | 1.5961E−12/9.2470e−13(<) | 2.2292E−14/6.8286e−14(<) | 6.2416E−14/1.0248e−13(<) | 3.1208E−14/7.9022e−14(<) | **0.0000E+00/0.0000e+00** |
| $f_{a_5}$ | 1.1369E−13/0.0000e+00(<) | 1.1146E−13/1.5919e−14(<) | 1.1592E−13/1.5919e−14(<) | 9.1395E−14/4.5586e−14(<) | 8.6937E−14/4.8704e−14(<) | **7.3562E−14/5.4870e−14** |
| $f_{a_6}$ | 8.3077E−09/4.4229e−08(<) | 7.2368E−10/2.4841e−09(<) | 6.4917E−11/4.4844e−10(<) | 5.6777E−12/3.8438e−11(<) | 1.0700E−13/4.7758e−14(<) | **2.8979E−14/5.0038e−14** |
| $f_{a_7}$ | 6.0879E−01/4.4764e−01(<) | 1.8790E−02/4.4427e−02(<) | 1.2862E−01/1.6020e−01(<) | 5.3299E−02/6.6670e−02(<) | 2.3277E−02/3.8942e−02(<) | **1.2410E−02/3.6481e−02** |
| $f_{a_8}$ | 2.0842E+01/1.4562e−01(<) | 2.0937E+01/5.9054e−02(<) | 2.0713E+01/1.7341e−01(>) | **2.0630E+01/1.6798e−01(>)** | 2.0665E+01/1.9722e−01(>) | 2.0838E+01/1.4092e−01 |
| $f_{a_9}$ | 2.6654E+01/1.4454e+00(<) | 2.3500E+01/3.5501e+00(<) | 2.5445E+01/2.1807e+00(<) | 2.7093E+01/1.4561e+00(<) | 2.4266E+01/4.7384e+00(<) | **1.5185E+01/8.9186e+00** |
| $f_{a_{10}}$ | 5.8008E−04/2.0082e−03(<) | 0.0000E+00/0.0000e+00(≈) | 1.4502E−04/1.0357e−03(<) | 0.0000E+00/0.0000e+00(≈) | 1.4502E−04/1.0357e−03(<) | 0.0000E+00/0.0000e+00 |
| $f_{a_{11}}$ | 7.4677E−14/4.1756e−14(>) | 1.6161E−13/5.9480e−14(<) | 1.8948E−14/2.7063e−14(>) | **1.6719E−14/2.6158e−14(>)** | 4.3468E−14/5.1585e−14(>) | 1.0366E−13/4.6467e−14 |
| $f_{a_{12}}$ | **5.5397E+00/1.7675e+00(>)** | 9.0362E+00/2.5876e+00(<) | 5.9854E+00/1.2339e+00(>) | 9.9548E+00/2.0208e+00(<) | 7.2441E+00/1.7476e+00(<) | 6.1411E+00/1.6367e+00 |
| $f_{a_{13}}$ | 6.0977E+00/3.2031e+00(>) | 9.9228E+00/4.7778e+00(<) | **5.2605E+00/2.5633e+00(>)** | 1.4768E+01/4.2273e+00(<) | 9.8516E+00/6.3665e+00(<) | 7.3124E+00/2.4364e+00 |
| $f_{a_{14}}$ | 2.4901E−02/2.3191e−02(>) | 1.1759E+01/5.5189e+00(<) | 2.3677E−02/2.6668e−02(>) | 1.5512E−02/1.9901e−02(>) | **8.5726E−03/1.1906e−02(>)** | 3.2249E−02/2.3300e−02 |
| $f_{a_{15}}$ | 2.6611E+03/2.7629e+02(<) | 2.7096E+03/2.9775e+02(<) | 2.6543E+03/3.0470e+02(<) | 2.7794E+03/2.5734e+02(<) | 2.8121E+03/3.1425e+02(<) | **2.4791E+03/3.0690e+02** |
| $f_{a_{16}}$ | 7.5627E−01/1.4254e−01(<) | 2.2904E+00/3.1035e−01(<) | **3.1942E−01/2.5024e−01(>)** | 7.3038E−01/1.6981e−01(<) | 3.9741E−01/3.8703e−01(>) | 6.1995E−01/3.7065e−01 |
| $f_{a_{17}}$ | 3.0434E+01/9.4299e−07(≈) | 3.0673E+01/1.1707e−01(<) | 3.0434E+01/5.4965e−14(≈) | 3.0434E+01/5.7187e−14(≈) | 3.0434E+01/5.2098e−14(≈) | 3.0434E+01/9.4299e−07 |
| $f_{a_{18}}$ | 5.1891E+01/3.5319e+00(>) | 5.6899E+01/4.7627e+00(<) | **4.9676E+01/3.2152e+00(>)** | 5.4813E+01/3.7372e+00(<) | 5.6196E+01/6.1281e+00(<) | 5.4014E+01/3.5512e+00 |
| $f_{a_{19}}$ | 1.1673E+00/8.6400e−02(>) | 1.2583E+00/9.2718e−02(<) | 1.1468E+00/9.1206e−02(>) | 1.1746E+00/1.0061e−01(<) | **1.1435E+00/1.1882e−01(>)** | 1.1682E+00/8.7295e−02 |
| $f_{a_{20}}$ | 1.0739E+01/1.7551e+00(<) | 9.5726E+00/3.9013e−01(<) | 9.0488E+00/3.5762e−01(<) | 9.9488E+00/1.3212e+00(<) | 9.1004E+00/5.6628e−01(<) | **9.0189E+00/3.8122e−01** |
| $f_{a_{21}}$ | 2.9608E+02/1.9604e+01(<) | 3.0342E+02/4.9361e+01(<) | 3.0085E+02/2.4726e+01(<) | 2.9804E+02/1.4003e+01(<) | 3.0367E+02/3.1788e+01(<) | **2.9216E+02/2.7152e+01** |
| $f_{a_{22}}$ | 1.0831E+02/2.5461e+00(<) | 1.1945E+02/3.8193e+00(<) | 1.0590E+02/3.7285e−01(>) | 1.0592E+02/3.9687e−01(>) | 1.0590E+02/9.1664e−01(>) | 1.0647E+02/1.6357e+00 |
| $f_{a_{23}}$ | 2.5258E+03/2.8207e+02(<) | 2.5135E+03/3.2438e+02(<) | 2.5666E+03/3.0098e+02(<) | 2.8235E+03/2.8588e+02(<) | 2.7251E+03/3.1036e+02(<) | **2.4302E+03/2.9568e+02** |
| $f_{a_{24}}$ | 2.0047E+02/1.1076e+00(<) | 2.0004E+02/4.9193e−02(<) | 2.0001E+02/2.4049e−02(>) | 2.0001E+02/2.2333e−02(>) | **2.0000E+02/7.3483e−03(>)** | 2.0002E+02/3.4659e−02 |
| $f_{a_{25}}$ | 2.4096E+02/6.9199e+00(<) | 2.3860E+02/4.7024e+00(<) | 2.2045E+02/2.3039e+01(<) | 2.0546E+02/1.5246e+01(>) | **2.0339E+02/1.1743e+01(>)** | 2.1063E+02/1.8434e+01 |
| $f_{a_{26}}$ | 2.0000E+02/0.0000e+00(≈) | 2.0000E+02/1.4171e−13(≈) | 2.0000E+02/3.2155e−14(≈) | 2.0000E+02/6.4311e−14(≈) | 2.0000E+02/7.8765e−14(≈) | 2.0000E+02/6.4311e−14 |
| $f_{a_{27}}$ | 3.0136E+02/3.0731e+00(<) | 3.0092E+02/1.1319e+00(>) | 3.0034E+02/6.8807e−01(>) | 3.0095E+02/1.4037e+00(>) | **3.0005E+02/1.7520e−01(>)** | 3.0098E+02/2.0117e+00 |
| $f_{a_{28}}$ | 3.0000E+02/0.0000e+00(≈) | 3.0000E+02/6.4311e−14(≈) | 3.0000E+02/0.0000e+00(≈) | 3.0000E+02/2.0017e−13(≈) | 3.0000E+02/1.7831e−13(≈) | 3.0000E+02/9.0949e−14 |
| >/≈/< | 6/4/18 | 2/4/22 | 11/4/13 | 7/5/16 | 10/4/14 | –/–/– |

of these DE variants on 10D, 30D, 50D and 100D comparison under the CEC2013, CEC2014 and CEC2017 test suites are given in Tables 2–11. A summary of the comparison results are also given below in Table 12 for an overall perception of the comparison. The symbols "$>$", "$\approx$", "$<$" behind the "Mean/Std", denote "better performance", "similar performance" and "worse performance", respectively. The fitness error smaller than eps=2.2204E−16 is regarded as 0.

The following conclusions can be drawn from the comparison results: (1) With regard to 10D, 30D, and 50D optimizations, our algorithm's performance is competitive with that of all rivals. (2) For all test suites, our algorithm outperforms jSO, Hip-DE, and CS-DE. It also outperforms jSO on the CEC2014 and CEC2017. It also exhibits much better performance over LSHADE. (3) Our algorithm is able to achieve significant performance improvements under the CEC2014 and CEC2017 test suites compared to the CEC2013 test suite. (4) Our algorithm achieves better performance on 30D and 50D optimization compared to 10D optimization. The summary table shows that our algorithm obtained 196 better performances compared with LSHADE, 206 better performances compared with jSO, 178 performance improvements in comparison with Hip-DE and TDE; obtained 185 performance improvements compared with CS-DE for the total 294 results in the comparison.

For the comparison in the non-DE variants, five recently proposed state-of-the-art algorithms including the PSO-sono algorithm [44], the E-QUATRE algorithm [45], the EBOwithCMAR algorithm [46], the HSES algorithm [47] and the EA4eig algorithm [48] are taken into consideration under the $f_{c_1} - f_{c_{30}}$ benchmarks from CEC2017 test suite on 10D, 30D and 50D optimization. The PSO-sono algorithm and the E-QUATRE algorithm are the recent proposed state-of-the-art PSO variant and QUATRE variant. The EBOwithCMAR algorithm is the winner algorithm among all the competitors in the CEC2017 competition, and the HSES algorithm is the winner algorithm in the CEC2018 competition (the competition also employed the CEC2017 test suite). The EA4eig algorithm is the winner competitor in the CEC2022 competition. Obviously, these algorithms are all the excellent algorithms proposed recently and this is also the reason why we

**Table 4**
Comparison between LSHADE, jSO, Hip-DE, CS-DE, TDE and our algorithm under CEC2013 on 50D optimization.

| DE Variants NO. | LSHADE Mean/Std | jSO Mean/Std | Hip-DE Mean/Std | CS-DE Mean/Std | TDE Mean/Std | our algorithm Mean/Std |
|---|---|---|---|---|---|---|
| $f_{a_1}$ | 4.9041E−14/9.4449e−14(<) | 5.7958E−14/1.0008e−13(<) | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00* |
| $f_{a_2}$ | 6.7423E+02/8.6125e+02(<) | 4.0540E+01/7.2598e+01(<) | 2.1364E+03/2.2367e+03(<) | 2.1481E+02/2.7264e+02(<) | 1.6414E+02/3.2344e+02(<) | **9.9048E−08/4.0299e−07** |
| $f_{a_3}$ | 1.2671E+04/6.5125e+04(<) | **2.9443E+00/1.4549e+01(>)** | 2.2823E+03/1.5217e+04(<) | 1.6660E+02/6.6726e+02(<) | 1.3147E+02/5.7050e+02(>) | 1.6194E+02/7.8495e+02 |
| $f_{a_4}$ | 7.9086E−11/2.0827e−10(<) | 1.5703E−08/2.5171e−08(<) | 8.1498E−12/5.0062e−12(<) | 1.8145E−11/3.1130e−11(<) | 1.0549E−10/1.1466e−10(<) | **7.2225E−13/3.5138e−13** |
| $f_{a_5}$ | 1.4044E−13/4.8704e−14(<) | 1.4489E−13/5.1240e−14(<) | 2.2069E−13/3.5311e−14(<) | 1.1369E−13/0.0000e+00(<) | 1.1369E−13/0.0000e+00(<) | **1.0923E−13/2.2287e−14** |
| $f_{a_6}$ | 4.3447E+01/0.0000e+00(<) | 4.3447E+01/1.6078e−14(<) | 4.3447E+01/2.7019e−14(<) | 4.3447E+01/1.6078e−14(<) | 4.3436E+01/4.0956e−02(<) | **4.3245E+01/4.3935e−02** |
| $f_{a_7}$ | 2.1970E+00/1.5443e+00(<) | **1.1096E−01/9.8286e−02(>)** | 5.9187E−01/5.3267e−01(<) | 3.4885E−01/5.6970e−01(<) | 2.2494E−01/3.8925e−01(>) | 2.3719E−01/9.8174e−02 |
| $f_{a_8}$ | 2.1081E+01/1.0125e−01(<) | 2.1137E+01/3.6389e−02(<) | 2.0889E+01/1.8848e−01(>) | **2.0853E+01/1.7400e−01(>)** | 2.0865E+01/1.5105e−01(>) | 2.1004E+01/1.2504e−01 |
| $f_{a_9}$ | 5.3181E+01/1.7825e+00(<) | 4.7390E+01/6.0204e+00(<) | 4.0702E+01/1.2230e+01(<) | 5.3157E+01/2.1307e+00(<) | 4.0116E+01/1.4452e+01(<) | **1.3645E+01/3.0788e+00** |
| $f_{a_{10}}$ | 9.9034E−03/8.2053e−03(<) | 1.3536E−03/2.9715e−03(<) | 8.3560E−03/8.5167e−03(<) | 5.7014E−03/6.8174e−03(<) | 4.3980E−03/5.2472e−03(<) | **1.1146E−15/7.9597e−15** |
| $f_{a_{11}}$ | 6.9846E−11/8.9419e−11(<) | 6.6023E−09/1.0054e−08(<) | 2.0731E−13/9.2934e−14(>) | **1.3598E−13/1.1316e−13(>)** | 1.9505E−13/1.4028e−13(>) | 5.8181E−13/2.5129e−13 |
| $f_{a_{12}}$ | 1.4404E+01/2.5423e+00(<) | 1.8127E+01/4.0285e+00(<) | 1.4549E+01/2.9729e+00(<) | 2.2363E+01/2.7168e+00(<) | 1.7425E+01/2.9591e+00(<) | **1.1889E+01/2.2835e+00** |
| $f_{a_{13}}$ | 2.1767E+01/8.3792e+00(<) | 2.6687E+01/1.2368e+01(<) | 2.5718E+01/6.9772e+00(<) | 4.6742E+01/1.0507e+01(<) | 3.5563E+01/1.0582e+01(<) | **2.0435E+01/7.4469e+00** |
| $f_{a_{14}}$ | 2.2927E−01/1.1510e−01(<) | 6.2689E+01/1.6273e+01(<) | 1.1777E−01/3.2191e−02(<) | 5.0461E−02/2.2451e−02(>) | **3.8099E−02/2.8606e−02(>)** | 8.8350E−02/3.6629e−02 |
| $f_{a_{15}}$ | 6.3644E+03/4.5097e+02(<) | 6.3182E+03/3.9725e+02(<) | 6.3772E+03/4.0331e+02(<) | 6.3090E+03/3.7611e+02(<) | 6.6118E+03/4.5645e+02(<) | **6.1306E+03/4.2404e+02** |
| $f_{a_{16}}$ | 1.1599E+00/2.0453e−01(<) | 3.1041E+00/5.1843e−01(<) | **6.1986E−01/4.3286e−01(>)** | 1.1383E+00/2.2874e−01(<) | 7.5152E−01/5.4328e−01(>) | 1.0766E+00/5.1515e−01 |
| $f_{a_{17}}$ | 5.0788E+01/4.4648e−03(<) | 5.2455E+01/3.4576e−01(<) | *5.0786E+01/2.6464e−03(≈)* | *5.0786E+01/2.5963e−05(≈)* | *5.0786E+01/1.2247e−08(≈)* | *5.0786E+01/8.9553e−06* |
| $f_{a_{18}}$ | 1.0282E+02/6.5621e+00(<) | 1.1094E+02/7.7117e+00(<) | **9.9416E+01/4.6798e+00(>)** | 1.0431E+02/6.2374e+00(<) | 1.0722E+02/8.2002e+00(<) | 1.0260E+02/5.0693e+00 |
| $f_{a_{19}}$ | 2.5539E+00/1.3281e−01(<) | 2.6398E+00/1.8963e−01(<) | 2.4978E+00/1.4014e−01(<) | 2.3882E+00/1.2657e−01(>) | **2.3874E+00/1.1786e−01(>)** | 2.4283E+00/1.4653e−01 |
| $f_{a_{20}}$ | 1.8164E+01/5.1610e−01(<) | 1.8836E+01/4.8032e−01(<) | 1.7773E+01/4.4979e−01(<) | 1.8035E+01/4.5007e−01(<) | 1.7698E+01/5.7421e−01(<) | **1.7387E+01/6.3206e−01** |
| $f_{a_{21}}$ | 7.6054E+02/4.5472e+02(>) | **7.6004E+02/3.9949e+02(>)** | 8.1606E+02/4.2597e+02(<) | 1.0038E+03/2.8090e+02(<) | 8.2854E+02/4.1678e+02(<) | 7.8245E+02/4.1122e+02 |
| $f_{a_{22}}$ | 1.3713E+01/1.3369e+00(<) | 5.5648E+01/1.0764e+01(<) | 1.1718E+01/5.6189e−01(>) | 1.1596E+01/6.3990e−01(>) | **1.1479E+01/6.5472e−01(>)** | 1.2047E+01/1.1810e+00 |
| $f_{a_{23}}$ | 5.8224E+03/3.3537e+02(<) | **5.5551E+03/5.6912e+02(>)** | 6.1175E+03/4.6767e+02(<) | 6.2035E+03/4.4914e+02(<) | 6.3212E+03/3.9179e+02(<) | 5.6104E+03/5.0318e+02 |
| $f_{a_{24}}$ | 2.1102E+02/4.9491e+00(<) | 2.0055E+02/6.3083e−01(>) | 2.0116E+02/1.0748e+00(>) | 2.0058E+02/6.7615e−01(>) | **2.0032E+02/4.3565e−01(>)** | 2.0198E+02/7.7569e−01 |
| $f_{a_{25}}$ | 2.7714E+02/6.1710e+00(>) | **2.7652E+02/7.5487e+00(>)** | 2.8497E+02/5.9786e+00(≈) | 2.8381E+02/6.0358e+00(>) | 2.8040E+02/6.0702e+00(>) | 2.8497E+02/6.2864e+00 |
| $f_{a_{26}}$ | 2.5410E+02/5.3639e+01(>) | **2.1214E+02/3.3582e+01(>)** | 2.5451E+02/5.1935e+01(>) | 2.3416E+02/4.8802e+01(>) | 2.4213E+02/5.1248e+01(>) | 2.7619E+02/4.7363e+01 |
| $f_{a_{27}}$ | 4.1463E+02/8.2186e+01(<) | 3.3410E+02/6.3233e+01(>) | 3.2085E+02/1.3853e+01(>) | 3.2158E+02/1.9016e+01(>) | **3.1317E+02/1.4972e+01(>)** | 3.3450E+02/1.1756e+01 |
| $f_{a_{28}}$ | *4.0000E+02/0.0000e+00(≈)* | *4.0000E+02/2.8433e−13(≈)* | *4.0000E+02/7.1902e−14(≈)* | *4.0000E+02/1.1482e−13(≈)* | *4.0000E+02/5.5695e−14(≈)* | *4.0000E+02/1.1482e−13* |
| >/≈/< | 3/1/24 | 8/1/19 | 8/4/16 | 9/3/16 | 12/3/13 | –/–/– |

**Table 5**
Comparison between LSHADE, jSO, Hip-DE, CS-DE, TDE and our algorithm under CEC2014 on 10D optimization.

| DE Variants NO. | LSHADE Mean/Std | jSO Mean/Std | Hip-DE Mean/Std | CS-DE Mean/Std | TDE Mean/Std | our algorithm Mean/Std |
|---|---|---|---|---|---|---|
| $f_{b_1}$ | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00* |
| $f_{b_2}$ | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00* |
| $f_{b_3}$ | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00* |
| $f_{b_4}$ | 3.1541E+01/9.9473e+00(<) | 3.0006E+01/1.2088e+01(<) | 3.0689E+01/1.1317e+01(<) | 2.3869E+01/1.6299e+01(<) | 2.3107E+01/1.6027e+01(<) | **6.8925E+00/2.9407e+00** |
| $f_{b_5}$ | 1.4893E+01/8.3835e+00(>) | 1.5594E+01/8.5565e+00(>) | 1.4699E+01/8.6864e+00(>) | **1.2506E+01/9.1458e+00(>)** | 1.3969E+01/9.0353e+00(>) | 1.6568E+01/7.3902e+00 |
| $f_{b_6}$ | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00* |
| $f_{b_7}$ | 2.1280E−03/5.9728e−03(>) | 5.3153E−04/2.2212e−03(>) | 4.3507E−04/1.7576e−03(>) | **2.0419E−08/1.0560e−07(>)** | 1.2671E−02/1.9251e−02(>) | 1.7872E−02/2.1544e−02 |
| $f_{b_8}$ | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00* |
| $f_{b_9}$ | 2.4399E+00/8.2825e−01(<) | 1.9119E+00/8.6380e−01(>) | **1.8555E+00/8.4424e−01(>)** | 2.3234E+00/8.3587e−01(<) | 3.0268E+00/1.7887e+00(<) | 1.9257E+00/1.1799e+00 |
| $f_{b_{10}}$ | 3.6738E−03/1.4841e−02(>) | 2.3409E−02/3.7307e−02(<) | 1.2246E−03/8.7454e−03(>) | **5.3500E−14/2.1613e−13(>)** | 9.7968E−03/2.6119e−02(≈) | 9.7968E−03/2.6119e−02 |
| $f_{b_{11}}$ | **2.6262E+01/3.2833e+01(>)** | 3.6423E+01/5.1459e+01(<) | 3.5211E+01/5.9586e+01(<) | 4.4941E+01/5.4585e+01(<) | 4.4976E+01/7.8166e+01(<) | 3.1573E+01/4.2315e+01 |
| $f_{b_{12}}$ | 6.8688E−02/1.8600e−02(<) | 1.7716E−01/1.3314e−01(<) | **5.0180E−02/3.5362e−02(>)** | 5.7134E−02/1.4993e−02(>) | 7.1090E−02/6.1576e−02(<) | 6.2465E−02/4.8884e−02 |
| $f_{b_{13}}$ | 5.0926E−02/1.2680e−02(>) | 6.9408E−02/1.7535e−02(<) | **3.8281E−02/1.9252e−02(>)** | 4.7152E−02/1.5769e−02(>) | 4.4209E−02/1.7964e−02(>) | 5.0955E−02/1.4406e−02 |
| $f_{b_{14}}$ | 7.6585E−02/3.0235e−02(<) | **5.7618E−02/2.0221e−02(>)** | 9.3147E−02/3.3580e−02(<) | 7.9901E−02/2.2152e−02(<) | 9.3224E−02/3.4470e−02(<) | 5.9224E−02/2.4605e−02 |
| $f_{b_{15}}$ | 3.7523E−01/7.1027e−02(<) | 3.9010E−01/8.3116e−02(<) | **3.3934E−01/8.2343e−02(>)** | 3.9761E−01/6.6245e−02(<) | 4.2152E−01/1.2019e−01(<) | 3.7464E−01/6.3093e−02 |
| $f_{b_{16}}$ | 1.1545E+00/2.8465e−01(<) | 9.3216E−01/3.2523e−01(<) | 1.0054E+00/3.2978e−01(<) | 1.2600E+00/3.5501e−01(<) | 1.0528E+00/4.3370e−01(<) | **7.2690E−01/2.3230e−01** |
| $f_{b_{17}}$ | 9.8877E−01/9.0234e−01(<) | 1.3347E+00/2.0036e+00(<) | 1.2062E+00/1.0777e+00(<) | 1.6698E+00/2.2425e+00(<) | 2.9451E+00/4.5539e+00(<) | **6.5245E−01/6.1207e−01** |
| $f_{b_{18}}$ | 2.1746E−01/1.7456e−01(<) | 1.5675E−01/1.5412e−01(<) | 1.2709E−01/2.3341e−01(<) | **6.2460E−02/8.4539e−02(>)** | 1.6976E−01/3.5532e−01(<) | 8.3552E−02/1.7566e−01 |
| $f_{b_{19}}$ | 7.8554E−02/3.8712e−02(<) | **4.1196E−02/2.7005e−02(>)** | 7.5280E−02/5.5976e−02(<) | 5.7196E−02/4.3841e−02(<) | 5.4561E−02/4.3448e−02(<) | 4.1496E−02/1.7964e−02 |
| $f_{b_{20}}$ | **1.3303E−01/1.0672e−01(>)** | 2.8128E−01/1.9954e−01(<) | 2.4634E−01/1.9335e−01(<) | 1.9623E−01/1.4932e−01(<) | 1.6085E−01/1.7029e−01(>) | 1.6239E−01/1.7085e−01 |
| $f_{b_{21}}$ | 3.7363E−01/3.0179e−01(<) | 4.7898E−01/3.1123e−01(<) | 2.9136E−01/2.8283e−01(<) | **2.4836E−01/2.6289e−01(>)** | 5.2800E−01/2.8271e−01(<) | 2.5149E−01/2.6062e−01 |
| $f_{b_{22}}$ | **7.1247E−02/3.0817e−02(>)** | 2.9357E−01/1.9759e−01(<) | 8.9447E−02/3.3048e−02(>) | 8.1265E−02/2.8274e−02(>) | 9.3439E−02/5.5751e−02(>) | 1.2911E−01/6.7655e−02 |
| $f_{b_{23}}$ | 3.2946E+02/0.0000e+00(≈) | 3.2946E+02/2.8705e−13(≈) | 3.2946E+02/0.0000e+00(≈) | **3.2300E+02/4.6133e+01(>)** | 3.2946E+02/0.0000e+00(≈) | 3.2946E+02/0.0000e+00 |
| $f_{b_{24}}$ | 1.0745E+02/2.0556e+00(<) | 1.0728E+02/1.5645e+00(<) | **1.0522E+02/3.2771e+00(>)** | 1.0787E+02/1.2470e+00(<) | 1.0796E+02/3.3771e+00(<) | 1.0666E+02/3.4973e+00 |
| $f_{b_{25}}$ | 1.3992E+02/4.2483e+01(<) | 1.2534E+02/2.5099e+01(<) | 1.2733E+02/3.3534e+01(<) | **1.1988E+02/1.2672e+01(>)** | 1.2324E+02/2.4960e+01(<) | 1.2062E+02/1.6361e+01 |
| $f_{b_{26}}$ | 1.0005E+02/1.4936e−02(≈) | 1.0007E+02/1.8091e−02(<) | 1.0005E+02/2.2598e−02(≈) | 1.0005E+02/1.6190e−02(≈) | **1.0004E+02/1.4714e−02(>)** | 1.0005E+02/1.4891e−02 |
| $f_{b_{27}}$ | 6.7747E+01/1.2854e+02(>) | 5.5861E+01/1.2008e+02(>) | 5.9910E+01/1.1975e+02(>) | **7.4938E+00/4.1779e+01(>)** | 4.4503E+01/1.1996e+02(>) | 7.3419E+01/1.4111e+02 |
| $f_{b_{28}}$ | 3.8668E+02/3.9914e+01(>) | 3.7809E+02/2.7179e+01(>) | 3.9583E+02/5.3116e+01(<) | 3.8323E+02/4.6272e+01(>) | **3.7545E+02/4.0361e+01(>)** | 3.8910E+02/4.9190e+01 |
| $f_{b_{29}}$ | 2.2197E+02/4.1379e−01(<) | 2.2193E+02/4.2078e−01(<) | 2.2283E+02/2.6392e−01(<) | 2.2178E+02/1.3586e−01(<) | **2.2159E+02/1.2490e+00(>)** | 2.2160E+02/5.2018e−01 |
| $f_{b_{30}}$ | 4.6321E+02/5.2134e+00(>) | 4.6498E+02/9.5899e+00(>) | 4.6372E+02/2.6359e+00(>) | **4.6304E+02/1.0477e+00(>)** | 4.6635E+02/7.4522e+00(>) | 4.6915E+02/1.4554e+01 |
| >/≈/< | 10/7/13 | 8/6/16 | 11/7/12 | 13/6/11 | 10/7/13 | –/–/– |

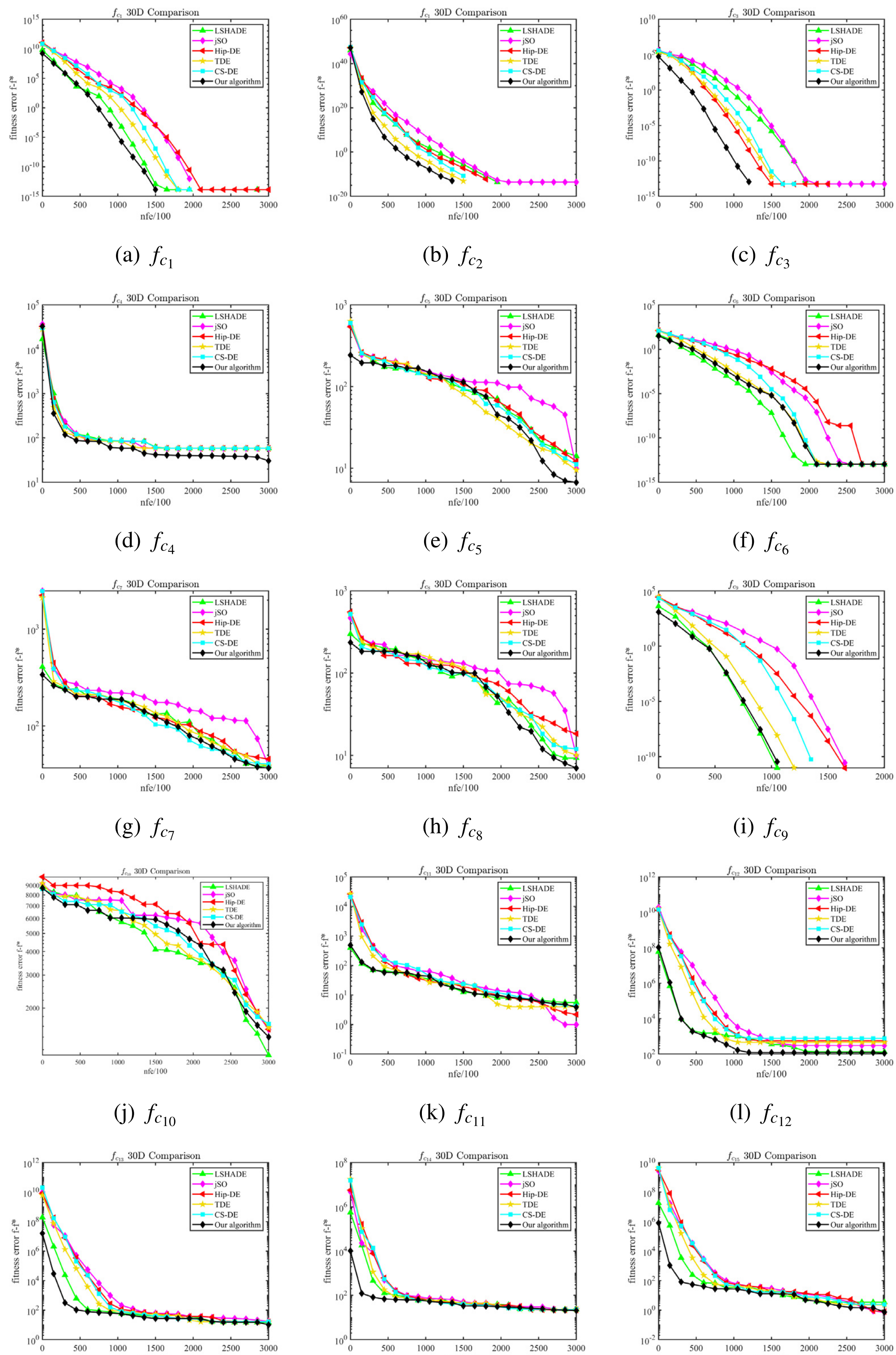
take these algorithms into consideration. The experiment results of the comparison are given in Table 13, Table 14 and Table 15, respectively. It can be seen from the results that the proposed algorithm has better performance over these state-of-the-art non-DE algorithms.

### 4.2. Convergence speed

For verifying the performance of our algorithm in terms of convergence speed, we compare the median value of 51 runs obtained by different algorithms under the CEC2017 test suite on 30D optimization. The convergence curve comparison are illustrated in Figs. 1–2. From these figures, we can conclude that our algorithm has obtained better convergence performance than comparing algorithms. In comparison with LSHADE, our algorithm obtains better performance on benchmarks $f_{c_1} - f_{c_4}$, $f_{c_6} - f_{c_8}$, $f_{c_{11}} - f_{c_{12}}$, $f_{c_{14}} - f_{c_{16}}$, $f_{c_{18}}$, $f_{c_{20}}$, $f_{c_{22}} - f_{c_{27}}$ and $f_{c_{29}} - f_{c_{30}}$; In comparison with jSO, our algorithm obtains better performance on benchmarks $f_{c_1} - f_{c_{10}}$, $f_{c_{12}} - f_{c_{15}}$ and $f_{c_{18}} - f_{c_{30}}$; Comparing with Hip-DE, our algorithm performs better on $f_{c_1} - f_{c_5}$, $f_{c_{11}} - f_{c_{30}}$; Comparing with CS-DE, our algorithm performs better on $f_{c_1} - f_{c_{27}}$, $f_{c_{29}} - f_{c_{30}}$; Comparing with TDE, our algorithm performs better on $f_{c_1} - f_{c_{23}}$ and $f_{c_{25}} - f_{c_{30}}$. The summary of the comparison results is given in Table 16, and we also summarize the best-performing benchmark of each algorithm in Table 17. From the results we can conclude that our algorithm shows strong competitiveness in terms of convergence speed as well.

### 4.3. The weighting strategy in parameter control

Here in this part, the parameter $m$ in the Minkowski distance based parameter control is discussed. From Eq. (29), we can see that the distance degrades to be Manhattan distance when the parameter, $m$, is set to 1 and it also can be degraded to Euclidean distance when $m$ is set to 2. Generally, in Minkowski distance, the choice of $m$ is flexible, and here we set six different settings of $m$ for further analysis. Table 18 presents the overall results of these different settings. Five ranges of $m$ including $m = 1 \sim 2$, $m = 1 \sim 3$, $m = 1 \sim 4$, $m = 1 \sim 5$ and $m = 1 \sim 6$ are tested. From Table 18, we can observe that different range of




(a) $f_{c_1}$ (b) $f_{c_2}$ (c) $f_{c_3}$

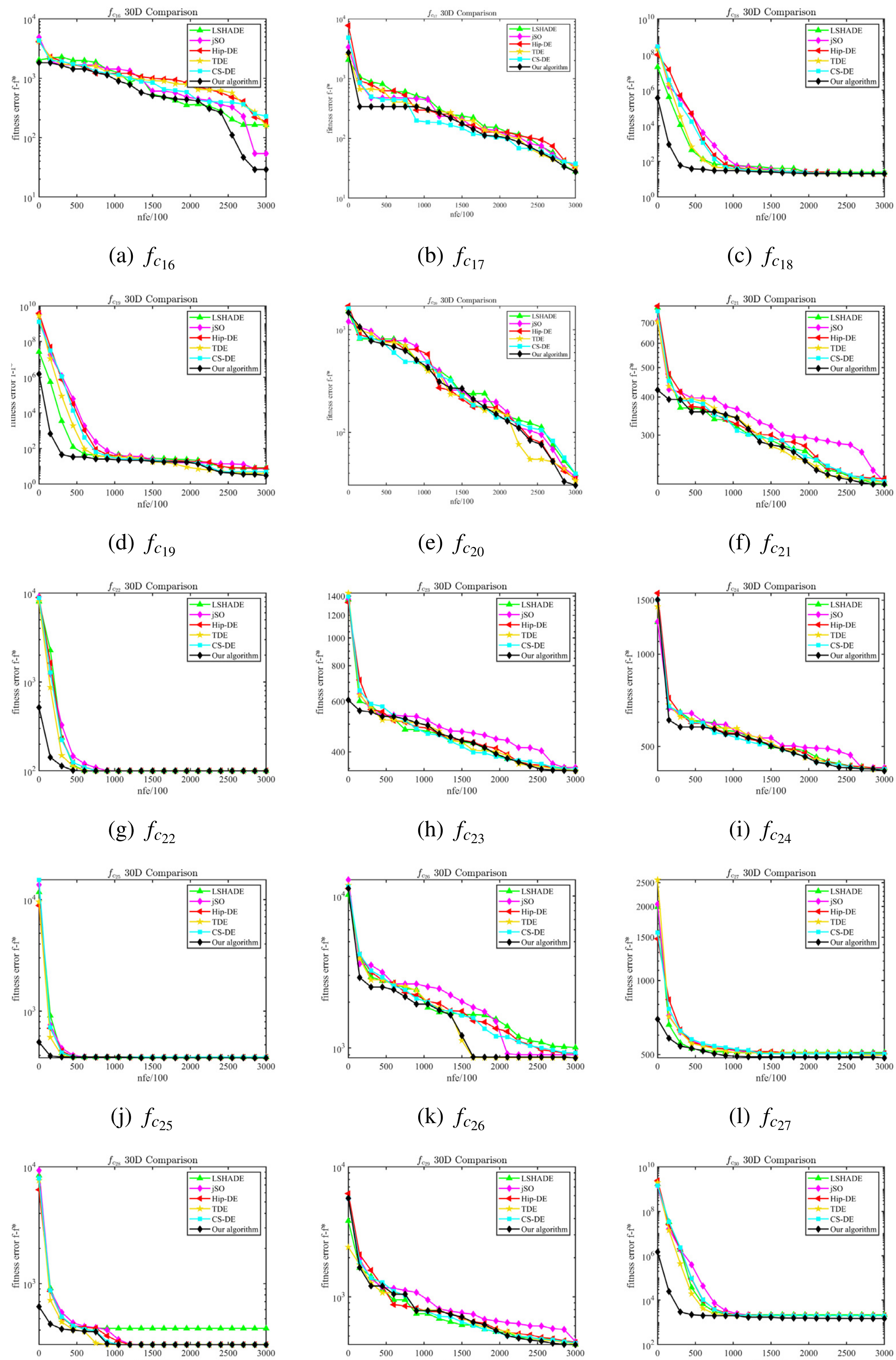
(d) $f_{c_4}$ (e) $f_{c_5}$ (f) $f_{c_6}$

(g) $f_{c_7}$ (h) $f_{c_8}$ (i) $f_{c_9}$

(j) $f_{c_{10}}$ (k) $f_{c_{11}}$ (l) $f_{c_{12}}$

(m) $f_{c_{13}}$ (n) $f_{c_{14}}$ (o) $f_{c_{15}}$

**Fig. 1.** Here provides the convergence rate comparison by employing the median value of 51 runs under $f_{c_1} - f_{c_{15}}$ from CEC2017 test suite on 30D optimization.




(a) $f_{c_{16}}$ (b) $f_{c_{17}}$ (c) $f_{c_{18}}$

(d) $f_{c_{19}}$ (e) $f_{c_{20}}$ (f) $f_{c_{21}}$

(g) $f_{c_{22}}$ (h) $f_{c_{23}}$ (i) $f_{c_{24}}$

(j) $f_{c_{25}}$ (k) $f_{c_{26}}$ (l) $f_{c_{27}}$

(m) $f_{c_{28}}$ (n) $f_{c_{29}}$ (o) $f_{c_{30}}$

**Fig. 2.** As a continued from Fig. 1, comparisons on benchmarks $f_{c_{16}} - f_{c_{30}}$ from CEC2017 test suite on 30D optimization.

**Table 6**
Comparison between LSHADE, jSO, Hip-DE, CS-DE, TDE and our algorithm under CEC2014 on 30D optimization.

| DE Variants<br>NO. | LSHADE<br>Mean/Std | jSO<br>Mean/Std | Hip-DE<br>Mean/Std | CS-DE<br>Mean/Std | TDE<br>Mean/Std | our algorithm<br>Mean/Std |
|---|---|---|---|---|---|---|
| $f_{b_1}$ | 9.4739E−15/6.7657e−15(<) | 4.2633E−14/2.9944e−14(<) | 1.0310E−14/7.0072e−15(<) | 6.1302E−15/7.6554e−15(<) | 8.6380E−15/7.5617e−15(<) | **0.0000E+00/0.0000e+00** |
| $f_{b_2}$ | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00* |
| $f_{b_3}$ | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00* |
| $f_{b_4}$ | 5.6843E−14/3.5951e−14(<) | 5.9073E−14/1.9562e−14(<) | 5.2385E−14/2.7481e−14(<) | 3.1208E−14/3.0745e−14(<) | 1.3375E−14/2.4352e−14(<) | **0.0000E+00/0.0000e+00** |
| $f_{b_5}$ | 2.0119E+01/2.3934e−02(>) | 2.0759E+01/1.6120e−01(<) | 2.0068E+01/5.1551e−02(>) | **2.0054E+01/1.8646e−02(>)** | 2.0126E+01/1.1429e−01(>) | 2.0140E+01/1.0153e−01 |
| $f_{b_6}$ | 6.2972E−07/4.4971e−06(<) | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | 5.6761E−06/3.1953e−05(<) | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00* |
| $f_{b_7}$ | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00(≈)* | *0.0000E+00/0.0000e+00* |
| $f_{b_8}$ | 1.2483E−13/3.4143e−14(>) | 2.9425E−13/1.0213e−13(<) | **3.3437E−14/5.2316e−14(>)** | 3.5666E−14/5.3276e−14(>) | 1.1369E−13/1.3258e−13(>) | 1.9839E−13/9.0503e−14 |
| $f_{b_9}$ | **6.4144E+00/1.5112e+00(>)** | 8.7632E+00/2.1548e+00(<) | 7.1198E+00/1.1622e+00(>) | 1.1191E+01/1.3137e+00(<) | 1.0074E+01/2.0292e+00(<) | 7.6575E+00/1.6116e+00 |
| $f_{b_{10}}$ | 3.6740E−03/8.0157e−03(<) | 1.1131E+00/8.7136e−01(<) | 1.6329E−03/7.0209e−03(>) | *1.2247E−03/4.9474e−03(>)* | *1.2247E−03/4.9474e−03(>)* | 2.0411E−03/6.2526e−03 |
| $f_{b_{11}}$ | 1.2144E+03/1.8854e+02(<) | 2.0396E+03/5.1897e+02(<) | 1.2380E+03/2.0173e+02(<) | 1.2792E+03/2.0870e+02(<) | 1.2003E+03/1.8800e+02(<) | **1.1174E+03/1.8830e+02** |
| $f_{b_{12}}$ | 1.5811E−01/2.8434e−02(>) | 7.4891E−01/4.4729e−01(<) | **1.3268E−01/3.7064e−02(>)** | 1.4334E−01/2.4230e−02(>) | 1.8078E−01/5.7436e−02(>) | 1.8383E−01/2.8904e−02 |
| $f_{b_{13}}$ | 1.1687E−01/1.7832e−02(>) | 1.5161E−01/2.2846e−02(<) | **1.1358E−01/1.7614e−02(>)** | 1.4204E−01/2.0295e−02(<) | 1.2792E−01/3.0454e−02(>) | 1.2921E−01/1.8234e−02 |
| $f_{b_{14}}$ | 2.3833E−01/2.4433e−02(<) | 2.0234E−01/2.7517e−02(<) | 2.2812E−01/2.5689e−02(<) | 2.0188E−01/2.7025e−02(<) | 2.2218E−01/3.3356e−02(<) | **1.6484E−01/2.6250e−02** |
| $f_{b_{15}}$ | 2.1264E+00/2.4035e−01(<) | 3.2651E+00/6.1435e−01(<) | 2.1643E+00/2.2567e−01(<) | 2.2434E+00/2.5221e−01(<) | 2.2932E+00/2.4073e−01(<) | **2.1255E+00/2.3078e−01** |
| $f_{b_{16}}$ | 8.5244E+00/4.3208e−01(<) | 9.3582E+00/7.1362e−01(<) | 8.6613E+00/4.4900e−01(<) | 9.0123E+00/3.6830e−01(<) | 8.8314E+00/6.6493e−01(<) | **8.1648E+00/4.4921e−01** |
| $f_{b_{17}}$ | 2.1952E+02/1.0531e+02(<) | 7.8189E+01/3.6285e+01(<) | 2.1157E+02/9.7936e+01(<) | 1.2480E+02/5.8406e+01(<) | 6.2898E+01/4.4588e+01(<) | **5.8212E+01/2.2460e+01** |
| $f_{b_{18}}$ | 5.8581E+00/2.5188e+00(<) | **2.1400E+00/1.3711e+00(>)** | 8.7687E+00/4.1251e+00(<) | 5.4137E+00/1.9954e+00(<) | 4.0054E+00/1.5948e+00(<) | 2.6960E+00/1.2351e+00 |
| $f_{b_{19}}$ | 3.6776E+00/5.1518e−01(<) | 2.3563E+00/5.9773e−01(<) | 3.5779E+00/5.3213e−01(<) | 3.1404E+00/4.7378e−01(<) | 2.4006E+00/6.1326e−01(<) | **1.9478E+00/6.5313e−01** |
| $f_{b_{20}}$ | 2.3661E+00/1.0688e+00(<) | 2.6261E+00/1.0587e+00(<) | 2.8417E+00/1.1984e+00(<) | 3.2442E+00/1.1519e+00(<) | 3.2945E+00/1.2045e+00(<) | **2.2327E+00/9.0739e−01** |
| $f_{b_{21}}$ | 6.2906E+01/6.6881e+01(<) | 3.2913E+01/5.0125e+01(<) | 6.4661E+01/6.3208e+01(<) | 3.1419E+01/4.4247e+01(<) | 1.1755E+01/1.9385e+01(<) | **9.5310E+00/1.8000e+01** |
| $f_{b_{22}}$ | **2.7112E+01/1.7458e+01(>)** | 6.7141E+01/5.5945e+01(<) | 7.4355E+01/5.7613e+01(<) | 8.0152E+01/5.6239e+01(<) | 8.4406E+01/5.6044e+01(<) | 6.5640E+01/5.4410e+01 |
| $f_{b_{23}}$ | 3.1524E+02/3.5915e−13(<) | 3.1524E+02/6.4311e−14(<) | 3.1524E+02/0.0000e+00(<) | 3.1524E+02/0.0000e+00(<) | 3.1524E+02/1.3401e−02(<) | **3.1520E+02/2.3059e−02** |
| $f_{b_{24}}$ | 2.2384E+02/1.1857e+00(<) | *2.0043E+02/3.0427e+00(≈)* | 2.2309E+02/9.8241e−01(<) | 2.2226E+02/7.3618e−01(<) | 2.2202E+02/3.2772e+00(<) | *2.0043E+02/3.0438e+00* |
| $f_{b_{25}}$ | 2.0261E+02/9.0783e−02(<) | 2.0258E+02/3.6873e−02(<) | 2.0269E+02/1.3055e−01(<) | 2.0261E+02/5.7255e−02(<) | 2.0261E+02/4.9071e−02(<) | **2.0253E+02/5.5923e−02** |
| $f_{b_{26}}$ | *1.0011E+02/1.5534e−02(>)* | 1.0015E+02/2.4329e−02(<) | *1.0011E+02/1.4342e−02(>)* | 1.0013E+02/1.9725e−02(<) | 1.0013E+02/2.4566e−02(<) | 1.0012E+02/1.5634e−02 |
| $f_{b_{27}}$ | *3.0000E+02/1.2862e−13(≈)* | *3.0000E+02/1.9493e−13(≈)* | *3.0000E+02/6.4311e−14(≈)* | *3.0000E+02/2.4720e−05(≈)* | *3.0000E+02/6.4311e−14(≈)* | *3.0000E+02/1.5852e−03* |
| $f_{b_{28}}$ | 8.2834E+02/2.0412e+01(<) | 8.2858E+02/2.3826e+01(<) | 8.6139E+02/2.0977e+01(<) | 8.4340E+02/1.5896e+01(<) | 8.3629E+02/2.1451e+01(<) | **8.1691E+02/2.6850e+01** |
| $f_{b_{29}}$ | 7.1608E+02/2.7147e+00(<) | 5.9936E+02/2.2692e+02(<) | 7.1769E+02/3.2011e+00(<) | 6.8405E+02/1.3402e+02(<) | **3.3827E+02/2.8339e+02(>)** | 3.8632E+02/2.7868e+02 |
| $f_{b_{30}}$ | 1.3276E+03/5.5426e+02(<) | 4.3295E+02/2.5195e+01(<) | 5.7850E+02/2.1047e+02(<) | 4.6169E+02/7.5137e+01(<) | 4.1153E+02/2.9059e+01(<) | **3.7865E+02/2.4646e+01** |
| >/≈/< | 7/4/19 | 1/6/23 | 7/5/18 | 4/4/22 | 6/5/19 | –/–/– |

**Table 7**
Comparison between LSHADE, jSO, Hip-DE, CS-DE, TDE and our algorithm under CEC2014 on 50D optimization.

| DE Variants<br>NO. | LSHADE<br>Mean/Std | jSO<br>Mean/Std | Hip-DE<br>Mean/Std | CS-DE<br>Mean/Std | TDE<br>Mean/Std | our algorithm<br>Mean/Std |
|---|---|---|---|---|---|---|
| $f_{b_1}$ | 4.6323E+02/6.5743e+02(<) | 1.8680E+01/5.2984e+01(<) | 1.3129E+03/1.6470e+03(<) | 4.0693E+01/6.3685e+01(<) | 7.0208E+01/3.0189e+02(<) | **3.5832E−10/1.1213e−09** |
| $f_{b_2}$ | 3.5666E−14/1.2510e−14(<) | 5.8515E−14/1.8342e−14(<) | 4.4026E−14/1.5373e−14(<) | 3.0651E−14/1.1144e−14(<) | 2.7307E−14/5.5718e−15(<) | **2.5635E−14/8.5358e−15** |
| $f_{b_3}$ | 5.5729E−14/7.9597e−15(<) | 6.3531E−14/1.8497e−14(<) | 5.5729E−14/7.9597e−15(<) | 4.0125E−14/2.6158e−14(<) | 2.4521E−14/2.8433e−14(<) | **6.6875E−15/1.8497e−14** |
| $f_{b_4}$ | 5.0432E+01/4.7448e+01(<) | 4.8279E+01/4.9343e+01(<) | 1.4624E+01/3.2171e+01(<) | 6.1009E+00/2.3231e+01(>) | **3.9066E+00/1.9220e+01(>)** | 7.2913E+00/2.3048e+01 |
| $f_{b_5}$ | 2.0266E+01/3.0727e−02(<) | 2.1079E+01/1.3235e−01(<) | 2.0154E+01/1.0827e−01(>) | **2.0138E+01/2.6818e−02(>)** | 2.0213E+01/1.3707e−01(>) | 2.0260E+01/1.0558e−01 |
| $f_{b_6}$ | 3.7164E−01/6.2436e−01(<) | 3.1160E−02/2.0308e−01(>) | 5.1826E−03/2.2898e−02(>) | 5.3605E−03/3.2250e−02(>) | **1.1367E−03/8.1002e−03(>)** | 2.3559E−01/1.5393e−01 |
| $f_{b_7}$ | 8.6937E−14/4.8704e−14(<) | 1.0477E−13/3.0869e−14(<) | 6.6875E−14/5.6508e−14(<) | 1.3375E−14/3.6993e−14(<) | 4.4583E−15/2.2287e−14(<) | **0.0000E+00/0.0000e+00** |
| $f_{b_8}$ | 1.1716E−10/1.6197e−10(<) | 2.4691E−08/9.8723e−08(<) | 3.9679E−13/1.8663e−13(>) | **1.9171E−13/1.3735e−13(>)** | 3.4998E−13/2.6305e−13(>) | 8.2033E−13/2.4522e−13 |
| $f_{b_9}$ | **1.1381E+01/2.3529e+00(>)** | 1.6330E+01/2.8643e+00(<) | 1.2618E+01/2.0666e+00(>) | 2.0657E+01/1.7892e+00(<) | 1.9251E+01/3.9915e+00(<) | 1.5053E+01/2.3955e+00 |
| $f_{b_{10}}$ | 4.4257E−02/2.1457e−02(<) | 1.0186E+01/3.1509e+00(<) | 1.0672E−02/1.1276e−02(<) | 5.7085E−03/6.8531e−03(>) | **3.4291E−03/6.6469e−03(>)** | 6.8701E−03/8.7729e−03 |
| $f_{b_{11}}$ | 3.2807E+03/2.7862e+02(<) | 3.3736E+03/3.1922e+02(<) | 3.2697E+03/2.7049e+02(<) | 3.2136E+03/2.5034e+02(<) | 3.2736E+03/3.2888e+02(<) | **3.0978E+03/2.8033e+02** |
| $f_{b_{12}}$ | 2.0778E−01/2.8704e−02(>) | 3.7038E−01/4.8703e−01(<) | 1.9121E−01/3.5817e−02(>) | **1.6506E−01/1.9950e−02(>)** | 2.3251E−01/4.2725e−02(<) | 2.2845E−01/3.1681e−02 |
| $f_{b_{13}}$ | **1.6400E−01/1.6833e−02(>)** | 1.9599E−01/3.5732e−02(<) | 1.7280E−01/1.5395e−02(>) | 2.0044E−01/2.0400e−02(<) | 2.0158E−01/2.5031e−02(<) | 1.8225E−01/2.4802e−02 |
| $f_{b_{14}}$ | 3.0734E−01/2.0586e−02(<) | 2.8395E−01/4.7225e−02(<) | 3.0511E−01/2.0451e−02(<) | 2.8362E−01/1.7905e−02(<) | 2.9802E−01/2.8858e−02(<) | **1.7097E−01/2.0777e−02** |
| $f_{b_{15}}$ | 5.1425E+00/4.9110e−01(<) | 5.5266E+00/4.5823e−01(<) | 5.1936E+00/4.5943e−01(<) | 5.1969E+00/3.3948e−01(<) | 5.2061E+00/4.5316e−01(<) | **4.7269E+00/4.3451e−01** |
| $f_{b_{16}}$ | 1.6934E+01/4.2713e−01(<) | 1.6955E+01/4.5562e−01(<) | 1.7005E+01/4.8695e−01(<) | 1.7226E+01/3.4652e−01(<) | 1.7012E+01/8.3679e−01(<) | **1.6350E+01/4.1922e−01** |
| $f_{b_{17}}$ | 1.4631E+03/4.3951e+02(<) | 3.5096E+02/1.8031e+02(<) | 1.7422E+03/4.4036e+02(<) | 1.0319E+03/3.0694e+02(<) | 5.4017E+02/2.0368e+02(<) | **1.3584E+02/9.1734e+01** |
| $f_{b_{18}}$ | 1.0261E+02/1.5044e+01(<) | 1.2017E+01/3.8417e+00(<) | 1.0408E+02/1.3951e+01(<) | 7.2956E+01/1.7337e+01(<) | 2.2661E+01/6.2558e+00(<) | **7.7745E+00/3.2645e+00** |
| $f_{b_{19}}$ | 8.5149E+00/1.9699e+00(>) | 9.3987E+00/5.9690e−01(<) | **7.5140E+00/1.7928e+00(>)** | 8.4981E+00/2.0286e+00(>) | 8.0620E+00/1.3593e+00(>) | 8.8252E+00/7.5525e−01 |
| $f_{b_{20}}$ | 1.2638E+01/4.0514e+00(<) | 5.6288E+00/1.7718e+00(<) | 1.4155E+01/4.8204e+00(<) | 1.0671E+01/2.6294e+00(<) | 9.1966E+00/3.5667e+00(<) | **4.5647E+00/1.7240e+00** |
| $f_{b_{21}}$ | 4.8600E+02/1.4933e+02(<) | 2.8415E+02/8.7876e+01(<) | 5.1903E+02/1.5543e+02(<) | 3.9141E+02/1.1516e+02(<) | 3.6538E+02/1.1771e+02(<) | **2.3409E+02/7.1208e+01** |
| $f_{b_{22}}$ | **1.1532E+02/8.5949e+01(>)** | 1.4103E+02/8.3745e+01(<) | 1.4872E+02/5.9586e+01(<) | 2.0654E+02/9.0662e+01(<) | 2.0482E+02/8.4264e+01(<) | 1.2635E+02/6.1694e+01 |
| $f_{b_{23}}$ | 3.4400E+02/1.1139e−13(<) | 3.4400E+02/3.8986e−13(<) | 3.4400E+02/9.0949e−14(<) | 3.4400E+02/2.1735e−13(<) | 3.4400E+02/2.0017e−13(<) | **3.4340E+02/2.1470e−01** |
| $f_{b_{24}}$ | 2.7512E+02/7.2698e−01(<) | 2.7179E+02/2.0112e+00(<) | 2.7491E+02/7.0369e−01(<) | 2.7320E+02/1.1469e+00(<) | 2.7281E+02/1.4533e+00(<) | **2.6845E+02/1.7270e+00** |
| $f_{b_{25}}$ | 2.0530E+02/3.2508e−01(<) | 2.0500E+02/1.4826e−01(<) | 2.0568E+02/3.4811e−01(<) | 2.0543E+02/2.7784e−01(<) | 2.0555E+02/3.5116e−01(<) | **2.0489E+02/1.0318e−01** |
| $f_{b_{26}}$ | **1.0016E+02/1.5470e−02(>)** | 1.0019E+02/3.1938e−02(>) | 1.1583E+02/3.6666e+01(<) | 1.2172E+02/4.1460e+01(<) | 1.1336E+02/2.8472e+01(<) | 1.0021E+02/7.6505e−02 |
| $f_{b_{27}}$ | 3.3281E+02/3.3079e+01(<) | 3.0997E+02/1.9755e+01(<) | 3.1243E+02/2.4083e+01(<) | 3.1064E+02/2.0684e+01(<) | 3.0508E+02/1.5745e+01(<) | **3.0351E+02/6.6065e+00** |
| $f_{b_{28}}$ | 1.1142E+03/2.7419e+01(>) | **1.0850E+03/2.9862e+01(>)** | 1.2504E+03/5.4025e+01(<) | 1.2294E+03/5.6944e+01(<) | 1.2503E+03/7.3111e+01(<) | 1.1984E+03/4.9295e+01 |
| $f_{b_{29}}$ | 8.1276E+02/4.5464e+01(<) | 8.1535E+02/5.2901e+01(<) | 6.1881E+02/1.0444e+02(<) | 6.2629E+02/1.1352e+02(<) | 5.0763E+02/1.4454e+02(<) | **4.1145E+02/3.1298e+01** |
| $f_{b_{30}}$ | 8.7786E+03/4.4313e+02(<) | 8.3041E+03/3.2456e+02(<) | 9.2567E+03/5.7401e+02(<) | 9.1336E+03/5.8297e+02(<) | 9.3230E+03/5.3793e+02(<) | **7.8788E+03/4.5856e+02** |
| >/≈/< | 7/0/23 | 3/0/27 | 7/0/23 | 7/0/23 | 6/0/24 | –/–/– |

$m$ have a significant impact on the performance of the proposed algorithm, and the default choice of $m$, $[1 \sim 4]$ is preferable to other settings. To further validate the assumption, we checked the experiment results of our algorithm employing two different $m$ settings, $m = 1$ and $m = 2$, and the experiment results are displayed in Table 19. From the results we can also conclude that the default setting of $m$ with a linear decreased value can obtain better performance in comparison with constant settings, e.g. $m = 1$ and $m = 2$. Therefore, introducing Minkowski distance into parameter weighting scheme is a promising direction to improve the performance of DE with respect to optimization accuracy.

### 4.4. Parameter in population diversity mechanism

The population diversity mechanism contains two parameters: the first one is the number of stagnation, which represents that no improvement is attained during $N$ generations; the second one is $\xi$ determining the threshold of stagnation indicator $VOL$. The parameter $N$ determines the frequency of horizontal or vertical crossover. A smaller value of $N$ results in more crossovers being performed, while a larger value of $N$ leads to fewer crossovers. The number of stagnation is also associated with the dimension $D$. In this case, we examine five different choices of $N$, which include $N = D$, $N = 2 \cdot D$, $N = 3 \cdot D$, $N = 4 \cdot D$ and $N = 5 \cdot D$. For threshold $\xi$, its value close to 1 represents that the algorithm is at the early stage while its value less than 1 denotes that the algorithm is at the later stage. Total six choices of $\xi$ including $\xi = 0.0005$, $\xi = 0.001$, $\xi = 0.0015$, $\xi = 0.002$ and $\xi = 0.0025$ are tested in our experiment. The experiment results of $N$ and $\xi$ are shown in Table 20 and Table 21 respectively. We can observe that when $N = 2 \cdot D$ and $\xi = 0.001$, the proposed algorithm obtains the best performance.

### 4.5. Analysis of components in population diversity mechanism

To verify the effectiveness of the components in the population diversity mechanism, we compare three variants of our zDE algorithm: variant 1 (zDE without the population diversity mechanism), variant (zDE without the dimensional interchange

**Table 8**
Comparison between LSHADE, jSO, Hip-DE, CS-DE, TDE and our algorithm under CEC2017 on 10D optimization.

| DE Variants NO. | LSHADE Mean/Std | jSO Mean/Std | Hip-DE Mean/Std | CS-DE Mean/Std | TDE Mean/Std | our algorithm Mean/Std |
|---|---|---|---|---|---|---|
| $f_{c_1}$ | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00* |
| $f_{c_2}$ | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00* |
| $f_{c_3}$ | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00* |
| $f_{c_4}$ | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00* |
| $f_{c_5}$ | 2.6361E+00/8.1587E−01(<) | 1.8729E+00/8.3593E−01(<) | 1.9525E+00/8.1945E−01(<) | 2.4606E+00/7.7859E−01(<) | 3.0073E+00/1.4136E+00(<) | **1.7976E+00/8.6578E−01** |
| $f_{c_6}$ | 4.4583E−15/2.2287E−14(<) | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | 4.4583E−14/5.6058E−14(<) | 8.6937E−14/4.8704E−14(<) | *0.0000E+00/0.0000E+00* |
| $f_{c_7}$ | 1.2097E+01/6.3890E−01(>) | 1.2078E+01/7.2070E−01(>) | **1.1763E+01/6.2667E−01(>)** | 1.2301E+01/6.0065E−01(<) | 1.2308E+01/1.1318E+00(<) | 1.2216E+01/1.2165E+00 |
| $f_{c_8}$ | 2.4414E+00/8.7339E−01(<) | **2.0289E+00/8.4333E−01(>)** | 2.1087E+00/7.8737E−01(<) | 2.5518E+00/7.9801E−01(<) | 2.7738E+00/1.7145E+00(<) | 2.0693E+00/1.3326E+00 |
| $f_{c_9}$ | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00* |
| $f_{c_{10}}$ | **2.8095E+01/4.2078E+01(>)** | 2.8622E+01/4.5822E+01(>) | 5.4977E+01/1.0564E+02(<) | 5.6217E+01/5.9176E+01(<) | 4.5781E+01/9.2213E+01(<) | 3.6361E+01/5.7347E+01 |
| $f_{c_{11}}$ | 3.9590E−01/6.8874E−01(<) | 4.4583E−15/3.1839E−14(<) | 2.8376E−01/6.5957E−01(<) | 1.5529E+00/6.5403E−01(<) | 8.5579E−02/3.2067E−01(<) | **0.0000E+00/0.0000E+00** |
| $f_{c_{12}}$ | 3.4019E+01/5.4563E+01(<) | 5.0792E+00/2.3459E+01(>) | 3.3247E+01/5.8590E+01(<) | 1.6843E+01/4.1374E+01(>) | **3.0609E−01/1.9236E−01(>)** | 2.4523E+01/4.7474E+01 |
| $f_{c_{13}}$ | 3.5270E+00/2.2884E+00(<) | 2.9004E+00/2.3060E+00(<) | 2.1097E+00/2.5678E+00(>) | **1.5403E+00/2.1830E+00(>)** | 2.0295E+00/2.3799E+00(>) | 2.4649E+00/2.5652E+00 |
| $f_{c_{14}}$ | 2.0907E−01/4.3055E−01(<) | 1.3656E−01/3.4579E−01(<) | 8.5840E−01/7.9645E−01(<) | 3.3571E−01/5.1432E−01(<) | 5.5177E−01/7.4937E−01(<) | **1.0790E−01/2.9697E−01** |
| $f_{c_{15}}$ | 1.8762E−01/2.1487E−01(<) | 2.9709E−01/2.1862E−01(<) | 1.3775E−01/2.0125E−01(<) | 1.2969E−01/1.6833E−01(<) | 1.3775E−01/1.9055E−01(<) | **1.0574E−01/1.8922E−01** |
| $f_{c_{16}}$ | **3.4327E−01/1.7852E−01(>)** | 5.1955E−01/2.9781E−01(<) | 4.4220E−01/2.4157E−01(<) | 5.0583E−01/1.6584E−01(<) | 5.3994E−01/2.7718E−01(<) | 3.8818E−01/2.0333E−01 |
| $f_{c_{17}}$ | **1.3891E−01/1.4731E−01(>)** | 4.9298E−01/3.9129E−01(<) | 7.9989E−01/2.7181E+00(<) | 2.2755E−01/2.2790E−01(<) | 1.6916E−01/1.7433E−01(<) | 1.5126E−01/1.5643E−01 |
| $f_{c_{18}}$ | 2.3745E−01/2.0231E−01(<) | 2.5619E−01/1.9455E−01(<) | 1.5345E−01/1.6948E−01(<) | 2.6464E−01/1.8584E−01(<) | 2.2620E−01/1.9216E−01(<) | **1.0228E−01/1.7060E−01** |
| $f_{c_{19}}$ | 1.1055E−02/1.0560E−02(<) | 9.4262E−03/1.0798E−02(<) | 1.4125E−02/2.5601E−02(<) | 1.8034E−02/1.2768E−02(<) | 1.2058E−02/1.0601E−02(<) | **6.1122E−03/1.2021E−02** |
| $f_{c_{20}}$ | *0.0000E+00/0.0000E+00(≈)* | 3.6114E−01/1.3055E−01(<) | 6.1210E−03/4.3713E−02(<) | *0.0000E+00/0.0000E+00(≈)* | 1.2242E−02/6.1198E−02(<) | *0.0000E+00/0.0000E+00* |
| $f_{c_{21}}$ | 1.5359E+02/5.1750E+01(<) | 1.4656E+02/5.1886E+01(>) | 1.6097E+02/5.0211E+01(<) | **1.4607E+02/4.8991E+01(>)** | 1.4680E+02/5.2163E+01(>) | 1.5277E+02/5.2263E+01 |
| $f_{c_{22}}$ | 1.0001E+02/5.6308E−02(≈) | 1.0000E+02/1.0925E−13(>) | 1.0000E+02/0.0000E+00(>) | **9.8902E+01/7.8434E+00(>)** | 1.0005E+02/1.1986E−01(<) | 1.0001E+02/5.6299E−02 |
| $f_{c_{23}}$ | 3.0328E+02/1.4013E+00(<) | 3.0146E+02/1.6873E+00(<) | 3.0122E+02/1.4469E+00(<) | **2.9630E+02/4.2299E+01(>)** | 3.0218E+02/2.0513E+00(<) | 3.0088E+02/1.5790E+00 |
| $f_{c_{24}}$ | 3.0366E+02/7.5121E+01(>) | 2.7055E+02/1.0084E+02(>) | 2.9062E+02/8.2896E+01(>) | **2.6545E+02/1.0244E+02(>)** | 3.0064E+02/7.4376E+01(>) | 3.0442E+02/6.8446E+01 |
| $f_{c_{25}}$ | 4.0881E+02/1.9694E+01(>) | 4.0953E+02/1.9977E+01(>) | 4.1663E+02/2.2574E+01(>) | **4.0684E+02/1.8204E+01(>)** | 4.1305E+02/2.1643E+01(>) | 4.1843E+02/2.2809E+01 |
| $f_{c_{26}}$ | *3.0000E+02/0.0000E+00(≈)* | *3.0000E+02/0.0000E+00(≈)* | *3.0000E+02/0.0000E+00(≈)* | *3.0000E+02/0.0000E+00(≈)* | *3.0000E+02/0.0000E+00(≈)* | *3.0000E+02/0.0000E+00* |
| $f_{c_{27}}$ | 3.8941E+02/2.1287E−01(>) | **3.8936E+02/2.4015E−01(>)** | 3.9358E+02/1.2076E+00(<) | 3.9288E+02/1.7221E+00(>) | 3.9278E+02/1.8967E+00(>) | 3.9328E+02/1.4789E+00 |
| $f_{c_{28}}$ | 3.5894E+02/1.2069E+02(<) | 3.1340E+02/5.7489E+01(>) | 3.3950E+02/1.0007E+02(<) | **3.0556E+02/3.9732E+01(>)** | 3.1056E+02/5.2812E+01(>) | 3.1947E+02/6.7409E+01 |
| $f_{c_{29}}$ | 2.3410E+02/2.7555E+00(<) | 2.3394E+02/3.0671E+00(<) | 2.3325E+02/3.7920E+00(<) | 2.3795E+02/5.0444E+00(<) | 2.3419E+02/4.2552E+00(<) | **2.3208E+02/2.6442E+00** |
| $f_{c_{30}}$ | 3.2446E+04/1.6020E+05(<) | 3.9452E+02/4.3407E−02(>) | 3.9828E+02/1.3075E+01(>) | **3.9451E+02/2.3305E−02(>)** | 3.9453E+02/5.4511E−02(>) | 8.0816E+03/5.4840E+04 |
| $>/\approx/<$ | 7/8/15 | 11/7/12 | 6/7/17 | 10/7/13 | 8/6/16 | –/–/– |

**Table 9**
Comparison between LSHADE, jSO, Hip-DE, CS-DE, TDE and our algorithm under CEC2017 on 30D optimization.

| DE Variants NO. | LSHADE Mean/Std | jSO Mean/Std | Hip-DE Mean/Std | CS-DE Mean/Std | TDE Mean/Std | our algorithm Mean/Std |
|---|---|---|---|---|---|---|
| $f_{c_1}$ | 2.7864E−16/1.9899E−15(<) | 1.3932E−15/4.2679E−15(<) | 5.5729E−16/2.7859E−15(<) | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00* |
| $f_{c_2}$ | 3.3437E−15/9.2483E−15(<) | 2.7864E−15/8.5358E−15(≈) | 8.9166E−15/1.3319E−14(<) | 8.3593E−15/1.5352E−14(<) | **2.2292E−15/7.7172E−15(>)** | 2.7864E−15/8.5358E−15 |
| $f_{c_3}$ | 3.3437E−15/1.3508E−14(<) | 4.2354E−14/2.5019E−14(<) | 3.3437E−15/1.3508E−14(<) | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00* |
| $f_{c_4}$ | 5.8562E+01/2.1269E−14(<) | 5.8562E+01/2.7847E−14(<) | 5.8670E+01/7.7797E−01(<) | 5.7413E+01/8.2003E+00(<) | 5.5480E+01/1.3810E+01(<) | **3.0368E+01/7.9033E+00** |
| $f_{c_5}$ | **6.3199E+00/1.5399E+00(>)** | 9.0011E+00/1.9905E+00(<) | 7.2833E+00/1.4568E+00(<) | 1.1038E+01/1.6227E+00(<) | 9.7720E+00/1.8951E+00(<) | 7.1670E+00/1.5422E+00 |
| $f_{c_6}$ | 5.3677E−09/2.6833E−08(>) | 8.0803E−09/3.2519E−08(>) | 6.0409E−09/2.7124E−08(>) | 1.7445E−08/4.4548E−08(>) | **1.1146E−13/1.5919E−14(>)** | 9.2326E−08/6.5447E−07 |
| $f_{c_7}$ | 3.7385E+01/1.3641E+00(>) | 3.8746E+01/1.9227E+00(<) | **3.5880E+01/9.6925E−01(>)** | 4.0215E+01/1.7286E+00(<) | 3.8798E+01/1.9240E+00(<) | 3.8358E+01/1.2269E+00 |
| $f_{c_8}$ | **7.3618E+00/1.4181E+00(>)** | 8.8054E+00/1.6649E+00(<) | 7.3763E+00/1.4044E+00(>) | 1.1463E+01/1.9147E+00(<) | 9.9245E+00/2.4457E+00(<) | 8.0305E+00/1.6725E+00 |
| $f_{c_9}$ | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00* |
| $f_{c_{10}}$ | 1.4191E+03/2.0794E+02(<) | 1.5401E+03/2.3970E+02(<) | 1.4894E+03/2.2988E+02(<) | 1.6281E+03/2.1799E+02(<) | 1.5419E+03/2.2276E+02(<) | **1.4052E+03/2.3353E+02** |
| $f_{c_{11}}$ | 1.8634E+01/2.4363E+01(<) | **4.1677E+00/8.5999E+00(>)** | 1.4561E+01/2.1375E+01(<) | 9.3239E+00/1.6467E+01(<) | 9.1811E+00/1.6329E+01(<) | 6.4434E+00/1.2370E+01 |
| $f_{c_{12}}$ | 1.0373E+03/3.3002E+02(<) | 1.6009E+02/9.2030E+01(<) | 1.0934E+03/3.7777E+02(<) | 7.8296E+02/2.9598E+02(<) | 4.6572E+02/2.3659E+02(<) | **8.7751E+01/7.5130E+01** |
| $f_{c_{13}}$ | 1.5708E+01/5.2229E+00(<) | 1.6778E+01/3.3253E+00(<) | 1.5258E+01/6.3961E+00(<) | 1.4849E+01/5.9753E+00(<) | 1.3915E+01/5.7923E+00(<) | **1.3138E+01/6.0085E+00** |
| $f_{c_{14}}$ | 2.1348E+01/4.3814E+00(<) | 2.1457E+01/3.2159E+00(<) | 2.2518E+01/1.1845E+00(<) | 2.2452E+01/3.3134E+00(<) | **1.9994E+01/8.2217E+00(>)** | 2.0448E+01/5.3070E+00 |
| $f_{c_{15}}$ | 4.0008E+00/2.3587E+00(<) | **1.0103E+00/7.0740E−01(>)** | 2.8528E+00/1.4990E+00(<) | 2.8842E+00/1.3246E+00(<) | 2.1867E+00/1.2635E+00(<) | 1.1042E+00/8.5681E−01 |
| $f_{c_{16}}$ | **5.1347E+01/4.3559E+01(>)** | 5.4949E+01/6.7056E+01(>) | 1.1182E+02/9.1892E+01(<) | 1.9707E+02/1.0213E+02(<) | 1.6426E+02/8.9100E+01(<) | 6.2728E+01/6.5293E+01 |
| $f_{c_{17}}$ | 3.1879E+01/6.2712E+00(<) | 3.3496E+01/6.4682E+00(<) | 3.1126E+01/6.7203E+00(<) | 3.6153E+01/6.7814E+00(<) | 3.2809E+01/7.3068E+00(<) | **2.4296E+01/6.2151E+00** |
| $f_{c_{18}}$ | 2.1763E+01/8.8785E−01(<) | 2.0765E+01/3.3103E−01(<) | 2.2606E+01/1.5667E+00(<) | 2.1506E+01/8.0943E−01(<) | 2.1268E+01/7.1838E−01(<) | **2.0205E+01/2.3373E+00** |
| $f_{c_{19}}$ | 5.3009E+00/1.2616E+00(<) | 4.6243E+00/1.7513E+00(<) | 5.1407E+00/1.7925E+00(<) | 5.5169E+00/2.0586E+00(<) | **3.9648E+00/9.8463E−01(>)** | 4.1399E+00/1.3862E+00 |
| $f_{c_{20}}$ | 3.1817E+01/6.3336E+00(<) | 2.9930E+01/7.9389E+00(<) | 3.9194E+01/2.4154E+01(<) | 3.9653E+01/8.0348E+00(<) | 3.3117E+01/9.1059E+00(<) | **2.8539E+01/8.4673E+00** |
| $f_{c_{21}}$ | **2.0710E+02/1.4812E+00(>)** | 2.0957E+02/1.8002E+00(<) | 2.0747E+02/1.6222E+00(>) | 2.1145E+02/2.0328E+00(<) | 2.0913E+02/2.1731E+00(<) | 2.0767E+02/1.7816E+00 |
| $f_{c_{22}}$ | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/1.0047E−13(≈)* | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/6.3901E−14* |
| $f_{c_{23}}$ | 3.4950E+02/2.7650E+00(<) | 3.5041E+02/3.8216E+00(<) | 3.4463E+02/4.2466E+00(>) | 3.4809E+02/3.5422E+00(<) | **3.4314E+02/3.4879E+00(>)** | 3.4512E+02/2.4712E+00 |
| $f_{c_{24}}$ | 4.2552E+02/1.7469E+00(<) | 4.2729E+02/2.5303E+00(<) | 4.2094E+02/2.2052E+00(>) | 4.2180E+02/2.3801E+00(<) | **4.1962E+02/3.5323E+00(>)** | 4.2124E+02/2.1478E+00 |
| $f_{c_{25}}$ | 3.8674E+02/2.3732E−02(<) | 3.8670E+02/1.0235E−02(<) | 3.8678E+02/2.7130E−02(<) | 3.8675E+02/1.1362E−02(<) | 3.8670E+02/2.2770E−01(<) | **3.8567E+02/4.7920E−01** |
| $f_{c_{26}}$ | 9.2546E+02/3.4801E+01(<) | 9.3878E+02/4.2029E+01(<) | 8.7924E+02/3.1269E+01(<) | 9.3195E+02/4.1173E+01(<) | **8.5829E+02/4.1708E+01(>)** | 8.6085E+02/4.3963E+01 |
| $f_{c_{27}}$ | 5.0334E+02/5.8744E+00(<) | 4.9773E+02/6.7188E+00(<) | 5.0545E+02/5.8970E+00(<) | 5.0302E+02/5.3567E+00(<) | 4.9585E+02/6.4060E+00(<) | **4.9131E+02/8.9913E+00** |
| $f_{c_{28}}$ | 3.3305E+02/5.2139E+01(<) | 3.1320E+02/3.6533E+01(<) | 3.2842E+02/4.9133E+01(<) | 3.2333E+02/4.4992E+01(<) | 3.2821E+02/4.8782E+01(<) | **3.1075E+02/3.2984E+01** |
| $f_{c_{29}}$ | 4.3156E+02/7.1649E+00(<) | 4.3590E+02/1.1192E+01(<) | 4.3469E+02/7.4613E+00(<) | 4.4216E+02/6.9851E+00(<) | 4.3323E+02/9.1769E+00(<) | **4.2796E+02/1.1900E+01** |
| $f_{c_{30}}$ | 1.9910E+03/5.1984E+01(<) | 1.9708E+03/2.6363E+01(<) | 2.0514E+03/5.5328E+01(<) | 2.0122E+03/4.4439E+01(<) | 1.9965E+03/2.3905E+01(<) | **1.9340E+03/2.9223E+01** |
| $>/\approx/<$ | 6/2/22 | 4/3/23 | 6/2/22 | 1/4/25 | 7/4/19 | –/–/– |

operations), and variant 3 (zDE without the prescreening strategy). The experiment results of these three variants are given in Table 22, and from the results, we can observe that the population diversity mechanism and its components are crucial to the performance of the proposed algorithm. The proposed algorithm with population diversity mechanism is significantly better than three variants. It can be attributed to following aspects: (1) If the stagnant individuals jump out of the local optimum in one dimension through the vertical crossover operation, they will quickly disperse to the whole population through the horizontal crossover operation. Then the updated dimension will enable the other dimensions that fall into the local optimum have more chances to escape. Through the combination of horizontal crossover and vertical crossover, the convergence speed and solution accuracy of the proposed algorithm can be further improved. (2) By prescreening individuals with poor individual fitness values, the population diversity in the later stage of evolution can be effectively increased. Therefore, the proposed population diversity enhancement mechanism can further enhance the performance of the propose algorithm.

## 4.6. Time complexity analysis

In this section, we compare the time complexity of our proposed method to that of DE variants. The time complexity comparison is conducted according to the method recommended by CEC2013 competition, and the results are provided in Table 23. $T_0$ represents the time spent on basic arithmetic expressions, $T_1$ represents the time necessary for 200,000 function evaluations for benchmark 14 from CEC2013 on 30D optimization and $\widehat{T_2}$ represents the overall cost of optimizing $f_{14}$, which is calculated by executing 5 times. $\frac{\widehat{T_2}-T_1}{T_0}$ is used to determine the time complexity of each algorithm. The experiment results show that our algorithm has less time complexity than LSHADE, jSO, CS-DE and TDE, indicating that the performance improvement of the

**Table 10**
Comparison between LSHADE, jSO, Hip-DE, CS-DE, TDE and our algorithm under CEC2017 on 50D optimization.

| DE Variants NO. | LSHADE Mean/Std | jSO Mean/Std | Hip-DE Mean/Std | CS-DE Mean/Std | TDE Mean/Std | our algorithm Mean/Std |
|---|---|---|---|---|---|---|
| $f_{c_1}$ | 1.7555E−14/6.0880E−15(<) | 2.9258E−14/8.7194E−15(<) | 2.5914E−14/6.7891E−15(<) | 1.5604E−14/4.2679E−15(<) | 1.3654E−14/2.7859E−15(<) | **1.2539E−14/4.6242E−15** |
| $f_{c_2}$ | 1.0092E−12/5.7168E−12(<) | 1.8391E−13/7.1747E−13(<) | 4.5140E−14/2.9633E−14(<) | 6.2416E−14/9.3579E−14(<) | **2.0062E−14/1.9103E−14(>)** | 2.0620E−14/1.4014E−14 |
| $f_{c_3}$ | 1.6496E−13/5.8253E−14(<) | 2.6750E−13/7.2166E−14(<) | 1.4824E−13/4.9657E−14(<) | 1.2260E−13/3.8339E−14(<) | 1.1257E−13/3.1114E−14(<) | **4.3468E−14/2.4352E−14** |
| $f_{c_4}$ | 7.4401E+01/5.0815E+01(<) | **5.2699E+01/4.3116E+01(>)** | 8.0188E+01/4.7197E+01(<) | 8.1661E+01/4.7635E+01(<) | 7.5297E+01/4.6764E+01(<) | 6.2329E+01/4.8277E+01 |
| $f_{c_5}$ | **1.2781E+01/2.0421E+00(>)** | 1.6472E+01/3.3435E+00(<) | 1.4647E+01/1.9140E+00(>) | 2.2319E+01/2.8008E+00(<) | 1.9692E+01/3.5045E+00(<) | 1.5672E+01/2.2970E+00 |
| $f_{c_6}$ | 1.3996E−07/2.6902E−07(>) | 4.5306E−07/1.0754E−06(>) | 2.5672E−05/1.8260E−04(>) | 5.0715E−07/6.7352E−07(>) | **5.6404E−09/1.5600E−08(>)** | 7.9284E−04/7.4654E−04 |
| $f_{c_7}$ | 6.2661E+01/2.0497E+00(>) | 6.6291E+01/2.8331E+00(<) | **6.1420E+01/1.3780E+00(>)** | 6.7415E+01/3.0906E+00(<) | 6.6370E+01/3.8529E+00(<) | 6.4947E+01/2.0932E+00 |
| $f_{c_8}$ | **1.2592E+01/2.1023E+00(>)** | 1.7579E+01/2.9018E+00(<) | 1.5243E+01/2.0900E+00(>) | 2.1839E+01/2.4886E+00(<) | 2.0500E+01/3.3224E+00(<) | 1.6519E+01/2.4316E+00 |
| $f_{c_9}$ | 6.0187E−14/5.7310E−14(<) | 7.5791E−14/5.4126E−14(<) | 1.1146E−14/3.4143E−14(<) | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00(≈)* | *0.0000E+00/0.0000E+00* |
| $f_{c_{10}}$ | 3.1037E+03/3.2509E+02(<) | 3.1900E+03/2.9785E+02(<) | 3.1438E+03/3.1760E+02(<) | 3.1401E+03/3.7133E+02(<) | 3.1186E+03/3.1459E+02(<) | **2.8870E+03/2.3735E+02** |
| $f_{c_{11}}$ | 4.9702E+01/8.0592E+00(<) | 2.7732E+01/3.2713E+00(<) | 4.7692E+01/7.6651E+00(<) | 4.5959E+01/7.3351E+00(<) | 3.9154E+01/5.9297E+00(<) | **2.3020E+01/3.5336E+00** |
| $f_{c_{12}}$ | 2.1818E+03/4.5692E+02(<) | 1.6365E+03/4.2894E+02(<) | 2.3693E+03/6.1342E+02(<) | 2.1772E+03/4.6676E+02(<) | 1.7829E+03/5.1572E+02(<) | **8.2840E+02/3.3083E+02** |
| $f_{c_{13}}$ | 4.9925E+01/2.3914E+01(<) | 3.1840E+01/1.7620E+01(<) | 6.5626E+01/2.2012E+01(<) | 4.9846E+01/1.9987E+01(<) | 3.9641E+01/2.3152E+01(<) | **1.5760E+01/1.4307E+01** |
| $f_{c_{14}}$ | 2.9563E+01/3.3497E+00(<) | **2.4844E+01/1.9528E+00(>)** | 3.2397E+01/4.2989E+00(<) | 2.8735E+01/2.4162E+00(<) | 2.7634E+01/2.3058E+00(<) | 2.5123E+01/2.1456E+00 |
| $f_{c_{15}}$ | 3.8002E+01/8.6118E+00(<) | 2.3728E+01/3.0491E+00(<) | 5.1579E+01/1.5661E+01(<) | 3.6408E+01/8.7737E+00(<) | 2.8765E+01/3.2275E+00(<) | **1.8276E+01/1.7694E+00** |
| $f_{c_{16}}$ | 3.4340E+02/1.2380E+02(<) | 4.4427E+02/1.3553E+02(<) | 3.7666E+02/1.0247E+02(<) | 5.0217E+02/1.2300E+02(<) | 3.9703E+02/1.0713E+02(<) | **2.8843E+02/1.1686E+02** |
| $f_{c_{17}}$ | **2.4250E+02/6.8884E+01(>)** | 2.7824E+02/9.4804E+01(<) | 2.9637E+02/5.9646E+01(<) | 3.6134E+02/8.0999E+01(<) | 3.3001E+02/8.7503E+01(<) | 2.5411E+02/6.1572E+01 |
| $f_{c_{18}}$ | 4.0906E+01/1.4363E+01(<) | 2.4922E+01/2.1399E+00(<) | 5.3481E+01/1.5632E+01(<) | 3.3606E+01/6.2667E+00(<) | 2.7780E+01/3.3531E+00(<) | **2.1296E+01/6.4447E−01** |
| $f_{c_{19}}$ | 2.4516E+01/6.3755E+00(<) | 1.4514E+01/2.5875E+00(<) | 4.0751E+01/1.4438E+01(<) | 2.1504E+01/4.0598E+00(<) | 1.5292E+01/2.3821E+00(<) | **1.1184E+01/2.4717E+00** |
| $f_{c_{20}}$ | 1.5170E+02/5.2989E+01(<) | 1.2969E+02/7.4729E+01(<) | 1.6931E+02/6.9511E+01(<) | 2.0718E+02/6.9671E+01(<) | 1.9350E+02/8.1794E+01(<) | **1.2612E+02/5.1699E+01** |
| $f_{c_{21}}$ | **2.1289E+02/2.5945E+00(>)** | 2.1880E+02/3.1300E+00(<) | 2.1625E+02/2.2651E+00(>) | 2.2416E+02/2.6529E+00(<) | 2.2150E+02/3.9995E+00(<) | 2.1679E+02/2.3917E+00 |
| $f_{c_{22}}$ | 2.0179E+03/1.7758E+03(<) | 1.5263E+03/1.8031E+03(<) | **1.0261E+02/1.5993E+01(>)** | 3.3004E+02/8.8254E+02(>) | 1.6813E+02/4.7342E+02(>) | 4.3240E+02/1.0171E+03 |
| $f_{c_{23}}$ | 4.3021E+02/3.9508E+00(<) | 4.3274E+02/6.0158E+00(<) | 4.2759E+02/6.9648E+00(<) | 4.2996E+02/4.9945E+00(<) | 4.2353E+02/6.0631E+00(<) | **4.1769E+02/5.9885E+00** |
| $f_{c_{24}}$ | 5.0668E+02/2.9836E+00(<) | 5.0930E+02/3.7851E+00(<) | 5.0766E+02/5.8008E+00(<) | 5.0145E+02/5.5717E+00(<) | 5.0076E+02/5.9937E+00(<) | **4.9891E+02/5.7952E+00** |
| $f_{c_{25}}$ | 4.8336E+02/1.2014E+01(<) | 4.8116E+02/3.1464E+00(<) | 4.8183E+02/4.0653E+00(<) | 4.8468E+02/1.0853E+01(<) | 4.9015E+02/2.0005E+01(<) | **4.8041E+02/5.2291E+00** |
| $f_{c_{26}}$ | 1.1394E+03/4.4333E+01(<) | 1.1578E+03/5.1982E+01(<) | 1.1307E+03/7.1604E+01(<) | 1.1598E+03/6.1031E+01(<) | 1.0764E+03/6.9792E+01(<) | **1.0137E+03/9.1296E+01** |
| $f_{c_{27}}$ | 5.3343E+02/1.8105E+01(<) | 5.1152E+02/1.0414E+01(<) | 5.3516E+02/7.5460E+00(<) | 5.2731E+02/6.9044E+00(<) | 5.1365E+02/1.4209E+01(<) | **5.0288E+02/7.4740E+00** |
| $f_{c_{28}}$ | 4.6938E+02/2.0290E+01(<) | 4.6172E+02/1.1607E+01(<) | 4.8758E+02/2.4279E+01(<) | 4.8279E+02/2.4661E+01(<) | 4.8077E+02/2.1005E+01(<) | **4.5774E+02/8.9392E−01** |
| $f_{c_{29}}$ | 3.5285E+02/1.0007E+01(<) | 3.6471E+02/1.4654E+01(<) | 3.7017E+02/1.2459E+01(<) | 3.8120E+02/1.3547E+01(<) | 3.5675E+02/1.5458E+01(<) | **3.5207E+02/9.3607E+00** |
| $f_{c_{30}}$ | 6.5923E+05/7.0107E+04(<) | 6.0465E+05/3.5617E+04(<) | 6.1904E+05/3.5079E+04(<) | 6.0998E+05/3.5421E+04(<) | 6.0586E+05/3.2428E+04(<) | **5.5063E+05/2.9494E+04** |
| >/≈/< | 6/0/24 | 3/0/27 | 6/0/24 | 2/1/27 | 3/1/26 | –/–/– |

**Table 11**
Comparison between LSHADE, jSO, Hip-DE, CS-DE, TDE and our algorithm under CEC2017 on 100D optimization.

| DE Variants NO. | LSHADE Mean/Std | jSO Mean/Std | Hip-DE Mean/Std | CS-DE Mean/Std | TDE Mean/Std | our algorithm Mean/Std |
|---|---|---|---|---|---|---|
| $f_{c_1}$ | 7.3896E−13/8.9686E−13(>) | 1.7047E−08/7.6147E−12(<) | 2.3256E−12/5.4658E−12(>) | 4.0376E−13/1.0636E−12(>) | **8.1921E−14/2.6901E−14(>)** | 9.7013E−10/1.6647E−09 |
| $f_{c_2}$ | 3.0065E+02/1.6475E+03(>) | 9.0966E+01/3.7725E+01(>) | 6.2828E+02/2.4471E+03(>) | 6.1215E−06/2.6485E−05(>) | **2.8572E−09/1.3816E−08(>)** | 7.0388E+03/3.4544E+04 |
| $f_{c_3}$ | 8.5931E−07/1.0218E−06(<) | 3.1320E−07/3.1046E−07(<) | 1.9368E−07/4.1047E−07(<) | **1.8217E−08/2.5011E−08(>)** | 1.3228E−06/2.6002E−06(<) | 1.1729E−07/1.7478E−07 |
| $f_{c_4}$ | 1.9668E+02/1.1574E+01(>) | 1.9378E+02/2.1805E+01(>) | 1.9590E+02/3.3743E+00(>) | **1.7912E+02/4.1699E+01(>)** | 1.8216E+02/3.6610E+01(>) | 1.9702E+02/2.2929E+01 |
| $f_{c_5}$ | 3.7352E+01/5.6033E+00(>) | **2.9772E+01/6.0453E+00(>)** | 3.9906E+01/4.3826E+00(>) | 5.5196E+01/5.2892E+00(<) | 5.2153E+01/6.2635E+00(<) | 4.1501E+01/4.2892E+00 |
| $f_{c_6}$ | 6.5851E−03/3.8162E−03(>) | **2.0632E−04/4.8850E−04(>)** | 1.0858E−03/1.1299E−03(>) | 1.1754E−03/1.4870E−03(>) | 8.1807E−04/1.1000E−03(>) | 1.8059E−01/6.0533E−02 |
| $f_{c_7}$ | 1.4008E+02/3.7571E+00(<) | **1.3125E+02/8.0417E+00(>)** | 1.3220E+02/3.7489E+00(<) | 1.4365E+02/4.6367E+00(<) | 1.3922E+02/5.2438E+00(<) | 1.3164E+02/3.3483E+00 |
| $f_{c_8}$ | 3.9456E+01/4.5198E+00(>) | **3.0340E+01/4.4281E+00(>)** | 4.3640E+01/4.3073E+00(<) | 5.6699E+01/5.5716E+00(<) | 5.3118E+01/5.2425E+00(<) | 4.1303E+01/3.9234E+00 |
| $f_{c_9}$ | 5.3220E−01/5.6448E−01(>) | **1.2037E−13/2.7016E−14(>)** | 3.8620E−02/7.6503E−02(>) | 2.4707E−02/7.2587E−02(>) | 4.9546E−02/1.5529E−01(>) | 2.5222E+00/9.2745E−01 |
| $f_{c_{10}}$ | 1.0346E+04/4.9641E+02(<) | 1.1346E+04/7.7446E+02(<) | 1.0383E+04/6.5123E+02(<) | 9.5769E+03/5.2098E+02(<) | 9.7407E+03/4.6635E+02(<) | **9.0734E+03/5.1173E+02** |
| $f_{c_{11}}$ | 4.6550E+02/1.1355E+02(<) | 1.1919E+02/3.2361E+01(<) | 4.5567E+02/7.8064E+01(<) | 3.1395E+02/8.6859E+01(<) | 2.6243E+02/9.3696E+01(<) | **3.9369E+01/1.9225E+01** |
| $f_{c_{12}}$ | 2.1519E+04/6.7812E+03(<) | 1.8260E+04/8.8971E+03(<) | 2.3315E+04/8.6249E+03(<) | 1.8685E+04/5.9155E+03(<) | 1.8376E+04/7.5452E+03(<) | **6.3872E+03/2.3063E+03** |
| $f_{c_{13}}$ | 4.9293E+02/3.9935E+02(<) | 2.1862E+02/6.2157E+01(<) | 1.8580E+03/8.0657E+02(<) | 6.5764E+02/2.1529E+02(<) | 1.7372E+02/5.0264E+01(<) | **1.3535E+02/4.2802E+01** |
| $f_{c_{14}}$ | 2.5249E+02/3.2490E+01(<) | 7.0393E+01/1.2821E+01(<) | 2.5379E+02/2.9448E+01(<) | 2.0830E+02/2.3427E+01(<) | 1.3496E+02/4.0905E+01(<) | **3.1282E+01/2.2953E+00** |
| $f_{c_{15}}$ | 2.5019E+02/5.5026E+01(<) | 1.9962E+02/4.2506E+01(<) | 2.3688E+02/4.6663E+01(<) | 2.5154E+02/4.7025E+01(<) | 2.5228E+02/5.0502E+01(<) | **6.3206E+01/3.3597E+01** |
| $f_{c_{16}}$ | 1.7080E+03/2.5313E+02(<) | 1.7876E+03/3.7649E+02(<) | 1.8912E+03/3.0987E+02(<) | 2.0269E+03/2.0414E+02(<) | 1.6023E+03/3.2037E+02(<) | **1.4752E+03/2.6583E+02** |
| $f_{c_{17}}$ | 1.1030E+03/1.7351E+02(<) | 1.2518E+03/2.8538E+02(<) | 1.2923E+03/1.7638E+02(<) | 1.3737E+03/1.5546E+02(<) | 1.2545E+03/2.0872E+02(<) | **1.0562E+03/2.1126E+02** |
| $f_{c_{18}}$ | 2.2707E+02/5.1504E+01(<) | 1.9851E+02/4.0267E+01(<) | 2.1770E+02/4.6402E+01(<) | 2.0801E+02/3.6057E+01(<) | 2.1013E+02/4.7503E+01(<) | **3.4309E+01/5.5957E+00** |
| $f_{c_{19}}$ | 1.7210E+02/2.5433E+01(<) | 1.4435E+02/1.7947E+01(<) | 1.7315E+02/2.2927E+01(<) | 1.6943E+02/2.5448E+01(<) | 1.6833E+02/2.5322E+01(<) | **3.9938E+01/4.1262E+00** |
| $f_{c_{20}}$ | 1.5994E+03/1.8845E+02(<) | 1.3490E+03/2.5210E+02(<) | 1.6243E+03/1.7776E+02(<) | 1.4716E+03/2.0857E+02(<) | 1.6461E+03/1.8794E+02(<) | **1.3479E+03/1.9548E+02** |
| $f_{c_{21}}$ | 2.5871E+02/6.5915E+00(<) | 2.5955E+02/4.6235E+00(<) | 2.6713E+02/5.9443E+00(<) | 2.8307E+02/6.7429E+00(<) | 2.7286E+02/4.8409E+00(<) | **2.5727E+02/5.7838E+00** |
| $f_{c_{22}}$ | 1.1167E+04/1.5373E+03(<) | 1.1442E+04/8.0037E+02(<) | 1.1635E+04/5.5910E+02(<) | 1.0677E+04/1.5646E+03(<) | 1.0689E+04/2.1773E+03(<) | **9.9699E+03/5.3504E+02** |
| $f_{c_{23}}$ | 5.6892E+02/8.6729E+00(<) | 5.9110E+02/1.0543E+01(<) | 6.0766E+02/1.2630E+01(<) | 5.7142E+02/6.9787E+00(<) | 5.6750E+02/9.3346E+00(<) | **5.6281E+02/1.3550E+01** |
| $f_{c_{24}}$ | 9.1118E+02/7.4354E+00(<) | 9.0300E+02/1.1805E+01(<) | 9.2928E+02/1.3172E+01(<) | 9.2915E+02/1.2480E+01(<) | 8.9985E+02/1.1396E+01(<) | **8.7132E+02/1.4509E+01** |
| $f_{c_{25}}$ | 7.4622E+02/2.8359E+01(<) | 7.1784E+02/4.2077E+01(<) | 7.3344E+02/4.1661E+01(<) | 7.4036E+02/3.4771E+01(<) | 7.3555E+02/3.3199E+01(<) | **6.7505E+02/4.0031E+01** |
| $f_{c_{26}}$ | 3.3042E+03/8.9139E+01(<) | 3.1735E+03/1.1047E+02(<) | 3.3343E+03/9.3665E+01(<) | 3.4939E+03/9.8045E+01(<) | 3.2187E+03/1.0098E+02(<) | **2.9160E+03/1.2166E+02** |
| $f_{c_{27}}$ | 6.2777E+02/1.9876E+01(<) | 6.0566E+02/1.7111E+01(<) | 6.3851E+02/1.3582E+01(<) | 6.1888E+02/1.8594E+01(<) | 6.2555E+02/2.7851E+01(<) | **5.6562E+02/1.1097E+01** |
| $f_{c_{28}}$ | 5.2971E+02/3.2387E+01(<) | 5.3478E+02/3.0677E+01(<) | 5.2982E+02/2.8022E+01(<) | 5.2513E+02/3.2606E+01(<) | 5.2244E+02/2.9050E+01(<) | **5.2217E+02/2.9713E+01** |
| $f_{c_{29}}$ | 1.2654E+03/1.7396E+02(<) | 1.4931E+03/2.3702E+02(<) | 1.2337E+03/1.8583E+02(<) | 1.4746E+03/1.7948E+02(<) | 1.0657E+03/1.6003E+02(<) | **1.0484E+03/1.4229E+02** |
| $f_{c_{30}}$ | 2.3863E+03/1.5950E+02(<) | 2.4451E+03/1.4975E+02(<) | 2.5758E+03/1.6872E+02(<) | 2.5269E+03/1.6872E+02(<) | 2.3923E+03/1.3255E+02(<) | **2.1336E+03/1.0150E+02** |
| >/≈/< | 7/0/23 | 7/0/23 | 6/0/24 | 6/0/24 | 5/0/25 | –/–/– |

**Table 12**
Summary of the comparison results between our algorithm and other algorithms under CEC2013, CEC2014 and CEC2017 test suites on 10D, 30D, 50D and 100D optimization (only in CEC2017) respectively.

| Test suit: | CEC2013 | | | CEC2014 | | | CEC2017 | | | | All |
|---|---|---|---|---|---|---|---|---|---|---|---|
| >/≈/< | D=10 | D=30 | D=50 | D=10 | D=30 | D=50 | D=10 | D=30 | D=50 | D=100 | ∑ |
| LSHADE | 3/10/15 | 6/4/18 | 3/1/24 | 10/7/13 | 7/4/19 | 7/0/23 | 7/8/15 | 6/2/22 | 6/0/24 | 7/0/23 | 62/36/196 |
| jSO | 6/7/15 | 2/4/22 | 8/1/19 | 8/6/16 | 1/6/23 | 3/0/27 | 11/7/12 | 4/3/23 | 3/0/27 | 7/0/23 | 58/30/206 |
| Hip-DE | 8/11/9 | 11/4/13 | 8/4/16 | 11/7/12 | 7/5/18 | 7/0/23 | 6/7/17 | 6/2/22 | 6/0/24 | 6/0/24 | 76/40/178 |
| CS-DE | 12/8/8 | 7/5/16 | 9/3/16 | 13/6/11 | 4/4/22 | 7/0/23 | 10/7/13 | 1/4/25 | 2/1/27 | 6/0/24 | 71/38/185 |
| TDE | 9/10/9 | 10/4/14 | 12/3/13 | 10/7/13 | 6/5/19 | 6/0/24 | 8/6/16 | 7/4/19 | 3/1/26 | 5/0/25 | 76/40/178 |

proposed algorithm does not come at the cost of increased time complexity.

## 5. Conclusion

In this paper, a new DE variant, namely zDE, is proposed with a bi-stage parameter adaptation scheme, perturbation strategy and new population diversity mechanism. By separating the adaptation of scale factor into two stages, a better balance between exploration and exploitation abilities can be achieved. In addition, the new weighting strategy based on Minkowski distance is also used for modify the direction of parameter adaptation. A more promising trial vector can be generated by target vector and donor vector perturbed by t-distribution during crossover oper-

**Table 13**
Comparison between PSO-sono, E-QUATRE, EBOwithCMAR, HSES, EA4eig and our algorithm under CEC2017 on 10D optimization.

| DE Variants NO. | PSO-sono Mean/Std | E-QUATRE Mean/Std | EBOwithCMAR Mean/Std | HSES Mean/Std | EA4eig Mean/Std | our algorithm Mean/Std |
|---|---|---|---|---|---|---|
| $f_{c_1}$ | 6.3116E+02/1.1615E+03(<) | *0/0(≈)* | *0/0(≈)* | 5.4263E−11/6.6214E−11(<) | 7.9196E−09/1.3179E−09(<) | *0/0* |
| $f_{c_2}$ | 1.3423E−05/1.3838E−05(<) | *0/0(≈)* | *0/0(≈)* | 1.3294E−10/1.4107E−10(<) | 7.4903E−09/2.0053E−09(<) | *0/0* |
| $f_{c_3}$ | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | 1.9103E−11/3.0939E−11(<) | 7.9623E−09/1.5244E−09(<) | *0/0* |
| $f_{c_4}$ | 1.7870E−09/4.3894E−09(<) | *0/0(≈)* | *0/0(≈)* | 1.8577E−11/1.8084E−11(<) | 7.7607E−09/1.7080E−09(<) | *0/0* |
| $f_{c_5}$ | 6.6916E+00/2.7435E+00(<) | 2.6525E+00/1.1569E+00(<) | **7.8036E−02/2.7016E−01(>)** | 7.4134E−01/7.1313E−01(>) | 1.4046E+00/1.1802E+00(>) | 1.7976E+00/8.6578E−01 |
| $f_{c_6}$ | 3.8940E−03/9.3890E−03(<) | 1.1039E−11/7.2404E−12(<) | 8.9166E−15/3.0869E−14(<) | 2.9394E−11/2.4266E−11(<) | 9.0013E−09/9.1598E−10(<) | **0/0** |
| $f_{c_7}$ | 1.7566E+01/4.3691E+00(<) | 1.3843E+01/1.3892E+00(<) | **1.0599E+01/2.1771E−01(>)** | 1.1382E+01/6.9149E−01(>) | 1.1759E+01/9.5348E−01(>) | 1.2216E+01/1.2165E+00 |
| $f_{c_8}$ | 7.6670E+00/3.8137E+00(<) | 2.8627E+00/1.1253E+00(<) | **9.7545E−02/2.9881E−01(>)** | 7.8036E−01/9.1868E−01(>) | 1.3851E+00/1.2117E+00(>) | 2.0693E+00/1.3326E+00 |
| $f_{c_9}$ | 2.1197E−02/9.3206E−02(<) | *0/0(≈)* | *0/0(≈)* | 1.5805E−12/5.0411E−12(<) | 8.2683E−09/1.3542E−09(<) | *0/0* |
| $f_{c_{10}}$ | 3.7609E+02/2.2854E+02(<) | 1.0447E+02/7.1206E+01(<) | 4.5233E+01/5.4518E+01(<) | 1.3276E+02/1.6893E+02(<) | 5.3758E+01/6.9689E+01(<) | **3.6361E+01/5.7347E+01** |
| $f_{c_{11}}$ | 1.4363E+01/1.1600E+01(<) | 7.8615E−02/2.6103E−01(<) | 1.8324E−12/6.5468E−12(<) | 1.3211E+00/8.7294E+00(<) | 8.0232E−09/1.3697E−09(<) | **0/0** |
| $f_{c_{12}}$ | 6.8123E+03/7.7539E+03(<) | 7.2638E+00/2.8434E+01(>) | 9.5153E+01/6.8571E+01(<) | 1.1303E+01/3.2652E+01(>) | **4.8742E+00/2.3327E+01(>)** | 2.4523E+01/4.7474E+01 |
| $f_{c_{13}}$ | 1.3453E+02/1.2087E+02(<) | 1.3243E+00/1.6830E+00(>) | 3.3603E+00/2.4006E+00(<) | 3.7501E+00/2.5333E+00(<) | **1.0727E+00/1.9334E+00(>)** | 2.4649E+00/2.5652E+00 |
| $f_{c_{14}}$ | 4.5170E+01/2.0260E+01(<) | 1.6192E−03/5.8597E−03(>) | 4.9652E−02/1.9396E−01(>) | 6.7773E+00/2.7853E+01(<) | **8.2413E−09/1.4386E−09(>)** | 1.0790E−01/2.9697E−01 |
| $f_{c_{15}}$ | 4.0651E+01/3.5330E+01(<) | 9.1302E−02/8.7164E−02(>) | 1.6636E−01/1.9498E−01(<) | 5.6252E−01/5.4385E−01(<) | **1.1985E−02/3.5835E−02(>)** | 1.0574E−01/1.8922E−01 |
| $f_{c_{16}}$ | 1.0114E+01/2.9761E+01(<) | 6.2828E−01/1.7731E−01(<) | 4.7551E−01/1.7674E−01(<) | 3.0691E+00/1.6557E+01(<) | **1.8707E−01/1.8006E−01(>)** | 3.8818E−01/2.0333E−01 |
| $f_{c_{17}}$ | 2.8422E+01/1.0135E+01(<) | 2.3333E−01/1.6406E−01(<) | **1.3268E−01/1.4807E−01(>)** | 1.6455E+01/1.1205E+01(<) | 2.1754E−01/2.6978E−01(<) | 1.5126E−01/1.5643E−01 |
| $f_{c_{18}}$ | 5.7336E+02/3.6656E+03(<) | 1.3001E−01/1.4268E−01(<) | 7.7802E−01/2.8166E+00(<) | 5.1094E−01/3.5352E−01(<) | **9.9269E−02/1.3889E−01(>)** | 1.0228E−01/1.7060E−01 |
| $f_{c_{19}}$ | 2.2310E+01/1.8084E+01(<) | 5.4933E−02/3.7395E−02(<) | 1.7631E−02/1.8862E−02(<) | 4.2541E−01/5.3053E−01(<) | 8.9779E−03/1.0455E−02(<) | **6.1122E−03/1.2021E−02** |
| $f_{c_{20}}$ | 2.2088E+01/9.7098E+00(<) | *0/0(≈)* | 1.3466E−01/1.5615E−01(<) | 1.0754E+01/1.0150E+01(<) | 8.0719E−09/1.7641E−09(<) | *0/0* |
| $f_{c_{21}}$ | 1.8310E+02/4.6390E+01(<) | 1.2892E+02/4.7480E+01(>) | **1.2408E+02/4.3542E+01(>)** | 1.9585E+02/2.4364E+01(<) | 1.5429E+02/5.1700E+01(<) | 1.5277E+02/5.2263E+01 |
| $f_{c_{22}}$ | 9.8571E+01/1.1530E+01(>) | **9.6085E+01/1.9605E+01(>)** | 1.0000E+02/0.0000E+00(>) | 1.0000E+02/0.0000E+00(>) | 1.0001E+02/4.0101E−02(≈) | 1.0001E+02/5.6299E−02 |
| $f_{c_{23}}$ | 3.0787E+02/3.8636E+00(<) | 3.0384E+02/2.0214E+00(<) | **3.0050E+02/1.0846E+00(>)** | 3.0056E+02/1.1548E+00(>) | 3.0091E+02/1.7027E+00(<) | 3.0088E+02/1.5790E+00 |
| $f_{c_{24}}$ | 3.0402E+02/8.2255E+01(>) | 2.3238E+02/1.1680E+02(>) | **1.7337E+02/1.0423E+02(>)** | 3.2413E+02/3.2026E+01(<) | 2.7840E+02/9.4578E+01(>) | 3.0442E+02/6.8446E+01 |
| $f_{c_{25}}$ | 4.3133E+02/2.2791E+01(<) | **3.9963E+02/8.9210E+00(>)** | 4.1525E+02/2.2084E+01(>) | 4.4636E+02/9.8342E−01(<) | 4.1133E+02/2.0987E+01(>) | 4.1843E+02/2.2809E+01 |
| $f_{c_{26}}$ | 3.0081E+02/1.8240E+01(<) | 2.9412E+02/4.2008E+01(>) | **2.8824E+02/4.7527E+01(>)** | 3.0000E+02/2.8884E−13(≈) | 3.0000E+02/0.0000E+00(≈) | 3.0000E+02/0.0000E+00 |
| $f_{c_{27}}$ | 3.8337E+02/1.3616E+01(>) | 3.8842E+02/1.2301E+00(>) | 3.9160E+02/2.3562E+00(>) | 3.9744E+02/1.8856E+00(<) | **3.7955E+02/9.2742E+00(>)** | 3.9328E+02/1.4789E+00 |
| $f_{c_{28}}$ | 4.6461E+02/1.3806E+02(<) | **2.9968E+02/5.8396E+01(>)** | 3.0844E+02/4.6402E+01(>) | 5.9586E+02/2.2104E+01(<) | 3.3065E+02/4.8587E+01(<) | 3.1947E+02/6.7409E+01 |
| $f_{c_{29}}$ | 2.6864E+02/2.2545E+01(<) | 2.3695E+02/5.0439E+00(<) | 2.3264E+02/2.2842E+00(<) | 2.6478E+02/1.1268E+01(<) | **2.3121E+02/2.7910E+00(>)** | 2.3208E+02/2.6442E+00 |
| $f_{c_{30}}$ | 1.8822E+05/4.2336E+05(<) | 3.9527E+02/3.0722E+00(>) | 4.0649E+02/1.7225E+01(>) | 4.0979E+02/2.3821E+01(>) | **2.0267E+02/4.5353E+00(>)** | 8.0816E+03/5.4840E+04 |
| >/≈/< | 3/1/26 | 12/6/12 | 14/5/11 | 7/1/22 | 14/2/14 | –/–/– |

**Table 14**
Comparison between PSO-sono, E-QUATRE, EBOwithCMAR, HSES, EA4eig and our algorithm under CEC2017 on 30D optimization.

| DE Variants NO. | PSO-sono Mean/Std | E-QUATRE Mean/Std | EBOwithCMAR Mean/Std | HSES Mean/Std | EA4eig Mean/Std | our algorithm Mean/Std |
|---|---|---|---|---|---|---|
| $f_{c_1}$ | 1.5758E+03/1.9114E+03(<) | 1.0867E−14/6.0880E−15(<) | *0/0(≈)* | 2.6968E−10/2.7008E−10(<) | 9.0249E−09/8.3123E−10(<) | *0/0* |
| $f_{c_2}$ | 4.0345E+13/1.8381E+14(<) | 6.7989E−14/6.8937E−14(<) | **0/0(>)** | 1.5171E−07/1.9033E−07(<) | 2.7500E+00/1.9639E+01(<) | 2.7864E−15/8.5358E−15 |
| $f_{c_3}$ | 5.8438E−02/3.1588E−01(<) | 5.6843E−14/1.1369E−14(<) | *0/0(≈)* | 2.1580E−10/2.2087E−10(<) | 9.5115E−09/4.5406E−10(<) | *0/0* |
| $f_{c_4}$ | 1.0236E+02/5.2555E+01(<) | 2.9234E+01/3.0498E+01(>) | 5.6565E+01/1.1123E+01(<) | **4.3019E+00/9.8107E+00(>)** | 1.8368E+01/1.4059E+01(>) | 3.0368E+01/7.9033E+00 |
| $f_{c_5}$ | 3.2326E+01/9.0347E+00(<) | 3.6128E+01/7.1430E+00(<) | **3.5655E+00/1.7456E+00(>)** | 8.4279E+00/2.8723E+00(<) | 2.5011E+01/6.6298E+00(<) | 7.1670E+00/1.5422E+00 |
| $f_{c_6}$ | 2.8914E−01/4.2848E−01(<) | 4.7120E−11/1.0970E−10(>) | **0/0(>)** | 1.1369E−13/0(>) | 1.2185E−07/3.5047E−07(<) | 9.2326E−08/6.5447E−07 |
| $f_{c_7}$ | 6.4112E+01/1.3983E+01(<) | 6.7158E+01/8.5441E+00(<) | **3.3566E+01/8.3745E−01(>)** | 4.0901E+01/6.1020E+00(<) | 5.8696E+01/5.9282E+00(<) | 3.8358E+01/1.2269E+00 |
| $f_{c_8}$ | 3.3633E+01/9.4872E+00(<) | 3.8090E+01/7.7163E+00(<) | **2.0233E+00/1.3200E+00(>)** | 7.8621E+00/2.8019E+00(>) | 2.7820E+01/8.5450E+00(<) | 8.0305E+00/1.6725E+00 |
| $f_{c_9}$ | 2.8888E+00/8.0568E+00(<) | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | 3.5109E−03/1.7551E−02(<) | *0/0* |
| $f_{c_{10}}$ | 3.0802E+03/6.4300E+02(<) | 1.9947E+03/4.0893E+02(<) | 1.4156E+03/2.1532E+02(<) | **9.7003E+02/3.1135E+02(>)** | 1.8450E+03/2.8897E+02(<) | 1.4052E+03/2.3353E+02 |
| $f_{c_{11}}$ | 1.4136E+02/4.7455E+01(<) | 1.2560E+01/1.1842E+01(<) | 7.6565E+00/8.7545E+00(<) | 9.9998E+00/1.8378E+01(<) | 1.0875E+01/5.6216E+00(<) | **6.4434E+00/1.2370E+01** |
| $f_{c_{12}}$ | 1.6141E+05/6.0326E+05(<) | 8.1707E+02/4.8904E+02(<) | 4.6356E+02/2.6323E+02(<) | **6.7426E+01/1.2517E+02(>)** | 3.7680E+02/2.2126E+02(<) | 8.7751E+01/7.5130E+01 |
| $f_{c_{13}}$ | 1.5922E+04/2.0479E+04(<) | 2.5021E+01/1.8180E+01(<) | 1.4960E+01/6.2656E+00(<) | 3.6408E+01/1.6625E+01(<) | 2.3614E+01/1.7710E+01(<) | **1.3138E+01/6.0085E+00** |
| $f_{c_{14}}$ | 1.9490E+03/6.3754E+03(<) | **1.5355E+01/9.7451E+00(>)** | 2.1989E+01/3.9495E+00(<) | 1.7009E+01/8.8094E+00(>) | 1.5759E+01/1.0795E+01(>) | 2.0448E+01/5.3070E+00 |
| $f_{c_{15}}$ | 5.6144E+03/6.6936E+03(<) | 9.0078E+00/3.8733E+00(<) | 3.6899E+00/2.1546E+00(<) | 6.0466E+00/3.6024E+00(<) | 4.9745E+00/2.7908E+00(<) | **1.1042E+00/8.5681E−01** |
| $f_{c_{16}}$ | 6.3594E+02/2.7452E+02(<) | 3.7708E+02/1.6342E+02(<) | 6.5990E+01/5.6989E+01(<) | 2.5703E+02/1.8323E+02(<) | 2.6101E+02/1.4074E+02(<) | **6.2728E+01/6.5293E+01** |
| $f_{c_{17}}$ | 2.2415E+02/1.2619E+02(<) | 3.9135E+01/9.0990E+00(<) | 2.9855E+01/7.5626E+00(<) | 4.1548E+01/6.1590E+01(<) | 2.7706E+01/1.2990E+01(<) | **2.4296E+01/6.2151E+00** |
| $f_{c_{18}}$ | 6.0574E+04/1.3025E+05(<) | 2.6041E+01/3.5569E+00(<) | 2.2155E+01/1.0990E+00(<) | **1.9113E+01/6.3259E+00(>)** | 2.1546E+01/4.5306E+00(<) | 2.0205E+01/2.3373E+00 |
| $f_{c_{19}}$ | 6.7466E+03/8.7875E+03(<) | 1.1719E+01/2.5459E+00(<) | 8.0545E+00/2.2889E+00(<) | **4.0660E+00/1.8995E+00(>)** | 5.3706E+00/1.4107E+00(<) | 4.1399E+00/1.3862E+00 |
| $f_{c_{20}}$ | 2.1479E+02/9.6852E+01(<) | 6.2579E+01/5.7720E+01(<) | 3.5745E+01/7.5687E+00(<) | 1.6421E+02/5.8428E+01(<) | 3.2403E+01/3.8201E+01(<) | **2.8539E+01/8.4673E+00** |
| $f_{c_{21}}$ | 2.3744E+02/1.0700E+01(<) | 2.3774E+02/7.3901E+00(<) | 2.0892E+02/2.0256E+01(<) | 2.0813E+02/2.8177E+00(<) | 2.2536E+02/6.6971E+00(<) | **2.0767E+02/1.7816E+00** |
| $f_{c_{22}}$ | 1.0045E+02/1.2035E+00(<) | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/0(≈)* | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/8.0991E−06(≈)* | *1.0000E+02/6.3901E−14* |
| $f_{c_{23}}$ | 3.8588E+02/1.0111E+01(<) | 3.8304E+02/8.2400E+00(<) | 3.5155E+02/3.5122E+00(<) | 3.5067E+02/7.5643E+00(<) | 3.7703E+02/9.2290E+00(<) | **3.4512E+02/2.4712E+00** |
| $f_{c_{24}}$ | 4.5245E+02/1.1272E+01(<) | 4.5177E+02/7.2973E+00(<) | 4.2366E+02/4.5563E+01(<) | **4.1919E+02/4.4718E+00(>)** | 4.4465E+02/8.3676E+00(<) | 4.2124E+02/2.1478E+00 |
| $f_{c_{25}}$ | 3.9849E+02/1.2548E+01(<) | 3.8682E+02/9.8589E−02(<) | 3.8766E+02/7.5632E−01(<) | 3.8675E+02/2.4560E−02(<) | **3.7967E+02/2.3702E+00(>)** | 3.8567E+02/4.7920E−01 |
| $f_{c_{26}}$ | 1.2751E+03/3.7716E+02(<) | 1.2978E+03/1.8236E+02(<) | **5.3757E+00/3.0656E+02(>)** | 8.9408E+02/1.2325E+02(<) | 1.2079E+03/2.5055E+02(<) | 8.6085E+02/4.3963E+01 |
| $f_{c_{27}}$ | 5.3660E+02/2.7631E+01(<) | 5.0721E+02/5.8780E+00(<) | 5.0256E+02/4.0357E+00(<) | 5.1600E+02/5.6741E+00(<) | 4.9395E+02/1.8589E+01(<) | **4.9131E+02/8.9913E+00** |
| $f_{c_{28}}$ | 3.8931E+02/5.9323E+01(<) | **3.0894E+02/3.0947E+01(>)** | 3.1123E+02/2.8858E+01(<) | 3.2430E+02/4.4248E+01(<) | 3.4037E+02/5.3390E+01(<) | 3.1075E+02/3.2984E+01 |
| $f_{c_{29}}$ | 7.7091E+02/1.4341E+02(<) | 4.6216E+02/3.7766E+01(<) | 4.3356E+02/1.1266E+01(<) | 4.6742E+02/5.5686E+01(<) | **4.1194E+02/4.9578E+01(>)** | 4.2796E+02/1.1900E+01 |
| $f_{c_{30}}$ | 1.6864E+04/1.5445E+04(<) | 2.1701E+03/1.1632E+02(<) | 1.9967E+03/4.2126E+02(<) | 2.0528E+03/3.8234E+01(<) | **2.7323E+02/1.6829E+01(>)** | 1.9340E+03/2.9223E+01 |
| >/≈/< | 0/0/30 | 4/2/24 | 6/4/20 | 9/2/19 | 5/1/24 | –/–/– |

ation. In the population diversity mechanism, when individuals are detected as in the stagnant state, they will be renewed by dimensional interchange.

A large test suite including 88 benchmark functions from CEC2013, CEC2014, and CEC2017 test suites is employed in algorithm validation. By the comparison with the recent state-of-the-art DE variants and non-DE variants, our algorithm demonstrates highly priority in solving these complex problems in terms of optimization accuracy and convergence rate. However, there are still a few challenges, such as incorporating parameter adaptation schemes into multi-objective evolutionary algorithms and developing more efficient methods for escaping local optima. These issues will be tackled in our future work.

## CRediT authorship contribution statement

**Zhenghao Song:** Software, Writing – original draft. **Zhenyu Meng:** Conceptualization, Methodology, Supervision, Writing – review & editing.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

Data will be made available on request.

## Acknowledgments

This work is supported by the Natural Science Foundation of Fujian Province (Grant No. 2021J05227), and Scientific Research Startup Foundation of Fujian University of Technology (GY-Z19013).

**Table 15**
Comparison between PSO-sono, E-QUATRE, EBOwithCMAR, HSES, EA4eig and our algorithm under CEC2017 on 50D optimization.

| DE Variants NO. | PSO-sono Mean/Std | E-QUATRE Mean/Std | EBOwithCMAR Mean/Std | HSES Mean/Std | EA4eig Mean/Std | our algorithm Mean/Std |
|---|---|---|---|---|---|---|
| $f_{c_1}$ | 2.7795E+03/3.4768E+03(<) | 3.7338E−14/1.9887E−14(<) | 1.6567E−14/5.5959E−15(<) | 2.6783E−09/8.9954E−09(<) | 9.8152E−09/2.3919E−10(<) | **1.2539E−14/4.6242E−15** |
| $f_{c_2}$ | 1.3791E+36/9.2480E+36(<) | 1.1699E−10/6.1009E−10(<) | 2.7888E−14/3.4554E−14(<) | 3.4155E−08/7.2647E−08(<) | 5.9754E−08/1.0982E−07(<) | **2.0620E−14/1.4014E−14** |
| $f_{c_3}$ | 2.0982E+02/3.2578E+02(<) | 1.6496E−13/4.2923E−14(<) | 6.4554E−13/2.4565E−12(<) | 6.5560E−09/2.2860E−08(<) | 9.8809E−09/9.1158E−11(<) | **4.3468E−14/2.4352E−14** |
| $f_{c_4}$ | 2.2800E+02/5.3103E+01(<) | 4.3511E+01/4.6421E+01(>) | **4.2690E+01/3.3265E+01(>)** | 5.3354E+01/4.9905E+01(>) | 6.0442E+01/4.3603E+01(>) | 6.2329E+01/4.8277E+01 |
| $f_{c_5}$ | 6.3664E+01/1.5403E+01(<) | 8.2260E+01/1.4179E+01(<) | 7.5690E+00/2.4266E+00(>) | **1.0340E+00/9.3252E−01(>)** | 7.2417E+01/1.5186E+01(<) | 1.5672E+01/2.2970E+00 |
| $f_{c_6}$ | 5.1183E−01/1.4133E+00(<) | **2.0285E−13/4.7224E−14(>)** | 8.5654E−08/1.1455E−07(>) | 5.8171E−07/6.8744E−07(>) | 2.5008E−06/4.1812E−06(>) | 7.9284E−04/7.4654E−04 |
| $f_{c_7}$ | 1.2032E+02/2.0509E+01(<) | 1.2410E+02/1.3557E+01(<) | 6.5842E+01/1.6521E+00(<) | 6.5153E+01/6.7496E−01(<) | 1.2089E+02/9.3407E+00(<) | **6.4947E+01/2.0932E+00** |
| $f_{c_8}$ | 6.6020E+01/1.4720E+01(<) | 8.3561E+01/1.3336E+01(<) | 7.9155E+00/2.4785E+00(>) | **1.4827E+00/8.5294E−01(>)** | 6.8008E+01/1.4322E+01(<) | 1.6519E+01/2.4316E+00 |
| $f_{c_9}$ | 1.0334E+01/1.1035E+01(<) | 6.3983E−02/1.7731E−01(<) | *0/0(≈)* | *0/0(≈)* | 3.3396E−01/4.6692E−01(<) | *0/0* |
| $f_{c_{10}}$ | 5.9742E+03/8.9391E+02(<) | 3.9573E+03/6.5769E+02(<) | 3.1125E+03/4.0156E+02(<) | **3.4595E+02/2.7277E+02(>)** | 3.7967E+03/4.8984E+02(<) | 2.8870E+03/2.3735E+02 |
| $f_{c_{11}}$ | 2.4999E+02/6.5333E+01(<) | 5.4777E+01/1.4297E+01(<) | 2.6541E+01/3.3625E+00(<) | 2.4822E+01/2.2031E+00(<) | 5.8310E+01/1.6760E+01(<) | **2.3020E+01/3.5336E+00** |
| $f_{c_{12}}$ | 1.8894E+06/5.6260E+06(<) | 2.1826E+04/2.0600E+04(<) | 1.9723E+03/8.3466E+02(<) | **1.4874E+02/1.3646E+02(>)** | 3.3423E+04/5.8807E+04(<) | 8.2840E+02/3.3083E+02 |
| $f_{c_{13}}$ | 4.5487E+03/4.6911E+03(<) | 1.8778E+02/1.6112E+02(<) | 4.1356E+01/2.4895E+01(<) | 3.7732E+01/2.7048E+01(<) | 9.8452E+01/3.7358E+01(<) | **1.5760E+01/1.4307E+01** |
| $f_{c_{14}}$ | 2.9340E+04/5.3191E+04(<) | 5.1434E+01/1.0881E+01(<) | 3.1266E+01/3.5265E+00(<) | **1.5351E+01/8.6637E+00(>)** | 5.3667E+01/1.1247E+01(<) | 2.5123E+01/2.1456E+00 |
| $f_{c_{15}}$ | 3.6908E+03/3.2790E+03(<) | 4.7484E+01/1.0893E+01(<) | 2.9566E+01/5.2323E+00(<) | **1.7283E+01/2.3518E+00(>)** | 4.9206E+01/1.8719E+01(<) | 1.8276E+01/1.7694E+00 |
| $f_{c_{16}}$ | 1.3775E+03/3.6473E+02(<) | 8.1635E+02/2.3891E+02(<) | 3.4656E+02/1.4633E+02(<) | 5.1388E+02/1.5180E+02(<) | 7.4657E+02/1.7412E+02(<) | **2.8843E+02/1.1686E+02** |
| $f_{c_{17}}$ | 1.1184E+03/2.8239E+02(<) | 4.9635E+02/1.4422E+02(<) | 2.7566E+02/8.6326E+01(<) | 2.7971E+02/1.2748E+02(<) | 4.5612E+02/1.3678E+02(<) | **2.5411E+02/6.1572E+01** |
| $f_{c_{18}}$ | 6.4114E+05/1.0881E+06(<) | 1.1442E+02/1.0565E+02(<) | 3.2090E+01/5.9895E+00(<) | 2.1548E+01/2.8718E+00(<) | 4.7030E+01/1.1258E+01(<) | **2.1296E+01/6.4447E−01** |
| $f_{c_{19}}$ | 1.5369E+04/1.0469E+04(<) | 2.0847E+01/4.4584E+00(<) | 2.4523E+01/3.9874E+00(<) | 1.1581E+01/4.6618E+00(<) | 3.1828E+01/8.4263E+00(<) | **1.1184E+01/2.4717E+00** |
| $f_{c_{20}}$ | 6.5418E+02/2.3121E+02(<) | 3.0861E+02/1.4236E+02(<) | 1.4732E+02/7.4422E+01(<) | **4.2065E+01/4.8140E+01(>)** | 2.2643E+02/8.0916E+01(<) | 1.2612E+02/5.1699E+01 |
| $f_{c_{21}}$ | 2.7369E+02/1.4861E+01(<) | 2.8310E+02/1.4911E+01(<) | 2.1766E+02/4.0652E+00(<) | 2.1792E+02/1.1036E+00(<) | 2.7114E+02/1.4883E+01(<) | **2.1679E+02/2.3917E+00** |
| $f_{c_{22}}$ | 5.3900E+03/2.3066E+03(<) | 2.7848E+03/2.3094E+03(<) | 3.6566E+02/9.2466E+02(>) | **1.0000E+02/5.4568E−09(>)** | 3.2736E+03/2.0159E+03(<) | 4.3240E+02/1.0171E+03 |
| $f_{c_{23}}$ | 5.1712E+02/2.3779E+01(<) | 5.0782E+02/1.4048E+01(<) | 4.3405E+02/8.1656E+00(<) | 4.2428E+02/8.3844E+00(<) | 5.0179E+02/1.6629E+01(<) | **4.1769E+02/5.9885E+00** |
| $f_{c_{24}}$ | 5.7142E+02/2.4573E+01(<) | 5.6700E+02/1.3977E+01(<) | 5.0666E+02/3.8545E+00(<) | **4.9012E+02/3.4017E+00(>)** | 5.6377E+02/1.5682E+01(<) | 4.9891E+02/5.7952E+00 |
| $f_{c_{25}}$ | 6.4803E+02/3.3328E+01(<) | 5.2426E+02/2.8178E+01(<) | 4.8965E+02/2.4787E+01(<) | 5.3803E+02/3.4249E+01(<) | **4.5946E+02/2.0193E+01(>)** | 4.8041E+02/5.2291E+00 |
| $f_{c_{26}}$ | 1.8216E+03/4.0653E+02(<) | 1.9458E+03/1.4631E+02(<) | 1.3563E+03/4.0656E+02(<) | **6.1350E+02/1.5159E+02(>)** | 1.8322E+03/2.0942E+02(<) | 1.0137E+03/9.1296E+01 |
| $f_{c_{27}}$ | 8.9218E+02/1.2642E+02(<) | 5.3159E+02/1.4916E+01(<) | 5.2236E+02/7.7565E+00(<) | 5.8644E+02/1.9204E+01(<) | **4.9884E+02/8.3540E+00(>)** | 5.0288E+02/7.4740E+00 |
| $f_{c_{28}}$ | 6.1682E+02/4.9693E+01(<) | 4.8299E+02/2.3320E+01(<) | 4.6766E+02/1.7952E+01(<) | 5.0327E+02/9.5974E+00(<) | **4.5452E+02/1.7839E+01(>)** | 4.5774E+02/8.9392E−01 |
| $f_{c_{29}}$ | 1.0573E+03/3.0111E+02(<) | 4.6162E+02/9.9445E+01(<) | 3.5766E+02/1.9788E+01(<) | 4.7529E+02/1.8534E+02(<) | 4.0460E+02/1.0270E+02(<) | **3.5207E+02/9.3607E+00** |
| $f_{c_{30}}$ | 4.0126E+06/6.0698E+06(<) | 5.9851E+05/2.3237E+04(<) | 6.1570E+05/3.6544E+04(<) | 6.0030E+05/1.4864E+04(<) | **1.1640E+03/4.3426E+02(>)** | 5.5063E+05/2.9494E+04 |
| >/≈/< | 0/0/30 | 2/0/28 | 5/1/24 | 12/1/17 | 6/0/24 | –/–/– |

**Table 16**
Our algorithm obtains the best or tie best in comparison with a given algorithm on 30D optimization under $f_{c_1} - f_{c_{30}}$.

| Algorithm | Benchmarks on which our algorithm obtain best or tie best performance comparing with a given algorithm |
|---|---|
| LSHADE | $f_{c_1} - f_{c_4}, f_{c_6} - f_{c_8}, f_{c_{11}} - f_{c_{12}}, f_{c_{14}} - f_{c_{16}}, f_{c_{18}}, f_{c_{20}}, f_{c_{22}} - f_{c_{27}}, f_{c_{29}} - f_{c_{30}}$ |
| jSO | $f_{c_1} - f_{c_{10}}, f_{c_{12}} - f_{c_{15}}, f_{c_{18}} - f_{c_{30}}$ |
| Hip-DE | $f_{c_1} - f_{c_5}, f_{c_{11}} - f_{c_{30}}$ |
| CS-DE | $f_{c_1} - f_{c_{27}}, f_{c_{29}} - f_{c_{30}}$ |
| TDE | $f_{c_1} - f_{c_{23}}, f_{c_{25}} - f_{c_{30}}$ |

**Table 17**
A certain algorithm obtains the best or tier best performance on 30D optimization under $f_{c_1} - f_{c_{30}}$.

| Algorithm | Benchmarks on which a given algorithm obtain best or similar performance comparing with other algorithms | Total |
|---|---|---|
| LSHADE | $f_{c_5}, f_{c_{10}}, f_{c_{13}}, f_{c_{17}}, f_{c_{19}}, f_{c_{21}}$ | 6 |
| jSO | $f_{c_{11}}, f_{c_{16}}$ | 2 |
| Hip-DE | $f_{c_6} - f_{c_9}$ | 4 |
| CS-DE | $f_{c_{28}}$ | 1 |
| TDE | $f_{c_{24}}$ | 1 |
| Our algorithm | $f_{c_1} - f_{c_4}, f_{c_{12}}, f_{c_{14}} - f_{c_{15}}, f_{c_{18}}, f_{c_{20}}, f_{c_{22}}, f_{c_{23}}, f_{c_{25}} - f_{c_{27}}, f_{c_{29}} - f_{c_{30}}$ | 16 |

**Table 18**
The analysis of $m$ in the weighting strategy on 30D optimization under CEC2017 test suite. The default settings are $m_{\min} = 1$ and $m_{\max} = 4$.

| A given setting versus the default setting under the CEC2017 test suite. | | | | | | |
|---|---|---|---|---|---|---|
| $[m_{min}, m_{max}]$ | [1, 1] | [1, 2] | [1, 3] | [1, 5] | [1, 6] | default: [1, 4] |
| NO. | Mean/Std. | Mean/Std. | Mean/Std. | Mean/Std. | Mean/Std. | Mean/Std. |
| $f_{c_1}$ | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0* |
| $f_{c_2}$ | 5.5729E−15/1.1397E−14(<) | 2.7864E−15/8.5358E−15(<) | 5.5729E−15/1.1397E−14(<) | 2.7864E−15/8.5358E−15(<) | *2.2292E−15/7.7172E−15(≈)* | *2.2292E−15/7.7172E−15* |
| $f_{c_3}$ | 1.1146E−15/7.9597E−15(<) | *0/0(≈)* | 1.1146E−15/7.9597E−15(<) | *0/0(≈)* | *0/0(≈)* | *0/0* |
| $f_{c_4}$ | 5.8562E+01/2.4117E−14(<) | **3.0106E+01/5.1579E+00(>)** | 3.0179E+01/1.0982E+00(>) | 3.1908E+01/5.8014E+00(<) | 3.0971E+01/2.3453E+00(>) | 3.1180E+01/2.5599E+00 |
| $f_{c_5}$ | **5.9139E+00/1.3956E+00(>)** | 6.9889E+00/1.6546E+00(<) | 6.5627E+00/1.2109E+00(<) | 6.9774E+00/1.4619E+00(<) | 7.2713E+00/1.6093E+00(<) | 6.5517E+00/1.5637E+00 |
| $f_{c_6}$ | 1.6426E−06/3.4671E−06(<) | 4.6507E−07/2.1871E−06(<) | 1.3690E−06/3.0501E−06(<) | 1.8868E−07/9.1566E−07(<) | 1.8666E−07/9.1589E−07(<) | **1.3446E−07/6.8280E−07** |
| $f_{c_7}$ | **3.7329E+01/1.3512E+00(>)** | 3.8160E+01/1.3790E+00(<) | 3.7489E+01/1.4577E+00(>) | 3.8239E+01/1.1951E+00(<) | 3.8252E+01/1.6123E+00(<) | 3.7745E+01/1.4634E+00 |
| $f_{c_8}$ | 6.7166E+00/1.3581E+00(>) | 7.6727E+00/1.8224E+00(<) | **6.5700E+00/1.5396E+00(>)** | 7.7589E+00/1.4748E+00(<) | 8.0624E+00/1.7655E+00(<) | 7.6593E+00/1.6710E+00 |
| $f_{c_9}$ | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0* |
| $f_{c_{10}}$ | 1.3401E+03/2.3998E+02(>) | 1.3878E+03/2.1463E+02(>) | **1.3307E+03/2.1367E+02(>)** | 1.4375E+03/2.1244E+02(<) | 1.4414E+03/2.0761E+02(<) | 1.4325E+03/2.0577E+02 |
| $f_{c_{11}}$ | 1.2174E+01/2.0047E+01(<) | **2.6372E+00/1.8125E+00(>)** | 9.5140E+00/1.7232E+01(<) | 8.8743E+00/1.7672E+01(<) | 4.1854E+00/8.3435E+00(>) | 8.6379E+00/1.8032E+01 |
| $f_{c_{12}}$ | 2.4950E+02/1.4897E+02(<) | **9.2044E+01/7.0507E+01(>)** | 2.0581E+02/1.5373E+02(<) | 1.0183E+02/8.3460E+01(<) | 9.7524E+01/7.4238E+01(<) | 9.6268E+01/8.5502E+01 |
| $f_{c_{13}}$ | 1.4405E+01/5.9623E+00(<) | 1.3896E+01/4.6965E+00(>) | 1.4343E+01/6.5366E+00(<) | **1.3271E+01/6.3971E+00(>)** | 1.4520E+01/4.8913E+00(<) | 1.4029E+01/5.6635E+00 |
| $f_{c_{14}}$ | 2.1659E+01/4.4644E+00(<) | 2.0543E+01/4.9057E+00(<) | 2.1999E+01/1.2760E+00(<) | 2.1281E+01/4.1810E+00(<) | **1.8632E+01/6.9426E+00(>)** | 1.9595E+01/6.6411E+00 |
| $f_{c_{15}}$ | 2.4617E+00/1.4661E+00(<) | 1.2870E+00/9.6313E−01(<) | 2.7015E+00/2.1802E+00(<) | 1.1402E+00/8.1653E−01(<) | 1.1009E+00/7.5256E−01(<) | **1.0886E+00/7.2661E−01** |
| $f_{c_{16}}$ | 3.5747E+01/5.0141E+01(>) | 6.4535E+01/6.6296E+01(>) | **2.3211E+01/1.7882E+01(>)** | 8.1511E+01/9.3708E+01(<) | 5.5282E+01/6.4016E+01(>) | 7.1095E+01/8.6289E+01 |
| $f_{c_{17}}$ | 2.6963E+01/6.3277E+00(>) | 2.6021E+01/6.3429E+00(>) | 2.7908E+01/6.5273E+00(<) | **2.5502E+01/7.1508E+00(>)** | 2.6664E+01/6.9751E+00(>) | 2.7271E+01/6.4240E+00 |
| $f_{c_{18}}$ | 2.0884E+01/7.0104E−01(<) | 2.0576E+01/1.5379E−01(<) | 2.0422E+01/2.9108E+00(>) | 2.0587E+01/2.0528E−01(<) | **2.0220E+01/2.8258E+00(>)** | 2.0557E+01/8.2141E−02 |
| $f_{c_{19}}$ | 4.6804E+00/1.3839E+00(<) | 4.3395E+00/1.6560E+00(>) | 4.7655E+00/1.5164E+00(<) | 4.4266E+00/1.6522E+00(>) | **4.0814E+00/1.5875E+00(>)** | 4.4763E+00/1.8698E+00 |
| $f_{c_{20}}$ | 2.4495E+01/7.1612E+00(>) | 2.8077E+01/6.7841E+00(<) | **2.4451E+01/6.1019E+00(>)** | 2.8921E+01/6.7935E+00(<) | 3.0630E+01/1.8362E+01(<) | 2.6557E+01/6.4134E+00 |
| $f_{c_{21}}$ | 2.0701E+02/1.3202E+00(>) | 2.0756E+02/1.4671E+00(<) | **2.0698E+02/1.4368E+00(>)** | 2.0741E+02/1.5162E+00(<) | 2.0752E+02/1.8666E+00(<) | 2.0705E+02/1.3145E+00 |
| $f_{c_{22}}$ | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/6.3901E−14(≈)* | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/6.3901E−14(≈)* | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/6.3901E−14* |
| $f_{c_{23}}$ | 3.5026E+02/3.0048E+00(<) | 3.4494E+02/3.4735E+00(<) | 3.5034E+02/3.0353E+00(<) | 3.4467E+02/3.2264E+00(<) | **3.4380E+02/2.9167E+00(>)** | 3.4404E+02/3.2446E+00 |
| $f_{c_{24}}$ | 4.2573E+02/1.8619E+00(<) | 4.2142E+02/2.5187E+00(<) | 4.2538E+02/1.4069E+00(<) | **4.2043E+02/2.8191E+00(>)** | 4.2060E+02/2.4249E+00(>) | 4.2141E+02/2.3859E+00 |
| $f_{c_{25}}$ | 3.8670E+02/1.0499E−02(<) | 3.8557E+02/3.9916E−01(<) | 3.8586E+02/4.2642E−01(<) | 3.8579E+02/3.5626E−01(<) | 3.8560E+02/4.8574E−01(<) | **3.8556E+02/4.4299E−01** |
| $f_{c_{26}}$ | 9.1495E+02/4.1736E+01(<) | 8.5055E+02/7.3274E+01(>) | 9.0880E+02/3.6754E+01(<) | **8.5003E+02/4.8099E+01(>)** | 8.5878E+02/4.0556E+01(>) | 8.6547E+02/9.0427E+01 |
| $f_{c_{27}}$ | 5.0436E+02/4.6336E+00(<) | 4.9067E+02/9.8176E+00(>) | 5.0429E+02/4.6418E+00(<) | **4.8842E+02/7.9077E+00(>)** | 4.9112E+02/9.9273E+00(<) | 4.9074E+02/7.9930E+00 |
| $f_{c_{28}}$ | *3.0831E+02/2.8800E+01(>)* | *3.0831E+02/2.8799E+01(>)* | 3.1459E+02/3.6999E+01(<) | 3.1054E+02/3.2343E+01(>) | 3.1117E+02/3.4222E+01(>) | 3.1320E+02/3.6528E+01 |
| $f_{c_{29}}$ | 4.3109E+02/8.2984E+00(<) | 4.3026E+02/5.2994E+00(<) | 4.3191E+02/8.8855E+00(<) | 4.3016E+02/7.3518E+00(<) | 4.3038E+02/5.2898E+00(<) | **4.2941E+02/6.0039E+00** |
| $f_{c_{30}}$ | 1.9782E+03/3.3671E+01(<) | 1.9349E+03/2.6326E+01(>) | **1.9128E+03/3.4045E+01(>)** | 1.9368E+03/3.3938E+01(<) | 1.9349E+03/3.0419E+01(>) | 1.9353E+03/2.2151E+01 |
| >/≈/< | 9/3/18 | 12/4/14 | 9/3/18 | 7/4/19 | 12/5/13 | –/–/– |

**Table 19**
Comparison results between constant settings of $m$ ($m = 1$, $m = 2$) and our default setting ($m = 1 \sim 4$) on 30D optimization under CEC2017 $f_{c_1}$ - $f_{c_{30}}$.

| A given constant setting versus the default setting. | | | |
|---|---|---|---|
| $m$ | $m = 1$ | $m = 2$ | default |
| NO. | Mean/Std. | Mean/Std. | Mean/Std. |
| $f_{c_1}$ | *0/0(≈)* | *0/0(≈)* | *0/0* |
| $f_{c_2}$ | 1.2260E−14/1.7293E−14(<) | 5.5729E−15/1.1397E−14(<) | **2.7864E−15/8.5358E−15** |
| $f_{c_3}$ | 1.1146E−15/7.9597E−15(<) | 1.1146E−15/7.9597E−15(<) | **0/0** |
| $f_{c_4}$ | 3.0799E+01/3.7338E+00(<) | 5.8562E+01/2.4117E−14(<) | **3.0368E+01/7.9033E+00** |
| $f_{c_5}$ | 6.5656E+00/1.6224E+00(<) | **5.9139E+00/1.3956E+00(>)** | 7.1670E+00/1.5422E+00 |
| $f_{c_6}$ | 6.4825E−06/8.7293E−06(<) | 1.6426E−06/3.4671E−06(<) | **9.2326E−08/6.5447E−07** |
| $f_{c_7}$ | **3.6970E+01/1.2222E+00(>)** | 3.7329E+01/1.3512E+00(>) | 3.8358E+01/1.2269E+00 |
| $f_{c_8}$ | 7.0598E+00/1.7968E+00(>) | **6.7166E+00/1.3581E+00(>)** | 8.0305E+00/1.6725E+00 |
| $f_{c_9}$ | *0/0(≈)* | *0/0(≈)* | *0/0* |
| $f_{c_{10}}$ | 1.4449E+03/2.0156E+02(<) | **1.3401E+03/2.3998E+02(>)** | 1.4052E+03/2.3353E+02 |
| $f_{c_{11}}$ | 7.7414E+00/1.6098E+01(<) | 1.2174E+01/2.0047E+01(<) | **6.4434E+00/1.2370E+01** |
| $f_{c_{12}}$ | 1.0626E+02/9.1908E+01(<) | 2.4950E+02/1.4897E+02(<) | **8.7751E+01/7.5130E+01** |
| $f_{c_{13}}$ | 1.3139E+01/5.7530E+00(<) | 1.4405E+01/5.9623E+00(<) | **1.3138E+01/6.0085E+00** |
| $f_{c_{14}}$ | 2.1078E+01/3.5263E+00(<) | 2.1659E+01/4.4644E+00(<) | **2.0448E+01/5.3070E+00** |
| $f_{c_{15}}$ | **1.0147E+00/7.7094E−01(>)** | 2.4617E+00/1.4661E+00(<) | 1.1042E+00/8.5681E−01 |
| $f_{c_{16}}$ | 7.7568E+01/7.1024E+01(<) | **3.5747E+01/5.0141E+01(>)** | 6.2728E+01/6.5293E+01 |
| $f_{c_{17}}$ | 2.5456E+01/7.4802E+00(<) | 2.6963E+01/6.3277E+00(<) | **2.4296E+01/6.2151E+00** |
| $f_{c_{18}}$ | **1.9774E+01/3.9376E+00(>)** | 2.0884E+01/7.0104E−01(<) | 2.0205E+01/2.3373E+00 |
| $f_{c_{19}}$ | **3.8414E+00/1.2757E+00(>)** | 4.6804E+00/1.3839E+00(<) | 4.1399E+00/1.3862E+00 |
| $f_{c_{20}}$ | 2.8558E+01/7.6393E+00(<) | **2.4495E+01/7.1612E+00(>)** | 2.8539E+01/8.4673E+00 |
| $f_{c_{21}}$ | **2.0658E+02/1.5627E+00(>)** | 2.0701E+02/1.3202E+00(>) | 2.0767E+02/1.7816E+00 |
| $f_{c_{22}}$ | *1.0000E+02/8.9787E−14(≈)* | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/6.3901E−14* |
| $f_{c_{23}}$ | **3.4415E+02/2.8038E+00(>)** | 3.5026E+02/3.0048E+00(<) | 3.4512E+02/2.4712E+00 |
| $f_{c_{24}}$ | 4.2157E+02/2.6066E+00(<) | 4.2573E+02/1.8619E+00(<) | **4.2124E+02/2.1478E+00** |
| $f_{c_{25}}$ | 3.8584E+02/4.0721E−01(<) | 3.8670E+02/1.0499E−02(<) | **3.8567E+02/4.7920E−01** |
| $f_{c_{26}}$ | 8.7136E+02/4.5615E+01(<) | 9.1495E+02/4.1736E+01(<) | **8.6085E+02/4.3963E+01** |
| $f_{c_{27}}$ | **4.9023E+02/7.9081E+00(>)** | 5.0436E+02/4.6336E+00(<) | 4.9131E+02/8.9913E+00 |
| $f_{c_{28}}$ | **3.0426E+02/2.1320E+01(>)** | 3.0831E+02/2.8800E+01(>) | 3.1075E+02/3.2984E+01 |
| $f_{c_{29}}$ | 4.3035E+02/7.9204E+00(<) | 4.3109E+02/8.2984E+00(<) | **4.2796E+02/1.1900E+01** |
| $f_{c_{30}}$ | 1.9485E+03/3.0646E+01(<) | 1.9782E+03/3.3671E+01(<) | **1.9340E+03/2.9223E+01** |
| >/≈/< | 10/3/17 | 8/3/19 | –/–/– |

**Table 20**
Comparison between different value of $N$.

| A given setting of versus our default setting ($N = 2 \cdot D$). | | | | | | |
|---|---|---|---|---|---|---|
| $N$ | $N = 1 \cdot D$ | $N = 3 \cdot D$ | $N = 4 \cdot D$ | $N = 5 \cdot D$ | $N = 6 \cdot D$ | $N = 2 \cdot D$ |
| NO. | Mean/Std. | Mean/Std. | Mean/Std. | Mean/Std. | Mean/Std. | Mean/Std. |
| $f_{c_1}$ | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0* |
| $f_{c_2}$ | 1.6719E−15/6.7540E−15(<) | 1.2260E−14/1.7293E−14(<) | 3.3437E−15/9.2483E−15(<) | 2.2292E−15/7.7172E−15(<) | 1.6719E−15/6.7540E−15(<) | **1.1146E−15/5.5718E−15** |
| $f_{c_3}$ | *0/0(≈)* | 1.1146E−15/7.9597E−15(<) | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0* |
| $f_{c_4}$ | 3.1230E+01/6.3091E+00(<) | 3.0799E+01/3.7338E+00(<) | 3.1165E+01/2.4089E+00(<) | **2.9888E+01/3.9497E+00(>)** | 3.0727E+01/2.1985E+00(<) | 3.0553E+01/2.5993E+00 |
| $f_{c_5}$ | 7.4367E+00/1.9404E+00(<) | **6.5656E+00/1.6224E+00(>)** | 7.1687E+00/1.5626E+00(<) | 6.8897E+00/1.8718E+00(<) | 6.7071E+00/1.4615E+00(<) | 6.6155E+00/1.6288E+00 |
| $f_{c_6}$ | **2.4086E−08/1.5366E−07(>)** | 6.4825E−06/8.7293E−06(<) | 2.0807E−07/9.2416E−07(>) | 9.2326E−08/6.5447E−07(>) | 1.9894E−06/1.3400E−05(<) | 2.8512E−07/2.0118E−06 |
| $f_{c_7}$ | 3.8203E+01/1.3682E+00(<) | **3.6970E+01/1.2222E+00(>)** | 3.8188E+01/1.4183E+00(<) | 3.7647E+01/1.5455E+00(<) | 3.8181E+01/1.3811E+00(<) | 3.7344E+01/1.2870E+00 |
| $f_{c_8}$ | 7.7221E+00/1.6431E+00(<) | **7.0598E+00/1.7968E+00(>)** | 7.5488E+00/1.5052E+00(<) | 7.8945E+00/1.7301E+00(<) | 7.9713E+00/1.7426E+00(<) | 7.0987E+00/1.6990E+00 |
| $f_{c_9}$ | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0* |
| $f_{c_{10}}$ | 1.4222E+03/1.9160E+02(>) | 1.4449E+03/2.0156E+02(>) | **1.4170E+03/1.7125E+02(>)** | 1.4492E+03/1.9152E+02(>) | 1.4415E+03/2.0427E+02(>) | 1.4634E+03/2.5788E+02 |
| $f_{c_{11}}$ | 4.6876E+00/8.3673E+00(<) | 7.7414E+00/1.6098E+01(<) | 6.2712E+00/1.4102E+01(<) | 7.0478E+00/1.4013E+01(<) | 9.8766E+00/1.9129E+01(<) | **3.0911E+00/2.2560E+00** |
| $f_{c_{12}}$ | 1.0704E+02/8.0028E+01(<) | 1.0626E+02/9.1908E+01(<) | 9.6091E+01/7.7756E+01(<) | 9.6594E+01/8.9048E+01(<) | 1.0164E+02/8.3614E+01(<) | **9.4810E+01/6.8389E+01** |
| $f_{c_{13}}$ | 1.4774E+01/4.5350E+00(<) | 1.3139E+01/5.7530E+00(<) | 1.2290E+01/6.7317E+00(<) | 1.1929E+01/7.2476E+00(<) | 1.3085E+01/5.6097E+00(<) | **1.1612E+01/6.5918E+00** |
| $f_{c_{14}}$ | 2.0581E+01/5.5380E+00(>) | 2.1078E+01/3.5263E+00(<) | 1.9814E+01/6.4607E+00(>) | 2.0339E+01/5.4702E+00(>) | **1.9166E+01/7.0341E+00(>)** | 2.0681E+01/4.8234E+00 |
| $f_{c_{15}}$ | **9.2128E−01/9.0863E−01(>)** | 1.0147E+00/7.7094E−01(>) | 1.0553E+00/5.9761E−01(>) | 1.2621E+00/8.3554E−01(<) | 1.0760E+00/6.4497E−01(>) | 1.2575E+00/7.4841E−01 |
| $f_{c_{16}}$ | 5.5974E+01/5.3805E+01(<) | 7.7568E+01/7.1024E+01(<) | 6.2919E+01/6.7425E+01(<) | 5.2091E+01/5.9857E+01(<) | 6.7200E+01/7.0495E+01(<) | **3.4698E+01/3.6556E+01** |
| $f_{c_{17}}$ | 2.5863E+01/7.3272E+00(>) | 2.5456E+01/7.4802E+00(>) | 2.7827E+01/5.6110E+00(<) | **2.4274E+01/6.2150E+00(>)** | 2.5055E+01/6.8253E+00(>) | 2.6736E+01/6.0414E+00 |
| $f_{c_{18}}$ | 2.0172E+01/2.8139E+00(<) | **1.9774E+01/3.9376E+00(>)** | 1.9795E+01/3.9384E+00(>) | 2.0568E+01/1.5248E−01(<) | 1.9798E+01/3.9454E+00(>) | 2.0171E+01/2.8176E+00 |
| $f_{c_{19}}$ | 4.5746E+00/1.4871E+00(<) | **3.8414E+00/1.2757E+00(>)** | 4.2665E+00/1.5500E+00(<) | 4.4454E+00/1.8239E+00(<) | 4.5837E+00/1.7827E+00(<) | 4.1502E+00/1.6999E+00 |
| $f_{c_{20}}$ | 3.0276E+01/8.0248E+00(<) | 2.8558E+01/7.6393E+00(<) | 2.8532E+01/6.3490E+00(<) | 2.8369E+01/6.5820E+00(<) | 2.8207E+01/5.9989E+00(<) | **2.7842E+01/7.2002E+00** |
| $f_{c_{21}}$ | 2.0781E+02/1.5586E+00(<) | **2.0658E+02/1.5627E+00(>)** | 2.0777E+02/1.6389E+00(<) | 2.0709E+02/1.6554E+00(>) | 2.0744E+02/1.3463E+00(<) | 2.0710E+02/1.7509E+00 |
| $f_{c_{22}}$ | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/8.9787E−14(≈)* | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/1.0808E−13* |
| $f_{c_{23}}$ | 3.4382E+02/3.8009E+00(>) | 3.4415E+02/2.8038E+00(>) | 3.4385E+02/3.7461E+00(>) | **3.4381E+02/3.1235E+00(>)** | 3.4429E+02/3.3549E+00(>) | 3.4570E+02/3.9223E+00 |
| $f_{c_{24}}$ | 4.2140E+02/2.3967E+00(>) | 4.2157E+02/2.6066E+00(>) | **4.2047E+02/2.7141E+00(>)** | 4.2115E+02/3.2155E+00(>) | 4.2134E+02/2.8699E+00(>) | 4.2171E+02/2.6368E+00 |
| $f_{c_{25}}$ | 3.8570E+02/4.4761E−01(<) | 3.8584E+02/4.0721E−01(<) | 3.8568E+02/4.3278E−01(<) | 3.8574E+02/4.5644E−01(<) | 3.8564E+02/4.5004E−01(<) | **3.8552E+02/4.0968E−01** |
| $f_{c_{26}}$ | 8.6071E+02/6.2529E+01(<) | 8.7136E+02/4.5615E+01(<) | 8.6787E+02/6.2302E+01(<) | 8.8006E+02/5.2381E+01(<) | **8.5946E+02/5.2832E+01(>)** | 8.6058E+02/6.4010E+01 |
| $f_{c_{27}}$ | 4.9142E+02/9.2261E+00(<) | **4.9023E+02/7.9081E+00(>)** | 4.9202E+02/9.6212E+00(<) | 4.9095E+02/1.0093E+01(>) | 4.9030E+02/7.4274E+00(>) | 4.9111E+02/1.0051E+01 |
| $f_{c_{28}}$ | 3.1480E+02/3.7538E+01(<) | **3.0426E+02/2.1320E+01(>)** | 3.1236E+02/3.4215E+01(<) | 3.1117E+02/3.4221E+01(<) | 3.1299E+02/3.5963E+01(<) | 3.0873E+02/3.0245E+01 |
| $f_{c_{29}}$ | 4.2907E+02/7.0706E+00(>) | 4.3035E+02/7.9204E+00(<) | **4.2890E+02/8.6657E+00(>)** | 4.3001E+02/7.9033E+00(>) | 4.2985E+02/6.8361E+00(>) | 4.3029E+02/8.2594E+00 |
| $f_{c_{30}}$ | 1.9330E+03/3.2178E+01(<) | 1.9485E+03/3.0646E+01(<) | 1.9355E+03/3.1817E+01(<) | 1.9377E+03/2.3871E+01(<) | 1.9341E+03/2.5150E+01(<) | **1.9264E+03/2.7002E+01** |
| >/≈/< | 8/4/18 | 13/3/14 | 8/4/18 | 10/4/16 | 10/4/16 | –/–/– |

**Table 21**
Comparison between different value of $\xi$.

| A given setting of versus our default setting ($\xi = 0.001$). $\xi$ | $\xi = 0.0005$ | $\xi = 0.0015$ | $\xi = 0.002$ | $\xi = 0.0025$ | $\xi = 0.001$ |
|---|---|---|---|---|---|
| NO. | Mean/Std. | Mean/Std. | Mean/Std. | Mean/Std. | Mean/Std. |
| $f_{c_1}$ | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0* |
| $f_{c_2}$ | 2.7864E−15/8.5358E−15(<) | 2.7864E−15/8.5358E−15(<) | *1.1146E−15/5.5718E−15(≈)* | 2.2292E−15/7.7172E−15(<) | *1.1146E−15/5.5718E−15* |
| $f_{c_3}$ | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0* |
| $f_{c_4}$ | 3.1386E+01/6.6488E+00(<) | 3.0681E+01/1.9072E+00(<) | 3.0097E+01/6.9974E+00(<) | 3.0431E+01/2.2364E+00(<) | **3.0055E+01/2.7886E+00** |
| $f_{c_5}$ | 7.0830E+00/1.5660E+00(<) | 7.0720E+00/1.4702E+00(<) | 7.3060E+00/1.5839E+00(<) | 7.5012E+00/1.4016E+00(<) | **6.8588E+00/1.5954E+00** |
| $f_{c_6}$ | 1.1574E−07/6.6898E−07(>) | 2.8579E−07/2.0117E−06(<) | **1.0923E−13/2.2287E−14(>)** | 8.0344E−09/4.2403E−08(>) | 1.8331E−07/9.1636E−07 |
| $f_{c_7}$ | 3.8074E+01/1.4788E+00(>) | 3.8143E+01/1.4200E+00(<) | **3.7992E+01/1.4934E+00(>)** | 3.8387E+01/1.2524E+00(<) | 3.8123E+01/1.1386E+00 |
| $f_{c_8}$ | 8.3670E+00/1.6248E+00(<) | 8.0791E+00/1.7117E+00(<) | 7.8993E+00/1.4318E+00(<) | 7.7689E+00/1.4101E+00(<) | **7.2358E+00/1.5675E+00** |
| $f_{c_9}$ | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0* |
| $f_{c_{10}}$ | 1.4510E+03/1.8201E+02(<) | 1.4137E+03/1.9485E+02(<) | 1.3843E+03/1.8219E+02(<) | 1.3931E+03/2.1932E+02(<) | **1.3757E+03/2.1985E+02** |
| $f_{c_{11}}$ | 9.5201E+00/1.7687E+01(<) | 7.9213E+00/1.6171E+01(<) | **6.0582E+00/1.1838E+01(>)** | 7.2064E+00/1.6233E+01(<) | 6.3433E+00/1.3834E+01 |
| $f_{c_{12}}$ | 9.4509E+01/7.5013E+01(>) | **8.8312E+01/7.1148E+01(>)** | 1.0285E+02/7.9324E+01(>) | 1.0776E+02/7.6767E+01(<) | 1.0774E+02/7.7489E+01 |
| $f_{c_{13}}$ | 1.3344E+01/5.8590E+00(<) | 1.2688E+01/5.9645E+00(<) | **1.1664E+01/6.4723E+00(>)** | 1.2644E+01/6.2377E+00(<) | 1.2172E+01/7.0661E+00 |
| $f_{c_{14}}$ | 2.0872E+01/4.6620E+00(<) | **1.9172E+01/6.9414E+00(>)** | 2.0960E+01/4.7997E+00(<) | 2.0053E+01/5.9555E+00(>) | 2.0115E+01/6.2967E+00 |
| $f_{c_{15}}$ | 1.2351E+00/8.6054E−01(>) | **1.1051E+00/6.7444E−01(>)** | 1.2281E+00/8.7452E−01(>) | 1.2679E+00/8.3475E−01(>) | 1.3904E+00/9.1432E−01 |
| $f_{c_{16}}$ | 7.0064E+01/7.1548E+01(<) | 7.5518E+01/7.7478E+01(<) | **5.2198E+01/5.9090E+01(>)** | 6.3667E+01/6.6539E+01(<) | 6.1723E+01/6.8581E+01 |
| $f_{c_{17}}$ | 2.5415E+01/6.1615E+00(<) | 2.5081E+01/6.3250E+00(<) | 2.4750E+01/7.1910E+00(<) | 2.6882E+01/6.6407E+00(<) | **2.4394E+01/6.8513E+00** |
| $f_{c_{18}}$ | 2.0541E+01/5.9907E−02(>) | 2.0236E+01/2.8394E+00(>) | 2.0568E+01/1.4962E−01(<) | **2.0194E+01/2.8212E+00(>)** | 2.0559E+01/1.4852E−01 |
| $f_{c_{19}}$ | 4.1842E+00/1.6578E+00(>) | 4.2134E+00/1.8657E+00(>) | 4.2713E+00/1.7026E+00(>) | **3.9472E+00/1.4011E+00(>)** | 4.6601E+00/1.5061E+00 |
| $f_{c_{20}}$ | 2.7942E+01/7.0552E+00(>) | 2.8178E+01/6.2245E+00(>) | **2.6842E+01/8.2072E+00(>)** | 2.9366E+01/8.8214E+00(<) | 2.8659E+01/7.9773E+00 |
| $f_{c_{21}}$ | 2.0752E+02/1.6271E+00(<) | 2.0751E+02/1.8037E+00(<) | 2.0760E+02/1.4374E+00(<) | 2.0718E+02/1.6226E+00(<) | **2.0714E+02/1.7946E+00** |
| $f_{c_{22}}$ | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/1.4352E−14* |
| $f_{c_{23}}$ | 3.4380E+02/3.4561E+00(>) | 3.4413E+02/4.1239E+00(<) | **3.4336E+02/3.2420E+00(>)** | 3.4416E+02/3.0245E+00(<) | 3.4402E+02/3.1596E+00 |
| $f_{c_{24}}$ | 4.2067E+02/2.7295E+00(<) | 4.2112E+02/2.4264E+00(<) | 4.2098E+02/2.4639E+00(<) | 4.2097E+02/2.7935E+00(<) | **4.2053E+02/2.0158E+00** |
| $f_{c_{25}}$ | 3.8582E+02/4.4738E−01(<) | 3.8578E+02/4.2435E−01(<) | 3.8574E+02/4.2837E−01(<) | 3.8580E+02/3.8955E−01(<) | **3.8569E+02/3.8482E−01** |
| $f_{c_{26}}$ | 8.7469E+02/4.6080E+01(<) | 8.5789E+02/5.2181E+01(>) | 8.6107E+02/4.9816E+01(<) | **8.4827E+02/6.0671E+01(>)** | 8.5956E+02/4.7685E+01 |
| $f_{c_{27}}$ | 4.9042E+02/9.3460E+00(>) | **4.9021E+02/7.9311E+00(>)** | 4.9147E+02/7.7317E+00(<) | 4.9117E+02/9.1341E+00(<) | 4.9052E+02/8.6781E+00 |
| $f_{c_{28}}$ | 3.1480E+02/3.7533E+01(<) | 3.1096E+02/3.3605E+01(<) | 3.0649E+02/2.6264E+01(<) | 3.1257E+02/3.4805E+01(<) | **3.0578E+02/3.5393E+01** |
| $f_{c_{29}}$ | 4.2905E+02/6.4315E+00(<) | 4.2812E+02/6.7248E+00(<) | 4.2834E+02/7.8150E+00(<) | 4.3082E+02/6.0382E+00(<) | **4.2764E+02/5.6985E+00** |
| $f_{c_{30}}$ | **1.9384E+03/2.5511E+01(>)** | 1.9443E+03/3.2106E+01(<) | 1.9440E+03/2.8992E+01(<) | 1.9442E+03/2.4100E+01(<) | 1.9391E+03/1.8233E+01 |
| >/≈/< | 10/4/16 | 8/4/18 | 10/5/15 | 6/4/20 | –/–/– |

**Table 22**
Comparison between our algorithm and its variants.

| Variants | Variants-1 | Variants-2 | Variants-3 | Our algorithms |
|---|---|---|---|---|
| $f_{c_1}$ | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0* |
| $f_{c_2}$ | 1.1146E−15/5.5718E−15(>) | 1.6719E−15/6.7540E−15(>) | **5.5729E−16/3.9798E−15(>)** | 2.7864E−15/8.5358E−15 |
| $f_{c_3}$ | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0* |
| $f_{c_4}$ | 3.0323E+01/5.3508E+00(>) | 3.1106E+01/6.4718E+00(<) | **2.9819E+01/4.1477E+00(>)** | 3.0753E+01/2.5032E+00 |
| $f_{c_5}$ | 7.2478E+00/1.6489E+00(<) | 7.0875E+00/1.5865E+00(<) | 7.1785E+00/1.7241E+00(<) | **7.0637E+00/1.6343E+00** |
| $f_{c_6}$ | 9.4339E−08/6.5444E−07(>) | **6.7106E−10/4.7915E−09(>)** | 2.9972E−07/1.1152E−06(>) | 3.6331E−07/1.5332E−06 |
| $f_{c_7}$ | 3.8311E+01/1.4435E+00(<) | 3.8316E+01/1.5044E+00(<) | 3.8040E+01/1.4989E+00(<) | **3.7898E+01/1.5722E+00** |
| $f_{c_8}$ | 7.4268E+00/1.5147E+00(>) | 7.7766E+00/1.3377E+00(<) | **7.3547E+00/1.5592E+00(>)** | 7.5079E+00/1.7624E+00 |
| $f_{c_9}$ | *0/0(≈)* | *0/0(≈)* | *0/0(≈)* | *0/0* |
| $f_{c_{10}}$ | 1.4991E+03/2.4704E+02(<) | **1.3211E+03/2.3405E+02(>)** | 1.4176E+03/1.9951E+02(>) | 1.4263E+03/1.9201E+02 |
| $f_{c_{11}}$ | 4.5904E+00/8.2236E+00(>) | **4.3540E+00/8.5743E+00(>)** | 6.9779E+00/2.1640E+01(>) | 7.9305E+00/1.6426E+01 |
| $f_{c_{12}}$ | **8.0785E+01/7.5019E+01(>)** | 1.1636E+02/7.7809E+01(<) | 9.6293E+01/8.0228E+01(<) | 9.0902E+01/6.0188E+01 |
| $f_{c_{13}}$ | 1.4012E+01/5.1115E+00(<) | 1.3456E+01/5.5302E+00(<) | 1.2875E+01/6.1209E+00(<) | **1.1921E+01/6.7109E+00** |
| $f_{c_{14}}$ | **1.8766E+01/7.4089E+00(>)** | 2.0276E+01/5.3396E+00(<) | 2.0186E+01/5.9947E+00(<) | 1.9998E+01/6.3340E+00 |
| $f_{c_{15}}$ | 1.5302E+00/9.8232E−01(<) | **1.0463E+00/7.1352E−01(>)** | 1.2703E+00/8.8486E−01(>) | 1.2896E+00/9.6177E−01 |
| $f_{c_{16}}$ | 8.7084E+01/9.1348E+01(<) | 7.2459E+01/8.0773E+01(<) | 6.3929E+01/6.7793E+01(<) | **5.2237E+01/6.0326E+01** |
| $f_{c_{17}}$ | **2.4942E+01/5.7675E+00(>)** | 2.6327E+01/7.2787E+00(<) | 2.6067E+01/6.1322E+00(<) | 2.5453E+01/5.0701E+00 |
| $f_{c_{18}}$ | 2.0244E+01/2.8373E+00(<) | 2.0618E+01/2.2779E−01(<) | **2.0170E+01/2.7771E+00(>)** | 2.0191E+01/2.8190E+00 |
| $f_{c_{19}}$ | 4.3064E+00/1.6456E+00(>) | 4.3204E+00/1.6421E+00(>) | **4.1694E+00/1.7651E+00(>)** | 4.4074E+00/1.5908E+00 |
| $f_{c_{20}}$ | 2.9553E+01/5.5369E+00(<) | **2.6916E+01/7.1016E+00(>)** | 2.7318E+01/8.3800E+00(<) | 2.7187E+01/7.7068E+00 |
| $f_{c_{21}}$ | 2.0745E+02/1.5339E+00(<) | **2.0718E+02/1.5216E+00(>)** | 2.0732E+02/1.7246E+00(<) | 2.0721E+02/1.5489E+00 |
| $f_{c_{22}}$ | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/6.3901E−14(≈)* | *1.0000E+02/1.4352E−14(≈)* | *1.0000E+02/1.4352E−14* |
| $f_{c_{23}}$ | 3.4393E+02/3.4107E+00(<) | 3.4382E+02/3.1022E+00(<) | 3.4377E+02/3.1730E+00(<) | **3.4359E+02/3.5593E+00** |
| $f_{c_{24}}$ | 4.2284E+02/2.5512E+00(<) | 4.2139E+02/2.0261E+00(<) | 4.2114E+02/2.3518E+00(<) | **4.2054E+02/2.2818E+00** |
| $f_{c_{25}}$ | 3.8579E+02/3.9215E−01(>) | 3.8584E+02/4.3319E−01(>) | **3.8573E+02/4.9612E−01(>)** | 3.8589E+02/3.6298E−01 |
| $f_{c_{26}}$ | 8.6567E+02/8.0034E+01(<) | 8.6150E+02/4.1689E+01(<) | 8.5390E+02/5.8546E+01(<) | **8.5372E+02/6.0958E+01** |
| $f_{c_{27}}$ | 4.9241E+02/7.6963E+00(<) | 4.9134E+02/8.8214E+00(<) | 4.9151E+02/8.7542E+00(<) | **4.9129E+02/9.1523E+00** |
| $f_{c_{28}}$ | 3.1520E+02/3.6526E+01(<) | **3.0873E+02/3.0245E+01(>)** | 3.1320E+02/3.6523E+01(<) | 3.1096E+02/3.3605E+01 |
| $f_{c_{29}}$ | 4.3030E+02/5.5328E+00(<) | **4.2775E+02/6.7386E+00(>)** | 4.2893E+02/5.4169E+00(>) | 4.2906E+02/7.2819E+00 |
| $f_{c_{30}}$ | 1.9548E+03/2.7934E+01(<) | **1.9400E+03/2.2654E+01(>)** | 1.9528E+03/3.1498E+01(<) | 1.9419E+03/2.8002E+01 |
| >/≈/< | 10/4/16 | 12/4/14 | 11/4/15 | –/–/– |

**Table 23**
The time complexity comparison on benchmark $f_{a_{14}}$ according to the suggestion of CEC2013 competition.

| Algorithms | $T_0$ | $T_1$ | $\widehat{T}_2$ | $\frac{\widehat{T}_2 - T_1}{T_0}$ |
|---|---|---|---|---|
| LSHADE | 0.0583 | 0.6117 | 1.4834 | 14.9520 |
| jSO | | | 1.5076 | 15.3671 |
| Hip-DE | | | 1.9941 | 23.7118 |
| CS-DE | | | 4.0929 | 59.7118 |
| TDE | | | 1.7714 | 19.8919 |
| Our algorithm | | | 1.8397 | 21.0635 |

## References


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

[2] S.J. Nanda, G. Panda, A survey on nature inspired metaheuristic algorithms for partitional clustering, Swarm Evol. Comput. 16 (2014) 1–18.

[3] H. Wang, W. Wang, Z. Cui, X. Zhou, J. Zhao, Y. Li, A new dynamic firefly algorithm for demand estimation of water resources, Inform. Sci. 438 (2018) 95–106.

[4] Z. Meng, Dimension improvements based adaptation of control parameters in differential evolution: A fitness-value-independent approach, Expert Syst. Appl. 223 (2023) 119848.

[5] M. Pant, H. Zaheer, L. Garcia-Hernandez, A. Abraham, et al., Differential evolution: A review of more than two decades of research, Eng. Appl. Artif. Intell. 90 (2020) 103479.

[6] Z. Meng, J.-S. Pan, K.-K. Tseng, PaDE: An enhanced differential evolution algorithm with novel control parameter adaptation schemes for numerical optimization, Knowl.-Based Syst. 168 (2019) 80–99.

[7] J.-S. Pan, L.-G. Zhang, R.-B. Wang, V. Snášel, S.-C. Chu, Gannet optimization algorithm: A new metaheuristic algorithm for solving engineering optimization problems, Math. Comput. Simulation 202 (2022) 343–373.

[8] J. Brest, M.S. Maučec, Population size reduction for the differential evolution algorithm, Appl. Intell. 29 (3) (2008) 228–247.

[9] T.-Y. Wu, H. Li, S.-C. Chu, CPPE: An improved phasmatodea population evolution algorithm with chaotic maps, Mathematics 11 (9) (2023) 1977.

[10] S. Das, A. Abraham, U.K. Chakraborty, A. Konar, Differential evolution using a neighborhood-based mutation operator, IEEE Trans. Evol. Comput. 13 (3) (2009) 526–553.

[11] Z. Meng, J.-S. Pan, H. Xu, QUasi-Affine TRansformation Evolutionary (QUATRE) algorithm: a cooperative swarm based algorithm for global optimization, Knowl.-Based Syst. 109 (2016) 104–121.

[12] Z. Meng, J.-S. Pan, QUasi-Affine TRansformation Evolution with External ARchive (QUATRE-EAR): an enhanced structure for differential evolution, Knowl.-Based Syst. 155 (2018) 35–53.

[13] R. Tanabe, A.S. Fukunaga, Improving the search performance of SHADE using linear population size reduction, in: 2014 IEEE Congress on Evolutionary Computation, CEC, IEEE, 2014, pp. 1658–1665.

[14] G. Sun, G. Yang, G. Zhang, Two-level parameter cooperation-based population regeneration framework for differential evolution, Swarm Evol. Comput. 75 (2022) 101122.

[15] J. Brest, A. Zamuda, I. Fister, B. Boskovic, Some improvements of the self-adaptive jde algorithm, in: 2014 IEEE Symposium on Differential Evolution, SDE, IEEE, 2014, pp. 1–8.

[16] G. Sun, C. Li, L. Deng, An adaptive regeneration framework based on search space adjustment for differential evolution, Neural Comput. Appl. 33 (2021) 9503–9519.

[17] T.-Y. Wu, A. Shao, J.-S. Pan, CTOA: Toward a chaotic-based tumbleweed optimization algorithm, Mathematics 11 (10) (2023) 2339.

[18] N. Noman, H. Iba, Enhancing differential evolution performance with local search for high dimensional function optimization, in: Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation, ACM, 2005, pp. 967–974.

[19] Z. Meng, Y. Zhong, C. Yang, CS-DE: Cooperative strategy based differential evolution with population diversity enhancement, Inform. Sci. 577 (2021) 663–696.

[20] L.M. Zheng, S.X. Zhang, K.S. Tang, S.Y. Zheng, Differential evolution powered by collective information, Inform. Sci. 399 (2017) 13–29.

[21] J. Liao, Y. Cai, T. Wang, H. Tian, Y. Chen, Cellular direction information based differential evolution for numerical optimization: an empirical study, Soft Comput. 20 (7) (2016) 2801–2827.

[22] Y. Cai, G. Sun, T. Wang, H. Tian, Y. Chen, J. Wang, Neighborhood-adaptive differential evolution for global numerical optimization, Appl. Soft Comput. 59 (2017) 659–706.

[23] G. Sun, Y. Cai, T. Wang, H. Tian, C. Wang, Y. Chen, Differential evolution with individual-dependent topology adaptation, Inform. Sci. 450 (2018) 1–38.

[24] A.K. Qin, V.L. Huang, P.N. Suganthan, Differential evolution algorithm with strategy adaptation for global numerical optimization, IEEE Trans. Evol. Comput. 13 (2) (2008) 398–417.

[25] M. Tian, X. Gao, Differential evolution with neighborhood-based adaptive evolution mechanism for numerical optimization, Inform. Sci. 478 (2019) 422–448.

[26] G. Wu, R. Mallipeddi, P.N. Suganthan, Problem definitions and evaluation criteria for the CEC 2017 competition on constrained real-parameter optimization, Technical Report, IEEE, 2017.

[27] Z. Meng, J.-S. Pan, W.-m. Zheng, Differential evolution utilizing a handful top superior individuals with bionic bi-population structure for the enhancement of optimization performance, Enterp. Inf. Syst. (2018) 1–22.

[28] S.M. Islam, S. Das, S. Ghosh, S. Roy, P.N. Suganthan, An adaptive differential evolution algorithm with novel mutation and crossover strategies for global numerical optimization, IEEE Trans. Syst. Man Cybern. B 42 (2) (2011) 482–500.

[29] W.-J. Yu, M. Shen, W.-N. Chen, Z.-H. Zhan, Y.-J. Gong, Y. Lin, O. Liu, J. Zhang, Differential evolution with two-level parameter adaptation, IEEE Trans. Cybern. 44 (7) (2013) 1080–1099.

[30] Y. Cai, J. Liao, T. Wang, Y. Chen, H. Tian, Social learning differential evolution, Inform. Sci. 433 (2018) 464–509.

[31] J. Liu, J. Lampinen, A fuzzy adaptive differential evolution algorithm, Soft Comput. 9 (6) (2005) 448–462.

[32] J. Brest, S. Greiner, B. Boskovic, M. Mernik, V. Zumer, Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems, IEEE Trans. Evol. Comput. 10 (6) (2006) 646–657.

[33] J. Zhang, A.C. Sanderson, JADE: adaptive differential evolution with optional external archive, IEEE Trans. Evol. Comput. 13 (5) (2009) 945–958.

[34] L. Tang, Y. Dong, J. Liu, Differential evolution with an individual-dependent mechanism, IEEE Trans. Evol. Comput. 19 (4) (2014) 560–574.

[35] Y.-Z. Zhou, W.-C. Yi, L. Gao, X.-Y. Li, Adaptive differential evolution with sorting crossover rate for continuous optimization problems, IEEE Trans. Cybern. 47 (9) (2017) 2742–2753.

[36] W. He, W. Gong, L. Wang, X. Yan, C. Hu, Fuzzy neighborhood-based differential evolution with orientation for nonlinear equation systems, Knowl.-Based Syst. 182 (2019) 104796.

[37] W. Li, X. Meng, Y. Huang, Fitness distance correlation and mixed search strategy for differential evolution, Neurocomputing 458 (2021) 514–525.

[38] M. Yang, C. Li, Z. Cai, J. Guan, Differential evolution with auto-enhanced population diversity, IEEE Trans. Cybern. 45 (2) (2014) 302–315.

[39] J. Brest, M.S. Maučec, B. Bošković, Single objective real-parameter optimization: Algorithm jSO, in: 2017 IEEE Congress on Evolutionary Computation, CEC, IEEE, 2017, pp. 1311–1318.

[40] Z. Meng, C. Yang, Hip-DE: Historical population based mutation strategy in differential evolution with parameter adaptive mechanism, Inform. Sci. 562 (2021) 44–77.

[41] Z. Meng, C. Yang, Two-stage differential evolution with novel parameter control, Inform. Sci. 596 (2022) 321–342.

[42] W. Deng, J. Xu, Y. Song, H. Zhao, Differential evolution algorithm with wavelet basis function and optimal mutation strategy for complex optimization problem, Appl. Soft Comput. 100 (2021) 106724.

[43] V. Osuna-Enciso, E. Cuevas, B.M. Castañeda, A diversity metric for population-based metaheuristic algorithms, Inform. Sci. 586 (2022) 192–208.

[44] Z. Meng, Y. Zhong, G. Mao, Y. Liang, PSO-sono: A novel PSO variant for single-objective numerical optimization, Inform. Sci. 586 (2022) 176–191.

[45] Z. Meng, Y. Chen, X. Li, C. Yang, Y. Zhong, Enhancing QUasi-Affine TRansformation Evolution (QUATRE) with adaptation scheme on numerical optimization, Knowl.-Based Syst. 197 (2020) 105908.

[46] A. Kumar, R.K. Misra, D. Singh, Improving the local search capability of effective butterfly optimizer using covariance matrix adapted retreat phase, in: 2017 IEEE Congress on Evolutionary Computation, CEC, IEEE, 2017, pp. 1835–1842.

[47] G. Zhang, Y. Shi, Hybrid sampling evolution strategy for solving single objective bound constrained problems, in: 2018 IEEE Congress on Evolutionary Computation, CEC, IEEE, 2018, pp. 1–7.

[48] P. Bujok, P. Kolenovsky, Eigen crossover in cooperative model of evolutionary algorithms applied to cec 2022 single objective numerical optimisation, in: 2022 IEEE Congress on Evolutionary Computation, CEC, IEEE, 2022, pp. 1–8.