Soft Comput (2019) 23:1615–1642
https://doi.org/10.1007/s00500-017-2885-z

METHODOLOGIES AND APPLICATION

Differential evolution with Gaussian mutation and dynamic parameter adjustment

Gaoji Sun[1] · Yanfei Lan[2] · Ruiqing Zhao[2]

Published online: 14 November 2017

Abstract Differential evolution (DE) is a remarkable evolutionary algorithm for global optimization over continuous search space, whose performance is significantly influenced by its mutation operator and control parameters (scaling factor and crossover rate). In order to enhance the performance of DE, we adopt a novel Gaussian mutation operator and a modified common mutation operator to collaboratively produce new mutant vectors, and employ a periodic function and a Gaussian function to generate the required values of scaling factor and crossover rate, respectively. In the proposed variant of DE (denoted by GPDE), the two adopted mutation operators are adaptively applied to generate the corresponding mutant vector of each individual based on their own cumulative scores, the periodic scaling factor can provide a better balance between exploration ability and exploitation ability, and the Gaussian function-based crossover rate will possess fluctuant value, which possibly enhance the population diversity. To verify the performance of proposed GPDE, a suite of thirty benchmark functions and four real-world problems are applied to conduct the simulation experiment. The simulation results demonstrate that the proposed GPDE performs significantly better than five state-of-the-art DE variants and other two meta-heuristics algorithms.

Keywords Differential evolution · Gaussian mutation · Dynamic parameter adjustment · Evolutionary computation · Global optimization

1 Introduction

Differential evolution (DE), proposed by Storn and Price (Storn and Price 1997), is a simple yet efficient evolutionary algorithm (EA) for the global numerical optimization. Due to its simple structure and ease of use, DE has been successfully applied to solve many real-world problems, including decision-making (Zhang et al. 2010), dynamic scheduling (Tang et al. 2014), parameter optimization (Gong and Cai 2014), spam detection (Idris et al. 2014), system fault diagnosis (Zhao et al. 2014), motion estimation (Cuevas et al. 2013) and so on. More details on the recent research about DE can be found in the literature reviews (Neri and Tirronen 2010; Das and Suganthan 2011a) and the references therein.

Three evolutionary operators (mutation, crossover and selection) and three control parameters (population size, scaling factor and crossover rate) are included in the original DE algorithm, which have significant influence on its performance. Thus, many researchers have been engaged in improving DE by designing new evolutionary operators, combining multiple operators and adopting adaptive or self-adaptive strategies for those control parameters. Although various variants of DE have been proposed, there still exists a big room for improvement, owing to the thorny work of balancing the global exploration ability and local exploitation ability (Lin and Gen 2009; Črepinšek et al. 2013).

Using Gaussian function can randomly produce new solutions around a given position, which may provide an excellent exploitation ability. Meanwhile, periodic or fluctuant parameter adjustment strategy possibly can achieve a good balance

Communicated by V. Loia.

✉ Gaoji Sun
gsun@zjnu.edu.cn

✉ Yanfei Lan
lanyf@tju.edu.cn

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

[2] Institute of Systems Engineering, Tianjin University, Tianjin 300072, China

between the exploitation operation around the already-found good solutions and the exploration operation for seeking out non-visited regions in the search space. Inspired by the above observations, we design a novel Gaussian mutation operator (which, respectively, takes the position of the best individual among three randomly selected individuals and the distance between the other two as the mean and standard deviation of Gaussian distribution) and a modified common mutation operator (denoted by DE/rand-worst/1) to collaboratively produce the new potential position for every individual, and the collaborative rule between them relies on their own cumulative scores during the evolutionary process. In addition, the scaling factor adopts a cosine function to realize the objective of adjusting its value periodically, and the crossover rate employs a Gaussian function to dynamically adjust the population diversity during the evolutionary process. At last, a novel DE variant is proposed via combining the above-mentioned Gaussian mutation operator, DE/rand-worst/1 and parameter adjustment strategies, which is called GPDE for short. A suite of 30 benchmark functions with different dimensions and four real-world optimization problems are applied to evaluate the performance of GPDE, and its performance is compared to five excellent DE variants and two up-to-date meta-heuristics algorithms. The comparative results show that GPDE obviously outperforms the seven compared algorithms. Moreover, the parameter analysis expresses that the adopted control parameters within GPDE are robust.

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 GPDE algorithm and its overall procedure. Section 5 presents the comparison between GPDE and seven compared algorithms. Section 6 draws the conclusions.

2 Differential evolution

DE is a population-based stochastic search algorithm which simulates the natural evolutionary process via mutation, crossover and selection to move its population toward the global optimum. The DE algorithm mainly contains the following four operations.

2.1 Initialization operation

Similar to other EAs, DE searches for a global optimum in the D-dimensional real parameter space with a population of vectors $\boldsymbol{x}_i = [x_{i,1}, x_{i,2}, \ldots, x_{i,D}]$, $i = 1, 2, \ldots, \text{NP}$, where NP is the population size. An initial population should cover the entire search space by uniformly randomizing individuals between the prescribed lower bounds $\boldsymbol{L} = \left[L_1, L_2, \ldots, L_D\right]$ and upper bounds $\boldsymbol{U} = \left[U_1, U_2, \ldots, U_D\right]$. The jth component of the ith individual can be initialized as follows,

$$x_{i,j} = L_j + \text{rand}[0,1] \cdot \left(U_j - L_j\right), \tag{1}$$

where rand[0, 1] represents a uniformly distributed random number within the interval [0, 1] and is used throughout the paper.

2.2 Mutation operation

After the initialization operation, DE employs a mutation operation to produce a mutant vector $\boldsymbol{v}_i = \left[v_{i,1}, v_{i,2}, \ldots, v_{i,D}\right]$ for each target vector $\boldsymbol{x}_i$. The followings are five most frequently used mutation operators implemented in various DE algorithms.

(1) DE/rand/1

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

(2) DE/best/1

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

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

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

(4) DE/best/2

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

(5) DE/rand/2

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

The indices r_1, r_2, r_3, r_4 and r_5 in the above equations are mutually exclusive integers randomly generated from set $\{1, 2, \ldots, \text{NP}\}$ and are also different from the index i. The parameter F is called the scaling factor, which is a positive real number for scaling the difference vectors. The vector $\boldsymbol{x}_{\text{best}} = (x_{\text{best},1}, x_{\text{best},2}, \ldots, x_{\text{best},D})$ is the best individual in the current population.

2.3 Crossover operation

After the mutation operation, DE performs a binomial crossover operator on target vector $\boldsymbol{x}_i$ and its corresponding mutant vector $\boldsymbol{v}_i$ to produce a trial vector $\boldsymbol{u}_i = \left[u_{i,1}, u_{i,2}, \ldots, u_{i,D}\right]$. This process can be expressed as

$$u_{i,j} = \begin{cases} v_{i,j}, & \text{if } \left(\text{rand}[0,1] \le \text{CR or } j = j_{\text{rand}}\right) \\ x_{i,j}, & \text{otherwise.} \end{cases} \tag{7}$$

The crossover rate CR is a user-specified constant within the interval (0, 1) in original DE, which controls the fraction of trial vector components inherited from the mutant vector. The index j_{rand} is an integer randomly chosen from set $\{1, 2, \ldots, D\}$, which is used to ensure that the trial vector has at least one component different from the target vector.

2.4 Selection operation

After the crossover operation, a selection operation is executed between the trial vector and the target vector according to their fitness values $f(\cdot)$, and the better one will survive to the next generation. Without loss of generality, we only consider minimization problems. Specifically, the selection operator can be expressed as follows:

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

From 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 the past decades, many meta-heuristic algorithms have been proposed, such genetic algorithm (Goldberg 1989), differential evolution (Storn and Price 1997), particle swarm optimization (Kennedy et al. 2001), ant colony optimization (Dorigo and Blum 2005), joint operations algorithm (Sun et al. 2016) and so on. These meta-heuristic algorithms have been successfully applied in various fields, such as production planning (Lan et al. 2012), procurement planning (Sun et al. 2010), location problems (Wang and Watada 2012), workforce planning (Yang et al. 2017). Among these meta-heuristic algorithms, DE has shown outstanding performance in solving many test functions and real-world problems, but its performance highly depends on the selected evolutionary operators and the values of control parameters. To overcome these drawbacks, many variants have been proposed to improve the performance of DE. In this section, we only provide a brief overview of the enhanced approaches which is related to our work.

There are lots of researchers who tried to enhance DE via designing new mutation operators or combining multiple operators. Qin et al. (2009) proposed a self-adaptive DE (SADE) which focuses on the mutation operator selection and crossover rate of DE. Zhang and Sanderson (2009) presented a self-adaptive DE with optional external archive (JADE) which employs a new mutation operator called "DE/current-to-*p*best." Han et al. (2013) introduced a group-based DE variant (GDE) which divides the population into two groups and each group employs a different mutation operator. Wang et al. (2013) proposed a modified Gaussian bare-bones DE variant (MGBDE) which combines two mutation operators, and one of the mutation operators is designed based on Gaussian distribution. Das et al. (2009) presented two kinds of topological neighborhood models and embedded them into the mutation operators of DE. Gong et al. (2011a) introduced a simple strategy adaptation mechanism (SaM) which can be used for coordinating different mutation operators. Many other DE variants also adopted new designed mutation operator or multi-mutation operator strategies with different searching features, such as NDi-DE (Cai and Wang 2013), MS-DE (Wang et al. 2014), CoDE (Wang et al. 2011), HLXDE (Cai and Wang 2015), MDE_*p*BX (Islam et al. 2012), AdapSS-JADE (Gong et al. 2011b), IDDE (Sun et al. 2017).

Some other researchers applied parameter adjustment to improve the performance of DE. For instance, Draa et al. (2015) introduced sinusoidal differential evolution (SinDE), which adopts two sinusoidal formulas to adjust the values of scaling factor and crossover rate. Brest et al. (2006) proposed a self-adaptive scheme for the DE's control parameters. Liu and Lampinen (2005) applied fuzzy logic controllers to adapt the value of crossover rate. Zhu et al. (2013) adopted an adaptive population tuning scheme to enhance DE. Ghosh et al. (2011) introduced a control parameter adaptation strategy, which is based on the fitness values of individuals in DE population. Yu et al. (2014) proposed a two-level adaptive parameter control strategy, which is based on the optimization states and the fitness values of individuals. Sarker et al. (2014) introduced a new mechanism to dynamically select the best performing combinations of control parameters, which is based on the success rate of each parameter combination. Karafotias et al. (2015) provided a comprehensive overview about the parameter control in evolutionary algorithms.

Actually, many aforementioned references simultaneously utilize new evolutionary operators and adaptive control parameters to enhance the performance of DE, including SADE (Qin et al. 2009), JADE (Zhang and Sanderson 2009), MGBDE (Wang et al. 2013) and CoDE (Wang et al. 2011). In addition, Mallipeddi et al. (2011) employed a pool of distinct mutation operators along with a pool of values for each control parameter which coexists and competes to produce offsprings during the evolutionary process. Yang et al. (2015) proposed a mechanism named auto-enhanced population diversity to automatically enhance the performance of DE, which is based on the population diversity at the dimensional level. Biswas et al. (2015) presented an improved information-sharing mechanism among the individuals to enhance the niche behavior of DE. 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. However, these DE variants still cannot resolve the problems of premature convergence or stagnation when handling complex optimization problems.

4 Description of GPDE

In this section, we firstly provide a detailed description of the new Gaussian mutation operator, the modified common mutation operator and the cooperative rule between them, and then summarize the overall procedure of GPDE.

4.1 Gaussian mutation operator

Gaussian distribution is very important and often used in statistics and natural sciences to represent real-valued random variable, which can be denoted by $N(\mu, \sigma^2)$, where μ and σ are its mean and standard deviation, respectively. It is well known that there is 3-σ rule which exists in Gaussian distribution. Specifically, about 68% of the values drawn from the Gaussian distribution $N(\mu, \sigma^2)$ are within the interval $[\mu - \sigma, \mu + \sigma]$; about 95% of the values lie within the interval $[\mu - 2\sigma, \mu + 2\sigma]$; and about 99.7% are within the interval $[\mu - 3\sigma, \mu + 3\sigma]$.

The 3-σ rule of Gaussian distribution provides a wonderful chance to control the hunting zone which depends on the requirement of considered problem. Actually, Gaussian distribution has been widely used to adjust the values of control parameters, such as SADE (Qin et al. 2009), MGBDE (Wang et al. 2013), DEGL (Das et al. 2009) and MDE_pBE (Islam et al. 2012), but rarely applied to generate new mutation operator. In order to take full advantage of Gaussian distribution, we propose the following novel mutation operator which combines crossover operator to directly produce the new trial vector (denoted by $\boldsymbol{u}_i^g = (u_{i,1}^g, u_{i,2}^g, \ldots, u_{i,D}^g)$) for the ith individual, $i = 1, 2, \ldots, \mathrm{NP}$,

$$u_{i,j}^g = \begin{cases} N\left(x_{r_1,j}, \left(x_{r_2,j} \quad - x_{r_3,j}\right)^2\right), & \text{if } \left(j = j_{\text{rand}}\right. \\ \quad \text{or rand}[0,1] \leq \mathrm{CR}_t^i\left.\right), \\ x_{i,j}, & \text{otherwise}, \end{cases} \tag{9}$$

where the indices r_1, r_2, r_3 are mutually exclusive integers randomly generated from the set $\{1, 2, \ldots, \mathrm{NP}\}$ and are also different from the base index i. Note that the r_1th individual is the best one among the three randomly selected individuals, and the novel Gaussian mutation operator $N\left(x_{r_1,j}, \left(x_{r_2,j} - x_{r_3,j}\right)^2\right)$ in formula (9), respectively, takes the position $x_{r_1,j}$ of the best one and the distance $|x_{r_2,j} - x_{r_3,j}|$ between the other two as the mean and standard deviation; meanwhile, it will only be executed when meeting the triggering condition $\left(j = j_{\text{rand}} \text{ or rand}[0,1] \leq \mathrm{CR}_t^i\right)$, which means that the proposed Gaussian mutation operator only transfers a certain proportional dimensions of each corresponding individual to new positions around the best selected one. Furthermore, if the new position of one dimension is within one standard deviation $|x_{r_2,j} - x_{r_3,j}|$ away from the mean $x_{r_1,j}$, we can call that it executes the exploitation operation in this dimension; otherwise, it carries out the exploration operation. Therefore, according to the 3–σ rule, the designed Gaussian mutation operator can simultaneously conduct exploitation and exploration works, especially the former one. In addition, dynamic parameter CR_t^i expresses the crossover rate of the ith individual in the tth generation, which can be computed by

$$\mathrm{CR}_t^i = N(0.5, V), \quad i = 1, 2, \ldots, \mathrm{NP}, \ t = 1, 2, \ldots, T, \tag{10}$$

where V indicates the variance of Gaussian distribution $N(0.5, V)$, and it is applied to control the fluctuation of crossover rate, and T is the maximum allowable generation. It should be pointed out that V is a user-specified constant and it has to simultaneously ensure that the value of dynamic crossover rate CR_t^i has a certain extent of fluctuation and almost falls into the range [0, 1], and thus its reasonable interval is [0.01, 0.1]. Actually, the individuals employing different crossover rates in the same generation can potentially enhance the population diversity.

4.2 DE/rand-worst/1

In mutation operator, 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. In a general way, a better base individual has a higher probability to produce better offsprings, a more proper direction induces a more efficient searching behavior of the population, and a periodic scaling factor has potential advantage in balancing exploration ability and exploitation ability. Based on the aforementioned considerations, we incorporate the fitness information of selected individuals and a periodic scaling factor to modify the most popular mutation operator (DE/rand/1). The obtained modified mutation operator DE/rand/1 (denoted by DE/rand-worst/1) which combines crossover operator is applied to directly produce the new trial vector (denoted by $\boldsymbol{u}_i^m = (u_{i,1}^m, u_{i,2}^m, \ldots, u_{i,D}^m)$, $i = 1, 2, \ldots, \mathrm{NP}$) of each individual, the corresponding formula can be described as follows,

$$u_{i,j}^m = \begin{cases} x_{r'_1,j} + F_t \cdot (x_{r'_2,j} - x_{r'_3,j}), & \text{if } \left(j = j_{\text{rand}} \text{ or}\right. \\ \quad \text{rand}[0,1] \leq \mathrm{CR}_t^i\left.\right), \\ x_{i,j}, & \text{otherwise}, \end{cases} \tag{11}$$

where the indices r_1', r_2' and r_3' are mutually exclusive integers randomly chosen from the set $\{1, 2, \ldots, \mathrm{NP}\}$, which are also different from the index i. Moreover, the r_3'th individual is the worst one among the three randomly selected individuals, which can ensure that the base individual is not the worst one and the searching direction is relatively better. In addition, note that the mutation operator DE/rand-worst/1 in formula (11) has the same triggering condition with Gaussian mutation operator in formula (9). About the periodic scaling factor, we apply a cosine function to realize the periodic adjustment strategy, which can be expressed via the following formula,

$$F_t = \left|\cos(t \cdot \mathrm{FR} \cdot \pi)\right|, \tag{12}$$

where F_t is the value of scaling factor in the tth generation, and FR represents the frequency of cosine function, which is a user-specified constant and applied to adjust the turnover rate between the exploration and exploitation operations. Usually, a smaller frequency FR corresponds to a smaller turnover rate.

4.3 Cooperative rule

Up to now, two mutation operators (Gaussian and DE/rand-worst/1) have been introduced, which combine a same crossover operator to produce the new trial vector $\boldsymbol{u}_i$ for each individual, and then the cooperative rule between them becomes a burning problem. A natural and reasonable rule is that adaptively executing one of the two mutation operators in terms of their own performance. To evaluate the performance of adopted mutation operators during the evolutionary process, we introduce a new concept called “cumulative score” into the mutation operation. For the two adopted mutation operators, their cumulative scores during the evolutionary process can be obtained via the following three steps. Firstly, the values of their initial cumulative scores are set (denoted by CS_0^g and CS_0^m) to 0.5. Secondly, suppose that the values of their historical cumulative scores are CS_{t-1}^g and CS_{t-1}^m, and then their single-period scores in the current generation (denoted by S_t^g and S_t^m) can be obtained via the following two formulas, respectively.

$$S_t^g = \begin{cases} \frac{C_t^g}{N_t^g}, & \text{if } N_t^g > 0, \\ \frac{\mathrm{CS}_{t-1}^g}{t}, & \text{otherwise}, \end{cases} \tag{13}$$

$$S_t^m = \begin{cases} \frac{C_t^m}{N_t^m}, & \text{if } N_t^m > 0, \\ \frac{\mathrm{CS}_{t-1}^m}{t}, & \text{otherwise}, \end{cases} \tag{14}$$

where the indices N_t^g and N_t^m represent the numbers of new trial vectors produced by Gaussian mutation operator and DE/rand-worst/1 in the tth generation, respectively. Actually, the value of N_t^g always is equals to $\mathrm{NP} - N_t^m$, owing to the fact that the population only executes NP times of mutation operators in one generation. And the indices C_t^g and C_t^m, respectively, represent the success times of the two operators' execution, where the concept of success means that the new produced trial vector is better than the original target vector. Note that formulas (13) and (14) express that the current single-period score of each adopted mutation operator is equal to its current success rate when executing at least once in the current generation, otherwise takes its average value of historical cumulative score. Thirdly, after obtaining the current single-period scores of the two adopted mutation operators, their current cumulative scores can be updated by the following two formulas,

$$\mathrm{CS}_t^g = \mathrm{CS}_{t-1}^g + S_t^g, \tag{15}$$

$$\mathrm{CS}_t^m = \mathrm{CS}_{t-1}^m + S_t^m. \tag{16}$$

Now, the value of a parameter involved in cooperative rule can be derived in terms of the two mutation operators' cumulative scores, which can be calculated by,

$$\mathrm{CS}_t = \frac{\mathrm{CS}_t^g}{\mathrm{CS}_t^g + \mathrm{CS}_t^m}, \tag{17}$$

where parameter CS_t is applied to control the selection probability of Gaussian mutation operator in the next generation. Furthermore, the detailed cooperative rule can be described as follows,

$$\boldsymbol{u}_i = \begin{cases} \boldsymbol{u}_i^g, & \text{if } \mathrm{rand}[0, 1] < \mathrm{CS}_t, \\ \boldsymbol{u}_i^m, & \text{otherwise}. \end{cases} \tag{18}$$

The cooperative rule (18) shows that the chance of executing the adopted two mutation operators relies on their own cumulative scores, and the one with higher cumulative score has more chance to produce the trial vectors. After the new trial vector produced, GPDE will compare the fitness values of each individual $\boldsymbol{x}_i$ and its new trial vector $\boldsymbol{u}_i$ and then produce the offspring via the selection operator (8).

4.4 The overall procedure of GPDE

We have provided a detailed description of Gaussian mutation operator, DE/rand-worst/1, and the cooperative rule between them. Now, we summarize the overall procedure of GPDE into Algorithm 1.

5 Comparison and result analysis

In this section, we firstly provide the test functions, real-world problems and compared DE algorithms, secondly present the

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's function	800
	f_9	Shifted and rotated Rastrigin's function	900
	f_{10}	Shifted Schwefel's 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: $[-100, 100]^D$	

comparative results between GPDE and the other seven algorithms, and analyze the effects of control parameters on the performance of GPDE at last.

5.1 Test functions and real-world problems

In order to evaluate the performance of GPDE, we apply a set of 30 well-known test functions from IEEE CEC 2014 (Liang et al. 2013) and four real-world problems to conduct the comparative experiment. Specifically, based on the characteristics of the 30 test functions, they can be divided into four classes, which are summarized in Table 1. Moreover, the adopted test functions are carried out in the comparative experiment when their dimensions are equal to 30, 50 and 100, respectively. In addition, the four real-world problems (denoted by rf_1, rf_2, rf_3 and rf_4, respectively) are widely used to evaluate the performance of various algorithms, which are applications to parameter estimation for frequency-modulated sound waves (Das and Suganthan 2011b), spread spectrum radar poly-phase code design (Das and Suganthan 2011b), systems of linear equations (García-Martínez et al. 2008), and parameter optimization for polynomial fitting problem (Herrera and Lozano 2000), respectively.

5.2 Compared algorithms and parameter configurations

In our comparative experiment, GPDE is compared with five excellent DE variants, including SADE (Qin et al. 2009), JADE (Zhang and Sanderson 2009), GDE (Han et al. 2013), MGBDE (Wang et al. 2013) and SinDE (Draa et al. 2015).

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

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	C-ABC	CCPSO2	GPDE
f_1	Mean	3.22e+04	3.76e+04	6.85e+03	**5.57e+03**	1.92e+06	8.63e+04	9.12e+05	5.21e+04
	Std.	2.17e+04	3.66e+04	6.41e+03	**3.45e+03**	1.11e+06	6.15e+04	6.40e+05	3.51e+04
	+/=/−	−	−	−	−	+	+	+	−
f_2	Mean	4.84e−20	3.08e−20	3.06e−22	3.92e−15	**2.90e−24**	7.49e−01	8.87e+03	1.35e−23
	Std.	2.39e−19	9.38e−20	3.80e−22	1.95e−14	**1.01e−23**	1.12e+00	9.26e+03	2.17e−22
	+/=/−	+	+	+	+	=	+	+	−
f_3	Mean	9.82e−10	1.34e−01	3.56e−03	2.08e−18	2.18e−12	4.32e+00	1.95e+04	**5.42e−25**
	Std.	4.91e−09	6.05e−01	9.98e−03	9.54e−18	9.18e−12	8.66e+00	1.76e+04	**1.83e−24**
	+/=/−	+	+	+	+	+	+	+	−
f_4	Mean	1.39e+01	1.94e+01	5.48e+00	**9.95e−05**	1.43e+01	1.15e+01	4.09e+01	2.99e+00
	Std.	2.72e+01	3.09e+01	1.89e+01	**4.48e−04**	2.28e+01	2.54e+01	5.16e+01	1.49e+01
	+/=/−	+	+	+	−	+	+	+	−
f_5	Mean	2.04e+01	**2.00e+01**	2.09e+01	2.02e+01	2.06e+01	2.09e+01	**2.00e+01**	**2.00e+01**
	Std.	4.74e−02	**3.99e−03**	1.48e−01	3.92e−02	4.77e−02	1.21e−01	**9.03e−03**	**6.53e−06**
	+/=/−	+	+	+	+	+	+	+	−
f_6	Mean	8.93e+00	1.14e+01	8.25e+00	2.22e+01	**1.93e−02**	4.03e+00	1.33e+01	1.33e+00
	Std.	2.03e+00	1.75e+00	2.85e+00	3.64e+00	**9.27e−02**	2.02e+00	3.13e+00	1.16e+00
	+/=/−	+	+	+	+	−	+	+	−
f_7	Mean	1.89e−02	2.69e−02	1.30e−02	1.12e−02	**0.00e+00**	2.55e−02	2.32e−02	2.17e−03
	Std.	2.10e−02	2.17e−02	1.75e−02	1.44e−02	**0.00e+00**	4.00e−02	2.63e−02	4.17e−03
	+/=/−	+	+	+	+	=	+	+	−
f_8	Mean	4.78e+00	3.98e−02	6.38e+01	1.29e+02	4.70e−01	2.60e+01	**0.00e+00**	9.79e+00
	Std.	2.54e+00	1.99e−01	1.58e+01	3.17e+01	5.69e−01	7.64e+00	**0.00e+00**	3.72e+00
	+/=/−	−	−	+	+	−	+	−	−
f_9	Mean	4.59e+01	4.97e+01	7.46e+01	1.56e+02	3.58e+01	**3.13e+01**	7.34e+01	3.46e+01
	Std.	1.09e+01	8.71e+00	2.74e+01	2.95e+01	7.51e+00	**1.02e+01**	1.87e+01	9.26e+00
	+/=/−	+	+	+	+	=	=	+	−
f_{10}	Mean	3.40e+00	6.98e+00	1.80e+03	1.25e+03	9.31e+00	2.42e+03	**4.49e−03**	1.25e+02
	Std.	2.09e+00	2.40e+01	6.43e+02	7.96e+02	4.77e+00	1.18e+03	**8.05e−03**	9.65e+01
	+/=/−	−	−	+	+	−	+	−	−
f_{11}	Mean	2.42e+03	2.04e+03	5.06e+03	2.85e+03	2.35e+03	6.58e+03	2.07e+03	**1.97e+03**
	Std.	5.45e+02	2.43e+02	1.60e+03	6.29e+02	4.08e+02	4.87e+02	3.70e+02	**4.71e+02**
	+/=/−	+	=	+	+	+	+	=	−
f_{12}	Mean	5.91e−01	1.28e−01	1.77e+00	3.36e−01	8.04e−01	1.90e+00	**8.49e−02**	1.49e−01
	Std.	8.56e−02	2.61e−02	8.70e−01	3.67e−02	1.24e−01	3.80e−01	**1.85e−02**	7.84e−02
	+/=/−	+	=	+	+	+	+	−	−
f_{13}	Mean	2.80e−01	3.09e−01	3.64e−01	4.40e−01	**2.06e−01**	3.10e−01	3.27e−01	2.40e−01
	Std.	5.43e−02	5.90e−02	7.23e−02	7.72e−02	**5.27e−02**	6.09e−02	9.95e−02	6.84e−02
	+/=/−	+	+	+	+	=	+	+	−
f_{14}	Mean	2.41e−01	2.50e−01	2.95e−01	2.58e−01	2.42e−01	2.82e−01	**2.06e−01**	2.22e−01
	Std.	4.47e−02	1.01e−01	9.12e−02	5.78e−02	2.60e−02	4.64e−02	**1.32e−01**	3.40e−02
	+/=/−	=	=	+	+	+	+	−	−
f_{15}	Mean	4.57e+00	1.23e+01	8.12e+00	1.33e+01	4.81e+00	6.57e+00	7.15e+00	**3.75e+00**
	Std.	1.31e+00	6.69e+00	2.95e+00	2.37e+00	9.80e−01	5.03e+00	3.14e+00	**9.44e−01**
	+/=/−	+	+	+	+	+	+	+	−

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

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	C-ABC	CCPSO2	GPDE
f_{16}	Mean	1.03e+01	1.02e+01	1.10e+01	1.07e+01	1.00e+01	1.24e+01	9.76e+00	**9.64e+00**
	Std.	4.16e−01	3.56e−01	1.12e+00	6.49e−01	5.22e−01	2.49e−01	6.60e−01	**7.85e−01**
	+/=/−	+	+	+	+	+	+	=	−
f_{17}	Mean	5.95e+03	6.29e+04	1.91e+04	**1.71e+03**	1.25e+05	1.37e+04	7.32e+05	3.84e+03
	Std.	4.17e+03	6.21e+04	3.03e+04	**7.57e+02**	1.20e+05	9.70e+03	4.57e+05	3.53e+03
	+/=/−	+	+	+	−	+	+	+	−
f_{18}	Mean	8.34e+02	7.06e+02	7.43e+01	1.03e+02	5.15e+02	2.70e+03	5.53e+03	**2.16e+01**
	Std.	1.22e+03	9.62e+02	1.90e+02	3.60e+01	6.94e+02	3.24e+03	5.52e+03	**9.20e+00**
	+/=/−	+	+	+	+	+	+	+	−
f_{19}	Mean	4.23e+00	1.11e+01	4.73e+00	2.33e+01	3.71e+00	6.54e+00	1.14e+01	**3.45e+00**
	Std.	1.23e+00	1.66e+01	1.20e+00	2.55e+01	7.18e−01	8.46e+00	1.84e+01	**1.18e+00**
	+/=/−	+	+	+	+	+	+	+	−
f_{20}	Mean	1.08e+02	1.37e+03	2.88e+01	7.80e+01	2.57e+01	3.32e+02	3.34e+04	**1.71e+01**
	Std.	1.42e+02	2.06e+03	2.19e+01	4.38e+01	2.85e+01	2.95e+02	1.77e+04	**1.12e+01**
	+/=/−	+	+	+	+	+	+	+	−
f_{21}	Mean	4.84e+03	5.74e+03	3.29e+03	**8.50e+02**	9.23e+03	9.11e+03	4.50e+05	3.63e+03
	Std.	4.42e+03	7.17e+03	5.36e+03	**3.90e+02**	7.52e+03	7.06e+03	3.82e+05	4.61e+03
	+/=/−	+	+	=	−	+	+	+	−
f_{22}	Mean	1.55e+02	2.10e+02	4.82e+02	7.17e+02	**7.25e+01**	1.52e+02	6.73e+02	2.90e+02
	Std.	8.61e+01	7.94e+01	2.09e+02	2.74e+02	**6.28e+01**	1.30e+02	1.78e+02	1.41e+02
	+/=/−	−	=	+	+	−	−	+	−
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**	**1.78e−13**	**3.27e−05**	**1.04e−13**
	+/=/−	=	=	=	=	=	=	=	−
f_{24}	Mean	2.28e+02	2.30e+02	2.35e+02	2.42e+02	**2.23e+02**	2.36e+02	2.27e+02	2.27e+02
	Std.	5.41e+00	3.87e+00	7.27e+00	1.18e+01	**8.57e−01**	7.48e+00	3.12e+00	4.51e+00
	+/=/−	=	+	+	+	−	+	=	−
f_{25}	Mean	2.10e+02	2.12e+02	**2.04e+02**	2.20e+02	**2.04e+02**	**2.04e+02**	2.09e+02	**2.04e+02**
	Std.	2.07e+00	1.47e+00	**1.19e+00**	6.22e+00	**6.64e−01**	**8.41e−01**	4.60e+00	**7.80e−01**
	+/=/−	+	+	=	+	=	=	+	−
f_{26}	Mean	1.12e+02	1.56e+02	**1.00e+02**	1.56e+02	**1.00e+02**	**1.00e+02**	1.42e+02	1.08e+02
	Std.	3.31e+01	5.05e+01	**9.32e−02**	5.04e+01	**3.99e−02**	**7.14e−02**	6.27e+01	2.76e+01
	+/=/−	=	+	=	+	−	=	+	−
f_{27}	Mean	4.36e+02	4.29e+02	4.28e+02	8.61e+02	**3.02e+02**	4.14e+02	6.08e+02	3.34e+02
	Std.	5.74e+01	6.28e+01	5.93e+01	3.11e+02	**7.08e+00**	4.31e+01	1.42e+02	3.49e+01
	+/=/−	+	+	+	+	−	+	+	−
f_{28}	Mean	9.12e+02	9.19e+02	9.56e+02	2.60e+03	7.94e+02	8.65e+02	1.22e+03	**7.93e+02**
	Std.	4.37e+01	6.51e+01	6.23e+01	7.61e+02	3.23e+01	6.06e+01	4.64e+02	**2.60e+01**
	+/=/−	+	+	+	+	=	+	+	−
f_{29}	Mean	6.83e+02	7.84e+02	**4.80e+02**	7.18e+02	1.32e+03	3.48e+05	1.02e+06	6.32e+02
	Std.	2.64e+02	2.69e+02	**2.73e+02**	1.26e+02	2.39e+02	1.73e+06	2.82e+06	1.96e+02
	+/=/−	=	+	−	=	+	+	+	−
f_{30}	Mean	1.96e+03	2.05e+03	1.11e+03	2.14e+03	**8.10e+02**	1.26e+03	3.17e+03	1.62e+03
	Std.	6.22e+02	5.50e+02	3.00e+02	7.18e+02	**1.69e+02**	4.43e+02	8.42e+02	7.04e+02
	+/=/−	+	+	−	+	−	=	+	−

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

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	C-ABC	CCPSO2	GPDE
f_1	Mean	1.81e+05	7.59e+04	4.26e+05	**6.96e+04**	2.96e+06	5.85e+05	2.52e+06	9.52e+05
	Std.	7.78e+04	3.67e+04	1.89e+05	**3.61e+04**	1.02e+06	2.12e+05	1.00e+06	3.05e+05
	+/=/−	−	−	−	−	+	−	+	−
f_2	Mean	4.72e+03	4.03e+03	2.27e+01	**1.37e−10**	4.15e+03	3.27e+03	9.70e+03	1.75e+00
	Std.	4.07e+03	5.19e+03	3.58e+01	**4.09e−10**	2.76e+03	3.92e+03	9.99e+03	3.35e+00
	+/=/−	+	+	+	−	+	+	+	−
f_3	Mean	1.91e+01	3.88e−02	3.09e+01	**2.70e−05**	5.81e+02	5.67e+03	3.90e+04	1.45e+00
	Std.	2.32e+01	1.04e−01	3.14e+01	**4.57e−05**	4.24e+02	2.11e+03	1.59e+04	4.09e+00
	+/=/−	+	−	+	−	+	+	+	−
f_4	Mean	6.24e+01	6.26e+01	2.41e+01	**1.50e+01**	9.28e+01	7.40e+01	7.88e+01	4.37e+01
	Std.	3.62e+01	2.72e+01	3.31e+01	**3.83e+01**	3.51e+00	3.22e+01	3.09e+01	3.58e+01
	+/=/−	+	+	−	−	+	+	+	−
f_5	Mean	2.07e+01	2.01e+01	2.11e+01	2.04e+01	2.08e+01	2.11e+01	**2.00e+01**	**2.00e+01**
	Std.	3.48e−02	8.33e−03	5.23e−02	2.87e−02	3.49e−02	4.32e−02	**1.06e−03**	**4.73e−06**
	+/=/−	+	+	+	+	+	+	+	−
f_6	Mean	1.95e+01	2.54e+01	1.53e+01	4.19e+01	**4.07e−02**	4.77e+00	2.39e+01	4.97e+00
	Std.	3.73e+00	2.37e+00	2.82e+00	4.31e+00	**1.34e−01**	2.50e+00	6.37e+00	2.99e+00
	+/=/−	+	+	+	+	−	=	+	−
f_7	Mean	8.70e−03	2.16e−02	9.03e−03	6.24e−03	**1.18e−16**	4.60e−03	1.72e−02	6.57e−04
	Std.	9.17e−03	4.29e−02	9.96e−03	6.16e−03	**1.29e−16**	5.88e−03	2.44e−02	2.55e−03
	+/=/−	+	+	+	+	=	+	+	−
f_8	Mean	9.42e+00	6.63e−02	1.11e+02	2.58e+02	1.10e+01	4.24e+01	**0.00e+00**	1.57e+01
	Std.	3.62e+00	2.57e−01	2.01e+01	4.42e+01	4.33e+00	1.24e+01	**0.00e+00**	3.93e+00
	+/=/−	−	−	+	+	−	+	−	−
f_9	Mean	9.15e+01	1.04e+02	1.09e+02	2.95e+02	7.04e+01	7.81e+01	1.49e+02	**6.76e+01**
	Std.	1.15e+01	1.46e+01	2.81e+01	3.68e+01	1.92e+01	7.51e+01	3.34e+01	**9.61e+00**
	+/=/−	+	+	+	+	=	=	+	−
f_{10}	Mean	2.50e+00	3.05e+00	3.46e+03	3.37e+03	1.57e+02	9.45e+03	**2.56e−03**	2.03e+02
	Std.	1.07e+00	9.81e−01	1.13e+03	1.93e+03	8.15e+01	3.01e+03	**6.98e−03**	1.46e+02
	+/=/−	−	−	+	+	=	+	−	−
f_{11}	Mean	7.04e+03	**4.07e+03**	1.26e+04	5.65e+03	4.95e+03	1.33e+04	4.23e+03	4.52e+03
	Std.	4.82e+02	**4.54e+02**	2.13e+03	6.00e+02	7.01e+02	2.71e+02	5.64e+02	6.60e+02
	+/=/−	+	=	+	+	=	+	=	−
f_{12}	Mean	7.82e−01	1.57e−01	3.15e+00	4.10e−01	1.34e+00	3.12e+00	**8.73e−02**	1.54e−01
	Std.	9.98e−02	1.51e−02	4.66e−01	3.78e−02	1.25e−01	4.71e−01	**2.73e−02**	6.97e−02
	+/=/−	+	=	+	+	+	+	−	−
f_{13}	Mean	4.17e−01	4.56e−01	5.06e−01	5.46e−01	3.46e−01	4.01e−01	4.28e−01	**3.44e−01**
	Std.	5.86e−02	6.73e−02	9.63e−02	1.12e−01	5.18e−02	5.08e−02	6.72e−02	**6.10e−02**
	+/=/−	+	+	+	+	=	+	+	−
f_{14}	Mean	3.14e−01	2.85e−01	3.50e−01	3.60e−01	2.48e−01	3.23e−01	3.48e−01	**2.46e−01**
	Std.	3.21e−02	1.76e−02	1.57e−01	1.30e−01	3.08e−02	2.99e−02	2.44e−01	**2.42e−02**
	+/=/−	+	+	+	+	=	+	=	−
f_{15}	Mean	1.44e+01	3.03e+01	1.85e+01	2.77e+01	8.38e+00	2.32e+01	1.28e+01	**6.86e+00**
	Std.	3.35e+00	7.10e+00	1.20e+01	4.11e+00	1.56e+00	1.20e+01	5.75e+00	**1.81e+00**
	+/=/−	+	+	+	+	+	+	+	−

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

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	C-ABC	CCPSO2	GPDE
f_{16}	Mean	1.98e+01	1.86e+01	2.22e+01	1.92e+01	2.00e+01	2.23e+01	**1.74e+01**	1.86e+01
	Std.	2.50e−01	4.04e−01	3.04e−01	4.33e−01	6.39e−01	1.34e−01	**8.29e−01**	9.38e−01
	+/=/−	+	=	+	+	+	+	−	−
f_{17}	Mean	2.23e+04	9.29e+04	5.48e+04	**1.06e+04**	3.94e+05	5.19e+04	6.64e+05	5.41e+04
	Std.	1.41e+04	7.40e+04	3.40e+04	**4.95e+03**	2.26e+05	2.49e+04	4.44e+05	2.85e+04
	+/=/−	−	+	=	−	+	=	+	−
f_{18}	Mean	4.02e+02	8.88e+02	1.50e+02	6.03e+02	3.53e+02	5.89e+02	2.01e+03	**4.28e+01**
	Std.	3.23e+02	7.82e+02	1.56e+02	1.20e+03	3.13e+02	7.44e+02	1.28e+03	**3.36e+01**
	+/=/−	+	+	+	+	+	+	+	−
f_{19}	Mean	1.36e+01	3.40e+01	9.41e+00	1.92e+01	9.58e+00	2.38e+01	1.48e+01	**6.82e+00**
	Std.	5.79e+00	2.14e+01	4.50e+00	2.18e+00	7.10e−01	2.03e+01	2.35e+00	**1.34e+00**
	+/=/−	+	+	+	+	+	+	+	−
f_{20}	Mean	2.39e+02	6.87e+02	4.80e+02	1.81e+02	2.16e+02	1.12e+03	6.45e+04	**1.17e+02**
	Std.	6.34e+01	1.37e+03	5.13e+02	4.31e+01	1.53e+02	2.91e+02	2.38e+04	**1.45e+02**
	+/=/−	+	+	+	+	+	+	+	−
f_{21}	Mean	2.65e+04	4.22e+04	2.93e+04	**3.02e+03**	2.62e+05	4.97e+04	8.82e+05	3.13e+04
	Std.	1.94e+04	4.93e+04	2.48e+04	**1.81e+03**	1.49e+05	3.74e+04	8.11e+05	3.80e+04
	+/=/−	=	+	=	−	+	+	+	−
f_{22}	Mean	4.22e+02	5.80e+02	1.36e+03	1.23e+03	2.57e+02	5.87e+02	1.26e+03	**2.14e+02**
	Std.	1.04e+02	1.07e+02	4.14e+02	3.62e+02	1.49e+02	3.61e+02	2.80e+02	**1.64e+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**	**1.04e−13**	**3.46e−12**	**8.73e−14**
	+/=/−	=	=	=	=	=	=	=	−
f_{24}	Mean	2.72e+02	2.78e+02	2.81e+02	3.02e+02	2.64e+02	2.76e+02	**2.58e+02**	2.63e+02
	Std.	6.57e+00	3.95e+00	3.36e+00	1.13e+01	3.28e+00	4.35e+00	**3.10e+00**	6.01e+00
	+/=/−	+	+	+	+	=	+	=	−
f_{25}	Mean	2.10e+02	2.28e+02	**2.08e+02**	2.38e+02	2.10e+02	**2.08e+02**	2.15e+02	**2.08e+02**
	Std.	1.01e+01	2.16e+00	**2.06e+00**	4.72e+00	1.29e+00	**1.83e+00**	3.63e+00	**2.23e+00**
	+/=/−	=	+	=	+	=	=	+	−
f_{26}	Mean	1.54e+02	1.20e+02	**1.00e+02**	1.07e+02	**1.00e+02**	1.07e+02	1.75e+02	1.09e+02
	Std.	5.15e+01	4.12e+01	**7.72e−02**	2.57e+01	**2.36e−02**	2.57e+01	9.01e+01	3.51e+01
	+/=/−	+	+	+	+	=	=	+	−
f_{27}	Mean	7.70e+02	7.75e+02	7.05e+02	1.58e+03	**3.22e+02**	4.57e+02	1.05e+03	4.32e+02
	Std.	8.80e+01	1.53e+02	1.05e+02	1.48e+02	**2.13e+01**	6.65e+01	1.75e+02	3.60e+01
	+/=/−	+	+	+	+	−	=	+	−
f_{28}	Mean	1.42e+03	1.61e+03	1.45e+03	5.36e+03	**1.09e+03**	1.15e+03	2.12e+03	1.19e+03
	Std.	1.14e+02	2.46e+02	1.10e+02	1.03e+03	**3.62e+01**	6.78e+01	6.65e+02	5.11e+01
	+/=/−	+	+	+	+	−	=	+	−
f_{29}	Mean	1.05e+03	9.94e+02	**8.36e+02**	8.96e+02	1.89e+03	1.36e+03	2.48e+03	1.27e+03
	Std.	1.93e+02	1.17e+02	**2.51e+02**	2.04e+02	3.13e+02	3.54e+02	6.52e+02	2.25e+02
	+/=/−	−	−	−	−	+	=	+	−
f_{30}	Mean	1.06e+04	1.14e+04	9.89e+03	1.18e+04	**8.99e+03**	9.12e+03	1.24e+04	9.07e+03
	Std.	1.68e+03	1.71e+03	5.53e+02	9.07e+02	**2.86e+02**	4.41e+02	1.48+03e	4.86e+02
	+/=/−	+	+	+	+	=	=	+	−

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

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	C-ABC	CCPSO2	GPDE
f_1	Mean	8.96e+05	4.70e+05	8.29e+06	**4.38e+05**	2.19e+07	4.73e+06	1.92e+07	9.59e+06
	Std.	1.45e+05	1.93e+05	3.20e+06	**1.51e+05**	5.50e+06	9.62e+05	6.89e+06	2.74e+06
	+/=/−	−	−	=	−	+	−	+	−
f_2	Mean	1.31e+04	6.38e+03	1.74e+04	**2.80e−10**	1.06e+04	1.27e+04	3.67e+04	1.23e+01
	Std.	8.76e+03	1.08e+04	2.13e+04	**4.04e−10**	6.32e+03	1.02e+04	3.80e+04	1.50e+01
	+/=/−	+	+	+	−	+	+	+	−
f_3	Mean	7.66e+01	8.65e+00	5.22e+03	**2.51e−02**	3.22e+03	5.46e+04	5.01e+04	3.45e+02
	Std.	6.52e+01	2.83e+00	4.07e+03	**6.79e−02**	1.34e+03	1.10e+04	2.24e+04	2.62e+02
	+/=/−	−	−	+	−	+	+	+	−
f_4	Mean	1.71e+02	1.62e+02	1.89e+02	**1.35e+02**	1.59e+02	1.69e+02	2.09e+02	1.60e+02
	Std.	4.05e+01	4.80e+01	3.38e+01	**5.92e+01**	2.57e+01	3.63e+01	3.14e+01	2.77e+01
	+/=/−	=	=	+	−	=	=	+	−
f_5	Mean	2.10e+01	2.03e+01	2.13e+01	2.07e+01	2.12e+01	2.13e+01	**2.00e+01**	**2.00e+01**
	Std.	2.30e−02	1.75e−02	2.31e−02	2.06e−02	2.46e−02	3.04e−02	**1.18e−03**	**9.75e−07**
	+/=/−	+	+	+	+	+	+	+	−
f_6	Mean	6.44e+01	7.23e+01	4.56e+01	1.03e+02	**4.79e+00**	8.30e+00	6.35e+01	6.36e+00
	Std.	5.35e+00	3.68e+00	7.45e+00	5.42e+00	**2.59e+00**	2.67e+00	1.07e+01	3.85e+00
	+/=/−	+	+	+	+	=	+	+	−
f_7	Mean	2.63e−03	6.39e−03	6.07e−03	3.61e−03	7.52e−11	**2.00e−16**	3.78e−03	3.77e−16
	Std.	6.20e−03	1.06e−02	9.15e−03	6.02e−03	6.49e−11	**1.47e−16**	5.88e−03	2.44e−16
	+/=/−	+	+	+	+	+	=	+	−
f_8	Mean	1.92e+01	4.17e+00	2.32e+02	5.84e+02	6.22e+01	2.42e+02	**5.09e−09**	3.92e+01
	Std.	5.89e+00	9.84e−01	4.44e+01	4.27e+01	8.16e+00	1.76e+02	**3.75e−09**	6.47e+00
	+/=/−	−	−	+	+	+	+	−	−
f_9	Mean	2.77e+02	2.72e+02	3.30e+02	6.57e+02	**1.36e+02**	8.17e+02	4.21e+02	1.45e+02
	Std.	2.84e+01	2.04e+01	1.65e+02	7.80e+01	**2.61e+01**	7.54e+01	1.23e+02	2.72e+01
	+/=/−	+	+	+	+	=	+	+	−
f_{10}	Mean	1.26e+02	1.49e+01	6.69e+03	1.01e+04	5.53e+03	2.69e+04	**1.40e−05**	5.73e+02
	Std.	1.97e+01	2.49e+00	1.25e+03	3.99e+03	7.98e+02	5.42e+02	**3.37e−05**	2.32e+02
	+/=/−	−	−	+	+	+	+	−	−
f_{11}	Mean	2.05e+04	**1.06e+04**	3.03e+04	1.41e+04	1.53e+04	3.03e+04	1.12e+04	1.22e+04
	Std.	5.94e+02	**6.56e+02**	5.24e+02	1.21e+03	2.10e+03	4.83e+02	1.04e+03	2.12e+03
	+/=/−	+	=	+	+	+	+	=	−
f_{12}	Mean	1.62e+00	2.91e−01	4.01e+00	6.80e−01	2.49e+00	4.05e+00	**1.19e−01**	3.18e−01
	Std.	1.51e−01	3.69e−02	1.74e−01	5.26e−02	1.30e−01	1.68e−01	**3.19e−02**	1.22e−01
	+/=/−	+	=	+	+	+	+	−	−
f_{13}	Mean	**4.67e−01**	4.85e−01	6.53e−01	5.90e−01	5.33e−01	5.40e−01	5.30e−01	4.71e−01
	Std.	**3.88e−02**	5.58e−02	8.10e−02	8.60e−02	4.26e−02	4.92e−02	6.22e−02	6.06e−02
	+/=/−	=	=	+	+	+	+	+	−
f_{14}	Mean	3.21e−01	2.88e−01	3.47e−01	3.67e−01	2.83e−01	3.34e−01	3.33e−01	**2.80e−01**
	Std.	1.97e−02	2.10e−02	3.77e−02	1.07e−01	2.40e−02	2.64e−02	1.97e−01	**1.74e−02**
	+/=/−	+	=	+	+	=	+	+	−
f_{15}	Mean	4.09e+01	6.61e+01	8.70e+01	6.95e+01	2.26e+01	7.43e+01	4.36e+01	**1.59e+01**
	Std.	1.12e+01	1.82e+01	1.80e+01	7.08e+00	5.45e+00	1.79e+00	2.14e+01	**2.52e+00**
	+/=/−	+	+	+	+	+	+	+	−

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

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	C-ABC	CCPSO2	GPDE
f_{16}	Mean	4.38e+01	4.14e+01	4.66e+01	4.21e+01	4.48e+01	4.65e+01	**3.93e+01**	4.13e+01
	Std.	3.77e−01	3.46e−01	2.57e−01	4.73e−01	3.32e−01	2.69e−01	**1.07e+00**	8.40e−01
	+/=/−	+	=	+	+	+	+	−	−
f_{17}	Mean	1.50e+05	3.27e+05	4.82e+05	**8.97e+04**	2.49e+06	1.92e+06	3.98e+06	1.01e+06
	Std.	4.37e+04	1.63e+05	2.07e+05	**4.39e+04**	1.09e+06	5.58e+05	1.99e+06	5.12e+05
	+/=/−	−	−	−	−	+	+	+	−
f_{18}	Mean	8.02e+02	6.84e+02	7.02e+02	1.50e+03	**2.61e+02**	5.46e+02	4.71e+03	5.74e+02
	Std.	8.63e+02	4.40e+02	7.69e+02	1.64e+03	**3.12e+02**	5.69e+02	4.68e+03	1.08e+03
	+/=/−	+	+	+	+	−	=	+	−
f_{19}	Mean	**8.39e+01**	1.01e+02	9.83e+01	8.95e+01	9.02e+01	8.51e+01	1.02e+02	8.91e+01
	Std.	**3.13e+01**	4.27e+01	1.86e+01	3.56e+01	8.56e−01	1.99e+01	1.95e+01	9.74e−01
	+/=/−	=	+	+	=	+	=	+	−
f_{20}	Mean	7.14e+02	1.09e+03	3.21e+03	**4.54e+02**	6.32e+03	1.83e+04	1.08e+05	5.71e+02
	Std.	2.09e+02	1.71e+03	1.17e+03	**1.29e+02**	1.63e+03	5.75e+03	4.45e+04	8.60e+01
	+/=/−	+	+	+	−	+	+	+	−
f_{21}	Mean	7.28e+04	1.34e+05	1.87e+05	**2.96e+04**	1.98e+06	1.29e+06	3.02e+06	3.85e+05
	Std.	5.04e+04	8.18e+04	7.22e+04	**1.54e+04**	6.00e+05	5.08e+05	1.16e+06	1.64e+05
	+/=/−	−	−	−	−	+	+	+	−
f_{22}	Mean	1.38e+03	1.41e+03	3.65e+03	2.69e+03	1.35e+03	4.03e+03	2.62e+03	**1.16e+03**
	Std.	2.94e+02	2.58e+02	9.48e+02	4.76e+02	3.96e+02	2.20e+02	5.07e+02	**2.59e+02**
	+/=/−	+	+	+	+	+	+	+	−
f_{23}	Mean	**3.48e+02**	**3.48e+02**	**3.48e+02**	**3.48e+02**	**3.48e+02**	**3.48e+02**	3.49e+02	**3.48e+02**
	Std.	**4.07e−13**	**6.44e−12**	**5.33e−05**	**3.19e−12**	**1.11e−04**	**3.73e−12**	7.64e−01	**5.08e−13**
	+/=/−	=	=	=	=	=	=	+	−
f_{24}	Mean	3.78e+02	3.99e+02	4.06e+02	4.52e+02	3.62e+02	3.78e+02	**3.31e+02**	3.64e+02
	Std.	6.30e+00	7.55e+00	8.35e+00	2.28e+01	1.69e+00	4.21e+00	**1.05e+01**	2.65e+00
	+/=/−	+	+	+	+	−	+	−	−
f_{25}	Mean	**2.00e+02**	2.57e+02	2.45e+02	2.88e+02	2.49e+02	2.45e+02	2.54e+02	2.39e+02
	Std.	**6.26e−14**	8.37e+00	9.57e+00	1.66e+01	4.19e+00	1.19e+01	1.17e+01	4.76e+00
	+/=/−	−	+	=	+	+	=	+	−
f_{26}	Mean	**2.00e+02**	**2.00e+02**	2.01e+02	**2.00e+02**	2.01e+02	**2.00e+02**	2.02e+02	2.01e+02
	Std.	**2.49e−02**	**7.47e−03**	1.10e−01	**1.24e−02**	2.09e−01	**6.94e−02**	4.20e−01	1.77e−01
	+/=/−	−	−	=	−	=	−	+	−
f_{27}	Mean	1.43e+03	1.37e+03	1.40e+03	3.12e+03	**3.06e+02**	4.43e+02	2.09e+03	3.74e+02
	Std.	1.01e+02	2.15e+02	1.63e+02	2.39e+02	**1.24e+01**	3.76e+01	3.55e+02	5.91e+01
	+/=/−	+	+	+	+	−	+	+	−
f_{28}	Mean	3.20e+03	4.72e+03	2.91e+03	1.39e+04	2.20e+03	**1.95e+03**	3.87e+03	2.08e+03
	Std.	3.34e+02	6.34e+02	3.15e+02	1.92e+03	4.63e+01	**3.32e+02**	8.04e+02	1.78e+02
	+/=/−	+	+	+	+	+	=	+	−
f_{29}	Mean	1.37e+03	1.45e+03	1.98e+03	**1.23e+03**	2.81e+03	5.68e+03	4.37e+03	1.91e+03
	Std.	2.51e+02	1.85e+02	4.05e+02	**2.18e+02**	4.80e+02	2.50e+03	8.66e+02	1.41e+02
	+/=/−	−	−	=	−	+	+	+	−
f_{30}	Mean	8.17e+03	8.59e+03	**6.01e+03**	7.45e+03	9.53e+03	6.64e+03	1.91e+04	8.90e+03
	Std.	9.43e+02	1.58e+03	**9.47e+02**	2.02e+03	1.25e+03	1.44e+03	6.97e+03	8.01e+02
	+/=/−	−	=	−	−	=	−	+	−

Algorithm 1 The overall procedure of GPDE

```
Set the values of parameters $NP$, $FR$, $V$ and $T$;
Initialize $NP$ individuals with random positions via the formula (1);

for ($t = 1; t <= T; t++$), do
    Compute the value of scaling factor $F_t$ in the $t$th generation via the formula (12);
    Compute the value of parameter $CS_t$ in the $t$th generation via the formula (17);
    for ($i = 1; i <= NP; i++$), do
        Compute the crossover rate $CR_t^i$ of the $i$th individual in the $t$th generation via the formula (10);
        if rand[0, 1] $<$ CS$_t$ then
            Generate the new trial vector via the formula (9) included Gaussian mutation operator;
        else
            Generate the new trial vector via the formula (11) containing mutation operator DE/rand-worst/1;
        end if;
        Update the $i$th individual via the selection operator (8);
        Replace the best individual $\boldsymbol{x}_{best}$ by the new individual $\boldsymbol{x}_i$ if $\boldsymbol{x}_i$ is better than $\boldsymbol{x}_{best}$;
    end for
end for
Output the position of the best individual as the global optimal solution.
```

To be specific, SADE and JADE are two state-of-the-art DE variants, and GDE and MGBDE are recently proposed variants both of which adopt two different mutation operators; in particular, MGBDE employs a similar Gaussian mutation operator with GPDE, and SinDE is an up-to-date DE variant, which applies two sinusoidal functions to adjust the values of mutation scaling factor and crossover rate. These selected DE variants not only have outstanding performance, but also have some similar aspects to our proposed GPDE, that is why we take them as the comparison object. In addition, two good performing state-of-the-art meta-heuristic algorithms, i.e., cooperative coevolving particle swarm optimization with random grouping [denoted by CCPSO2 (Li and Yao 2012] and collective resource-based artificial bee colony with decentralized tasking (denoted by C-ABC (Bose et al. 2014) for short), are used to enrich the comparative experiment.

For all the aforementioned compared algorithms, except the population sizes NP, which, respectively, are equal to D and $5D$ for test functions and real-world problems, the other involved control parameters keep the same with their corresponding literature. In GPDE, there are only three user-specified control parameters, including population size NP, periodic adjustment parameter FR and the variance V of crossover rate. Note that the values of FR and V are, respectively, set to 0.05 and 0.1, and the value of NP always takes the same as its competitors, and all these three values will keep no change for all the adopted test functions and real-world problems. In addition, all the compared algorithms are

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

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	C-ABC	CCPSO2	GPDE
rf_1	Mean	1.05e+00	9.95e+00	6.35e+00	1.26e+01	**4.68e−01**	2.23e+00	1.24e+01	1.34e+00
	Std.	3.66e+00	4.74e+00	6.87e+00	4.41e+00	**2.34e+00**	4.62e+00	7.67e+00	3.72e+00
	+/=/−	=	+	+	+	=	=	+	−
rf_2	Mean	1.93e+00	1.79e+00	2.38e+00	1.95e+00	1.78e+00	2.39e+00	1.80e+00	**1.63e+00**
	Std.	8.64e−02	9.98e−02	9.90e−02	1.56e−01	1.69e−01	9.67e−02	1.96e−01	**2.19e−01**
	+/=/−	+	+	+	+	+	+	+	−
rf_3	Mean	1.76e−04	4.38e−09	**4.63e−14**	1.09e+01	7.07e+00	7.98e+00	2.66e+02	2.82e+00
	Std.	4.55e−04	1.49e−08	**7.08e−14**	6.47e+00	4.25e+00	6.99e+00	1.76e+02	5.11e+00
	+/=/−	−	−	−	+	+	+	+	−
rf_4	Mean	**0.00e+00**	**0.00e+00**	**0.00e+00**	**0.00e+00**	4.63e+00	9.83e+00	2.87e+02	**0.00e+00**
	Std.	**0.00e+00**	**0.00e+00**	**0.00e+00**	**0.00e+00**	7.68e+00	1.16e+01	3.13e+02	**0.00e+00**
	+/=/−	=	=	=	=	+	+	+	−

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

Func.	Metric	SADE	JADE	GDE	MGBDE	SinDE	C-ABC	CCPSO2	GPDE
30-D	+/=/−	21/5/4	22/5/3	23/4/3	24/2/4	15/7/8	23/6/1	22/4/4	−
50-D	+/=/−	22/3/5	21/4/5	23/4/3	22/1/7	14/12/4	19/10/1	22/4/4	−
100-D	+/=/−	16/4/10	14/8/8	22/5/3	18/2/10	20/7/3	20/7/3	24/1/5	−
Real-word	+/=/−	1/2/1	2/1/1	2/1/1	3/1/0	3/1/0	3/1/0	4/0/0	−
Total	+/=/−	60/14/20	59/18/17	70/14/10	67/6/21	52/27/15	65/24/5	72/9/13	−

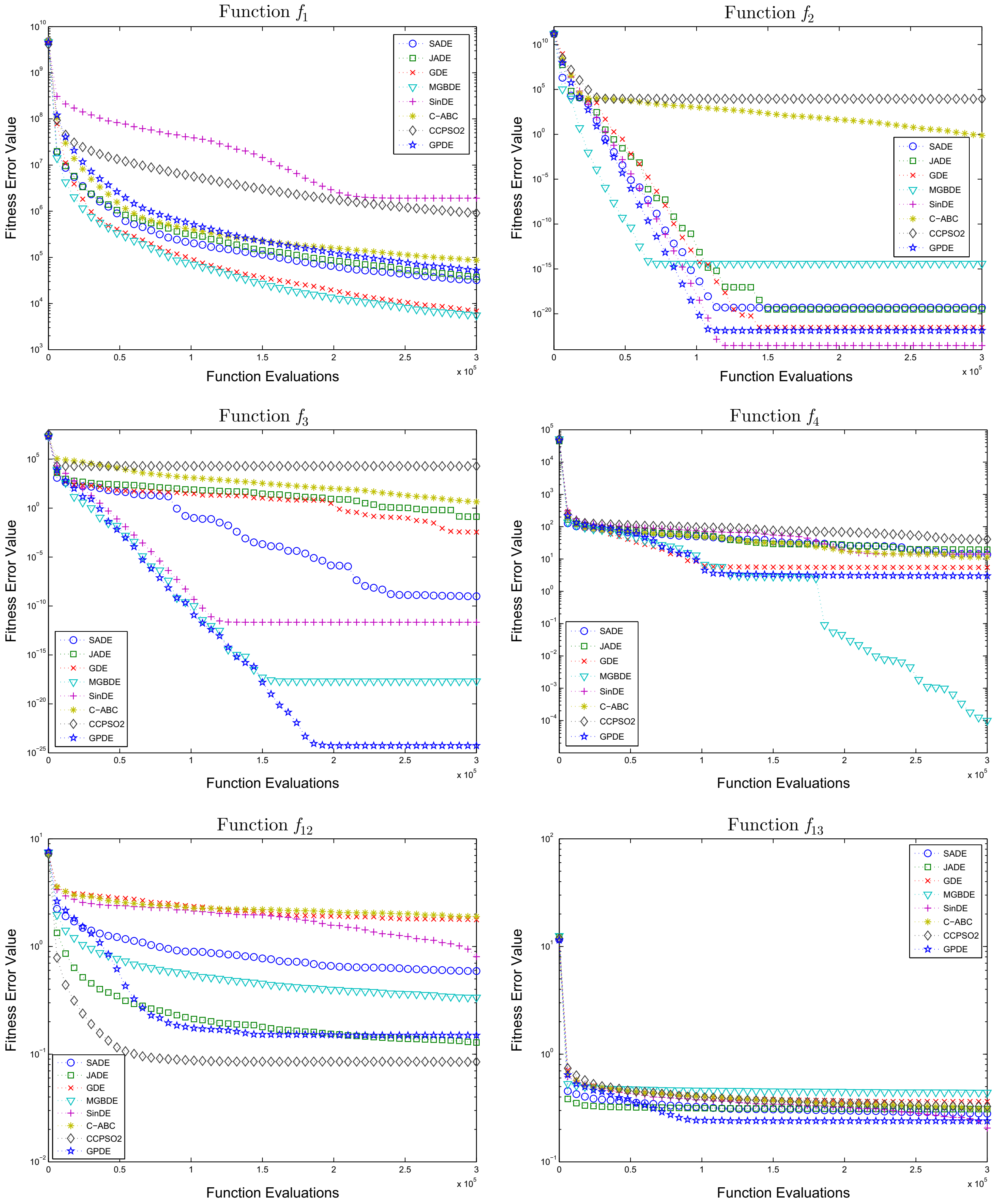


Fig. 1 Convergence graphs (mean curves) for eight algorithms on functions f_1, f_2, f_3, f_4, f_{12} and f_{13} with $D = 30$ over 50 independent runs

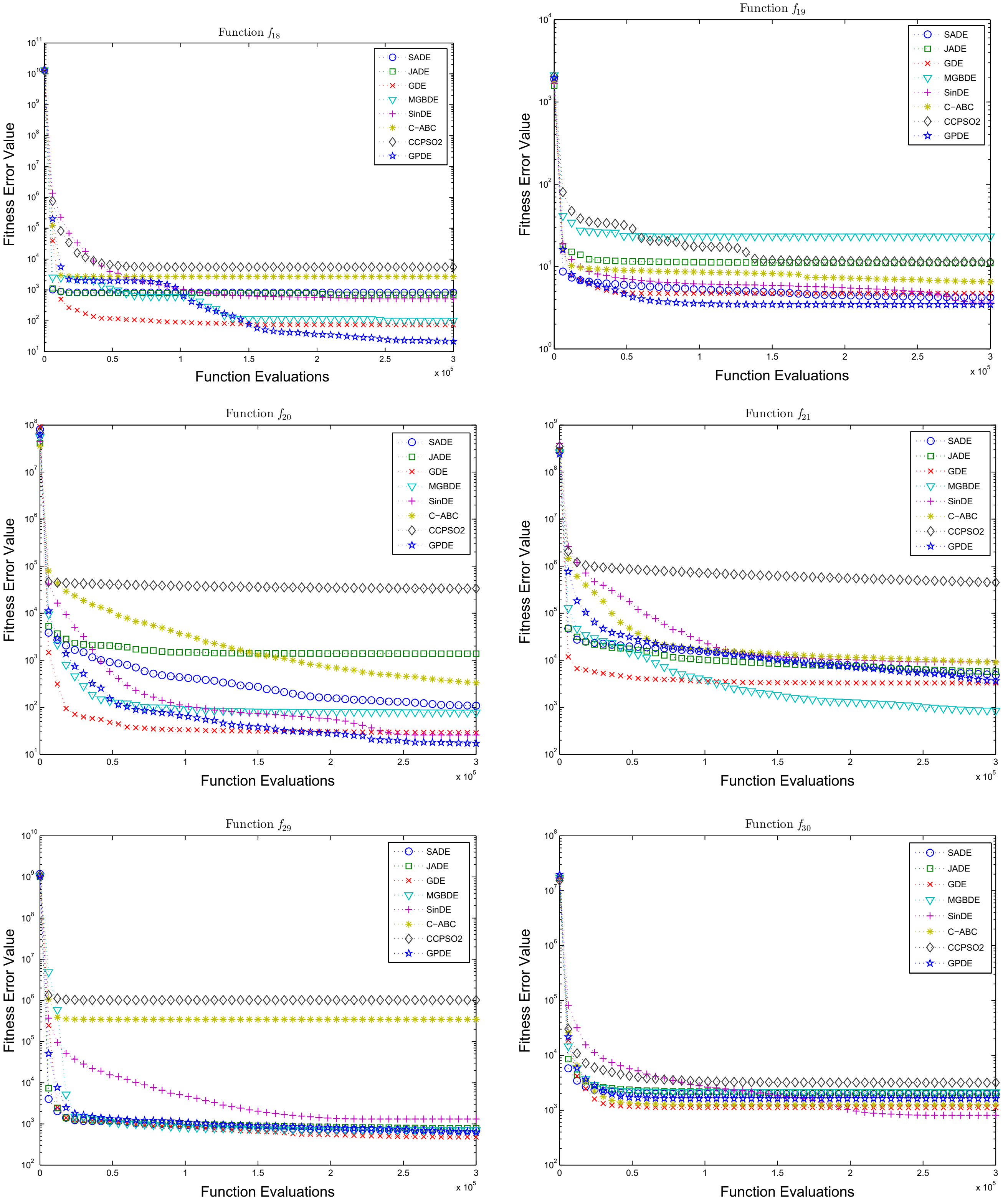


Fig. 2 Convergence graphs (mean curves) for eight algorithms on functions f_{18}, f_{19}, f_{20}, f_{21}, f_{29} and f_{30} with $D = 30$ over 50 independent runs

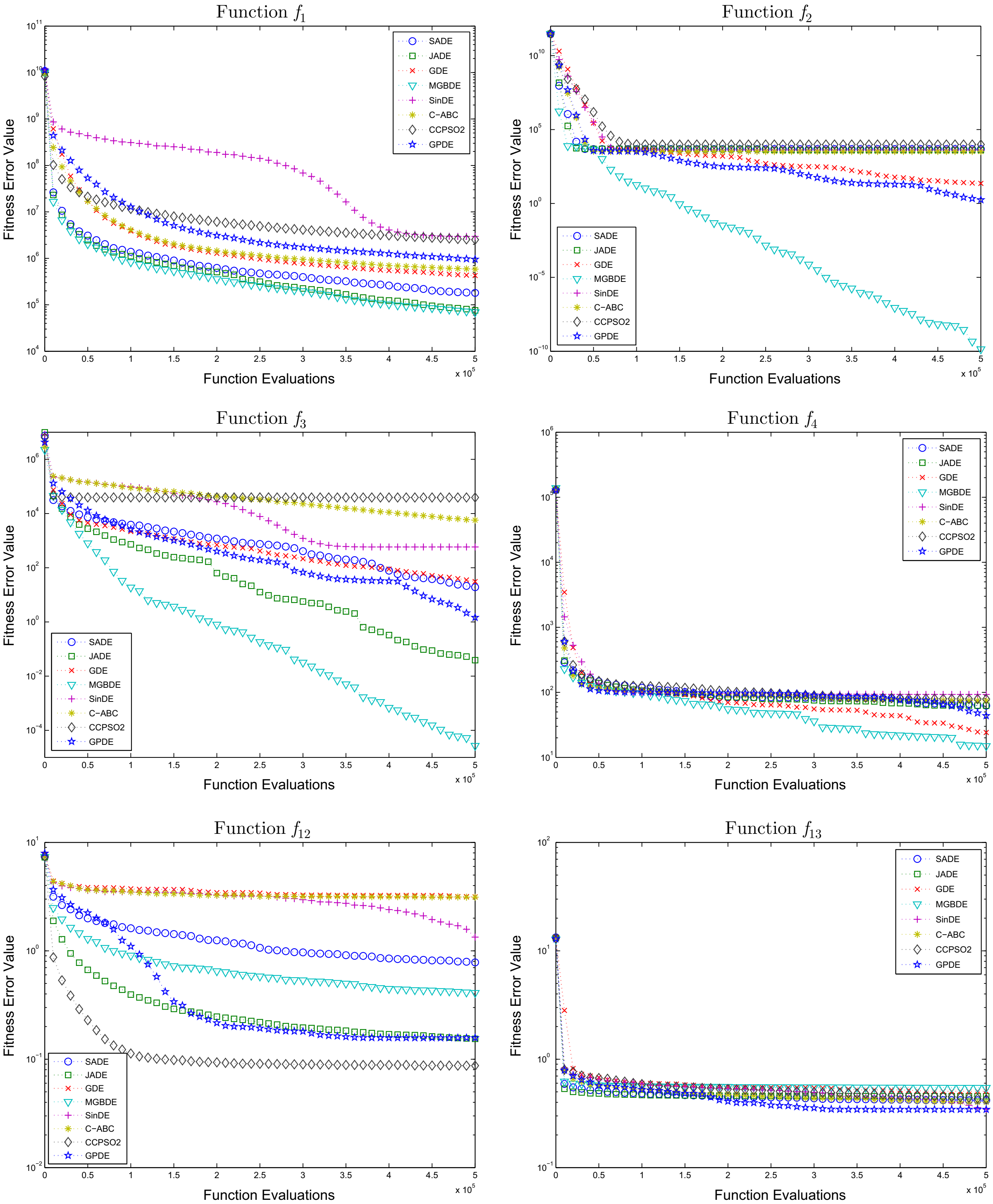


Fig. 3 Convergence graphs (mean curves) for eight algorithms on functions f_1, f_2, f_3, f_4, f_{12} and f_{13} with $D = 50$ over 50 independent runs

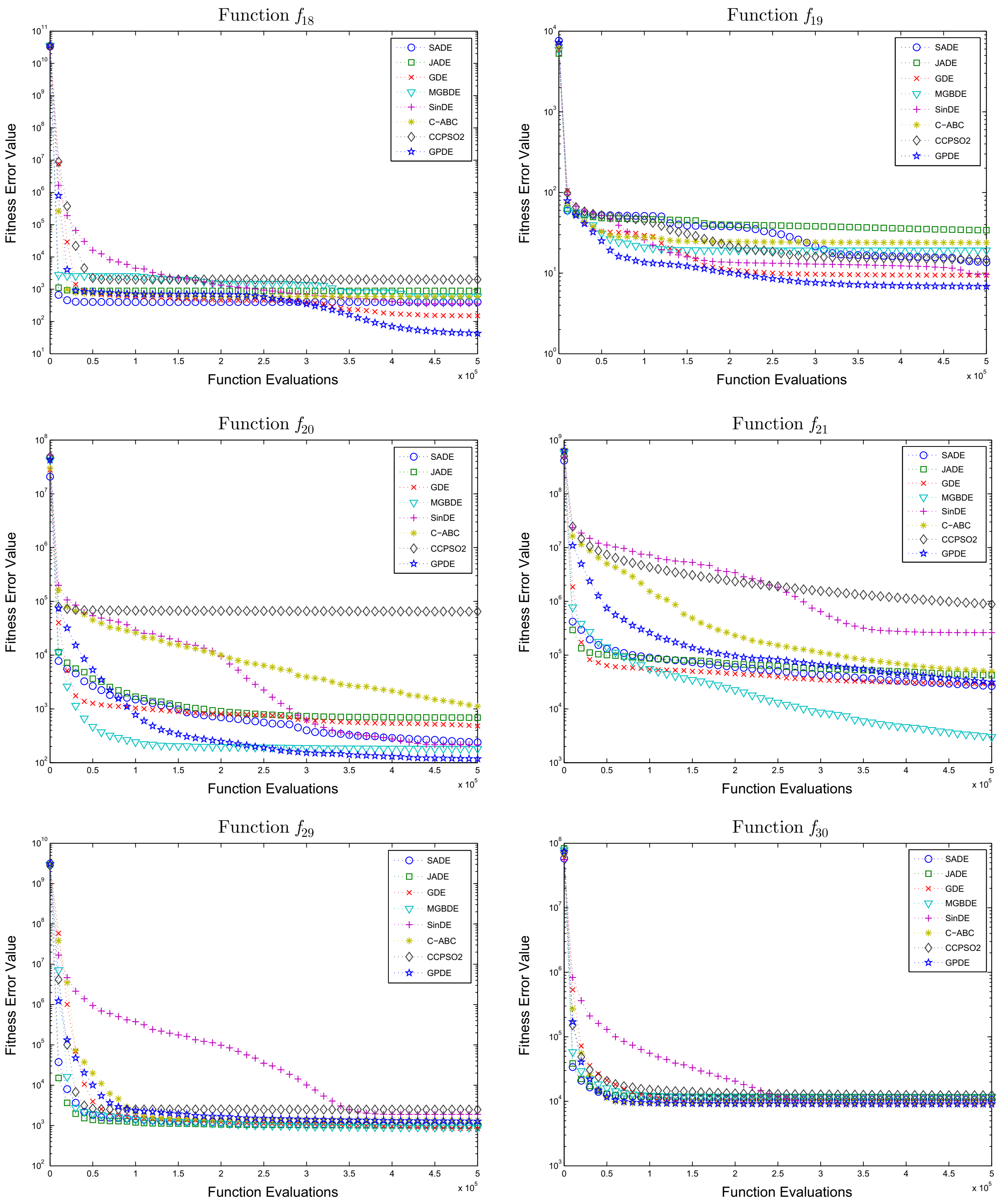


Fig. 4 Convergence graphs (mean curves) for eight algorithms on functions f_{18}, f_{19}, f_{20}, f_{21}, f_{29} and f_{30} with $D = 50$ over 50 independent runs

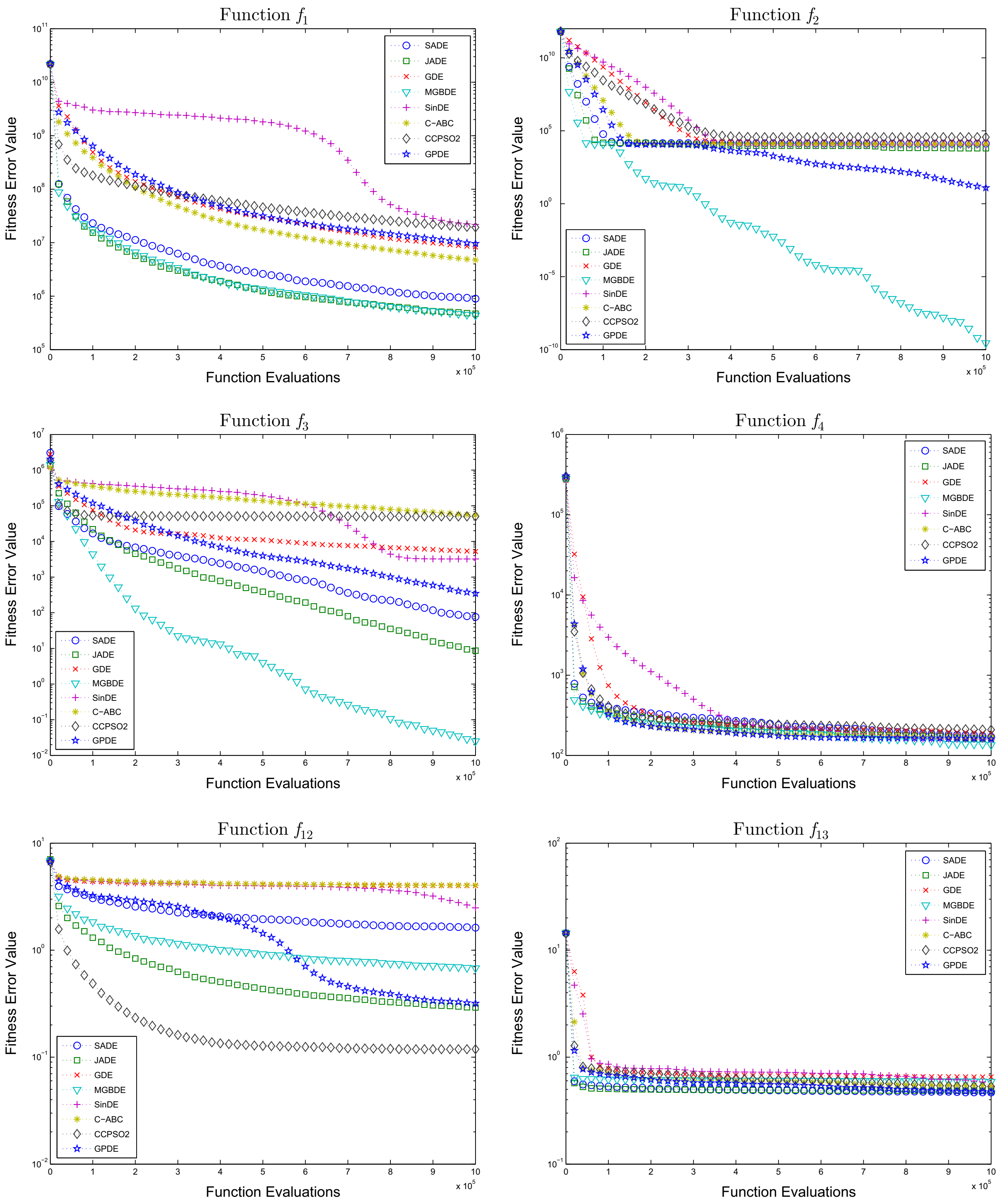
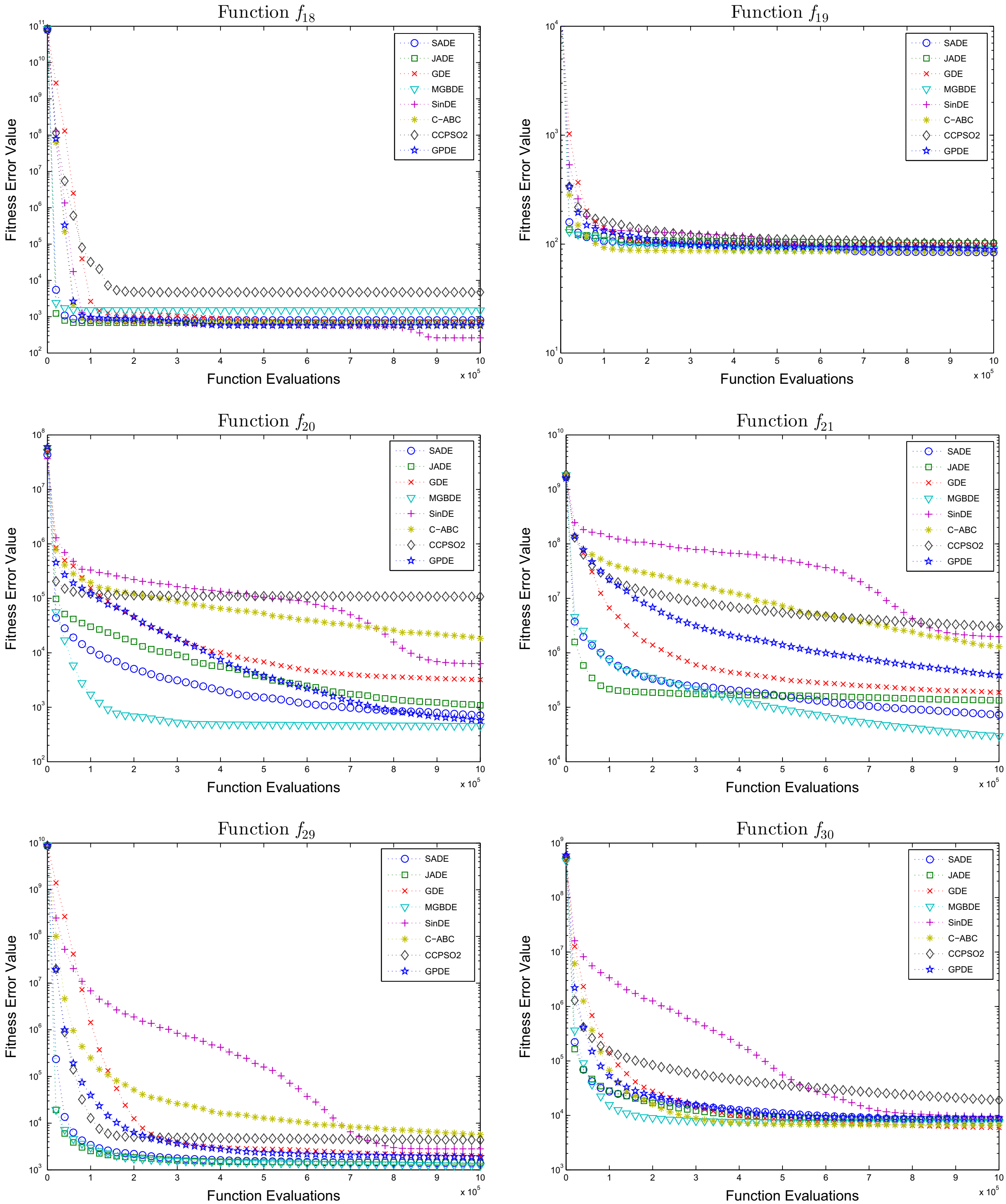


Fig. 5 Convergence graphs (mean curves) for eight algorithms on functions f_1, f_2, f_3, f_4, f_{12} and f_{13} with $D = 100$ over 50 independent runs


Function f_{18}

Function f_{19}

Function f_{20}

Function f_{21}

Function f_{29}

Function f_{30}

Fig. 6 Convergence graphs (mean curves) for eight algorithms on functions f_{18}, f_{19}, f_{20}, f_{21}, f_{29} and f_{30} with $D = 100$ over 50 independent runs

Table 10 Parameter configurations of different GPDEs

Par.	GPDE-C1	GPDE-C2	GPDE-C3	GPDE-C4	GPDE-C5	GPDE-C6	GPDE-C7	GPDE-C8	GPDE-C9
FR	0.03	0.05	0.1	0.03	0.05	0.1	0.03	0.05	0.1
V	0.03	0.03	0.03	0.05	0.05	0.05	0.1	0.1	0.1

tested 50 independent runs for every function and the mean results are used in the comparison, and the maximum allowable generations are set to 10000 for all the test functions and real-world problems.

5.3 Comparative results

To evaluate the performances of the participant algorithms and provide a comprehensive comparison, we, respectively, report the mean (denoted by "Mean") fitness error value $\left(f\left(\boldsymbol{x}_{\text{best}}\right) - f\left(\boldsymbol{x}^*\right)\right)$, the corresponding standard deviation (Std.) and the statistical conclusion of comparative results based on 50 independent runs, where $\boldsymbol{x}_{\text{best}}$ is the best obtained solution and $\boldsymbol{x}^*$ is the known optimal solution. In addition, the statistical conclusions of the comparative results are based on the paired Wilcoxon rank sum test which is conducted at 0.05 significance level to assess the significance of the performance difference between GPDE and each competitor. We mark the three kinds of statistical significance cases with "+," "=" and "−" to indicate that GPDE is significantly better than, similar to, or worse than the corresponding competitor, respectively. The comparative results of test functions with different dimensions are summarized in Tables 2, 3, 4, 5, 6, 7 and 8, and the best results are indicated with boldface font to highlight the best algorithm for each test function. Moreover, the numbers of the three cases $(+/=/-)$ obtained by the compared results are summarized in Table 9.

Table 9 shows that GPDE obtains the best overall performance among the eight compared algorithms. In details, for the 94 functions, GPDE performs better than SADE, JADE, GDE, MGBDE, SinDE, C-ABC and CCPSO2 on 60, 59, 70, 67, 52, 65 and 72 functions, and only loses in 20, 17, 10, 21, 15, 5 and 13 functions, respectively. Moreover, GPDE outperforms its competitors on every adopted dimensions of test functions and has no worst "Mean" in 94 functions, which means that GPDE is robust, and thus it is a reliable algorithm for handling various problems with different dimensions.

In addition, to observe the convergence characteristics of the compared algorithms, we select 36 functions with different dimensions and plot their convergence graphs based on the mean values over 50 runs in Figs. 1, 2, 3, 4, 5 and 6. Obviously, GPDE has a wonderful convergence rate. As a conclusion, the experimental results and convergence graphs have demonstrated that GPDE performs significantly better than the other seven compared algorithms.

5.4 Robustness analysis of control parameters

Generally speaking, the involved control parameters may have important effects on the algorithmic performance, but it is good news for the users if the control parameters is robust. To verify the robustness of control parameters in GPDE, we compare various GPDEs with different parameter configurations, which are listed in Table 10. Note that we only evaluate the robustness of periodic adjustment parameter FR and the variance V of crossover rate, because population size NP usually has no obvious effect on the performance of DE.

Since most of the results obtained by different GPDEs are very close to each other, we only select 36 test functions with different dimensions whose results have relatively clear differences to reveal the robustness of control parameters FR and V via convergence graphs. The convergence graphs of 36 selected functions obtained by GPDEs with different parameter configurations are plotted in Figs. 7, 8, 9, 10, 11 and 12.

Three discoveries can be found out from Figs. 7, 8, 9, 10, 11 and 12. Above all, the results of GPDEs with different parameter configurations have no obvious fluctuation, which implies that the control parameters of GPDE are robust. Secondly, the variance V of crossover rate has a slightly bigger influence on the performance of GPDE than periodic adjustment parameter FR, because V has more direct effect on the population diversity than FR. At last, under a prescribed limit, a bigger value of V leads to a better result, because a bigger value of V is often corresponding to a better population diversity.

6 Conclusions

Differential evolution is an excellent evolutionary algorithm for global numerical optimization, but it is not completely free from the problems of premature convergence and stagnation. In order to alleviate these problems and enhance DE, we propose a new variant called GPDE. In GPDE, a novel Gaussian mutation operator who, respectively, takes the position of the best individual among three randomly selected individuals and the distance between the other two as the mean and standard deviation, and a modified common mutation operator is applied to cooperatively generate the mutant vectors. Moreover, scaling factor adopts a cosine function to

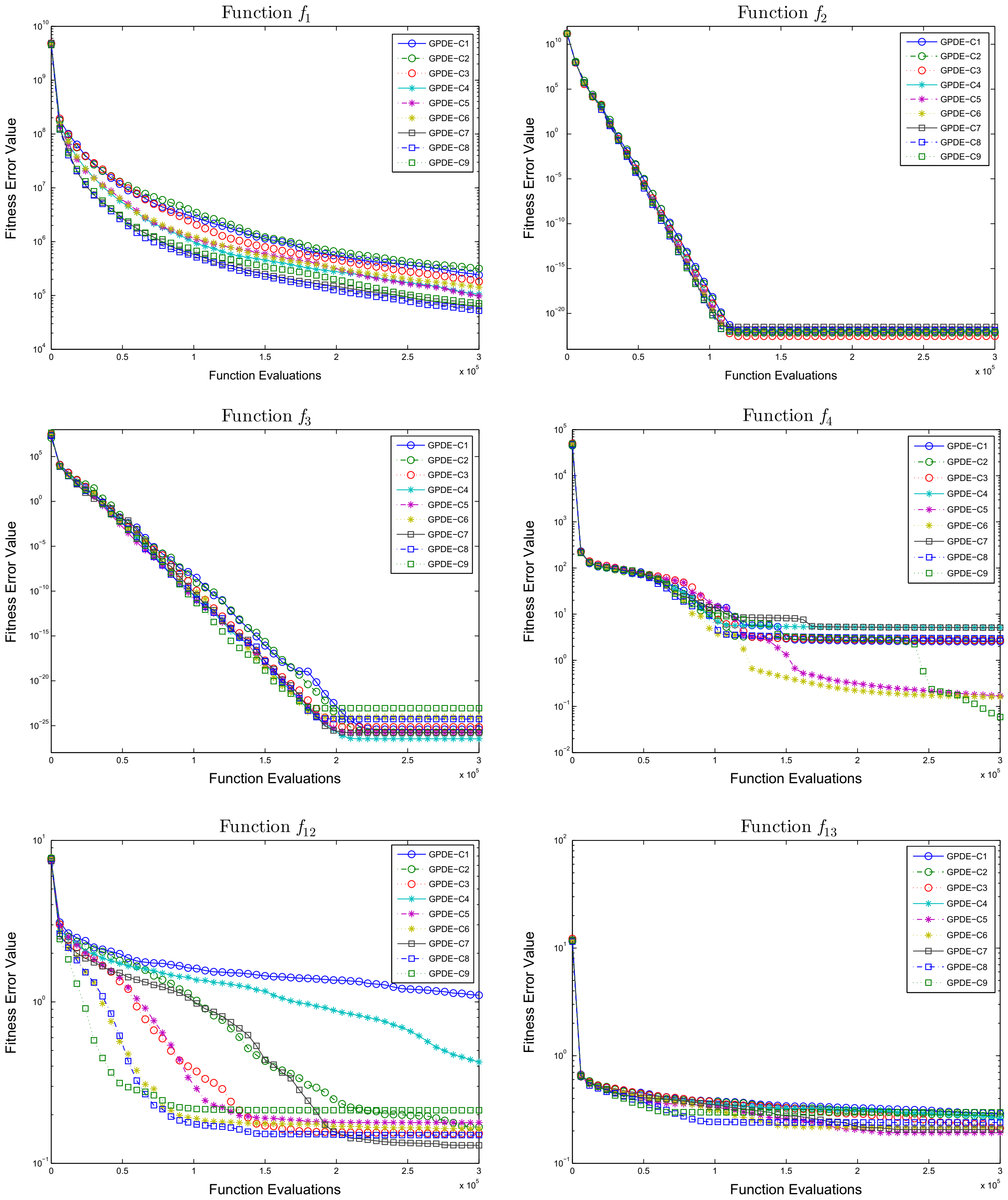


Fig. 7 Convergence graphs (mean curves) for the GPDE with different parameter configurations on functions f_1, f_2, f_3, f_4, f_{12} and f_{13} with $D = 30$ over 50 independent runs

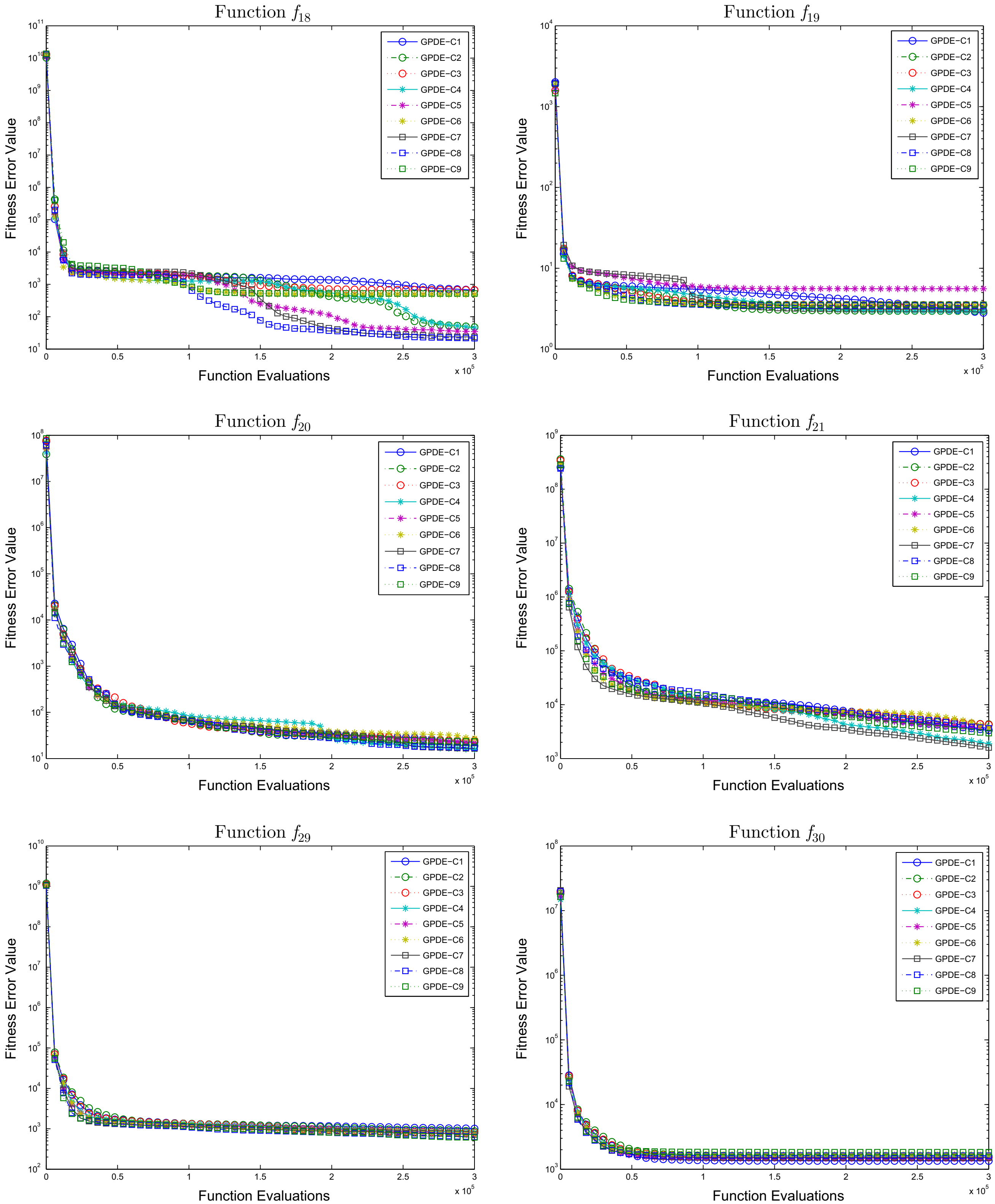


Fig. 8 Convergence graphs (mean curves) for the GPDE with different parameter configurations on functions f_{18}, f_{19}, f_{20}, f_{21}, f_{29} and f_{30} with $D = 30$ over 50 independent runs

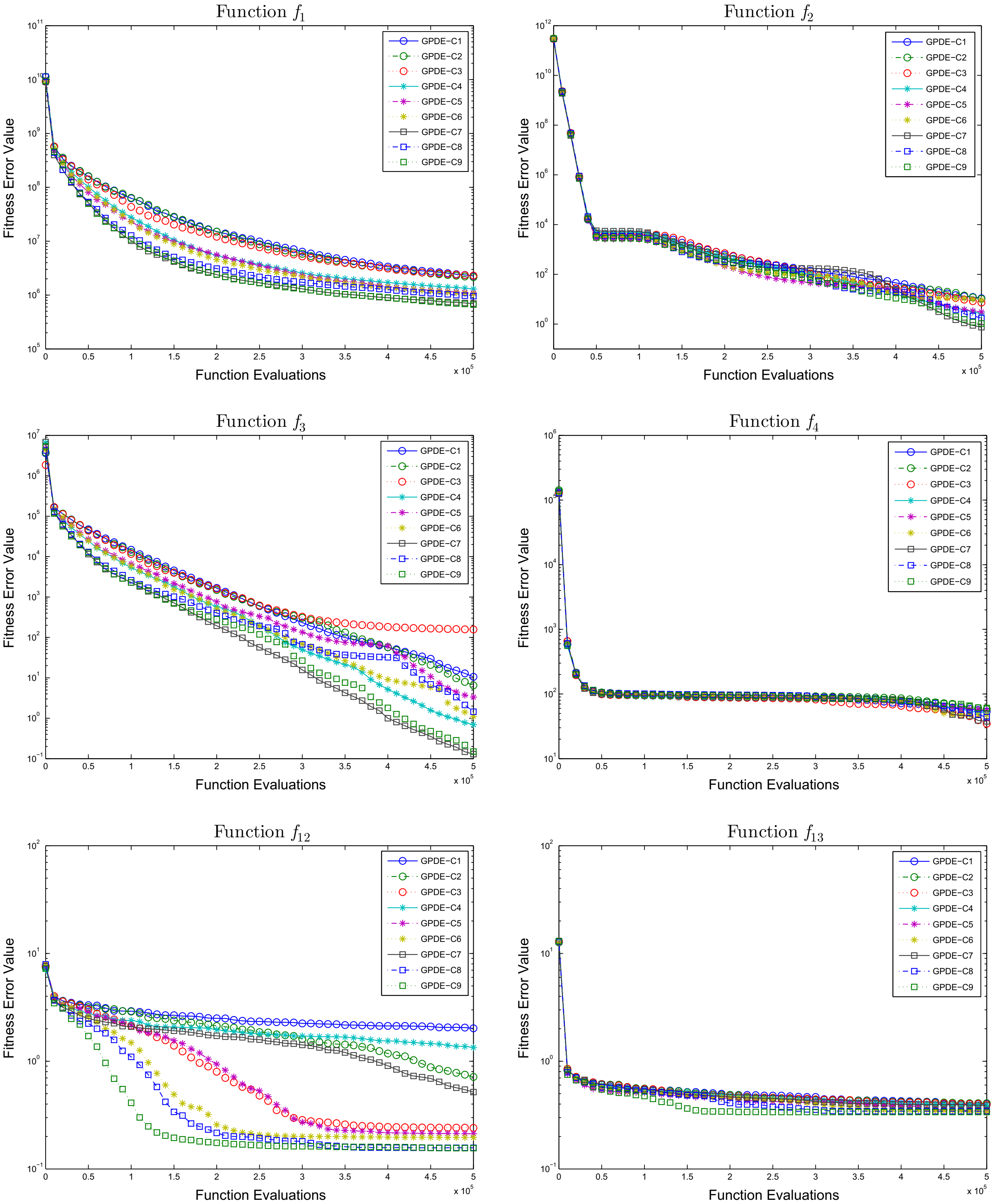


Fig. 9 Convergence graphs (mean curves) for the GPDE with different parameter configurations on functions f_1, f_2, f_3, f_4, f_{12} and f_{13} with $D = 50$ over 50 independent runs

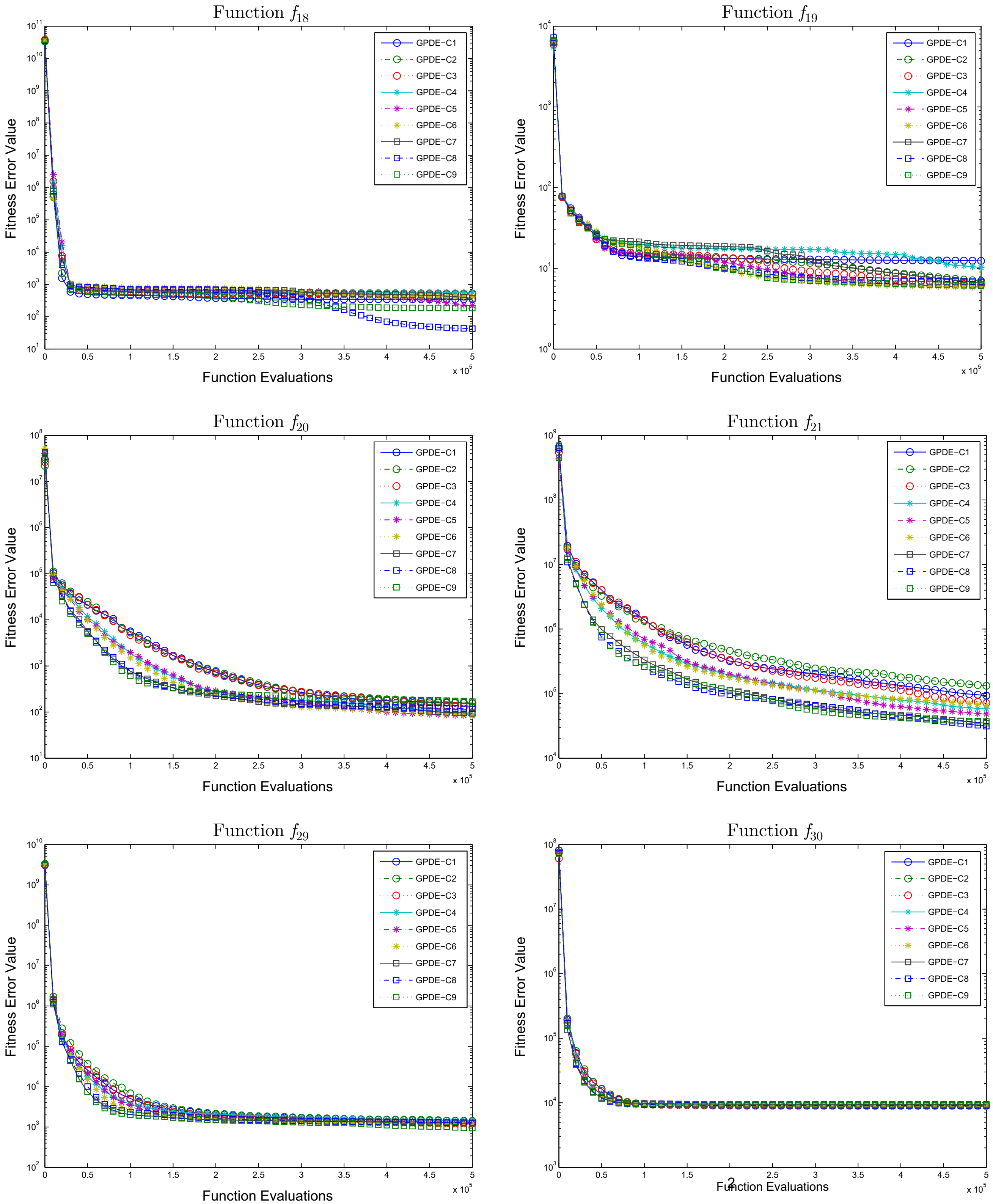
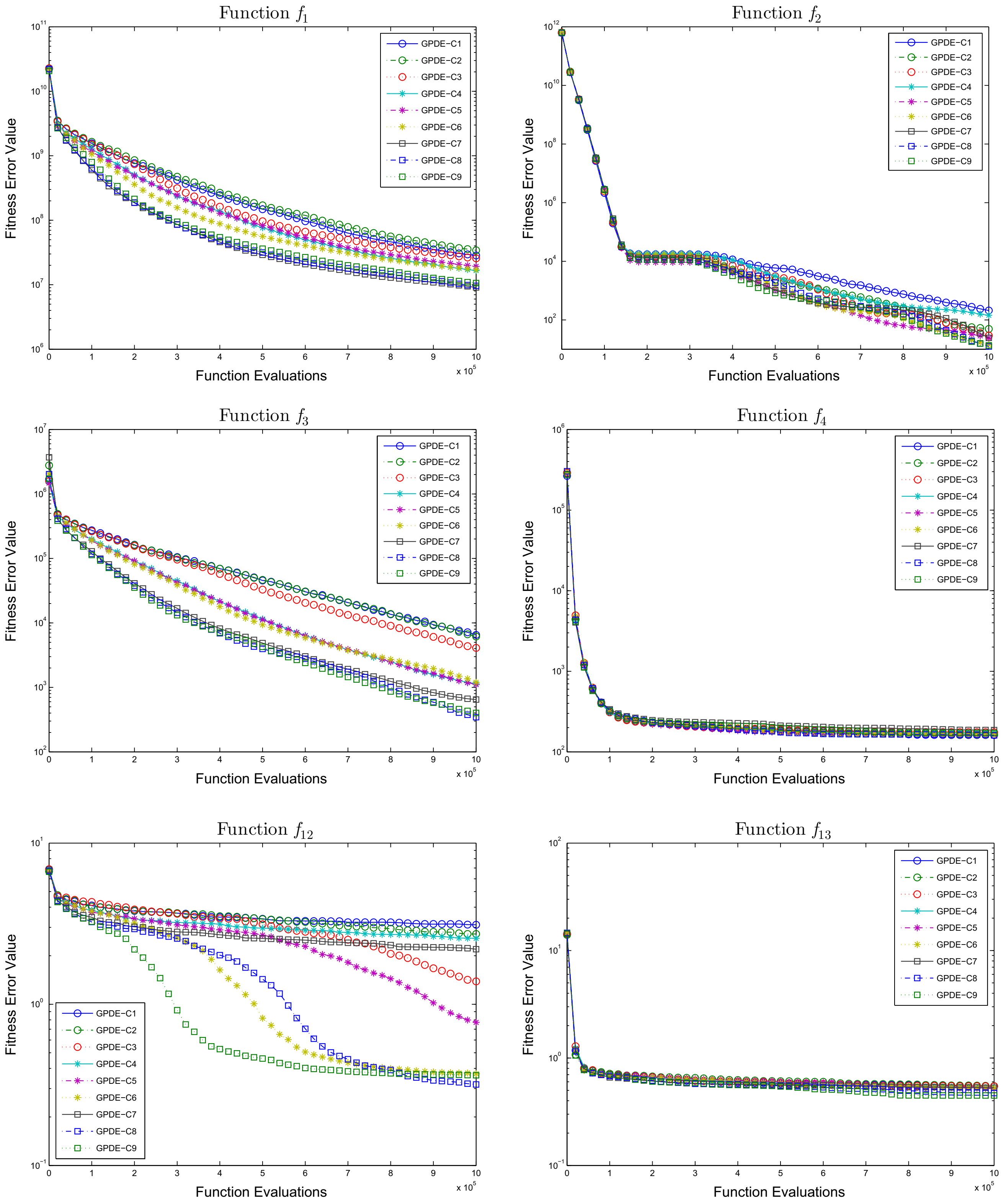


Fig. 10 Convergence graphs (mean curves) for the GPDE with different parameter configurations on functions f_{18}, f_{19}, f_{20}, f_{21}, f_{29} and f_{30} with $D = 50$ over 50 independent runs



Function f_1

Function f_2

Function f_3

Function f_4

Function f_{12}

Function f_{13}

Fig. 11 Convergence graphs (mean curves) for the GPDE with different parameter configurations on functions f_1, f_2, f_3, f_4, f_{12} and f_{13} with $D = 100$ over 50 independent runs

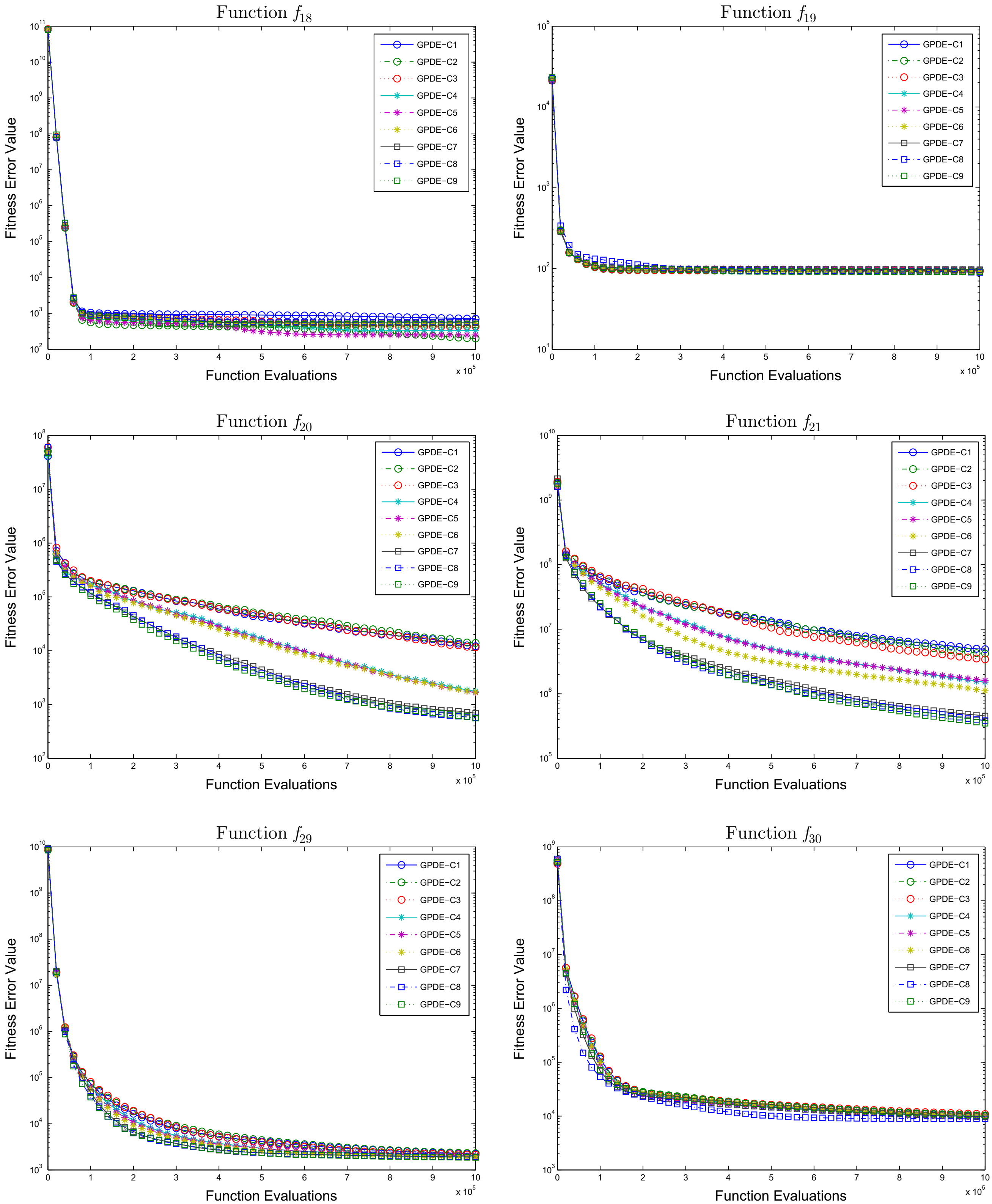


Fig. 12 Convergence graphs (mean curves) for the GPDE with different parameter configurations on functions f_{18}, f_{19}, f_{20}, f_{21}, f_{29} and f_{30} with $D = 100$ over 50 independent runs

adjust its value periodically, which has potential advantage in balancing the exploration and exploitation abilities, and crossover rate employs a Gaussian function to produce its value dynamically, which can adjust the population diversity. The test suite of IEEE CEC-2014 which contains 30 test function and four real-world problems, and seven remarkable meta-heuristic algorithms are used to evaluate the performance of GPDE, and the obtained results show that GPDE performs much better than the other seven compared DE algorithms. In addition, the parameter analysis indicates that the control parameters involved in GPDE are robust.

Acknowledgements The authors wish to thank the anonymous reviewers, whose valuable comments lead to an improved version of the paper. This work was supported by the National Natural Science Foundation of China under Grant Nos. 71701187, 71771166 and 71471126, and Research Project of Zhejiang Education Department under Grant No. Y201738184, and High Performance Computing Center of Tianjin University, China.

Compliance with ethical standards

Conflict of interest All the authors declares that they have no conflict of interest.

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

References

Biswas S, Kundu S, Das S (2015) Inducing niching behavior in differential evolution through local information sharing. IEEE Trans Evol Comput 19(2):246–263

Bose D, Biswas S, Vasilakos AW, Laha S (2014) Optimal filter design using an improved artificial bee colony algorithm. Inf Sci 281:443–461

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

Cai Y, Wang J (2013) Differential evolution with neighborhood and direction information for numerical optimization. IEEE Trans Cybern 43(6):2202–2215

Cai Y, Wang J (2015) Differential evolution with hybrid linkage crossover. Inf Sci 320:244–287

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

Cuevas E, Zaldívar D, Pérez-Cisneros M, Oliva D (2013) Block-matching algorithm based on differential evolution for motion estimation. Eng Appl Artif Intell 26:488–498

Das S, Suganthan PN (2011a) Differential evolution: a survey of the state-of-the-art. IEEE Trans Evol Comput 15(1):4–31

Das S, Suganthan PN (2011b) Problem definitions and evaluation criteria for CEC 2011 competition on testing evolutionary algorithms on real world optimization problems. Jadavpur University, Kolkata, India, and Nanyang Technol. Univ., Singapore, Dec. 2010

Das S, Konar A, Chakraborty UK, Abraham A (2009) Differential evolution using a neighborhood based mutation operator. IEEE Trans Evol Comput 13(3):526–553

Dorigo M, Blum C (2005) Ant colony optimization theory: a survey. Theor Comput Sci 344:243–278

Draa A, Bouzoubia S, Boukhalfa I (2015) A sinusoidal differential evolution algorithm for numerical optimisation. Appl Soft Comput 27:99–126

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(3):1088–1113

Ghosh A, Das S, Chowdhury A, Giri R (2011) An improved differential evolution algorithm with fitness-based adaptation of the control parameters. Inf Sci 181(18):3749–3765

Goldberg D (1989) Genetic algorithms in search, optimization and machine learning. Addison-Wesley, New York

Gong WY, Cai ZH (2014) Parameter optimization of PEMFC model with improved multi-strategy adaptive differential evolution. Eng Appl Artif Intell 27:28–40

Gong WY, Cai ZH, Ling CX, Li H (2011a) Enhanced differential evolution with adaptive strategies for numerical optimization. IEEE Trans Syst Man Cybern B Cybern 41(2):397–413

Gong WY, Fialho A, Cai ZH, Li H (2011b) Adaptive strategy selection in differential evolution for numerical optimization: an empirical study. Inf Sci 181(24):5364–5386

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(1):43–63

Idris I, Selamat A, Omatu S (2014) Hybrid email spam detection model with negative selection algorithm and differential evolution. Eng Appl Artif Intell 28:97–110

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

Karafotias G, Hoogendoorn M, Eiben AE (2015) Parameter control in evolutionary algorithms: trends and challenges. IEEE Trans Evol Comput 19(2):167–187

Kennedy J, Eberhart R, Shi Y (2001) Swarm intelligence. Morgan Kaufman, San Francisco

Lan Y, Liu Y, Sun G (2012) Modeling fuzzy multi-period production planning and sourcing problem with credibility service levels. J Comput Appl Math 231:208–221

Li XD, Yao X (2012) Cooperatively coevolving particle swarms for large scale optimization. IEEE Trans Evol Comput 16(2):210–224

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

Lin L, Gen M (2009) Auto-tuning strategy for evolutionary algorithms: balancing between exploration and exploitation. Soft Comput 13(2):157–168

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

Mallipeddi R, Suganthan PN, Pan QK, Tasgetiren MF (2011) Differential evolution algorithm with ensemble of parameters and mutation strategies. Appl Soft Comput 11(2):1679–1696

Neri F, Tirronen V (2010) Recent advances in differential evolution: a survey and experimental analysis. Artif Intell Rev 33(1–2):61–106

Qin AK, Huang VL, Suganthan PN (2009) Differential evolution algorithm with strategy adaptation for global numerical optimization. IEEE Trans Evol Comput 13(2):398–417

Sarker R, Elsayed SM, Ray T (2014) Differential evolution with dynamic parameters selection for optimization problems. IEEE Trans Evol Comput 18(5):689–707

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

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

Sun G, Zhao R, Lan Y (2016) Joint operations algorithm for large-scale global optimization. Appl Soft Comput 38:1025–1039

Sun G, Peng J, Zhao R (2017) Differential evolution with individual-dependent and dynamic parameter adjustment. Soft Comput. doi:10.1007/s00500-017-2626-3

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

Tang LX, Dong Y, Liu J (2015) Differential evolution with an individual-dependent mechanism. IEEE Trans Evol Comput 19(4):560–574

Wang S, Watada J (2012) A hybrid modified PSO approach to VaR-based facility location problems with variable capacity in fuzzy random uncertainty. Inf Sci 192(1):3–18

Wang Y, Cai Z, Zhang Q (2011) Differential evolution with composite trial vector generation strategies and control parameters. IEEE Trans Evol Comput 15(1):55–66

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

Wang J, Liao J, Zhou Y, Cai Y (2014) Differential evolution enhanced with multiobjective sorting based mutation operators. IEEE Trans Cybern 44(12):2792–2805

Yang M, Li C, Cai Z, Guan J (2015) Differential evolution with auto-enhanced population diversity. IEEE Trans Cybern 45(2):302–315

Yang G, Tang W, Zhao R (2017) An uncertain workforce planning problem with job satisfaction. Int J Mach Learn Cybern 8(5):1681–1693

Yu W, Shen M, Chen W, Zhan Z, Gong Y, Lin Y, Liu O, Zhang J (2014) Differential evolution with two-level parameter adaptation. IEEE Trans Cybern 44(7):1080–1099

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

Zhang J, Avasarala V, Subbu R (2010) Evolutionary optimization of transition probability matrices for credit decision-making. Eur J Oper Res 200(2):557–567

Zhao J, Xu Y, Luo F, Dong Z, Peng Y (2014) Power system fault diagnosis based on history driven differential evolution and stochastic time domain simulation. Inf Sci 275:13–29

Zhu W, Tang Y, Fang J, Zhang W (2013) Adaptive population tuning scheme for differential evolution. Inf Sci 223:164–191