Information Sciences 596 (2022) 321–342

Contents lists available at ScienceDirect

Information Sciences

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

Two-stage differential evolution with novel parameter control

Zhenyu Meng [a,b,*], Cheng Yang [a]

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

ARTICLE INFO

Article history:
Received 8 May 2021
Received in revised form 23 January 2022
Accepted 6 March 2022
Available online 10 March 2022

Keywords:
Differential evolution
Novel parameter control
Population enhancement technique
Single-objective optimization
Stagnation detection

ABSTRACT

In this paper, we propose a **T**wo-stage **D**ifferential **E**volution (TDE) with novel parameter control for real parameter single objective global optimization. In the TDE algorithm, the whole evolution is divided into two stages and each stage employs a unique mutation strategy. The mutation strategy in the first stage is a novel historical-solution based mutation strategy, which can get better perception of the landscape of the objective; the mutation strategy in the second stage is an inferior-solution based mutation strategy, which can maintain better diversity of trial vector candidates while keeping better convergence speed. Furthermore, the parameter control of our TDE is novel, which means that these adaptations of control parameters are different from those in the literature: First, the adaptation schemes both for scale factor *F* and crossover rate *CR* are fitness-independent. Second, different from the fixed population size and the gradually reduced population size, the population adaptation in TDE has two different stages. Third, a stagnation indicator is proposed and a population enhancement technique can be launched if necessary when a certain individual is in the stagnation status. We examine the TDE algorithm under a relative large number of benchmarks from CEC2013, CEC2014 and CEC2017 test suites for real-parameter single objective global optimization, and the experiment results show the competitiveness of our TDE algorithm with several recently proposed state-of-the-art DE variants, e.g. it obtained 20 similar or better performance improvements out of the total 30 benchmarks in comparison with the winner algorithm, the LSHADE algorithm, of the CEC2014 competition and it also obtained 19 similar or better performance improvements out of the total 30 benchmarks in comparison with the winner DE variant, the jSO algorithm, of the CEC2017 competition.

1. Introduction

With the development of science and technology, more and more optimization problems arise in different areas, and developing an efficient algorithm to tackle these optimization problems becomes increasingly more popular with scientific researchers and engineers [6,16,24–26]. Generally, these optimization algorithms can be classified into two different categories: deterministic optimization algorithms and stochastic optimization algorithms [23]. For the deterministic algorithms, the model of the problem is fully specified and the relation between the characteristics of the possible solutions and their utilities for the given problem is explicit. However, there are many cases in the real-world optimization that the system

* 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.ins.2022.03.043
0020-0255/

is too complex or the detailed equations of the objectives modeling the system are unavailable, then algorithms in the second category are of the better choice in tackling these optimization problems.

In the past few decades, many stochastic optimization algorithms have been developed, such as Genetic Algorithm (GA) [8,?], Simulated Annealing (SA) [9], Particle Swarm Optimization (PSO) [5], Covariance Matrix Adaptation based Evolution Strategy (CMA-ES) [7], Differential Evolution (DE) [31,29,35], Monkey King Evolution (MKE) [27] and QUATRE algorithm [19,14,12] etc. Among these algorithms, the DE variants drew increasing attention of the researchers because of their easy-implementation and excellent performance. The first version of DE algorithm was proposed by Storn and Price in 1995 [30], which aimed at tackling Chebyshev polynomial fitting problem [29]. By using floating-point instead of bit-string encoding, the differential mutation operator showed great success in numerical optimization. After decades of development, the DE variants became general tools for different kinds of optimization applications, moreover, some of them secured front ranks in real-parameter single-objective optimization competitions [32,2] as well. However, these winner DE variants were also criticized for the following aspects: 1) bad perception of the landscape of the objective, which usually leads to a wrong search direction; 2) the dependence of fitness-difference in the adaptation of control parameters, which restricts applicable scope of these algorithms only to fitness available optimization; 3) a lack of population diversity when the algorithm enters the later part of the evolution, which usually results in a premature convergence. Here in this paper a novel TDE algorithm is proposed aiming at improving the performance of DE on real-parameter single-objective optimization, and the main highlights of our TDE algorithm can be summarized as follows:

- A two-stage differential evolution is proposed to tackle single-objective real-parameter optimization, and the mutation strategy in the first stage is a novel historical-solution based mutation strategy while the mutation strategy in the second stage is an inferior-solution based mutation strategy. This composition of mutation strategies can get better perception of the landscape of the objective in the earlier stage of the evolution while maintaining a better diversity in the later part of the evolution.
- Better parameter control schemes for both F and CR are proposed in our TDE algorithm, which means that the adaptations of control parameters F and CR are different from those in the literature.
- Some seeds are maintained during the evolution far from entering the stagnation status of the population, and an indicator is proposed for the definition of the stagnation status. A population enhancement technique can be launched if necessary by employing the seeds maintained during the evolution when a certain individual enters the stagnation status.
- A larger test suite containing benchmarks from CEC2013, CEC2014, and CEC2017 test suites for single-objective real-parameter optimization is used for the validation of our TDE algorithms, which can avoid over-fitting problems in comparison with using a single test suite containing a smaller number of benchmarks.

The remainder of the paper is arranged as follows: Section 2 presents the related work of DE. Section 3 introduces the evolution structure of DE. Section 4 introduces the details of our TDE algorithm. Section 5 shows the experiment analysis and algorithm validation of our TDE algorithm in comparison with several winner DE variants in recent competitions. Finally, Section 6 gives the conclusion.

2. Related work

Generally, there are two components influencing the overall performance of a certain mutation strategy: the base vector and the number of difference-vector pairs. Consequently, the earlier investigated mutation strategies of DE can be written in "DE/x/y" form, where x denotes the base vector and y denotes the number of difference-vector pairs, e.g. DE/rand/1, DE/best/1, DE/target-to-best/1, and DE/rand/1/either-or [29]. As we know, the mutation strategy DE/rand/1 has better exploration capacity while the mutation strategy DE/best/1 has better exploitation capacity. DE/target-to-best/1 obtains a balance between the exploration of DE/rand/1 and the exploitation of DE/best/1 [3]. Obviously, both the base vector and the difference pairs in these mutation strategies are only focusing on the employment of the population in the current generation, and the knowledge obtained during the evolution is missing [21,20]. Zhang and Sanderson made an attempt to employ some of the inferior solutions of the past population and proposed a novel mutation strategy, DE/target-to-$p$best/1 with external archive, in which these inferior solutions were stored [38]. Because of its excellent performance, the recent winner DE variants in the competitions of the Congress on Evolution Computation (CEC) for single-objective real-parameter optimization employed the same or a similar mutation strategy [32,2,18,15].

Besides the mutation strategy, the control parameters are also important factors influencing the overall optimization performance. There are three control parameters, population size PS, scale factor F and crossover rate CR, in the canonical DE algorithm, and they are all set constant values during the evolution [29]. Besides the canonical DE algorithm, there are several DE variants employing the fixed constant values of control parameters in the earlier development of DE. Generally, the fix values of control parameters can neither fit different objectives nor fit different stages of the evolution under the same objective. Furthermore, there are also claims and counter-claims mentioning the recommended fixed values of these parameters in the literature, which often results in confusions to both DE researchers and users [4]. Regarding these issues, parameter adaptive DE algorithms become increasingly popular with both scientific researchers and optimization engineers. Brest et al. [1] proposed the jDE algorithm with self-adapting control parameters. The thought "Better control parameters lead to better individuals which are more likely to survive, hence, propagate these better control parameters." was proposed and employed in this algorithm, and it also had a profound influence on the development of DE algorithm proposed thereafter. The JADE algorithm [38] further developed this thought: all the control parameters in the JADE algorithm obeyed semi-fixed distributions and the good control parameters were used for updating the distributions of themselves. Peng et al. [28] further improved JADE algorithm by incorporating fitness-difference which was used as weight in the adaptation of the corresponding control parameters F and CR, and this fitness-difference based adaptation scheme was also incorporated into LSHADE which obtained the first rank in CEC 2014 competition on single-objective real-parameter optimization [32]. Later winner DE variants in CEC competitions on real-parameter single-objective optimization both employed a similar mutation strategy to the JADE algorithm and a similar fitness-difference based parameter control to the LSHADE algorithm [2,18,15].

3. The evolution structure of DE

In this part, we present a brief review of the evolution structure of DE algorithms, and all the variants can be separated into two parts, the initialization and the iteration of mutation, crossover and selection until termination.

(A) Initialization:

There are PS individuals of the population initialized in this stage, and uniform distribution of the D-dimensional solution space Ω is employed in scattering these individuals, where the lower border $X^D_{\min}, X^D_{\min} = (x_{\min,1}, x_{\min,2}, \ldots, x_{\min,D})$ and the upper border $X^D_{\max}, X^D_{\max} = (x_{\max,1}, x_{\max,2}, \ldots, x_{\max,D})$ are used for delimiting the solution space. Eq. 1 presents the calculation of the j^{th} dimension of the i^{th} individual in the initialization (the 0^{th} generation):

$$x_{ij,0} = x_{\min,j} + rand_{ij}(0,1) \cdot (x_{\max,j} - x_{\min,j}) \tag{1}$$

where $rand_{i,j}(0,1)$ denotes a random value generated according to uniform distribution of the range $(0,1)$ during the calculation of $x_{ij,0}$. After all the individuals in the population are initialized, the fitness values of these individuals can be calculated by function calls and then the global best of the current population is labeled, then, the algorithm enters the evolution stage which contains operations of mutation, crossover and selection.

(B) Evolution:

The evolution stage contains many iterations of three basic operations including mutation, crossover and selection before algorithm termination. For the mutation operation, a certain mutation strategy should be specified before conducting this operation, and the mutation strategy of the canonical DE algorithm [30] is given in Eq. 2:

$$V_{i,G} = X_{r_0,G} + F \cdot (X_{r_1,G} - X_{r_2,G}) \tag{2}$$

where $X_{r_0,G}, X_{r_1,G}$ and $X_{r_2,G}$ denote randomly selected individuals from the population with the indices r_0, r_1 and r_2 obeying random selection with restriction, $r_0 \neq r_1 \neq r_2$ [29]. Then the mutant vector $V_{i,G}$ is taken as an input together with the target vector $X_{i,G}$ into the crossover operation.

For the crossover operation, there are many different kinds of crossover schemes mentioned in the literature, such as 1-point, 2-point, exponential crossover, binomial crossover [29], recombination [4] and evolution matrix [13] etc. Generally, many DE researchers believe that DE variants employing binomial crossover usually perform well on real-parameter single-objective optimization [3], but, we find that a DE variant with exponential crossover can also beat those state-of-the-art DE variants with binomial crossover on real-parameter single objective optimization under the CEC test suites by employing proper initial values and proper adaptation schemes of the control parameters. We'll discuss it in our next paper. Here in this paper, we mainly focus on binomial crossover because these state-of-the-art DE variants taken into comparison all employ the binomial crossover. Fig. 1 presents an example of calculating an 8-dimensional trial vector $U_{i,G}$ with the two inputs $X_{i,G}$ and $V_{i,G}$ under a certain CR by employing binomial crossover.

The last operation at the end of the iteration is selection which is actually to make a choice between the target vector $X_{i,G}$ and the trial vector $U_{i,G}$ regarding their performance under a certain objective, and this operation is presented in Eq. 3 [29]:

$$X_{i,G+1} = \begin{cases} U_{i,G}, & if\ U_{i,G} \text{isbetterthan} X_{i,G}. \\ X_{i,G}, & \text{otherwise} \end{cases} \tag{3}$$

4. The novel TDE algorithm

In this section, we mainly describe the details of the TDE algorithm, and the whole algorithm can be further divided into three subsections: the first subsection presents the trial vector generation strategy; the second subsection gives the novel parameter control during the whole evolution; and the third subsection describes a diversity enhancement technique of the population.

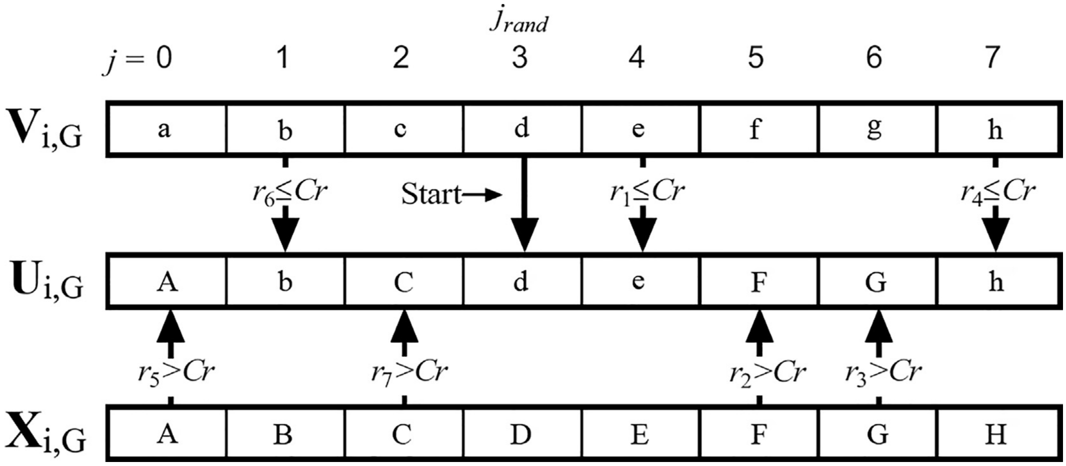


Fig. 1. An example of calculating an 8-dimensional trial vector $U_{i,G}$ with the two inputs $X_{i,G}$ and $V_{i,G}$ under a certain *CR* by employing binomial crossover.

4.1. Trial vector generation strategy

As it is known to all that the choice of the trial vector generation strategy has a significant impact on the overall performance of a DE variant. In the literature, DE researchers have already proposed different kinds of trial vector generation strategies. Among them, the strategy DE/target-to-*p*best/bin/1 with external archive proposed in the JADE algorithm showed excellent performance in recent CEC competitions [38,32]. The detailed equation of this strategy [38] is given below:

$$V_{i,G} = X_{i,G} + F \cdot \left(X^{p}_{best,G} - X_{i,G}\right) + F \cdot \left(X_{r_1,G} - \widetilde{X}_{r_2,G}\right) \tag{4}$$

Here $X^{p}_{best,G}$ denotes a randomly selected elite from the top $p \cdot 100\%$ individuals where p is the ratio of elites to the whole population. $\widetilde{X}_{r_2,G}$ denotes a random individual from the union $\boldsymbol{P} \cup \boldsymbol{A}$ where $\boldsymbol{P}$ denotes the set of current population and $\boldsymbol{A}$ denotes the set of one external archive that records inferior solutions. Furthermore, i, r_1 and r_2 obey random selection with restriction, $i \neq r_1 \neq r_2$. Generally, the latest powerful DE variants on the CEC test suite for real-parameter single-objective optimization choose either the same trial vector generation strategy or some improved versions of it [2,17,18]. However, these strategies also have the same shortcoming that the historical information during the evolution is absent. Generally, the relation of historical individuals in different generations of the evolution can reflect the landscape of the objective. By extracting the depth-information of the historical individuals and incorporating this knowledge into the mutation strategy, the perception of the landscape of the objective can be enhanced and the possibility converging into some local optima can be decreased [15,20]. To be more explicated, the depth-information is the linkage between the historical population and the current population. $X_{r_1,G} - \widehat{X}_{r_2,G}$ in Fig. 2 presents the depth-information-component of the trial vector generation strategy in Eq. 5:

$$V_{i,G} = X_{i,G} + F \cdot \left(X^{p}_{best,G} - X_{i,G}\right) + F_w \cdot \left(X_{r_1,G} - \widehat{X}_{r_2,G}\right) \tag{5}$$

where $\widehat{X}_{r_2,G}$ denotes a randomly selected individual form the union $\boldsymbol{P} \cup \boldsymbol{B}$ where $\boldsymbol{B}$ denotes the set of the other external archive which records historical solutions.

In our TDE algorithm, the trial vector generation strategy in Eq. 5 is employed in the first-stage of the evolution due to its good perception of the landscape of the objective, and the strategy of the JADE algorithm with external archive recording inferior solutions is employed in the second-stage of the evolution. Therefore, the whole trial vector generation strategy of our TDE algorithm can be given below in Eq. 6:

$$\begin{cases} V_{i,G} = X_{i,G} + F \cdot \left(X^{p}_{best,G} - X_{i,G}\right) + F_w \cdot \left(X_{r_1,G} - \widehat{X}_{r_2,G}\right), \textit{if nfe} < \rho \\ V_{i,G} = X_{i,G} + F \cdot \left(X^{p}_{best,G} - X_{i,G}\right) + F \cdot \left(X_{r_1,G} - \widetilde{X}_{r_2,G}\right), \textit{otherwise} \end{cases} \tag{6}$$

Here F_w is a constricted scale factor, and it equals to $0.9 \cdot F$. The new parameter ρ defines the ratio of the earlier stage of the evolution to the whole evolution, and it is recommended to be set $2/3 \cdot nfe_{\max}$ in our TDE algorithm. As to the external archives $\boldsymbol{A}$ and $\boldsymbol{B}$, the maximum size of $\boldsymbol{A}$ equals to the current population size *PS*, and the maximum size of $\boldsymbol{B}$ equals to $3 \cdot PS$. By the way, $\boldsymbol{A}$ and $\boldsymbol{B}$ are initialized empty at the beginning of the evolution, then the inferior solutions of the current population are gradually recorded in archive $\boldsymbol{A}$ while the entire population of individuals are recorded in archive $\boldsymbol{B}$ during the

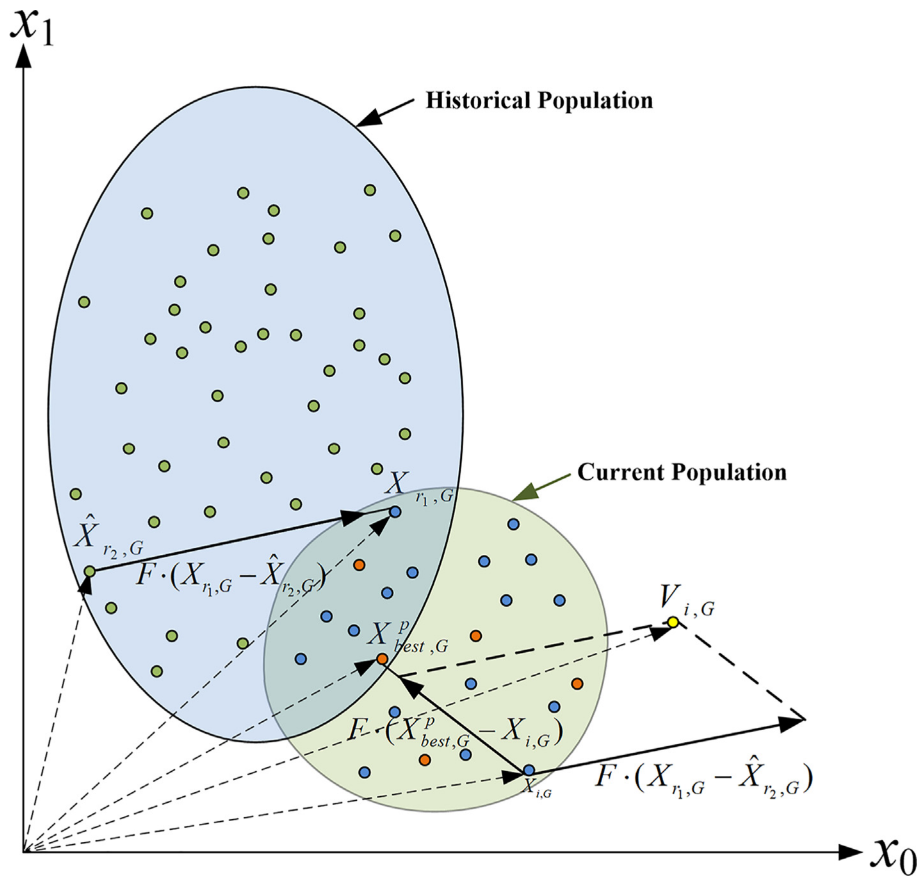


Fig. 2. Illustration of the relationships between the historical population and the current population. The depth-information-component $X_{r_1,G} - \widehat{X}_{r_2,G}$ is also presented in this figure.

evolution. When the number of records exceeds the maximum size of any external archive, some of the records will be randomly removed from the archive to keep the size equalling to the fixed maximum.

4.2. The parameter control

The winner DE variants in recent CEC competitions for real-parameter single-objective optimization mainly employed the fitness-difference based adaptation schemes for the control parameters F and CR. The dependence of fitness value of the objective hampers the effectiveness of these algorithms in tackling optimization problems especially for those the fitness values are unavailable. Here in this part, fitness-independent parameter control is presented for the adaptations of these control parameters.

In our TDE algorithm, all the individuals are categorized into K groups, and the scale factor of each individual obeys semi-fixed Cauchy distribution, $F \sim C(\mu_F, 0.1)$. In each group, there are two parameters associating with each individual: the Gaussian distribution mean value μ_{CR} and the selection probability $P(\cdot)$ of a certain group. The initial values of μ_{CR} for all groups are the same, $\mu_{CR_1} = \mu_{CR_2} = \ldots = \mu_{CR_k} = \ldots = \mu_{CR_K} = 0.8$, and the initial probabilities of $P(\cdot)$ equal to $P(k) = \frac{1}{K}, j = 1, 2, \ldots, K$. Adjustments of these parameters are conducted at the end of each generation: the selection probability of the k^{th} group is updated according to Eq. 7:

$$\begin{cases} r_k = \begin{cases} \frac{ns_k^2}{ns\cdot(ns_k+nf_k)}, & \textit{if } ns_k > 0 \\ \epsilon, & \textit{otherwise} \end{cases} \\ P(k) = \frac{r_k}{\sum_{k=1}^{K}(r_k)} \end{cases} \tag{7}$$

where nf_k and ns_k denote the number of failure and success trial vectors in the k^{th} group respectively, and ns denotes the number of success trial vectors in the current generation. The adaptations of μ_F and μ_{CR} are given below in Eq. 8 and Eq. 9:

$$\begin{cases} w_s = \frac{std(\Delta loc_i)}{\sum_{s=1}^{|S_F|} std(\Delta loc_i)} \\ \Delta loc_i = loc(U_{i,G} - X_{i,G}) \\ \text{mean}_{WL}(S_F) = \frac{\sum_{s=1}^{|S_F|} w_s \cdot S_F^2(s)}{\sum_{s=1}^{|S_F|} w_s \cdot S_F(s)} \\ \mu_F = \begin{cases} \text{mean}_{WL}(S_F), & if\ S_F \neq \emptyset \\ \mu_F, & otherwise \end{cases} \end{cases} \tag{8}$$

$$\begin{cases} w_s = \frac{std(\Delta loc_i)}{\sum_{s=1}^{|S_F|} std(\Delta loc_i)} \\ \Delta loc_i = loc(U_{i,G} - X_{i,G}) \\ mean_{WL}(S_{CR}) = \frac{\sum_{s=1}^{|S_{CR}|} w_s \cdot S_{CR}^2(s)}{\sum_{s=1}^{|S_{CR}|} w_s \cdot S_{CR}(s)} \\ \mu_{CR,k} = \begin{cases} mean_{WL}(S_{CR}),\ if\ S_{CR} \neq \emptyset \& \max\{CR\} > 0 \\ 0,\ if\ S_{CR} \neq \emptyset \max\{CR\} = 0 \\ \mu_{CR,k},\ otherwise \end{cases} \end{cases} \tag{9}$$

where the new operation $loc(X_{i,G} - U_{i,G})$ denotes locating the effective dimensional changes of the displacement, $U_{i,G} - X_{i,G}$, of the i^{th} individual and $std(\cdot)$ denotes the operation of calculating the standard deviation of these D-dimensional effective changes or parameters. The μ_{CR} of each group is renewed in a loop: from the first group to the last and then back to the first, and only one group is renewed in each generation.

For the population size PS, a two-stage based reduction scheme is proposed in our TDE algorithm. It is based on the observation that a large population size at the earlier stage of the evolution helps the algorithm to get a better perception of the objective while a linear reduction of population size in the later part of the evolution helps the algorithm to make a better exploitation. Therefore, a fixed larger population size is employed in the first stage and a linear-reducing population size is employed in the later part of the evolution. The detailed scheme of the population size during the evolution is given in Eq. 10:

$$PS = \begin{cases} PS,\ if\ nfe < p_s \cdot nfe_{\max} \\ \lceil \frac{PS_{\min} - PS_{\text{ini}}}{(1-p_s) \cdot nfe_{\max}} + PS_{\text{ini}},\ otherwise \end{cases} \tag{10}$$

where PS_{ini} denotes the initial population size, $PS_{\min}$ denotes the terminal (also minimum) population size, p_s defines the ratio of the fixed population evolution to the whole population, and the default value of p_s is 0.05 in our TDE algorithm.

4.3. Diversity enhancement of the population

A diversity enhancement technique of the population is employed in our TDE algorithm, which aims at tackling the premature convergence during the evolution. Generally, an efficient way to increase the probability of converging into the global optimum mentioned in the literature [37] is to draw the population out of the stagnation status before termination of the evolution. Here we give a criterion which defines the stagnation status, and there are two aspects for the definition: the first is about the diversity of the population and the second is about the number of generations without performance improvement of a certain individual.

Table 1
Recommended parameter settings of all these contrasted algorithms.

Algorithm	Parameters initial settings
SHADE	$NP = 100, F \sim C(\mu_F, 0.1), \mu_F = 0.5, CR \sim N(\mu_{CR}, 0.1), \mu_{CR} = 0.5, p = 0.2, H = 100$
LSHADE	$\mu_F = 0.5, F \sim C(\mu_F, 0.1), \mu_{CR} = 0.5, CR \sim C(\mu_C R, 0.1), NP = 18 \cdot D \sim 4, r^{rac} = 2.6, p = 0.11, H = 6$
iLSHADE	$H, F, CR\&r^{rac}$ same as LSHADE, $\mu_F = 0.8, \mu_C R = 0.5, \mu_{F_H} = \mu_{CR_H} = 0.9, NP = 12 \cdot D \sim 4, p = 0.2 \sim 0.1$
jSO	$F, CR\&r^{rac}$ same as iLSHADE, $\mu_F = 0.3, \mu_C R = 0.8, NP = 25 \cdot log(D) \cdot \sqrt{D} \sim 4, p = 0.25 \sim 0.125, H = 5$
LPalmDE	$\mu_{F_j} = 0.8, F_{j,i} \sim C\left(\mu_{F_j}, 0.2\right), \mu_{CR} = 0.6, CR \sim N(\mu_{CR}, 0.1), k = 20, NP = 23 \cdot D \sim k, p = 0.11, r^{rac} = 1.6, T_0 = 70$
HARD-DE	$\mu_F = 0.3, \mu_C R = 0.8, F\&CR$ same as LSHADE, $p = 0.11, NP = 25 \cdot log(D) \cdot \sqrt{D} \sim 4, r^{rac,A} = 1.6, r^{rac,B} = 3, k = 4$
TDE	$\mu_F = 0.3, \mu_{CR} = 0.8, F \sim C(\mu_F, 0.1), CR \sim N(\mu_{CR}, 0.1), p = 0.25 \sim 0.11, r^{rac,A} = 1.6, r^{rac,B} = 3, k = 4, NP = 25 \cdot log(D) \cdot \sqrt{D} \sim 4, n = 2 \cdot D, \rho = \frac{2}{3} \cdot nfe_{\max}$

Table 2
Comparison of our TDE algorithm with SHADE, LSHADE, iLSHADE, jSO, LPalmDE and HARD-DE in 10-D optimization.

DE Variants NO.	SHADE Mean/Std	LSHADE Mean/Std	iLSHADE Mean/Std	jSO Mean/Std	LPalmDE Mean/Std	HARD-DE Mean/Std	TDE Mean/Std
f_{a_1}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
f_{a_2}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
f_{a_3}	1.0974E−001/ 7.6340E−001(<)	5.5966E−003/ 1.9375E−002(<)	1.1193E−002/ 2.6209E−002(<)	1.3992E−003/ 9.9919E−003(>)	5.5966E−003/ 1.9375E−002(<)	2.7992E−003/ 1.3989E−002(<)	2.7983E−003/ 1.3989E−002
f_{a_4}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
f_{a_5}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
f_{a_6}	8.6580E+000/ 3.1929E+000(<)	5.0024E+000/ 4.9541E+000(<)	6.1568E+000/ 4.7914E+000(<)	1.3468E+000/ 3.4102E+000(<)	2.8860E+000/ 4.5155E+000(<)	9.6200E−001/ 2.9469E+000(<)	0/0
f_{a_7}	4.6560E−003/ 4.8715E−003(<)	8.3113E−006/ 2.5344E−005(>)	1.4712E−005/ 4.1995E−005(>)	3.4153E−005/ 1.0479E−004(<)	4.4716E−006/ 8.4668E−006(>)	3.0803E−003/ 4.5803E−003(<)	2.0613E−005/ 6.7555E−005
f_{a_8}	2.0367E+001/ 7.1822E−002(<)	2.0221E+001/ 1.6198E−001(<)	2.0338E+001/ 9.3426E−002(<)	2.0358E+001/ 8.3752E−002(<)	2.0198E+001/ 1.5173E−001(<)	2.0203E+001/ 1.1728E−001(<)	2.0051E+001/ 1.0038E−001
f_{a_9}	3.2557E+000/ 9.9109E−001(<)	2.6792E+000/ 1.3036E+000(<)	5.8850E−001/ 8.7988E−001(<)	7.0117E−001/ 8.6243E−001(<)	5.6465E−001/ 8.1119E−001(<)	2.3361E+000/ 1.3518E+000(<)	3.5594E−001/ 7.8729E−001
$f_{a_{10}}$	1.1660E−002/ 7.5163E−003(<)	1.4985E−003/ 3.3466E−003(>)	6.0844E−003/ 8.2879E−003(>)	1.7401E−003/ 3.8597E−003(>)	5.7999E−004/ 2.3430E−003(>)	9.6277E−007/ 6.8547E−006(>)	1.0953E−002/ 1.6421E−002
$f_{a_{11}}$	0/0(≈)	0/0(≈)	1.9509E−002/ 1.3932E−001(<)	0/O(≈)	0/0(≈)	1.1146E−014/ 2.2793E−014(<)	0/0
$f_{a_{12}}$	3.6701E+000/ 9.9258E−001(<)	1.8638E+000/ 9.7511E−001(>)	2.0906E+000/ 7.7796E−001(>)	2.3801E+000/ 8.2236E−001(>)	2.7703E+000/ 1.1991E+000(>)	2.3826E+000/ 8.8148E−001(>)	2.8385E+000/ 1.8435E+000
$f_{a_{13}}$	3.5892E+000/ 1.4849E+000(<)	1.7952E+000/ 8.2689E−001(>)	1.7814E+000/ 8.5561E−001(>)	2.1201E+000/ 1.0770E+000(>)	2.0016E+000/ 1.0231E+000(>)	2.8389E+000/ 1.5088E+000(<)	2.5370E+000/ 2.1667E+000
$f_{a_{14}}$	2.4492E−003/ 1.2244E−002(>)	1.9594E−002/ 3.4185E−002(≈)	2.8030E−001/ 6.4621E−001(<)	3.6075E−002/ 4.3136E−002(<)	0/0(>)	7.3116E−013/ 3.6469E−013(>)	9.7968E−003/ 2.2939E−002
$f_{a_{15}}$	4.3274E+002/ 1.2858E+002(<)	3.0600E+002/ 1.2030E+000(>)	2.5327E+002/ 1.1714E+002(>)	2.8348E+002/ 1.1125E+002(>)	3.5320E+002/ 1.5017E+002(>)	3.3661E+002/ 1.2678E+002(>)	4.0786E+002/ 1.7934E+002
$f_{a_{16}}$	6.7928E−001/ 1.9768E−001(<)	2.4686E−001/ 1.4875E−001(<)	8.3311E−001/ 3.0579E−001(<)	1.0942E+000/ 2.0347E−001(<)	1.8616E−001/ 1.6592E−001(<)	3.0552E−001/ 2.6017E−001(<)	1.1148E−001/ 1.4387E−001
$f_{a_{17}}$	1.0122E+001/ 4.4466E−014(≈)	1.0122E+001/ 1.7940E−015(≈)	1.0126E+001/ 6.6650E−003(<)	1.0123E+001/ 8.5962E−004(<)	1.0122E+001/ 1.8501E−014(≈)	1.0122E+001/ 1.7940E−015(≈)	1.0122E+001/ 1.7940E−015
$f_{a_{18}}$	1.7192E+001/ 1.5687E+000(<)	1.3558E+001/ 1.1299E+000(>)	1.3379E+001/ 1.2681E+000(>)	1.6434E+001/ 1.9716E+000(<)	1.5126E+001/ 1,7189E+000(<)	1.5371E+001/ 1.4172E+000(<)	1.5104E+001/ 2.4461E+000
$f_{a_{19}}$	3.2097E−001/ 5.2957E−002(<)	2.2934E−001/ 3.9715E−002(>)	3.1047E−001/ 5.9523E−002(<)	2.7388E−001/ 4.9680E−002(<)	2.7145E−001/ 5.9158E−002(<)	2.3745E−001/ 5.7242E−002(<)	2.3079E−001/ 4.3846E−002
$f_{a_{20}}$	2.2493E+000/ 4.1469E−001(<)	2.0782E+000/ 3.6061E−001(<)	1.8281E+000/ 5.2566E−001(<)	1.7248E+000/ 3.1470E−001(<)	1.7162E+000/ 3.9189E−001(>)	1.6965E+000/ 3.5179E−001(>)	1.7203E+000/ 3.2924E−001
$f_{a_{21}}$	3.9627E+002/ 2.8033E+001(>)	3.9627E+002/ 2.8033E+000(>)	4.0019E+002/0 (≈)	3.9627E+002/ 2.8033E+001(>)	4.0019E+002/0 (≈)	4.0019E+002/0 (≈)	4.0019E+002/ 0.0000E+000
$f_{a_{22}}$	4.8362E+000/ 5.3147E+000(<)	9.3321E+000/ 1.8973E+001(<)	2.2267E+001/ 3.2619E+001(>)	6.5426E+000/ 4.8560E+000(<)	3.2731E+000/ 3.6231E+000(<)	2.8905E+000/ 3.7024E+000(<)	2.3381E+000/ 3.2980E+000
$f_{a_{23}}$	4.3764E+002/ 1.6590E+002(<)	3.0463E+002/ 1.2679E+002(>)	2.2308E+002/ 1.1810E+002(>)	2.2153E+002/ 1.1252E+002(>)	3.6548E+002/ 1.7272E+002(<)	3.5105E+002/ 1.3203E+002(<)	3.2664E+002/ 1.8688E+002
$f_{a_{24}}$	2.0002E+002/ 4.7469E+000(<)	2.0157E+002/ 1.4883E+001(<)	2.0109E+002/ 1.1946E+001(<)	1.9992E+002/ 1.1073E+001(>)	2.0000E+002/0 (≈)	1.9915E+002/ 6.0564E+000(>)	2.0000E+002/ 4.3294E−006
$f_{a_{25}}$	2.0038E+002/ 1.8329E+000(<)	2.0053E+002/ 1.4773E+000(<)	2.0054E+002/ 1.7517E+000(<)	2.0009E+002/ 6.3574E−001(<)	2.0000E+002/ 5.6773E−006(<)	2.0000E+002/ 8.5538E−004(<)	1.9481E+002/ 2.1105E+001
$f_{a_{26}}$	1.2359E+002/ 3.8210E+001(<)	1.2821E+002/ 4.2846E+001(<)	1.2384E+002/ 4.0550E+001(<)	1.0258E+002/ 1.8040E+000(>)	1.1001E+002/ 2.6552E+001(<)	1.0675E+002/ 1.9081E+001(<)	1.0550E+002/ 1.3685E+001
$f_{a_{27}}$	3.0000E+002/ 5.3249E−005(≈)	3.0000E+002/0 (≈)	3.1415E+002/ 4.9002E+001(<)	3.0000E+002/0 (≈)	3.0000E+002/0 (≈)	3.0196E+002/ 1.4003E+001(<)	3.0000E+002/ 1.5082E−013
$f_{a_{28}}$	2.9216E+002/ 3.9208E+001(>)	3.0000E+002/0 (≈)	3.0000E+002/0 (≈)	3.0000E+002/0 (≈)	3.0000E+002/0 (≈)	2.9216E+002/ 3.9208E+001(>)	3.0000E+002/ 4.5475E−014
>/≈/<	3/7/18	9/8/11	8/6/14	9/7/12	7/10/11	7/6/15	-/-/-

For the diversity of the population, we follow other research and employ the standard deviation of all the individuals as an indicator of the population diversity [33]. The indictor R_{DP} of population diversity is given below in Eq. 11:

$$R_{DP} = \frac{DP}{DP_{ini}} \tag{11}$$

where DP denotes the current diversity of the population while DP_{ini} denotes the initial diversity of the population. The parameter DP can be calculated according to Eq. 12:

Table 3
Comparison of our TDE algorithm with SHADE, LSHADE, iLSHADE, jSO, LPalmDE and HARD-DE in 30-D optimization.

DE Variants NO.	SHADE Mean/Std	LSHADE Mean/Std	iLSHADE Mean/Std	jSO Mean/Std	LPalmDE Mean/Std	HARD-DE Mean/Std	TDE Mean/Std
f_{a_1}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
f_{a_2}	7.8132E+003/ 5.8848E+003(<)	5.3054E−013/ 1.7230E−012(<)	2.0823E−009/ 1.0525E−008(<)	3.7115E−010/ 7.7174E−010(<)	1.0967E−012/ 3.9397E−012(<)	1.5961E−012/ 1.7958E−012(<)	2.3629E−013/ 1.6991E−013
f_{a_3}	3.7295E+004/ 2.6285E+005(<)	8.1661E−003/ 3.9626E−002(<)	5.3028E−001/ 2.7556E−002(<)	2.3793E−009/ 1.5886E−008(<)	1.5347E+000/ 8.1136E+000(<)	3.4180E−007/ 2.3979E−006(<)	1.5158E−013/ 1.0825E−013
f_{a_4}	5.5458E−006/ 8.0136E−006(<)	9.3624E−014/ 1.1302E−013(<)	3.9679E−013/ 2.5255E−013(<)	2.5457E−012/ 1.7191E−012(<)	1.4267E−013/ 1.1103E−013(<)	3.3437E−013/ 1.5983E−013(<)	5.7958E−014/ 1.0008E−013
f_{a_5}	1.1146E−013/ 1.5919E−014(<)	1.1369E−013/0 (<)	1.1369E−013/0 (<)	1.1369E−013/0 (<)	1.1369E−013/0 (<)	1.1146E−013/ 1.5919E−014(<)	9.1395E−014/ 4.5586E−014
f_{a_6}	1.7097E+000/ 6.2845E+000(<)	5.5484E−008/ 3.6146E−007(<)	1.0356E+000/ 5.1769E+000(<)	1.0356E+000/ 5.1769E+000(<)	3.0762E−013/ 6.2695E−013(<)	2.9882E−010/ 1.3806E−009(<)	1.1146E−013/ 4.8181E−014
f_{a_7}	2.7398E+000/ 2.9392E+000(<)	6.1822E−001/ 4.1038E−001(<)	2.8039E−001/ 2.2748E−001(<)	4.9068E−002/ 1.5007E−001(<)	1.8401E−001/ 1.9798E−001(<)	2.8010E−002/ 4.8142E−002(<)	2.2649E−002/ 3.9786E−002
f_{a_8}	2.0891E+001/ 1.3262E−001(<)	2.0878E+001/ 1.2773E−001(<)	2.0833E+001/ 1.0683E−001(<)	2.0957E+001/ 4.8471E−002(<)	2.0828E+001/ 1.1058E−001(<)	2.0804E+001/ 1.0978E−001(<)	2.0658E+001/ 1.9145E−001
f_{a_9}	2.7683E+001/ 1.7186E+000(<)	2.6559E+001/ 1.4929E+000(<)	2.0158E+001/ 4.9504E+000(>)	2.3704E+001/ 2.6676E+000(<)	2.0070E+001/ 3.4212E+000(>)	2.5651E+001/ 1.1308E+000(<)	2.3565E+001/ 5.0159E+000
$f_{a_{10}}$	6.3239E−002/ 3.9021E−002(<)	8.7012E−004/ 2.4066E−003(<)	0/0(≈)	0/0(≈)	1.4500E−003/ 3.2015E−003(<)	0/0(≈)	0/0
$f_{a_{11}}$	0/0(>)	7.0218E−014/ 3.6993E−014(<)	6.5760E−014/ 2.0878E−014(<)	1.6496E−013/ 7.3021E−014(<)	2.8979E−014/ 2.8699E−014(>)	1.7610E−013/ 3.0621E−014(<)	4.5698E−014/ 5.3348E−014
$f_{a_{12}}$	2.0319E+001/ 4.2916E+000(<)	5.3711E+000/ 1.7052E+000(>)	7.0237E+000/ 2.0623E+000(<)	9.0538E+000/ 2.4992E+000(<)	9.9338E+000/ 2.8556E+000(<)	1.1144E+001/ 2.0277E+000(<)	6.9632E+000/ 1.7706E+000
$f_{a_{13}}$	4.4444E+001/ 1.1650E+001(<)	5.4697E+000/ 2.5325E+000(>)	1.0136E+001/ 5.7137E+000(<)	1.0715E+001/ 5.2616E+000(<)	1.4342E+001/ 7.2444E+000(<)	1.9139E+001/ 6.9990E+000(<)	8.4974E+000/ 3.4473E+000
$f_{a_{14}}$	1.3879E−002/ 1.6999E−002(<)	2.7396E−002/ 2.5786E−002(<)	4.6537E−002/ 2.9907E−002(<)	8.4381E+000/ 4.4500E+000(<)	7.3480E−003/ 1.0877E−002(≈)	1.1430E−002/ 1.4611E−002(<)	7.3480E−003/ 1.2368E−002
$f_{a_{15}}$	3.1191E+003/ 3.7887E+002(<)	2.6234E+003/ 3.2267E+002(>)	2.5137E+003/ 2.7530E+002(>)	2.6816E+003/ 3.3684E+002(>)	2.7368E+003/ 3.7405E+002(>)	2.8013E+003/ 2.6830E+002(<)	2.7972E+003/ 3.6795E+002
$f_{a_{16}}$	8.4733E−001/ 1.3660E−001(<)	7.5695E−001/ 1.8855E−001(<)	9.2240E−001/ 4.3183E−001(<)	2.3393E+000/ 3.0407E−001(<)	6.3052E−001/ 2.7709E−001(<)	6.4975E−001/ 3.4775E−001(<)	3.1683E−001/ 3.5176E−001
$f_{a_{17}}$	3.0434E+001/ 1.1369E−014(≈)	3.0434E+002/ 9.4299E−007(≈)	3.0434E+001/ 2.1913E−006(≈)	3.0669E+001/ 1.0979E−001(<)	3.0434E+001/ 1.3202E−006(≈)	3.0434E+001/ 9.4299E−007(≈)	3.0434E+001/ 4.4970E−014
$f_{a_{18}}$	6.9118E+001/ 5.2733E+000(<)	5.1967E+001/ 3.1366E+000(>)	4.4237E+001/ 4.0750E+000(>)	5.6911E+001/ 6.2991E+000(<)	4.5913E+001/ 4.3222E+000(>)	6.1756E+001/ 5.3215E+000(<)	5.4593E+001/ 5.7393E+000
$f_{a_{19}}$	1.1407E+000/ 1.0270E−001(<)	1.1746E+000/ 9.5320E−002(<)	1.0496E+000/ 1.3757E−001(>)	1.2898E+000/ 1.0311E−001(<)	1.1615E+000/ 1.6683E−001(<)	1.1579E+000/ 9.3429E−002(<)	1.1393E+000/ 8.5026E−002
$f_{a_{20}}$	1.0777E+001/ 5.9591E−001(<)	1.1494E+001/ 2.1569E+000(<)	1.0761E+001/ 1.5341E+000(<)	9.7472E+000/ 3.8117E−001(<)	9.2354E+000/ 5.0056E−001(<)	1.0072E+001/ 1.1175E+000(<)	9.0976E+000/ 4.0442E−001
$f_{a_{21}}$	3.1442E+002/ 7.2862E+001(<)	2.9497E+002/ 3.4443E+001(<)	3.0316E+002/ 6.2148E+001(<)	3.0708E+002/ 5.8493E+001(<)	2.9608E+002/ 1.9604E+001(<)	3.1297E+002/ 5.1972E+001(<)	2.9020E+002/ 3.0033E+001
$f_{a_{22}}$	8.9695E+001/ 3.5731E+001(>)	1.0811E+002/ 2.3996E+000(<)	1.0676E+002/ 1.1981E+000(<)	1.1976E+002/ 3.4337E+000(<)	1.0598E+002/ 3.5441E−001(<)	1.0599E+002/ 1.3311E+000(<)	1.0573E+002/ 4.2686E−001
$f_{a_{23}}$	3.3430E+003/ 3.9093E+002(<)	2.5667E+003/ 3.4307E+002(>)	2.1826E+003/ 4.0411E+002(>)	2.3659E+003/ 3.3190E+002(>)	2.7478E+003/ 4.2784E+002(>)	2.9952E+003/ 3.5601E+002(<)	2.7925E+003/ 3.7871E+002
$f_{a_{24}}$	2.0485E+002/ 5.4111E+000(<)	2.0038E+002/ 7.7501E−001(<)	2.0069E+002/ 4.1575E+000(<)	2.0003E+002/ 3.2709E−002(<)	2.0001E+002/ 2.7675E−002(<)	2.0001E+002/ 5.8235E−003(<)	2.0000E+002/ 3.7890E−003
$f_{a_{25}}$	2.6780E+002/ 1.6816E+001(<)	2.4102E+002/ 9.1896E+000(<)	2.4000E+002/ 4.6140E+000(<)	2.3924E+002/ 6.8915E+000(<)	2.2090E+002/ 2.2685E+001(<)	2.0703E+002/ 1.5367E+001(<)	2.0644E+002/ 1.5094E+001
$f_{a_{26}}$	2.1540E+002/ 3.9407E+001(<)	2.0000E+002/ 1.4171E−013(≈)	2.0784E+002/ 2.7153E+001(<)	2.0000E+002/0 (≈)	2.0000E+002/ 1.4352E−013(≈)	2.0000E+002/ 1.4262E−013(≈)	2.0000E+002/ 1.7845E−013
$f_{a_{27}}$	4.3086E+002/ 1.7338E+002(<)	3.0272E+002/ 9.1927E+000(<)	3.0104E+002/ 2.6188E+000(<)	3.0083E+002/ 9.8159E−001(<)	3.0113E+002/ 3.0229E+000(<)	3.0015E+002/ 1.0279E−001(<)	3.0005E+002/ 1.0796E−001
$f_{a_{28}}$	3.0000E+002/ 2.0385E−013(≈)	3.0000E+002/0 (≈)	3.0000E+002/0 (≈)	3.0000E+002/0 (≈)	3.0000E+002/ 3.2155E−014(≈)	3.0000E+002/ 2.5472E−013(≈)	3.0000E+002/ 1.3258E−013
>/≈/<	2/3/23	5/4/19	5/4/19	2/4/22	5/5/18	0/5/23	-/-/-

Table 4
Comparison of our TDE algorithm with SHADE, LSHADE, iLSHADE, jSO, LPalmDE and HARD-DE in 50-D optimization.

DE Variants NO.	SHADE Mean/Std	LSHADE Mean/Std	iLSHADE Mean/Std	jSO Mean/Std	LPalmDE Mean/Std	HARD-DE Mean/Std	TDE Mean/Std
f_{a_1}	1.9171E−013/ 8.3512E−014(<)	4.9041E−014/ 9.4449E−014(<)	4.4583E−014/ 9.1172E−014(<)	2.6750E−014/ 7.3986E−014(<)	4.4583E−014/ 9.1172E−014(<)	1.3375E−013/ 1.1302E−013(<)	0/0
f_{a_2}	2.4548E+004/ 1.1961E+004(<)	7.8477E+002/ 1.0560E+003(<)	3.4366E+003/ 3.9672E+003(<)	2.2197E+002/ 1.6572E+002(<)	2.4642E+003/ 2.5369E+003(<)	3.4419E+001/ 7.7680E+001(>)	1.8363E+002/ 2.7837E+002
f_{a_3}	2.8709E+005/ 5.8709E+005(<)	1.6454E+004/ 1.0090E+005(<)	4.9184E+003/ 1.7221E+004(<)	1.3171E+002/ 8.0003E+002(<)	5.0226E+003/ 2.4174E+004(<)	3.7217E+001/ 1.7895E+002(>)	5.6457E+000/ 2.7211E+001
f_{a_4}	5.5955E−004/ 1.7401E−003(<)	5.8636E−011/ 6.7342E−011(>)	1.6106E−009/ 3.6941E−009(<)	4.7411E−009/ 7.7628E−009(<)	1.9880E−010/ 2.9676E−010(>)	1.8402E−010/ 3.7860E−010(>)	2.4730E−010/ 5.3184E−010
f_{a_5}	1.2706E−013/ 3.6993E−014(<)	1.5381E−013/ 5.4870E−014(<)	1.7164E−013/ 6.1737E−014(<)	2.0954E−013/ 4.7545E−014(<)	1.6719E−013/ 5.7310E−014(<)	1.5827E−013/ 5.6058E−014(<)	1.1369E−013/ 1.1369E−013
f_{a_6}	4.3447E+001/ 8.5525E−014(≈)	4.3447E+001/ 1.6078E−014(≈)	4.3447E+001/0 (≈)	4.3447E+001/0 (≈)	4.3447E+001/ 2.2737E−014(≈)	4.3447E+001/ 1.6078E−014(≈)	4.3447E+001/ 2.8603E−003
f_{a_7}	1.7772E+001/ 7.4288E+000(<)	2.1563E+000/ 1.1996E+000(<)	4.5763E−001/ 4.7477E−001(<)	1.4960E−001/ 1.2100E−001(>)	1.1921E+000/ 1.0295E+000(<)	2.0747E−001/ 2.8108E−001(>)	2.6887E−001/ 3.8896E−001
f_{a_8}	2.0931E+001/ 2.0998E−001(<)	2.1044E+001/ 1.2381E−001(<)	2.1062E+001/ 7.5472E−002(<)	2.1123E+001/ 4.4397E−002(<)	2.1036E+001/ 1.1030E−001(<)	2.1035E+001/ 1.1086E−001(<)	2.0879E+001/ 1.7878E−001
f_{a_9}	5.5776E+001/ 2.2378E+000(<)	5.3274E+001/ 1.9360E+000(<)	4.0393E+001/ 8.6307E+000(<)	4.7942E+001/ 5.2101E+000(<)	3.8640E+001/ 7.1576E+000(<)	5.1175E+001/ 2.0283E+000(<)	3.7001E+001/ 1.5526E+001
$f_{a_{10}}$	5.1684E−002/ 3.6852E−002(<)	1.0289E−002/ 7.8898E−003(<)	4.1078E−003/ 4.8959E−003(>)	4.5698E−014/ 2.2793E−014(>)	1.3860E−002/ 1.3468E−002(<)	2.8026E−003/ 4.8289E−003(>)	5.5559E−003/ 6.3440E−003
$f_{a_{11}}$	5.5729E−015/ 1.7072E−014(>)	7.6006E−011/ 1.0166E−010(<)	8.4373E−013/ 6.2676E−013(<)	5.9902E−009/ 1.9493E−008(<)	7.4677E−014/ 2.8963E−014(>)	3.6224E−013/ 5.9030E−014(<)	2.1400E−013/ 1.5404E−013
$f_{a_{12}}$	5.1102E+001/ 8.6713E+000(<)	1.4573E+001/ 2.3375E+000(>)	1.2747E+001/ 2.9177E+000(>)	1.5065E+001/ 3.4463E+000(>)	2.2185E+001/ 4.8594E+000(<)	2.4577E+001/ 3.3414E+000(<)	1.7913E+001/ 2.2176E+000
$f_{a_{13}}$	1.2695E+002/ 2.1007E+001(<)	2.3060E+001/ 6.5162E+000(>)	1.8351E+001/ 8.3932E+000(>)	1.8860E+001/ 1.0259E+001(>)	4.3049E+001/ 1.4658E+001(<)	5.7968E+001/ 1.0094E+001(<)	3.3666E+001/ 9.2724E+000
$f_{a_{14}}$	2.5963E−002/ 1.6161E−002(>)	2.1974E−001/ 5.3923E−002(<)	2.5666E−001/ 6.8396E−002(<)	5.9348E+001/ 1.4388E+001(<)	1.4468E−001/ 7.0954E−002(<)	5.6629E−002/ 2.7150E−002(<)	3.5059E−002/ 1.8820E−002
$f_{a_{15}}$	6.7829E+003/ 5.5970E+002(<)	6.4212E+003/ 3.9460E+002(>)	5.3857E+003/ 5.6631E+002(>)	5.9500E+003/ 5.6668E+002(>)	6.0312E+003/ 5.7952E+002(>)	6.5717E+003/ 3.9271E+002(<)	6.5629E+003/ 4.2271E+002
$f_{a_{16}}$	1.3019E+000/ 1.7031E−001(<)	1.2738E+000/ 1.9699E−001(<)	1.3932E+000/ 6.1530E−001(<)	3.1480E+000/ 3.3212E−001(<)	8.6600E−001/ 4.0141E−001(<)	1.2459E+000/ 5.4416E−001(<)	8.3918E−001/ 6.6164E−001
$f_{a_{17}}$	5.0786E+001/ 5.1738E−014(≈)	5.0787E+001/ 2.2381E−003(<)	5.0787E+001/ 2.1396E−003(<)	5.2434E+001/ 4.6550E−001(<)	5.0786E+001/ 1.2575E−009(≈)	5.0786E+001/ 2.0444E−009(≈)	5.0786E+001/ 2.7590E−009
$f_{a_{18}}$	1.3304E+002/ 8.7625E+000(<)	1.0265E+002/ 5.2847E+000(>)	8.4367E+001/ 8.8521E+000(>)	1.1175E+002/ 8.8481E+000(<)	7.8151E+001/ 1.0628E+001(>)	1.1840E+002/ 9.3497E+000(<)	1.0870E+002/ 9.7743E+000
$f_{a_{19}}$	2.5823E+000/ 2.6831E−001(<)	2.4990E+000/ 1.3300E−001(<)	2.3122E+000/ 2.0796E−001(>)	2.6998E+000/ 1.4630E−001(<)	2.3639E+000/ 3.0511E−001(>)	2.4112E+000/ 1.7658E−001(<)	2.3705E+000/ 1.6041E−001
$f_{a_{20}}$	1.9690E+001/ 6.5598E−001(<)	1.8318E+001/ 4.4478E−001(<)	1.8579E+001/ 4.4491E−001(<)	1.9189E+001/ 4.2809E−001(<)	1.7662E+001/ 5.7990E−001(>)	1.8418E+001/ 4.0251E−001(<)	1.7703E+001/ 5.6287E−001
$f_{a_{21}}$	8.7413E+002/ 3.5626E+002(>)	7.7302E+002/ 4.4771E+002(>)	7.9671E+002/ 4.4508E+002(>)	6.8389E+002/ 4.4920E+002(>)	9.1208E+002/ 3.8122E+002(>)	7.6869E+002/ 4.3228E+002(>)	9.3143E+002/ 3.5232E+002
$f_{a_{22}}$	1.2929E+001/ 6.1427E+000(<)	1.3716E+001/ 1.4066E+000(<)	1.3775E+001/ 1.4722E+000(<)	5.9697E+001/ 1.2125E+001(<)	1.1823E+001/ 8.1179E−001(<)	1.1497E+001/ 8.1760E−001(<)	1.1466E+001/ 6.9437E−001
$f_{a_{23}}$	7.5588E+003/ 7.3453E+002(<)	5.6928E+003/ 4.2566E+002(>)	4.7427E+003/ 5.5371E+002(>)	5.0592E+003/ 6.1531E+002(>)	5.8521E+003/ 5.8930E+002(>)	6.5127E+003/ 4.2232E+002(<)	6.3876E+003/ 4.3813E+002
$f_{a_{24}}$	2.3089E+002/ 9.7275E+000(<)	2.0998E+002/ 5.9772E+000(<)	2.0943E+002/ 1.3038E+001(<)	2.0453E+002/ 1.0014E+001(<)	2.0465E+002/ 2.8534E+000(<)	2.0026E+002/ 4.2913E−001(>)	2.0051E+002/ 9.0898E−001
$f_{a_{25}}$	3.5663E+002/ 2.4876E+001(<)	2.7852E+002/ 6.2581E+000(>)	2.7591E+002/ 6.8752E+000(>)	2.7417E+002/ 7.7462E+000(>)	2.8734E+002/ 7.4554E+000(<)	2.8428E+002/ 6.7248E+000(<)	2.8173E+002/ 7.0093E+000
$f_{a_{26}}$	2.4212E+002/ 7.7111E+001(<)	2.4569E+002/ 5.3013E+001(<)	2.9033E+002/ 3.3377E+001(<)	2.3174E+002/ 4.7404E+001(<)	2.6697E+002/ 5.2181E+001(<)	2.3050E+002/ 4.7740E+001(<)	2.2817E+002/ 4.6289E+001
$f_{a_{27}}$	9.5124E+002/ 3.4012E+002(<)	3.9541E+002/ 4.4320E+001(<)	4.9832E+002/ 1.4892E+002(<)	4.4705E+002/ 1.4816E+002(<)	3.6836E+002/ 4.2697E+001(<)	3.1117E+002/ 1.5637E+001(>)	3.1288E+002/ 1.6130E+001
$f_{a_{28}}$	4.5857E+002/ 4.1826E+002(<)	4.0000E+002/ 2.8705E−013(≈)	4.0000E+002/ 2.8705E−013(≈)	4.0000E+002/ 2.8705E−013(≈)	4.0000E+002/ 2.8433E−013(≈)	4.0000E+002/ 2.8705E−013(≈)	4.0000E+002/ 2.7035E−013
>/≈/<	3/2/23	8/2/18	9/2/17	8/2/18	8/3/17	8/3/17	-/-/-

Table 5
Comparison results between the LSHADE algorithm and our TDE algorithm under CEC2014 on 30D optimization.

Algo. NO.	LSHADE Mean/Std	TDE Mean/Std
f_{b_1}	1.1424E−014/6.3677E−015(<)	6.4088E−015/7.1416E−015
f_{b_2}	0/0(≈)	0/0
f_{b_3}	0/0(≈)	0/0
f_{b_4}	5.4612E−014/2.2626E−014(<)	7.8020E−015/1.9755E−014
f_{b_5}	2.0124E+001/2.5747E−002(<)	2.0110E+001/8.8055E−002
f_{b_6}	9.0055E−003/6.4312E−002(<)	0/0
f_{b_7}	0/0(≈)	0/0
f_{b_8}	1.2706E−013/4.9015E−014(<)	6.0187E−014/8.9091E−014
f_{b_9}	6.9189E+000/1.3656E+000(>)	9.1592E+000/2.1064E+000
$f_{b_{10}}$	3.2658E−003/9.6513E−003(<)	8.1644E−004/4.0814E−003
$f_{b_{11}}$	1.1886E+003/1.9425E+002(>)	1.2752E+003/3.1708E+002
$f_{b_{12}}$	1.6158E−001/2.7332E−002(>)	1.7748E−001/5.9978E−002
$f_{b_{13}}$	1.1731E−001/1.4076E−002(>)	1.3159E−001/2.6531E−002
$f_{b_{14}}$	2.3632E−001/2.6563E−002(<)	2.2565E−001/3.0665E−002
$f_{b_{15}}$	2.1445E+000/2.3466E−001(>)	2.3038E+000/2.6549E−001
$f_{b_{16}}$	8.4799E+000/4.8709E−001(>)	8.9292E+000/8.6585E−001
$f_{b_{17}}$	2.0937E+002/9.1752E+001(<)	5.2187E+001/2.8856E+001
$f_{b_{18}}$	5.6560E+000/2.1611E+000(<)	4.1440E+000/1.8060E+000
$f_{b_{19}}$	3.4962E+000/6.9868E−001(<)	2.4583E+000/5.6299E−001
$f_{b_{20}}$	2.7982E+000/1.1935E+000(>)	2.8321E+000/1.3087E+000
$f_{b_{21}}$	7.8709E+001/7.2821E+001(<)	1.4222E+001/2.5864E+001
$f_{b_{22}}$	2.7364E+001/1.7274E+001(>)	7.3762E+001/5.3422E+001
$f_{b_{23}}$	3.1524E+002/4.0186E−013(≈)	3.1524E+002/1.0846E−002
$f_{b_{24}}$	2.2416E+002/1.2545E+000(<)	2.2246E+002/9.2772E−001
$f_{b_{25}}$	2.0261E+001/9.1181E−002(≈)	2.0261E+002/4.5624E−002
$f_{b_{26}}$	1.0011E+002/1.5643E−002(>)	1.0013E+002/2.5665E−002
$f_{b_{27}}$	3.0000E+002/6.4311E−014(≈)	3.0000E+002/0.0000E+000
$f_{b_{28}}$	8.2920E+002/2.1448E+001(>)	8.3429E+002/1.9817E+001
$f_{b_{29}}$	7.1691E+002/3.5624E+000(<)	3.6435E+002/2.8600E+002
$f_{b_{30}}$	1.3008E+003/4.9671E+002(<)	4.3533E+002/5.9699E+001
>/≈/<	10/6/14	-/-/-

$$\begin{cases} \overline{X} = \frac{1}{PS} \cdot \sum_{i=1}^{PS} X_{i,G} \\ DP = \sqrt{\sum_{i=1}^{PS} \left\| X_{i,G} - \overline{X} \right\|^2} \end{cases} \tag{12}$$

For the number of generations without performance improvement of a certain individual, a counter ct is used for recording it. Here in the TDE algorithm, if R_{DP} is smaller than a threshold ξ and the individual obtains no performance improvement for N continuous generations, then the individual is labeled stagnation status.

Different from those re-initialization techniques [22], some seeds are recorded during the evolution far before many individuals enter the stagnation status. A threshold τ is used for recording the seeds, and once R_{DP} satisfies $R_{DP} \leqslant \tau$, a percentage of individuals, 15%, in the current generation are recorded as the seeds. Eq. 13 presents the details of the re-initialization of a certain individual when it is in stagnation status and is not the global best individual of the population:

$$X_{i,G+1} = X_{r_0,seeds} + F_{seeds} \cdot \left(X_{r_1,seeds} - X_{r_2,seeds}\right) \tag{13}$$

where $X_{r_0,seeds}, X_{r_1,seed}$ and $X_{r_2,seeds}$ denotes randomly selected individual from the seeds, and F_{seeds} is the corresponding scale factor which satisfies Cauchy distribution, $F_{seeds} \sim C(0.5, 0.1)$. The pseudo code of the TDE algorithm is also presented in Algo. 1.

Table 6
Comparison results between the jSO algorithm and our TDE algorithm under CEC2017 on 30D optimization.

Algo. NO.	jSO Mean/Std	TDE Mean/Std
f_{c_1}	4.1797E−015/6.3595E−015(<)	0/0
f_{c_2}	1.3932E−014/2.6887E−014(<)	1.1146E−015/5.5718E−015
f_{c_3}	5.2385E−014/1.5434E−014(<)	0/0
f_{c_4}	5.8562E+001/3.2700E−014(<)	5.6197E+001/1.1722E+001
f_{c_5}	8.6497E+000/2.0934E+000(>)	1.0176E+001/2.5030E+000
f_{c_6}	3.2367E−008/9.3835E−008(<)	1.1369E−013/0.0000E+000
f_{c_7}	3.9339E+001/1.8841E+000(<)	3.9231E+001/2.0657E+000
f_{c_8}	9.4583E+000/2.1279E+000(>)	1.0198E+001/2.8515E+000
f_{c_9}	0/0(≈)	0/0
$f_{c_{10}}$	1.4808E+003/2.5423E+002(>)	1.5634E+003/2.5952E+002
$f_{c_{11}}$	7.1131E+000/1.6029E+001(>)	8.1339E+000/1.4283E+001
$f_{c_{12}}$	2.2700E+002/1.6507E+002(>)	4.6436E+002/2.4447E+002
$f_{c_{13}}$	1.7375E+001/2.6842E+000(<)	1.2644E+001/7.3317E+000
$f_{c_{14}}$	2.1897E+001/1.0315E+000(<)	2.0124E+001/8.0203E+000
$f_{c_{15}}$	1.2503E+000/9.0304E−001(>)	2.3631E+000/1.2650E+000
$f_{c_{16}}$	5.0255E+001/6.3397E+001(>)	1.6206E+002/1.2912E+002
$f_{c_{17}}$	3.1718E+001/7.5669E+000(>)	3.2135E+001/5.8276E+000
$f_{c_{18}}$	2.0419E+001/2.8814E+000(>)	2.1394E+001/8.1393E−001
$f_{c_{19}}$	5.2409E+000/1.7148E+000(<)	3.9771E+000/1.2594E+000
$f_{c_{20}}$	2.8954E+001/5.8323E+000(>)	3.5720E+001/9.5116E+000
$f_{c_{21}}$	2.1030E+002/1.8239E+000(<)	2.0916E+002/2.1797E+000
$f_{c_{22}}$	1.0000E+002/1.2158E−013(≈)	1.0000E+002/1.0047E−013
$f_{c_{23}}$	3.5735E+002/3.3985E+000(<)	3.4351E+002/5.5446E+000
$f_{c_{24}}$	4.3131E+002/2.8783E+000(<)	4.2052E+002/2.6902E+000
$f_{c_{25}}$	3.8669E+002/5.1663E−003(<)	3.8578E+002/1.7109E+000
$f_{c_{26}}$	9.9050E+002/4.5187E+001(<)	8.5932E+002/4.5594E+001
$f_{c_{27}}$	5.0229E+002/5.7289E+000(<)	4.9555E+002/5.4493E+000
$f_{c_{28}}$	3.1341E+002/3.7087E+001(<)	3.1257E+002/3.4812E+001
$f_{c_{29}}$	4.3420E+002/1.5194E+001(<)	4.3283E+002/1.0920E+001
$f_{c_{30}}$	1.9694E+003/1.7286E+001(>)	1.9993E+003/3.6477E+001
>/≈/<	11/2/17	-/-/-

Table 7
Comparison results between EDEV and our TDE algorithm under benchmarks f_{a_1} - $f_{a_{28}}$ of our test suite on 10D, 30D and 50D optimization respectively.

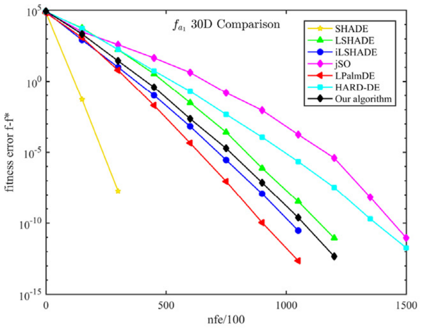
Dim.	10D		30D		50D	
Algo. NO.	EDEV Mean/Std.	TDE Mean/Std.	EDEV Mean/std.	TDE Mean/Std.	EDEV Mean/Std.	TDE Mean/std
f_{a_1}	0/0(≈)	0/0	0/0(≈)	0/0	0/0(≈)	0/0
f_{a_2}	7.336E−01/5.239E+00(<)	0/0	1.344E+04/9.769E+03(<)	2.363E−13/1.699E−13	4.021E+04/2.028E+04(<)	1.836E+02/2.784E+02
f_{a_3}	1.256E+01/2.886E+01(<)	2.798E−03/1.399E−02	5.126E+04/3.589E+05(<)	1.516E−13/1.083E−13	3.369E+06/6.225E+06(<)	5.646E+00/2.721E+01
f_{a_4}	1.463E−01/2.624E−01(<)	0/0	8.281E+00/8.901E+00(<)	5.796E−14/1.001E−13	1.201E+01/1.603E+01(<)	2.473E−10/5.318E−10
f_{a_5}	0/0(≈)	0/0	0/0(>)	9.140E−14/4.559E−14	0/0(>)	1.137E−13/0.000E+00
f_{a_6}	4.046E−07/1.478E−06(<)	0/0	1.653E+00/3.962E+00(<)	1.115E−13/4.818E−14	3.958E+01/3.027E+00(>)	4.345E+01/2.860E−03
f_{a_7}	1.650E−01/3.017E−01(<)	2.061E−05/6.756E−05	2.389E+00/2.913E+00(<)	2.265E−02/3.979E−02	2.043E+01/7.012E+00(<)	2.689E−01/3.890E−01
f_{a_8}	2.034E+01/7.869E−02(<)	2.005E+01/1.004E−01	2.094E+01/4.558E−02(<)	2.066E+01/1.915E−01	2.114E+01/3.524E−02(<)	2.088E+01/1.788E−01
f_{a_9}	4.272E+00/6.127E−01(<)	3.559E−01/7.873E−01	2.499E+01/2.253E+00(<)	2.357E+01/5.016E+00	4.334E+01/9.406E+00(<)	3.700E+01/1.553E+01
$f_{a_{10}}$	2.523E−02/1.660E−02(<)	1.095E−02/1.642E−02	4.269E−02/2.480E−02(<)	0/0	3.936E−02/3.411E−02(<)	5.556E−03/6.344E−03
$f_{a_{11}}$	0/0(≈)	0/0	0/0(>)	4.570E−14/5.335E−14	0/0(>)	2.140E−13/1.540E−13
$f_{a_{12}}$	6.383E+00/1.441E+00(<)	2.839E+00/1.844E+00	2.646E+01/6.972E+00(<)	6.963E+00/1.771E+00	5.622E+01/1.298E+01(<)	1.791E+01/2.218E+00
$f_{a_{13}}$	8.204E+00/2.614E+00(<)	2.537E+00/2.167E+00	4.568E+01/1.156E+01(<)	8.497E+00/3.447E+00	1.322E+02/2.653E+01(<)	3.367E+01/9.272E+00
$f_{a_{14}}$	5.778E−03/2.411E−02(>)	9.797E−03/2.294E−02	2.104E−01/1.495E−01(<)	7.348E−03/1.237E−02	1.139E+00/5.642E−01(<)	3.506E−02/1.882E−02
$f_{a_{15}}$	8.613E+02/2.687E+02(<)	4.079E+02/1.793E+02	4.239E+03/5.838E+02(<)	2.797E+03/3.680E+02	8.145E+03/4.827E+02(<)	6.563E+03/4.227E+02
$f_{a_{16}}$	1.135E+00/2.044E−01(<)	1.115E−01/1.439E−01	2.432E+00/2.750E−01(<)	3.168E−01/3.518E−01	3.104E+00/4.283E−01(<)	8.392E−01/6.616E−01
$f_{a_{17}}$	1.012E+01/3.221E−08(≈)	1.012E+01/1.794E−15	3.043E+01/9.430E−07(≈)	3.043E+01/4.497E−14	5.079E+01/1.451E−04(≈)	5.079E+01/2.759E−09
$f_{a_{18}}$	2.634E+01/4.546E+00(<)	1.510E+01/2.446E+00	9.374E+01/7.365E+00(<)	5.459E+01/5.739E+00	1.635E+02/2.588E+01(<)	1.087E+02/9.774E+00
$f_{a_{19}}$	4.338E−01/6.579E−02(<)	2.308E−01/4.385E−02	2.006E+00/1.779E−01(<)	1.139E+00/8.503E−02	3.845E+00/2.501E−01(<)	2.371E+00/1.604E−01
$f_{a_{20}}$	2.535E+00/2.889E−01(<)	1.720E+00/3.292E−01	1.096E+01/4.724E−01(<)	9.098E+00/4.044E−01	1.994E+01/5.431E−01(<)	1.770E+01/5.629E−01
$f_{a_{21}}$	4.002E+02/0.000E+00(≈)	4.002E+02/0.000E+00	3.183E+02/6.912E+01(<)	2.902E+02/3.003E+01	7.794E+02/3.800E+02(>)	9.314E+02/3.523E+02
$f_{a_{22}}$	2.857E+01/1.885E+01(<)	2.338E+00/3.298E+00	1.222E+02/2.650E+01(<)	1.057E+02/4.269E−01	3.638E+01/4.918E+01(<)	1.147E+01/6.944E−01
$f_{a_{23}}$	8.453E+02/2.698E+02(<)	3.266E+02/1.869E+02	4.390E+03/4.769E+02(<)	2.792E+03/3.787E+02	8.673E+03/7.267E+02(<)	6.388E+03/4.381E+02
$f_{a_{24}}$	2.033E+02/1.279E+01(<)	2.000E+02/4.329E−06	2.304E+02/2.881E+01(<)	2.000E+02/3.789E−03	2.891E+02/5.061E+01(<)	2.005E+02/9.090E−01
$f_{a_{25}}$	2.031E+02/2.115E+01(<)	1.948E+02/2.110E+01	2.763E+02/7.340E+00(<)	2.064E+02/1.509E+01	3.495E+02/8.113E+00(<)	2.817E+02/7.009E+00
$f_{a_{26}}$	1.510E+02/4.506E+01(<)	1.055E+02/1.368E+01	2.000E+02/3.922E−03(≈)	2.000E+02/1.785E−13	2.046E+02/2.316E+01(>)	2.282E+02/4.629E+01
$f_{a_{27}}$	3.486E+02/6.975E+01(<)	3.000E+02/3.000E+02	6.754E+02/2.128E+02(<)	3.001E+02/1.080E−01	6.658E+02/2.279E+02(<)	3.130E+02/1.613E+01
$f_{a_{28}}$	2.765E+02/6.508E+01(>)	3.000E+02/4.548E−14	3.000E+02/0.000E+00(≈)	3.000E+02/1.326E−13	3.000E+02/0.000E+00(>)	4.000E+02/2.704E−13
>/≈/<	2/5/21	-/-/-	2/4/22	-/-/-	6/2/20	-/-/-

Algorithm 1: Pseudo code of TDE algorithm

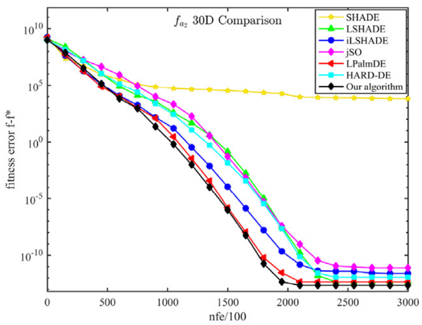
Input: Number of dimensions D, boundary constraints $\left[R_{\min}^{D}, R_{\max}^{D}\right]$, the objective $f(X)$, maximum number of function evaluations $nfe_{\max}$.
Output: Global best individual X_{best}, global best fitness value $f(X_{gbest})$, number of function evaluations nfe;
Initialize the population $P = \{X_1, X_2, \ldots, X_{PS}\}, K = 4, P(j) = \frac{1}{K}, A = \varnothing, B = \varnothing, \mu_F = 0.3, \mu_{CR} = 0.8, p = 0.25 \sim 0.05, r^{rac,A} = 1, r^{rac,B} = 3,$ $PS = 25 \cdot lnD \cdot \sqrt{D} \sim K, N = 2 \cdot D, \xi = 0.001, ct = 0, G = 1$;
for $i = 1; i \leqslant PS; i++$ **do**
  $X_{i,G} = X_i$; Calculate $f(X_{i,G})$;
end for
$nfe = PS$;
Label $X_{gbest,G}$ and $f(X_{gbest,G})$;
while $nfe < nfe_{\max}$ **do**
  Generate the top p elites of the current population $\boldsymbol{P}$;
  Divide the population into K groups according to Stochastic Universal Selection [17];
  Generate F, CR and F_w of all individuals;
  for $i = 1; i \leqslant PS; i++$ **do**
    Generating trial vectors $U_{i,G}$ according to Eq. 6;
    Calculate fitness value $f(U_{i,G})$;
  end for
  $nfe = nfe + PS$;
  for $i = 1; i \leqslant PS; i++$ **do**
    if $f(U_{i,G}) < f(X_{i,G})$ **then**;
      $X_{i,G+1} = U_{i,G}$;
    else
      $X_{i,G+1} = X_{i,G}$;
    end if
  end for
  if $S_F \neq \varnothing$ **then**
    Update $P(j)$, μ_F, and $\mu_{CR_{idx}}$ according to Eq. 7, Eq. 8, and Eq. 9 respectively;
  end if
  Adjust archives A and B;
  Label X_{gbest} and $f(X_{gbest})$;
  Calculate the counter ct for all of individuals and R_{DP} according to Eq. 11;
  if $R_{DP} < \tau$ **then**
    for $i = 1; i \leqslant PS; i++$ **do**
      if $ct(i) > N$ && $R_{DP} < \xi$ **then**
        Update the individual according to Eq. 13 and calculate its fitness value $f(X_{i,G})$;
        $nfe = nfe + 1$;
      end if
    end for
  end if
  Adjust population size according to Eq. 10;
  $G = G + 1$;
end while
$f(X_{gbest}) = f(X_{gbest,G}), X_{gbest} = X_{gbest,G}$;
return $f(X_{gbest}), X_{gbest}$ and nfe;

5. Experiment analysis

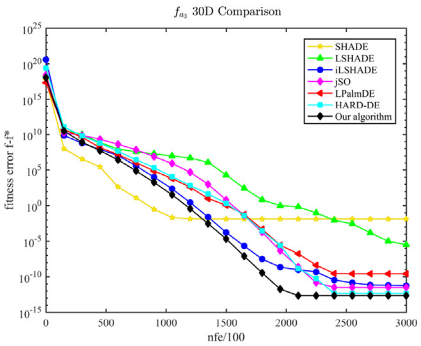
In this section, a large test suite containing 88 benchmarks from CEC2013 test suite [11], CEC2014 test suite [10] and CEC2017 test suite [34] are used for the validation of our TDE algorithm. The reason why we employ such a large test suite rather than a single standard test suite is that algorithm validation on only one test suite usually has over-fitting problems. In our test suite, the CEC2013 test suite containing 28 benchmark functions are labeled as $f_{a_1} - f_{a_{28}}$ in our experiment. These 28 benchmark functions can be further divided into three groups: $f_{a_1} - f_{a_5}$ are in uni-modal function group, $f_{a_6} - f_{a_{20}}$ are in multi-modal function group, and $f_{a_{21}} - f_{a_{28}}$ are in composition function group. The CEC2014 test suit containing 30 bench-



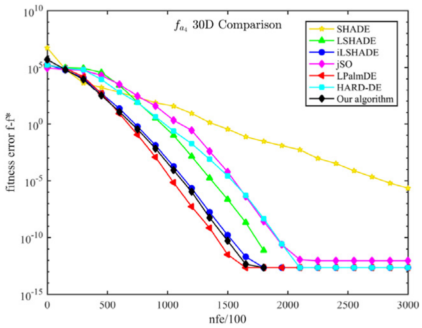
(a) f_{a_1}



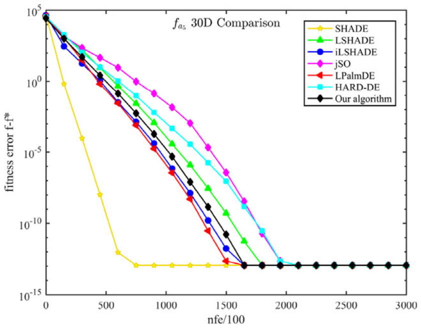
(b) f_{a_2}



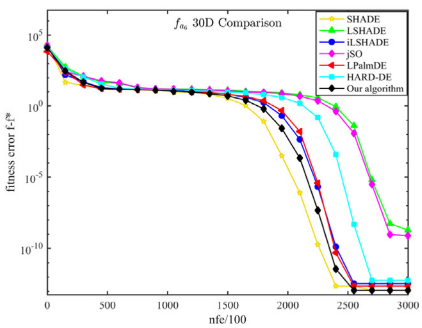
(c) f_{a_3}



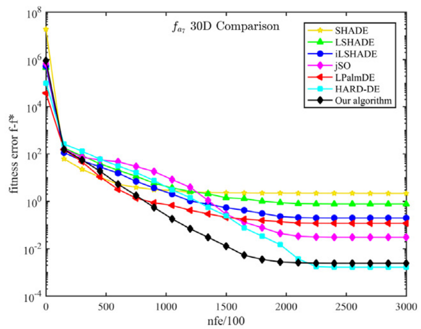
(d) f_{a_4}



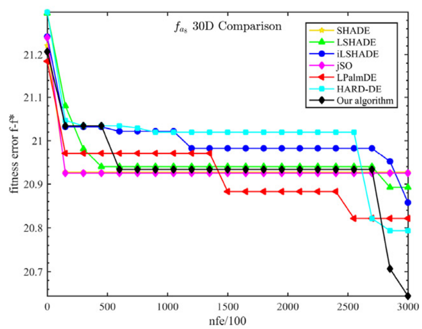
(e) f_{a_5}



(f) f_{a_6}

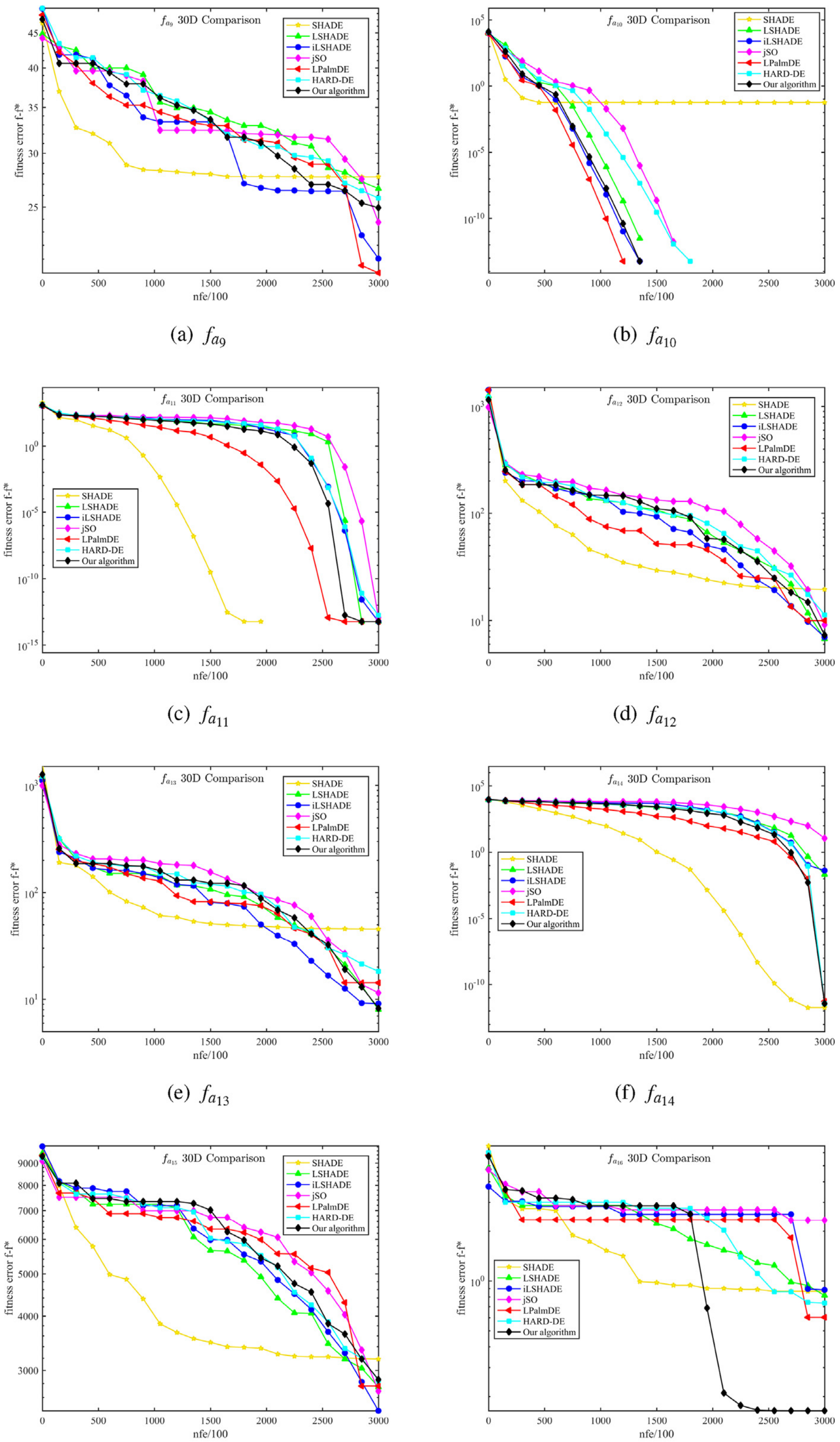


(g) f_{a_7}



(h) f_{a_8}

Fig. 3. Here presents the convergence speed comparison by employing the median value of 51 runs obtained by each algorithm on 30D optimization. There are total 28 comparison figures and the first 8 figures are presented here.



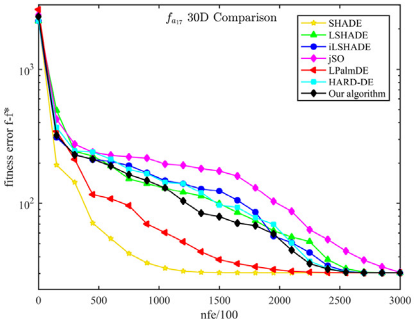
(a) f_{a_9} (b) $f_{a_{10}}$

(c) $f_{a_{11}}$ (d) $f_{a_{12}}$

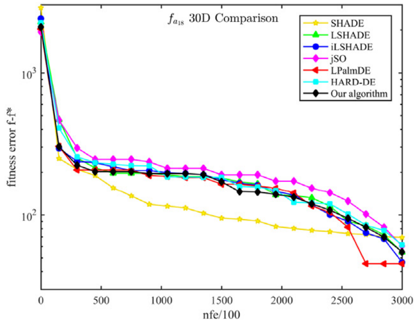
(e) $f_{a_{13}}$ (f) $f_{a_{14}}$

(g) $f_{a_{15}}$ (h) $f_{a_{16}}$

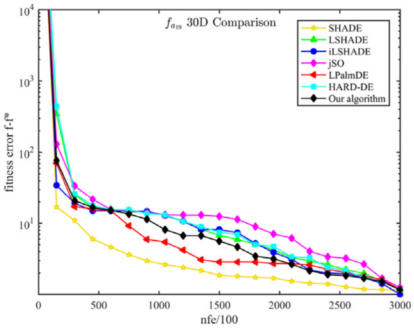
Fig. 4. As a continued part from Fig.1, comparisons on benchmarks f_{a_9}-$f_{a_{16}}$.



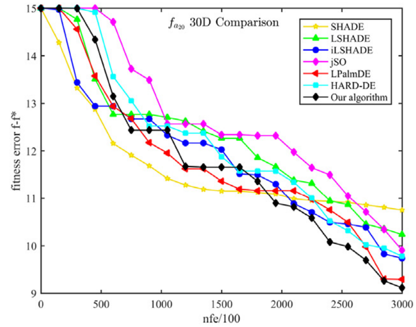
(a) $f_{a_{17}}$



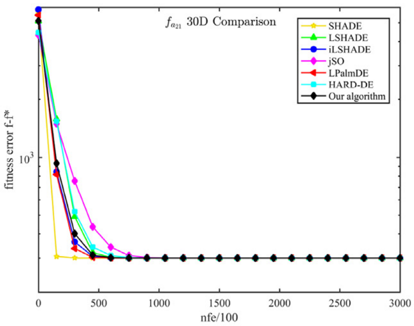
(b) $f_{a_{18}}$



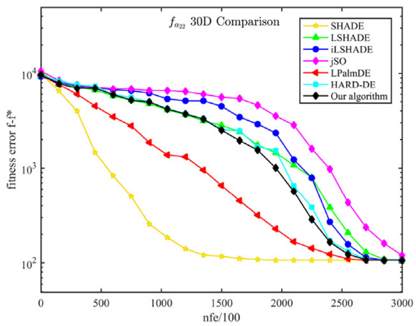
(c) $f_{a_{19}}$



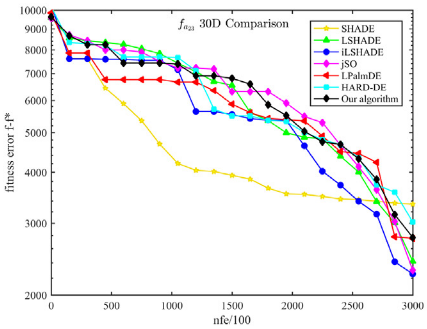
(d) $f_{a_{20}}$



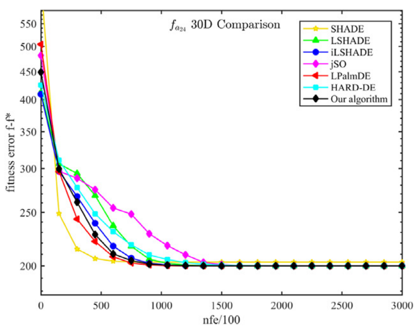
(e) $f_{a_{21}}$



(f) $f_{a_{22}}$

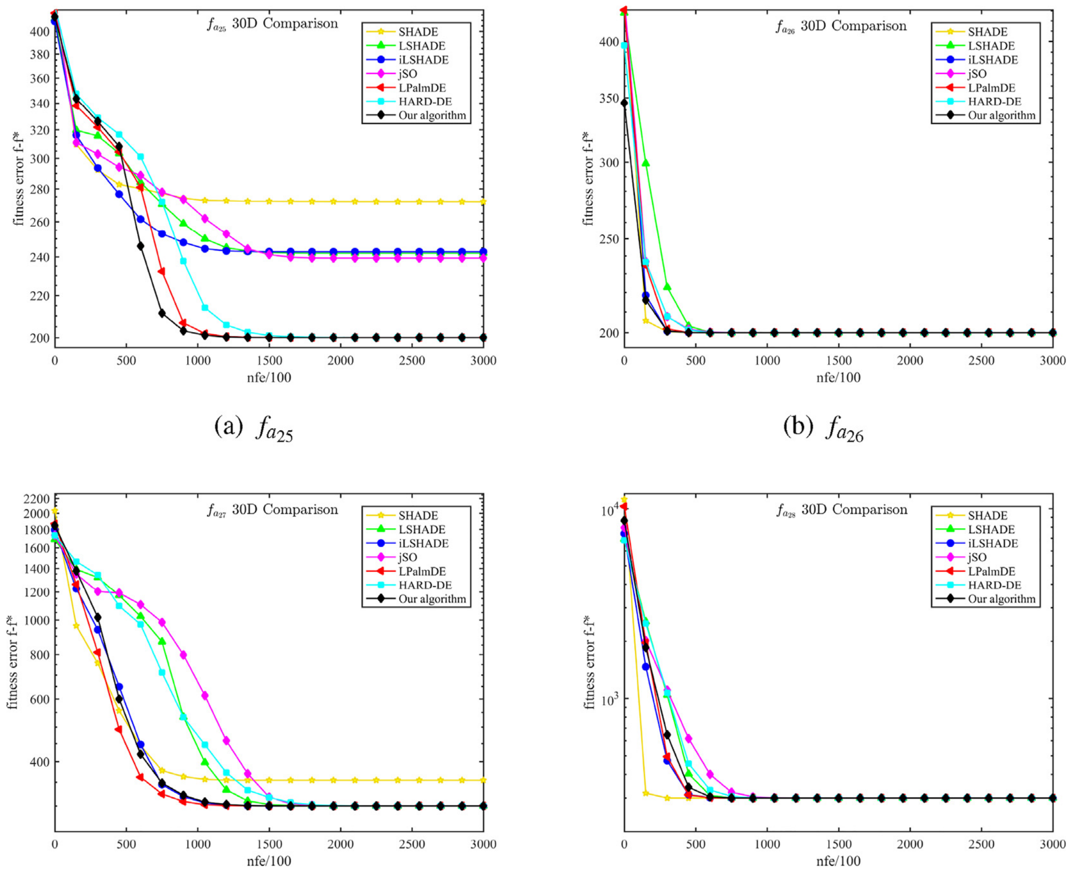


(g) $f_{a_{23}}$



(h) $f_{a_{24}}$

Fig. 5. As a continued part from Fig.2, comparisons on benchmarks $f_{a_{17}}$-$f_{a_{24}}$.



(a) $f_{a_{25}}$ (b) $f_{a_{26}}$

(c) $f_{a_{27}}$ (d) $f_{a_{28}}$

Fig. 6. As a continued part from Fig.3, comparisons on benchmarks $f_{a_{25}}$-$f_{a_{28}}$.

mark functions are labeled as $f_{b_1} - f_{b_{30}}$, and the CEC2017 test suit containing 30 benchmark functions are labeled as $f_{c_1} - f_{c_{30}}$. Moreover, $f_{b_1} - f_{b_3}$ and $f_{c_1} - f_{c_3}$ are in the unimodal function group; $f_{b_4} - f_{b_{16}}$ and $f_{c_4} - f_{c_{10}}$ are in basic multimodal function group; $f_{b_{17}} - f_{b_{22}}$ and $f_{c_{11}} - f_{c_{20}}$ are in hybrid function group; $f_{b_{23}} - f_{b_{30}}$ and $f_{c_{21}} - f_{c_{30}}$ are in composition function group. Our TDE algorithm is verified under the large test suite in comparison with several well-known DE variants including SHADE, LSHADE, iLSHADE, jSO, LPalmDE and HARD-DE. The parameter settings for all algorithms in comparison are listed in Table 1.

All experiments were performed on Windows 10 operating system of a personal computer with Inter(R) Core(TM) i7-7700HQ CPU @ 2.80 GHz. 51 runs were performed on each benchmark function of an algorithm, and the fitness errors $\Delta f = f(X) - f(X^*)$ of the algorithm on each benchmark of the total 51 runs were recorded first, and then the mean and standard deviation of the fitness errors were calculated. Wilcoxon signed rank test with the significant level $\alpha = 0.05$ was used for the algorithm comparison, and symbols "$>$", "$\approx$", "$<$" in the tables behind "Mean/Std" represent "Better performance", "Similar performance" and "Worse performance" respectively under this criterion. The terminal function evaluation is $nfe_{\max} = 10,000 \cdot D$ where D is the dimension of the objective. By the way, a value smaller than $eps = 2.2204e - 16$ is considered as 0 in this paper.

5.1. Optimization accuracy

In this part, we examine the TDE algorithm from optimization accuracy perspective. First, the optimization results of the above mentioned six well-know DE variants and our TDE algorithm are compared under the CEC2013 test suit for real-parameter single objective optimization on 10D, 30D and 50D, and these results are listed in Table 2, Table 3 and Table 4 respectively.

For 10D optimization, the TDE algorithm obtains an overall better performance in comparison with these well-known DE variants. Comparing with SHADE, the TDE algorithm guarantees performance improvements on 18 benchmarks and it also obtains 7 similar results out of the 28 functions; Comparing with LSHADE, the TDE algorithm shows better or similar performance on 19 out of the 28 benchmark functions; Comparing with iLSHADE, the TDE algorithm shows better or similar performance on 20 out of the 28 benchmark functions; In comparison with jSO, similar results or better performance

Table 8
Comparison of different τ settings in 30-D optimization under f_{a_1}–$f_{a_{28}}$.

τ=0.2 NO.	τ=0.3 Mean/Std	τ=0.4 Mean/Std	τ=0.5 Mean/Std	τ=0.7 Mean/Std	τ=0.8 Mean/Std	τ=0.6 Mean/Std
f_{a_1}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
f_{a_2}	2.4075E−013/ 1.5362E−013(≈)	2.4967E−013/ 2.9560E−013(<)	1.9171E−013/ 1.1480E−013(>)	2.2737E−013/ 2.1809E−013(>)	(≈)	2.4075E−013/ 1.7263E−013
f_{a_3}	1.1798E−010/ 5.9008E−010(>)	1.7763E−005/ 1.2685E−004(<)	4.1856E−009/ 2.9887E−008(<)	4.3489E−010/ 2.5889E−009(>)	6.7026E−011/ 4.4953E−010(>)	1.0001E−009/ 7.1167E−009
f_{a_4}	1.7833E−014/ 6.1737E−014(<)	1.7833E−014/ 6.1737E−014(<)	1.7833E−014/ 6.1737E−014(<)	1.7833E−014/ 6.1737E−014(<)	2.2292E−014/ 6.8286E−014(<)	1.3375E−014/ 5.4032E−014
f_{a_5}	8.4708E−014/ 5.0038E−014(<)	9.3624E−014/ 4.3771E−014(<)	8.9166E−014/ 4.7224E−014(<)	8.9166E−014/ 4.7224E−014(<)	9.1395E−014/ 4.5586E−014(<)	8.2479E−014/ 5.1240E−014
f_{a_6}	1.8725E−013/ 1.4879E−013(<)	1.3821E−013/ 7.6479E−014(>)	3.6558E−013/ 1.4940E−012(<)	1.2260E−013/ 6.7615E−014(>)	1.9839E−013/ 3.6297E−013(<)	1.5158E−013/ 1.1519E−013
f_{a_7}	1.8811E−002/ 3.9324E−002(>)	2.2862E−002/ 4.5744E−002(<)	2.5380E−002/ 4.3955E−002(<)	2.1579E−002/ 4.7455E−002(<)	2.3462E−002/ 5.9771E−002(<)	1.9304E−002/ 4.2915E−002
f_{a_8}	2.0788E+001/ 1.6143E−001(>)	2.0760E+001/ 1.8257E−001(>)	2.0754E+001/ 1.5814E−001(>)	2.0756E+001/ 1.6337E−001(>)	2.0711E+001/ 1.6757E−001(>)	2.0801E+001/ 1.5368E−001
f_{a_9}	2.4650E+001/ 3.7474E+000(>)	2.5057E+001/ 2.3363E+000(>)	2.4177E+001/ 3.9921E+000(>)	2.4272E+001/ 3.8558E+000(>)	2.4050E+001/ 4.5148E+000(>)	2.5475E+001/ 3.0555E+000
$f_{a_{10}}$	0/0(>)	0/0(>)	0/0(>)	0/0(>)	0/0(>)	1.4502E−004/ 1.0357E−003
$f_{a_{11}}$	3.0094E−014/ 3.0828E−014(>)	3.5666E−014/ 3.9318E−014(≈)	4.1239E−014/ 4.1114E−014(<)	3.7896E−014/ 3.3468E−014(<)	2.6750E−014/ 3.4768E−014(>)	3.5666E−014/ 3.9318E−014
$f_{a_{12}}$	7.3591E+000/ 1.4208E+000(<)	7.2058E+000/ 1.6411E+000(<)	7.0818E+000/ 1.5278E+000	6.8981E+000/ 1.6714E+000(>)	7.3195E+000/ 1.5775E+000(<)	7.1203E+000/ 1.3538E+000
$f_{a_{13}}$	9.3975E+000/ 3.9803E+000(<)	9.3314E+000/ 3.5348E+000(<)	1.0289E+001/ 4.4265E+000(<)	1.0913E+001/ 5.6145E+000(<)	9.1944E+000/ 4.4585E+000(>)	9.2989E+000/ 4.2780E+000
$f_{a_{14}}$	1.6737E−002/ 1.7178E−002(<)	2.2481E−002/ 2.0753E−002(<)	2.0411E−002/ 2.0606E−002(<)	1.7145E−002/ 1.9841E−002(<)	1.4704E−002/ 1.5735E−002(<)	1.2247E−002/ 1.3920E−002
$f_{a_{15}}$	2.8557E+003/ 2.7720E+002(<)	2.8241E+003/ 2.8189E+002(<)	2.8369E+003/ 2.8359E+002(<)	2.7474E+003/ 3.1840E+002(>)	2.8693E+003/ 2.8602E+002(<)	2.8204E+003/ 2.9655E+002
$f_{a_{16}}$	6.4355E−001/ 4.1367E−001(<)	6.2781E−001/ 4.7798E−001(<)	5.8265E−001/ 4.7245E−001(>)	7.1103E−001/ 5.0253E−001(<)	6.8561E−001/ 5.3423E−001(<)	6.2281E−001/ 4.7959E−001
$f_{a_{17}}$	3.0434E+001/ 4.7842E−014(≈)	3.0434E+001/ 5.7116E−014(≈)	3.0434E+001/ 5.3324E−014(≈)	3.0434E+001/ 5.2970E−014(≈)	3.0434E+001/ 5.9624E−014(≈)	3.0434E+001/ 4.6384E−014
$f_{a_{18}}$	5.6227E+001/ 4.5218E+000(<)	5.6063E+001/ 5.5631E+000(<)	5.5363E+001/ 5.0515E+000(>)	5.6223E+001/ 5.2313E+000(<)	5.5376E+001/ 5.1420E+000(>)	5.5678E+001/ 5.3274E+000
$f_{a_{19}}$	1.1273E+000/ 9.6321E−002(>)	1.1513E+000/ 1.0430E−001(<)	1.1561E+000/ 8.8069E−002(<)	1.1128E+000/ 1.0578E−001(>)	1.1497E+000/ 9.0850E−002(<)	1.1400E+000/ 9.9348E−002
$f_{a_{20}}$	9.2176E+000/ 3.6896E−001(<)	9.1875E+000/ 3.9557E−001(<)	9.3992E+000/ 1.1350E+000(<)	9.2245E+000/ 4.0743E−001(<)	9.2349E+000/ 3.1522E−001(<)	9.1616E+000/ 3.5539E−001
$f_{a_{21}}$	3.0171E+002/ 3.4946E+001(<)	3.0171E+002/ 3.4946E+001(<)	3.0367E+002/ 3.1788E+001(<)	2.8824E+002/ 3.2540E+001(>)	2.9412E+002/ 2.3764E+001(≈)	2.9412E+002/ 2.3764E+001
$f_{a_{22}}$	1.0604E+002/ 1.1366E+000(<)	1.0581E+002/ 5.0510E−001(>)	1.0594E+002/ 1.1562E+000(<)	1.0602E+002/ 1.1504E+000(<)	1.0612E+002/ 1.2618E+000(<)	1.0586E+002/ 3.9296E−001
$f_{a_{23}}$	2.7379E+003/ 3.9632E+002(>)	2.7271E+003/ 2.9688E+002(>)	2.7700E+003/ 3.5142E+002(>)	2.7938E+003/ 3.1525E+002(>)	2.7349E+003/ 3.3276E+002(>)	2.8345E+003/ 2.9686E+002
$f_{a_{24}}$	2.0000E+002/ 9.9035E−004(≈)	2.0000E+002/ 7.0716E−003(≈)	2.0000E+002/ 1.8169E−003(≈)	2.0000E+002/ 3.9256E−003(≈)	2.0000E+002/ 4.1184E−003(≈)	2.0000E+002/ 2.5013E−003
$f_{a_{25}}$	2.0446E+002/ 1.3694E+001(<)	2.0236E+002/ 9.5482E+000(>)	2.0482E+002/ 1.3381E+001(<)	2.0351E+002/ 1.2209E+001(<)	2.0266E+002/ 1.0794E+001(>)	2.0322E+002/ 1.1177E+001
$f_{a_{26}}$	2.0000E+002/ 1.7845E−013(≈)	2.0000E+002/ 1.3613E−013(≈)	2.0000E+002/ 1.3895E−013(≈)	2.0000E+002/ 1.4262E−013(≈)	2.0000E+002/ 1.4080E−013(≈)	2.0000E+002/ 1.9306E−013
$f_{a_{27}}$	3.0011E+002/ 3.2212E−001(<)	3.0009E+002/ 2.8260E−001(<)	3.0012E+002/ 3.0937E−001(<)	3.0011E+002/ 3.2147E−001(<)	3.0011E+002/ 4.4772E−001(<)	3.0008E+002/ 4.0988E−001
$f_{a_{28}}$	3.0000E+002/ 3.5471E−013(≈)	3.0000E+002/ 3.0968E−013(≈)	3.0000E+002/ 2.6160E−013(≈)	3.0000E+002/ 2.9602E−013(≈)	3.0000E+002/ 2.8535E−013(≈)	3.0000E+002/ 2.8895E−013
>/≈/<	8/6/14	7/6/15	8/5/15	11/5/12	9/7/12	-/-/-

improvements are obtained by our TDE algorithm on 19 functions out of the total 28 functions; In comparison with LPalmDE, our TDE algorithm obtains similar or better performance improvements on 21 benchmarks out of the 28 functions; In comparison with HARD-DE, our TDE algorithm obtains similar or better or performance improvements on 21 out of the 28 benchmark functions. For 30D optimization and 50D optimization, we also can see from Table 3 and Table 4 that our TDE algorithm obtains an overwhelming better performance in comparison with these state-of-the-art DE variants. It secures 23 performance improvements and 3 similar results in comparison with SHADE, secures 19 performance improvements and 4 similar results in comparison with LSHADE and iLSHADE, obtains 22 performance improvements and 4 similar results in comparison with jSO, obtains 18 performance improvements and 5 similar results in comparison with LPalmDE, and obtains 23 perfor-

Table 9
Comparison of different ρ in 30-D optimization.

ρ NO.	$\rho = 0$ Mean/Std	$\rho = 1/3$ Mean/Std	$\rho = 1/2$ Mean/Std	$\rho = 1$ Mean/Std	$\rho = 2/3$ Mean/Std
f_{a_1}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
f_{a_2}	2.5412E−013/ 2.7151E−013(<)	2.0954E−013/ 1.4267E−013(<)	1.4712E−013/ 1.0974E−013(>)	2.3629E−013/ 1.2831E−013(<)	2.0062E−013/ 9.8030E−014
f_{a_3}	5.0379E−013/ 1.1372E−012(<)	4.3691E−013/ 1.2099E−012(<)	2.1682E−006/ 1.5484E−005(<)	6.7031E−009/ 4.3540E−008(<)	1.5158E−013/ 1.7415E−013
f_{a_4}	7.5791E−014/ 1.0825E−013(<)	1.7833E−014/ 6.1737E−014(>)	2.6750E−014/ 7.3986E−014(>)	6.2416E−014/ 1.0248E−013(<)	5.3500E−014/ 9.7408E−014
f_{a_5}	9.8083E−014/ 3.9511E−014(<)	1.0254E−013/ 3.4143E−014(<)	9.1395E−014/ 4.5586E−014(>)	6.2416E−014/ 5.7132E−014(>)	9.5854E−014/ 4.1756E−014
f_{a_6}	2.3183E−013/ 3.0671E−013(<)	1.3375E−013/ 1.0093E−013(>)	1.4712E−013/ 1.7060E−013(<)	1.1146E−013/ 5.3276E−014(>)	1.4489E−013/ 2.4602E−013
f_{a_7}	2.0505E−002/ 7.8724E−002(>)	1.2725E−002/ 2.9665E−002(>)	1.8604E−002/ 2.8239E−002(>)	2.2435E−002/ 4.1648E−002(>)	3.2622E−002/ 5.7992E−002
f_{a_8}	2.0807E+001/ 1.5439E−001(<)	2.0753E+001/ 1.7113E−001(<)	2.0671E+001/ 1.7674E−001(<)	2.0661E+001/ 2.2669E−001(<)	2.0602E+001/ 1.8824E−001
f_{a_9}	2.5355E+001/2.6896E +000(>)	2.4462E+001/4.2012E +000(>)	2.3662E+001/5.0993E +000(>)	2.4168E+001/4.5296E +000(>)	2.5484E+001/2.7447E +000
$f_{a_{10}}$	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
$f_{a_{11}}$	3.3437E−014/ 2.8254E−014(>)	3.4552E−014/ 2.8029E−014(>)	4.0125E−014/ 4.4460E−014(>)	2.5635E−014/ 3.4695E−014(>)	5.4614E−014/ 6.0115E−014
$f_{a_{12}}$	8.3010E+000/1.9206E +000(<)	7.3839E+000/1.9502E +000(<)	6.7542E+000/1.7879E +000(<)	6.7251E+000/1.4222E +000(>)	6.7452E+000/1.8864E +000
$f_{a_{13}}$	1.1823E+001/4.5162E +000(<)	8.6469E+000/3.8286E +000(>)	7.2988E+000/2.7025E +000(>)	8.2680E+000/3.8503E +000(>)	8.6871E+000/3.4658E +000
$f_{a_{14}}$	1.7665E−002/ 1.8235E−002(<)	1.9186E−002/ 1.9901E−002(<)	1.3334E−002/ 1.5446E−002(<)	7.5791E−002/ 3.7435E−002(<)	1.0206E−002/ 1.2048E−002
$f_{a_{15}}$	2.8096E+003/3.0925E +002(>)	2.8258E+003/3.0670E +002(>)	2.8742E+003/2.8789E +002(<)	2.8226E+003/3.0728E +002(>)	2.8700E+003/3.2088E +002
$f_{a_{16}}$	6.5859E−001/ 5.2125E−001(<)	6.5245E−001/ 4.8793E−001(<)	6.2729E−001/ 4.4679E−001(<)	4.0182E−001/ 3.5790E−001(<)	3.2866E−001/ 3.3817E−001
$f_{a_{17}}$	3.0434E+001/ 9.4299E−007(≈)	3.0434E+001/ 5.2714E−014(≈)	3.0434E+001/ 4.7332E−014(≈)	3.0440E+001/ 7.3452E−003(<)	3.0434E+001/ 9.4299E−007
$f_{a_{18}}$	5.8488E+001/4.3205E +000(<)	5.6130E+001/4.9969E +000(<)	5.4249E+001/5.6752E +000(<)	5.4359E+001/5.2192E +000(<)	5.4078E+001/5.6070E +000
$f_{a_{19}}$	1.1775E+000/ 9.1472E−002(<)	1.1445E+000/ 1.0855E−001(<)	1.1479E+000/ 7.8924E−002(<)	1.1093E+000/ 8.1206E−002(>)	1.1361E+000/ 9.3554E−002
$f_{a_{20}}$	9.4716E+000/ 6.6411E−001(<)	9.2176E+000/ 3.7973E−001(<)	9.2111E+000/ 3.5355E−001(<)	9.1902E+000/ 4.2504E−001(<)	9.1841E+000/ 3.8751E−001
$f_{a_{21}}$	2.9975E+002/3.7738E +001(<)	2.9864E+002/4.7270E +001(<)	2.9693E+002/3.1662E +001(<)	2.9693E+002/3.1662E +001(<)	2.9608E+002/1.9604E +001
$f_{a_{22}}$	1.0589E+002/ 3.6570E−001(<)	1.0588E+002/ 3.2138E−001(<)	1.0595E+002/ 7.1063E−001(<)	1.0875E+002/2.7076E +000(<)	1.0574E+002/ 4.9410E−001
$f_{a_{23}}$	2.7537E+003/3.1861E +002(>)	2.7956E+003/2.4039E +002(<)	2.7920E+003/3.4954E +002(<)	2.6761E+003/3.4069E +002(>)	2.7632E+003/3.1426E +002
$f_{a_{24}}$	2.0000E+002/ 5.3291E−003(≈)	2.0000E+002/ 2.3861E−002(≈)	2.0000E+002/ 9.2847E−003(≈)	2.0000E+002/ 4.5970E−003(≈)	2.0000E+002/ 6.2106E−003
$f_{a_{25}}$	2.0434E+002/1.3392E +001(<)	2.0339E+002/1.1747E +001(<)	2.0528E+002/1.4675E +001(<)	2.0327E+002/1.1335E +001(<)	2.0258E+002/1.0424E +001
$f_{a_{26}}$	2.0000E+002/ 1.3422E−013(≈)	2.0000E+002/ 1.3895E−013(≈)	2.0000E+002/ 1.3707E−013(≈)	2.0000E+002/ 1.3080E−013(≈)	2.0000E+002/ 1.7626E−013
$f_{a_{27}}$	3.0007E+002/ 1.5689E−001(<)	3.0016E+002/ 4.7839E−001(<)	3.0007E+002/ 2.2697E−001(<)	3.0006E+002/ 1.9955E−001(<)	3.0003E+002/ 7.4742E−002
$f_{a_{28}}$	3.0000E+002/ 3.3523E−013(≈)	3.0000E+002/ 3.2595E−013(≈)	3.0000E+002/ 2.8895E−013(≈)	3.0000E+002/0.0000E +000(≈)	3.0000E+002/ 1.7316E−013
>/≈/<	5/6/17	7/6/15	7/6/15	10/5/13	-/-/-

mance improvements and 5 similar results in comparison with HARD-DE from 30D view. It secures 23 performance improvements and 2 similar results in comparison with SHADE, secures 18 performance improvements and 2 similar results in comparison with LSHADE and jSO, obtains 17 performance improvements and 2 similar results in comparison with iLSHADE, obtains 17 performance improvements and 3 similar results in comparison with LPalmDE and HARD-DE from 50D view. By comparing the results on 10D, 30D and 50D optimization, we also can see that the performance do not deteriorate with the increasing dimensions in comparison with these state-of-the-art DE variants. To summarize, our TDE algorithm is very competitive in comparison with these state-of-the-art DE variants from the optimization accuracy perspective of view.

Table 10
Comparison of different population size reduction schemes for 30-D optimization under f_{a_1}–$f_{a_{28}}$.

Schemes: NO.	Linear reduction Mean/Std	Our scheme Mean/Std
f_{a_1}	0/0(≈)	0/0
f_{a_2}	3.4329E−013/3.8953E−013(<)	2.3629E−013/1.6991E−013
f_{a_3}	1.3492E−009/7.1833E−009(<)	1.5158E−013/1.0825E−013
f_{a_4}	5.7958E−014/1.0008E−013(≈)	5.7958E−014/1.0008E−013
f_{a_5}	9.8083E−014/3.9511E−014(<)	9.1395E−014/4.5586E−014
f_{a_6}	1.3152E−013/1.4974E−013(<)	1.1146E−013/4.8181E−014
f_{a_7}	2.0729E−002/4.0235E−002(>)	2.2649E−002/3.9786E−002
f_{a_8}	2.0665E+001/2.0377E−001(<)	2.0658E+001/1.9145E−001
f_{a_9}	2.4766E+001/4.1759E+000(<)	2.3565E+001/5.0159E+000
$f_{a_{10}}$	0/0(≈)	0/0
$f_{a_{11}}$	4.0125E−014/4.8626E−014(>)	4.5698E−014/5.3348E−014
$f_{a_{12}}$	7.0165E+000/1.5898E+000(<)	6.9632E+000/1.7706E+000
$f_{a_{13}}$	9.3142E+000/4.2565E+000(<)	8.4974E+000/3.4473E+000
$f_{a_{14}}$	5.7886E−003/1.1052E−002(>)	7.3480E−003/1.2368E−002
$f_{a_{15}}$	2.7920E+003/3.7897E+002(>)	2.7972E+003/3.6795E+002
$f_{a_{16}}$	3.4375E−001/3.6155E−001(<)	3.1683E−001/3.5176E−001
$f_{a_{17}}$	3.0434E+001/4.4880E−014(≈)	3.0434E+001/4.4970E−014
$f_{a_{18}}$	5.3936E+001/4.1458E+000(>)	5.4593E+001/5.7393E+000
$f_{a_{19}}$	1.1156E+000/1.0007E−001(>)	1.1393E+000/8.5026E−002
$f_{a_{20}}$	9.2248E+000/4.4132E−001(<)	9.0976E+000/4.0442E−001
$f_{a_{21}}$	2.9804E+002/1.4003E+001(<)	2.9020E+002/3.0033E+001
$f_{a_{22}}$	1.0575E+002/4.2497E−001(<)	1.0573E+002/4.2686E−001
$f_{a_{23}}$	2.7815E+003/3.3183E+002(>)	2.7925E+003/3.7871E+002
$f_{a_{24}}$	2.0000E+002/1.0383E−002(≈)	2.0000E+002/3.7890E−003
$f_{a_{25}}$	2.0000E+002/6.7900E−004(>)	2.0644E+002/1.5094E+001
$f_{a_{26}}$	2.0000E+002/1.9104E−013(≈)	2.0000E+002/1.7845E−013
$f_{a_{27}}$	3.0014E+002/3.9372E−001(<)	3.0005E+002/1.0796E−001
$f_{a_{28}}$	3.0000E+002/1.0263E−013(≈)	3.0000E+002/1.3258E−013
>/≈/<	8/7/13	-/-/-

The TDE algorithm was also compared with the winner algorithm, the LSHADE algorithm, of the CEC2014 competition on single-objective real-parameter optimization, moreover, it was also compared with the winner DE variant, the jSO algorithm, of the CEC2017 competition on single-objective real-parameter optimization. The results on 30D optimization are listed in Table 5 and Table 6 respectively. We can see that our TDE algorithm obtains 14 performance improvements and 6 similar results out of the total 30 benchmarks in comparison with LSHADE and it also obtains 17 performance improvements and 2 similar results out of 30 benchmarks in comparison with jSO. As a result, our TDE algorithm secures overwhelming better performance under our test suite containing 88 benchmarks.

According to the associate editor's suggestion, we also make a brief comparison between our TDE algorithm and the EDEV algorithm [36]. The comparison is conducted under benchmarks f_{a_1}–$f_{a_{28}}$ of our test suite on 10D, 30D and 50D respectively, and these results are given in Table 7. It can be seen from the results that our TDE algorithm obtains overwhelming better performance in comparison with the EDEV algorithm from optimization accuracy view as well.

5.2. Convergence speed

The TDE algorithm is also examined from convergence speed perspective of view. The convergence curves of all algorithms including SHADE, LSHADE, iLSHADE, jSO, LPalmDE, HARD-DE and our TDE are plotted according to the median of 51 runs of each algorithm on each benchmark, and they are shown in Fig. 3–5-Fig. 6 respectively. In comparison with SHADE algorithm, the proposed TDE algorithm performs better on $f_{a_1}-f_{a_4}, f_{a_6}-f_{a_{10}}, f_{a_{12}}, f_{a_{13}}, f_{a_{15}}, f_{a_{16}}, f_{a_{18}}, f_{a_{20}}, f_{a_{23}}-f_{a_{25}}$ and $f_{a_{27}}$, performs similar on $f_{a_5}, f_{a_{11}}, f_{a_{17}}, f_{a_{19}}, f_{a_{21}}, f_{a_{22}}, f_{a_{26}}$ and $f_{a_{28}}$; it also performs better on $f_{a_1}-f_{a_4}, f_{a_6}-f_{a_{10}}, f_{a_{14}}, f_{a_{16}}, f_{a_{20}}, f_{a_{25}}$ and performs similar on $f_{a_5}, f_{a_{11}}, f_{a_{13}}, f_{a_{17}}-f_{a_{19}}, f_{a_{21}}, f_{a_{22}}, f_{a_{24}}$ and $f_{a_{26}}-f_{a_{28}}$ in comparison with LSHADE; It also performs better on $f_{a_1}-f_{a_3}, f_{a_6}-f_{a_8}, f_{a_{10}}, f_{a_{13}}, f_{a_{14}}, f_{a_{16}}, f_{a_{20}}, f_{a_{22}}, f_{a_{25}}$ and performs similar on $f_{a_4}, f_{a_5}, f_{a_{11}}, f_{a_{17}}, f_{a_{21}}, f_{a_{24}}$, and $f_{a_{26}}-f_{a_{28}}$ in comparison with iLSHADE; it also performs better on $f_{a_1}-f_{a_4}, f_{a_6}-f_{a_8}, f_{a_{10}}, f_{a_{12}}-f_{a_{14}}, f_{a_{16}}, f_{a_{18}}-f_{a_{20}}, f_{a_{25}}$ and performs similar on $f_{a_5}, f_{a_{11}}, f_{a_{17}}, f_{a_{21}}, f_{a_{22}}, f_{a_{24}}$ and $f_{a_{26}}-f_{a_{28}}$ in comparison with jSO algorithm; performs better on $f_{a_2}, f_{a_4}, f_{a_6}-f_{a_8}, f_{a_{10}}-f_{a_{13}}, f_{a_{16}}, f_{a_{20}}$ and performs similar on $f_{a_4}, f_{a_5}, f_{a_{14}}, f_{a_{17}}, f_{a_{19}}$ and $f_{a_{21}}-f_{a_{28}}$ in comparison with LPalmDE; performs better on $f_{a_1}-f_{a_3}, f_{a_6}, f_{a_8}, f_{a_9}, f_{a_{11}}-f_{a_{13}}, f_{a_{16}}, f_{a_{18}}, f_{a_{20}}, f_{a_{23}}$ and performs similar on $f_{a_4}, f_{a_5}, f_{a_{14}}, f_{a_{17}}, f_{a_{19}}, f_{a_{21}}, f_{a_{22}}$ and $f_{a_{24}}-f_{a_{28}}$ in comparison with HARD-DE. Therefore, the proposed TDE algorithm is also competitive with other well-known DE variants in terms of convergence speed.

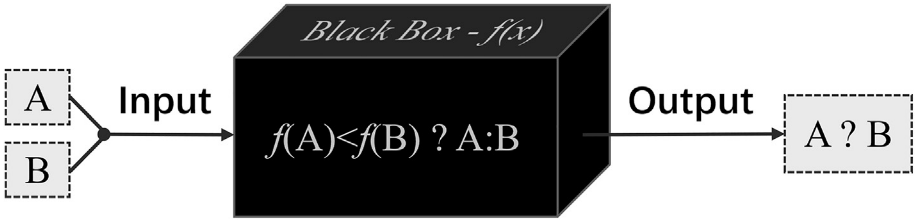


Fig. 7. The black-box system without any fitness-based outputs. The figure herein depicts an unit of the system in which input A and B are two vectors. The unit outputs the better vector A (or B) under the objective $f(X)$.

5.3. Examination the parameters and reduction schemes

In this part, we also examine the parameter settings of τ, ρ in our TDE algorithm. τ denotes the time when to record the seeds and ρ is the ratio of the earlier part of evolution employing the historical-population based mutation strategy to the whole evolution. We examine six cases of the τ settings, $\tau = 0.3, \tau = 0.4, \tau = 0.5, \tau = 0.6, \tau = 0.7, \tau = 0.8$, and 5 cases of the ρ settings, $\rho = 0, \rho = 1/3, \rho = 1/2, \rho = 2/3, \rho = 1$. The comparison results of these comparisons are given in Table 8 and Table 9 respectively, and we can see that $\tau \in [0.55, 0.65]$ and $\rho = 2/3$ are the good choices.

Besides the default platform-based reduction scheme, we also examine the optimization performance under the commonly used linear population size reduction scheme [32]. The comparison results of these two schemes are given below in Table 10, and we can see that our scheme is still competitive with the linear reduction scheme.

6. Conclusion

DE algorithms receive close attention from both researchers and engineers because of its easy implementation and excellent performance. The overall performance of a certain DE algorithm is usually affected by the mutation strategy and its corresponding parameter control. Generally, different mutation strategies have different characteristics. In our paper, we propose a two-stage DE algorithm with novel parameter control. Our algorithm employs a mutation strategy with better perception of the landscape of the objective in the earlier stage of the evolution and it employs a mutation strategy with better diversity of trial vector candidates and convergence speed in the later part of the evolution. The experiment results under a test suite containing 88 benchmarks reveal the competitiveness of our TDE algorithm.

Moreover, fitness-value independent parameter control is also proposed in our paper, and this parameter control can be able to tackle the optimization cases that the exact fitness values of the objectives are unavailable. Here we present a general model of this optimization, see Fig. 7, we can see that the output contains no exact fitness values of the black-box system, and the output is the better one of the inputs A and B. In these optimization cases with unavailable fitness values, many fitness-difference based state-of-the-art DE variants, e.g. LSHADE, iLSHADE and jSO, are unable to work properly. This is also one of the main highlights of our TDE algorithm.

A platform-based population size reduction scheme is also proposed in this paper, and it is also verified to be very effective in comparison with the linear reduction scheme. Furthermore, the algorithm validation under the large test suite containing 88 benchmarks from CEC2013, CEC2014 and CEC2017 test suites for real-parameter single-objective optimization also avoids over-fitting problem in comparison with algorithm validation under a small number of benchmarks. As a result, our TDE algorithm is very competitive with these state-of-the-art DE variants.

All the DE variants in our paper employ the binomial crossover as the majority of DE researchers believe that DE variants with binomial crossover are good at tackling real-parameter optimization while DE variants with exponential crossover are good at tackling optimization problems in which there are some linkages among the neighboring variables/parameters, however, we find that both the exponential crossover and the binomial crossover can obtain equivalent performance in optimization applications. In order words, we can develop a DE variant with exponential crossover which can also beat the winner DE variants (LSHADE and jSO) in competitions on single-objective real-parameter optimization. Generally, the proper *CR* values in exponential crossover are in a much narrower range than the *CR* values in binomial crossover when obtaining the equivalent performance. Therefore, finding these proper *CR* values are very difficult for a DE variant employing exponential crossover let alone finding its corresponding parameter control. This issue will be tackled in our next paper in the near future.

CRediT authorship contribution statement

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

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.

Acknowledgements

This work is supported by the National Natural Science Foundation of China with the Grant No. 61906042, Natural Science Foundation of Fujian Province with the Grant No. 2021J05227 and Scientific Research Startup Foundation of Fujian University of Technology (GY-Z19013).

References

[1] 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) (Dec. 2006) 646–657.
[2] J. Brest, M.S. Maučec, B. Bošković, Single objective real-parameter optimization: algorithm jso, In: Proc. IEEE Congr. Evol. Comput. (CEC), IEEE (2017) 1311–1318.
[3] S. Das, S.S. Mullick, P.N. Suganthan, Recent advances in differential evolution–an updated survey, Swarm Evol. Comput. 27 (Apr. 2016) 1–30.
[4] S. Das, P.N. Suganthan, Differential evolution: A survey of the state-of-the-art, IEEE Trans. Evol. Comput. 15 (1) (Feb. 2010) 4–31.
[5] R. Eberhart, J. Kennedy, Particle swarm optimization, Proc. IEEE Int. Conf. on Neural Netw. (ICNN) 4 (1995) 1942–1948, Citeseer.
[6] A.E. Eiben, J. Smith, From evolutionary computation to the evolution of things, Nature 521 (7553) (2015) 476–482.
[7] N. Hansen, S.D. Müller, P. Koumoutsakos, Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es), Evol. Comput. 11 (1) (2003) 1–18.
[8] J.H. Holland et al, Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence, MIT Press, 1975.
[9] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simulated annealing, Science 220 (4598) (1983) 671–680.
[10] J. Liang, B. Qu, P. Suganthan, Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization (Tech. rep.), Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China, 2014.
[11] J.J. Liang, B. Qu, P. Suganthan, A.G. Hernández-Díaz, Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization (Tech. rep.), Nanyang Technological University, Singapore, 2013.
[12] 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.
[13] Z. Meng, J.-S. Pan, A Competitive QUasi-Affine TRansformation Evolutionary (C-QUATRE) algorithm for global optimization, Proc. IEEE Int. Conf. Syst. Man Cybern. SMC (2016) 1644–1649.
[14] 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.
[15] Z. Meng, J.-S. Pan, HARD-DE: Hierarchical archive based mutation strategy with depth information of evolution for the enhancement of differential evolution on numerical optimization, IEEE Access 7 (2019) 12832–12854.
[16] Z. Meng, J.-S. Pan, Monkey king evolution: a new memetic evolutionary algorithm and its application in vehicle fuel consumption optimization, Knowl.-Based Syst. 97 (Apr 2016) 144–157.
[17] Z. Meng, J.-S. Pan, L. Kong, Parameters with adaptive learning mechanism (PALM) for the enhancement of differential evolution, Knowl.-Based Syst. 141 (2018) 92–112.
[18] 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.
[19] 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 (Oct. 2016) 104–121.
[20] Z. Meng, C. Yang, Hip-DE: Historical population based mutation strategy in differential evolution with parameter adaptive mechanism, Inf. Sci. 562 (2021) 44–77.
[21] Z. Meng, C. Yang, X. Li, Y. Chen, Di-DE: Depth information based differential evolution with adaptive parameter control for numerical optimization, IEEE Access 8 (2020) 40809–40827.
[22] Z. Meng, Y. Zhong, C. Yang, CS-DE: Cooperative strategy based differential evolution with population diversity enhancement, Inf. Sci. 577 (2021) 663–696.
[23] J. Nocedal, S. Wright, Numerical optimization, Springer Science & Business Media, 2006.
[24] Ş. Öztürk, R. Ahmad, N. Akhtar, Variants of artificial bee colony algorithm and its applications in medical image processing, Appl. Soft Comput. 106799 (2020).
[25] S. Ozturk, Y. Unal, A two-stage whale optimization method for classification of parkinson's disease voice recordings, Int. J. Intell. Syst. Appl. Eng. 8 (2) (2020) 84–93.
[26] Ş. Öztürk, E. Yiğit, U. Özkaya, Fused deep features based classification framework for covid-19 classification with optimized mlp, Konya Mühendislik Bilimleri Dergisi 8 (2021) 15–27.
[27] J.-S. Pan, Z. Meng, S.-C. Chu, H.-R. Xu, Monkey king evolution: an enhanced ebb-tide-fish algorithm for global optimization and its application in vehicle navigation under wireless sensor network environment, Telecommun. Syst. 65 (3) (Jul. 2017) 351–364.
[28] F. Peng, K. Tang, G. Chen, X. Yao, Multi-start JADE with knowledge transfer for numerical optimization, in: Proc. IEEE Congr. Evol. Comput. (CEC), IEEE, 2009, pp. 1889–1895..
[29] K. Price, R.M. Storn, J.A. Lampinen, Differential evolution: a practical approach to global optimization, Springer, Berlin, Heidelberg, 2006.
[30] R. Storn, K. Price, Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces. Tech. rep., Int. Comput. Sci. Inst., Berkeley, CA, USA, 1995..
[31] R. Storn, K. Price, Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces, J. Global Optim. 11 (4) (Dec. 1997) 341–359.
[32] R. Tanabe, A.S. Fukunaga, Improving the search performance of shade using linear population size reduction, in: Proc. IEEE Congr. Evol. Comput. (CEC), IEEE, 2014, pp. 1658–1665..
[33] A. Viktorin, R. Senkerik, M. Pluhacek, T. Kadavy, A. Zamuda, Distance based parameter adaptation for success-history based differential evolution, Swarm Evol. Comput. 50 (Nov. 2019) 100462.
[34] G. Wu, R. Mallipeddi, P. Suganthan, Problem definitions and evaluation criteria for the cec 2017 competition on constrained real-parameter optimization (Tech. Rep.), Nanyang Technol. Univ., Singapore, 2017.
[35] G. Wu, R. Mallipeddi, P.N. Suganthan, Ensemble strategies for population-based optimization algorithms–a survey, Swarm Evol. Comput. 44 (2019) 695–711.
[36] G. Wu, X. Shen, H. Li, H. Chen, A. Lin, P.N. Suganthan, Ensemble of differential evolution variants, Inf. Sci. 423 (2018) 172–186.
[37] M. Yang, C. Li, Z. Cai, J. Guan, Differential evolution with auto-enhanced population diversity, IEEE Trans. Cybern. 45 (2) (2014) 302–315.
[38] J. Zhang, A.C. Sanderson, JADE: adaptive differential evolution with optional external archive, IEEE Trans. Evol. Comput. 13 (5) (2009) 945–958.