Soft Computing
https://doi.org/10.1007/s00500-019-03934-3

FOCUS

An adaptive differential evolution with combined strategy for global numerical optimization

Gaoji Sun[1] · Bai Yang[2] · Zuqiao Yang[3] · Geni Xu[4]

Abstract

Differential evolution (DE) is a simple yet powerful evolutionary algorithm for numerical optimization. However, the performance of DE significantly relies on its mutation operator and control parameters (scaling factor and crossover rate). In this paper, we propose a novel DE variant by introducing a series of combined strategies into DE, called CSDE. Specifically, in CSDE, to obtain a proper balance between global exploration ability and local exploitation ability, we adopt two mutation operators with different characteristics to produce the mutant vector, and provide a mechanism based on their own historical success rate to coordinate the two adopted mutation operators. Moreover, we combine a periodic function based on one modulo operation, an individual-independence macro-control function and an individual-dependence function based on individual's fitness value information to adaptively produce scaling factor and crossover rate. To verify the effectiveness of the proposed CSDE, comparison experiments contained seven other state-of-the-art DE variants are tested on a suite of 30 benchmark functions and four real-world problems. The simulation results demonstrate that CSDE achieves the best overall performance among the eight DE variants.

Keywords Differential evolution · Adaptive parameter · Combined strategy · Evolutionary algorithm · Global optimization

1 Introduction

Differential evolution (DE) algorithm, proposed by Storn and Price (1997), is a simple but powerful population-based evolutionary algorithm, and its advantages include ease of use, simple structure, efficacy and robustness. In fact, DE has exhibited the predominant performance for handling many kinds of mathematical optimization, such as global numerical optimization (Zhou et al. 2013), constrained optimization (Mohamed and Sabry 2012), multi-objective optimization (Wang et al. 2016) and dynamic optimization (Halder et al. 2013). Moreover, many DE variants and other evolutionary algorithms are proposed to solve practical problems in diverse domains, such as dynamic scheduling (Tang et al. 2014), power system (Pereira and Soares 2015), car engine design (Tayarani-N et al. 2015), image segmentation (Sarkar et al. 2016), neural networks (Arce et al. 2018) and material procurement planning (Sun et al. 2010), supply chain planning (Liu et al. 2018). However, the performance of DE not only depends on the choice of its trial vector generation strategy (mutation operator and crossover operator), but also relies on the selection of its three main control parameters (population size, NP; scale factor, F; and crossover rate, CR). In recent years, many researchers pay attention to design an unexceptionable scheme and propose different adaptive or self-adaptive mechanisms to update the control parameters during the evolutionary process. In nature, the core guideline of designing scheme or parameter adjustment mechanisms is to maintain a proper balance between the global exploration ability and local exploitation ability during the optimization routine (Črepinšek et al. 2013). Unfortunately, how to balance those two abilities is a thorny work;

Communicated by X. Li.

✉ Gaoji Sun
gsun@zjnu.edu.cn

✉ Bai Yang
stockyangb@sina.com

1 College of Economic and Management, Zhejiang Normal University, Jinhua 321004, China

2 School of Management, Chongqing Technology and Business University, 400067 Chongqing, China

3 College of Mathematics and Statistics, Huanggang Normal University, Hubei 438000, China

4 School of Statistics, Xi'an University of Finance and Economics, Xi'an 710100, Shaanxi, China

Published online: 25 March 2019

hence, there still exists a big room to enhance the performance of DE.

Considering the aforementioned analysis, based on the basic guideline about the exploration ability and exploitation ability, we introduce two different mutation operators in CSDE, and both mutation operators use the same concept of $p$best (first proposed in JADE Zhang and Sanderson 2009), but one takes the corresponding parent individual as base vector, and the other one takes the randomly selected $p$best individual as base vector. Meanwhile, the value of parameter p is dynamic, which is determined by a periodic function based on modulo operation and an individual-independence decreasing function. In addition, one cooperation rule of the two mutation operators is relied on their own historical success rate and one reset strategy. Moreover, according to their different features of the two mutation operators, some different combined strategies, which are based on the periodic function, individual-independence decreasing function and an individual-dependence function, are proposed to adaptively determine each individual's scaling factor and crossover rate. At last, a suite of 30 benchmark functions and four real-life problems, and seven state-of-the-art DE variants are applied to verify the performance of our proposed CSDE.

The remainder of this paper is organized as follows. Section 2 briefly introduces the basic operators of original DE algorithm. Section 3 reviews some currently related works on DE. Section 4 provides a detailed description of the proposed CSDE algorithm and its overall procedure. Section 5 presents the comparison between CSDE and seven compared algorithms. Section 6 draws the conclusions.

2 Differential evolution

In DE algorithm, the population consists of NP individuals, and each individual is expressed by a D-dimensional vector. The process of executing DE usually contains four parts: initialization operation, mutation operation, crossover operation and selection operation. In the following subsection, we will provide a brief description about the four parts of classical DE algorithm.

2.1 Initialization operation

In the initialization step, DE usually generates NP individuals, expressed as $\boldsymbol{x}_i = [x_{i,1}, x_{i,2}, \ldots, x_{i,D}]$, $i = 1, 2, \ldots, \mathrm{NP}$, via the following formula:

$$x_{i,j} = L_j + \mathrm{rand}_{i,j} \times (U_j - L_j), \quad j = 1, 2, \ldots, D, \tag{1}$$

where vectors $\boldsymbol{L} = [L_1, L_2, \ldots, L_D]$ and $\boldsymbol{U} = [U_1, U_2, \ldots, U_D]$ are the prescribed lower bounds and upper bounds of the search space and $\mathrm{rand}_{i,j}$ denotes a real number randomly generated from the interval [0, 1].

2.2 Mutation operation

After the initialization operation, for each target vector $\boldsymbol{x}_i$, the mutation operation is employed to generate a mutant vector $\boldsymbol{v}_i = [v_{i,1}, v_{i,2}, \ldots, v_{i,D}]$. The following mutation operators are generally utilized and found to be effective:

(1) DE/rand/1

$$\boldsymbol{v}_i = \boldsymbol{x}_{r_1} + F \times (\boldsymbol{x}_{r_2} - \boldsymbol{x}_{r_3}). \tag{2}$$

(2) DE/best/1

$$\boldsymbol{v}_i = \boldsymbol{x}_{\mathrm{best}} + F \times (\boldsymbol{x}_{r_1} - \boldsymbol{x}_{r_2}). \tag{3}$$

(3) DE/current-to-best/1

$$\boldsymbol{v}_i = \boldsymbol{x}_i + F \times (\boldsymbol{x}_{\mathrm{best}} - \boldsymbol{x}_{r_1}) + F \times (\boldsymbol{x}_{r_2} - \boldsymbol{x}_{r_3}). \tag{4}$$

(4) DE/best/2

$$\boldsymbol{v}_i = \boldsymbol{x}_{\mathrm{best}} + F \times (\boldsymbol{x}_{r_1} - \boldsymbol{x}_{r_2}) + F \times (\boldsymbol{x}_{r_3} - \boldsymbol{x}_{r_4}). \tag{5}$$

(5) DE/rand/2

$$\boldsymbol{v}_i = \boldsymbol{x}_{r_1} + F \times (\boldsymbol{x}_{r_2} - \boldsymbol{x}_{r_3}) + F \times (\boldsymbol{x}_{r_4} - \boldsymbol{x}_{r_5}). \tag{6}$$

The indices r_1, r_2, r_3, r_4 and r_5 are mutually exclusive integers randomly selected from set $\{1, 2, \ldots, \mathrm{NP}\}$ and also are not equal to i. The control parameter F is a real number, called scaling factor, influencing the length of differential vector. The vector $\boldsymbol{x}_{\mathrm{best}} = (x_{\mathrm{best},1}, x_{\mathrm{best},2}, \ldots, x_{\mathrm{best},D})$ represents the best individual in the current population.

In the mutation operator, (i.e., DE/rand/1), the vector $\boldsymbol{x}_{r_1}$ denotes the parent vector to be perturbed and also is referred to as the base vector, and $\boldsymbol{x}_{r_2} - \boldsymbol{x}_{r_3}$ is regarded as the difference vector. Actually, $\boldsymbol{x}_{r_2}$ is often regarded as the directional vector. As a matter of fact, the base individual can be taken as the center point of the searching area, the difference vector is applied to set the searching direction, and the scale factor is employed to control the step size.

2.3 Crossover operation

Following mutation step, DE employs crossover operation to generate trial vector $\boldsymbol{u}_i = [u_{i,1}, u_{i,2}, \ldots, u_{i,D}]$ by recombining the variables of target vector $\boldsymbol{x}_i$ and its corresponding mutant vector $\boldsymbol{v}_i$. The most commonly used binary crossover operator can be formulated as follows:

$$u_{i,j} = \begin{cases} v_{i,j}, & \text{if } (\text{rand}_{i,j} \leq \text{CR or } j = j_{\text{rand}}) \\ x_{i,j}, & \text{otherwise,} \end{cases} \tag{7}$$

where CR is the crossover rate and j_{rand} is an integer randomly generated from set $\{1, 2, \ldots, D\}$. The binomial crossover operator copies the jth variable of mutant vector $\boldsymbol{v}_i$ to its corresponding element in the trial vector $\boldsymbol{u}_i$ only when it meets the given condition in the formula (7).

2.4 Selection operation

Following crossover operation, a selection operator is executed to determine whether the target vector $\boldsymbol{x}_i$ or trial vector $\boldsymbol{u}_i$ will survive into the next generation. For a minimization problem, the selection operation can be expressed as:

$$\boldsymbol{x}_i = \begin{cases} \boldsymbol{u}_i, & \text{if } f(\boldsymbol{u}_i) \leq f(\boldsymbol{x}_i) \\ \boldsymbol{x}_i, & \text{otherwise,} \end{cases} \tag{8}$$

where $f(\boldsymbol{u}_i)$ and $f(\boldsymbol{x}_i)$ are the fitness values of target vector $\boldsymbol{u}_i$ and trial vector $\boldsymbol{x}_i$, respectively. According to the expression of selection operator (8), it is easy to see that the population of DE either gets better or remains the same in fitness status, but never deteriorates.

3 Related work

In recent years, more and more researchers focus on determining the most suitable mutation strategies or appropriate parameter control methods (especially for scale factor, F; and crossover rate, CR). Here, we only briefly review some relevant DE variants; the other recent research about DE can be found in the up-to-date literature reviews (Das et al. 2016; Al-Dabbagh et al. 2018) and the references therein.

Many researchers proposed new kinds of mutation strategy or combined some different mutation strategies together to enhance the performance of DE. For example, Zhang and Sanderson (2009) applied a novel current-to-$p$best mutation strategy and adaptive parameters based on historical data to enhance the performance of DE (denoted by JADE). In the current-to-$p$best mutation strategy, the top $100 \times p\%$ best individuals (denoted by $\boldsymbol{x}_{p\text{best}}$) among the current population are applied to provide directional information. Inspired by JADE, Islam et al. (2012) proposed a new DE variant called MDE_pBX, in which a novel directional vector $\boldsymbol{x}_{\text{gr\_best}}$ (adopting the best individual in a dynamic group of $100 \times q\%$ randomly selected individuals from the current population) is used to replace $\boldsymbol{x}_{p\text{best}}$ in current-to-$p$best mutation strategy. Qu et al. (2012) proposed a neighborhood mutation operator, in which the difference vector is selected from a set of similar individuals according to their corresponding Euclidean distance. Gong and Cai (2013) proposed a ranking-based mutation operator for DE, in which the selection rate of one individual is set to its fitness ranking in the current population. Wang et al. (2013) designed a novel mutation operator based on Gaussian distribution and combined DE/best/1 to propose a DE variant (denoted by MGBDE). Sun et al. (2019) combined a novel Gaussian mutation operator (taking the best one of three randomly selected individuals as the mean value) and a modified common mutation operator (called DE/rand-worst/1) to collaboratively produce new mutant vectors and employed a periodic function and a Gaussian function to generate the required values of scaling factor and crossover rate; the proposed DE variant is named GPDE. Han et al. (2013) introduced a group-based DE variant (GDE), in which all the individuals are grouped into a superior group and an inferior group based on their fitness values, and two mutation operators with different search features are designated to the two groups, respectively. Qin et al. (2009) proposed a self-adaptive DE (denoted by SADE), in which various ensembles of multiple mutation strategies with different parameter settings were proposed to realize the advantages sharing. Zheng et al. (2018) proposed a novel interactive information scheme to produce some new mutation operators, in which both the ranking information based on fitness and the interactive information between individuals are fully considered. Zhao et al. (2019) proposed a failure remember-driven self-adaptive differential evolution algorithm, in which a parameter self-adapting strategy driven by "Failure Remember" operation and a "Top-Bottom" strategy are applied to guide individuals toward the potential more promising regions in an optional archive manner. Li et al. (2019) presented a new self-feedback DE algorithm, in which one optimal variation strategy is proposed by extracting the local fitness landscape characteristics in each generation population and combining the probability distributions of modality in each local fitness landscape.

It is well known that selecting appropriate parameters can effectively improve the performance of DE. Therefore, many kinds of parameter control methods (i.e., deterministic, adaptive and self-adaptive control strategy) were proposed by the researchers to make that DE is insensitive to its control parameters when handling different optimization problems. For example, Draa et al. (2015) applied two preset sinusoidal formulas to periodically control the scale factor and crossover rate, respectively. Sarker et al. (2014) applied a new mechanism to dynamically select the best performing combinations of parameters (scale factor, crossover rate and population size) for a problem during the course of a single run. Tang et al. (2015) introduced a novel variant of DE with an individual-dependent mechanism which includes an individual-dependent parameter setting and mutation operator. Yu et al. (2014) established a two-level adaptive parameter control scheme for DE: adjusting the population-level scale factor (F_p) and crossover rate (CR_p) based on the information of exploration and exploitation statuses, and controlling the individual-level scale factor (F_i) and crossover rate (CR_i) according to the individual's fitness value and its distance to the global best individual. Fu et al. (2017) proposed an adaptive differential evolution algorithm with an aging leader and challengers mechanism, called ADE-ALC. In ADE-ALC, the key control parameters are adaptively updated based on given probability distributions which could learn from their successful experiences, and are used to generate the promising parameters at the next generation. Cui et al. (2018) introduced a new parameter self-adaptation method into DE, called ADEDE. In ADEDE, a parameter population is established for the solution population, which is also updated from generation to generation based on the differential evolution under the basic principle that the good parameter individuals will go into the next generation at a high probability. Sun et al. (2018) proposed a novel DE variant (IDDE) with individual-dependent and dynamic parameter adjustment. In IDDE, the parameter adjustment mechanism applies the fitness value information of each individual and one dynamic fluctuation rule, which can provide a better balance between the exploration ability and exploitation ability. Tian et al. (2017) proposed a DE variant with improved individual-based parameter setting, in which the fitness values of original and guiding individuals are used to guide the parameter setting.

4 Description of CSDE

In this section, we firstly provide a detailed description about the two proposed mutation operators, the adopted parameter control methods for scale factor F and crossover rate CR, and the cooperative rule between them, and then summarize the overall procedure of CSDE.

4.1 DE/current-to-*p*best/1

Based on the fast but less reliable convergence performance of the existing greedy mutation operators DE/best/1 (formula 3) and DE/current-to-best/1 (formula 4), Zhang and Sanderson (2009) proposed a new mutation operator, named DE/current-to-$p$best/1. In DE/current-to-$p$best/1, a mutation vector is generated in the following manner:

$$\boldsymbol{v}_{i,g}^{C} = \boldsymbol{x}_{i,g} + F_{i,g}^{C} \times (\boldsymbol{x}_{p\text{best},g} - \boldsymbol{x}_{i,g}) + F_{i,g}^{C} \times (\boldsymbol{x}_{r1,g} - \boldsymbol{x}_{r2,g}), \tag{9}$$

where $\boldsymbol{x}_{p\text{best},g}$ is randomly chosen from the top $100 \times p\%$ individuals in the current population with $p \in (0, 1]$, and the value of scaling factor $F_{i,g}^{C}$ in JADE is associated with the corresponding individual $\boldsymbol{x}_{i,g}$, a Cauchy distribution function and the historical successful mutation factors. In fact, DE/current-to-$p$best/1 is a generalization of DE/current-to-best/1, i.e., any one of the top $100 \times p\%$ individuals can be randomly chosen to play the role of the only best individual in DE/current-to-best/1.

In JADE, the parameter p is a constant value, which equals to 0.05 during the whole evolutionary process. Actually, the value of p determines the greediness of the mutation strategy, which has important influence on the convergence speed and population diversity. Generally speaking, DE/current-to-$p$best/1 has more powerful exploration ability when p is bigger, but has better exploitation ability when p is smaller. Therefore, to get a better balance between the global exploration ability and local exploitation ability, we propose a dynamic adjustment strategy to determine the value of p, and the strategy can be described as follows:

$$p = \text{floor}(0.5 \times M_g \times P_g \times P_g \times \text{NP} + 1), \tag{10}$$

where function "floor(x)" means that the value x is rounded down. $M_g = (G - g + 1)/G$ is an individual-independence macro-control function and $P_g = (\text{FP} - \text{Mod}(g, \text{FP}))/\text{FP}$ is a periodic function based on modulo operation $\text{Mod}(x, y)$, where G is the maximum allowable generations, g is the index of current generation, FP is a periodic adjustment parameter and applied to adjust the cyclic frequency of parameter p. It should be noted that M_g is applied to control the exploration ability and exploitation ability on macro-level, i.e., the emphasis should transfer the exploration work to exploitation work along with evolutionary process, and P_g is used to relieve the problem of premature convergence via periodically recovering the value of parameter p. In order to provide an intuitive understanding about the change rule of parameter p, we plot the graph of p when $\text{FP} = 200$, $G = 10{,}000$ and $\text{NP} = 50$ in Fig. 1.

Different from the generative rule of $F_{i,g}^{C}$ in JADE, the value of scaling factor $F_{i,g}^{C}$ in this paper can be generated via the following manner:

$$F_{i,g}^{C} = M_g \times M_g + (1 - M_g) \times I_{i,g}, \tag{11}$$

where $I_{i,g} = (f(\boldsymbol{x}_{i,g}) - f(\boldsymbol{x}_{\text{best}}))/(f(\boldsymbol{x}_{\text{worst}}) - f(\boldsymbol{x}_{\text{best}}) + 1.0\text{e}{-99})$, which essentially represents the status of individual $\boldsymbol{x}_{i,g}$ in the current population. Normally speaking, in the early stage of the evolutionary process, the population has no enough accumulated knowledge and experience, every individual could be the most potential one, and then we should provide a relatively fair and objective treatment to each individual. However, during the later stage, the individual with better fitness value should focus on its neighborhood to exploit that potential area, but the individual with poorer fitness value should pay attention to exploring a wider area. Based on the aforementioned cognition and the fact that

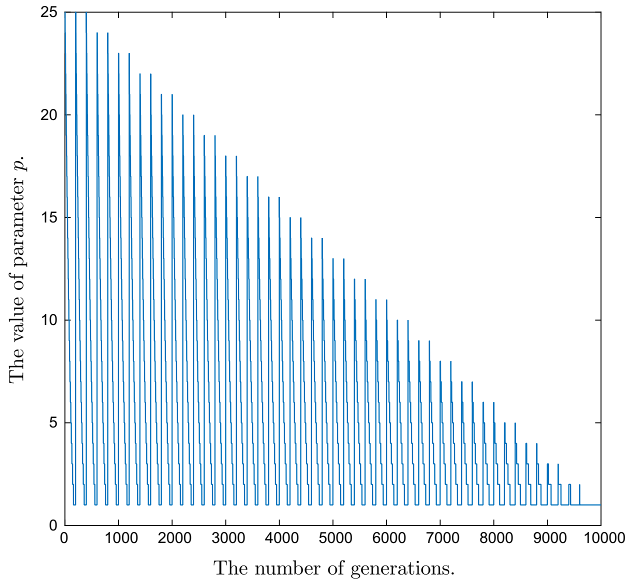

Fig. 1 Changing value of p during the optimization process

DE/current-to-$p$best/1 takes the current individual $\boldsymbol{x}_{i,g}$ as the base vector, we set the value of parameter $F_{i,g}^{C}$ via formula (10), which implies that in the early stage of the evolutionary process, the value of $F_{i,g}^{C}$ mainly relies on the individual-independence macro-control parameter M_g, but in the later stage, $F_{i,g}^{C}$ mainly depends on the individual-dependence function $I_{i,g}$.

4.2 DE/*p*best-to-rand/1

In DE/current-to-$p$best/1, the current individual $\boldsymbol{x}_{i,g}$ is taken as the base vector, and the randomly selected $p$best individual is used to be the directional vector. In order to reinforce the exploitation work around the promising $p$best individual, we propose a novel mutation operator, called DE/$p$best-to-rand/1, which can be expressed as follows:

$$\boldsymbol{v}_{i,g}^{R} = \boldsymbol{x}_{p\text{best},g} + F_{i,g}^{R} \cdot (\boldsymbol{x}_{r_1,g} - \boldsymbol{x}_{i,g}) + F_{i,g}^{R} \cdot (\boldsymbol{x}_{r_2,g} - \boldsymbol{x}_{i,g}), \tag{12}$$

where $F_{i,g}^{R}$ is scaling factor, which is applied to adjust the step size of mutation operator DE/$p$best-to-rand/1, and its value is generated by the following formula,

$$F_{i,g}^{R} = M_g \times M_g + (1 - M_g) \times P_g. \tag{13}$$

It is easy to see that the value of $F_{i,g}^{R}$ mainly relies on the individual-independence macro-control parameter M_g in the early stage, but largely depends on the modulo-based periodic parameter P_g. Formula (13) implies that DE/$p$best-to-rand/1 can periodically adjust the max search radius around the $p$best individual, which essentially influences the global exploration ability and local exploitation ability. Moreover, since the base individual can be taken as the center point of the searching area, DE/current-to-$p$best/1, which can take every individual in the population as the base vector, actually has more powerful global exploration ability than DE/$p$best-to-rand/1. Meanwhile, DE/$p$best-to-rand/1, which only takes the selected $p$best individual as the base vector, has better local exploitation ability than DE/current-to-$p$best/1. Therefore, combining the DE/current-to-$p$best/1 and DE/$p$best-to-rand/1 has the potential advantages to obtain a good balance between the global exploration ability and local exploitation ability, which will lead to an enhancement in the performance of DE.

4.3 Cooperative rule

Now, we have provided the detailed description about the two adopted mutation operators (DE/current-to-$p$best/1 and DE/$p$best-to-rand/1). For each specific individual, it is important to decide that which one from the two mutation operators is selected to produce the mutant vector, and historical success rate becomes the natural selection. To be more specific, taking DE/current-to-$p$best/1 as an example, the historical success rate (denoted by SR_g^C) in the gth generation can be calculated via the cumulative success times (S_g^C) and the cumulative run times (R_g^C), i.e., $\mathrm{SR}_g^C = S_g^C / R_g^C$. We execute operation $R_g^C = R_g^C + 1$ when one individual in the population takes DE/current-to-$p$best/1 to produce the mutant vector, and if the following obtained trial vector is better than its parent vector, then execute $S_g^C = S_g^C + 1$. Furthermore, if the obtained trial vector is better than the best individual in the current population, execute $S_g^C = S_g^C + 1$ again as an extra reward. Using the same operation rule, we can obtain the historical success rate of DE/$p$best-to-rand/1 in the gth generation: $\mathrm{SR}_g^R = S_g^R / R_g^R$. Note that at the beginning of the optimization process, all the initial values of S_0^C, R_0^C, S_0^R and R_0^R are set to 1.

Based on the obtained value of SR_g^C and SR_g^R in the gth generation, at the beginning of the following $(g+1)$th generation, the value of a parameter (SR_{g+1}) involved in cooperative rule can be derived in terms of the two mutation operators' historical success rate, which can be calculated by

$$\mathrm{SR}_{g+1} = \frac{\mathrm{SR}_g^C}{\mathrm{SR}_g^C + \mathrm{SR}_g^R}, \tag{14}$$

where parameter SR_{g+1} is applied to control the selection probability of DE/$p$best-to-rand/1 in the next generation. The detailed coordination mechanism in the gth generation can be described as follows:

$$\boldsymbol{v}_{i,g} = \begin{cases} \boldsymbol{v}_{i,g}^{C}, \text{ if } \text{rand}_i[0,1] < \text{SR}_g, \\ \boldsymbol{v}_{i,g}^{R}, \text{ otherwise.} \end{cases} \tag{15}$$

The coordination mechanism (14) shows that the chance of executing the adopted two mutation operators relies on their own historical success rate, and the one with higher cumulative success rate has the more chance to produce the mutant vectors. However, history usually cannot represent the present. Therefore, in order to avoid that the current value of SR_g is overly dependant on the historical success rate, we reset all the current values of S_g^C, R_g^C, S_g^R and R_g^R to 1 when $\text{Mod}(g, \text{FP}) = 0$.

After the new mutant vector $\boldsymbol{v}_{i,g}$ produced, comparing the two fitness values of $\boldsymbol{v}_i$ and its target vector $\boldsymbol{x}_{i,g}$ and then determining the offspring via the following selection operator:

$$u_{i,g}^{j} = \begin{cases} v_{i,g}^{j}, \text{ if } (\text{rand}_{i,j} \leq \text{CR}_{i,g} \text{ or } j = j_{\text{rand}}) \\ x_{i,g}^{j}, \text{ otherwise,} \end{cases} \tag{16}$$

where $\text{CR}_{i,g}$ is the crossover rate of the ith individual in the gth generation, and its value is determined by a combination of the decreasing function M_g, the individual-dependence function $I_{i,g}$ and a Gaussian function $N(\mu, \sigma^2)$, where μ is the mean value and σ is the standard deviation. The specific computational formula is as follows:

$$\text{CR}_{i,g} = M_g \times N_{i,g} + (1 - M_g) \times I_{i,g}, \tag{17}$$

where $N_{i,g}$ is a real number within [0, 1] and generated by the Gaussian function $N(\mu, \sigma^2)$ with $\mu = 0.5$ and $\sigma = 0.1$. Formula (17) tells us that the value of crossover rate $\text{CR}_{i,g}$ mainly relies on a random number generated by a Gaussian function $N(\mu, \sigma^2)$ in the earlier stage, but largely depends on the individual-dependence function $I_{i,g}$ in the later stage, which is exactly in accordance with the cognitive pattern about the law of individual change. In detail, for all the individuals in the early stage, the degree of their change should be impersonal and determined at random, but in the later stage, the individuals with better fitness value should has a lesser extent of change, which is beneficial to improve the accuracy of the solution.

4.4 The overall procedure of CSDE

We have provided a detailed description of DE/current-to-$p$best/1, DE/$p$best-to-rand/1, and the cooperative rule between them. Now we summarize the overall procedure of CSDE into Algorithm 1.

Algorithm 1 The overall procedure of CSDE

Set the values of parameters NP, FP and G;
Initialize NP individuals with random positions via the formula (1);
for ($g = 1$; $g <= G$; $g++$), **do**
    Compute the value of parameter p in the gth generation via the formula (10);
    Compute the value of parameter SR_g in the gth generation via the formula (14);
    for ($i = 1$; $i <= \text{NP}$; $i++$), **do**
        Compute the crossover rate $\text{CR}_{i,g}$ of the ith individual in the gth generation via the formula (17);
        if rand[0, 1] $< \text{SR}_g$ **then**
            Compute the scaling factor $F_{i,g}^{C}$ of the ith individual in the gth generation via the formula (11);
            Generate the new mutant vector via the formula (9) included DE/current-to-$p$best/1;
        else
            Compute the scaling factor $F_{i,g}^{R}$ of the ith individual in the gth generation via the formula (13);
            Generate the new mutant vector via the formula (12) contained DE/$p$best-to-rand/1;
        end if;
        Generate the new trial vector via the formula (16);
        Update the ith individual via the selection operator (8);
        Replace the best individual $\boldsymbol{x}_{\text{best}}$ by the new individual $\boldsymbol{x}_i$ if $\boldsymbol{x}_i$ is better than $\boldsymbol{x}_{\text{best}}$;
    end for
end for
Output the position of the best individual as the global optimal solution.

5 Experiments and discussion

5.1 Benchmark functions and real-world problems

To show the performance of CSDE, thirty well-known benchmark functions from IEEE CEC 2014 (Liang et al. 2013) are used to test, which are listed in Table 1. More precisely, the adopted test functions are carried out in the comparative experiment when $D = 30$, $D = 50$ and $D = 100$, respectively. In addition, four real-world problems, which are widely used to evaluate the performance of various algorithms, include parameter estimation for frequency-modulated sound waves (Das and Suganthan 2011), spread spectrum radar poly-phase code design (Das and Suganthan 2011), systems of linear equations (García-Martínez et al. 2008) and parameter optimization for polynomial fitting problem (Herrera and Lozano 2000), and they are denoted by rf_1, rf_2, rf_3 and rf_4 in our experiment, respectively.

5.2 Compared algorithms and parameter settings

The compared algorithms used in our comparative experiment contain five state-of-the-art DE variants and two up-to-date DE variants. To be specific, they are SADE (Qin et al. 2009), JADE (Zhang and Sanderson 2009), GDE (Han et al. 2013), MGBDE (Wang et al. 2013), SinDE (Draa et al.

Table 1 Summary of the IEEE CEC 2014 benchmark functions

Type	No.	Functions	f^*
Unimodal	f_1	Rotated high conditioned elliptic function	100
	f_2	Rotated bent cigar function	200
	f_3	Rotated discus function	300
Multimodal	f_4	Shifted and rotated Rosenbrock function	400
	f_5	Shifted and rotated Ackley's function	500
	f_6	Shifted and rotated Weierstrass function	600
	f_7	Shifted and rotated Griewank's function	700
	f_8	Shifted Rastrigin function	800
	f_9	Shifted and rotated Rastrigin's function	900
	f_{10}	Shifted Schwefel function	1000
	f_{11}	Shifted and rotated Schwefel's function	1100
	f_{12}	Shifted and rotated Katsuura function	1200
	f_{13}	Shifted and rotated HappyCat function	1300
	f_{14}	Shifted and rotated HGBat function	1400
	f_{15}	Shifted and rotated Expanded Griewank's plus Rosenbrock's function	1500
	f_{16}	Shifted and rotated Expanded Scaffer's F6 function	1600
Hybrid	f_{17}	Hybrid function 1	1700
	f_{18}	Hybrid function 2	1800
	f_{19}	Hybrid function 3	1900
	f_{20}	Hybrid function 4	2000
	f_{21}	Hybrid function 5	2100
	f_{22}	Hybrid function 6	2200
Composition	f_{23}	Composition function 1	2300
	f_{24}	Composition function 2	2400
	f_{25}	Composition function 3	2500
	f_{26}	Composition function 4	2600
	f_{27}	Composition function 5	2700
	f_{28}	Composition function 6	2800
	f_{29}	Composition function 7	2900
	f_{30}	Composition function 8	3000
		Search space: $[-100, 100]^D$	

2015), IDDE (Sun et al. 2018) and GPDE (Sun et al. 2019). These DE variants not only have outstanding performance, but also have some similar aspects to our proposed CSDE; that is why we select them as the competitors.

To make a fair comparison, the population sizes NP of all the adopted algorithms, respectively, equal to D and $5D$ for the test functions and real-world problems; the other involved control parameters keep the same with their corresponding literature. In CSDE, the periodic adjustment parameter FP is set to 200 for all different problems. Moreover, the maximal number of the function evaluations is used as the stop criterion, which is set to $10{,}000 \times D$. It should be pointed out that population size $\text{NP} = D$ or $\text{NP} = 5D$; then, $10{,}000 \times D$ function evaluations is equivalent to 10,000 or 2000 generations for the test functions and real-world problems, respectively.

5.3 Comparative results

All the compared algorithms conduct 50 independent runs on every problem. The *mean error* (Mean) and *standard deviation* (Std.) of the error function value $f(\boldsymbol{x}_{\text{best}}) - f(\boldsymbol{x}^*)$ are used to evaluate the performance, where $f(\boldsymbol{x}^*)$ is the fitness value of the corresponding global optimal solution and $f(\boldsymbol{x}_{\text{best}})$ is the fitness value of the best individual in the population obtained by at the end of optimization procedure. Furthermore, the *Wilcoxon's rank-sum test* with the 5% significant level is employed to test the significant differences between CSDE and each corresponding competitor. To be more specific, we apply the symbols "+", "=" and "−" to represent that performance of the CSDE is significantly better than, similar to, or worse than that of the corresponding compared algorithms, respectively. The experimental results are listed in Tables 2, 3, 4, 5, 6, 7 and 8. For the sake of clarity, the results of *Wilcoxon's rank-sum test* for different dimensional functions and real-life problems are summarized in Table 9.

According to the results of Wilcoxon's rank-sum test reported in Table 9, we can summarize four discoveries. First of all, CSDE has the best overall performance among the eight compared algorithms. To be more specific, CSDE is better than SADE, JADE, GDE, MGBDE, SinDE, IDDE and GPDE on 62, 53, 76, 63, 54, 45 and 42 test functions, respectively, similar to SADE, JADE, GDE, MGBDE, SinDE, IDDE and GPDE on 15, 18, 12, 13, 28, 34 and 27 test functions, respectively, and slightly worse than them on 17, 23, 6, 18, 12, 15 and 25 test functions, respectively. Secondly, in the view of dimensional contrast, CSDE has a similar performance on different dimensional problems and consistently keeps the comparative advantage than the other seven compared algorithms. Thirdly, while handling the problems with different characteristics (i.e., unimodal, multimodal, hybrid and composition), CSDE has better performance than SADE, JADE, GDE and SinDE on all the four classes of problems; compared to MGBDE, GPDE and IDDE, CSDE only has slightly worse performance on unimodal, multimodal, hybrid problems, respectively; moreover, for the most complex composition problems, CSDE defeats all the competitors, which

Table 2 Comparative results on functions $f_1 - f_{15}$ with $D = 30$

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	IDDE	GPDE	CSDE
f_1	Mean	3.22e+04	3.76e+04	6.85e+03	5.57e+03	1.92e+06	1.22e+05	5.21e+04	4.66e+04
	Std.	2.17e+04	3.66e+04	6.41e+03	3.45e+03	1.11e+06	9.89e+04	3.51e+04	4.06e+04
	+/=/–	=	=	−	−	+	+	=	
f_2	Mean	4.84e−20	3.08e−20	3.06e−22	3.92e−15	2.90e−24	9.43e−23	1.35e−23	7.77e−25
	Std.	2.39e−19	9.38e−20	3.80e−22	1.95e−14	1.01e−23	2.27e−22	2.17e−22	3.89e−24
	+/=/–	+	+	+	+	=	+	+	
f_3	Mean	9.82e−10	1.34e−01	3.56e−03	2.08e−18	2.18e−12	4.01e−21	5.42e−25	1.14e−28
	Std.	4.91e−09	6.05e−01	9.98e−03	9.54e−18	9.18e−12	9.44e−21	1.83e−24	2.44e−28
	+/=/–	+	+	+	+	+	+	+	
f_4	Mean	1.39e+01	1.94e+01	5.48e+00	9.95e−05	1.43e+01	4.00e−01	2.99e+00	3.31e−01
	Std.	2.72e+01	3.09e+01	1.89e+01	4.48e−04	2.28e+01	3.32e−01	1.49e+01	5.85e−02
	+/=/–	+	=	−	−	+	=	−	
f_5	Mean	2.04e+01	2.00e+01	2.09e+01	2.02e+01	2.06e+01	2.05e+01	2.00e+01	2.00e+01
	Std.	4.74e−02	3.99e−03	1.48e−01	3.92e−02	4.77e−02	6.17e−02	6.53e−06	1.76e−02
	+/=/–	+	+	+	+	+	+	−	
f_6	Mean	8.93e+00	1.14e+01	8.25e+00	2.22e+01	1.93e−02	6.44e−01	1.33e+00	2.46e−01
	Std.	2.03e+00	1.75e+00	2.85e+00	3.64e+00	9.27e−02	8.43e−01	1.16e+00	4.21e−01
	+/=/–	+	+	+	+	−	=	+	
f_7	Mean	1.89e−02	2.69e−02	1.30e−02	1.12e−02	0.00e+00	0.00e+00	2.17e−03	3.95e−04
	Std.	2.10e−02	2.17e−02	1.75e−02	1.44e−02	0.00e+00	0.00e+00	4.17e−03	1.97e−03
	+/=/–	+	+	+	+	=	=	=	
f_8	Mean	4.78e+00	3.98e−02	6.38e+01	1.29e+02	4.70e−01	9.51e+00	9.79e+00	7.72e+00
	Std.	2.54e+00	1.99e−01	1.58e+01	3.17e+01	5.69e−01	3.22e+00	3.72e+00	2.30e+00
	+/=/–	−	−	+	+	−	+	+	
f_9	Mean	4.59e+01	4.97e+01	7.46e+01	1.56e+02	3.58e+01	3.99e+01	3.46e+01	3.66e+01
	Std.	1.09e+01	8.71e+00	2.74e+01	2.95e+01	7.51e+00	7.47e+00	9.26e+00	7.95e+00
	+/=/–	+	+	+	+	=	=	=	
f_{10}	Mean	3.40e+00	6.98e+00	1.80e+03	1.25e+03	9.31e+00	3.76e+01	1.25e+02	1.34e+02
	Std.	2.09e+00	2.40e+01	6.43e+02	7.96e+02	4.77e+00	9.05e+01	9.65e+01	1.52e+02
	+/=/–	−	−	+	+	−	=	=	
f_{11}	Mean	2.42e+03	2.04e+03	5.06e+03	2.85e+03	2.35e+03	2.07e+03	1.97e+03	2.21e+03
	Std.	5.45e+02	2.43e+02	1.60e+03	6.29e+02	4.08e+02	4.54e+02	4.71e+02	5.37e+02
	+/=/–	=	=	+	+	=	=	=	
f_{12}	Mean	5.91e−01	1.28e−01	1.77e+00	3.36e−01	8.04e−01	4.35e−01	1.49e−01	2.17e−01
	Std.	8.56e−02	2.61e−02	8.70e−01	3.67e−02	1.24e−01	2.49e−01	7.84e−02	6.53e−02
	+/=/–	+	−	+	+	+	+	−	
f_{13}	Mean	2.80e−01	3.09e−01	3.64e−01	4.40e−01	2.06e−01	2.06e−01	2.40e−01	1.52e−01
	Std.	5.43e−02	5.90e−02	7.23e−02	7.72e−02	5.27e−02	5.78e−02	6.84e−02	3.23e−02
	+/=/–	+	+	+	+	+	+	+	
f_{14}	Mean	2.41e−01	2.50e−01	2.95e−01	2.58e−01	2.42e−01	2.25e−01	2.22e−01	1.94e−01
	Std.	4.47e−02	1.01e−01	9.12e−02	5.78e−02	2.60e−02	7.29e−02	3.40e−02	3.19e−02
	+/=/–	+	+	+	+	+	+	+	
f_{15}	Mean	4.57e+00	1.23e+01	8.12e+00	1.33e+01	4.81e+00	3.77e+00	3.75e+00	3.24e+00
	Std.	1.31e+00	6.69e+00	2.95e+00	2.37e+00	9.80e−01	9.67e−01	9.44e−01	7.17e−01
	+/=/–	+	+	+	+	+	+	+	

Table 3 Comparative results on functions $f_{16} - f_{30}$ with $D = 30$

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	IDDE	GPDE	CSDE
f_{16}	Mean	1.03e+01	1.02e+01	1.10e+01	1.07e+01	1.00e+01	9.50e+00	9.64e+00	1.01e+01
	Std.	4.16e−01	3.56e−01	1.12e+00	6.49e−01	5.22e−01	5.34e−01	7.85e−01	7.72e−01
	+/=/–	=	=	+	+	=	−	−	
f_{17}	Mean	5.95e+03	6.29e+04	1.91e+04	1.71e+03	1.25e+05	7.84e+03	3.84e+03	3.34e+03
	Std.	4.17e+03	6.21e+04	3.03e+04	7.57e+02	1.20e+05	7.96e+03	3.53e+03	3.41e+03
	+/=/–	+	+	+	=	+	+	=	
f_{18}	Mean	8.34e+02	7.06e+02	7.43e+01	1.03e+02	5.15e+02	1.66e+01	2.16e+01	2.87e+01
	Std.	1.22e+03	9.62e+02	1.90e+02	3.60e+01	6.94e+02	6.42e+00	9.20e+00	1.27e+01
	+/=/–	+	+	+	+	+	−	=	
f_{19}	Mean	4.23e+00	1.11e+01	4.73e+00	2.33e+01	3.71e+00	2.98e+00	3.45e+00	2.90e+00
	Std.	1.23e+00	1.66e+01	1.20e+00	2.55e+01	7.18e−01	7.58e−01	1.18e+00	9.31e−01
	+/=/–	+	+	+	+	+	=	=	
f_{20}	Mean	1.08e+02	1.37e+03	2.88e+01	7.80e+01	2.57e+01	7.34e+00	1.71e+01	1.16e+01
	Std.	1.42e+02	2.06e+03	2.19e+01	4.38e+01	2.85e+01	2.96e+00	1.12e+01	3.50e+00
	+/=/–	+	+	+	+	+	−	+	
f_{21}	Mean	4.84e+03	5.74e+03	3.29e+03	8.50e+02	9.23e+03	3.34e+02	3.63e+03	4.50e+02
	Std.	4.42e+03	7.17e+03	5.36e+03	3.90e+02	7.52e+03	1.88e+02	4.61e+03	3.04e+02
	+/=/–	+	+	+	+	+	=	+	
f_{22}	Mean	1.55e+02	2.10e+02	4.82e+02	7.17e+02	7.25e+01	3.98e+01	2.90e+02	4.96e+01
	Std.	8.61e+01	7.94e+01	2.09e+02	2.74e+02	6.28e+01	3.73e+01	1.41e+02	4.94e+01
	+/=/–	+	+	+	+	=	=	+	
f_{23}	Mean	3.15e+02	3.15e+02	3.15e+02	3.15e+02	3.15e+02	3.15e+02	3.15e+02	3.15e+02
	Std.	1.38e−13	3.48e−13	6.13e−13	1.21e−12	1.24e−13	4.78e−14	1.04e−13	1.27e−13
	+/=/–	=	=	=	=	=	=	=	
f_{24}	Mean	2.28e+02	2.30e+02	2.35e+02	2.42e+02	2.23e+02	2.00e+02	2.27e+02	2.21e+02
	Std.	5.41e+00	3.87e+00	7.27e+00	1.18e+01	8.57e−01	1.65e−02	4.51e+00	6.78e+00
	+/=/–	+	+	+	+	=	−	+	
f_{25}	Mean	2.10e+02	2.12e+02	2.04e+02	2.20e+02	2.04e+02	2.03e+02	2.04e+02	2.03e+02
	Std.	2.07e+00	1.47e+00	1.19e+00	6.22e+00	6.64e−01	2.53e−01	7.80e−01	2.98e−01
	+/=/–	+	+	+	+	+	=	+	
f_{26}	Mean	1.12e+02	1.56e+02	1.00e+02	1.56e+02	1.00e+02	1.00e+02	1.08e+02	1.00e+02
	Std.	3.31e+01	5.05e+01	9.32e−02	5.04e+01	3.99e−02	4.22e−02	2.76e+01	2.99e−02
	+/=/–	+	+	+	+	+	+	+	
f_{27}	Mean	4.36e+02	4.29e+02	4.28e+02	8.61e+02	3.02e+02	3.19e+02	3.34e+02	3.53e+02
	Std.	5.74e+01	6.28e+01	5.93e+01	3.11e+02	7.08e+00	2.13e+01	3.49e+01	4.75e+01
	+/=/–	+	+	+	+	−	−	=	
f_{28}	Mean	9.12e+02	9.19e+02	9.56e+02	2.60e+03	7.94e+02	7.87e+02	7.93e+02	7.93e+02
	Std.	4.37e+01	6.51e+01	6.23e+01	7.61e+02	3.23e+01	4.32e+01	2.60e+01	2.56e+01
	+/=/–	+	+	+	+	=	=	=	
f_{29}	Mean	6.83e+02	7.84e+02	4.80e+02	7.18e+02	1.32e+03	8.88e+02	6.32e+02	7.80e+02
	Std.	2.64e+02	2.69e+02	2.73e+02	1.26e+02	2.39e+02	9.32e+01	1.96e+02	4.83e+01
	+/=/–	−	=	−	=	+	+	−	
f_{30}	Mean	1.96e+03	2.05e+03	1.11e+03	2.14e+03	8.10e+02	1.14e+03	1.62e+03	9.01e+02
	Std.	6.22e+02	5.50e+02	3.00e+02	7.18e+02	1.69e+02	4.56e+02	7.04e+02	3.35e+02
	+/=/–	+	+	+	+	=	+	+	

Table 4 Comparative results on functions $f_1 - f_{15}$ with $D = 50$

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	IDDE	GPDE	CSDE
f_1	Mean	1.81e+05	7.59e+04	4.26e+05	6.96e+04	2.96e+06	4.07e+05	9.52e+05	4.63e+05
	Std.	7.78e+04	3.67e+04	1.89e+05	3.61e+04	1.02e+06	1.66e+05	3.05e+05	1.82e+05
	+/=/–	−	−	=	−	+	=	+	
f_2	Mean	4.72e+03	4.03e+03	2.27e+01	1.37e−10	4.15e+03	5.00e−02	1.75e+00	6.28e−04
	Std.	4.07e+03	5.19e+03	3.58e+01	4.09e−10	2.76e+03	1.08e−01	3.35e+00	1.39e−03
	+/=/–	+	+	+	−	+	+	+	
f_3	Mean	1.91e+01	3.88e−02	3.09e+01	2.70e−05	5.81e+02	6.05e+00	1.45e+00	9.17e−03
	Std.	2.32e+01	1.04e−01	3.14e+01	4.57e−05	4.24e+02	1.01e+01	4.09e+00	1.15e−02
	+/=/–	+	−	+	−	+	+	+	
f_4	Mean	6.24e+01	6.26e+01	2.41e+01	1.50e+01	9.28e+01	9.72e+01	4.37e+01	6.50e+01
	Std.	3.62e+01	2.72e+01	3.31e+01	3.83e+01	3.51e+00	3.34e+00	3.58e+01	3.97e+01
	+/=/–	=	=	−	−	=	+	−	
f_5	Mean	2.07e+01	2.01e+01	2.11e+01	2.04e+01	2.08e+01	2.07e+01	2.00e+01	2.01e+01
	Std.	3.48e−02	8.33e−03	5.23e−02	2.87e−02	3.49e−02	3.92e−02	4.73e−06	3.94e−02
	+/=/–	+	−	+	+	+	+	−	
f_6	Mean	1.95e+01	2.54e+01	1.53e+01	4.19e+01	4.07e−02	1.70e+00	4.97e+00	3.92e−01
	Std.	3.73e+00	2.37e+00	2.82e+00	4.31e+00	1.34e−01	1.44e+00	2.99e+00	5.41e−01
	+/=/–	+	+	+	+	−	+	+	
f_7	Mean	8.70e−03	2.16e−02	9.03e−03	6.24e−03	1.18e−16	1.04e−16	6.57e−04	1.85e−16
	Std.	9.17e−03	4.29e−02	9.96e−03	6.16e−03	1.29e−16	1.42e−16	2.55e−03	2.04e−16
	+/=/–	+	+	+	+	=	=	=	
f_8	Mean	9.42e+00	6.63e−02	1.11e+02	2.58e+02	1.10e+01	3.44e+01	1.57e+01	2.47e+01
	Std.	3.62e+00	2.57e−01	2.01e+01	4.42e+01	4.33e+00	4.86e+00	3.93e+00	4.88e+00
	+/=/–	−	−	+	+	−	+	−	
f_9	Mean	9.15e+01	1.04e+02	1.09e+02	2.95e+02	7.04e+01	8.09e+01	6.76e+01	6.15e+01
	Std.	1.15e+01	1.46e+01	2.81e+01	3.68e+01	1.92e+01	1.97e+01	9.61e+00	1.14e+01
	+/=/–	+	+	+	+	=	+	=	
f_{10}	Mean	2.50e+00	3.05e+00	3.46e+03	3.37e+03	1.57e+02	4.20e+01	2.03e+02	9.52e+02
	Std.	1.07e+00	9.81e−01	1.13e+03	1.93e+03	8.15e+01	1.06e+01	1.46e+02	3.63e+02
	+/=/–	−	−	+	+	−	−	−	
f_{11}	Mean	7.04e+03	4.07e+03	1.26e+04	5.65e+03	4.95e+03	4.45e+03	4.52e+03	4.86e+03
	Std.	4.82e+02	4.54e+02	2.13e+03	6.00e+02	7.01e+02	7.44e+02	6.60e+02	6.00e+02
	+/=/–	+	−	+	+	=	=	=	
f_{12}	Mean	7.82e−01	1.57e−01	3.15e+00	4.10e−01	1.34e+00	8.49e−01	1.54e−01	4.03e−01
	Std.	9.98e−02	1.51e−02	4.66e−01	3.78e−02	1.25e−01	2.96e−01	6.97e−02	1.43e−01
	+/=/–	+	−	+	=	+	+	−	
f_{13}	Mean	4.17e−01	4.56e−01	5.06e−01	5.46e−01	3.46e−01	3.38e−01	3.44e−01	2.29e−01
	Std.	5.86e−02	6.73e−02	9.63e−02	1.12e−01	5.18e−02	6.62e−02	6.10e−02	3.81e−02
	+/=/–	+	+	+	+	+	+	+	
f_{14}	Mean	3.14e−01	2.85e−01	3.50e−01	3.60e−01	2.48e−01	2.57e−01	2.46e−01	2.76e−01
	Std.	3.21e−02	1.76e−02	1.57e−01	1.30e−01	3.08e−02	8.07e−02	2.42e−02	1.70e−01
	+/=/–	+	+	+	+	+	+	+	
f_{15}	Mean	1.44e+01	3.03e+01	1.85e+01	2.77e+01	8.38e+00	6.35e+00	6.86e+00	5.55e+00
	Std.	3.35e+00	7.10e+00	1.20e+01	4.11e+00	1.56e+00	1.35e+00	1.81e+00	8.16e−01
	+/=/–	+	+	+	+	+	+	+	

Table 5 Comparative results on functions $f_{16} - f_{30}$ with $D = 50$

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	IDDE	GPDE	CSDE
f_{16}	Mean	1.98e+01	1.86e+01	2.22e+01	1.92e+01	2.00e+01	1.86e+01	1.86e+01	1.91e+01
	Std.	2.50e−01	4.04e−01	3.04e−01	4.33e−01	6.39e−01	1.03e+00	9.38e−01	5.43e−01
	+/=/–	+	−	+	=	+	=	−	
f_{17}	Mean	2.23e+04	9.29e+04	5.48e+04	1.06e+04	3.94e+05	8.66e+04	5.41e+04	3.45e+04
	Std.	1.41e+04	7.40e+04	3.40e+04	4.95e+03	2.26e+05	6.35e+04	2.85e+04	1.18e+04
	+/=/–	=	+	+	−	+	+	+	
f_{18}	Mean	4.02e+02	8.88e+02	1.50e+02	6.03e+02	3.53e+02	2.74e+02	4.28e+01	1.28e+02
	Std.	3.23e+02	7.82e+02	1.56e+02	1.20e+03	3.13e+02	2.10e+02	3.36e+01	4.23e+01
	+/=/–	+	+	=	=	+	=	−	
f_{19}	Mean	1.36e+01	3.40e+01	9.41e+00	1.92e+01	9.58e+00	8.50e+00	6.82e+00	9.43e+00
	Std.	5.79e+00	2.14e+01	4.50e+00	2.18e+00	7.10e−01	1.07e+00	1.34e+00	8.62e−01
	+/=/–	+	+	=	+	=	−	−	
f_{20}	Mean	2.39e+02	6.87e+02	4.80e+02	1.81e+02	2.16e+02	3.79e+01	1.17e+02	7.79e+01
	Std.	6.34e+01	1.37e+03	5.13e+02	4.31e+01	1.53e+02	1.55e+01	1.45e+02	2.48e+01
	+/=/–	+	+	+	+	+	−	=	
f_{21}	Mean	2.65e+04	4.22e+04	2.93e+04	3.02e+03	2.62e+05	3.46e+04	3.13e+04	9.52e+03
	Std.	1.94e+04	4.93e+04	2.48e+04	1.81e+03	1.49e+05	2.07e+04	3.80e+04	6.31e+03
	+/=/–	+	+	+	−	+	+	+	
f_{22}	Mean	4.22e+02	5.80e+02	1.36e+03	1.23e+03	2.57e+02	1.96e+02	2.14e+02	2.60e+02
	Std.	1.04e+02	1.07e+02	4.14e+02	3.62e+02	1.49e+02	1.43e+02	1.64e+02	1.66e+02
	+/=/–	+	+	+	+	=	=	=	
f_{23}	Mean	3.44e+02	3.44e+02	3.44e+02	3.44e+02	3.44e+02	3.44e+02	3.44e+02	3.44e+02
	Std.	1.55e−13	2.36e−13	1.87e−13	5.02e−13	1.15e−13	1.17e−13	8.73e−14	1.12e−13
	+/=/–	=	=	=	=	=	=	=	
f_{24}	Mean	2.72e+02	2.78e+02	2.81e+02	3.02e+02	2.64e+02	2.56e+02	2.63e+02	2.66e+02
	Std.	6.57e+00	3.95e+00	3.36e+00	1.13e+01	3.28e+00	2.12e+00	6.01e+00	3.45e+00
	+/=/–	+	+	+	+	=	−	=	
f_{25}	Mean	2.10e+02	2.28e+02	2.08e+02	2.38e+02	2.10e+02	2.07e+02	2.08e+02	2.06e+02
	Std.	1.01e+01	2.16e+00	2.06e+00	4.72e+00	1.29e+00	9.22e−01	2.23e+00	6.09e−01
	+/=/–	=	+	=	+	+	=	+	
f_{26}	Mean	1.54e+02	1.20e+02	1.00e+02	1.07e+02	1.00e+02	1.00e+02	1.09e+02	1.00e+02
	Std.	5.15e+01	4.12e+01	7.72e−02	2.57e+01	2.36e−02	5.72e−02	3.51e+01	2.57e−02
	+/=/–	+	+	+	+	+	+	+	
f_{27}	Mean	7.70e+02	7.75e+02	7.05e+02	1.58e+03	3.22e+02	3.46e+02	4.32e+02	3.31e+02
	Std.	8.80e+01	1.53e+02	1.05e+02	1.48e+02	2.13e+01	1.31e+01	3.60e+01	2.61e+01
	+/=/–	+	+	+	+	=	=	+	
f_{28}	Mean	1.42e+03	1.61e+03	1.45e+03	5.36e+03	1.09e+03	1.06e+03	1.19e+03	1.08e+03
	Std.	1.14e+02	2.46e+02	1.10e+02	1.03e+03	3.62e+01	2.12e+01	5.11e+01	3.38e+01
	+/=/–	+	+	+	+	=	−	+	
f_{29}	Mean	1.05e+03	9.94e+02	8.36e+02	8.96e+02	1.89e+03	1.43e+03	1.27e+03	1.29e+03
	Std.	1.93e+02	1.17e+02	2.51e+02	2.04e+02	3.13e+02	2.59e+02	2.25e+02	2.08e+02
	+/=/–	−	−	−	−	+	+	=	
f_{30}	Mean	1.06e+04	1.14e+04	9.89e+03	1.18e+04	8.99e+03	8.26e+03	9.07e+03	8.58e+03
	Std.	1.68e+03	1.71e+03	5.53e+02	9.07e+02	2.86e+02	2.93e+02	4.86e+02	4.42e+02
	+/=/–	+	+	+	+	+	=	+	

Table 6 Comparative results on functions $f_1 - f_{15}$ with $D = 100$

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	IDDE	GPDE	CSDE
f_1	Mean	8.96e+05	4.70e+05	8.29e+06	4.38e+05	2.19e+07	3.39e+06	9.59e+06	1.91e+06
	Std.	1.45e+05	1.93e+05	3.20e+06	1.51e+05	5.50e+06	1.41e+06	2.74e+06	6.23e+05
	+/=/–	−	−	+	−	+	+	+	
f_2	Mean	1.31e+04	6.38e+03	1.74e+04	2.80e−10	1.06e+04	3.31e+00	1.23e+01	8.95e−04
	Std.	8.76e+03	1.08e+04	2.13e+04	4.04e−10	6.32e+03	8.21e+00	1.50e+01	1.22e−03
	+/=/–	+	+	+	−	+	+	+	
f_3	Mean	7.66e+01	8.65e+00	5.22e+03	2.51e−02	3.22e+03	5.09e+02	3.45e+02	1.34e+01
	Std.	6.52e+01	2.83e+00	4.07e+03	6.79e−02	1.34e+03	4.66e+02	2.62e+02	1.18e+01
	+/=/–	+	=	+	−	+	+	+	
f_4	Mean	1.71e+02	1.62e+02	1.89e+02	1.35e+02	1.59e+02	1.87e+02	1.60e+02	1.66e+02
	Std.	4.05e+01	4.80e+01	3.38e+01	5.92e+01	2.57e+01	3.15e+01	2.77e+01	3.15e+01
	+/=/–	=	=	+	=	=	+	=	
f_5	Mean	2.10e+01	2.03e+01	2.13e+01	2.07e+01	2.12e+01	2.11e+01	2.00e+01	2.05e+01
	Std.	2.30e−02	1.75e−02	2.31e−02	2.06e−02	2.46e−02	3.01e−02	9.75e−07	8.81e−02
	+/=/–	+	−	+	+	+	+	−	
f_6	Mean	6.44e+01	7.23e+01	4.56e+01	1.03e+02	4.79e+00	7.71e+01	6.36e+00	7.24e+00
	Std.	5.35e+00	3.68e+00	7.45e+00	5.42e+00	2.59e+00	7.48e+00	3.85e+00	4.53e+00
	+/=/–	+	+	+	+	=	+	=	
f_7	Mean	2.63e−03	6.39e−03	6.07e−03	3.61e−03	7.52e−11	2.65e−14	3.77e−16	4.90e−15
	Std.	6.20e−03	1.06e−02	9.15e−03	6.02e−03	6.49e−11	1.39e−14	2.44e−16	1.83e−15
	+/=/–	=	+	+	+	+	+	−	
f_8	Mean	1.92e+01	4.17e+00	2.32e+02	5.84e+02	6.22e+01	8.49e+01	3.92e+01	7.80e+01
	Std.	5.89e+00	9.84e−01	4.44e+01	4.27e+01	8.16e+00	1.57e+01	6.47e+00	1.01e+01
	+/=/–	−	−	+	+	−	+	−	
f_9	Mean	2.77e+02	2.72e+02	3.30e+02	6.57e+02	1.36e+02	1.23e+02	1.45e+02	1.37e+02
	Std.	2.84e+01	2.04e+01	1.65e+02	7.80e+01	2.61e+01	2.77e+01	2.72e+01	1.91e+01
	+/=/–	+	+	+	+	=	=	=	
f_{10}	Mean	1.26e+02	1.49e+01	6.69e+03	1.01e+04	5.53e+03	1.46e+03	5.73e+02	5.99e+03
	Std.	1.97e+01	2.49e+00	1.25e+03	3.99e+03	7.98e+02	3.64e+02	2.32e+02	7.30e+02
	+/=/–	−	−	+	+	=	−	−	
f_{11}	Mean	2.05e+04	1.06e+04	3.03e+04	1.41e+04	1.53e+04	1.17e+04	1.22e+04	1.29e+04
	Std.	5.94e+02	6.56e+02	5.24e+02	1.21e+03	2.10e+03	1.25e+03	2.12e+03	1.27e+03
	+/=/–	+	−	+	+	+	=	=	
f_{12}	Mean	1.62e+00	2.91e−01	4.01e+00	6.80e−01	2.49e+00	1.79e+00	3.18e−01	8.51e−01
	Std.	1.51e−01	3.69e−02	1.74e−01	5.26e−02	1.30e−01	3.92e−01	1.22e−01	1.98e−01
	+/=/–	+	−	+	−	+	+	−	
f_{13}	Mean	4.67e−01	4.85e−01	6.53e−01	5.90e−01	5.33e−01	5.16e−01	4.71e−01	3.50e−01
	Std.	3.88e−02	5.58e−02	8.10e−02	8.60e−02	4.26e−02	7.23e−02	6.06e−02	4.41e−02
	+/=/–	+	+	+	+	+	+	+	
f_{14}	Mean	3.21e−01	2.88e−01	3.47e−01	3.67e−01	2.83e−01	2.80e−01	2.80e−01	2.61e−01
	Std.	1.97e−02	2.10e−02	3.77e−02	1.07e−01	2.40e−02	2.47e−02	1.74e−02	2.06e−02
	+/=/–	+	+	+	+	+	+	+	
f_{15}	Mean	4.09e+01	6.61e+01	8.70e+01	6.95e+01	2.26e+01	1.36e+01	1.59e+01	1.45e+01
	Std.	1.12e+01	1.82e+01	1.80e+01	7.08e+00	5.45e+00	2.08e+00	2.52e+00	2.62e+00
	+/=/–	+	+	+	+	+	=	=	

Table 7 Comparative results on functions $f_{16} - f_{30}$ with $D = 100$

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	IDDE	GPDE	CSDE
f_{16}	Mean	4.38e+01	4.14e+01	4.66e+01	4.21e+01	4.48e+01	4.25e+01	4.13e+01	4.27e+01
	Std.	3.77e−01	3.46e−01	2.57e−01	4.73e−01	3.32e−01	8.11e−01	8.40e−01	1.13e+00
	+/=/–	+	−	+	=	+	=	−	
f_{17}	Mean	1.50e+05	3.27e+05	4.82e+05	8.97e+04	2.49e+06	5.76e+05	1.01e+06	5.35e+05
	Std.	4.37e+04	1.63e+05	2.07e+05	4.39e+04	1.09e+06	2.19e+05	5.12e+05	1.94e+05
	+/=/–	−	−	=	−	+	=	+	
f_{18}	Mean	8.02e+02	6.84e+02	7.02e+02	1.50e+03	2.61e+02	2.07e+02	5.74e+02	8.20e+02
	Std.	8.63e+02	4.40e+02	7.69e+02	1.64e+03	3.12e+02	2.59e+02	1.08e+03	3.81e+02
	+/=/–	=	=	=	=	−	−	−	
f_{19}	Mean	8.39e+01	1.01e+02	9.83e+01	8.95e+01	9.02e+01	8.94e+01	8.91e+01	9.07e+01
	Std.	3.13e+01	4.27e+01	1.86e+01	3.56e+01	8.56e−01	1.63e+00	9.74e−01	1.73e+00
	+/=/–	=	=	+	=	=	=	−	
f_{20}	Mean	7.14e+02	1.09e+03	3.21e+03	4.54e+02	6.32e+03	1.00e+03	5.71e+02	5.30e+02
	Std.	2.09e+02	1.71e+03	1.17e+03	1.29e+02	1.63e+03	3.47e+02	8.60e+01	8.87e+01
	+/=/–	+	=	+	−	+	+	=	
f_{21}	Mean	7.28e+04	1.34e+05	1.87e+05	2.96e+04	1.98e+06	2.95e+05	3.85e+05	1.61e+05
	Std.	5.04e+04	8.18e+04	7.22e+04	1.54e+04	6.00e+05	1.55e+05	1.64e+05	9.18e+04
	+/=/–	−	=	=	−	+	+	+	
f_{22}	Mean	1.38e+03	1.41e+03	3.65e+03	2.69e+03	1.35e+03	9.86e+02	1.16e+03	1.58e+03
	Std.	2.94e+02	2.58e+02	9.48e+02	4.76e+02	3.96e+02	3.79e+02	2.59e+02	4.94e+02
	+/=/–	=	=	+	+	=	−	−	
f_{23}	Mean	3.48e+02	3.48e+02	3.48e+02	3.48e+02	3.48e+02	3.48e+02	3.48e+02	3.48e+02
	Std.	4.07e−13	6.44e−12	5.33e−05	3.19e−12	1.11e−04	6.80e−13	5.08e−13	2.53e−13
	+/=/–	=	=	=	=	=	=	=	
f_{24}	Mean	3.78e+02	3.99e+02	4.06e+02	4.52e+02	3.62e+02	3.63e+02	3.64e+02	3.70e+02
	Std.	6.30e+00	7.55e+00	8.35e+00	2.28e+01	1.69e+00	1.42e+00	2.65e+00	2.56e+00
	+/=/–	+	+	+	+	−	=	−	
f_{25}	Mean	2.00e+02	2.57e+02	2.45e+02	2.88e+02	2.49e+02	2.34e+02	2.39e+02	2.36e+02
	Std.	6.26e−14	8.37e+00	9.57e+00	1.66e+01	4.19e+00	5.56e+00	4.76e+00	4.35e+00
	+/=/–	−	+	+	+	+	=	+	
f_{26}	Mean	2.00e+02	2.00e+02	2.01e+02	2.00e+02	2.01e+02	1.40e+02	2.01e+02	2.00e+02
	Std.	2.49e−02	7.47e−03	1.10e−01	1.24e−02	2.09e−01	1.77e−01	1.77e−01	3.76e−02
	+/=/–	−	−	+	−	+	=	+	
f_{27}	Mean	1.43e+03	1.37e+03	1.40e+03	3.12e+03	3.06e+02	4.19e+02	3.74e+02	3.28e+02
	Std.	1.01e+02	2.15e+02	1.63e+02	2.39e+02	1.24e+01	1.13e+01	5.91e+01	2.15e+01
	+/=/–	+	+	+	+	−	+	+	
f_{28}	Mean	3.20e+03	4.72e+03	2.91e+03	1.39e+04	2.20e+03	2.03e+03	2.08e+03	1.88e+03
	Std.	3.34e+02	6.34e+02	3.15e+02	1.92e+03	4.63e+01	7.03e+01	1.78e+02	3.03e+02
	+/=/–	+	+	+	+	+	+	+	
f_{29}	Mean	1.37e+03	1.45e+03	1.98e+03	1.23e+03	2.81e+03	1.78e+03	1.91e+03	1.69e+03
	Std.	2.51e+02	1.85e+02	4.05e+02	2.18e+02	4.80e+02	1.36e+02	1.41e+02	1.98e+02
	+/=/–	−	−	+	−	+	=	+	
f_{30}	Mean	8.17e+03	8.59e+03	6.01e+03	7.45e+03	9.53e+03	7.42e+03	8.90e+03	6.22e+03
	Std.	9.43e+02	1.58e+03	9.47e+02	2.02e+03	1.25e+03	8.80e+02	8.01e+02	6.38e+02
	+/=/–	+	+	=	+	+	+	+	

Table 8 Comparative results on real-world problems $rf_1 - rf_4$

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	IDDE	GPDE	CSDE
rf_1	Mean	1.05e+00	9.95e+00	6.35e+00	1.26e+01	4.68e−01	2.71e+00	1.34e+00	9.36e−01
	Std.	3.66e+00	4.74e+00	6.87e+00	4.41e+00	2.34e+00	5.00e+00	3.72e+00	3.03e+00
	+/=/–	−	+	+	+	−	−	−	
rf_2	Mean	1.93e+00	1.79e+00	2.38e+00	1.95e+00	1.78e+00	1.52e+00	1.63e+00	1.79e+00
	Std.	8.64e−02	9.98e−02	9.90e−02	1.56e−01	1.69e−01	1.42e−01	2.19e−01	2.13e−01
	+/=/–	+	=	+	+	=	−	−	
rf_3	Mean	1.76e−04	4.38e−09	4.63e−14	1.09e+01	7.07e+00	4.97e+00	2.82e+00	4.47e−03
	Std.	4.55e−04	1.49e−08	7.08e−14	6.47e+00	4.25e+00	6.42e+00	5.11e+00	6.12e−03
	+/=/–	−	−	−	+	+	+	+	
rf_4	Mean	0.00e+00	0.00e+00	0.00e+00	0.00e+00	4.63e+00	7.78e−05	0.00e+00	0.00e+00
	Std.	0.00e+00	0.00e+00	0.00e+00	0.00e+00	7.68e+00	3.89e−04	0.00e+00	0.00e+00
	+/=/–	=	=	=	=	+	=	=	

Table 9 Statistical results on all test functions and real-world problems

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	IDDE	GPDE	CSDE
30-D	+/=/–	23/4/3	21/6/3	26/1/3	25/3/2	16/10/4	13/12/5	14/11/5	−−
50-D	+/=/–	22/4/4	19/2/9	23/5/2	19/4/7	17/10/3	15/10/5	14/8/8	−−
100-D	+/=/–	16/6/8	12/8/10	25/5/0	16/5/9	19/7/4	16/11/3	13/7/10	−−
Unimodal	+/=/–	6/1/2	4/2/3	7/1/1	2/0/7	8/1/0	8/1/0	8/1/0	−−
Multimodal	+/=/–	28/5/6	19/5/15	37/0/2	32/4/3	20/12/7	23/13/3	11/12/16	−−
Hybrid	+/=/–	12/4/2	12/5/1	13/5/0	9/4/5	12/5/1	5/7/6	7/6/5	−−
Composition	+/=/–	15/4/5	17/4/3	17/5/2	17/4/3	12/9/3	8/12/4	15/7/2	−−
Real-world	+/=/–	1/1/2	1/2/1	2/1/1	3/1/0	2/1/1	1/1/2	1/1/2	−−
Total	+/=/–	62/15/17	53/18/23	76/12/6	63/13/18	54/28/12	45/34/15	42/27/25	−−

indicates the superiority of CSDE when dealing with complex optimization problem. Lastly, we find out that CSDE has no worst "Mean" on the 94 functions, which means that CSDE is a reliable algorithm for handling various problems with different dimensions and characteristics.

For further illustration, we show the convergence graphs of the eight compared algorithms based on their mean values over 50 runs on 12 benchmarks functions when $D = 50$, and the results are exhibited in Figs. 2 and 3. As evident from the convergence characteristics, we can see that in the early stage of the whole evolutionary process, the performance of CSDE is not eye-catching, but in the later stage, CSDE has the best overall performance among the contestant algorithms. We restrained from giving all the graphs in order to save space.

The involved control parameters usually effect the algorithmic performance. In fact, except for the population size NP, there is only one pre-given control parameter FP. In order to observe the influence degree caused by different values of periodic adjustment parameter FP, we actually obtain the results of all the test functions with different parameter settings, which are listed in Table 10, on different dimensional problems. For saving space and providing visible results, we adopt the convergence graphs of CSDEs with different FP on 12 benchmarks functions when $D = 50$ to show the influence of parameter FP, and the convergence graphs are plotted in Figs. 4 and 5.

From Figs. 4 and 5, it is easy to see that the value of periodic adjustment parameter FP indeed affects the performance of CSDE, but the influence is very slight, which also shows that parameter FP is robust in CSDE. The users have a greater degree of freedom to set the value of FP when dealing with different problems.

6 Conclusions

It is well known that using proper mutation operator and control parameters can effectively improve the performance of DE. Therefore, based on the core guideline of maintaining a proper balance between the global exploration ability and local exploitation ability during the optimization routine, we have proposed a novel DE variant (called CSDE) in this

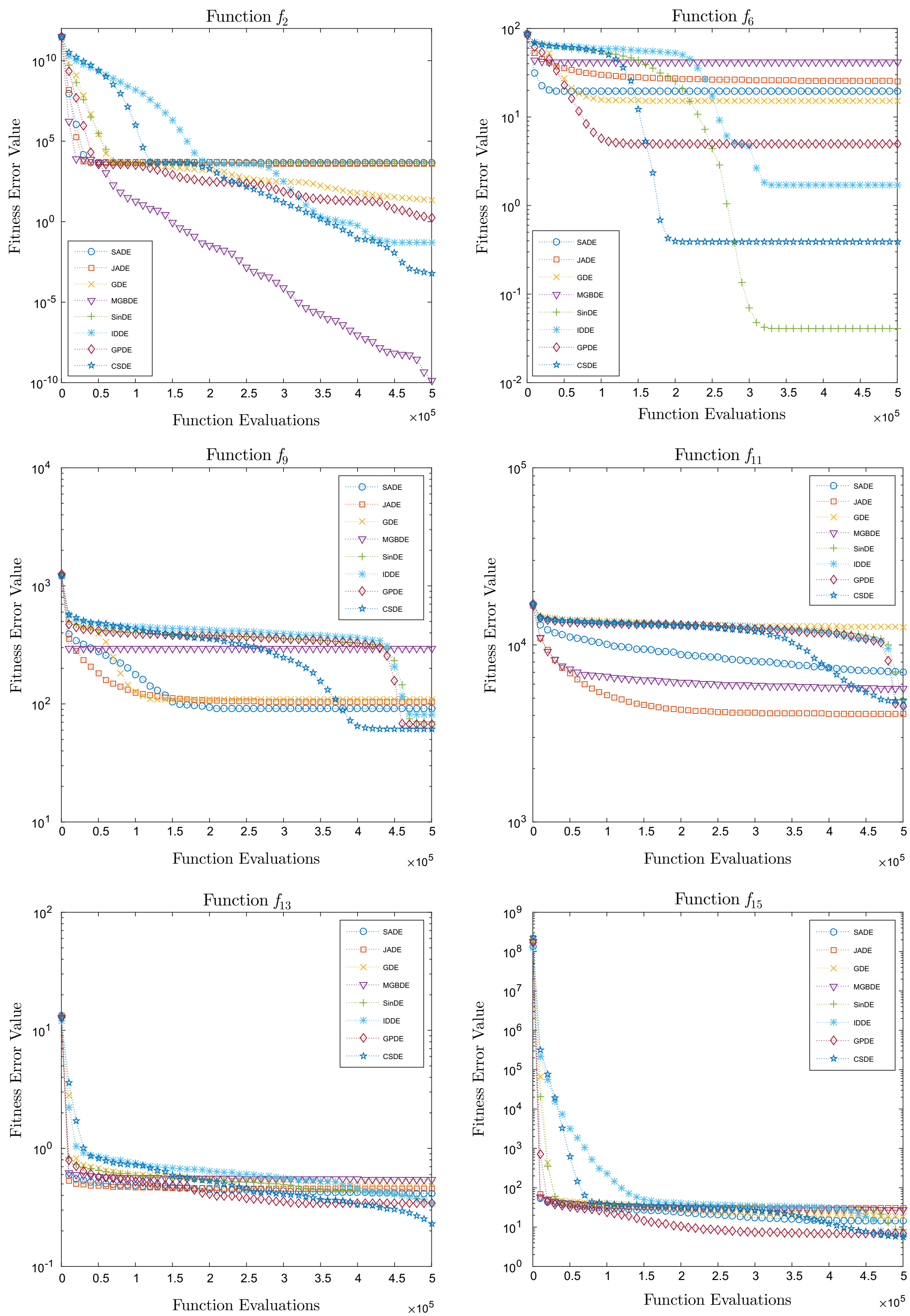

Fig. 2 Convergence graphs (mean curves) for eight algorithms on functions f_2, f_6, f_9, f_{11}, f_{13} and f_{15} with $D = 50$ over 50 independent runs

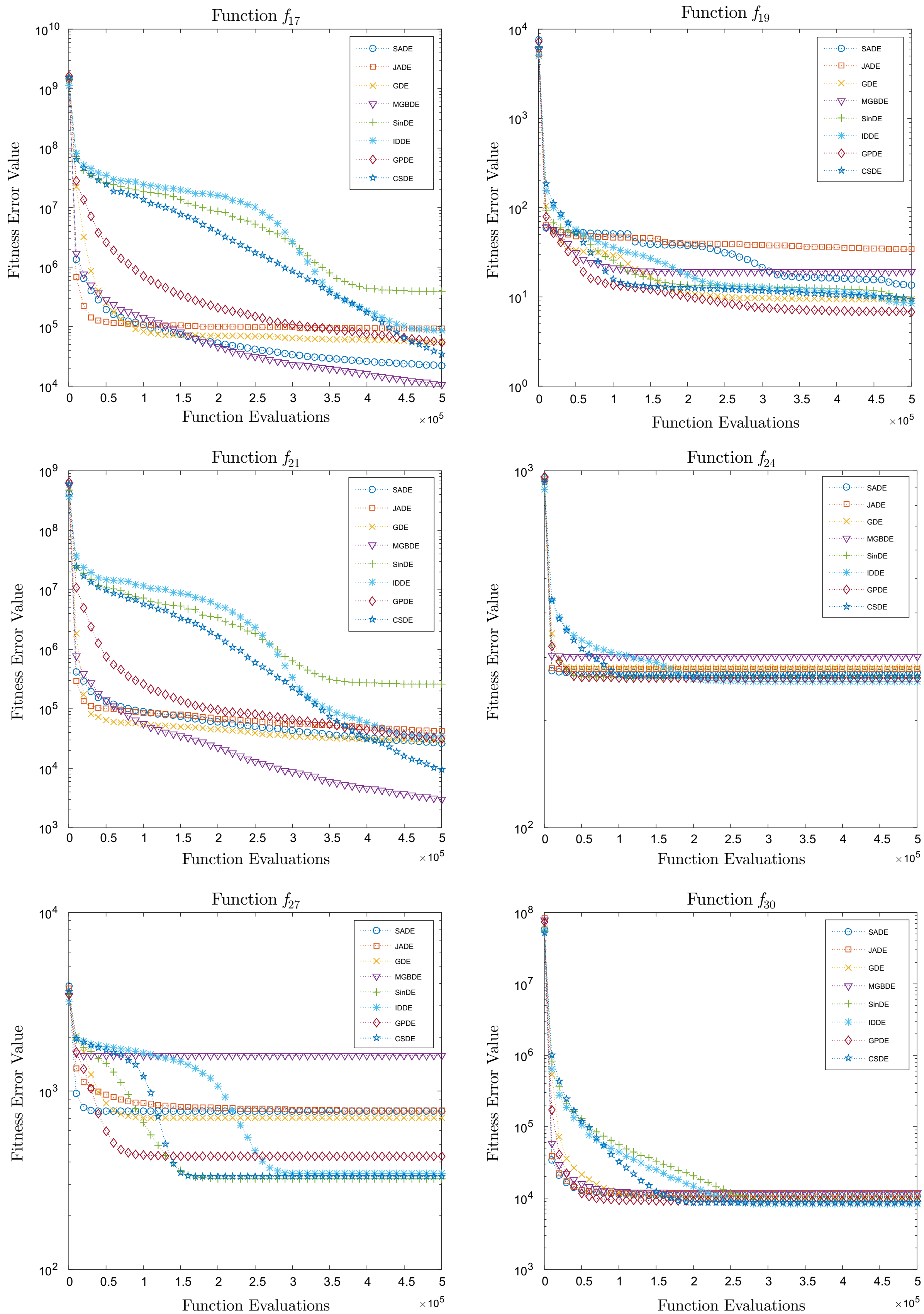

Fig. 3 Convergence graphs (mean curves) for eight algorithms on functions f_{17}, f_{19}, f_{21}, f_{24}, f_{27} and f_{30} with $D = 50$ over 50 independent runs

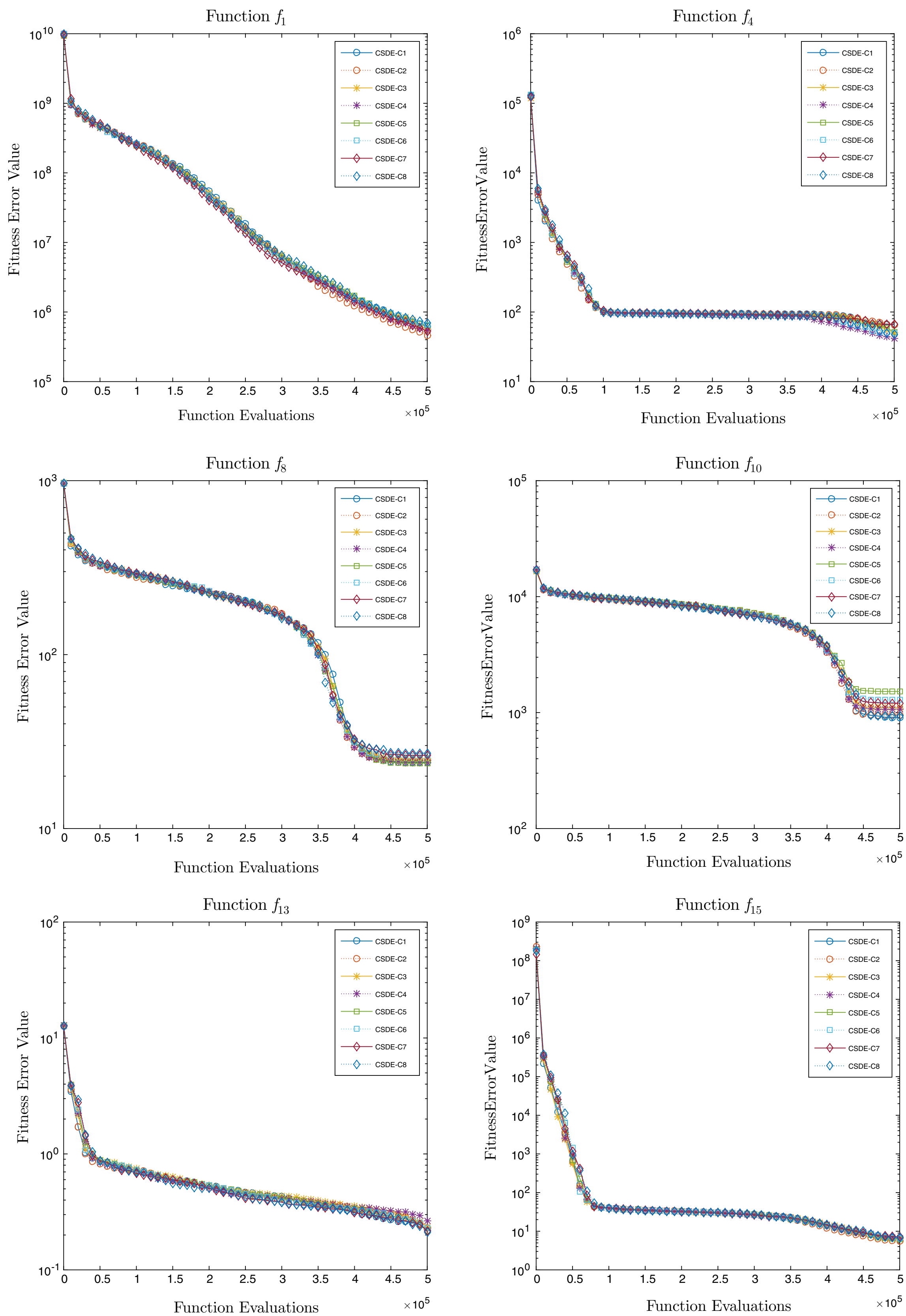

Fig. 4 Convergence graphs (mean curves) for the CSDE with different values of parameter FP on functions f_1, f_4, f_8, f_{10}, f_{13} and f_{15} with $D = 50$ over 50 independent runs

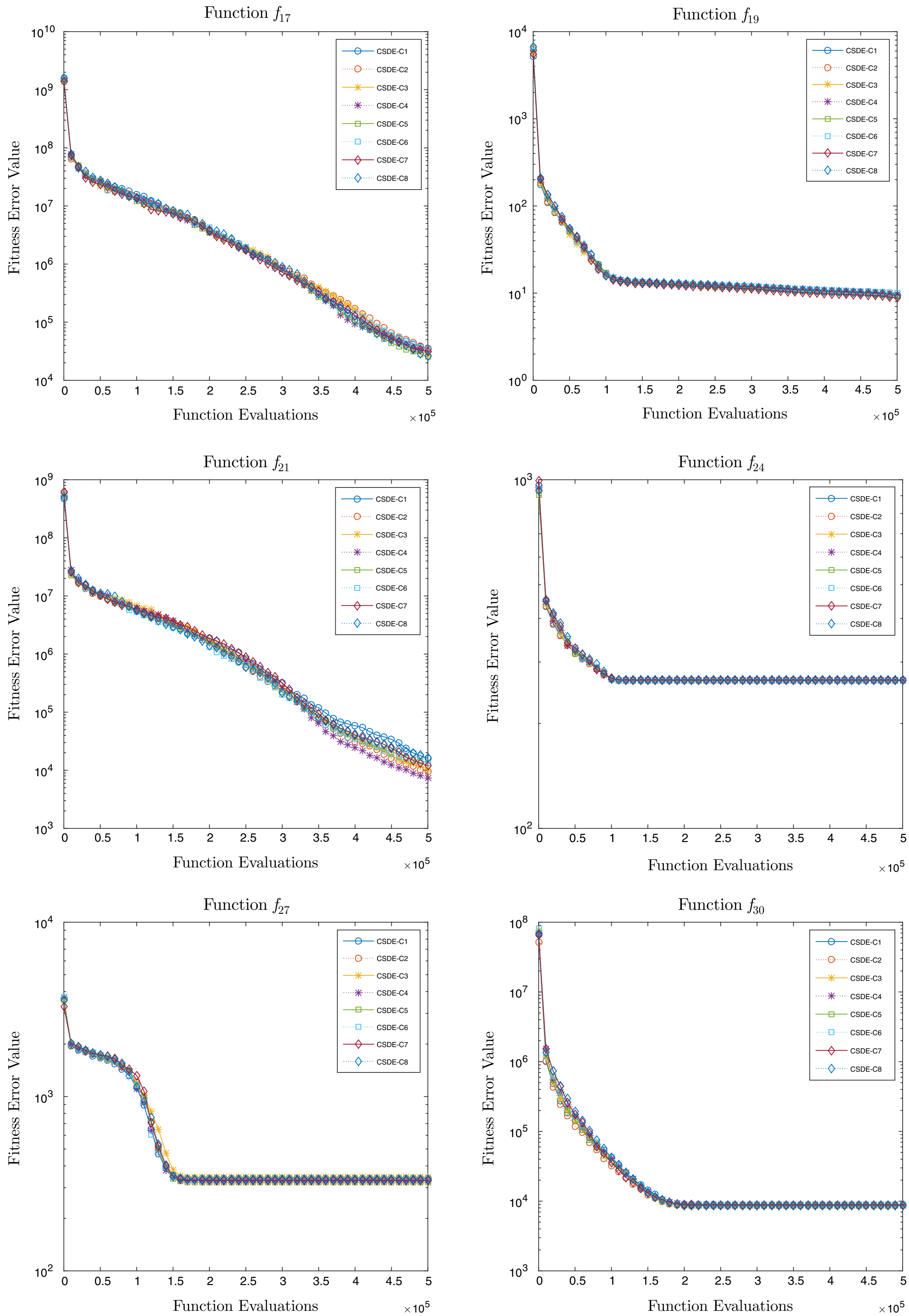

Fig. 5 Convergence graphs (mean curves) for the CSDE with different values of parameter FP on functions f_{17}, f_{19}, f_{21}, f_{24}, f_{27} and f_{30} with $D = 50$ over 50 independent runs

Table 10 Parameter setting of different CSDEs

Par.	CSDE-C1	CSDE-C2	CSDE-C3	CSDE-C4	CSDE-C5	CSDE-C6	CSDE-C7	CSDE-C8
FP	100	200	300	400	500	600	800	1000

paper. To be specific, in CSDE, two mutation operators with different features are used to generate the mutant vector; both the two mutation operators applied the *p*best concept, but one has good global exploration ability and the other has all-right local exploitation ability. Furthermore, a coordination mechanism based on the historical success rate is used to adjust the two mutation operators. To design an unexceptionable control parameters adjustment mechanism, a periodic function based on modulo operation, an individual-independence macro-control function and an individual-dependence function based on fitness value information are used to produce the adaptive scaling factor and crossover rate. The experimental results on 30 benchmark functions and 4 real-world problems show that CSDE is better than seven state-of-the-art DE variants on most cases. In addition, the parameter analysis indicates that the periodic adjustment parameter involved in CSDE is robust.

Acknowledgements This work was supported by the National Natural Science Foundation of China under Grant Nos. 71701187, 61702389, Research Project of Zhejiang Education Department under Grant No. Y201738184 and Yanta Scholars Foundation of Xi'an University of Finance and Economics.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Human and animals rights This article does not contain any studies with human participants or animals performed by any of the authors.

References

Al-Dabbagh R, Neri F, Idris N, Baba M (2018) Algorithmic design issues in adaptive differential evolution schemes: review and taxonomy. Swarm Evol Comput 43:284–311

Arce F, Zamora E, Sossa H, Barróna R (2018) Differential evolution training algorithm for dendrite morphological neural networks. Appl Soft Comput 68:303–313

Črepinšek M, Liu SH, Mernik M (2013) Exploration and exploitation in evolutionary algorithms: a survey. ACM Comput Surv 45(3):1–33

Cui L, Li G, Zhu Z, Wen Z, Lu N, Lu J (2018) A novel differential evolution algorithm with a self-adaptation parameter control method by differential evolution. Soft Comput 22:6171C6190

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

Das S, Suganthan PN (2011) Problem definitions and evaluation criteria for CEC 2011 competition on testing evolutionary algorithms on real world optimization problems. Jadavpur University, Kolkata, India, and Nanyang Technological University, Singapore

Draa A, Bouzoubia S, Boukhalfa I (2015) A sinusoidal differential evolution algorithm for numerical optimisation. Appl Soft Comput 27:99C126

Fu CM, Jiang C, Chen GS, Liu QM (2017) An adaptive differential evolution algorithm with an aging leader and challengers mechanism. Appl Soft Comput 57:60–73

García-Martínez C, Lozano M, Herrera F, Molina D, Sánchez A (2008) Global and local real-coded genetic algorithms based on parent-centric crossover operators. Eur J Oper Res 185:1088–1113

Gong W, Cai Z (2013) Differential evolution with ranking-based mutation operators. IEEE Trans Cybern 43(6):2066–2081

Halder U, Das S, Maity D (2013) A cluster-based differential evolution algorithm with external archive for optimization in dynamic environments. IEEE Trans Cybern 43(3):881–897

Han MF, Liao SH, Chang JY, Lin CT (2013) Dynamic group-based differential evolution using a self-adaptive strategy for global optimization problems. Appl Intell 39(1):41–56

Herrera F, Lozano M (2000) Gradual distributed real-coded genetic algorithms. IEEE Trans Evol Comput 4:43–63

Islam SM, Das S, Ghosh S, Roy S, Suganthan PN (2012) An adaptive differential evolution algorithm with novel mutation and crossover strategies for global numerical optimization. IEEE Trans Syst Man Cybern Part B Cybern 42(2):482–500

Li W, Li SN, Chen ZX, Zhong L, Ouyang CT (2019) Self-feedback differential evolution adapting to fitness landscape characteristics. Soft Comput 23:1151–1163

Liang JJ, Qu BY, Suganthan PN (2013) Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization. Zhengzhou University, China, and Nanyang Technological University, Singapore

Liu YK, Chen YJ, Yang GQ (2018) Developing multi-objective equilibrium optimization method for sustainable uncertain supply chain planning problems. IEEE Trans Fuzzy Syst. https://doi.org/10.1109/TFUZZ.2018.2851508

Mohamed AW, Sabry HZ (2012) Constrained optimization based on modified differential evolution algorithm. Inf Sci 194:171–208

Pereira W, Soares M (2015) Horizontal multilayersoil parameter estimation through differential evolution. IEEE Trans Power Deliv 31(2):622–629

Qin AK, Huang VL, Suganthan PN (2009) Differential evolution algorithm with strategy adaptation for global numerical optimization. IEEE Trans Evol Comput 13(2):398C417

Qu BY, Suganthan PN, Liang JJ (2012) Differential evolution with neighborhood mutation for multimodal optimization. IEEE Trans Evol Comput 16(5):601–614

Sarkar S, Das S, Chaudhuri S (2016) Hyper-spectral image segmentation using Rényi entropy based multi-level thresholding aided with differential evolution. Expert Syst Appl 50:120–129

Sarker RA, Elsayed SM, Ray T (2014) Differential evolution with dynamic parameters selection for optimization problems. IEEE Trans Evol Comput 18(5):689C707

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

Sun G, Liu YK, Lan YF (2010) Optimizing material procurement planning problem by two-stage fuzzy programming. Comput Ind Eng 58:97–107

Sun G, Peng J, Zhao R (2018) Differential evolution with individual-dependent and dynamic parameter adjustment. Soft Comput 22:5747–5773

Sun G, Lan Y, Zhao R (2019) Differential evolution with Gaussian mutation and dynamic parameter adjustment. Soft Comput 23:1615–1642

Tang L, Zhao Y, Liu J (2014) An improved differential evolution algorithm for practical dynamic scheduling in steel making continuous casting production. IEEE Trans Evol Comput 18(2):209–225

Tang L, Dong Y, Liu J (2015) Differential evolution with an individual-dependent mechanism. IEEE Trans Evol Comput 19(4):560C574

Tayarani-N M, Yao X, Xu H (2015) Meta-heuristic algorithms in car engine design: a literature survey. IEEE Trans Evol Comput 19(5):609–629

Tian M, Gao X, Dai C (2017) Differential evolution with improved individual-based parameter setting and selection strategy. Appl Soft Comput 56:286–297

Wang H, Rahnamayan S, Sun H, Omran MGH (2013) Gaussian bare-bones differential evolution. IEEE Trans Cybern 43(2):634–647

Wang J, Zhang W, Zhang J (2016) Cooperative differential evolution with multiple populations for multiobjective optimization. IEEE Trans Cybern 46(12):2848–2861

Yu WJ, Shen M, Chen WN, Zhan ZH, Gong YJ, Lin Y (2014) Differential evolution with two-level parameter adaption. IEEE Trans Cybern 44(7):1080C1099

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

Zhao XC, Xu GZ, Rui L, Liu DY, Liu HP, Yuan JH (2019) A failure remember-driven self-adaptive differential evolution with top-bottom strategy. Swarm Evol Comput 45:1–14

Zheng LM, Liu L, Zhang SX, Zheng SY (2018) Enhancing differential evolution with interactive information. Soft Comput 22:7919–7938

Zhou Y, Li X, Gao L (2013) A differential evolution algorithm with intersect mutation operator. Appl Soft Comput 13:390–401