Applied Soft Computing 146 (2023) 110750

Contents lists available at ScienceDirect

Applied Soft Computing

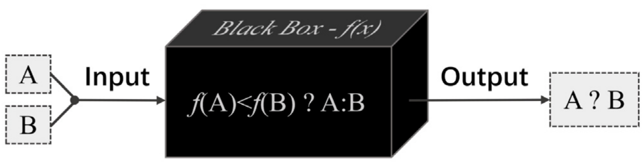
journal homepage: www.elsevier.com/locate/asoc

Differential Evolution with exponential crossover can be also competitive on numerical optimization

Zhenyu Meng [a,*], Yuxin Chen [a]

[a] Institute of Artificial Intelligence, Fujian University of Technology, Fuzhou, China

GRAPHICAL ABSTRACT



A black-box model reflecting the complex real-world optimization with fitness value unavailable.

ARTICLE INFO

Article history:
Received 27 October 2022
Received in revised form 23 July 2023
Accepted 5 August 2023
Available online 16 August 2023

Keywords:
Differential Evolution
Exponential crossover
Fitness-value-independent characteristic
Novel parameter control

ABSTRACT

Differential Evolution (DE) is a powerful population-based evolutionary algorithm for global optimization, and it is widely used in many scientific and engineering applications. There are two commonly used crossover schemes in the literature: one is binomial crossover and the other is exponential crossover. The majority of DE researchers believe that DE variants employing binomial crossover usually obtain superior performance than the ones employing exponential crossover on numerical optimization and DE variants with exponential crossover are good at tackling optimization problems with linkages among neighboring variables. On the contrary, here in this paper, a new perspective is proposed that DE variant with exponential crossover can obtain competitive performance with the ones employing binomial crossover on numerical optimization regardless of whether there are linkages among the variables or not after discovering the proper crossover rate *Cr* and its corresponding parameter control, and the main contributions of the paper can be summarized as follows: (1) The first powerful DE variant with exponential crossover, namely the DE-EXP algorithm, which is superior to the recent winner DE variants, e.g. LSHADE, iLSHADE and jSO, in Congress on Evolutionary Computation (CEC) competitions, is developed for numerical optimization; (2) A novel parameter control of crossover rate *Cr* is developed for exponential crossover and the value of *Cr* can be automatically generated not only in the initialization stage but also during the evolution. (3) A black-box model illustrating the fitness-value-dependency weakness of the recent winner DE variants in CEC competitions is given and a novel fitness-value-independent adaptation scheme for scale factor *F* is proposed in the DE-EXP algorithm to overcome the fitness-value-dependency weakness. A larger test suite containing 88 benchmarks is used for the validation of our DE-EXP algorithm, and experiment results show its superiority in comparison with several state-of-the-art DE variants.

1. Introduction

Differential Evolution (DE) was a population-based powerful evolutionary algorithm proposed by Storn and Price in 1995 [1]. As it originated with the Genetic Annealing Algorithm [2], a hybrid Genetic Algorithm (GA) and Simulated Annealing (SA), DE also employed the same bio-inspired operations such as mutation, crossover and selection as GA in the simulation of the evolution process of organisms [3,4]. Biologically, mutation is the slightly change of a chromosome, however, in the context of DE, it is the generation of mutant vector by using the information of the population [5–7]. Chromosomal crossover is the swapping of

* 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.2023.110750
1568-4946/

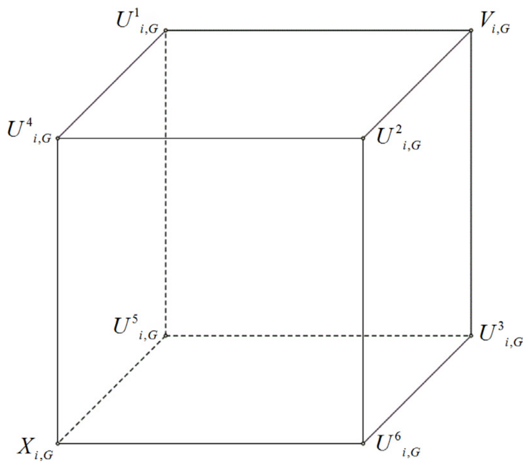


Fig. 1. The spatial search of DE variants in a 3-D view.

genetic information in the germ-line, and DE extends the operation to create trial vector during evolution. The selection operation which means that only the superior individuals between the parents and their offsprings were survived during the evolution mimicked the Darwin's Principle "survival of the fittest" [8–10] and it also reveals that DE is in the field of Evolutionary Computation (EC).

From the spatial search perspective, DE can be interpreted in another way. Mutation, to be more exactly, the mutant vector determines the search range of the target individual in each step/generation while crossover determines how to select the trial vector candidates from the range, then selection determines whether the target individual moves from the current location to the location of the trial vector or not regarding to the performance both of them [11–14]. Fig. 1 depicts the spatial search of DE from a 3-D view in which $X_{i,G}$ represents the target vector (current position) of the ith individual and $V_{i,G}$ denotes the mutant vector of the individual in the Gth generation. Obviously, the search range of the ith individual in each generation of DE is a hyper-cube, and the line segment $\overline{X_{i,G}V_{i,G}}$, also the body diagonal of the hyper-cube, determines its volume. The crossover operation decides how to choose a trial vector from all the trial vector candidates in the hyper-cube. It needs to be emphasized that the commonly used exponential crossover and binomial crossover actually implement the same effect, selecting a certain vertex from the hyper-cube. From this perspective of view, the binomial crossover and exponential crossover can be transformed into each other after finding the proper crossover rate Cr for a certain number of generations.

For a mathematical understanding of the same effect of binomial crossover and exponential crossover for several generations, some definitions firstly are presented. Here $U_{i,G}^{N=k}$ denotes a certain vector in the set of trial vectors that there are k parameters inherited from the mutant vector $V_{i,G}$, and $P(U_{i,G}^{N=k})$ denotes the probability of generating such a trial vector candidate with k parameters inherited from mutant vector $V_{i,G}$, then, the probability $P(U_{i,G}^{N=k})$ can be calculated according to Eqs. (1) and (2) by employing exponential crossover and binomial crossover respectively under a certain crossover rate Cr for a D-dimensional optimization:

$$P(U_{i,G}^{N=k}) = \begin{cases} Cr^{k-1} \times (1-Cr), \ \ if \ \ k < D \\ Cr^{D-1}, \ \ if \ \ k = D \end{cases} \tag{1}$$

$$\begin{cases} (D-1)! = \prod_{i=1}^{D-1} i \\ C_{D-1}^{k-1} = \dfrac{(D-1)!}{(k-1)! \times (D-k)!} \\ P(U_{i,G}^{N=k}) = C_{D-1}^{k-1} \times Cr^{k-1} \times (1-Cr)^{D-k} \end{cases} \tag{2}$$

Then the expectation that N parameters in the trial vector are inherited from the mutant vector can be calculated by Eqs. (3) and (4) respectively by employing exponential crossover and binomial crossover:

$$E(N) = \frac{1-Cr^D}{1-Cr} \tag{3}$$

$$E(N) = (D-1) \times Cr + 1 \tag{4}$$

Obviously, given a binomial crossover rate Cr_b, there always exits an exponential crossover rate Cr_e, $Cr_e \in [0, 1]$, satisfying Eq. (5) and vice versa:

$$\frac{1-{Cr_e}^D}{1-Cr_e} = (D-1) \times Cr_b + 1 \tag{5}$$

From this perspective of view, DE variant employing exponential crossover can be also competitive on numerical optimization after discovering the proper crossover rate Cr and its corresponding parameter control. However, researchers in the DE community still did not discover such a competitive DE variant with exponential crossover for numerical optimization though some researchers discussed some characteristics of exponential crossover [15–18]. Currently, the majority of DE researchers believe that DE variants with exponential crossover are good at tackling optimization problems with linkages among neighboring variables [19–22] and DE algorithms with exponential crossover usually perform worse than the ones employing binomial crossover on the majority of optimization applications [23–26].

Here in this paper, a new perspective is presented that DE variant with exponential crossover can obtain competitive performance with the ones employing binomial crossover on numerical optimization regardless of whether there are linkages among the variables or not after discovering the proper crossover rate Cr and its corresponding parameter control. Therefore, the main contributions can be summarized as follows:

- The first DE variant with exponential crossover, namely the DE-EXP algorithm, which is superior to the winner DE variants, e.g. LSHADE, iLSHADE and jSO, in CEC competitions is developed for numerical optimization;
- A novel generation strategy of the crossover rate CR is developed in the DE-EXP algorithm, where CR is automatically calculated not only in the evolution stage but also in the initialization.
- A novel fitness-value-independent parameter control of scale factor F is developed in the DE-EXP algorithm, therefore, the algorithm does not need the exact fitness value during the evolution and can be applied in much wider optimization scenarios, especially for those that the fitness values are unavailable (see Fig. 2).

To summarize, the aim of the paper is to present a new perspective that DE variant with exponential crossover can be also competitive on numerical optimization, and the first powerful DE variant with exponential crossover, the DE-EXP algorithm, which is superior to the recent winner DE variants, is given in the paper to verify our assertion. Besides the superior performance to the latest state-of-the-art DE variants, our DE-EXP algorithm can be applied in a much wider optimization scenario because of its fitness-independency characteristic. The remainder of the

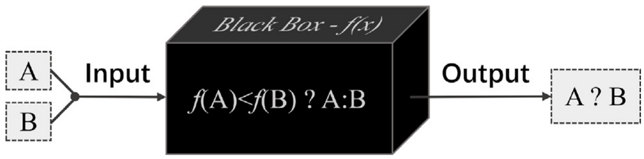


Fig. 2. A black-box model reflecting the complex real-world optimization with fitness value unavailable.

paper is organized as follows: Section 2 presents the literature review of DE with exponential crossover and the novel black-box model with unavailable fitness value. Section 3 introduces the details of our novel DE-EXP algorithm. Section 4 shows the experiment analysis and validation of our algorithm in comparison with several state-of-the-art DE variants in recent numerical competitions. Finally, the conclusion is given in Section 5.

2. Literature review

In this part, a brief introduction of DE with exponential crossover for black-box optimization is presented, and the whole part can be separated into the following two subsections, one is the DE algorithm with exponential crossover, and the other is the discussion of the novel black-box model.

2.1. DE algorithm with exponential crossover

The whole evolution of any DE variant for the tackling of optimization problems can be separated into two stages, the initialization stage and the iteration stage of mutation, crossover and selection before termination.

(A) Initialization:

In any DE algorithm, there are *PS* individuals maintained in the population during evolution. Uniform distribution is employed in scattering the individuals into the solution space Ω, where the lower $X^D_{\min}$ and upper $X^D_{\max}$ boundaries delimit the *D*-dimensional solution space. Here the initialization of the *j*th dimension of the *i*th individual in the initialization stage (the 0th generation) is taken for example, and it can be initialized according to Eq. (6):

$$x_{ij,0} = x_{\min,j} + rand_{ij}(0, 1) \cdot (x_{\max,j} - x_{\min,j}) \tag{6}$$

where $rand_{i,j}(0, 1)$ denotes a random value generated according to uniform distribution of the range (0, 1) during the calculation of $x_{ij,0}$. Obviously, all the left components of the individuals can be initialized like this as well.

After all individuals are generated, evaluations of the individuals in the population are conducted by function calls, and the best individual in the population is labeled as the global best solution. Then the algorithm enters into the evolution stage which contains operations of mutation, crossover and selection before termination.

(B) Evolution:

There are three basic operations including mutation, crossover and selection in the evolution stage of DE. Mutation operation aims to generate mutation vectors. Here the mutation strategy in the canonical DE algorithm is taken for example, and the mutation strategy is given in Eq. (7):

$$V_{i,G} = X_{r_0,G} + F \cdot (X_{r_1,G} - X_{r_2,G}) \tag{7}$$

where $X_{r_0,G}$, $X_{r_1,G}$ and $X_{r_2,G}$ denote three randomly selected vectors/individuals from the population under random selection with restriction, $r_0 \neq r_1 \neq r_2$ [2].

After the mutant vector $V_{i,G}$ is generated, the algorithm enters into crossover operation in which both the mutant vector $V_{i,G}$ and the target vector $X_{i,G}$ are two inputs of the operation. A specified crossover scheme and the corresponding crossover rate *Cr* are also involved in crossover operation, and Fig. 3 presents the process of calculating an 8-dimensional trial vector $U_{i,G}$ employing exponential crossover with a certain crossover rate *Cr*. Fig. 4 also presents a 2-D view of calculating the trial vector $U_{i,G}$ by mutation and crossover operations.

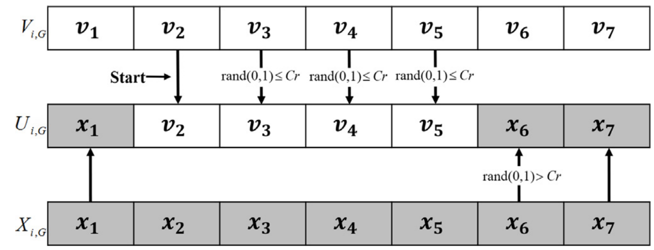


Fig. 3. The process of calculating an 8-dimensional $U_{i,G}$ with the two inputs $X_{i,G}$ and $V_{i,G}$ by employing exponential crossover under a certain *Cr*.

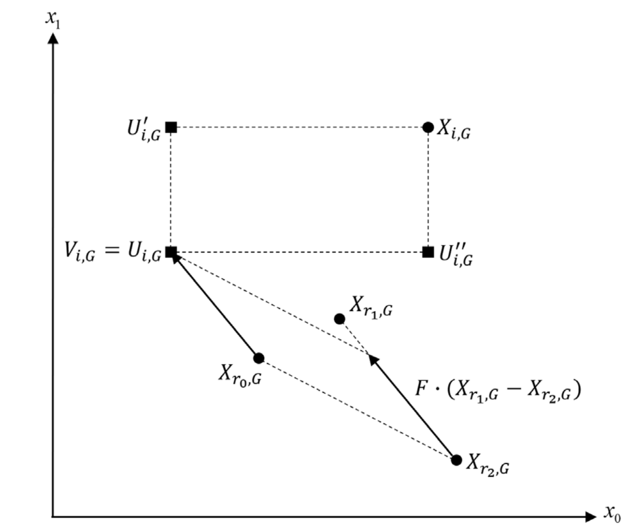


Fig. 4. A 2-D view of calculating the trial vector $U_{i,G}$ by mutation and crossover operations.

After the trial vectors are generated, the algorithm goes into the final operation, the selection operation, in each generation. Actually, the selection operation is to choose between the target vector $X_{i,G}$ and the trial vector $U_{i,G}$, and this operation can be explained via Eq. (8).

$$X_{i,G+1} = \begin{cases} U_{i,G}, & if\ U_{i,G} \text{ is better than } X_{i,G}. \\ X_{i,G}, & \text{otherwise} \end{cases} \tag{8}$$

Please note that selection can be made if and only if it can be judged which the better one is, between $X_{i,G}$ and $U_{i,G}$, and the exact fitness value is not the necessary condition though it is easy to judge which the better one is according to their exact fitness values. An explanation will be discussed further from a novel perspective of the black-box model in the next subsection.

2.2. A novel perspective of black-box model

Generally, a black box in computer science and engineering is a system that can be viewed in terms of its inputs and outputs without knowing any of its internal knowledge. Evolutionary Computation presents approaches to solving black-box optimization which considers the design and analysis of algorithms for problems in which the structure of the objective or the constraint defining the set is unknown, un-exploitable, or non-existent [27].

Obviously, the black-box model with an exact fitness output is just a special case of black-box optimization, and any black-box model with exact output can be converted into a more complex model with unavailable fitness values by adding a shell outside of it. Fig. 2 presents such a complex black-box model with unavailable fitness value, and it reflects the optimization situations (1) that the systems are too complex to model their exact fitness values and (2) that the exact fitness values of the models are deliberately hidden.

In DE, whether the trial vector $U_{i,G}$ is better than the target vector $X_{i,G}$ or not is usually judged by their fitness values, and the output of the black-box model in CEC competitions or Black-Box Optimization Benchmarking (BBOB) platform is exact fitness value as well. Obviously, it is easy to judge which the better one is between $U_{i,G}$ and $X_{i,G}$ according to their exact fitness values, however, the exact fitness values of $X_{i,G}$ and $U_{i,G}$ are not the necessary conditions for the judgment because from the mathematical view if some specific relations between $X_{i,G}$ and $U_{i,G}$ are known, for example, if $f(U_{i,G}) \leq f(X_{i,G})$ or $\frac{f(U_{i,G})}{f(X_{i,G})} < 1$ and $f(X_{i,G}) > 0$ in a minimum optimization, then it can be known which one is the better. In other words, the exact fitness values are not necessary in these cases when comparing $X_{i,G}$ and $U_{i,G}$. Actually, there are many cases in the real-world optimization demands that the exact fitness values of the objectives are unavailable [27,28], for example, the system is too complex to output the exact fitness and only a fuzzy cognitive which input is better can be known, or, the exact fitness value is visible from the inner of the black-box but deliberately hidden from its outside as the terminology "black-box" system means it is invisible from the outside, not the inside.

The novel black-box model reflecting such a complex real-world optimization problem without available fitness value has already been given in Fig. 2, and from the model, the choice can be made between the two inputs A and B even though the exact fitness values of A and B are unavailable (invisible from the outer of the model). As is known to all, the DE variants proposed in the earlier age, such as the canonical DE algorithm [29], the jDE algorithm [30] and the JADE algorithm [31] etc, can tackle this optimization model, however, they lost in recent competitions. On the contrary, all the excellent DE variants in recent CEC competitions employed fitness improvements (the exact fitness difference) in the adaptation of control parameters, obviously, they cannot tackle our optimization model. Our algorithm which will be described in the next section not only can tackle the optimization model in Fig. 2 but also can obtain competitive performance in comparison with these winner DE variants in recent competitions. This is also one of the contributions of the novel DE-EXP algorithm in the paper.

3. The novel DE-EXP algorithm

In this section, the details of the novel DE algorithm with exponential crossover is presented, and the whole description can be divided into four parts. In the first part, a novel parameter control of crossover rate *CR* is presented in the novel DE algorithm with exponential crossover. The highlight is that the exact value of *Cr* can be automatically generated during the whole generation including the initialization stage, and there is even no need to set its initial value. In the second part, the parameter control of scale factor *F* is given, and instead of employing fitness-improvement-based parameter control, a novel fitness-value-independent approach is proposed. In the third part, the adaptation scheme of population size is also given, and the last presents how to generate the trial vectors in our DE-EXP algorithm with exponential crossover.

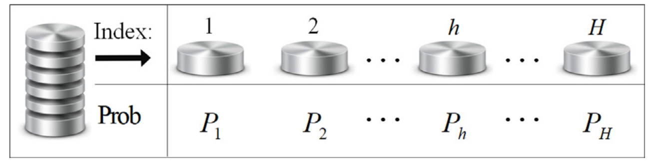


Fig. 5. The H-memory pool recording vector of probabilities P_h in our DE-EXP algorithm.

Table 1
Number of success ("s") and failure ("f") individuals in each group.

	1	2	···	k	···	D
f	nf_1	nf_2	···	nf_k	···	nf_D
s	ns_1	ns_2	···	ns_k	···	ns_D
v	v_1	v_2	···	v_k	···	v_D

3.1. The automatically generated crossover rate Cr

The automatically generated crossover rate *Cr* is also one of the main contributions in our paper because all the *Cr* values are automatically generated not only in the initialization stage but also during the evolution before termination. In order to calculate *Cr* values of the individuals, the thought of calculating "evolution matrix" M in our former proposed QUATRE algorithms [8,9,32] is incorporated into our DE-EXP algorithm. In the QUATRE algorithms, the evolution matrix M was developed for the individuals' evolution, and the key to the generation of M is the calculation of $N(k)$, which denotes the total number of vectors in which there are k parameters/components retained from the mutant vector. Actually, the number of parameters/components, k, in the trial vector that were retained from the mutant vector is determined by crossover rate *Cr* in DE, therefore, the expectation of k in the evolution matrix M can be taken as a statistic of *Cr*, and the calculation of *Cr* is presented in Eq. (9):

$$\begin{cases} k = P_i \cdot (1, 2, 3, \ldots, D)^T \\ c = \min(0.5 \cdot D, 10) \\ E_m = c \cdot k \\ Cr_i = \dfrac{E_m}{E_m + 1} \end{cases} \quad (9)$$

where P_i denotes the vector of probabilities that the parameters (or dimensions) in the mutant vector inherit into the ith trial vector and the value of P_i is randomly selected from an H-memory pool in which H vectors of probabilities are recorded. Fig. 5 presents the vector of probabilities in the H-memory pool, and the initial values of vectors are the same, $P_1 = P_2 = \cdots = P_h = \cdots = P_H = \frac{1}{D} \cdot (1, 1, 1, \ldots, 1)$. The parameter k has the same meaning as the one in QUATRE algorithms, and it means that there are k parameters in total in the mutant vector inherited into the trial vector.

During the evolution, both the success and failure individuals of the population can be categorized into D groups regarding how many parameters in the mutant vector are inherited into the trial vectors, and the set of success individuals can be further divided into D subsets, $\boldsymbol{S} = \sum_{k=1}^{D} \boldsymbol{S}_k$. Then the number of success individuals and failure individuals in the kth group can be denoted by ns_k and nf_k respectively, and the summarization of the weights (calculated according to Eq. (13)) of the kth group are denoted by v_k. Table 1 presents the summarization of these variables, and the vector of probabilities can be renewed according to Eq. (10). The update is conducted in a circle, from the first P_1 to the last P_H and then back to the first, and only one vector is renewed in a generation. An adjustment of P_h is also set, if and only if

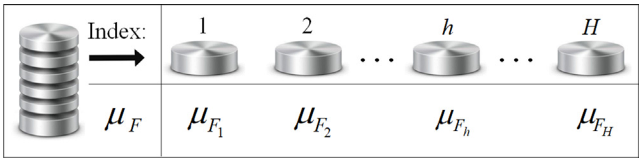


Fig. 6. The H-memory pool recording μ_F in our DE-EXP algorithm.

ns_1 equals to $\sum_{k=1}^{D} ns_k$, P_h is renewed by the D-dimensional zero vector, $P_h = (0, 0, \ldots, 0)$.

$$\begin{cases} r_k = \dfrac{ns_k}{ns} \\ v_k = \dfrac{\sum_{i\in \boldsymbol{S}_k} w_i}{ns_k} \\ P_h = \begin{cases} \dfrac{r_k \cdot v_k}{\sum_{k=1}^{D} v_k} + \min(1/PS, 1/D), & if\ ns_k > 0, \\ \min(1/PS, 1/D), & otherwise. \end{cases} \end{cases} \tag{10}$$

3.2. The fitness-value-independent adaptation of scale factor F

In our DE-EXP algorithm, the scale factor F of each individual in the population obeys semi-fixed Cauchy distribution, $F \sim C(\mu_F, \sigma_F)$, in which μ_F and σ_F denote the location parameter and scale parameter of the distribution respectively. The distribution is semi-fixed, because μ_F is adaptively changed while σ_F is fixed constant, $\sigma_F = 0.05$, during the whole evolution. There is an H-memory pool maintained during the evolution, and each memory in the pool records a unique location parameter μ_F. Each individual in the population randomly selects a memory and employs the corresponding μ_F in generating its scale factor. If the generated F value is out of the default range (0, 1], it should be adjusted according to Eq. (11):

$$F = \begin{cases} randc(\mu_F, 0.05), & \text{while } F \le 0 \\ 1, & \text{if } F > 1 \\ F, & \text{otherwise} \end{cases} \tag{11}$$

The initial values of the location parameters in the H-memory pool are the same, $\mu_{F_1} = \mu_{F_2} = \cdots = \mu_{F_h} = \mu_{F_H} = 0.5$, at the beginning of the evolution. The scale factors that generate good trial vectors in each generation are recorded in the success set S_F, and these success values of F are used for updating μ_F of the H-memory pool. The update is conducted in a circle, from the 1st to the last and then back to the first, and only one memory is updated in a generation. Fig. 6 illustrates the H-memory pool in the DE-EXP algorithm.

Different from the recent winner DE variants that employed fitness-difference (fitness improvement) in the adaptation of control parameters, the adaptation of scale factor F in our DE-EXP algorithm is of fitness-value-independency characteristic. The main idea is that the good changes on the dimensions of the movement, from $X_{i,G}$ to $U_{i,G}$, can be taken as weights in the adaptation of control parameters. In order to quantize these good changes, the volatility index of the ith individual in the Gth generation is proposed and given below in Eq. (12):

$$\begin{cases} VIX_{i,G} = \sqrt{\dfrac{1}{D-1}\sum_{d=1}^{D}(XU_{i,G}(d) - \overline{XU_{i,G}})^2} \\ \overline{XU_{i,G}} = \sum_{d=1}^{D} XU_{i,G}(d) \end{cases} \tag{12}$$

where D denotes the number of dimensions of the solution space, $XU_{i,G}$ denotes the good changes on the dimensions of the movement from $X_{i,G}$ to $U_{i,G}$, and it can be calculated by the projection of the movement under the corresponding crossover rate Cr of the ith individual, $(U_{i,G} - X_{i,G}) \xmapsto{Cr} XU_{i,G}$. An example is given here for the explanation of the projection: if $U_{i,G} - X_{i,G}$ equals to (a, b, c) and the crossover vector generated under Cr equals to $(1, 0, 1)$, then the projection actually implements a component-wise multiplication, therefore, $XU_{i,G}$ equals to $(a, 0, c)$. The adaptation of scale factor F in each generation is given in Eq. (13):

$$\begin{cases} w_i = \dfrac{VIX_{i,G}}{\sum_{i\in \boldsymbol{S}} VIX_{i,G}} \\ mean_{WL}(\boldsymbol{S}_F) = \dfrac{\sum_{i\in \boldsymbol{S}} w_i \cdot F_i^2}{\sum_{i\in \boldsymbol{S}} w_i \cdot F_i} \\ \mu_F = \begin{cases} mean_{WL}(\boldsymbol{S}_F), & if\ S \ne \emptyset \\ \mu_F. & otherwise \end{cases} \end{cases} \tag{13}$$

where $\boldsymbol{S}$ denotes the set of successful individuals, $\boldsymbol{S}_F$ denotes the set of corresponding scale factors of $\boldsymbol{S}$.

3.3. Population size reduction scheme

The linear population size reduction scheme proposed in LSHADE algorithm [33] was incorporated into our DE-EXP algorithm because of its excellent performance in the recent proposed state-of-the-art DE variants [6,34–38]. The main thought of the population size reduction scheme is that relative large population size is maintained at the initialization stage aiming at getting the better perception of the landscape of the objective, and then the gradually reduced population size PS extends the total number of generations under the fixed maximum number of function evaluations. This scheme actually makes a balance between exploration at the earlier part of the evolution and exploitation at the later part of the evolution. The details of the population size reduction scheme is given below in Eq. (14):

$$PS_{G+1} = round\left[\frac{PS_{\min} - PS_{init}}{nfe_{\max}} \cdot nfe + PS_{init}\right] \tag{14}$$

where $round[\cdot]$ denotes rounding to the nearest integer, $PS_{\min}$ denotes the minimum population size during the whole evolution while PS_{init} denotes the initial population size, $nfe_{\max}$ denotes the maximum number of function evaluations and nfe denotes the current number of function evaluations.

By the way, the initialization of the population is considered as the first generation of the evolution and the reduction of population size starts at the end of the second generation. After the reduction of population size, the exceeded individuals with worse performance are removed from the population, meanwhile, the external archive that records the inferior solutions during the evolution should be also adjusted according to the new population size because the archive size $|A|$ always equals to $r^{arc} \cdot PS$ during the evolution, $|A| = r^{arc} \cdot PS$, where r^{arc} denotes the ratio of the external archive size to the population size and r^{arc} is a fixed constant during the evolution.

3.4. Trial vector generation strategy

The trial vector generation strategy "DE/target-to-pbest/1/bin" with external archive proposed in JADE [31] showed excellent performance in the CEC competitions and the recently proposed state-of-the-art DE variants employed the same or a similar strategy. The thought of this strategy mainly focused on two aspects: one is the incorporation of some elites rather than the global best individual in the generation of trial vectors, which helps to avoid premature convergence; the other is the incorporation of the inferior individuals during the evolution for the enhancement of the population diversity.

In our DE-EXP algorithm, a novel trial vector generation strategy "DE/target-to-$p$best/1/exp" with external archive is employed, and this strategy is the same as the one in JADE except for the different crossover scheme. The mutation strategy is given below in Eq. (15):

$$V_{i,G} = X_{i,G} + F_{i,G} \cdot (X^{p}_{best,G} - X_{i,G}) + F_{i,G} \cdot (X_{r_1,G} - \widetilde{X}_{r_2,G}) \quad (15)$$

where $V_{i,G}$ denotes the mutant vector of the ith individual, $X_{i,G}$ is the target vector, the vector of the ith individual, $F_{i,G}$ denotes the scale factor of the ith individual, $X^{p}_{best,G}$ denotes a random vector selected from the top $100 \cdot p\%$ individuals of the population, $X_{r_1,G}$ is a random vector selected from the current population, $X_{r_1,G} \in \boldsymbol{P}$, and $\widetilde{X}_{r_2,G}$ is a random vector selected from the union of current population and the external archive recording inferior solutions, $\widetilde{X}_{r_2,G} \in \boldsymbol{P} \cup \boldsymbol{A}$. After the generation of mutant vector $V_{i,G}$, exponential crossover is employed in generating the trial vector $U_{i,G}$. The pseudo-code describing the DE-EXP algorithm is presented in Algorithm 1.

Algorithm 1 Pseudo code of the DE-EXP algorithm

Input: Bound constraints $[R^{D}_{\min}, R^{D}_{\max}]$, the fixed maximum number of function evaluations $nfe_{\max}$;
Output: Global best individual X_{gbest}, best fitness value $f(X_{gbest})$, consumed number of function evaluations nfe;

Initialize the population size $PS = PS_{ini}$, all individuals $X = \{X_1, X_2, \ldots, X_{ps}\}$, $H = 6$, $A = \emptyset$, $\mu_{F_j} = Cr_j = 0.5$, $p = 0.11$, $r^{arc} = 2.0$, $G = 2$, $Prob = \frac{1}{D} \cdot (1, 1, \ldots, 1)$;
while $nfe \le nfe_{\max}$ **do**
  for $i = 1$; $i \le PS$; $i++$ **do**
    Generate $X^{p}_{best,G}$, $X_{r_1,G}$ and $\widetilde{X}_{r_2,G}$;
  end for
  if $G > 2$ **then**
    Adjust the individuals of the population;
    Adjust external archive A;
  end if
  for $i = 1$;$i \le PS$; $i++$ **do**
    Generate F and Cr according to Eq. (11) and Eq. (9) respectively;
  end for
  for $i = 1$;$i \le PS$; $i++$ **do**
    Generate $X_{r_1,G}$ and $\widetilde{X}_{r_2,G}$ and calculate the mutant vector according to Eq. (15);
    Generate trial vector $U_{i,G}$ by exponential crossover.
    Calculate fitness value $f(U_{i,G})$;
  end for
  $nfe = nfe + PS$;
  for $i = 1$; $i \le PS$; $i++$ **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
  end for
  if $S_F \ne \emptyset$ **then**
    Update μ_F according to Eq. (13);
    Update $Prob$ according to Eq. (10);
  end if
  $G++$;
  Update archive A;
  Label $X_{gbest,G}$ and the corresponding $f(X_{gbest,G})$;
  Adjust population size according to Eq. (14);
end while
$f(X_{gbest}) = f(X_{gbest,G})$, $X_{gbest} = X_{gbest,G}$;
return $f(X_{gbest})$, X_{gbest} and nfe;

4. Experiment analysis

The performance prediction of an evolutionary algorithm is tough because of the lack of related theory, therefore, optimization test suites with different benchmarks are proposed in this field for algorithm validation. Generally, algorithm validation on a test suite containing a smaller number of benchmarks usually has over-fitting problem. Here in the paper, a large test suite containing 88 benchmarks selected from CEC2013, CEC2014 and CEC2017 test suits for real-parameter numerical optimization is employed in the comparison of the DE variants. The benchmarks in our test suite selected from CEC2013 are labeled as f_{a_1}–$f_{a_{28}}$, the benchmarks from CEC2014 test suite are labeled as f_{b_1}–$f_{b_{30}}$ and benchmarks from CEC2017 are labeled as f_{c_1}–$f_{c_{30}}$ respectively. These benchmarks can be further categorized into unimodal function group including f_{a_1}–f_{a_5}, f_{b_1}–f_{b_3} and f_{c_1}–f_{c_3}, basic multimodal function group including f_{a_6}–$f_{a_{20}}$, f_{b_4}–$f_{b_{16}}$ and f_{c_4}–$f_{c_{10}}$, hybrid function group including $f_{b_{17}}$–$f_{b_{22}}$ and $f_{c_{11}}$–$f_{c_{20}}$ and composition function group including $f_{a_{21}}$–$f_{a_{28}}$, $f_{b_{23}}$–$f_{b_{30}}$ and $f_{c_{21}}$–$f_{c_{30}}$. In our comparison, the fixed cost criterion for algorithm validation is used, and the maximum function calls are set to $nfe_{\max} = 10000 \cdot D$, where D is the dimension number of the solution space. 51 runs on each benchmark are conducted independently for each algorithm and the results are considered as samples that are used for statistical analysis of the performance. All these algorithms are coded on Matlab 2019a version of a PC with Intel(R) Core(TM)i5-9600k CPU @ 3.7 GHz on Windows 10 Professional Edition Operating System with 16 GB RAM.

4.1. Default parameter settings of the DE variants

In order to verify the performance of our DE-EXP algorithm, several state-of-the-art DE variants including LSHADE [33], iLSHADE [39], jSO [34], LPalmDE [6], HARD-DE [36] and DE-NPC [40] are taken into comparison with our DE-EXP algorithm. The parameter settings of these algorithms are all the default ones recommended by the authors and they are listed in Table 2. For LSHADE, the scale factor F obeys semi-fixed Cauchy distribution, $F \sim C(\mu_F, 0.1)$, and the initial value of μ_F is $\mu_F = 0.5$. Crossover rate Cr obeys semi-fixed Gaussian distribution, $Cr \sim N(\mu_{Cr}, 0.1)$, and the initial value of μ_{Cr} equals to 0.5. The population size is dynamically reduced from $PS_{init} = 18 \cdot D$ to the fixed minimum $PS_{\min} = 4$. The size of the entry pool is $H = 6$. The ratio p of top superior individuals to the whole population is 0.11 and the factor r^{arc} of the external archive is $r^{arc} = 2.6$ during the evolution. For iLSHADE algorithm, the same distribution of F and Cr as LSHADE are employed, but μ_F is initialized with a different value, $\mu_F = 0.8$. Moreover, the control parameters in the last entry are set fixed constants, $\mu_{F_H} = \mu_{Cr_H} = 0.9$, during the whole evolution. The same reduction scheme of population size is employed in iLSHADE, but with a different initial value, $PS_{init} = 12 \cdot D$. Moreover, instead of employing a fixed ratio p in LSHADE algorithm, iLSHADE employed a dynamically increased ratio with its initial value equaling 0.1 and terminal value equaling 0.2. For jSO, the distributions of F and Cr and the factor of the external archive r^{arc} are the same as LSHADE algorithm, but μ_F and μ_{Cr} are initialized by different values $\mu_F = 0.3$ and $\mu_{Cr} = 0.8$. The same reduction scheme of population size and the ratio p as the LSHADE algorithm are used in the jSO algorithm, however, the initial and the terminal values are different, $PS_{init} = 25 \cdot \ln D \cdot \sqrt{D}$, $PS_{\min} = 4$, $p_{init} = 0.25$ and $p_{\min} = p_{init}/2$. Moreover, a smaller size of the entry pool is used in jSO, $H = 5$. In LPalmDE algorithm, F and Cr also obey semi-fixed Cauchy distribution and Gaussian distribution respectively, $F \sim C(\mu_F, 0.2)$, $Cr \sim N(\mu_{Cr}, 0.1)$ and the initial values of μ_F and μ_{Cr} are all set to 0.5. The LPalmDE employed a larger number of groups, $K = 19$, and population

Table 2
Recommended parameter settings of all the DE variants in the comparison.

Algorithms	The default parameter settings of the DE variants in comparison
LSHADE	$\mu_F = 0.5$, $F \sim C(\mu_F, 0.1)$, $\mu_{Cr} = 0.5$, $Cr \sim N(\mu_{Cr}, 0.1)$, $PS = 18 \cdot D \sim 4$, $H = 6$, $p = 0.11$, $r^{arc} = 1.6$
iLSHADE	F, Cr, H, r^{arc} same as LSHADE, $\mu_F = 0.8$, $\mu_{Cr} = 0.5$, $\mu_{F_H} = \mu_{F_{Cr}} = 0.9$, $PS = 12 \cdot D \sim 4$, $p = 0.2 \sim 0.1$
jSO	F, Cr, r^{arc} same as LSHADE, $\mu_F = 0.3$, $\mu_{Cr} = 0.8$, $H = 5$, $PS = 25 \cdot \ln D \cdot \sqrt{D} \sim 4$, $p = 0.25 \sim 0.125$
LPalmDE	$F_j = 0.5$, $F_{ji} \sim C(F_j, 0.2)$, $\mu_{Cr} = 0.5$, $Cr \sim N(\mu_{Cr}, 0.1)$, $K = 19$, $PS = 23 \cdot D \sim K$, $p = 0.11$, $r^{arc} = 1.6$, $T_0 = 70$
HARD-DE	F, Cr, μ_F, μ_{Cr} same as jSO, $H = 4$, $p = 0.11$, $PS = 25 \cdot \ln D \cdot \sqrt{D} \sim 4$, $r^{arcA} = 1.0$, $r^{arcB} = 3.0$
DE-NPC	F, Cr, H same as LSHADE, $\mu_F = 0.5$, $\mu_{Cr} = 0.5$, $p = 0.11$, $PS = 25 \cdot \ln D \cdot \sqrt{D} \sim 4$, $r^{arc} = 1.4$
DE-EXP	H, F, r^{arc} same as LSHADE, $\mu_F = 0.5$, $Prob = [1/D, 1/D, \ldots, 1/D]$, $PS = 25 \cdot \ln D \cdot \sqrt{D} \sim 4$, $p = 0.11$

Table 3
Mean and standard deviation (Mean/Std) of the 51-run fitness errors $\Delta f = f - f^*$ of the seven DE variants on 10D optimization under f_{a_1}–$f_{a_{28}}$ of our test suite is given below. The overall performance behind "Mean/Std" is measure under Wilcoxon's signed rank test under the significant level $\alpha = 0.05$.

10D	LSHADE	iLSHADE	jSO	LPalmDE	HARD-DE	DE-NPC	DE-EXP
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	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}	8.3949E−003/2.3219E−002(<)	1.1193E−002/2.6209E−002(<)	6.9964E−003/2.1430E−002(≈)	**1.3992E−003/9.9920E−003**(>)	8.3950E−003/2.3219E−002(<)	1.2592E−002/2.7473E−002(<)	6.9964E−003/2.1430E−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}	2.5012E+000/4.3189E+000(<)	3.4632E+000/4.7356E+000(<)	**0/0**(>)	3.0784E+000/4.5983E+000(<)	1.9240E−001/1.3740E+000(≈)	3.8480E−001/1.9236E+000(<)	1.9240E−001/1.3740E+000
f_{a_7}	6.9600E−006/3.8588E−005(<)	**2.8340E−006/7.0866E−006**(>)	6.2705E−006/1.4260E−005(<)	1.9398E−005/8.6151E−005(<)	2.3012E−003/4.8941E−003(<)	6.3744E−006/1.4121E−005(<)	5.8045E−006/2.1079E−005
f_{a_8}	2.0359E+001/8.2796E−002(<)	2.0287E+001/8.7981E−002(<)	2.0354E+001/6.4220E−002(<)	2.0201E+001/1.4292E−001(<)	2.0216E+001/1.2233E−001(<)	2.0238E+001/1.4787E−001(<)	**2.0127E+001/1.7375E−001**
f_{a_9}	1.7256E+000/1.6187E+000(<)	4.9009E−001/6.5722E−001(<)	4.9340E−001/8.7512E−001(<)	5.3955E−001/7.8032E−001(<)	2.8243E+000/9.5285E−001(<)	2.4708E+000/1.5635E+000(<)	**3.9671E−001/8.6065E−001**
$f_{a_{10}}$	1.6911E−003/4.0781E−003(>)	3.7680E−003/6.7574E−003(>)	1.8837E−003/4.4727E−003(>)	4.3487E−004/2.1878E−003(>)	**1.4534E−004/1.0356E−003**(>)	4.1076E−003/7.9795E−003(>)	1.1186E−002/1.6584E−002
$f_{a_{11}}$	*0/0*(≈)	4.4583E−015/1.5434E−014(<)	3.9018E−002/1.9505E−001(<)	*0/0*(≈)	1.1146E−014/2.2793E−014(<)	*0/0*(≈)	*0/0*
$f_{a_{12}}$	**2.1942E+000/1.0865E+000**(>)	2.2703E+000/8.2441E−001(>)	2.4988E+000/1.0033E+000(>)	2.5947E+000/1.1616E+000(>)	2.3410E+000/9.3068E−001(>)	2.3005E+000/8.3476E−001(>)	2.6367E+000/1.3885E+000
$f_{a_{13}}$	**1.7497E+000/9.4902E−001**(>)	1.7515E+000/1.0521E+000(>)	2.2833E+000/9.5428E−001(<)	2.5157E+000/1.2820E+000(<)	2.6225E+000/1.2756E+000(<)	2.0142E+000/8.8679E−001(<)	1.8651E+000/1.1067E+000
$f_{a_{14}}$	2.3267E−002/4.8314E−002(<)	2.6440E−001/6.5369E−001(<)	1.5634E+000/3.7949E+001(<)	*1.2246E−003/8.7454E−003*(≈)	2.4492E−003/1.2244E−002(<)	1.9594E−002/3.4185E−002(<)	*1.2246E−003/8.7454E−003*
$f_{a_{15}}$	2.8687E+002/1.0716E+002(>)	**2.5196E+002/1.1010E+002**(>)	2.9162E+002/1.1968E+002(>)	3.5999E+002/1.6108E+002(<)	3.4837E+002/1.2024E+002(<)	2.8781E+002/1.1392E+002(>)	3.0576E+002/1.1694E+002
$f_{a_{16}}$	2.4396E−001/1.4032E−001(<)	6.9735E−001/2.9074E−001(<)	1.2113E+000/1.9345E−001(<)	2.0280E−001/1.5878E−001(<)	3.0161E−001/2.3051E−001(<)	2.5611E−001/1.2827E−001(<)	**1.7961E−001/1.2902E−001**
$f_{a_{17}}$	*1.0122E+001/1.3510E−014*(≈)	1.0126E+001/6.4636E−003(<)	1.0675E+001/1.1749E+000(<)	*1.0122E+001/1.5461E−014*(≈)	*1.0122E+001/1.7940E−015*(≈)	*1.0122E+001/1.5461E−014*(≈)	*1.0122E+001/1.7940E−015*
$f_{a_{18}}$	1.3580E+001/1.1681E+000(>)	**1.3353E+001/1.2861E+000**(>)	1.5859E+001/2.5582E+000(<)	1.4499E+001/1.8963E+000(<)	1.5138E+001/1.4434E+000(<)	1.3961E+001/1.1314E+000(≈)	1.3962E+001/1.0354E+000
$f_{a_{19}}$	2.3497E−001/3.4565E−003(<)	2.8698E−001/6.2216E−002(<)	3.8780E−001/1.0080E−001(<)	2.7863E−001/8.0868E−002(<)	2.3002E−001/3.9401E−002(<)	**2.2195E−001/4.1063E−002**(>)	2.2227E−001/3.9958E−002
$f_{a_{20}}$	1.7480E+000/3.7899E−001(<)	1.5464E+000/4.0109E−001(<)	1.4704E+000/3.3435E−001(<)	1.6418E+000/3.5392E−001(<)	1.7932E+000/3.9283E−001(<)	1.8349E+000/2.9815E−001(<)	**1.4266E+000/3.7328E−001**
$f_{a_{21}}$	4.0019E+002/0(≈)	4.0019E+002/0(≈)	4.0019E+002/0(≈)	4.0019E+002/0(≈)	**3.8057E+002/6.0124E+001**(>)	4.0019E+002/0(≈)	4.0019E+002/0
$f_{a_{22}}$	2.5747E+000/3.7371E+000(>)	1.6998E+001/2.6024E+001(>)	**2.0706E+001/2.2398E+001**(>)	3.9205E+000/3.9719E+000(>)	3.8700E+000/4.1789E+000(>)	2.2344E+000/3.2883E+000(>)	5.6989E+000/1.4402E+001
$f_{a_{23}}$	2.9142E+002/1.2231E+002(<)	**1.7365E+002/1.2867E+002**(>)	1.9849E+002/1.2034E+002(>)	3.2209E+002/1.7275E+002(<)	3.8483E+002/1.6357E+002(<)	2.7368E+002/1.3966E+002(≈)	2.7351E+002/1.3320E+002
$f_{a_{24}}$	*2.0000E+002/0*(≈)	*2.0000E+002/0*(≈)	*2.0000E+002/2.5019E−006*(≈)	*2.0000E+002/0*(≈)	*2.0000E+0022.0655E−004*(≈)	*2.0000E+002/1.2399E−006*(≈)	2.0000E+002/1.2326E−006
$f_{a_{25}}$	2.0000E+002/0(≈)	2.0000E+002/5.0544E+000(≈)	2.0009E+002/1.1288E−005(<)	1.9816E+002/1.3109E+001(>)	1.9632E+002/1.8401E+001(>)	**1.9629E+002/1.8548E+001**(>)	2.0000E+002/1.5931E−013
$f_{a_{26}}$	1.1997E+002/3.7441E+001(<)	1.1815E+002/3.2169E+001(<)	1.0424E+002/1.3722E+001(>)	1.0796E+002/2.0052E+001(>)	**1.0334E+002/2.2341E+000**(>)	1.1004E+002/2.6533E+001(>)	1.1546E+002/3.1709E+001
$f_{a_{27}}$	*3.0000E+002/0*(≈)	*3.0000E+002/0*(≈)	*3.0000E+002/0*(≈)	*3.0000E+002/0*(≈)	*3.0000E+002/1.2350E−005*(≈)	*3.0000E+002/0*(≈)	*3.0000E+002/0*
$f_{a_{28}}$	3.0000E+002/0(<)	3.0000E+002/0(<)	3.0000E+002/0(<)	3.0000E+002/0(<)	3.0000E+002/0(<)	3.0000E+002/0(<)	**2.9608E+002/2.8006E+001**
>/≈/<	6/10/12	8/8/12	7/8/13	6/10/12	6/8/14	7/11/10	−/−/−

size is dynamically decreased from $PS_{init} = 23 \cdot D$ to $PS_{min} = K$. The factor of the external archive r^{arc} and the ratio of top superior individuals p are all set to constant values, $r^{arc} = 1.6$ and $p = 0.11$. Moreover, time stamp T_0 which makes a balance between mutation strategy with external archive and without external archive in LPalmDE is set to a constant value, $T_0 = 70$. In HARD-DE algorithm, control parameters F and Cr obey the same distributions, μ_F, μ_{Cr} and PS_{init} are also initialized with the same values as the ones in the jSO algorithm. The setting of parameter p is the same as LSHADE, $p = 0.11$, and the number of groups in HARD-DE algorithm is set to a smaller value, $H = 4$. In DE-NPC algorithm, F and Cr obey the same distributions as LSHADE, moreover, the size of the memory pool and the values of p and r^{arc} are also the same as LSHADE. The initial population size is the same as jSO, $PS_{init} = 25 \cdot \ln D \cdot \sqrt{D}$. In the novel DE-EXP algorithm, the scale factor F obeys a novel semi-fixed Cauchy distribution, $F \sim C(\mu_F, 0.05)$, and the initial value of μ_F is set to 0.5. The vector of probabilities P_m is set to $Prob = [1/D, 1/D, \ldots, 1/D]$. Moreover, the setting of parameter p and the size of the memory pool is set to $p = 0.11$ and $H = 6$. The population size obeys a linear reduction scheme with its initial value equaling to $PS_{init} = 25 \cdot lnD \cdot \sqrt{D}$ and its terminal value equaling to $PS_{\min} = 4$.

4.2. Optimization accuracy

In this part, the optimization comparisons among the algorithms mentioned in the last subsection are given including LSHADE [33], iLSHADE [39], jSO [34], LPalmDE [6], HARD-DE [36], DE-NPC [40] and our DE-EXP algorithm on our test suite containing 88 benchmarks from CEC2013, CEC2014 and CEC2017 test suits on 10D, 30D and 50D respectively. 51 runs are conducted independently on each benchmark by a certain algorithm, and the maximum number of function evaluations, $nfe_{max} = 10000 \cdot D$, is employed in the algorithm validation. Tables 3–5 present the comparison results on 28 benchmarks from CEC2013 test suites, Tables 6–8 present the results on 30 benchmarks from CEC2014 test suite and Tables 9–11 present the results on 30 benchmarks from CEC2017 test suite on 10D, 30D and 50D optimization respectively. The mean value (Mean) and standard deviation (Std) of the fitness error $f - f^*$ of the total 51-run are calculated and presented in these tables, and symbols ">", "≈" and "<" in the parentheses behind "Mean/Std" mean "Better Performance", "Similar Performance" and "Worse Performance" respectively in comparison with our DE-EXP algorithm. The measure is conducted under Wilcoxon's signed rank test with the significant level $\alpha = 0.05$, and the tier best results in the comparison are highlighted in *ITALIC* fonts while the best result in the comparison is emphasized in **BOLDFACE** on each benchmark.

From Tables 3–11, it can be seen that our DE-EXP algorithm is competitive with these state-of-the-art DE variants on 10D, 30D and 50D optimization, for example, the DE-EXP algorithm obtains 37 performance improvements and 22 similar results in comparison with LSHADE, obtains 44 performance improvements and 20 similar results in comparison with iLSHADE, obtains 41 performance improvements and 20 similar performance in comparison with jSO, obtains 36 performance improvements and 23 similar results in comparison with LPalmDE, obtains 42 performance improvements and 21 similar results in comparison with HARD-DE, obtains 33 performance improvements and 26 similar results in comparison with DE-NPC under the 88 benchmarks of our test suite. Furthermore, the results of Tables 3–11 are also summarized into Table 12 for a better understanding of the overall performance of our DE-EXP. By comparing the results on 10D, 30D and 50D optimization, it can be seen that our DE-EXP algorithm performs increasingly better on 30D and 50D optimization in

Table 4
Mean and standard deviation (Mean/Std) of the 51-run fitness errors $\varDelta f = f - f^*$ of the seven DE variants on 30D optimization under f_{a_1}–$f_{a_{28}}$ of our test suite is given below. The overall performance behind "Mean/Std" is measure under Wilcoxon's signed rank test under the significant level $\alpha = 0.05$.

30D	LSHADE	iLSHADE	jSO	LPalmDE	HARD-DE	DE-NPC	DE-EXP
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{a_1}	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0*
f_{a_2}	2.8979E−013/2.2773E−013(<)	1.6835E−009/1.2008E−008(<)	1.0602E−011/2.9044E−011(<)	1.2711E−011/8.3087E−011(<)	2.2336E−012/2.4109E−012(<)	3.9679E−013/1.9740E−013(<)	**0/0**
f_{a_3}	8.9182E−002/4.2136E−001(<)	9.8214E−006/438902E-005(<)	1.7111E−010/8.7695E−010(>)	3.3958E−002/1.5100E−001(<)	1.3736E−007/8.3837E−007(<)	**5.8511E−011/4.1522E−010**(>)	3.4661E−009/2.4751E−008
f_{a_4}	8.0250E−014/1.09740E−013(<)	1.5604E−013/1.0655E−013(<)	6.8658E−013/5.7430E−013(<)	1.1146E−013/1.1480E−013(<)	3.2100E−013/2.0426E−013(<)	2.0508E−013/1.8890E−013(<)	**0/0**
f_{a_5}	1.1369E−013/0(≈)	1.0923E−013/2.2871E−014(>)	1.1369E−013/0(≈)	1.1369E−013/0(≈)	1.1369E−013/0(≈)	**1.0823E−013/2.2287E−014**(>)	1.1369E−013/2.2737E−014
f_{a_6}	4.0972E−012/2.3391E−0111(<)	1.5604E−013/2.1081E−013(<)	1.6589E−011/9.6257E−011(<)	2.13400E−013/1.3189E−013(<)	3.2827E−008/2.3433E−007(<)	1.5827E−013/8.5313E−014(<)	**1.4267E−013/1.0867E−013**
f_{a_7}	1.7153E−001/2.2666E−001(<)	1.7782E−002/3.8511E−002(<)	5.4777E−003/1.3985E−002(<)	1.6989E−001/2.0828E−001(<)	2.4925E−002/3.7736E−002(<)	2.1776E−002/3.0092E−002(<)	**4.8007E−003/1.1567E−002**
f_{a_8}	2.0848E+001/1.1995E−001(<)	2.0882E+001/1.0041E−001(<)	2.0946E+001/5.3232E−002(<)	2.0868E+001/1.1252E−001(<)	2.0787E+001/1.5431E−001(<)	2.0807E+001/1.3555E−001(<)	**2.0604E+001/1.9914E−001**
f_{a_9}	2.5907E+001/1.7552E+000(<)	**1.5623E+001/7.8144E+000**(>)	2.1347E+001/5.2616E+000(>)	1.9417E+001/4.4706E+000(>)	2.5672E+001/1.4240E+000(<)	2.6395E+001/1.3400E+000(<)	2.3582E+001/5.2220E+000
$f_{a_{10}}$	4.4382E−004/2.2957E−003(<)	*0/0(≈)*	*0/0(≈)*	1.2084E−003/3.1325E−003(<)	*0/0(≈)*	*0/0(≈)*	*0/0*
$f_{a_{11}}$	6.9104E−014/2.6207E−014(>)	7.8036E−002/2.7016E−001(<)	1.3656E+000/2.7856E+000(<)	2.1177E−014/2.7756E−014(>)	1.6607E−013/4.5251E−014(>)	**5.7958E−014/1.7640E−014**(>)	1.6719E−013/2.6447E−014
$f_{a_{12}}$	**5.9753E+000/1.4404E+000**(>)	7.2578E+000/1.8632E+000(<)	9.4671E+000/3.1203E+000(<)	9.3849E+000/2.4623E+000(<)	1.1116E+001/1.7749E+000(<)	7.2758E+000/1.6292E+000(<)	6.7762E+000/3.1934E+000
$f_{a_{13}}$	**6.8696E+000/2.6338E+000**(>)	8.5540E+000/4.8440E+000(<)	9.8862E+000/7.0392E+000(<)	1.3901E+001/6.2850E+000(<)	1.9158E+001/6.3415E+000(<)	9.9705E+000/4.9349E+000(<)	8.2574E+000/5.9276E+000
$f_{a_{14}}$	3.1025E−002/2.2948E−002(<)	7.8086E+000/1.6971E+001(<)	8.0230E+001/1.9408E+002(<)	**8.5726E−003/1.1906E−002**(>)	1.1430E−002/1.3372E−002(>)	2.2452E−002/1.9010E−002(<)	1.8370E−002/2.0672E−002
$f_{a_{15}}$	2.6983E+003/2.6163E+002(<)	**2.4976E+003/3.6658E+002**(>)	2.9395E+003/6.3977E+002(<)	2.8164E+003/4.2390E+002(<)	2.8465E+003/2.3832E+002(<)	2.6440E+003/3.0005E+002(<)	2.5812E+003/3.0053E+002
$f_{a_{16}}$	7.3260E−001/1.7508E−001(<)	1.6819E+000/5.1517E−001(<)	2.1124E+000/3.9900E−001(<)	**5.3721E−001/3.0302E−001**(>)	7.4101E−001/4.4157E−001(<)	7.1920E−001/1.5914E−001(<)	5.4322E−001/2.4416E−001
$f_{a_{17}}$	*3.0434E+001/1.8285E−006(≈)*	3.0583E+001/1.0449E−001(<)	3.0669E+001/1.0979E−001(<)	*3.0434E+001/9.4299E−007(≈)*	*3.0434E+001/4.4578E−014(≈)*	*3.0434E+001/9.4299E−007(≈)*	*3.0434E+001/1.9302E−012*
$f_{a_{18}}$	5.2782E+001/2.7522E+000(<)	5.6715E+001/6.3370E+000(<)	7.4321E+001/1.9615E+001(<)	**4.6144E+001/5.0000E+000**(>)	6.2502E+001/4.7293E+000(<)	5.3188E+001/3.3317E+000(<)	4.9702E+001/2.9301E+000
$f_{a_{19}}$	1.1636E+000/7.5338E−002(<)	1.4227E+000/1.7112E−001(<)	1.4180E+000/2.2050E−001(<)	1.2011E+000/1.5304E−001(<)	1.1670E+000/1.0659E−001(<)	**1.1404E+000/7.9575E−002**(>)	1.1602E+000/9.1644E−002
$f_{a_{20}}$	1.1660E+001/2.2028E+000(<)	9.1699E+000/3.7804E−001(<)	9.4427E+000/3.9441E−001(<)	9.1274E+000/4.7762E−001(<)	9.9035E+000/8.4388E−001(<)	1.1006E+001/2.1000E+000(<)	**8.9848E+000/3.6837E−001**
$f_{a_{21}}$	3.0648E+002/3.7329E+001(≈)	3.0427E+002/5.5153E+001(>)	**2.9693E+002/3.1662E+001**(>)	3.0085E+002/2.4726E+001(>)	3.1126E+002/3.8976E+001(<)	3.0000E+002/1.4016E−013(>)	3.0648E+002/3.7329E+001
$f_{a_{22}}$	1.0625E+002/8.5427E−001(<)	1.1441E+002/2.9576E+000(<)	1.3152E+002/2.4845E+002(<)	**1.0589E+002/4.0471E−001**(>)	1.0608E+002/8.5515E−001(<)	1.0594E+002/3.3834E−001(>)	1.0599E+002/2.6425E−001
$f_{a_{23}}$	2.6481E+003/3.5124E+002(<)	**2.3353E+003/3.2700E+002**(>)	2.5111E+003/5.0809E+002(<)	2.6956E+003/3.7933E+002(<)	2.9329E+003/3.0792E+002(<)	2.6363E+003/2.9409E+002(<)	2.4599E+003/3.3073E+002
$f_{a_{24}}$	2.0002E+002/2.9193E−002(<)	*2.0000E+002/5.7280E−004(≈)*	2.0001E+002/1.1231E−002(<)	2.0002E+002/2.9880E−002(<)	2.0001E+002/6.2000E−003(<)	*2.0000E+002/3.5858E−003(≈)*	*2.0000E+002/6.0437E−003*
$f_{a_{25}}$	2.3534E+002/2.1775E+001(<)	2.2534E+002/2.1775E+001(>)	2.3295E+002/1.6954E+001(<)	2.3242E+002/1.8585E+001(<)	**2.0432E+002/1.3327E+001**(>)	2.2384E+002/2.0356E+001(>)	2.2815E+002/1.9585E+001
$f_{a_{26}}$	*2.0000E+002/1.4352E−013(≈)*	2.0196E+002/1.4003E+001(>)	*2.0000E+002/1.4352E−013(≈)*	*2.0000E+002/1.4262E−013(≈)*	*2.0000E+002/1.4352E−013(≈)*	*2.0000E+002/1.4352E−013(≈)*	*2.0000E+002/1.4171E−013*
$f_{a_{27}}$	3.0054E+002/1.4439E+000(<)	*3.0002E+002/6.5843E−002(≈)*	3.0054E+002/1.4439E−001(<)	3.0126E+002/2.2790E+000(<)	3.0020E+002/2.8726E−001(<)	3.0014E+002/3.1664E−001(<)	*3.0002E+002/5.3521E−002*
$f_{a_{28}}$	*3.0000E+002/2.1254E−013(≈)*	*3.0000E+002/0(≈)*	*3.0000E+002/2.1735E−013(≈)*	*3.0000E+002/7.8765E−014(≈)*	*3.0000E+002/2.6747E−013(≈)*	*3.0000E+002/2.3847E−013(≈)*	*3.0000E+002/1.6766E−013*
>/≈/<	3/6/19	7/5/16	3/5/20	7/5/16	3/6/19	7/6/15	−/−/−

Table 5
Mean and standard deviation (Mean/Std) of the 51-run fitness errors $\varDelta f = f - f^*$ of the seven DE variants on 50D optimization under f_{a_1}–$f_{a_{28}}$ of our test suite is given below. The overall performance behind "Mean/Std" is measure under Wilcoxon's signed rank test under the significant level $\alpha = 0.05$.

50D	LSHADE	iLSHADE	jSO	LPalmDE	HARD-DE	DE-NPC	DE-EXP
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{a_1}	5.7958E−014/1.0008E−013(<)	1.7833E−014/6.1737E−014(>)	2.6750E−014/7.3986E−014(>)	6.6875E−014/1.0463E−013(<)	1.3821E−013/1.1212E−013(<)	**1.7383E−014/6.1737E−014**(>)	3.5666E−014/8.3512E−014
f_{a_2}	9.249E+002/1.3597E+003(<)	2.6714E+003/2.3244E+003(<)	1.9562E+002/2.3571E+002(<)	2.9256E+003/2.2181E+003(<)	5.2530E+001/1.9605E+002(<)	1.8411E+002/3.8032E+002(<)	**8.0117E−005/4.1644E−004**
f_{a_3}	2.6339E+003/6.5833E+003(<)	1.6216E+002/4.5507E+002(<)	1.4583E+002/4.3159E+002(<)	6.8586E+003/2.5011E+004(<)	9.9769E+001/5.3150E+002(<)	3.3977E+002/1.4785E+003(<)	**6.9945E+001/3.2637E+002**
f_{a_4}	8.2144E−011/1.9742E−010(<)	1.2320E−010/2.3097E−010(<)	1.2065E−009/2.3586E−009(<)	3.4098E−010/6.7529E−010(<)	2.3416E−010/3.9326E−010(<)	6.2665E−010/1.7415E−009(<)	**2.4527E−013/6.1738E−014**
f_{a_5}	1.5381E−013/5.4870E−014(>)	**1.2706E−013/3.6993E−014**(>)	1.5381E−013/5.4870E−014(>)	1.5604E−013/5.5513E−014(>)	1.5604E−013/5.5516E−014(>)	1.3821E−013/4.7724E−014(>)	2.1846E−013/6.7615E−014
f_{a_6}	*4.3447E+001/0(≈)*	*4.3447E+001/0(≈)*	*4.3447E+001/1.6078E−014(≈)*	*4.3447E+001/4.3139E−014(≈)*	*4.3447E+001/0(≈)*	*4.3447E+001/0(≈)*	*4.3447E+001/0*
f_{a_7}	7.7228E−001/6.7620E−001(<)	1.9306E−001/1.8258E−001(>)	**6.5763E−002/6.1737E−002**(>)	1.2253E+000/9.7301E−001(>)	2.3422E−001/2.6737E−001(<)	1.8082E−001/1.9449E−001(>)	2.1706E−001/1.6815E−001
f_{a_8}	2.1051E+001/1.157E−001(<)	2.1107E+001/5.6943E−002(<)	2.1128E+001/3.6722E−002(<)	2.1033E+001/1.1969E−001(<)	2.1016E+001/1.0743E−001(<)	2.1020E+001/1.2587E−001(<)	**2.0806E+001/1.6005E−001**
f_{a_9}	5.0414E+001/2.2777E+000(<)	**1.6182E+001/6.4905E+000**(>)	3.2249E+001/1.1182E+001(>)	4.1135E+001/5.8519E+000(>)	5.1787E+001/1.9363E+000(<)	5.2046E+001/1.9502E+000(<)	4.3314E+001/1.2786E+001
$f_{a_{10}}$	1.1642E−002/1.1247E−002(<)	3.1899E−003/4.9855E−003(<)	1.4502E−004/1.0356E−003(>)	1.3378E−002/1.2040E−002(<)	2.9963E−003/4.4717E−003(<)	3.8147E−003/6.3262E−003(<)	**4.3506E−004/1.7576E−003**
$f_{a_{11}}$	1.3221E−010/3.6555E−010(<)	4.5857E−001/7.5397E−001(<)	7.7648E+000/5.0926E+000(<)	**7.4677E−014/2.6638E−014**(>)	3.4775E−013/4.9093E−014(>)	3.9567E−013/1.2400E−013(<)	3.8007E−013/6.0682E−014
$f_{a_{12}}$	1.6709E+001/2.6412E+000(<)	1.2578E+001/3.9212E+000(<)	1.3509E+001/4.7574E+000(<)	2.1648E+001/5.1529E+000(<)	2.4586E+001/3.5151E+000(<)	1.4147E+001/2.4593E+000(<)	**1.1920E+001/4.0097E+000**
$f_{a_{13}}$	2.8363E+001/9.8731E+000(<)	2.3523E+001/1.2067E+001(<)	2.2754E+001/1.6445E+001(<)	5.0039E+001/1.7901E+001(<)	5.5767E+001/1.2911E+001(<)	2.6166E+001/9.2945E+000(<)	**2.1096E+001/1.1083E+001**
$f_{a_{14}}$	1.9727E−001/1.0789E−001(<)	3.6001E+001/1.5311E+001(<)	3.0964E+002/8.7951E+002(<)	1.5512E−001/7.0886E−002(<)	5.1794E−002/2.3576E−002(>)	**3.9129E−002/2.3422E−002**(>)	6.9441E−002/2.9201E−002
$f_{a_{15}}$	6.4542E+003/3.9877E+002(<)	6.2454E+003/5.0025E+002(<)	6.9122E+003/9.3992E+002(<)	6.0666E+003/6.3052E+002(<)	6.4640E+003/4.5372E+002(<)	6.2004E+003/4.3256E+002(<)	**6.0574E+003/3.3360E+002**
$f_{a_{16}}$	1.2129E+000/1.6519E−001(<)	2.1838E+000/5.7811E−001(<)	2.6916E+000/4.0196E−001(<)	**9.2173E−001/3.9366E−001**(>)	1.1915E+000/5.2479E−001(<)	1.1378E+000/1.6060E−001(<)	1.0061E+000/1.9906E−001
$f_{a_{17}}$	*5.0787E+001/3.3075E−003(≈)*	5.2030E+001/4.0685E−001(<)	5.6516E+001/3.0326E+000(<)	*5.0786E+001/2.1058E−010(≈)*	*5.0786E+001/5.3799E−010(≈)*	*5.0786E+001/2.6825E−010(≈)*	*5.0786E+001/2.0776E−011*
$f_{a_{18}}$	1.0280E+002/6.6368E+000(<)	1.1461E+002/1.1062E+001(<)	1.2958E+002/1.2305E+001(<)	**7.5333E+001/7.2630E+000**(>)	1.2122E+002/5.9866E+000(<)	9.9714E+001/5.6415E+000(<)	9.0528E+001/7.1742E+000
$f_{a_{19}}$	2.5311E+000/1.7194E−001(<)	3.1088E+000/3.0623E−001(<)	3.4462E+000/3.0034E−001(<)	2.3954E+000/2.4700E−001(<)	2.4169E+000/1.3590E−001(<)	**2.3766E+000/1.3670E−001**(>)	2.4132E+000/1.4231E−001
$f_{a_{20}}$	1.7668E+001/5.5237E−001(<)	1.8728E+001/4.9106E−001(<)	1.8487E+001/6.2468E−001(<)	1.7644E+001/5.5440E−001(<)	1.8544E+001/3.1125E−001(<)	1.8468E+001/1.3031E+000(<)	**1.7565E+001/5.6135E−001**
$f_{a_{21}}$	1.0038E+003/2.8090E+002(<)	8.7719E+002/3.8864E+002(<)	9.4519E+002/3.1906E+002(<)	1.0262E+003/2.7798E+002(<)	8.2421E+002/4.0077E+002(<)	7.8550E+002/4.4022E+002(>)	**8.0180E+002/3.9117E+002**
$f_{a_{22}}$	1.2339E+001/1.0464E+000(<)	4.1602E+001/1.0752E+001(<)	**1.0234E+002/9.5875E+000**(>)	1.1896E+001/7.3637E−001(<)	1.1407E+001/8.3877E−001(>)	1.1734E+001/7.0567E−001(>)	1.1762E+001/8.4501E−001
$f_{a_{23}}$	5.9844E+003/4.6908E+002(<)	**5.3110E+003/4.7675E+002**(>)	5.6318E+003/8.1372E+002(>)	5.8477E+003/6.1138E+002(<)	6.5263E+003/4.5941E+002(<)	6.0235E+003/4.3176E+002(<)	5.7407E+003/4.1858E+002
$f_{a_{24}}$	2.0267E+002/1.9079E+000(<)	2.0015E+002/1.5477E−001(>)	**2.0011E+002/6.9296E−002**(>)	2.0550E+002/4.1473E+000(<)	2.0013E+002/1.2767E−001(>)	2.0011E+002/9.8626E−002(>)	2.0045E+002/5.7637E−001
$f_{a_{25}}$	2.8693E+002/6.2195E+000(<)	2.8173E+002/7.4417E+000(<)	*2.7885E+002/4.8900E+000(≈)*	2.8772E+002/6.4083E+000(<)	2.8219E+002/6.2528E+000(<)	2.8115E+002/5.7740E+000(<)	*2.7884E+002/6.3686E+000*
$f_{a_{26}}$	2.2751E+002/4.1547E+001(>)	2.7925E+002/4.1987E+001(<)	2.3158E+002/4.7168E+001(>)	2.6471E+002/5.2566E+001(<)	2.2843E+002/4.6708E+001(>)	**2.1955E+002/3.7341E+001**(>)	2.5419E+002/5.1624E+001
$f_{a_{27}}$	3.4660E+002/3.1252E+001(<)	3.1232E+002/1.9528E+001(>)	**3.0545E+002/8.4079E+000**(>)	3.7488E+002/4.6618E+001(<)	3.0753E+002/8.1712E+000(>)	3.1098E+002/1.3524E+001(>)	3.1694E+002/1.7519E+001
$f_{a_{28}}$	*4.0000E+002/2.8433E−013(≈)*	*4.0000E+002/2.8433E−013(≈)*	*4.0000E+002/2.8433E−013(≈)*	*4.0000E+002/2.7882E−013(≈)*	*4.0000E+002/3.0282E−013(≈)*	*4.0000E+002/2.8705E−013(≈)*	*4.0000E+002/2.8705E−013*
>/≈/<	2/3/23	7/2/19	10/3/15	7/3/18	8/3/17	9/3/16	−/−/−

Table 6
Mean and standard deviation (Mean/Std) of the 51-run fitness errors $\varDelta f = f - f^*$ of the seven DE variants on 10D optimization under f_{b_1}–$f_{b_{28}}$ of our test suite is given below. The overall performance behind "Mean/Std" is measure under Wilcoxon's signed rank test under the significant level $\alpha = 0.05$.

10D	LSHADE	iLSHADE	jSO	LPalmDE	HARD-DE	DE-NPC	DE-EXP
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{b_1}	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0*
f_{b_2}	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0*
f_{b_3}	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0*
f_{b_4}	2.8643E+001/1.3391E+001(<)	2.9410E+001/1.2589E+001(<)	2.5318E+001/1.5547E+001(<)	3.0774E+001/1.1096E+001(<)	**1.8156E+001/1.7166E+001**(>)	2.4728E+001/1.6677E+001(≈)	2.4721E+001/1.5760E+001
f_{b_5}	1.2917E+001/9.3008E+000(<)	1.4334E+001/9.2312E+000(<)	1.5746E+001/8.1832E+000(<)	1.3513E+001/1.3407E+000(<)	1.2614E+001/9.2874E+000(<)	1.1646E+001/9.3908E+000(<)	**1.1428E+001/9.3051E+000**
f_{b_6}	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	4.1044E−003/2.9311E−002(<)	*0/0(≈)*	*0/0*
f_{b_7}	1.7877E−003/5.2151E−003(>)	1.3037E−003/4.0714E−003(>)	9.6640E−004/3.1965E−003(>)	9.6664E−004/2.7036E−003(>)	**6.8643E−004/2.3928E−003**(>)	5.7066E−003/8.8419E−003(>)	1.0567E−002/9.8557E−003
f_{b_8}	*0/0*(>)	4.4583E−015/2.2287E−014(>)	1.9509E−002/1.3932E−001(<)	*0/0*(>)	2.0062E−014/4.3771E−014(<)	*0/0*(>)	6.6875E−015/2.7016E−014
f_{b_9}	2.4404E+000/8.2810E−001(>)	**1.7775E+000/8.0437E−001**(>)	2.2241E+000/9.4581E−001(>)	2.7142E+000/1.2260E+000(>)	2.4239E+000/8.5067E−001(>)	2.7915E+000/8.9033E−001(>)	2.9274E+000/9.4175E−001
$f_{b_{10}}$	2.4492E−003/1.2244E−002(<)	2.5026E−001/6.7032E−001(<)	1.6165E+000/2.7412E+000(<)	**0/0**(>)	1.2246E−003/8.7454E−003(<)	7.3476E−003/2.3854E−002(<)	4.9933E−013/4.5706E−013
$f_{b_{11}}$	3.4377E+001/4.1882E+001(<)	3.8393E+001/5.2556E+001(<)	5.1877E+001/8.1131E+001(<)	6.6002E+001/8.7406E+001(<)	**2.6791E+001/3.8066E+001**(>)	3.5205E+001/4.2210E+001(<)	2.8800E+001/3.4380E+001
$f_{b_{12}}$	6.3747E−002/1.6325E−002(<)	7.2427E−002/4.0705E−002(<)	3.4238E−001/1.3941E−001(<)	**3.2878E−002/3.5793E−002**(>)	6.7807E−002/4.6228E−002(<)	6.0785E−002/1.7818E−002(<)	4.9903E−002/1.6398E−002
$f_{b_{13}}$	5.3490E−002/1.1847E−002(<)	6.1216E−002/1.8513E−002(<)	7.8240E−002/1.7179E−002(<)	5.7668E−002/1.3761E−002(<)	5.6665E−002/2.7154E−002(<)	5.3601E−002/1.5765E−002(<)	**4.8072E−002/9.9534E−003**
$f_{b_{14}}$	7.7271E−002/2.8093E−002(<)	6.4314E−002/2.4141E−002(<)	**5.8643E−002/1.5338E−002**(>)	7.6499E−002/2.6935E−002(<)	7.6713E−002/2.6868E−002(<)	6.8993E−002/2.3429E−002(<)	6.2240E−002/2.4254E−002
$f_{b_{15}}$	3.7205E−001/7.2872E−002(<)	3.8731E−001/7.8599E−002(<)	4.8187E−001/9.6368E−002(<)	4.2286E−001/9.5352E−002(<)	4.0415E−001/7.4329E−002(<)	3.8768E−001/6.3774E−002(<)	**3.5983E−001/7.7799E−002**
$f_{b_{16}}$	1.1104E+000/2.4880E−001(>)	**8.7711E−001/2.8846E−001**(>)	1.0483E+000/3.3512E−001(>)	1.3111E+000/3.9797E−001(<)	1.3762E+000/3.2230E−001(<)	1.2019E+000/2.8691E−001(>)	1.2446E+000/2.7155E−001
$f_{b_{17}}$	9.7098E−001/8.8170E−001(>)	9.3941E+000/2.4119E+001(<)	3.6925E+000/4.1900E+000(<)	**8.6549E−001/8.1912E−001**(>)	2.0826E+000/1.9141E+000(>)	9.5463E−001/1.2122E+000(>)	2.9647E+000/3.0025E+000
$f_{b_{18}}$	1.1200E−001/1.0958E−001(>)	1.7056E−001/1.2843E−001(>)	1.9543E−001/1.8980E−001(<)	**4.2454E−002/7.3100E−002**(>)	1.1603E−001/9.8590E−002(>)	7.8871E−002/9.6281E−002(>)	1.8183E−001/2.8934E−001
$f_{b_{19}}$	5.7219E−002/2.8940E−002(<)	8.3738E−002/2.2522E−001(<)	**4.1065E−002/4.2050E−002**(>)	6.0962E−002/3.4183E−002(<)	6.5786E−002/3.3864E−002(<)	5.5476E−002/2.8833E−002(<)	5.1848E−002/2.7327E−002
$f_{b_{20}}$	1.5828E−001/1.2275E−001(<)	2.6610E−001/2.0067E−001(<)	3.2640E−001/1.8476E−001(<)	1.41822E−001/9.4424E−002(<)	1.5696E−001/7.3592E−002(<)	1.6969E−001/1.0349E−001(<)	**1.3403E−001/9.7276E−002**
$f_{b_{21}}$	2.3399E−001/2.6232E−001(>)	2.8968E−001/2.9030E−001(<)	4.2358E−001/3.0483E−001(<)	**1.7171E−001/2.2415E−001**(>)	3.8724E−001/2.4126E−001(<)	2.3496E−001/2.5940E−001(>)	2.5409E−001/3.0078E−001
$f_{b_{22}}$	7.6473E−002/2.3716E−002(>)	4.5479E−001/2.8149E+000(<)	1.9194E+000/5.4528E+000(<)	**2.8462E−002/26.108E−002**(>)	1.1142E−001/3.9978E−002(<)	6.9815E−002/3.4824E−002(>)	1.0910E−001/3.7216E−002

(*continued on next page*)

Table 6 (*continued*).

$f_{b_{23}}$	3.2946E+002/2.8705E−013(≈)	3.2946E+002/2.8705E−013(≈)	3.2946E+002/2.8705E−013(≈)	3.2946E+002/2.8705E−013(≈)	3.2946E+002/2.8705E−013(≈)	3.2946E+002/2.8705E−013(≈)	3.2946E+002/2.8705E−013
$f_{b_{24}}$	1.0660E+002/2.5712E+000(>)	**1.0610E+002/2.9668E+000**(>)	1.0791E+002/2.1510E+000(>)	1.0657E+002/3.1381E+000(>)	1.0778E+002/1.6099E+000(>)	1.0804E+002/2.4396E+000(≈)	1.0802E+002/2.6214E+000
$f_{b_{25}}$	1.1939E+002/2.2603E+001(<)	1.3950E+002/4.2018E+001(<)	1.2298E+002/2.1742E+001(<)	1.2067E+002/2.8471E+001(<)	1.2422E+002/2.7927E+001(<)	**1.0938E+002/5.6995E+000**(>)	1.1760E+002/2.4365E+001
$f_{b_{26}}$	1.0005E+002/1.6684E−002(≈)	1.0006E+002/2.0917E−002(≈)	1.0008E+002/1.4474E−002(<)	1.0004E+002/1.6744E−002(≈)	1.0007E+002/2.9799E−002(<)	1.0006E+002/1.4303E−002(≈)	1.0005E+002/8.7390E−003
$f_{b_{27}}$	5.4209E+001/1.3423E+002(>)	9.8799E+001/1.4713E+002(<)	2.6608E+001/8.8777E+001(>)	**1.4935E+001/6.9110E+001**(>)	1.7120E+001/7.8151E+001(>)	5.4132E+001/1.3426E+002(>)	7.1660E+001/1.3700E+002
$f_{b_{28}}$	3.8485E+002/4.8766E+001(>)	3.9366E+002/5.3309E+001(>)	**3.7873E+002/4.4040E+001**(>)	3.9995E+002/5.5809E+001(<)	3.7989E+002/4.3673E+001(>)	3.8283E+002/4.6279E+001(>)	3.9950E+002/5.3158E+001
$f_{b_{29}}$	2.2187E+002/3.1229E−001(<)	2.2191E+002/3.4864E−001(<)	2.2186E+002/2.8459E−001(<)	2.2183E+002/2.9854E−001(≈)	**2.2004E+002/1.2613E+001**(>)	2.2207E+002/1.2624E+001(<)	2.2183E+002/2.6762E−001
$f_{b_{30}}$	4.6604E+002/1.0320E+001(<)	4.7845E+002/2.1950E+001(<)	4.6724E+002/1.0531E+001(<)	4.6628E+002/1.0518E+001(<)	4.6682E+002/1.3696E+001(<)	4.6492E+002/9.1788E+000(<)	**4.6276E+002/1.2223E+001**
>/≈/<	11/6/13	7/6/17	9/5/16	11/7/12	10/4/16	11/8/11	−/−/−

Table 7
Mean and standard deviation (Mean/Std) of the 51-run fitness errors $\Delta f = f - f^*$ of the seven DE variants on 30*D* optimization under f_{b_1}–$f_{b_{28}}$ of our test suite is given below. The overall performance behind "Mean/Std" is measure under Wilcoxon's signed rank test under the significant level $\alpha = 0.05$.

30D	LSHADE	iLSHADE	jSO	LPalmDE	HARD-DE	DE-NPC	DE-EXP
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{b_1}	1.0588E−014/6.8703E−015(<)	1.6719E−014/7.8897E−015(<)	4.7648E−014/2.8254E−014(<)	1.3932E−014/3.4695E−015(<)	4.5831E−014/2.5896E−014(<)	1.3932E−014/5.3098E−015(<)	**0/0**
f_{b_2}	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	5.5729E−016/3.9798E−015(<)	5.5729E−016/3.9798E−015(<)	*0/0*(≈)	*0/0*
f_{b_3}	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	3.3437E−015/1.3508E−014(<)	*0/0*(≈)	*0/0*(≈)	*0/0*
f_{b_4}	3.6781E−014/2.7435E−014(<)	2.1177E−014/2.7756E−014(<)	6.0187E−014/2.3879E−014(<)	4.2354E−014/2.5019E−014(<)	5.6843E−014/2.5421E−014(<)	3.3437E−014/3.0455E−014(<)	**1.2260E−014/2.3612E−014**
f_{b_5}	2.0112E+001/2.0019E−002(<)	2.0688E+001/2.7129E−001(<)	2.0914E+001/7.2603E−002(<)	2.0082E+001/9.3765E−002(<)	2.0143E+001/6.2016E−002(<)	2.0091E+001/1.9556E−002(<)	**2.0048E+001/1.2009E−002**
f_{b_6}	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	3.7875E−001/1.5852E+000(<)	2.8083E−005/1.3837E−004(>)	1.2468E−001/8.9040E−001(<)	*0/0*
f_{b_7}	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*
f_{b_8}	1.4044E−013/6.6633E−014(>)	7.8036E−002/2.7016E−001(<)	9.1693E−001/2.7919E+000(<)	**7.5791E−014/5.4126E−014**(>)	3.2992E−013/6.5328E−014(≈)	1.0700E−013/2.7016E−014(>)	3.2323E−013/6.5791E−014
f_{b_9}	**7.4492E+000/1.1677E+000**(>)	7.7881E+000/1.8335E+000(>)	9.8932E+000/2.7750E+000(>)	1.0892E+001/2.7318E+000(>)	1.2642E+001/1.8102E+000(<)	9.0170E+000/2.1722E+000(>)	1.1361E+001/3.1635E+000
$f_{b_{10}}$	2.0411E−003/6.2526E−003(<)	2.8247E+000/1.6709E+001(<)	5.4145E+001/2.5318E+002(<)	9.4868E+001/9.6695E+001(<)	**3.1743E−012/1.0188E−012**(>)	1.2247E−003/4.9474E−003(≈)	1.2247E−003/4.9474E−003
$f_{b_{11}}$	**1.2185E+003/1.8570E+002**(>)	1.5573E+003/2.8119E+002(<)	1.8032E+003/3.6543E+002(<)	1.4063E+003/2.9469E+002(<)	1.2600E+003/1.8535E+002(<)	1.267E+003/1.9949E+002(<)	1.2190E+003/1.9731E+002
$f_{b_{12}}$	1.5695E−001/2.5454E−002(<)	2.4798E−001/2.0682E−001(<)	1.0806E+000/4.1389E−001(<)	1.4980E−001/5.4082E−002(<)	1.8515E−001/3.5220E−002(<)	1.3928E−001/2.3011E−002(<)	**1.2480E−001/2.1567E−002**
$f_{b_{13}}$	1.1496E−001/1.9333E−002(>)	1.3297E−001/2.5120E−002(<)	1.5271E−001/2.1751E−002(<)	**1.0232E−001/2.4448E−002**(>)	1.4210E−001/2.8017E−002(<)	1.1921E−001/1.8702E−002(>)	1.1968E−001/2.0558E−002
$f_{b_{14}}$	2.0877E−001/2.3154E−002(<)	2.0947E−001/2.8171E−002(<)	2.0205E−001/3.5807E−002(<)	2.2578E−001/3.5593E−002(<)	2.0759E−001/2.7644E−002(<)	1.9096E−001/2.2394E−002(<)	**1.7712E−001/2.4888E−002**
$f_{b_{15}}$	2.1762E+000/2.3672E−001(<)	2.4232E+000/3.7199E−001(<)	2.6582E+000/4.3749E−001(<)	**2.0456E+000/4.7248E−001**(>)	2.3990E+000/2.7075E−001(<)	2.1054E+000/2.5422E−001(<)	2.0540E+000/2.0895E−001
$f_{b_{16}}$	8.5518E+000/4.7093E−001(<)	**8.3762E+000/7.1098E−001**(>)	9.5569E+000/9.2548E−001(<)	8.8923E+000/6.0375E−001(<)	8.8255E+000/3.4314E−001(<)	8.5875E+000/4.4687E−001(<)	8.5457E+000/4.7679E−001
$f_{b_{17}}$	1.7389E+002/9.6699E+001(<)	1.6268E+002/9.9703E+001(<)	7.3699E+001/3.6147E+001(<)	2.2437E+002/1.2956E+002(<)	1.0824E+002/6.2466E+001(<)	7.9757E+001/4.4185E+001(<)	**5.3267E+001/3.2671E+001**
$f_{b_{18}}$	5.3004E+000/2.0879E+000(<)	4.0912E+000/1.7524E+000(<)	2.1676E+000/1.0918E+000(<)	7.0246E+000/3.5178E+000(<)	4.3170E+000/1.7092E+000(<)	3.1463E+000/1.3089E+000(<)	**1.6533E+000/1.0946E+000**
$f_{b_{19}}$	3.6959E+000/5.3606-001(<)	2.8962E+000/6.8683E−001(<)	**2.2423E+000/6.6130E−001**(>)	3.0607E+000/5.2074E−001(<)	3.0072E+000/5.0998E−001(<)	2.5817E+000/5.7241E−001(<)	2.4130E+000/6.8816E−001
$f_{b_{20}}$	2.9006E+000/1.0494E+000(>)	2.9674E+000/1.2775E+000(>)	**2.2072E+000/9.0519E−001**(>)	3.2820E+000/1.7851E+000(<)	3.3042E+000/1.2267E+000(<)	2.3131E+000/7.3208E−001(>)	2.9915E+000/1.0802E+000
$f_{b_{21}}$	7.9414E+001/6.9809E+001(<)	8.2068E+001/7.5471E+001(<)	2.3298E+001/3.8531E+001(<)	9.3862E+001/7.6847E+001(<)	3.8996E+001/5.1896E+001(<)	2.7895E+001/4.6523E+001(<)	**1.7459E+001/2.0246E+001**
$f_{b_{22}}$	5.6247E+001/4.9410E+001(<)	6.3487E+001/5.5824E+001(<)	6.5507E+001/5.5656E+001(<)	6.9793E+001/5.7971E+001(<)	5.8421E+001/5.2158E+001(<)	6.0016E+001/5.2349E+001(<)	**4.7141E+001/4.7288E+001**
$f_{b_{23}}$	*3.1524E+002/4.0186E−013*(≈)	*3.1524E+002/3.5915E−013*(≈)	*3.1524E+002/4.0186E−013*(≈)	*3.1524E+002/4.0186E−013*(≈)	*3.1524E+002/4.0186E−013*(≈)	*3.1524E+002/4.0186E−013*(≈)	*3.1524E+002/4.0186E−013*
$f_{b_{24}}$	2.2353E+002/8.9272E−001(<)	2.1254E+002/1.1047E+001(<)	**2.0085E+002/4.2399E+000**(>)	2.2371E+002/8.3526E−001(<)	2.2166E+002/3.1533E+000(<)	2.2110E+002/4.3617E+000(<)	2.1241E+002/1.0939E+001
$f_{b_{25}}$	2.0268E+002/1.2945E−001(<)	2.0263E+002/1.2083E−001(<)	*2.0259E+002/3.5603E−002*(≈)	2.0272E+002/1.3954E−001(<)	*2.0260E+002/4.6251E−002*(≈)	*2.0261E+002/4.9849E−002*(≈)	*2.0260E+002/5.2683E−002*
$f_{b_{26}}$	1.0012E+002/1.4692E−002(≈)	1.0013E+002/2.1180E−002(<)	1.0015E+002/1.9050E−002(<)	**1.0009E+002/1.9098E−002**(>)	1.0014E+002/2.6636E−002(<)	1.0012E+002/1.6611E−002(≈)	1.0011E+002/1.7762E−002
$f_{b_{27}}$	*3.0000E+002/9.0949E−014*(≈)	*3.0000E+002/6.4311E−014*(≈)	*3.0000E+002/2.1330E−013*(≈)	*3.0000E+002/1.7015E−013*(≈)	3.0590E+002/2.3833E+001(<)	3.0196E+002/1.4003E+001(<)	*3.0000E+002/1.2531E−013*
$f_{b_{28}}$	8.559E+002/1.9630E+001(<)	**8.1958E+002/2.6402E+001**(>)	8.2353E+002/2.5787E+001(<)	8.5921E+002/1.4595E+001(<)	8.4738E+002/1.8940E+001(<)	8.3445E+002/2.0261E+001(<)	8.2148E+002/2.1032E+001
$f_{b_{29}}$	6.5259E+002/1.7898E+000(<)	7.0586E+002/8.1020E+001(<)	6.4286E+002/1.8776E+002(<)	7.0755E+002/7.0083E+001(<)	5.4119E+002/2.6230E+002(<)	5.3687E+002/2.6863E+002(<)	**5.2335E+002/2.7633E+002**
$f_{b_{30}}$	5.2218E+002/1.3844E+002(<)	5.0455E+002/1.5801E+002(<)	**4.3044E+002/2.3970E+001**(>)	5.5758E+002/1.9170E+002(<)	4.5178E+002/6.9319E+001(<)	4.6398E+002/1.2426E+002(<)	4.3544E+002/9.7758E+001
>/≈/<	5/7/18	6/4/20	5/7/18	5/3/22	3/5/22	4/7/19	−/−/−

Table 8
Mean and standard deviation (Mean/Std) of the 51-run fitness errors $\Delta f = f - f^*$ of the seven DE variants on 50*D* optimization under f_{b_1}–$f_{b_{28}}$ of our test suite is given below. The overall performance behind "Mean/Std" is measure under Wilcoxon's signed rank test under the significant level $\alpha = 0.05$.

50D	LSHADE	iLSHADE	jSO	LPalmDE	HARD-DE	DE-NPC	DE-EXP
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{b_1}	7.2911E+002/1.2386E+003(<)	1.5886E+003/2.0369E+003(<)	2.3039E+001/6.1174E+001(<)	1.6984E+003/1.8924E+003(<)	1.1047E+001/2.1347E+001(<)	2.0641E+001/7.3679E+001(<)	**4.9983E−005/3.4712E−004**
f_{b_2}	3.6781E−014/1.3079E−014(>)	**3.2880E−014/1.0439E−014**(>)	5.0713E−014/1.4283E−014(>)	3.6224E−014/1.2810E−014(>)	6.6317E−014/1.7673E−014(>)	3.8453E−014/1.3717E−014(>)	1.4267E−013/4.8397E−014
f_{b_3}	**5.4614E−014/1.1144E−014**(>)	5.5729E−014/7.9597E−015(>)	5.9073E−014/1.1144E−014(≈)	9.9197E−014/3.0247E−013(<)	5.7958E−014/7.9597E−015(>)	5.6843E−014/1.1369E−014(>)	5.9073E−014/1.1144E−014
f_{b_4}	6.3681E+000/2.3174E+001(>)	1.1579E+001/3.1909E+001(<)	8.0071E+000/2.6568E+001(>)	1.1802E+001/3.1831E+001(<)	1.7793E+001/3.7554E+001(<)	**3.8787E+000/1.9226E+001**(>)	9.8788E+000/2.9396E+001
f_{b_5}	2.0233E+001/3.3718E−002(<)	2.0804E+001/2.6090E−001(<)	2.1065E+001/7.7530E−002(<)	2.0197E+001/1.4416E−001(<)	2.0271E+001/6.8348E−002(<)	2.0190E+001/2.8650E−002(<)	**2.0120E+001/1.8544E−002**
f_{b_6}	1.1513E−001/3.1524E−001(<)	1.4708E−002/1.0501E−001(>)	1.5887E−002/1.0811E−001(>)	5.1362E−001/2.4081E+000(<)	**1.0220E−002/7.2659E−002**(>)	1.8222E−002/1.2523E−001(>)	8.0019E−002/3.1183E−001
f_{b_7}	1.0031E−013/3.6993E−014(<)	**5.5729E−014/5.7398E−015**(>)	9.1395E−014/4.5586E−014(<)	1.0923E−013/2.2287E−014(<)	9.8083E−014/3.9511E−014(<)	6.4645E−014/5.6866E−014(>)	8.2479E−014/5.1240E−014
f_{b_8}	7.1950E−011/8.9461E−011(>)	5.4698E−001/7.7870E−001(<)	6.3024E+000/6.8627E+000(<)	**1.5158E−013/6.2957E−014**(>)	7.3785E−013/2.1495E−013(>)	7.7352E−013/1.5425E−013(>)	1.1705E−001/8.3593E−001
f_{b_9}	1.3791E+001/1.8387E+000(>)	**1.0975E+001/2.4399E+000**(>)	1.3299E+001/3.7158E+000(>)	2.1245E+001/3.9937E+000(<)	2.4662E+001/2.5038E+000(<)	1.5066E+001/2.3527E+000(>)	1.7315E+001/4.7106E+000
$f_{b_{10}}$	3.2929E−002/1.8038E−002(<)	9.8567E+000/2.3477E+001(<)	2.9425E+001/3.4091E+001(<)	**4.7046E−003/7.6084E−003**(>)	4.9023E−003/8.3134E−003(>)	6.8582E−003/8.0234E−003(<)	6.1234E−003/9.4715E−003
$f_{b_{11}}$	3.2911E+003/2.9448E+002(<)	3.8767E+003/4.8712E+002(<)	4.3214E+003/1.0006E+003(<)	3.3840E+003/4.5434E+002(<)	3.2466E+003/3.3174E+002(<)	3.1657E+003/3.3430E+002(<)	**3.0756E+003/2.9638E+002**
$f_{b_{12}}$	2.0257E−001/2.9794E−002(<)	3.8234E−001/3.7087E−001(<)	1.6566E+000/5.7177E−001(<)	2.1640E−001/7.1044E−002(<)	2.2928E−001/3.1661E−002(<)	1.7975E−001/2.3311E−002(<)	**1.4845E−001/1.6356E−002**
$f_{b_{13}}$	1.7288E−001/1.7202E−002(<)	1.9026E−001/2.6643E−002(<)	1.9024E−001/2.7143E−002(<)	**1.5737E−001/2.7647E−002**(>)	2.2319E−001/3.2708E−002(<)	1.8139E−001/2.3625E−002(<)	1.6587E−001/2.2757E−002
$f_{b_{14}}$	2.9046E−001/2.1400E−002(<)	2.5239E−001/4.3092E−002(<)	2.4498E−001/3.7341E−002(<)	3.0400E−001/2.7880E−002(<)	2.8159E−001/2.3757E−002(<)	2.5420E−001/2.4085E−002(<)	**2.2137E−001/2.6177E−002**
$f_{b_{15}}$	5.0840E+000/4.3531E−001(<)	5.8149E+000/5.5733E−001(<)	6.4773E+000/8.7088E−001(<)	**4.3943E+000/6.9609E−001**(>)	5.3939E+000/4.8744E−001(<)	4.7776E+000/3.8271E−001(<)	4.6650E+000/4.4711E−001
$f_{b_{16}}$	1.6911E+001/4.3168E−001(<)	**1.6731E+001/5.7272E−001**(>)	1.8269E+001/1.3117E+000(<)	1.6934E+001/8.5908E−001(<)	1.7049E+001/5.4247E−001(<)	1.6903E+001/4.8263E−001(<)	1.6837E+001/4.1798E−001
$f_{b_{17}}$	1.6233E+003/3.8679E+002(<)	1.0505E+003/3.3014E+002(<)	4.6237E+002/2.0961E+002(<)	1.6386E+003/4.1198E+002(<)	7.6276E+002/3.0446E+002(<)	4.8732E+002/2.2611E+002(<)	**4.2686E+002/1.7010E+002**
$f_{b_{18}}$	9.9586E+001/1.2961E+001(<)	4.5221E+001/2.0016E+001(<)	**1.3600E+001/5.5881E+000**(>)	9.8882E+001/1.0605E+001(<)	4.0499E+001/1.5548E+001(<)	1.9762E+001/5.8060E+000(>)	2.1363E+001/7.3483E+000
$f_{b_{19}}$	8.7858E+000/1.8804E+000(>)	9.0548E+000/1.5072E+000(>)	9.6030E+000/1.1282E+000(<)	**7.7453E+000/1.3884E+000**(>)	8.7287E+000/1.6849E+000(>)	9.3843E+000/1.4081E+000(>)	9.5886E+000/9.4070E−001
$f_{b_{20}}$	1.3399E+001/4.5950E+000(<)	9.8423E+000/3.1744E+000(>)	**5.8403E+000/1.8461E+000**(>)	1.6329E+001/5.0696E+000(<)	1.1401E+001/3.9117E+000(<)	8.1078E+000/2.6421E+000(>)	9.9271E+000/3.2247E+000
$f_{b_{21}}$	5.3963E+002/1.4462E+002(<)	4.2496E+002/1.2846E+002(<)	**3.0733E+002/7.9338E+001**(>)	5.5794E+002/1.9924E+002(<)	4.0290E+002/1.3510E+002(<)	3.6028E+002/1.1359E+002(<)	3.0997E+002/9.6857E+001
$f_{b_{22}}$	1.4698E+002/6.0949E+001(>)	**1.3012E+002/6.2604E+001**(>)	1.3277E+002/5.8962E+001(>)	1.8443E+002/1.0030E+002(>)	1.8753E+002/7.8938E+001(>)	1.6385E+002/8.4173E+001(>)	2.1669E+002/1.2680E+002
$f_{b_{23}}$	*3.4400E+002/4.1771E−013*(≈)	*3.4400E+002/3.7049E−013*(≈)	*3.4400E+002/4.3943E−013*(≈)	*3.4400E+002/4.9267E−013*(≈)	*3.4400E+002/4.3618E−013*(≈)	*3.4400E+002/4.0888E−013*(≈)	*3.4400E+002/5.0511E−013*
$f_{b_{24}}$	2.7506E+002/6.0389E−001(<)	2.9356E+002/1.6417E+000(<)	**2.7137E+002/2.1145E+000**(>)	2.7507E+002/1.2495E+000(<)	2.7432E+002/1.1306E+000(<)	2.7359E+002/1.2520E+000(≈)	2.7356E+002/1.7398E+000
$f_{b_{25}}$	2.0576E+002/5.2586E−001(<)	2.0564E+002/3.5969E−001(<)	*2.0534E+002/1.9638E−001*(≈)	2.0581E+002/5.4333E−001(<)	2.0557E+002/2.9554E−001(<)	2.0550E+002/3.1917E−001(<)	*2.0534E+002/2.6680E−001*
$f_{b_{26}}$	1.0994E+002/2.9986E+001(<)	1.0999E+002/2.9712E+001(<)	**1.0021E+002/3.7152E−002**(>)	1.1972E+002/4.0043E+001(<)	1.0215E+002/1.3975E+001(>)	1.0213E+002/13.978E+001(>)	1.0409E+002/1.9570E+001
$f_{b_{27}}$	3.1250E+002/2.4923E+001(>)	3.2413E+003/2.7006E+001(<)	3.1801E+002/2.0177E+001(<)	3.2207E+002/3.2004E+001(<)	3.0588E+002/1.6436E+001(>)	**3.0371E+002/1.2830E+001**(>)	3.1628E+002/2.3606E+001
$f_{b_{28}}$	1.2591E+003/6.4020E+001(<)	**1.1640E+003/4.8553E+001**(>)	1.1676E+003/3.3851E+001(>)	1.2674E+003/6.4887E+001(<)	1.2517E+003/6.2591E+001(<)	1.1934E+003/4.6871E+001(<)	1.1772E+003/5.1534E+001
$f_{b_{29}}$	6.3384E+002/1.2465E+002(<)	4.8438E+002/1.0878E+002(<)	6.1916E+002/1.0847E+002(<)	6.0695E+002/1.3569E+002(<)	5.3394E+002/1.2423E+002	4.9875E+002/1.3567E+002(<)	**4.8276E+002/1.2967E+002**
$f_{b_{30}}$	9.3491E+003/5.6508E+002(<)	8.8597E+003/4.8304E+002(>)	**8.5887E+003/4.0433E+002**(>)	1.0015E+004/7.5158E+002(<)	9.1145E+003/5.0313E+002	9.1731E+003/5.2088E+002(<)	8.9474E+003/5.7901E+002
>/≈/<	8/1/21	11/1/18	12/3/15	8/1/21	9/1/20	13/2/15	−/−/−

Table 9
Mean and standard deviation (Mean/Std) of the 51-run fitness errors $\Delta f = f - f^*$ of the seven DE variants on 10*D* optimization under f_{c_1}–$f_{c_{28}}$ of our test suite is given below. The overall performance behind "Mean/Std" is measure under Wilcoxon's signed rank test under the significant level $\alpha = 0.05$.

10D	LSHADE	iLSHADE	jSO	LPalmDE	HARD-DE	DE-NPC	DE-EXP
NO.	Mean/Std	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(≈)	0/0
f_{c_2}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
f_{c_3}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
f_{c_4}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
f_{c_5}	2.4605E+00/8.2820E−01(>)	**1.9711E+00/8.3282E−01**(>)	2.1660E+00/9.8948E−01(>)	2.3216E+00/1.0654E+00(>)	2.3061E+00/7.0440E−01(>)	2.7524E+00/9.6627E−01(<)	2.6153E+00/7.9469E−01
f_{c_6}	4.4583E−15/2.2287E−14(<)	**0/0**(>)	1.3375E−14/3.993E−14(<)	1.7833E−14/4.1756E−14(<)	2.0062E−14/4.3771E−14(<)	1.7833E−14/4.1756E−14(<)	8.9166E−15/3.0869E−15
f_{c_7}	1.2034E+01/7.0465E−01(>)	**1.1997E+01/5.8191E−01**(>)	1.2489E+01/7.7781E−01(>)	1.2792E+01/1.3159E+00(<)	1.2473E+01/7.8338E−01(>)	1.2673E+01/8.1486E−01(<)	1.2550E+01/8.6592E−01
f_{c_8}	2.3231E+00/7.3572E−01(>)	**1.9320E+00/7.2914E−01**(>)	2.5167E+00/7.5387E−01(>)	2.5752E+00/1.0749E+00(>)	2.8925E+00/1.1128E+00(<)	2.8694E+00/6.7915E−01(<)	2.8098E+00/1.0086E+00
f_{c_9}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
$f_{c_{10}}$	3.9726E+01/5.2782E+01(<)	4.4008E+01/6.4195E+01(<)	8.5905E+01/1.0381E+02(<)	6.0081E+01/6.7817E+01(<)	4.2145E+01/5.2571E+01(<)	3.8716E+01/5.6200E+01(<)	**3.4568E+01/4.9967E+01**
$f_{c_{11}}$	3.3296E−01/6.3656E−01(>)	7.2577E−02/5.1830E−01(>)	4.1908E−13/1.6549E−12(>)	**0/0**(>)	1.6514E+00/5.0293E−01(<)	8.1559E−01/9.0414E−01(<)	3.8257E−01/7.0458E−01
$f_{c_{12}}$	3.0628E+01/5.2461E+01(<)	5.7164E+01/5.9695E+01(<)	7.6218E+00/2.8438E+01(<)	3.2991E+01/5.3743E+01(<)	2.1349E+01/4.5929E+01(<)	5.9895E+00/2.3499E+01(<)	**2.4487E−01/1.8004E−01**
$f_{c_{13}}$	1.7109E+00/2.2716E+00(<)	3.4599E+00/2.7185E+00(<)	1.3989E+00/2.0892E+00(<)	1.8653E+00/2.4115E+00(<)	1.1766E+00/1.7414E+00(>)	**8.6804E−01/1.7056E+00**(>)	1.2263E+00/1.8088E+00
$f_{c_{14}}$	2.5361E−01/4.5440E−01(>)	7.8444E−01/3.9211E+00(<)	**8.1603E−02/2.7030E−01**(>)	1.3656E−01/3.4579E−01(>)	2.7741E−01/3.7433E−01(>)	2.2887E−01/5.5976E−01(>)	6.4546E−01/6.4878E−01
$f_{c_{15}}$	1.7760E−01/2.0050E−01(<)	2.7469E−01/2.0760E−01(<)	3.1955E−01/2.0114E−01(<)	1.3893E−01/1.9930E−01(<)	2.1824E−01/1.3237E−01(<)	1.5778E−01/1.9635E−01(<)	**9.2215E−02/1.6006E−01**
$f_{c_{16}}$	38554E-01/1.8179E−01(>)	3.4985E−01/2.4186E−01(>)	6.4697E−01/3.3068E−01(<)	**2.3555E−01/1.8324E−01**(>)	4.5039E−01/2.0038E−01(<)	3.5158E−01/1.6671E−01(>)	4.0588E−01/2.4258E−01
$f_{c_{17}}$	**1.3290E−01/1.5179E−01**(>)	1.4125E+00/4.7569E+00(<)	4.2828E+00/7.6945E+00(<)	1.9363E−01/3.0306E−01(>)	3.1078E−01/2.2623E−01(<)	1.5667E−01/1.8773E−01(>)	2.0352E−01/2.0979E−01
$f_{c_{18}}$	2.2008E−01/2.0193E−01(<)	2.5016E−01/1.9343E−01(<)	3.6028E−01/1.9595E−01(<)	2.6642E−01/2.1470E−01(<)	2.9795E−01/1.9912E−01(<)	1.9642E−01/2.0415E−01(<)	**1.8619E−01/2.0613E−01**
$f_{c_{19}}$	**8.3787E−03/1.0307E−02**(>)	1.3548E−02/2.6891E−02(>)	1.2917E−02/1.7122E−02(>)	9.9239E−03/1.1942E−02(>)	2.8761E−02/9.7825E−03(<)	1.2855E−02/9.8415E−03(>)	1.5661E−02/8.7487E−03
$f_{c_{20}}$	0/0(>)	3.8973E−01/2.7833E+00(<)	1.3892E+00/4.7377E+00(<)	0/0(>)	2.7507E−08/1.9644E−07(>)	0/0(>)	4.8968E−02/1.3055E−01
$f_{c_{21}}$	1.3529E+02/4.8777E+01(>)	1.8053E+02/4.2654E+01(<)	1.5072E+02/5.2239E+01(>)	1.4281E+02/5.1680E+01(>)	**1.2379E+02/4.2859E+01**(>)	1.4338E+02/5.0813E+01(>)	1.6331E+02/5.1369E+01
$f_{c_{22}}$	1.0000E+02/8.9223E−14(>)	1.0005E+02/1.2759E−01(>)	1.0001E+02/4.0101E−02(>)	1.0000E+02/1.4838E−13(>)	**9.8039E+01/1.4003E+01**(>)	9.8057E+01/1.4006E+01(>)	1.0007E+02/1.4158E−01
$f_{c_{23}}$	3.0371E+02/1.6178E+00(<)	3.0051E+02/1.1209E+00(>)	**2.9479E+02/4.2126E+01**(>)	3.0335E+02/1.7951E+00(<)	3.0268E+02/1.4888E+00(<)	3.0277E+02/1.5790E+00(<)	3.0231E+02/1.9137E+00
$f_{c_{24}}$	2.7794E+02/9.2961E+01(<)	2.8207E+02/9.0922E+01(<)	2.8393E+02/9.1751E+01(<)	2.8646E+02/8.1744E+01(<)	2.6641E+02/9.8827E+01(>)	**2.5608E+02/1.0533E+02**(>)	2.7717E+02/9.4094E+01
$f_{c_{25}}$	4.1305E+02/2.1634E+01(<)	4.2036E+02/2.3061E+01(<)	4.0417E+02/1.5787E+01(>)	4.1395E+02/2.1926E+01(>)	**4.0414E+02/1.5787E+01**(>)	4.0863E+02/1.9563E+01(<)	4.1217E+02/2.1288E+01
$f_{c_{26}}$	3.0000E+02/0(≈)	3.0000E+02/0(≈)	3.0000E+02/0(≈)	3.0000E+02/0(≈)	3.0000E+02/0(≈)	30000E+02/0(≈)	3.0000E+02/0
$f_{c_{27}}$	3.9288E+02/1.8896E+00(<)	3.9266E+02/2.0809E+00(<)	3.9193E+02/2.0505E+00(<)	3.9313E+02/1.6469E+00(<)	3.9200E+02/2.1395E+00(<)	**3.9078E+02/2.4157E+00**(>)	3.9115E+02/2.5313E+00
$f_{c_{28}}$	3.0556E+02/3.9732E+01(>)	3.7398E+02/1.2788E+02(<)	3.4793E+02/1.0718E+02(<)	3.0556E+02/3.9732E+01(>)	**3.0000E+02/0**(>)	3.0556E+02/3.9732E+01(>)	3.4451E+02/1.0422E+02
$f_{c_{29}}$	2.3478E+02/3.7851E+00(<)	2.3462E+02/3.9792E+00(<)	2.3803E+02/4.0478E+00(<)	**2.3200E+02/2.7401E+00**(>)	2.3943E+02/3.9195E+00(<)	2.3590E+02/3.6657E+00(<)	2.3426E+02/3.2061E+00
$f_{c_{30}}$	3.9759E+02/1.1523E+01(<)	3.9687E+02/2.1541E+01(<)	3.9451E+02/2.7928E−02(≈)	3.9946E+02/1.4869E+01(<)	3.9451E+02/6.4970E−03(≈)	3.9450E+02/3.2589E−03(≈)	3.9450E+02/3.2874E−03
>/≈/<	12/6/12	9/6/15	11/7/12	12/6/12	9/9/12	11/7/12	−/−/−

Table 10
Mean and standard deviation (Mean/Std) of the 51-run fitness errors $\Delta f = f - f^*$ of the seven DE variants on 30*D* optimization under f_{c_1}–$f_{c_{28}}$ of our test suite is given below. The overall performance behind "Mean/Std" is measure under Wilcoxon's signed rank test under the significant level $\alpha = 0.05$.

30D	LSHADE	iLSHADE	jSO	LPalmDE	HARD-DE	DE-NPC	DE-EXP
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{c_1}	0/0(<)	0/0(<)	1.9505E−015/4.9939E−015(<)	1.1146E−015/3.8586E−015(<)	6.6875E−015/7.1637E−015(<)	5.5729E−016/2.7859E−015(<)	**0/0**
f_{c_2}	1.6719E−015/6.7540E−015(<)	9.4739E−015/1.5739E−014(<)	1.8948E−014/2.6459E−014(<)	2.7864E−015/8.5358E−015(<)	5.0156E−015/3.5819E−014(<)	2.2292E−015/7.7172E−015(<)	**5.5729E−016/3.9798E−015**
f_{c_3}	3.3437E−015/1.3508E−014(<)	5.5729E−015/1.7072E−014(<)	3.3437E−014/2.8254E−014(<)	1.1146E−014/2.2793E−014(<)	1.5604E−014/2.5620E−014(<)	1.1146E−014/2.2793E−014(<)	**0/0**
f_{c_4}	5.8351E+001/1.0183E+000(>)	5.7681E+001/8.3031E+000(>)	5.6265E+001/1.1480E+001(>)	**5.0138E+001/2.1924E+001**(>)	5.9215E+001/1.8078E+000(>)	5.6810E+001/1.1711E+001(>)	5.8997E+001/1.5085E+000
f_{c_5}	**7.2931E+000/1.4709E+000**(>)	7.7579E+000/1.6820E+000(>)	9.5923E+000/2.5129E+000(>)	1.1492E+001/3.5611E+000(<)	1.2847E+001/1.8534E+000(<)	9.4372E+000/1.9428E+000(>)	9.8363E+000/2.7475E+000
f_{c_6}	5.3677E−009/2.6833E−008(<)	1.2077E−008/5.0208E−008(<)	1.1247E−007/6.5329E−007(<)	7.8083E−008/5.5751E−007(<)	4.0946E−008/1.7106E−007(<)	**1.1369E−013/0**(>)	2.6974E−009/1.9164E−008
f_{c_7}	**3.7073E+001/1.3195E+000**(>)	3.7892E+001/1.4661E+000(>)	3.9084E+001/2.3758E+000(>)	4.1199E+001/3.1368E+000(>)	4.2453E+001/2.2784E+000(<)	3.9799E+001/2.3534E+000(>)	4.1855E+001/3.1955E+000
f_{c_8}	8.0941E+000/1.4854E+000(>)	**7.4480E+000/1.7729E+000**(>)	9.4778E+000/2.8537E+000(>)	1.2700E+001/3.1374E+000(<)	1.3448E+001/2.3598E+000(<)	1.0574E+001/2.4132E+000(≈)	1.0575E+001/3.1178E+000
f_{c_9}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
$f_{c_{10}}$	1.4373E+003/1.9107E+002(<)	1.7368E+003/3.1329E+002(<)	1.8220E+003/4.0675E+002(<)	1.6101E+003/2.9307E+002(<)	1.4927E+003/2.2856E+002(<)	1.4815E+003/2.0724E+002(<)	**1.4213E+003/2.1360E+002**
$f_{c_{11}}$	1.0251E+001/1.6235E+001(<)	1.4194E+001/2.2473E+001(<)	5.9899E+000/1.4077E+001(<)	9.5296E+000/1.6492E+001(<)	7.6263E+000/1.1703E+001(<)	5.6159E+000/1.1595E+001(<)	**3.6370E+000/4.3566E+000**
$f_{c_{12}}$	1.0056E+003/4.1683E+002(<)	8.8011E+002/3.9093E+002(<)	**3.0842E+002/1.5982E+002**(>)	1.0802E+003/3.4218E+002(<)	5.2112E+002/2.7212E+002(<)	4.0979E+002/2.0259E+002(<)	3.8780E+002/2.0304E+002
$f_{c_{13}}$	**1.3787E+001/5.8058E+000**(>)	1.8498E+001/8.4564E+000(<)	1.5875E+001/7.0381E+000(<)	1.5447E+001/6.7461E+000(<)	1.4985E+001/6.5261E+000(<)	1.2360E+001/6.1011E+000(>)	1.4576E+001/6.8877E+000
$f_{c_{14}}$	2.1943E+001/2.8573E+000(<)	2.1786E+001/1.0796E+000(<)	2.1724E+001/1.0776E+000(<)	2.0486E+001/6.0206E+000(<)	2.2465E+001/4.3292E+000(<)	2.1589E+001/4.5177E+000(<)	**1.4666E+001/7.5224E+000**
$f_{c_{15}}$	2.8371E+000/1.3450E+000(>)	3.6957E+000/1.9741E+000(<)	**1.3478E+000/9.9484E−001**(>)	3.0487E+000/1.8599E+000(>)	2.3090E+000/1.1690E+000(>)	1.7213E+000/9.4365E−001(>)	3.4018E+000/1.8734E+000
$f_{c_{16}}$	1.0827E+002/9.5197E+001(<)	**4.8754E+001/6.9117E+001**(>)	7.0278E+001/8.3611E+001(>)	1.3114E+002/1.2189E+002(<)	1.8818E+002/8.5053E+001(<)	1.1853E+002/9.0426E+001(<)	8.4142E+001/8.6376E+001
$f_{c_{17}}$	3.1251E+001/5.5560E+000(<)	3.8068E+001/5.2253E+000(<)	4.3132E+001/9.1891E+000(<)	3.3916E+001/1.2487E+001(<)	3.5672E+001/8.2123E+000(<)	3.2493E+001/735238E+000(<)	**2.6622E+001/8.8600E+000**
$f_{c_{18}}$	2.2055E+001/1.0562E+000(<)	2.1435E+001/8.0543E−001(<)	2.0739E+001/2.3340E−001(>)	2.2496E+001/1.4841E+000(<)	**2.0673E+001/4.6518E−001**(>)	2.0766E+001/5.1370E−001(>)	2.1049E+001/1.7001E+000
$f_{c_{19}}$	5.2897E+000/1.5221E+000(<)	8.5276E+000/1.9607E+000(<)	6.8840E+000/2.1626E+000(<)	5.3748E+000/1.9060E+000(<)	7.0429E+000/1.5852E+000(<)	**4.5569E+000/1.4809E+000**(>)	5.2251E+000/1.3742E+000
$f_{c_{20}}$	3.9241E+001/7.7126E+000(<)	4.7966E+001/1.8459E+001(<)	4.7584E+001/1.0330E+001(<)	3.5851E+001/2.8843E+001(<)	4.5613E+001/1.7334E+001(<)	3.9078E+001/8.2006E+000(<)	**2.8851E+000/3.08606E+001**
$f_{c_{21}}$	**2.0752E+002/1.4667E+000**(>)	2.0824E+001/1.6415E+000(>)	2.1077E+002/2.9848E+000(<)	2.1119E+002/2.9636E+000(<)	2.1276E+002/1.6281E+000(<)	2.1050E+002/1.8023E+000(≈)	2.1050E+002/3.0713E+000
$f_{c_{22}}$	1.0000E+002/1.0047E−013(≈)	1.0000E+002/1.0047E−013(≈)	1.0000E+002/1.0047E−013(≈)	1.0000E+002/1.0047E−013(≈)	1.0000E+002/1.3713E−013(≈)	1.0000E+002/1.0047E−013(≈)	1.0000E+002/1.0047E−013
$f_{c_{23}}$	**3.4489E+002/3.4048E+000**(>)	3.5058E+002/4.4819E+000(<)	3.5299E+002/5.2180E+000(<)	3.4979E+002/4.7944E+000(<)	3.4844E+002/4.0376E+000(<)	3.4597E+002/3.2373E+000(≈)	3.4596E+002/4.3531E+000
$f_{c_{24}}$	4.2119E+002/2.1633E+000(>)	4.2531E+002/2.6884E+000(<)	4.2604E+002/3.6101E+000(<)	4.2279E+002/2.9262E+000(<)	4.2162E+002/3.3418E+000(>)	**4.2092E+002/2.0321E+000**(>)	4.2230E+002/3.1373E+000
$f_{c_{25}}$	3.8675E+002/2.6782E−002(<)	3.8673E+002/2.3683E−002(<)	3.8670E+002/9.6970E−003(≈)	3.8677E+002/2.4526E−002(<)	3.8673E+002/2.1175E−002(<)	3.8673E+002/2.0464E−002(<)	3.8670E+002/1.3131E−002
$f_{c_{26}}$	8.8031E+002/3.0735E+001(<)	9.0822E+002/4.6641E+001(<)	9.0537E+002/4.4540E+001(<)	9.2519E+002/6.5520E+001(<)	9.1477E+002/4.4632E+001(<)	**8.3890E+002/3.5311E+001**(>)	8.5964E+002/4.7641E+001
$f_{c_{27}}$	5.0545E+002/6.5924E+000(<)	5.0416E+002/6.8481E+000(<)	5.0242E+002/6.5970E+000(<)	5.0842E+002/6.0369E+000(<)	**4.9844E+002/7.2804E+000**(>)	5.0040E+002/7.2873E+000(≈)	5.0038E+002/7.9802E+000
$f_{c_{28}}$	3.1641E+002/3.8453E+001(>)	3.1886E+002/4.1194E+001(>)	3.1257E+002/3.4812E+001(>)	3.3970E+002/5.5071E+001(<)	**3.1096E+002/3.3615E+001**(>)	3.2130E+002/4.3624E+001(>)	3.2717E+002/4.6976E+001
$f_{c_{29}}$	4.3257E+002/7.3701E+000(<)	4.4648E+002/1.0443E+001(<)	4.4357E+002/1.4053E+001(<)	4.4932E+002/9.6137E+000(<)	4.4093E+002/1.0124E+001(<)	4.2988E+002/1.0131E+001(<)	**4.2654E+002/1.6083E+001**
$f_{c_{30}}$	2.0041E+003/5.2556E+001(<)	2.0296E+003/5.8781E+001(<)	**1.9871E+003/2.7501E+001**(>)	2.0723E+003/7.4082E+001(<)	1.9971E+003/3.3366E+001(<)	1.9996E+003/3.7968E+001(<)	1.9989E+003/3.7312E+001
>/≈/<	10/2/18	7/2/21	10/3/17	3/2/25	6/2/22	11/6/13	−/−/−

comparison with 10*D* optimization under our test suite because these state-of-the-art DE variants perform relatively better on 10*D* optimization rather than higher dimensional optimization such as 30*D* and 50*D* optimization.

As a result, the novel DE-EXP algorithm employing exponential crossover can obtain even better performance in comparison with the state-of-the-art DE variants employing binomial crossover. This also proves our assertion that the DE variant with exponential crossover can obtain competitive performance with the ones employing binomial crossover on numerical optimization after discovering the proper crossover rate *Cr* and its corresponding parameter control.

4.3. Convergence analysis

The novel DE-EXP algorithm is also examined from the convergence speed perspective of view, and the convergence curve of the median value among the total 51-run results on a certain benchmark is taken into comparison. Figs. 7–10 present the comparison of convergence speed of these state-of-the-art DE variants including LSHADE, iLSHADE, jSO, LPalmDE, HARD-DE, DE-NPC and our DE-EXP.

From these convergence figures, it can be seen that the proposed DE-EXP algorithm can find the global best within the maximum number of function evaluations on $f_{a_1}, f_{a_2}, f_{a_4}$ and $f_{a_{10}}$, and it can obtain better or similar better performance on f_{a_2}–f_{a_8}, $f_{a_{12}}, f_{a_{14}}, f_{a_{17}}$ $f_{a_{20}}$–$f_{a_{22}}$, $f_{a_{24}}$, and $f_{a_{26}}$–$f_{a_{28}}$.

Moreover, It can be found that LSHADE performed best or tie-best on f_{a_5}, $f_{a_{13}}$, $f_{a_{21}}$, $f_{a_{24}}$ and $f_{a_{26}}$–$f_{a_{28}}$, iLSHADE performed best or tie-best on f_{a_5}, $f_{a_{15}}$, $f_{a_{19}}$, $f_{a_{21}}$, $f_{a_{23}}$–$f_{a_{24}}$, and $f_{a_{26}}$–$f_{a_{28}}$; jSO performed relative worse than the other algorithms in the comparison from convergence speed perspective of view; LPalmDE performed competitive on f_{a_1}, f_{a_5}, f_{a_9}–$f_{a_{11}}$, $f_{a_{14}}$, $f_{a_{16}}$–$f_{a_{18}}$, $f_{a_{20}}$–$f_{a_{22}}$, $f_{a_{24}}$ and $f_{a_{26}}$–$f_{a_{28}}$; HARD-DE was competitive on $f_{a_{17}}$, $f_{a_{21}}$, $f_{a_{22}}$ and $f_{a_{24}}$–$f_{a_{28}}$; DE-NPC was competitive on f_{a_3}, f_{a_5}–f_{a_6}, $f_{a_{11}}$, $f_{a_{21}}$

Table 11
Mean and standard deviation (Mean/Std) of the 51-run fitness errors $\Delta f = f - f^*$ of the seven DE variants on 50D optimization under f_{c_1}–$f_{c_{28}}$ of our test suite is given below. The overall performance behind "Mean/Std" is measure under Wilcoxon's signed rank test under the significant level $\alpha = 0.05$.

50D	LSHADE	iLSHADE	jSO	LPalmDE	HARD-DE	DE-NPC	Our DE-EXP
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{c_1}	2.0898E−014/7.1637E−015(>)	1.9226E−014/6.8587E−015(>)	2.9815E−014/9.1016E−015(>)	2.2013E−014/7.1416E−015(>)	3.3159E−014/8.8367E−015(>)	**1.7555E−014/6.0880E−015**(>)	9.3067E−014/2.9716E−014
f_{c_2}	1.3375E−013/2.0818E−013(>)	4.4583E−014/3.1779E−014(>)	2.7642E−012/1.8686E−011(>)	7.8020E−014/2.2864E−013(>)	6.4171E−001/4.5827E+000(<)	**2.7864E−014/2.3773E−014**(>)	1.6271E−011/1.1561E−011
f_{c_3}	1.7722E−013/5.6441E−014(<)	1.6273E−013/5.2122E−014(<)	2.7307E−013/7.5428E−014(<)	1.5270E−013/4.7545E−014(<)	2.8310E−013/6.7727E−014(<)	2.1288E−013/6.2106E−014(<)	**6.6875E−014/2.4662E−014**
f_{c_4}	8.2638E+001/4.3569E+001(<)	7.1712E+001/5.0120E+001(<)	7.7991E+001/5.1819E+001(<)	7.2917E+001/4.8941E+001(<)	7.0579E+001/4.5615E+001(<)	7.4926E+001/4.7323E+001(<)	**6.8243E+001/5.3481E+001**
f_{c_5}	1.3305E+001/2.1177E+000(>)	**1.2228E+001/3.0667E+000**(>)	1.5600E+001/3.7475E+000(>)	2.2752E+001/4.2620E+000(<)	2.7215E+001/2.3598E+000(<)	1.7067E+001/2.3998E+000(>)	1.7994E+001/4.0895E+000
f_{c_6}	9.9025E−008/2.8913E−007(>)	2.6035E−008/6.3691E−008(>)	3.7956E−007/6.4809E−007(<)	3.9656E−004/2.0272E−003(<)	1.7497E−007/2.8302E−007(>)	**1.4635E−008/3.9924E−008**(>)	3.1773E−007/5.0228E−007
f_{c_7}	6.3941E+001/1.7283E+000(>)	**6.3161E+001/2.0233E+000**(>)	6.5593E+001/4.0246E+000(>)	7.1086E+001/4.2874E+000(<)	7.3628E+001/3.4112E+000(<)	6.5985E+001/2.6065E+000(>)	7.0613E+001/5.1492E+000
f_{c_8}	**1.3419E+001/2.0742E+000**(>)	1.2078E+001/2.4784E+000(>)	1.4978E+001/3.5213E+000(>)	2.3705E+001/4.9813E+000(<)	2.5878E+001/3.3379E+000(<)	1.7587E+001/2.8736E+000(<)	1.7269E+001/3.8021E+000
f_{c_9}	2.8979E−014/5.0038E−014(>)	**1.7833E−014/4.1756E−014**(>)	6.0187E−014/5.7310E−014(>)	1.7555E−003/1.2536E−002(≈)	9.5854E−014/4.1756E−014(>)	3.7896E−014/5.4126E−014(>)	1.7555E−003/1.2536E−002
$f_{c_{10}}$	3.1112E+003/3.2073E+002(<)	3.6085E+003/3.4604E+002(<)	4.0027E+003/7.3271E+002(<)	3.3905E+003/3.8563E+002(<)	3.1984E+003/2.8815E+002(<)	3.0568E+003/2.4886E+002(<)	**2.9039E+003/2.6677E+002**
$f_{c_{11}}$	5.4847E+001/8.5566E+000(<)	3.6370E+001/6.5888E+000(<)	**2.5668E+001/3.4890E+000**(>)	7.0368E+001/1.2183E+001(<)	4.0976E+001/6.7729E+000(<)	3.4459E+001/4.8629E+000(<)	3.1012E+001/4.1898E+000
$f_{c_{12}}$	2.2602E+003/5.5373E+002(<)	2.0529E+003/5.2684E+002(<)	2.0017E+003/4.9303E+002(<)	2.2057E+003/4.2758E+002(<)	1.9738E+003/4.8212E+002(<)	1.8936E+003/4.9935E+002(<)	**1.8267E+003/5.3295E+002**
$f_{c_{13}}$	6.9853E+001/3.3457E+001(>)	5.2260E+001/2.6905E+001(>)	**3.2973E+001/1.7082E+001**(>)	7.2190E+001/3.2991E+001(>)	4.3437E+001/1.9865E+001(>)	4.2658E+001/3.0962E+001(>)	7.7046E+001/3.4306E+001
$f_{c_{14}}$	3.1217E+001/3.8058E+000(<)	2.5625E+001/1.6382E+000(<)	2.4545E+001/1.4663E+000(<)	3.1219E+001/3.9469E+000(<)	3.0648E+001/3.0766E+000(<)	2.6691E+001/2.0187E+000(<)	**2.4366E+001/2.1034E+000**
$f_{c_{15}}$	4.6395E+001/1.5035E+001(<)	2.9510E+001/6.2238E+000(<)	**2.3336E+001/3.2839E+000**(>)	4.9879E+001/1.6159E+001(<)	3.1060E+001/6.7132E+000(<)	2.6512E+001/3.2787E+000(>)	2.8059E+001/4.8086E+000
$f_{c_{16}}$	3.8406E+002/1.1932E+002(<)	2.9663E+002/1.1955E+002(<)	4.3460E+002/1.5915E+002(<)	4.3267E+002/1.1646E+002(<)	4.2674E+002/1.1130E+002(<)	3.8538E+002/1.2251E+002(<)	**2.8889E+002/1.1466E+002**
$f_{c_{17}}$	2.5879E+002/6.2217E+001(>)	**2.4654E+002/8.7589E+001**(>)	2.8770E+002/8.8839E+001(<)	3.2102E+002/1.1479E+002(<)	3.6945E+002/8.9655E+001(<)	2.8917E+002/6.9427E+001(<)	2.7499E+002/1.1420E+002
$f_{c_{18}}$	5.0283E+001/1.7870E+001(<)	3.1908E+001/5.9069E+000(<)	**2.4434E+001/2.0326E+000**(>)	5.8167E+001/3.8742+001(<)	2.8642E+001/3.3892E+000(<)	2.5760E+001/3.2095+000(<)	2.5733E+001/3.2023E+000
$f_{c_{19}}$	3.4233E+001/1.1328E+001(<)	1.9326E+001/3.1872E+000(<)	1.4134E+001/2.4317E+000(<)	3.9049E+001/1.3702E+001(<)	2.0662E+001/3.7582E+000(<)	1.4570E+001/1.9068E+000(<)	**1.1376E+001/2.6790E+000**
$f_{c_{20}}$	1.5728E+002/5.4246E+001(>)	**1.3378E+002/5.3771E+001**(>)	1.4051E+002/6.1084E+001(>)	1.5129E+002/1.0633E+002(>)	2.3284E+002/8.6946E+001(<)	1.7282E+002/6.5284E+001(>)	2.2646E+002/1.4977E+002
$f_{c_{21}}$	2.1548E+002/2.4791E+000(>)	**2.1431E+002/2.8921E+000**(>)	2.1846E+002/4.6772E+000(<)	2.2283E+002/4.8226E+000(<)	2.2799E+002/3.2311E+000(<)	2.1784E+002/2.9933E+000(≈)	2.1783E+002/5.1756E+000
$f_{c_{22}}$	6.9168E+002/1.3627E+003(<)	1.1178E+003/1.7767E+003(<)	4.5121E+002/1.2167E+003(<)	6.9399E+002/1.3919E+003(<)	1.0317E+002/1.0685E+001(>)	5.3251E+002/1.1734E+003(<)	**4.3660E+002/1.0010E+003**
$f_{c_{23}}$	4.2842E+002/5.5681E+000(<)	4.3981E+002/8.1671E+000(<)	4.3986E+002/9.5792E+000(<)	4.3432E+002/8.7671E+000(<)	4.2921E+002/5.9307E+000(<)	4.2321E+002/5.7407E+000(<)	**4.2313E+002/9.4145E+000**
$f_{c_{24}}$	5.0496E+002/4.9801E+000(<)	5.5528E+002/5.6708E+000(<)	5.0673E+002/5.8449E+000(<)	5.0500E+002/5.1297E+000(<)	**4.9861E+002/4.4250E+000**(>)	4.9992E+002/5.2312E+000(>)	5.0141E+002/6.4226E+000
$f_{c_{25}}$	4.8173E+002/5.7965E+000(>)	4.8358E+002/1.3066E+001(<)	**4.8072E+002/2.2707E+000**(>)	4.9948E+002/2.8711E+001(<)	4.9061E+002/1.8825E+001(<)	4.8609E+002/1.6144E+001(<)	4.8199E+002/1.1637E+001
$f_{c_{26}}$	1.0664E+003/5.5955E+001(<)	1.0657E+003/7.6915E+001(<)	1.0793E+003/8.5906E+001(<)	1.1480E+003/6.7297E+001(<)	1.1420E+003/6.8646E+001(<)	1.0333E+003/7.1134E+001(<)	**1.0268E+003/7.3711E+001**
$f_{c_{27}}$	5.4051E+002/1.0026E+001(<)	5.2718E+002/6.1334E+000(<)	**5.1806E+002/6.2981E+000**(>)	5.4281E+002/1.2234E+001(<)	5.2262E+002/1.0358E+001(>)	5.2657E+002/6.9778E+000(<)	5.2513E+002/9.2319E+000
$f_{c_{28}}$	4.9524E+002/2.1499E+001(<)	4.7896E+002/2.4279E+001(<)	**4.6076E+002/9.5757E+000**(>)	4.9524E+002/2.1499E+001(<)	4.7226E+002/2.2015E+001(<)	4.7130E+002/2.1499E+001(<)	4.6747E+002/1.8806E+001
$f_{c_{29}}$	3.4710E+002/1.1296E+001(<)	3.8113E+002/1.4711E+001(<)	3.9201E+002/1.9161E+001(<)	3.5293E+002/1.7288E+001(<)	3.7144E+002/1.2998E+001(<)	3.4930E+002/7.3086E+000(<)	**3.4597E+002/1.2278E+001**
$f_{c_{30}}$	6.1686E+005/3.2219E+004(<)	6.1786E+005/3.5173E+004(<)	**6.0574E+005/3.0529E+004**(>)	6.2874E+005/3.2086E+004(<)	6.1055E+005/3.1183E+004(<)	6.1538E+005/3.1082E+004(<)	6.1360E+005/3.3003E+004
>/≈/<	12/0/18	11/0/19	15/0/15	4/1/25	8/0/22	10/1/19	−/−/−

Table 12
Summary of the comparison results under our 88-benchmark test suite on 10D, 30D, and 50D respectively.

A given algorithm versus our DE-EXP algorithm			
>/≈/<	$D = 10$	$D = 30$	$D = 50$
LSHADE	29/22/37	18/15/55	22/4/62
iLSHADE	24/20/44	20/11/57	29/3/56
jSO	27/20/41	15/15/55	27/6/55
LPALMDE	29/23/36	15/10/63	19/5/64
HARD-DE	25/21/42	12/13/63	25/4/59
DE-NPC	29/26/33	12/19/57	32/6/50

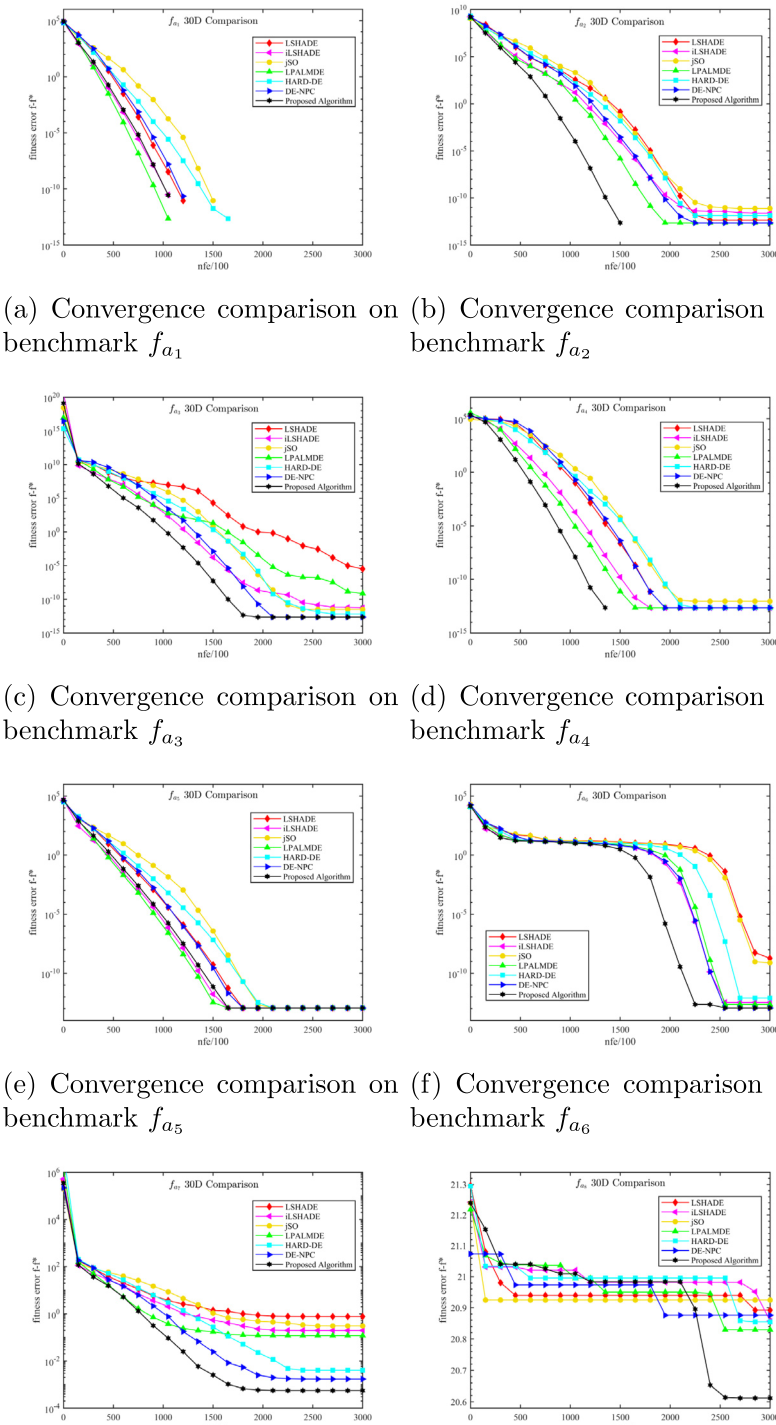
and $f_{a_{24}}$–$f_{a_{28}}$. To sum up, the proposed DE-EXP algorithm is still excellent from the convergence speed perspective of view.

4.4. Component analysis

In this part, we examine the effectiveness of each component of our DE-EXP algorithm. There are four components, the automatically generated crossover rate Cr, the adaptation of scale factor F, the reduction of population size ps and the trial vector generation strategy. In order to evaluate the effectiveness of the automatically generated crossover rate Cr, we made a controlled experiment by using the fitness-improvement based adaptation of Cr (the commonly used adaptation in the recent winner DE variants) instead of our default automatically generated crossover rate Cr. The comparison results of the 88 benchmarks from CEC2013, CEC2014 and CEC2017 test suites on 30D optimization are given in Table 13. From the results, we can see that the automatically generate Cr is more effective in our DE-EXP algorithm than the adaptations in the recent winner DE variants. In order to evaluate the effectiveness of the fitness-independent adaptation of scale factor F, we made a controlled experiment by using the fitness-improvement based adaptation of F (the commonly used adaptation in the recent winner DE variants) instead of our default fitness-independent adaptation of F. The comparison results of the 88 benchmarks from CEC2013, CEC2014

Table 13
Comparison between DE-EXP with adaptive Cr and DE-EXP algorithm with automatically generated Cr (the default version) under the 88 benchmarks on 30D optimization with the total number of function evaluation equaling to $nfe_{max} = 10000 \times D$.

	CEC2013 30D optimization		CEC2014 30D optimization		CEC2017 30D optimization	
No.:	DE-EXP with adaptive Cr	DE-EXP (default)	DE-EXP with adaptive Cr	DE-EXP (default)	DE-EXP with adaptive Cr	DE-EXP (default)
f_1	*0/0(≈)*	*0/0*	1.0867E−14/6.0880E−15(<)	**0/0**	*0/0(≈)*	*0/0*
f_2	1.5604E−13/1.2446E−13(<)	**0/0**	5.5729E−16/3.9798E−15(<)	**0/0**	1.6719E−15/6.7540E−15(<)	**5.5729E−16/3.9798E−15**
f_3	1.0025E−08/7.1588E−08(<)	**3.4661E−09/2.4751E−08**	*0/0(≈)*	*0/0*	4.4583E−15/1.5434E−14(<)	**0/0**
f_4	3.5666E−14/8.3512E−14(<)	**0/0**	3.2323E−14/2.8433E−14(<)	**1.2260E−14/2.3612E−14**	**5.7492E+01/7.6420E+00**(>)	5.8997E+01/1.5085E+00
f_5	**9.8083E−14/3.9511E−14**(>)	1.1369E−13/2.2737E−14	2.0057E+01/1.4518E−02(<)	**2.0048E+01/1.2009E−02**	2.5643E+01/3.6809E+00(<)	**9.8363E+00/2.7475E+00**
f_6	1.9839E−13/1.1559E−13(<)	**1.4267E−13/1.0867E−13**	8.2089E+00/1.1030E+00(<)	**0/0**	**1.6900E−09/3.0756E−09**(>)	2.6974E−09/1.9164E−08
f_7	5.6201E−03/1.8007E−02(<)	**4.8007E−03/1.1567E−02**	*0/0(≈)*	*0/0*	5.5075E+01/3.4768E+00(<)	**4.1855E+01/3.1955E+00**
f_8	2.0788E+01/1.7474E−01(<)	**2.0604E+01/1.9914E−01**	**1.5604E−14/3.9511E−14**(>)	3.2323E−13/6.5791E−14	2.7300E+01/2.8211E+00(<)	**1.0575E+01/3.1178E+00**
f_9	2.4819E+01/1.6393E+00(<)	**2.3582E+01/5.2220E+00**	2.6137E+01/3.7427E+00(<)	**1.1361E+01/3.1635E+00**	*0/0(≈)*	*0/0*
f_{10}	*0/0(≈)*	*0/0*	**4.0822E−04/2.9153E−03**(>)	1.2247E−03/4.9474E−03	1.4955E+03/2.1683E+02(<)	**1.4213E+03/2.1360E+02**
f_{11}	**1.2260E−14/2.3612E−14**(>)	1.6719E−13/2.6447E−14	1.2441E+03/2.6240E+02(<)	**1.2190E+03/1.9731E+02**	5.1421E+00/2.5691E+00(<)	**3.6370E+00/4.3566E+00**
f_{12}	1.7411E+01/2.4834E+00(<)	**6.7762E+00/3.1934E+00**	**1.1871E−01/1.8288E−02**(>)	1.2480E−01/2.1567E−02	**3.6306E+02/1.9835E+02**(>)	3.8780E+02/2.0304E+02
f_{13}	3.4495E+01/7.6789E+00(<)	**8.2574E+00/5.9276E+00**	2.0356E−01/2.9154E−02(<)	**1.1968E−01/2.0558E−02**	1.6820E+01/7.1190E+00(<)	**1.4576E+01/6.8877E+00**
f_{14}	5.5926E−02/2.8694E−02(<)	**1.8370E−02/2.0672E−02**	**1.7326E−01/2.0669E−02**(>)	1.7712E−01/2.4888E−02	1.9520E+01/5.9568E+00(<)	**1.4666E+01/7.5224E+00**
f_{15}	2.6088E+03/2.0406E+02(<)	**2.5812E+03/3.0053E+02**	2.3864E+00/4.0034E−01(<)	**2.0540E+00/2.0895E−01**	6.7045E+00/1.0429E+00(<)	**3.4018E+00/1.8734E+00**
f_{16}	6.5104E−01/1.2841E−01(<)	**5.4322E−01/2.4416E−01**	8.8736E+00/3.5527E−01(<)	**8.5457E+00/4.7679E−01**	2.7637E+02/9.7082E+01(<)	**8.4142E+01/8.6376E+01**
f_{17}	*3.0434E+01/3.5956E−04(≈)*	*3.0434E+01/1.9302E−12*	1.3425E+02/6.2862E+01(<)	**5.3267E+01/3.2671E+01**	3.5660E+01/1.0591E+01(<)	**2.6622E+01/8.8600E+00**
f_{18}	5.9156E+01/4.3807E+00(<)	**4.9702E+01/2.9301E+00**	3.8339E+00/9.2802E−01(<)	**1.6533E+00/1.0946E+00**	2.1085E+01/5.0787E+00(<)	**2.1049E+01/1.7001E+00**
f_{19}	1.1936E+00/1.0300E−01(<)	**1.1602E+00/9.1644E−02**	2.5025E+00/4.4087E−01(<)	**2.4130E+00/6.8816E−01**	7.6726E+00/1.1910E+00(<)	**5.2251E+00/1.3742E+00**
f_{20}	1.0193E+01/1.1931E+00(<)	**8.9848E+00/3.6837E−01**	6.1302E+00/1.2150E+00(<)	**2.9915E+00/1.0802E+00**	5.0293E+01/3.4892E+01(<)	**2.8851E+00/3.0861E+01**
f_{21}	**3.0538E+02/4.7105E+01**(>)	3.0648E+02/3.7329E+01	1.0997E+02/6.3323E+01(<)	**1.7459E+01/2.0246E+01**	2.2843E+02/3.9533E+00(<)	**2.1050E+02/3.0713E+00**
f_{22}	1.0608E+02/2.0908E−01(<)	**1.0599E+02/2.6425E−01**	8.5484E+01/5.5708E+01(<)	**4.7141E+01/4.7288E+01**	*1.0000E+02/6.3901E−14(≈)*	*1.0000E+02/1.0047E−13*
f_{23}	2.9079E+03/3.3103E+02(<)	**2.4599E+03/3.3073E+02**	*3.1524E+02/6.4311E−14(≈)*	*3.1524E+02/4.0186E−13*	3.7242E+02/4.2685E+00(<)	**3.4596E+02/4.3531E+00**
f_{24}	*2.0000E+02/4.7653E−03(≈)*	*2.0000E+02/6.0437E−03*	**2.0805E+02/1.0556E+01**(>)	2.1241E+02/1.0939E+01	4.3910E+02/4.7589E+00(<)	**4.2230E+02/3.1373E+00**
f_{25}	2.7654E+02/2.2992E+01(<)	**2.2815E+02/1.9585E+01**	*2.0260E+02/4.4938E−02(≈)*	*2.0260E+02/5.2683E−02*	*3.8670E+02/9.4627E−03(≈)*	*3.8670E+02/1.3131E−02*
f_{26}	*2.0000E+02/0(≈)*	*2.0000E+02/1.4171E−13*	1.0021E+02/2.9167E−02(<)	**1.0011E+02/1.7762E−02**	1.2021E+03/5.9456E+01(<)	**8.5964E+02/4.7641E+01**
f_{27}	3.6314E+02/1.7480E+02(<)	**3.0002E+02/5.3521E−02**	3.2954E+02/4.6220E+01(<)	**3.0000E+02/1.2531E−13**	**4.9454E+02/8.0503E+00**(>)	5.0038E+02/7.9802E+00
f_{28}	*3.0000E+02/2.1853E−13(≈)*	*3.0000E+02/1.6766E−13*	8.3116E+02/1.6397E+01(<)	**8.2148E+02/2.1032E+01**	**3.1075E+02/3.2987E+01**(>)	3.2717E+02/4.6976E+01
f_{29}	–	–	5.3687E+02/2.6867E+02(<)	**5.2335E+02/2.7633E+02**	4.4880E+02/1.5435E+01(<)	**4.2654E+02/1.6083E+01**
f_{30}	–	–	5.1406E+02/6.3553E+01(<)	**4.3544E+02/9.7758E+01**	**1.9892E+03/2.4824E+01**(>)	1.9989E+03/3.7312E+01
>/≈/<	3/6/19	−/−/−	5/4/21	−/−/−	6/4/20	−/−/−



(a) Convergence comparison on benchmark f_{a_1} (b) Convergence comparison on benchmark f_{a_2}

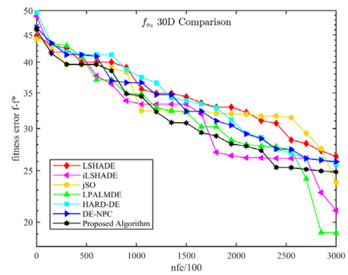
(c) Convergence comparison on benchmark f_{a_3} (d) Convergence comparison on benchmark f_{a_4}

(e) Convergence comparison on benchmark f_{a_5} (f) Convergence comparison on benchmark f_{a_6}

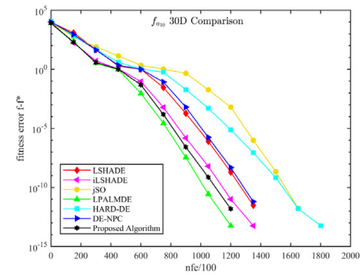
(g) Convergence comparison on benchmark f_{a_7} (h) Convergence comparison on benchmark f_{a_8}

Fig. 7. Convergence speed comparison by employing the median value of 51 runs obtained by each algorithm on 30*D* optimization. There are total 28 comparison figures and the first 8 figures are given here.

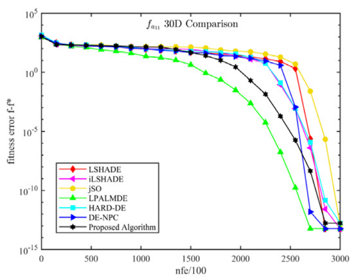
and CEC2017 test suites on 30D optimization are given in Table 14. From the results, we can see that the fitness-independent adaptation of scale factor *Cr* is more effective in our DE-EXP algorithm than the adaptations in the recent winner DE variants. In order to evaluate the effectiveness of the reduction of population size *ps*, we made a controlled experiment by using the fixed population instead of our default reduction of population size (also the commonly used reduction in the recent winner DE variants). The comparison results of the 88 benchmarks from CEC2013, CEC2014 and CEC2017 test suites on 30D optimization are given in Table 15. From the results, we can see that the reduction of population size *PS* is more effective in our DE-EXP algorithm than using fixed population size during the evolution. In order to evaluate the effectiveness of the mutation strategy,



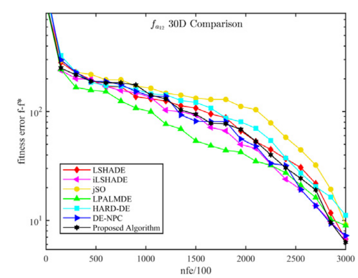
(a) Convergence comparison on benchmark f_{a_9}



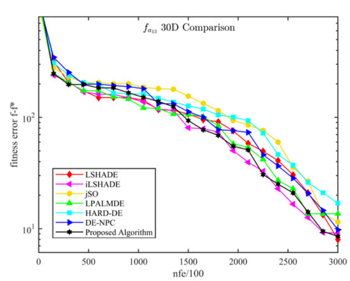
(b) Convergence comparison on benchmark $f_{a_{10}}$



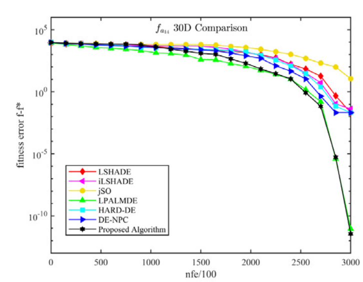
(c) Convergence comparison on benchmark $f_{a_{11}}$



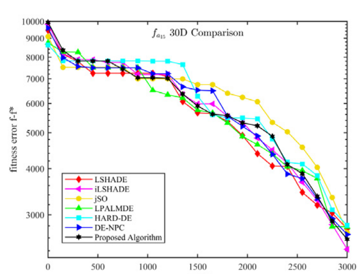
(d) Convergence comparison on benchmark $f_{a_{12}}$



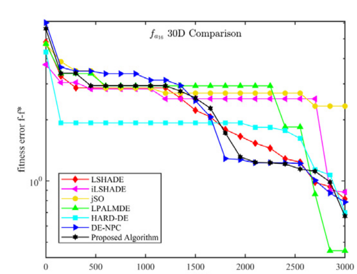
(e) Convergence comparison on benchmark $f_{a_{13}}$



(f) Convergence comparison on benchmark $f_{a_{14}}$



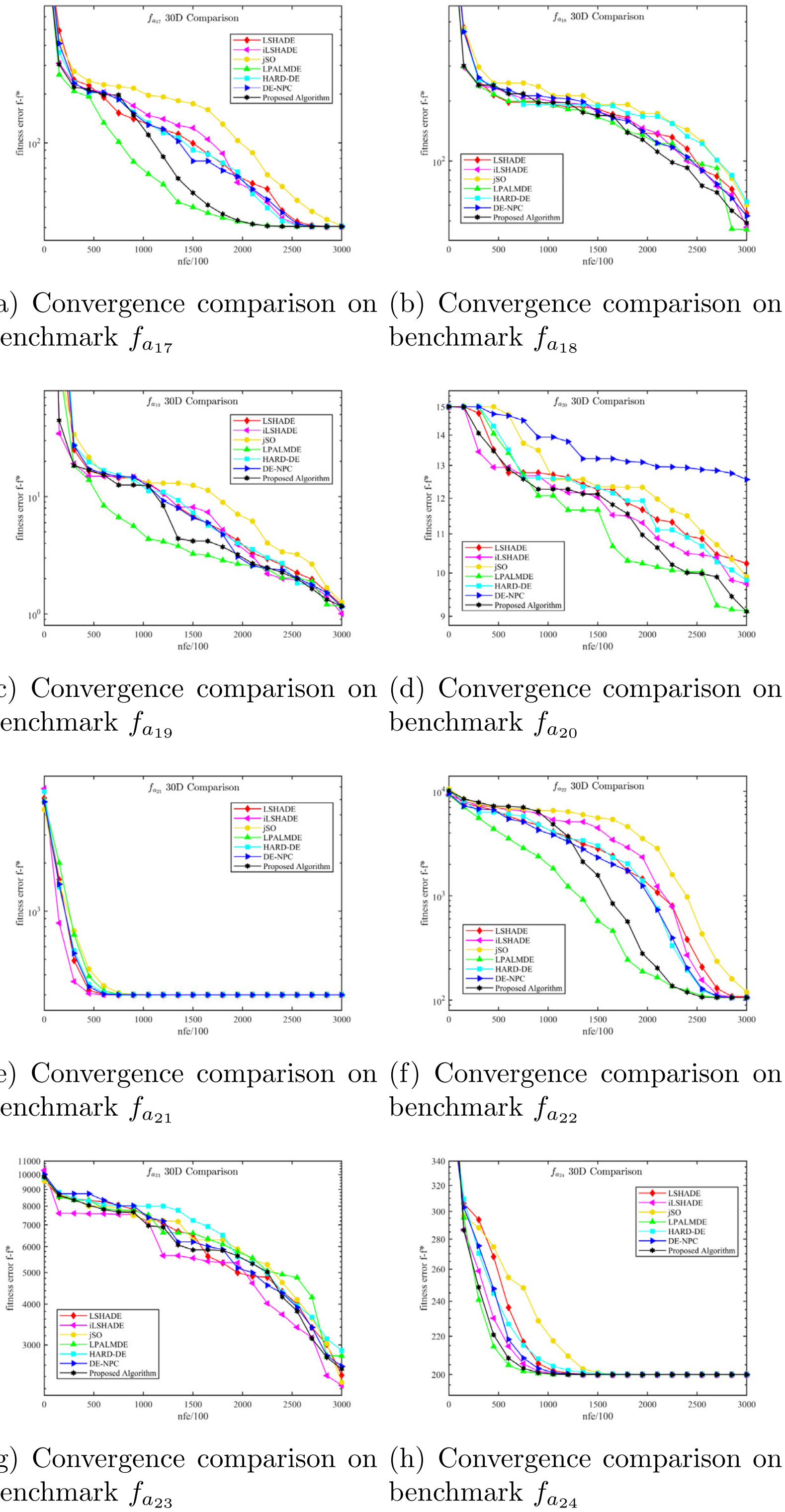
(g) Convergence comparison on benchmark $f_{a_{15}}$



(h) Convergence comparison on benchmark $f_{a_{16}}$

Fig. 8. As a continued part of Fig. 7, convergence comparisons on benchmarks f_{a_9}–$f_{a_{16}}$ are given here.

we also made a controlled experiment by using the trial vector generation strategy "DE/rand/1/exp" instead of our default strategy "DE/target-to-*p*best/1/exp". The comparison results of the 88 benchmarks from CEC2013, CEC2014 and CEC2017 test suites on 30D optimization are given in Table 16. From the results, we can see that our trial vector generation strategy is still very competitive. Generally, the DE-EXP algorithm is usually better than the DE-EXP algorithm with binomial crossover because the first component of the DE-EXP algorithm, the automatically generated crossover rate *Cr* in Section 3.1, is designed under the exponential crossover rather than binomial crossover. As one of the reviewers suggested, we also conducted extra experiment to examine the effectiveness of exponential crossover. The comparison results between DE-EXP algorithm with binomial crossover and the



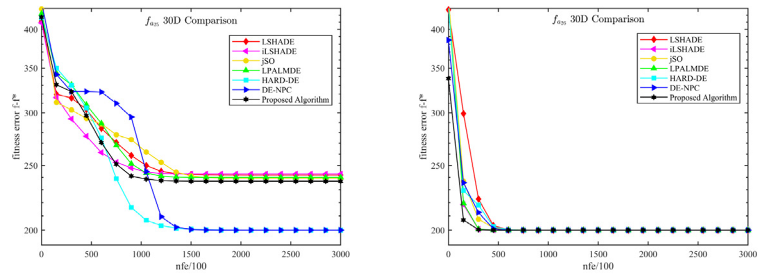
(a) Convergence comparison on benchmark $f_{a_{17}}$
(b) Convergence comparison on benchmark $f_{a_{18}}$
(c) Convergence comparison on benchmark $f_{a_{19}}$
(d) Convergence comparison on benchmark $f_{a_{20}}$
(e) Convergence comparison on benchmark $f_{a_{21}}$
(f) Convergence comparison on benchmark $f_{a_{22}}$
(g) Convergence comparison on benchmark $f_{a_{23}}$
(h) Convergence comparison on benchmark $f_{a_{24}}$

Fig. 9. As a continued part from Fig. 8, convergence comparisons on benchmarks $f_{a_{17}}$–$f_{a_{24}}$ are given here.

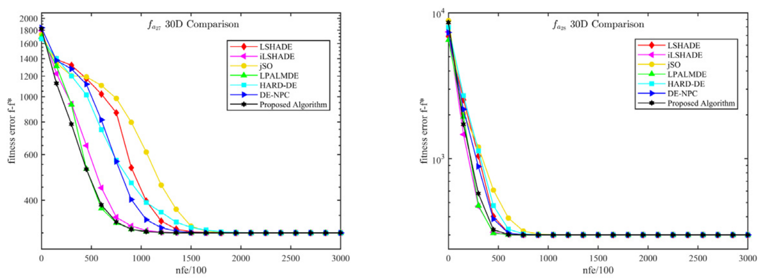
default DE-EXP algorithm are given in Table 17, and we can see from the results that the default DE-EXP algorithm performs better than employing binomial crossover. Moreover, if we change "the automatically generated crossover rate *Cr*" in the DE-EXP algorithm with binomial crossover to "the fitness-improvement based adaptation of *Cr* (the commonly used adaptation in the recent winner DE variants)", then the DE-EXP algorithm with binomial crossover is degraded into a LSHADE variant. By the way, the experiment results in Tables 3–11 have already proved the superiority of our DE-EXP algorithm to these LSHADE variants.

5. Conclusion

In this paper, a new assertion is presented that DE variants employing exponential crossover were also able to obtain competitive performance on numerical optimization in comparison



(a) Convergence comparison on benchmark $f_{a_{25}}$ (b) Convergence comparison on benchmark $f_{a_{26}}$



(c) Convergence comparison on benchmark $f_{a_{27}}$ (d) Convergence comparison on benchmark $f_{a_{28}}$

Fig. 10. As a continued part from Fig. 9, convergence comparisons on the last 4 benchmarks are given here.

Table 14
Comparison between DE-EXP algorithm with fitness-improvement based adaptation of scale factor F and our fitness-independent adaptation of F under the 88 benchmarks on 30D optimization with the total number of function evaluation equaling to $nfe_{max} = 10000 \times D$.

	CEC2013 30D optimization		CEC2014 30D optimization		CEC2017 30D optimization	
No.:	DE-EXP with adaptive F	DE-EXP (default)	DE-EXP with adaptive F	DE-EXP (default)	DE-EXP with adaptive F	DE-EXP (default)
f_1	*0/0(≈)*	*0/0*	4.4583E−15/6.6595E−15(<)	**0/0**	2.7864E−15/5.6983E−15(<)	**0/0**
f_2	8.9166E−14/1.1212E−13(<)	**0/0**	2.2292E−15/7.7172E−15(<)	**0/0**	1.5047E−14/1.7384E−14(<)	**5.5729E−16/3.9798E−15**
f_3	4.6818E−02/1.6577E−01(<)	**3.4661E−09/2.4751E−08**	2.2292E−15/1.1144E−14(<)	**0/0**	1.1146E−15/7.9597E−15(<)	**0/0**
f_4	4.4583E−15/3.1839E−14(<)	**0/0**	5.0156E−14/3.1438E−14(<)	**1.2260E−14/2.3612E−14**	**5.3225E+01/1.7514E+01**(>)	5.8997E+01/1.5085E+00
f_5	1.2260E−13/3.0869E−14(<)	**1.1369E−13/2.2737E−14**	2.0086E+01/1.8038E−02(<)	**2.0048E+01/1.2009E−02**	**7.5232E+00/1.6856E+00**(>)	9.8363E+00/2.7475E+00
f_6	5.1779E−01/3.6978E+00(<)	**1.4267E−13/1.0867E−13**	3.0520E−06/1.9383E−05(<)	**0/0**	2.1890E−07/4.3993E−07(<)	**2.6974E−09/1.9164E−08**
f_7	2.1830E−01/2.8949E−01(<)	**4.8007E−03/1.1567E−02**	*0/0(≈)*	*0/0*	**3.8106E+01/2.1535E+00**(>)	4.1855E+01/3.1955E+00
f_8	2.0610E+01/1.9056E−01(<)	**2.0604E+01/1.9914E−01**	3.3214E−13/7.4872E−14(<)	**3.2323E−13/6.5791E−14**	**8.1235E+00/1.6921E+00**(>)	1.0575E+01/3.1178E+00
f_9	**2.2082E+01/5.8059E+00**(>)	2.3582E+01/5.2220E+00	**8.0175E+00/1.6470E+00**(>)	1.1361E+01/3.1635E+00	*0/0(≈)*	*0/0*
f_{10}	*0/0(≈)*	*0/0*	*1.2247E−03/4.9474E−03(≈)*	*1.2247E−03/4.9474E−03*	1.4295E+03/2.1264E+02(<)	**1.4213E+03/2.1360E+02**
f_{11}	1.6830E−13/3.2077E−14(<)	**1.6719E−13/2.6447E−14**	1.2308E+03/1.5928E+02(<)	**1.2190E+03/1.9731E+02**	9.6242E+00/1.1603E+01(<)	**3.6370E+00/4.3566E+00**
f_{12}	**5.1209E+00/1.8443E+00**(>)	6.7762E+00/3.1934E+00	1.4077E−01/2.0143E−02(<)	**1.2480E−01/2.1567E−02**	1.1270E+03/4.2145E+02(<)	**3.8780E+02/2.0304E+02**
f_{13}	**7.8430E+00/7.2374E+00**(>)	8.2574E+00/5.9276E+00	**1.0920E−01/1.4853E−02**(>)	1.1968E−01/2.0558E−02	2.1091E+01/1.7226E+01(<)	**1.4576E+01/6.8877E+00**
f_{14}	2.1636E−02/1.9952E−02(<)	**1.8370E−02/2.0672E−02**	2.0885E−01/3.0753E−02(<)	**1.7712E−01/2.4888E−02**	2.0057E+01/5.7008E+00(<)	**1.4666E+01/7.5224E+00**
f_{15}	2.5911E+03/2.6298E+02(<)	**2.5812E+03/3.0053E+02**	2.0844E+00/2.2726E−01(<)	**2.0540E+00/2.0895E−01**	4.6051E+00/2.7400E+00(<)	**3.4018E+00/1.8734E+00**
f_{16}	6.3903E−01/2.2945E−01(<)	**5.4322E−01/2.4416E−01**	**8.4551E+00/3.8296E−01**(>)	8.5457E+00/4.7679E−01	**5.0082E+01/6.3377E+01**(>)	8.4142E+01/8.6376E+01
f_{17}	*3.0434E+01/9.4299E−07(≈)*	*3.0434E+01/1.9302E−12*	1.9793E+02/1.0535E+02(<)	**5.3267E+01/3.2671E+01**	**2.4419E+01/9.3807E+00**(>)	2.6622E+01/8.8600E+00
f_{18}	**4.9112E+01/2.9521E+00**(>)	4.9702E+01/2.9301E+00	9.1380E+00/5.3117E+00(<)	**1.6533E+00/1.0946E+00**	2.2666E+01/3.6856E+00(<)	**2.1049E+01/1.7001E+00**
f_{19}	**1.1290E+00/9.1811E−02**(>)	1.1602E+00/9.1644E−02	3.6523E+00/5.3214E−01(<)	**2.4130E+00/6.8816E−01**	5.2381E+00/1.1319E+00(<)	**5.2251E+00/1.3742E+00**
f_{20}	9.0520E+00/2.8564E−01(<)	**8.9848E+00/3.6837E−01**	3.5473E+00/1.4104E+00(<)	**2.9915E+00/1.0802E+00**	2.5537E+01/6.6792E+00(<)	**2.8851E+00/3.0861E+01**
f_{21}	**3.0427E+02/5.5153E+01**(>)	3.0648E+02/3.7329E+01	1.0817E+02/7.7647E+01(<)	**1.7459E+01/2.0246E+01**	**2.0764E+02/1.7364E+00**(>)	2.1050E+02/3.0713E+00
f_{22}	1.0617E+02/1.0794E+00(<)	**1.0599E+02/2.6425E−01**	5.0698E+01/4.9739E+01(<)	**4.7141E+01/4.7288E+01**	*1.0000E+02/8.9223E−14(≈)*	*1.0000E+02/1.0047E−13*
f_{23}	**2.4240E+03/3.2192E+02**(>)	2.4599E+03/3.3073E+02	*3.1524E+02/0(≈)*	*3.1524E+02/4.0186E−13*	3.4698E+02/4.5312E+00(<)	**3.4596E+02/4.3531E+00**
f_{24}	2.0012E+02/2.7617E−01(<)	**2.0000E+02/6.0437E−03**	2.2388E+02/1.1548E+00(<)	**2.1241E+02/1.0939E+01**	4.2441E+02/3.1267E+00(<)	**4.2230E+02/3.1373E+00**
f_{25}	**2.2382E+02/2.2072E+01**(>)	2.2815E+02/1.9585E+01	2.0267E+02/1.5925E−01(<)	**2.0260E+02/5.2683E−02**	3.8676E+02/3.3155E−02(<)	**3.8670E+02/1.3131E−02**
f_{26}	*2.0000E+02/0(≈)*	*2.0000E+02/1.4171E−13*	**1.0010E+02/1.4158E−02**(>)	1.0011E+02/1.7762E−02	9.1227E+02/4.7494E+01(<)	**8.5964E+02/4.7641E+01**
f_{27}	3.0188E+02/2.9894E+00(<)	**3.0002E+02/5.3521E−02**	3.0196E+02/1.4003E+01(<)	**3.0000E+02/1.2531E−13**	5.0639E+02/6.4713E+00(<)	**5.0038E+02/7.9802E+00**
f_{28}	*3.0000E+02/1.8101E−13(≈)*	*3.0000E+02/1.6766E−13*	8.5732E+02/1.8994E+01(<)	**8.2148E+02/2.1032E+01**	3.3101E+02/4.8570E+01(<)	**3.2717E+02/4.6976E+01**
f_{29}	–	–	6.6169E+02/1.6596E+02(<)	**5.2335E+02/2.7633E+02**	4.2894E+02/7.9906E+00(<)	**4.2654E+02/1.6083E+01**
f_{30}	–	–	5.6558E+02/1.9540E+02(<)	**4.3544E+02/9.7758E+01**	2.0727E+03/8.4912E+01(<)	**1.9989E+03/3.7312E+01**
>/≈/<	8/5/15	−/−/−	4/3/23	−/−/−	7/2/21	−/−/−

with these state-of-the-art DE variants with binomial crossover. A brief theoretical discussion of the equivalency is given between binomial crossover and exponential crossover from the perspective of selecting vertices from the hyper-cube during the evolution first, and then presented the detailed implementation of such a DE variant with exponential crossover, the novel DE-EXP algorithm. A large test suite containing 88 benchmarks is employed in algorithm validation, and the experiment results showed the superiority of the novel DE-EXP algorithm in comparison with the state-of-the-art DE variants on numerical optimization.

Moreover, it can be found that: (1) the proper *Cr* values in exponential crossover were usually in a much narrower range than in binomial crossover when obtaining equivalent performance; (2) finding these proper *Cr* values was very difficult for a DE variant employing exponential crossover let alone finding its corresponding parameter control; (3) there was no DE variant with exponential crossover secured front ranks [20] in recent competitions[1] because the community did not find the proper

1 https://www3.ntu.edu.sg/home/epnsugan/.

Table 15
Comparison between DE-EXP algorithm with and without reduction of population size *PS* under the 88 benchmarks on 30D optimization with the total number of function evaluation equaling to $nfe_{max} = 10000 \times D$.

CEC2013 30D optimization			CEC2014 30D optimization		CEC2017 30D optimization	
No.:	Without reduction of *PS*	DE-EXP (default)	Without reduction of *PS*	DE-EXP (default)	Without reduction of *PS*	DE-EXP (default)
f_1	1.5604E−13/1.0655E−13(<)	**0/0**	6.7890E+02/1.8009E+03(<)	**0/0**	2.0062E−14/8.1271E−15(<)	**0/0**
f_2	2.2619E+04/1.6135E+04(<)	**0/0**	3.0651E−14/9.5847E−15(<)	**0/0**	2.2236E−13/2.1962E−13(<)	**5.5729E−16/3.9798E−15**
f_3	1.3973E+05/5.0214E+05(<)	**3.4661E−09/2.4751E−08**	5.9073E−14/1.9562E−14(<)	**0/0**	1.0366E−13/3.8897E−14(<)	**0/0**
f_4	1.5624E−05/3.8247E−05(<)	**0/0**	1.0477E−13/5.2558E−14(<)	**1.2260E−14/2.3612E−14**	**3.7568E+01/2.9678E+01**(>)	5.8997E+01/1.5085E+00
f_5	1.6496E−13/6.5560E−14(<)	**1.1369E−13/2.2737E−14**	2.0067E+01/1.3648E−02(<)	**2.0048E+01/1.2009E−02**	1.8791E+01/5.2937E+00(<)	**9.8363E+00/2.7475E+00**
f_6	2.2275E+00/7.1668E+00(<)	**1.4267E−13/1.0867E−13**	1.7268E+00/1.1707E+00(<)	**0/0**	8.8474E−03/2.0908E−02(<)	**2.6974E−09/1.9164E−08**
f_7	5.8769E+00/4.5193E+00(<)	**4.8007E−03/1.1567E−02**	3.1888E−03/5.8568E−03(<)	**0/0**	5.2020E+01/4.3631E+00(<)	**4.1855E+01/3.1955E+00**
f_8	**2.0584E+01/3.0142E−01**(>)	2.0604E+01/1.9914E−01	1.9509E−02/1.3932E−01(<)	**3.2323E−13/6.5791E−14**	1.8849E+01/4.2405E+00(<)	**1.0575E+01/3.1178E+00**
f_9	2.5733E+01/5.9252E+00(<)	**2.3582E+01/5.2220E+00**	1.8658E+01/4.2065E+00(<)	**1.1361E+01/3.1635E+00**	1.6424E+00/1.5314E+00(<)	**0/0**
f_{10}	1.9020E−02/1.2851E−02(<)	**0/0**	4.4904E−03/9.5983E−03(<)	**1.2247E−03/4.9474E−03**	1.7406E+03/1.9840E+02(<)	**1.4213E+03/2.1360E+02**
f_{11}	1.1705E−01/7.0767E−01(<)	**1.6719E−13/2.6447E−14**	1.4227E+03/2.2685E+02(<)	**1.2190E+03/1.9731E+02**	3.0232E+01/2.4964E+01(<)	**3.6370E+00/4.3566E+00**
f_{12}	1.8312E+01/4.3704E+00(<)	**6.7762E+00/3.1934E+00**	1.3692E−01/2.4459E−02(<)	**1.2480E−01/2.1567E−02**	9.0582E+02/4.2132E+02(<)	**3.8780E+02/2.0304E+02**
f_{13}	4.5323E+01/1.3583E+01(<)	**8.2574E+00/5.9276E+00**	2.0493E−01/3.6283E−02(<)	**1.1968E−01/2.0558E−02**	5.4138E+01/3.6845E+01(<)	**1.4576E+01/6.8877E+00**
f_{14}	2.5428E+00/1.7849E+01(<)	**1.8370E−02/2.0672E−02**	2.2415E−01/3.7324E−02(<)	**1.7712E−01/2.4888E−02**	2.1882E+01/7.5019E+00(<)	**1.4666E+01/7.5224E+00**
f_{15}	2.9372E+03/4.0521E+02(<)	**2.5812E+03/3.0053E+02**	2.5236E+00/3.3714E−01(<)	**2.0540E+00/2.0895E−01**	1.6701E+01/8.6910E+00(<)	**3.4018E+00/1.8734E+00**
f_{16}	**5.3934E−01/3.0038E−01**(>)	5.4322E−01/2.4416E−01	9.1059E+00/4.6441E−01(<)	**8.5457E+00/4.7679E−01**	3.3023E+02/1.7103E+02(<)	**8.4142E+01/8.6376E+01**
f_{17}	*3.0434E+01/1.9191E−14(≈)*	*3.0434E+01/1.9302E−12*	4.2456E+02/2.2133E+02(<)	**5.3267E+01/3.2671E+01**	4.5858E+01/3.3178E+01(<)	**2.6622E+01/8.8600E+00**
f_{18}	6.4795E+01/6.9158E+00(<)	**4.9702E+01/2.9301E+00**	1.6180E+01/7.2394E+00(<)	**1.6533E+00/1.0946E+00**	3.5303E+01/1.6754E+01(<)	**2.1049E+01/1.7001E+00**
f_{19}	**1.1251E+00/1.0788E−01**(>)	1.1602E+00/9.1644E−02	4.0542E+00/6.3106E−01(<)	**2.4130E+00/6.8816E−01**	8.6253E+00/3.1262E+00(<)	**5.2251E+00/1.3742E+00**
f_{20}	1.0139E+01/9.8124E−01(<)	**8.9848E+00/3.6837E−01**	1.1749E+01/7.3284E+00(<)	**2.9915E+00/1.0802E+00**	8.3691E+01/5.5371E+01(<)	**2.8851E+00/3.0861E+01**
f_{21}	3.0768E+02/7.3815E+01(<)	**3.0648E+02/3.7329E+01**	1.5811E+02/8.6469E+01(<)	**1.7459E+01/2.0246E+01**	2.2122E+02/6.0360E+00(<)	**2.1050E+02/3.0713E+00**
f_{22}	**1.0224E+02/1.9381E+01**(>)	1.0599E+02/2.6425E−01	1.3615E+02/6.3955E+01(<)	**4.7141E+01/4.7288E+01**	1.0005E+02/3.5020E−01(<)	**1.0000E+02/1.0047E−13**
f_{23}	3.2320E+03/3.8919E+02(<)	**2.4599E+03/3.3073E+02**	*3.1524E+02/1.6455E−13(≈)*	*3.1524E+02/4.0186E−13*	3.6192E+02/7.6411E+00(<)	**3.4596E+02/4.3531E+00**
f_{24}	2.1067E+02/4.9112E+00(<)	**2.0000E+02/6.0437E−03**	2.2663E+02/3.6949E+00(<)	**2.1241E+02/1.0939E+01**	4.3308E+02/7.1997E+00(<)	**4.2230E+02/3.1373E+00**
f_{25}	2.4526E+02/1.6587E+01(<)	**2.2815E+02/1.9585E+01**	2.0348E+02/7.7415E−01(<)	**2.0260E+02/5.2683E−02**	3.8742E+02/2.3488E+00(<)	**3.8670E+02/1.3131E−02**
f_{26}	2.3776E+02/5.4040E+01(<)	**2.0000E+02/1.4171E−13**	1.0608E+02/2.3716E+01(<)	**1.0011E+02/1.7762E−02**	1.0661E+03/1.3769E+02(<)	**8.5964E+02/4.7641E+01**
f_{27}	4.1982E+02/5.6339E+01(<)	**3.0002E+02/5.3521E−02**	3.4216E+02/3.2593E+01(<)	**3.0000E+02/1.2531E−13**	5.0856E+02/9.4702E+00(<)	**5.0038E+02/7.9802E+00**
f_{28}	3.1985E+02/1.4178E+02(<)	**3.0000E+02/1.6766E−13**	8.5862E+02/5.1404E+01(<)	**8.2148E+02/2.1032E+01**	3.4291E+02/5.7258E+01(<)	**3.2717E+02/4.6976E+01**
f_{29}	–	–	5.6585E+02/2.3982E+02(<)	**5.2335E+02/2.7633E+02**	4.5137E+02/2.5848E+01(<)	**4.2654E+02/1.6083E+01**
f_{30}	–	–	7.4495E+02/3.6130E+02(<)	**4.3544E+02/9.7758E+01**	2.2106E+03/1.6846E+02(<)	**1.9989E+03/3.7312E+01**
>/≈/<	4/1/23	−/−/−	0/1/29	−/−/−	1/0/29	−/−/−

Table 16
Comparison between DE-EXP with "DE/rand/1/exp" and DE-EXP algorithm with the default trial vector generation strategy under the 88 benchmarks on 30D optimization with the total number of function evaluation equaling to $nfe_{max} = 10000 \times D$.

CEC2013 30D optimization			CEC2014 30D optimization		CEC2017 30D optimization	
No.:	DE-EXP with "DE/rand/1/exp"	DE-EXP (default)	DE-EXP with "DE/rand/1/exp"	DE-EXP (default)	DE-EXP with "DE/rand/1/exp"	DE-EXP (default)
f_1	3.1219E+04/1.6568E+04(<)	**0/0**	7.1756E+08/5.4164E+08(<)	**0/0**	3.5304E+10/2.1839E+10(<)	**0/0**
f_2	4.8492E+08/3.3800E+08(<)	**0/0**	5.1486E+10/2.2827E+10(<)	**0/0**	6.7469E+44/4.7870E+45(<)	**5.5729E−16/3.9798E−15**
f_3	3.8935E+15/1.4712E+16(<)	**3.4661E−09/2.4751E−08**	9.9631E+04/9.4177E+04(<)	**0/0**	2.0677E+05/5.5642E+04(<)	**0/0**
f_4	1.4155E+05/2.9347E+04(<)	**0/0**	7.3460E+03/5.0429E+03(<)	**1.2260E−14/2.3612E−14**	7.8333E+03/5.4624E+03(<)	**5.8997E+01/1.5085E+00**
f_5	1.2270E+04/9.0716E+03(<)	**1.1369E−13/2.2737E−14**	2.0926E+01/1.3136E−01(<)	**2.0048E+01/1.2009E−02**	3.6772E+02/9.0639E+01(<)	**9.8363E+00/2.7475E+00**
f_6	3.2453E+03/2.8153E+03(<)	**1.4267E−13/1.0867E−13**	3.6374E+01/5.0864E+00(<)	**0/0**	6.7740E+01/2.4501E+01(<)	**2.6974E−09/1.9164E−08**
f_7	3.2805E+04/1.0678E+05(<)	**4.8007E−03/1.1567E−02**	3.9976E+02/2.3269E+02(<)	**0/0**	1.2221E+03/3.9082E+02(<)	**4.1855E+01/3.1955E+00**
f_8	2.1197E+01/5.7257E−02(<)	**2.0604E+01/1.9914E−01**	2.8464E+02/9.3827E+01(<)	**3.2323E−13/6.5791E−14**	3.5488E+02/6.7598E+01(<)	**1.0575E+01/3.1178E+00**
f_9	4.3101E+01/3.2108E+00(<)	**2.3582E+01/5.2220E+00**	3.6150E+02/8.2833E+01(<)	**1.1361E+01/3.1635E+00**	1.0763E+04/6.1394E+03(<)	**0/0**
f_{10}	3.9812E+03/1.9472E+03(<)	**0/0**	5.9723E+03/1.3833E+03(<)	**1.2247E−03/4.9474E−03**	7.5006E+03/9.8511E+02(<)	**1.4213E+03/2.1360E+02**
f_{11}	5.7787E+02/2.7568E+02(<)	**1.6719E−13/2.6447E−14**	7.1149E+03/9.4284E+02(<)	**1.2190E+03/1.9731E+02**	7.9426E+03/5.5756E+03(<)	**3.6370E+00/4.3566E+00**
f_{12}	6.6724E+02/1.5126E+02(<)	**6.7762E+00/3.1934E+00**	2.4922E+00/8.9042E−01(<)	**1.2480E−01/2.1567E−02**	6.1059E+09/3.9824E+09(<)	**3.8780E+02/2.0304E+02**
f_{13}	6.5731E+02/1.7216E+02(<)	**8.2574E+00/5.9276E+00**	4.9931E+00/1.9911E+00(<)	**1.1968E−01/2.0558E−02**	4.7805E+09/3.6981E+09(<)	**1.4576E+01/6.8877E+00**
f_{14}	6.0227E+03/1.7342E+03(<)	**1.8370E−02/2.0672E−02**	1.2880E+02/7.6763E+01(<)	**1.7712E−01/2.4888E−02**	6.8782E+06/4.7170E+06(<)	**1.4666E+01/7.5224E+00**
f_{15}	8.7225E+03/4.3279E+02(<)	**2.5812E+03/3.0053E+02**	1.0468E+06/1.6216E+06(<)	**2.0540E+00/2.0895E−01**	7.2662E+08/5.5233E+08(<)	**3.4018E+00/1.8734E+00**
f_{16}	4.3513E+00/4.6128E−01(<)	**5.4322E−01/2.4416E−01**	1.3023E+01/4.9420E−01(<)	**8.5457E+00/4.7679E−01**	3.0196E+03/6.0635E+02(<)	**8.4142E+01/8.6376E+01**
f_{17}	1.1711E+03/4.7260E+02(<)	**3.0434E+01/1.9302E−12**	5.2453E+07/4.3234E+07(<)	**5.3267E+01/3.2671E+01**	1.6024E+03/3.8993E+02(<)	**2.6622E+01/8.8600E+00**
f_{18}	1.2188E+03/3.9275E+02(<)	**4.9702E+01/2.9301E+00**	1.6987E+09/1.1659E+09(<)	**1.6533E+00/1.0946E+00**	4.3076E+07/4.0505E+07(<)	**2.1049E+01/1.7001E+00**
f_{19}	7.1386E+05/9.9531E+05(<)	**1.1602E+00/9.1644E−02**	3.1275E+02/1.5998E+02(<)	**2.4130E+00/6.8816E−01**	8.7707E+08/6.6658E+08(<)	**5.2251E+00/1.3742E+00**
f_{20}	1.4953E+01/1.3349E−01(<)	**8.9848E+00/3.6837E−01**	4.1331E+05/7.5436E+05(<)	**2.9915E+00/1.0802E+00**	1.0852E+03/2.1389E+02(<)	**2.8851E+00/3.0861E+01**
f_{21}	3.3237E+03/8.0547E+02(<)	**3.0648E+02/3.7329E+01**	2.4678E+07/1.7202E+07(<)	**1.7459E+01/2.0246E+01**	5.3212E+02/6.6187E+01(<)	**2.1050E+02/3.0713E+00**
f_{22}	6.9780E+03/1.7964E+03(<)	**1.0599E+02/2.6425E−01**	1.6142E+03/4.6350E+02(<)	**4.7141E+01/4.7288E+01**	5.9445E+03/2.1969E+03(<)	**1.0000E+02/1.0047E−13**
f_{23}	9.3651E+03/3.9626E+02(<)	**2.4599E+03/3.3073E+02**	6.7316E+02/2.3987E+02(<)	**3.1524E+02/4.0186E−13**	9.0526E+02/1.3633E+02(<)	**3.4596E+02/4.3531E+00**
f_{24}	3.4412E+02/1.7464E+01(<)	**2.0000E+02/6.0437E−03**	3.7452E+02/4.4612E+01(<)	**2.1241E+02/1.0939E+01**	9.7398E+02/1.5935E+02(<)	**4.2230E+02/3.1373E+00**
f_{25}	3.5785E+02/1.4194E+01(<)	**2.2815E+02/1.9585E+01**	2.5547E+02/2.8800E+01(<)	**2.0260E+02/5.2683E−02**	3.4054E+03/2.0702E+03(<)	**3.8670E+02/1.3131E−02**
f_{26}	3.1503E+02/7.6951E+01(<)	**2.0000E+02/1.4171E−13**	1.0748E+02/2.2195E+01(<)	**1.0011E+02/1.7762E−02**	6.2763E+03/1.5504E+03(<)	**8.5964E+02/4.7641E+01**
f_{27}	1.4826E+03/8.6491E+01(<)	**3.0002E+02/5.3521E−02**	1.0318E+03/2.8720E+02(<)	**3.0000E+02/1.2531E−13**	1.0057E+03/2.4681E+02(<)	**5.0038E+02/7.9802E+00**
f_{28}	4.5339E+03/1.4444E+03(<)	**3.0000E+02/1.6766E−13**	4.2143E+03/1.5473E+03(<)	**8.2148E+02/2.1032E+01**	3.2357E+03/1.4757E+03(<)	**3.2717E+02/4.6976E+01**
f_{29}	–	–	1.0748E+08/1.0081E+08(<)	**5.2335E+02/2.7633E+02**	2.6419E+03/9.4330E+02(<)	**4.2654E+02/1.6083E+01**
f_{30}	–	–	9.9722E+05/8.9376E+05(<)	**4.3544E+02/9.7758E+01**	3.9483E+08/3.2121E+08(<)	**1.9989E+03/3.7312E+01**
>/≈/<	0/0/28	−/−/−	0/0/30	−/−/−	0/0/30	−/−/−

parameter control. In this paper, a novel DE variant with exponential crossover is proposed, namely DE-EXP, in which a novel parameter control was designed for the exponential crossover. The experiment results show that our DE-EXP algorithm is very competitive with the winner DE variants in Congress on Evolutionary Computation (CEC) competitions.

Besides these contributions, a novel fitness-value-independent parameter control was also proposed to tackle the fitness-value-dependency weakness existing in these state-of-the-art DE variants including LSHADE, iLSHADE, jSO, LPalmDE and HARD-DE etc. In these DE variants, the fitness improvement, to be exactly the fitness difference, is taken as weight in the adaptation of control parameters, therefore, they cannot tackle the optimization model shown in Fig. 2. In our paper, a novel fitness-value-independent adaptation of scale factor F and an automatical adaptation of crossover rate Cr are given in the DE-EXP algorithm, and these adaptations are all fitness-value-independent, therefore, our algorithm can be applied in much wider scenarios especially those that the exact fitness values of the models are unavailable or deliberately hidden.

As is already mentioned in Eq. (5) of Section 1, given a binomial crossover rate Cr_b, there always exists an exponential crossover rate Cr_e, $Cr_e \in [0, 1]$, satisfying the equation and vice versa. From this perspective of view, there always exists a DE framework in which the binomial crossover and exponential crossover can be converted into each other with equivalent performance, in other words, the DE framework with exponential crossover can obtain equivalent performance to the DE

Table 17
Comparison between DE-EXP with adaptive *Cr* and DE-EXP algorithm with automatically generated *Cr* (the default version) under the 88 benchmarks on 30D optimization with the total number of function evaluation equaling to $nfe_{max} = 10000 \times D$.

	CEC2013 30D optimization		CEC2014 30D optimization		CEC2017 30D optimization	
No.:	DE-EXP with binomial crossover	DE-EXP (default)	DE-EXP with binomial crossover	DE-EXP (default)	DE-EXP with binomial crossover	DE-EXP (default)
f_1	*0/0(≈)*	*0/0*	*0/0(≈)*	*0/0*	*0/0(≈)*	*0/0*
f_2	*0/0(≈)*	*0/0*	*0/0(≈)*	*0/0*	7.7514E−15/1.3276E−14(<)	**5.5729E−16/3.9798E−15**
f_3	**2.7533E−11/9.0487E−11**(>)	3.4661E−09/2.4751E−08	*0/0(≈)*	*0/0*	*0/0(≈)*	*0/0*
f_4	*0/0(≈)*	*0/0*	**5.1676E−15/1.7139E−14**(>)	1.2260E−14/2.3612E−14	5.9067E+01/1.6751E+00(<)	**5.8997E+01/1.5085E+00**
f_5	1.5503E−13/7.6648E−14(<)	**1.1369E−13/2.2737E−14**	**2.0001E+01/1.7159E−03**(>)	2.0048E+01/1.2009E−02	1.0673E+01/4.8354E+00(<)	**9.8363E+00/2.7475E+00**
f_6	**1.0335E−13/3.4278E−14**(>)	1.4267E−13/1.0867E−13	2.5675E−06/8.5153E−06(<)	**0/0**	4.4179E−07/6.2051E−07(<)	**2.6974E−09/1.9164E−08**
f_7	1.2688E−02/2.6420E−02(<)	**4.8007E−03/1.1567E−02**	*0/0(≈)*	*0/0*	4.2615E+01/3.7708E+00(<)	**4.1855E+01/3.1955E+00**
f_8	**2.0592E+01/1.8803E−01**(>)	2.0604E+01/1.9914E−01	1.2392E+01/2.8617E+00(<)	**3.2323E−13/6.5791E−14**	1.0673E+01/4.0069E+00(<)	**1.0575E+01/3.1178E+00**
f_9	**1.5536E+01/3.9953E+00**(>)	2.3582E+01/5.2220E+00	1.2482E+01/4.1111E+00(<)	**1.1361E+01/3.1635E+00**	*0/0(≈)*	*0/0*
f_{10}	*0/0(≈)*	*0/0*	1.9114E+03/6.8084E+02(<)	**1.2247E−03/4.9474E−03**	2.7003E+03/4.8956E+02(<)	**1.4213E+03/2.1360E+02**
f_{11}	1.0130E+01/4.0737E+00(<)	**1.6719E−13/2.6447E−14**	2.5470E+03/4.4755E+02(<)	**1.2190E+03/1.9731E+02**	9.3439E+00/1.7609E+01(<)	**3.6370E+00/4.3566E+00**
f_{12}	1.0130E+01/3.5546E+00(<)	**6.7762E+00/3.1934E+00**	2.3988E−01/1.8361E−01(<)	**1.2480E−01/2.1567E−02**	4.2025E+02/2.3670E+02(<)	**3.8780E+02/2.0304E+02**
f_{13}	1.3228E+01/9.4973E+00(<)	**8.2574E+00/5.9276E+00**	**1.0032E−01/2.1126E−02**(>)	1.1968E−01/2.0558E−02	**1.3092E+01/6.6023E+00**(>)	1.4576E+01/6.8877E+00
f_{14}	2.1019E+03/3.0076E+02(<)	**1.8370E−02/2.0672E−02**	1.9285E−01/2.2250E−02(<)	**1.7712E−01/2.4888E−02**	2.2200E+01/1.1570E+00(<)	**1.4666E+01/7.5224E+00**
f_{15}	2.9085E+03/7.8010E+02(<)	**2.5812E+03/3.0053E+02**	2.6537E+00/4.9016E−01(<)	**2.0540E+00/2.0895E−01**	**1.4966E+00/1.4671E+00**(>)	3.4018E+00/1.8734E+00
f_{16}	**2.5681E−01/1.7090E−01**(>)	5.4322E−01/2.4416E−01	9.3640E+00/7.1372E−01(<)	**8.5457E+00/4.7679E−01**	**2.4475E+01/3.5727E+01**(>)	8.4142E+01/8.6376E+01
f_{17}	4.3066E+01/3.8665E+00(<)	**3.0434E+01/1.9302E−12**	7.4933E+01/3.8504E+01(<)	**5.3267E+01/3.2671E+01**	3.6735E+01/8.1142E+00(<)	**2.6622E+01/8.8600E+00**
f_{18}	**4.2641E+01/3.1038E+00**(>)	4.9702E+01/2.9301E+00	2.6703E+00/7.8314E−01(<)	**1.6533E+00/1.0946E+00**	**2.0707E+01/3.7260E−01**(>)	2.1049E+01/1.7001E+00
f_{19}	2.2551E+00/5.6045E−01(<)	**1.1602E+00/9.1644E−02**	**2.4037E+00/5.2933E−01**(>)	2.4130E+00/6.8816E−01	**4.8893E+00/1.6558E+00**(>)	5.2251E+00/1.3742E+00
f_{20}	8.9923E+00/4.5033E−01(<)	**8.9848E+00/3.6837E−01**	**2.3745E+00/9.3772E−01**(>)	2.9915E+00/1.0802E+00	2.7993E+01/6.4086E+00(<)	**2.8851E+00/3.0861E+01**
f_{21}	**2.9091E+02/3.0151E+01**(>)	3.0648E+02/3.7329E+01	3.1099E+01/5.1566E+01(<)	**1.7459E+01/2.0246E+01**	2.1165E+02/3.2337E+00(<)	**2.1050E+02/3.0713E+00**
f_{22}	2.3206E+03/4.4654E+02(<)	**1.0599E+02/2.6425E−01**	**4.5804E+01/4.7180E+01**(>)	4.7141E+01/4.7288E+01	*1.0000E+02/0(≈)*	*1.0000E+02/1.0047E−13*
f_{23}	2.6177E+03/3.9712E+02(<)	**2.4599E+03/3.3073E+02**	*3.1524E+02/0(≈)*	*3.1524E+02/4.0186E−13*	3.5405E+02/5.3511E+00(<)	**3.4596E+02/4.3531E+00**
f_{24}	*2.0000E+02/1.1501E−02(≈)*	*2.0000E+02/6.0437E−03*	**2.1199E+02/1.1486E+01**(>)	2.1241E+02/1.0939E+01	4.2522E+02/2.6444E+00(<)	**4.2230E+02/3.1373E+00**
f_{25}	2.3535E+02/1.2513E+01(<)	**2.2815E+02/1.9585E+01**	2.0261E+02/5.9834E−02(<)	**2.0260E+02/5.2683E−02**	3.8671E+02/9.5918E−03(<)	**3.8670E+02/1.3131E−02**
f_{26}	*2.0000E+02/6.9036E−14(≈)*	*2.0000E+02/1.4171E−13*	**1.0009E+02/2.0726E−02**(>)	1.0011E+02/1.7762E−02	8.8033E+02/5.5389E+01(<)	**8.5964E+02/4.7641E+01**
f_{27}	3.0016E+02/3.4884E−01(<)	**3.0002E+02/5.3521E−02**	*3.0000E+02/2.3847E−13(≈)*	*3.0000E+02/1.2531E−13*	**4.9996E+02/6.0967E+00**(>)	5.0038E+02/7.9802E+00
f_{28}	*3.0000E+02/1.4714E−13(≈)*	*3.0000E+02/1.6766E−13*	8.2717E+02/2.3097E+01(<)	**8.2148E+02/2.1032E+01**	**3.1036E+02/3.4365E+01**(>)	3.2717E+02/4.6976E+01
f_{29}	–	–	6.6021E+02/1.8580E+02(<)	**5.2335E+02/2.7633E+02**	4.4083E+02/2.5508E+01(<)	**4.2654E+02/1.6083E+01**
f_{30}	–	–	4.5091E+02/7.1578E+01(<)	**4.3544E+02/9.7758E+01**	**1.9966E+03/2.3372E+01**(>)	1.9989E+03/3.7312E+01
>/≈/<	7/7/14	−/−/−	8/6/16	−/−/−	8/4/18	−/−/−

framework with binomial crossover, both of which can beat the recent state-of-the-art DE algorithms. To discover such a DE framework is one of the future work in our team.

CRediT authorship contribution statement

Zhenyu Meng: Conceptualization, Methodology, Software, Supervision, Writing – review & editing. **Yuxin Chen:** 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.

Data availability

Data will be made available on request.

Acknowledgments

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

References

[1] 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.

[2] K. Price, R.M. Storn, J.A. Lampinen, Differential Evolution: A Practical Approach to Global Optimization, Springer, Berlin, Heidelberg, 2006.

[3] 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 (2016) 144–157.

[4] Z. Zeng, M. Zhang, T. Chen, Z. Hong, A new selection operator for differential evolution algorithm, Knowl.-Based Syst. 226 (2021) 107150.

[5] S. Das, P.N. Suganthan, Differential evolution: A survey of the state-of-the-art, IEEE Trans. Evol. Comput. 15 (1) (2010) 4–31.

[6] 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.

[7] Z. Song, Z. Meng, Differential Evolution with wavelet basis function based parameter control and dimensional interchange for diversity enhancement, Appl. Soft Comput. (2023) 110492.

[8] 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.

[9] 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.

[10] A.E. Eiben, J. Smith, From evolutionary computation to the evolution of things, Nature 521 (7553) (2015) 476–482.

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

[12] 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.

[13] G. Sun, C. Li, L. Deng, An adaptive regeneration framework based on search space adjustment for differential evolution, Neural Comput. Appl. 33 (15) (2021) 9503–9519.

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

[15] D. Zaharie, Influence of crossover on the behavior of differential evolution algorithms, Appl. Soft Comput. 9 (3) (2009) 1126–1138.

[16] C. Lin, A. Qing, Q. Feng, A comparative study of crossover in differential evolution, J. Heuristics 17 (6) (2011) 675–703.

[17] S.-Z. Zhao, P.N. Suganthan, S. Das, Self-adaptive differential evolution with multi-trajectory search for large-scale optimization, Soft Comput. 15 (11) (2011) 2175–2185.

[18] S.-Z. Zhao, P.N. Suganthan, Empirical investigations into the exponential crossover of differential evolutions, Swarm Evol. Comput. 9 (2013) 27–36.

[19] K.R. Opara, J. Arabas, Differential Evolution: A survey of theoretical analyses, Swarm Evol. Comput. 44 (2019) 546–558.

[20] S. Das, S.S. Mullick, P.N. Suganthan, Recent advances in differential evolution–an updated survey, Swarm Evol. Comput. 27 (2016) 1–30.

[21] R. Tanabe, A. Fukunaga, Reevaluating exponential crossover in differential evolution, in: International Conference on Parallel Problem Solving from Nature, Springer, 2014, pp. 201–210.

[22] R. Tanabe, A. Fukunaga, Reviewing and benchmarking parameter control methods in differential evolution, IEEE Trans. Cybern. 50 (3) (2019) 1170–1184.

[23] E. Mezura-Montes, J. Velázquez-Reyes, C.A. Coello Coello, A comparative study of differential evolution variants for global optimization, in: Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation, 2006, pp. 485–492.

[24] X. Qiu, K.C. Tan, J.-X. Xu, Multiple exponential recombination for differential evolution, IEEE Trans. Cybern. 47 (4) (2016) 995–1006.

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

[26] F. Caraffini, A.V. Kononova, D. Corne, Infeasibility and structural bias in differential evolution, Inform. Sci. 496 (2019) 161–179.
[27] S. Alarie, C. Audet, A.E. Gheribi, M. Kokkolaras, S. Le Digabel, Two decades of blackbox optimization applications, EURO J. Comput. Optim. 9 (2021) 100011.
[28] S. Doncieux, J.-B. Mouret, Beyond black-box optimization: a review of selective pressures for evolutionary robotics, Evol. Intell. 7 (2) (2014) 71–93.
[29] 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.
[30] 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.
[31] J. Zhang, A.C. Sanderson, JADE: adaptive differential evolution with optional external archive, IEEE Trans. Evol. Comput. 13 (5) (2009) 945–958.
[32] 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.
[33] 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.
[34] J. Brest, M.S. Maučec, B. Bošković, Single objective real-parameter optimization: algorithm jSO, in: Proc. IEEE Congr. Evol. Comput. (CEC), IEEE, 2017, pp. 1311–1318.
[35] 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.
[36] 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.
[37] Z. Meng, Y. Zhong, C. Yang, CS-DE: Cooperative strategy based differential evolution with population diversity enhancement, Inform. Sci. 577 (2021) 663–696.
[38] Z. Meng, C. Yang, Hip-DE: Historical population based mutation strategy in Differential Evolution with parameter adaptive mechanism, Inform. Sci. 562 (2021) 44–77.
[39] J. Brest, M.S. Maučec, B. Bošković, iL-SHADE: Improved L-SHADE algorithm for single objective real-parameter optimization, in: Proc. IEEE Congr. Evol. Comput. (CEC), 2016, pp. 1188–1195.
[40] Z. Meng, Y. Chen, X. Li, Enhancing differential evolution with novel parameter control, IEEE Access 8 (2020) 51145–51167.