

Received January 17, 2020, accepted February 1, 2020, date of publication February 12, 2020, date of current version February 24, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2973460

# Correlations Between the Scaling Factor and Fitness Values in Differential Evolution

XIAOLONG ZHU<sup>1,2</sup>, HAO ZHANG<sup>1,2</sup>, AND YANG GAO<sup>2</sup>

<sup>1</sup>School of Aeronautics and Astronautics, University of Chinese Academy of Sciences, Beijing 100049, China

<sup>2</sup>Technology and Engineering Center for Space Utilization, Chinese Academy of Sciences, Beijing 100094, China

Corresponding author: Hao Zhang (hao.zhang.zhr@gmail.com)

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFB1900605, in part by the Key Research Program of the Chinese Academy of Sciences under Grant ZDRW-KT-2019-1-0102, and in part by the Strategic Priority Program on Space Science of the Chinese Academy of Sciences under Grant XDA15014902.

**ABSTRACT** Designing fitness-based adaptive scaling factor ( $F$ ) is an effective method to enhance the performance of differential evolution (DE) algorithms. This paper investigates the correlations between  $F$  and fitness values of target vectors, base vectors and difference vectors. The correlations are described by the notations of monotonicity and nonlinearity. Monotonicity is used to examine whether the optimization performance of DE and the fitness values of certain vectors have positive or negative correlation. Nonlinearity denotes the operation in which nonlinear mappings are used to redistribute the values of  $F$  in  $[0, 1]$  so as to boost the optimization performance. These two aspects of correlations are empirically tested on the Numerical Optimization Competition benchmark functions in IEEE Congress on Evolutionary Computation. Simulation results reveal different qualitative and quantitative correlations between  $F$  and fitness values of different vectors. Then, a new  $F$  that combines these relations is designed. Its strength is numerically verified by testing different CEC Benchmark functions.

**INDEX TERMS** Differential evolution, scaling factor, fitness values, CEC benchmark problems, single objective optimization.

## I. INTRODUCTION

The differential evolution (DE) algorithm invented by Storn and Price [1] is a powerful population-based global searching tool. DE is believed to be effective for problems involving nonlinear and non-differentiable functions [2]. The number of DE research articles indexed in Science Citation Index database (via Web of Science) during 2007 to 2015 was 8714, as indicated in Reference [3], [4]. DE has been successfully implemented in diverse areas, such as spacecraft trajectory design [5] and statistical fisheries model estimation [6]. The major applications were also summarized in Ref. [3], [4].

DE employs the difference between distinct members from the current population as a guidance to search for a better solution. Compared with other intelligent algorithms, DE has the merits of few control parameters, good optimization performance and low space complexity [3]. Nevertheless, as an evolutionary algorithm, DE needs to compute large numbers of fitness functions to obtain the global optimum. To improve

the performance, different types of enhanced DE have been developed in recent years. The comprehensive surveys of DE can be found in [3], [4], [7]. Among various techniques to improve DE, choosing good values of the scaling factor ( $F$ ) in each generation is commonly an efficient option. Generally, the mechanisms to design  $F$  can be categorized into four groups: fixed value, random value, history-based adaption and fitness-based adaption. Fixed value indicates that  $F$  remains constant during the whole optimization. Storn and Price [1] indicated that  $F$  is not difficult to choose for good results. In their opinion, 0.5 can be a good initial choice of  $F$ . After testing different parameter settings for DE on the Sphere, Rosenbrock's and Rastrigin's functions, Gämperle *et al.* [8] found that the global searching ability and the convergence are very sensitive to the value of  $F$ . They suggested 0.6 as the initial choice. In another paper, Rönkkönen *et al.* [9] stated that setting  $F = 0.9$  can balance well between the speed and probability of convergence. The benefits of fixed value lies in its simplicity. For complex problems such as the multimodal optimization problem [10] and problems with constrained experimental domain [11],

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Wei<sup>1</sup>.

the fixed value has been successfully employed. Alternatively, the value of  $F$  can be updated by random functions, namely, random value type. Das *et al.* [12] proposed the DE with Random Scale Factor (DERSF) and the DE with Time Varying Scale Factor (DETVSF). DERSF allows for stochastically scaling difference vectors, and thus, can help to retain population diversity. With DETVSF, individuals are encouraged to sample diverse zones of the search space during the early stages of the search. In the late stage, they tend to exploit the interior of a relatively small space in which the suspected global optimum lies. In the SaDE algorithm [13],  $F$  is varied by a normal distribution with a mean value of 0.5 and a standard deviation of 0.3. By doing so, SaDE attempts to maintain both exploration capability (with large  $F$  values) and exploitation capability (with small  $F$  values). Compared with fixed value, randomization can produce more values of  $F$ . Thus, it can enhance the performance to some extent. In some modified DE, such as TLBSaDE [14] and MDE [15], random values of  $F$  were used. The third type, namely history-based adaption technique, adaptively computes  $F$  by learning from the past generations of successes. It is widely applied in adaptive and self-adaptive DE algorithms, such as JADE [16], jDE [17] and SHADE [18]. In recent years, many SHADE-based algorithms [19] have been proposed; and some have performed well in testing different IEEE CEC benchmark functions. However, the mechanism to design the scaling factor in these improved DE algorithms is similar to that in SHADE. Based on the analysis in [20], an ensemble sinusoidal approach to automatically adapt the values of  $F$  was designed in LSHADE-EpSin [21]. It is believed that the performance of LSHADE-EpSin is better than that of SHADE. EsDEr-NR [22] is an enhanced version of LSHADE-EpSin. The last type of scaling-factor designing technique is fitness-based adaption, in which  $F$  is usually determined by fitness values from the current population. The first research concerning fitness-based adaption was by Ali and Törn [23]. It employed the minimum and maximum fitness values of current generation to calculate  $F$ . Ghosh *et al.* [24] developed a new fitness-based technique considering the fitness difference between the target vector and the best vector. Based on the idea that  $F$  for individuals with higher fitness values are larger, Tang *et al.* [25] designed the rank-based scheme and value-based scheme. In 2017, Mohamed introduced the triangular mutation scheme. In that paper, the adaptive scheme of  $F$  also takes into account both the minimum and maximum fitness values in the current generation [26].

As can be seen from the aforementioned reviews of  $F$ -designing techniques, history-based and fitness-based adaptive  $F$  are favorable in practice. However, compared to the abundance in history-based adaptive strategies, few researches have addressed fitness-based adaptive schemes. Besides, most existing fitness-based methods mainly focus on the minimum and maximum fitness values in each generation [23]–[26]. So far, the fitness values of more vectors have been largely ignored, which we believe has encoded

important information about general structure of the fitness function. Thus, each individual's fitness should be exploited so as to obtain good values of  $F$ . In order to understand the scaling factor from a fitness-based perspective, this paper comprehensively studies the correlations between  $F$  and fitness values.

To that end, we propose a way to define the fitness-based correlation. The correlation addresses fitness values of the target vector, the base vector and the difference vector. Then, qualitative and quantitative relations are found by testing on the IEEE CEC 2014 problems. To show the potential of these correlations, a new  $F$  that combines the correlations is designed. The performance of the new  $F$  are verified on IEEE CEC 2014 and 2017 problems. Several classic and recent  $F$ -designing techniques are employed as a comparison.

The remainder of this paper is organized as follows: Section II reviews the classical DE and improved DE. Section III details the method to establish the correlations between  $F$  and different fitness values. Section IV discusses the correlations based on the numerical experiments. Section V concludes the whole paper.

## II. CLASSICAL DE AND IMPROVED DE

### A. CLASSICAL DE ALGORITHM

In this subsection, the classical DE algorithm [1] is briefly reviewed. In the rest of the paper, it is assumed that minimization problems are to be resolved. There are four basic steps in the classical DE: initialization, mutation, crossover and selection.

#### 1) INITIALIZATION

The population of DE is represented as

$$\mathbf{x}_i = (x^1, x^2, \dots, x^D), \quad i = 1, 2, \dots, NP \quad (1)$$

where  $D$  is the dimension of variables and  $NP$  denotes the population size.

The minimum and maximum of  $\mathbf{x}$  are defined as

$$\begin{cases} \mathbf{x}_{\min} = (x_{\min}^1, x_{\min}^2, \dots, x_{\min}^D) \\ \mathbf{x}_{\max} = (x_{\max}^1, x_{\max}^2, \dots, x_{\max}^D) \end{cases} \quad (2)$$

Then, a common method to initialize the  $i$ -th individual  $\mathbf{x}_{i,0}$  is

$$x_{i,0}^j = x_{\min}^j + \text{unif}(0, 1) \cdot (x_{\max}^j - x_{\min}^j), \quad j = 1, 2, \dots, D \quad (3)$$

where  $\text{unif}(0, 1)$  is a uniformly distributed random variable within the range of  $[0, 1]$ .

#### 2) MUTATION

Let  $\mathbf{x}_{i,G}$  be an individual at generation  $G$ . After initialization, a **donor vector**  $\mathbf{v}_{i,G}$  with respect to the **target vector**  $\mathbf{x}_{i,G}$  is produced by the following mutation operator:

$$\mathbf{v}_{i,G} = \mathbf{x}_{r_1^i,G} + F \cdot (\mathbf{x}_{r_2^i,G} - \mathbf{x}_{r_3^i,G}) \quad (4)$$

where the indices  $r_1^i, r_2^i$  and  $r_3^i$  are randomly generated mutually exclusive integers within the range of  $[1, NP]$  and are all different from index  $i$ . These indices are randomly generated once for each donor vector. Here,  $\mathbf{x}_{r_1^i, G}$  is termed the **base vector**, and  $\mathbf{x}_{r_2^i, G} - \mathbf{x}_{r_3^i, G}$  is called the **difference vector**.  $F$  is the scaling factor and is usually constrained in the range of  $[0, 1]$ .

### 3) CROSSOVER

The purpose of the crossover operator is to produce a trial vector  $\mathbf{u}_{i, G}$  by combining  $\mathbf{x}_{i, G}$  and  $\mathbf{v}_{i, G}$ . Let  $u_{i, G}^j$  be the  $j$ -th components of  $\mathbf{u}_{i, G}$ . The following rule is applied elementwisely:

$$u_{i, G}^j = \begin{cases} v_{i, G}^j, & \text{if } (j = j_{\text{rand}}) \text{ or } (r_{i, j} \leq Cr) \\ x_{i, G}^j, & \text{otherwise} \end{cases} \quad (5)$$

where  $j_{\text{rand}}$  is a randomly chosen integer in the range of  $[1, D]$ ,  $r_{i, j} = \text{unif}(0, 1)$ , and  $Cr$  is the crossover rate and defined in the range of  $[0, 1]$ .

### 4) SELECTION

Once  $\mathbf{u}_{i, G}$  is generated, the fitness values of  $\mathbf{u}_{i, G}$  and  $\mathbf{x}_{i, G}$  are calculated and compared. The vector that survives to the next generation is selected by the following rule:

$$\mathbf{x}_{i, G+1} = \begin{cases} \mathbf{u}_{i, G}, & f(\mathbf{u}_{i, G}) \leq f(\mathbf{x}_{i, G}) \\ \mathbf{x}_{i, G}, & \text{otherwise} \end{cases} \quad (6)$$

where  $f(\mathbf{u}_{i, G})$  and  $f(\mathbf{x}_{i, G})$  represent the fitness values of  $\mathbf{u}_{i, G}$  and  $\mathbf{x}_{i, G}$ .

The framework of the classical DE is shown in Algorithm 1.

---

#### Algorithm 1 Framework of the Classical DE

---

```

1:  $G = 0$ 
2: // Initialization
3: Generate a random initial population  $\mathbf{x}_{i, 0}$ 
4: Calculate the fitness values of the initial population  $f(\mathbf{x}_{i, 0})$ 
5: while The termination criteria are not met do
6:    $G = G + 1$ 
7:   // Mutation
8:   Generate  $F_{i, G}$ 
9:    $\mathbf{v}_{i, G} = \mathbf{x}_{r_1^i, G} + F_{i, G} \cdot (\mathbf{x}_{r_2^i, G} - \mathbf{x}_{r_3^i, G})$ 
10:  // Crossover
11:  Generate  $Cr_{i, G}$ 
12:   $u_{i, G}^j = \begin{cases} v_{i, G}^j, & \text{if } (j = j_{\text{rand}}) \text{ or } (r_{i, j} \leq Cr_{i, G}) \\ x_{i, G}^j, & \text{otherwise} \end{cases}$ 
13:  // Selection
14:   $\mathbf{x}_{i, G+1} = \begin{cases} \mathbf{u}_{i, G}, & f(\mathbf{u}_{i, G}) \leq f(\mathbf{x}_{i, G}) \\ \mathbf{x}_{i, G}, & \text{otherwise} \end{cases}$ 
15: end while

```

---

### B. IMPROVED DE

Generally, the techniques to improve DE can be categorized into four aspects, by respectively or combinedly designing the following: mutation operator, population size  $NP$ , scaling factor  $F$ , and crossover rate  $Cr$ . To classify the different variants of mutation operators, the notation “DE/x/y/z” is introduced. Here, x represents the base vector to be perturbed, y donates the number of difference vectors, and z stands for the type of crossover. Two types of crossover have been considered, which are exponential (exp) and binomial (bin). The binomial type is mostly used and the “DE/x/y/z” notation is usually shortened as “DE/x/y”. In the first paper of DE [1], the classical DE can be noted as DE/rand/1. Then, DE/best/1, DE/rand/2, DE/best/2 [27] and DE/current-to-pbest/1 [16] were proposed and widely used in other improved DE. More complicated mutation operators were designed in [26], [28]–[30]. It should be noted that no matter which mutation operator is chosen for DE, designing the values of  $NP$ ,  $F$  and  $Cr$  are always necessary. In 2006, Teo [31] firstly demonstrated the feasibility of self-adapting the population size parameter in DE. L-SHADE [32] showed the powerful performance improvement and ranked as the best algorithm in IEEE CEC 2014 problems. In that paper, the linear population size reduction technique was used. Poláková *et al.* [33] improved the population reduction technique and enables to decrease or increase the population size during the search. In EsDEr-NR [22], the niching-based population reduction method was employed to determine the number of population in each generation. As for  $F$  and  $Cr$ , many significant developments have also been performed, such as the invention of jDE [17], JADE [16], and DE-RCO [34]. Table 1 lists six improved DE algorithms. Here “1” indicates that one of the aforementioned features has been elaborately designed, “0” refers to retaining the original setting; “–” means that the parameter is not mainly designed, and the bold “1” donates the primary concerned parameter in the paper.

TABLE 1. Improve DE (compared to classical DE).

	mutation	$NP$	$F$	$Cr$	Year
jDE	0	0	<b>1</b>	<b>1</b>	2006
JADE	<b>1</b>	0	<b>1</b>	<b>1</b>	2009
EsDEr-NR	1	<b>1</b>	<b>1</b>	1	2018
DE-RCO	–	–	–	<b>1</b>	2018
[29]	<b>1</b>	–	–	–	2019
[33]	–	<b>1</b>	–	–	2019

As can be seen, large numbers of modern DE algorithms have complex strategies to mutually tuning all these factors. However, recent work appears to be concentrating on studying only one parameter. One one hand, focusing on one parameter can help to provide insights into understanding DE algorithms. On the other hand, new advances in tuning a single parameter can be plugged into existed DE algorithms to further improve performance, as is evident in [29] and [34]. Note that  $F$  is the unique feature of DE algorithms,

as compared with  $NP$  and  $Cr$ . In fact,  $F$  is strongly bound to mutation operators. For example, In (4),  $F$  is used to scale the difference vector. In JADE [16],  $F$  is responsible for two difference vectors. As a first step to understand the fitness-adaptive DE, we choose mutation operator as in (4). Based on (4), the correlations between  $F$  and fitness values of different vectors will be discussed.

### III. CORRELATIONS BETWEEN $F$ AND FITNESS VALUES

#### A. ARCHITECTURE OF THE CORRELATIONS

Note from Section II-A that in a DE algorithm, the trail vector is the vector that is to be evaluated and compared. As given by the diagram in (7), the contribution of  $F$  on the trail vector is mainly through the donor vector. It may be at first glance that the donor vector and the target vector are independent with each other. However, donor vector is weakly coupled with the target vector since the base vector and the difference vector are generated deliberately different from the the target vector. Thus, in this work it is reasonable to assume that  $F$  is related to the fitness values of target vectors, base vectors and difference vectors. In the later context, these vectors will be referred as **tested vectors**.

$$\left. \begin{array}{l} \text{base vector} \\ \text{difference vector} \\ \text{target vector} \end{array} \right\} \xrightarrow{F} \text{donor vector} \xrightarrow{Cr} \text{trail vector} \quad (7)$$

Let  $f(\cdot)$  denote the fitness value of “ $\cdot$ ”. To evaluate the correlations between  $F$  and fitness values of these vectors, the functional form of  $F$  is written as

$$\begin{cases} F_T = g_T(f_{r_i, G}, f_{\min, G}, f_{\max, G}) \\ F_B = g_B(f_{r_1^i, G}, f_{\min, G}, f_{\max, G}) \\ F_D = g_D(f_{r_2^i, G} - f_{r_3^i, G}, f_{\min, G}, f_{\max, G}) \end{cases} \quad (8)$$

where  $r_i, r_1^i, r_2^i, r_3^i$  refer to the indices in classical DE,  $g_T, g_B, g_D$  represent the contributions of target vector, base vector and difference vector, respectively. In the following the functions  $g_T, g_B, g_D$  are to be resolved.

To that end, several functional forms of  $g(\cdot)$  functions are considered to computed  $F$ . The computed  $F$  are then plugged into existing DE algorithms to run a large number of numerical experiments. The performance of each  $g(\cdot)$  function is recorded and then compared. Good functional forms of  $F$  can thus be identified. Then, correlations between  $F$  and fitness values of tested vectors are obtained.

In this paper, the following two features of  $g(\cdot)$  functions are mainly considered: monotonicity and nonlinearity. Here “monotonicity” refers to the comparison between optimization performance improvement/deterioration with  $F$  computed from two reversely-designed formulae. For example, it is termed as “positive correlation” if  $F$  with  $g(f(v))$  performs better than that with  $1 - g(f(v))$ , where  $g(\cdot)$  is a monotonically increasing function of  $v$  and  $v$  is a tested vector. On the other hand, “nonlinearity” denotes the operation that redistributes values of  $F$  in  $[0, 1]$ . In this work we use

modified power functions (see later in (15)) to account for the nonlinearity.

#### B. MONOTONICITY

In the following,  $g_T, g_B, g_D$  are designed elaborately. For  $g_T, F$  can be designed from the perspective that how much relative deficiency of the target vector’s fitness as compared with the current best. It is denoted as “proportion type” and can be written as

$$F_{p,t} = \frac{f(x_{i,G}) - f_{\min,G}}{f_{\max,G} - f_{\min,G}} \quad (9)$$

where  $f_{\min,G}$  and  $f_{\max,G}$  are the minimum and maximum of fitness values of generation  $G$ .

On the contrary,  $F$  can be understood as the improvement of the target vector’s fitness by comparing with the current worst. It is termed as “reverse proportion” and it is written as

$$F_{rp,t} = \frac{f_{\max,G} - f(x_{i,G})}{f_{\max,G} - f_{\min,G}} \quad (10)$$

Note that  $F_{p,t} + F_{rp,t} = 1$ .

Similarly, for  $g_B$ , the proportion-type of  $F$  is

$$F_{p,b} = \frac{f(x_{r_1,G}) - f_{\min,G}}{f_{\max,G} - f_{\min,G}} \quad (11)$$

and the reverse-proportion type of  $F$  is

$$F_{rp,b} = \frac{f_{\max,G} - f(x_{r_1,G})}{f_{\max,G} - f_{\min,G}} \quad (12)$$

For  $g_D$ , the proportion-type of  $F$  is

$$F_{p,d} = \left| \frac{f(x_{r_2,G}) - f(x_{r_3,G})}{f_{\max,G} - f_{\min,G}} \right| \quad (13)$$

and reverse-proportion type of  $F$  is

$$F_{rp,d} = 1 - \left| \frac{f(x_{r_2,G}) - f(x_{r_3,G})}{f_{\max,G} - f_{\min,G}} \right| \quad (14)$$

Here  $F_{p,d}$  and  $F_{rp,d}$  can be understood as the local relative roughness/smoothness of the fitness function, thus providing an intuitive way to characterize the local structure.

#### C. NONLINEARITY

The better  $F$  in Section III-B, denoted by  $F_0$  is used as a benchmark of performance. Then, the following expressions of  $F_0$  are searched to improve the performance. The exact expressions are

$$\begin{cases} \text{power 1 : } F = F_0 \\ \text{power 1/2 : } F = \sqrt{F_0} \\ \text{power 2 : } F = F_0^2 \\ \text{power 1/3 : } F = 1/2 + 1/2 \cdot \sqrt[3]{2 \cdot F_0 - 1} \\ \text{power 3 : } F = 1/2 + 1/2 \cdot (2 \cdot F_0 - 1)^3 \end{cases} \quad (15)$$

Figure 1 presents 5 curves of (15), respectively. Apparently, (15) redistributes the original  $F_0$  into new concentrations in  $[0, 1]$  by using nonlinear maps. The 5 cases in (15) represent 5 typical maps. Given 1000 samples of  $F_0$  that are uniformly distributed in  $[0, 1]$ , Fig. 2 presents the new distribution of  $F$  after applying (15). As can be seen from Fig. 2, these maps have distinct behaviors. For power  $1/2$  and power 2, the new  $F$  spans the whole  $[0, 1]$ . However, the power-2-map tends to concentrate the value of  $F$  into smaller values whereas the power- $1/2$ -map prefers larger values. Similarly, the power- $1/3$ -map and the power-3-map have two reverse behaviors. The power- $1/3$ -map produces two peaks around the left end and right end regions, whereas a condense  $F$  around 0.5 can be seen from the power-3-map. Thus, (15) can be used to represent most of the change processes from  $F_0$  to  $F$ .

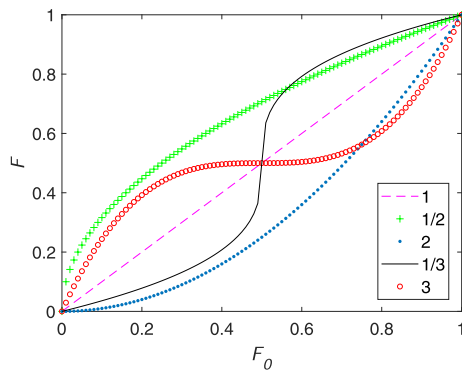


FIGURE 1. Different powers of  $F_0$ .

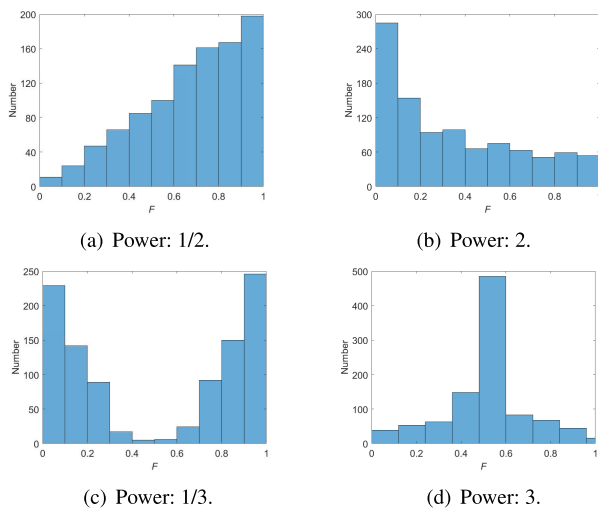


FIGURE 2. Histograms of  $F$  transformed from the uniform  $F_0$  with different powers.

IV. NUMERICAL EXPERIMENTS AND RESULTS

In this section, numerical experiments are designed to test the performance of different  $F$ . Inspired by the machine learning method, a “training set” is employed to determine the exact (proportion/reverse-proportion) types and the best powers of

$F$  for the tested vectors. Then a “test set” is used to validate the performance of these correlations. For the training set, “Real-Parameter Single Objective Optimization” of IEEE CEC 2014 (hereafter IEEE CEC 2014 problems) [35] are adopted. IEEE CEC 2014 problems have 30 benchmark functions: 3 unimodal functions, 13 simple multimodal functions, 6 hybrid functions and 8 composition functions. Both unimodal/multimodal and separable/non-separable problems are included. Many of the functions have large numbers of local optimums. In some cases, such as function 11 and 12, the second better local optimum is far from the global optimum.

We reversely design two relations to find out the better monotonicity formulation for each tested vector. For example, (9) and (10) on target vector. These two relations, denoted by proportional and reverse-proportional formula, are used to solve the IEEE CEC 2014 problems. The one that performs better is chosen as the representative monotonic relation.

For the test set, the benchmark functions (without function 2) in “Real-Parameter Single Objective Optimization” of IEEE CEC 2017 (hereafter IEEE CEC 2017 problems) [36] are considered. In IEEE CEC 2017 problems, there are also 30 benchmark functions: 3 unimodal functions, 7 simple multimodal functions, 10 hybrid functions and 10 composition functions.

A. MONOTONICITY AND NONLINEARITY OF A SINGLE VECTOR

Before solving an optimization problem,  $Cr$  and  $NP$  need to be designed. According to the paper of [13],  $Cr$  is usually sensitive to problems with different characteristics. Thus, the mechanism of determine the value  $Cr$  should be designed carefully. In this section, the method to design  $Cr$  is designed to be the same as that in  $EsDE_r$ -NR [22]. As for  $NP$ , two cases are concerned: (1) adaptive  $NP$ , the same as that in  $EsDE_r$ -NR [22]; (2) Fixed  $NP$ ,  $NP = 5D$  [8],  $D$  is the dimension of the problem.

1) ADAPTIVE NP

First, the adaptive  $NP$  is used.

a: TARGET VECTOR

Table 2 compares (9) and (10) in testing the 10-D version of IEEE CEC 2014 problems. Here the benchmark is set as (10). In order to analyze the solution quality from a statistical point of view, the results are compared using the Wilcoxon’s rank-sum test with a significance level of 0.05 [37]. For each  $F$  and each benchmark function, the tests are run 51 independently. The mean and standard deviation of the errors for these runs are recorded in the table. The best solutions with the smallest error mean values for each function are marked in boldface font. After comparison, one of three signs (+, -, =) is assigned. “+” means that (9) performs significantly better than (10); “-” means that (9) performs worse than (10); When the two  $F$  have no obvious performance difference, their relation is represented as “=”. Learning from the table,

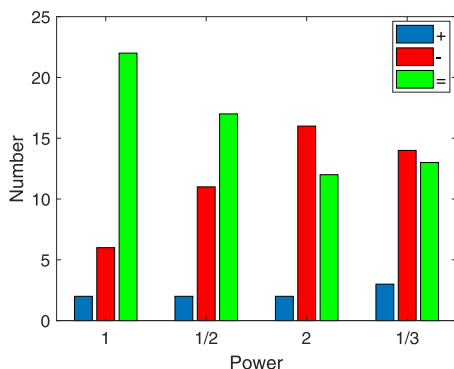


**TABLE 2. Comparison results for monotonicity of the target vector on IEEE CEC 2014 problems for  $D = 10$ .**

$f_i$	(9)			(10)	
	Mean	Std	+/-/=	Mean	Std
1	<b>4.33E-06</b>	2.05E-05	=	7.31E-05	4.69E-04
2	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00
3	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00
4	<b>3.36E+00</b>	9.37E+00	=	7.50E+00	1.37E+01
5	1.64E+01	6.81E+00	=	<b>1.63E+01</b>	5.86E+00
6	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00
7	4.66E-02	3.01E-02	-	<b>2.18E-02</b>	1.05E-02
8	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00
9	4.47E+00	1.94E+00	-	<b>3.20E+00</b>	8.94E-01
10	1.53E-01	4.67E-01	-	<b>8.57E-03</b>	2.17E-02
11	1.31E+02	1.22E+02	=	<b>9.74E+01</b>	5.78E+01
12	<b>1.64E-01</b>	1.14E-01	+	2.56E-01	5.31E-02
13	<b>2.95E-02</b>	1.37E-02	+	1.04E-01	3.30E-02
14	<b>9.89E-02</b>	4.11E-02	+	1.34E-01	4.50E-02
15	<b>6.41E-01</b>	1.75E-01	=	6.52E-01	8.78E-02
16	<b>1.48E+00</b>	4.73E-01	=	<b>1.48E+00</b>	2.53E-01
17	<b>3.71E+00</b>	6.56E+00	=	9.63E+00	2.84E+01
18	7.16E-01	6.24E-01	=	<b>7.14E-01</b>	6.50E-01
19	2.84E-01	3.55E-01	=	<b>1.15E-01</b>	4.67E-02
20	3.56E-01	3.35E-01	-	<b>1.85E-01</b>	1.85E-01
21	<b>2.96E-01</b>	2.52E-01	=	1.46E+00	8.11E+00
22	1.83E-01	1.93E-01	=	<b>1.55E-01</b>	1.45E-01
23	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00
24	1.11E+02	3.03E+00	-	<b>1.10E+02</b>	1.35E+00
25	1.42E+02	2.72E+01	-	<b>1.20E+02</b>	1.66E+01
26	<b>1.00E+02</b>	1.29E-02	+	<b>1.00E+02</b>	4.53E-02
27	<b>5.44E+00</b>	2.78E+01	=	2.10E+01	5.96E+01
28	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	3.25E-12
29	1.98E+02	9.33E+00	=	<b>1.85E+02</b>	3.06E+01
30	<b>3.48E+02</b>	1.35E+02	=	3.52E+02	1.47E+02
Number of (+/-/=) 4/6/20					

(10) performs better in 6 functions and worse in 4 functions. Thus, (10) is slightly better than (9).

Furthermore, the performance of mapping  $F$  in (10) via (15) is tested. Table 7 (See Appendix) records the detailed optimization results and the comparison results for each function. Figure 3 depicts the total numbers of “+”, “-” and “=” for different powers. Here the benchmark is set as power 3 of (10). It can be found that the 3 power of (10) outperforms 1/2, 2 and 1/3 to a large extent and is slightly better than 1. Thus, 3 is regarded as the best power of (10).



**FIGURE 3. Comparison results for different powers of the target vector on IEEE CEC 2014 problems for  $D = 10$ .**

*b: BASE VECTOR*

Table 3 shows the comparison results of (11) and (12). Here the benchmark is set as (11). Compared to (12), (11) is better in 13 functions but only worse in 6 functions. Intuitively, if the fitness value of the base vector is low, a smaller  $F$  is better because the donor vector can inherit more from the base vector. In other words, it prefers the exploitation operator. Then, the performances of different powers of (11) are compared. The optimization results and the comparison results for each function are given in Table 8 (See Appendix). Fig. 4 is the total numbers of “+”, “-” and “=” for different powers. Here the benchmark is set as power 3. Among the 5 types of powers, 1 and 3 perform the best. Considering that 3 finds more of the best solutions, 3 is chosen as the best power of (11).

**TABLE 3. Comparison results for monotonicity of the base vector on IEEE CEC 2014 problems for  $D = 10$ .**

$f_i$	(12)			(11)	
	Mean	Std	+/-/=	Mean	Std
1	1.56E+03	1.48E+03	-	<b>1.59E-08</b>	5.46E-08
2	2.01E+00	6.94E+00	-	<b>0.00E+00</b>	0.00E+00
3	1.45E-01	8.95E-01	-	<b>0.00E+00</b>	0.00E+00
4	<b>2.40E-01</b>	2.30E-01	+	1.70E+00	6.85E+00
5	1.76E+01	6.50E+00	-	<b>1.55E+01</b>	5.86E+00
6	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00
7	<b>1.16E-03</b>	3.00E-03	+	8.93E-03	1.13E-02
8	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00
9	3.24E+00	1.49E+00	=	<b>2.79E+00</b>	8.38E-01
10	3.31E-02	4.39E-02	-	<b>6.12E-03</b>	1.88E-02
11	1.08E+02	9.96E+01	=	<b>6.33E+01</b>	5.01E+01
12	4.65E-01	7.95E-02	-	<b>2.45E-01</b>	5.41E-02
13	1.06E-01	2.50E-02	-	<b>4.27E-02</b>	2.66E-02
14	1.49E-01	3.54E-02	-	<b>9.84E-02</b>	2.61E-02
15	1.20E+00	2.50E-01	-	<b>6.62E-01</b>	1.93E-01
16	1.29E+00	3.54E-01	=	<b>1.17E+00</b>	4.18E-01
17	2.83E+02	5.70E+02	-	<b>7.87E-01</b>	6.76E-01
18	3.89E+01	8.95E+01	-	<b>3.74E-01</b>	4.64E-01
19	<b>8.66E-02</b>	4.27E-02	+	1.49E-01	6.96E-02
20	<b>2.38E-01</b>	2.61E-01	=	2.42E-01	1.30E-01
21	7.80E+00	4.13E+01	-	<b>3.83E-01</b>	4.03E-01
22	<b>3.29E-02</b>	8.66E-02	+	1.17E-01	5.00E-02
23	<b>1.96E+02</b>	2.80E+01	=	2.00E+02	0.00E+00
24	<b>1.08E+02</b>	2.62E+00	+	1.09E+02	2.12E+00
25	1.20E+02	1.42E+01	=	<b>1.16E+02</b>	1.39E+01
26	<b>1.00E+02</b>	4.16E-02	-	<b>1.00E+02</b>	1.65E-02
27	<b>3.58E+00</b>	8.94E+00	+	5.69E+00	2.78E+01
28	<b>2.00E+02</b>	5.48E-12	=	<b>2.00E+02</b>	0.00E+00
29	2.00E+02	2.91E-07	=	<b>1.99E+02</b>	5.58E+00
30	<b>3.35E+02</b>	1.49E+02	=	3.50E+02	1.32E+02
Number of (+/-/=) 6/13/11					

*c: DIFFERENCE VECTOR*

Table 4 shows the comparison results for (13) and (14). Here the benchmark is set as (14). Equation (14) is better than (13) in 19 functions and worse in only 4 functions. Intuitively, when the fitness of difference vectors is small,  $F$  should be large enough to have a substantial perturbation on the base vector. In other words, it encourages exploration during searching. Then, based on (14), 5 different powers are tested, and the results are recorded in Table 9 (See Appendix). The total numbers of “+”, “-” and “=” for different powers

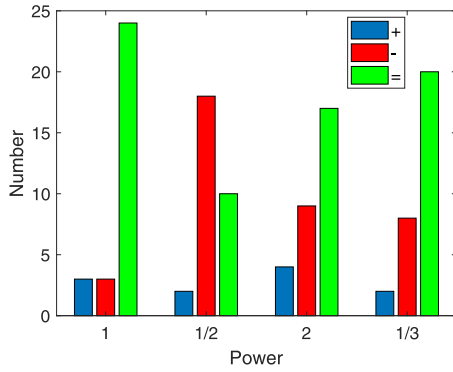


FIGURE 4. Comparison results for different powers of the base vector on IEEE CEC 2014 problems for  $D = 10$ .

TABLE 4. Comparison results for monotonicity of the difference vector on IEEE CEC 2014 problems for  $D = 10$ .

$f_i$	(13)			(14)	
	Mean	Std	+/-/=	Mean	Std
1	5.91E+03	9.64E+03	-	6.04E-07	3.75E-06
2	2.02E+02	5.97E+02	-	0.00E+00	0.00E+00
3	1.40E+00	7.77E+00	-	0.00E+00	0.00E+00
4	1.01E+01	1.54E+01	-	1.45E+00	6.83E+00
5	1.68E+01	6.05E+00	=	1.46E+01	6.74E+00
6	2.21E-03	9.60E-03	=	5.31E-02	1.39E-01
7	6.65E-02	3.76E-02	-	1.86E-02	1.04E-02
8	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00
9	5.62E+00	2.14E+00	-	3.17E+00	9.39E-01
10	6.52E-01	1.40E+00	-	1.22E-03	8.75E-03
11	1.71E+02	1.41E+02	-	9.24E+01	5.52E+01
12	1.41E-01	1.04E-01	+	2.66E-01	5.86E-02
13	3.19E-02	1.35E-02	+	1.27E-01	4.35E-02
14	1.35E-01	5.00E-02	-	1.07E-01	2.21E-02
15	6.31E-01	1.57E-01	+	6.96E-01	1.15E-01
16	1.68E+00	4.99E-01	-	1.46E+00	3.13E-01
17	1.32E+02	1.82E+02	-	1.16E+00	8.20E-01
18	9.81E+00	1.88E+01	-	5.42E-01	5.54E-01
19	5.47E-01	4.34E-01	-	1.45E-01	5.40E-02
20	1.47E+00	1.35E+00	-	2.87E-01	1.51E-01
21	1.07E+01	1.64E+01	-	2.94E-01	3.01E-01
22	3.07E-01	4.44E-01	=	1.70E-01	7.44E-02
23	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00
24	1.13E+02	3.15E+00	-	1.11E+02	1.66E+00
25	1.43E+02	2.49E+01	-	1.19E+02	1.23E+01
26	1.00E+02	1.49E-02	+	1.00E+02	4.65E-02
27	2.90E+01	6.62E+01	-	2.31E+00	4.50E-01
28	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00
29	2.01E+02	4.42E+00	=	1.99E+02	1.01E+01
30	3.92E+02	1.81E+02	-	3.71E+02	1.25E+02
Number of (+/-/=) 4/19/7					

are shown in Fig. 5. Here the benchmark is set as power 3. Obviously, the power of 3 outperforms the others.

d: DISCUSSIONS OF MONOTONICITY AND NONLINEARITY

Learning from experiments above, it is found that the correlations between  $F$  and fitness values of different vectors are different. Firstly, the number of equivalent results (tie) of (9)/(10), (11)/(12) and (13)/(14) with power-1 are 20, 11 and 7. It can be understood that the larger the number of ties, the less sensitive  $F$  is to the fitness of this vector. Namely,  $F$  is most sensitive to the fitness values of difference vectors.

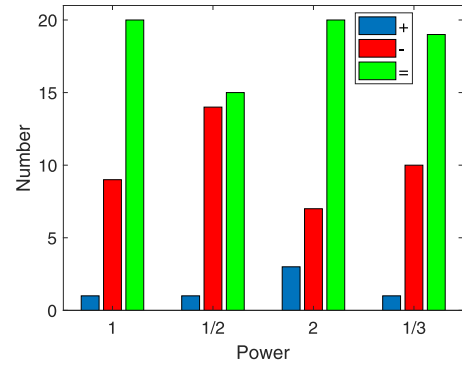


FIGURE 5. Comparison results for different powers of the difference vector on IEEE CEC 2014 problems for  $D = 10$ .

This is one of the most important primary features of the DE algorithms. By contrast,  $F$  is insensitive to the fitness values of target vectors.

Secondly, the number of good and bad results of (10)/(9), (11)/(12) and (14)/(13) with power-1 are 6/4, 13/6 and 19/4. Thus,  $F$  is in proportion to the fitness values of base vectors whereas having the opposite dependence on that of target vectors and difference vectors. It is interesting to note that the sensitivities are different. For example, the ratio 6/4 again indicates that a good  $F$  is weakly dependent on the target vector, since (9) and (10) have roughly the same trend of improvement. Thus, the analysis above provides insights of how DE works in a quantitative manner, which can guide the designing of  $F$ .

Thirdly, from the analysis of nonlinearity, it can be found that the best power of these vectors are all 3. Figure 2 reveals that the power-3-map tends to produce a condense  $F$  around 0.5. In the previous studies of [1] and [8], 0.5 and 0.6 are suggested to be initial value of  $F$ . Thus, power-3 is consistent with the conclusions in [1] and [8].

2) FIXED NP

In order to study the effect of  $NP$  on the correlations, in this subsection, the population size is fixed, namely,  $NP = 5D$ . Another round of comparison reveals that the correlations remain the same. The results are recorded in Table 5.

TABLE 5. Correlations of the tested vectors for adaptive NP and fixed NP.

Vector	Correlations	Adaptive NP	Fixed NP
Target vector	Monotonicity	(10)	(10)
	Nonlinearity	3	3
Base vector	Monotonicity	(11)	(11)
	Nonlinearity	3	3
Differential vector	Monotonicity	(14)	(14)
	Nonlinearity	3	3

B. COMBINATION OF THESE CORRELATIONS

In fact, the previously discovered correlations can be combined to obtain a novel way of designing  $F$ . As an illustrative example, in this section we choose a simple combination, namely, the average of the power-3 formulae

of (10), (11) and (14):

$$\bar{F} = \frac{1}{6}[(2F_{rp,t} - 1)^3 + (2F_{p,b} - 1)^3 + (2F_{rp,d} - 1)^3 + 3] \quad (16)$$

Similar as the treatment in Section III-C, we first evaluate the distribution of  $\bar{F}$ . Without loss of generality, we assume  $f(x) \in [0, 1]$ . We uniformly take 1000 samples of  $f(x)$ ; Fig. 6 presents the distribution of  $\bar{F}$ . Different from Section III-C, here  $\bar{F}$  is computed intermediately via  $F_{p,t}$ ,  $F_{rp,b}$  and  $F_{rp,d}$ , which are all directly computed by fitness functions. After being mapped by (15), values of (10), (11) and (14) are mostly concentrated around 0.5, of which the shape appears a normal distribution.

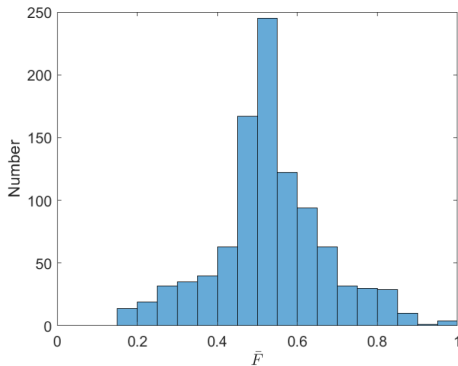


FIGURE 6. Histogram of  $\bar{F}$ .

Next, we consider applying a composite nonlinear function on (16). Although previously it is empirically revealed that the power-3 map in (15) is a good option, it is inappropriate to be directly used here since the distribution of  $\bar{F}$  is far from uniform distribution. On the contrary, we adopt a new strategy via a simple translation:

$$F = I(u) \quad (17)$$

where  $u = \bar{F} + \Delta F$ ,  $\Delta F = 0, \pm 0.1, \pm 0.2, \pm 0.3$  and

$$I(u) = \begin{cases} 1, & u \geq 1 \\ 0, & u \leq 0 \\ u, & \text{otherwise} \end{cases} \quad (18)$$

The performance of (17) under each  $\Delta F$  is compared in solving the 10-D version IEEE CEC 2014 problems. Here the benchmark is set as  $\Delta F = 0$ . Figure 7 shows that negative  $\Delta F$  can achieve better results; and  $\Delta F = -0.2$  is the best.

Inspired by this translation behavior, the final  $F$  takes the following form:

$$F = \bar{F}^2 = \frac{1}{36}[3 + (2F_{rp,t} - 1)^3 + (2F_{p,b} - 1)^3 + (2F_{rp,d} - 1)^3]^2 \quad (19)$$

This form comes from the observation that squaring a randomized variable in  $[0, 1]$  decreases the expectation, which is

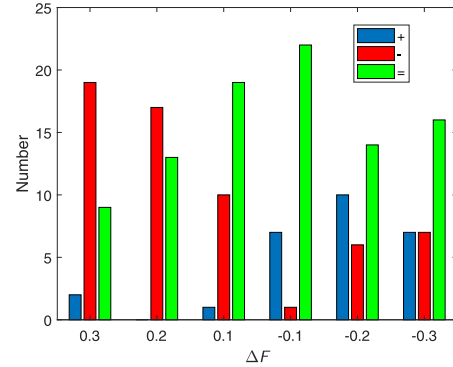


FIGURE 7. Comparisons of different  $\Delta F$  on IEEE CEC 2014 problems for  $D = 10$ .

similar to the translation behavior with  $\Delta F < 0$ . In addition, (19) increases the nonlinearity of (16).

Though (19) appears to be complicated, it only involves algebraic calculation of known fitness values. It is worthy to mention that these fitness values of the tested vectors have already been calculated in the stage of selection, namely, the fourth step in the classical DE, or Line 14 in Algorithm 1. Thus, (19) does not add extra burden on computer resources. In fact, numerical test shows that it only takes about 3 milliseconds to produce all the  $F$  values (1800 in total) in each generation (CPU: 3.60 GHz, RAM: 8 GB).

#### 1) TESTING $F$ IN (19) ON IEEE CEC 2014 PROBLEMS

The performance of (19) is tested on IEEE CEC 2014 problems. As mentioned in previous context,  $F$  is strongly associated with mutation operators. Since the current work is built onto the classic mutation operator, numerical comparisons are mainly limited into this type. Comparisons with other DE variants and other metanephritic algorithms are beyond the current scope. The adaptive  $Cr$  and  $NP$  are used. In order to comprehensively evaluate the strength, 7 types of  $F$  in literatures are used, which are:

- (1) 0.1 and 0.9 [9] are the fixed values;
- (2) rand [12] is the random value, and can be described as

$$F = \frac{1}{2}(1 + \text{unif}(0, 1)) \quad (20)$$

- (3) SinDE and SHADE are the history-based adaption; SinDE is designed to be the same as that used in [22] and is used during the whole generations, SHADE represents the method to design  $F$  in the SHADE algorithm;

- (4) FiADE [24] and Rbs [25] are the fitness-based adaption. FiADE refers to the following equation:

$$F_i = \max\left(0.8\left(1 - e^{-\Delta f_i}\right), 0.8\left(\Delta f_i / (\lambda + \Delta f)\right)\right) \quad (21)$$

where  $\Delta f_i = |f(x_i) - f(x_{best})|$  and  $\lambda = \Delta f_i / 10 + 10^{-14}$ .

Rbs is short for rank-based scheme:

$$F_i = \frac{i}{NP} \quad (22)$$



Table 10 (See Appendix) and Table 11 (See Appendix) record the optimization results and comparison results on the 30-D and 50-D IEEE CEC 2014 problems. Figure 8 and Figure 9 show the total numbers of “+”, “-” and “=” for different schemes of  $F$ . Here, the benchmark is set as the proposed  $F$  in (19). Compared with 0.1, 0.9, rand, SinDE, FiADE and Rbs, the performance of the proposed  $F$  is much better. If compared with SHADE, the proposed  $F$  seems to be slightly better. For 30-dimension problems, the proposed method has an 8–8 tie with SHADE. When the dimension is set to 50, the proposed method beats SHADE to a larger extent, namely, performing better in 13 functions whereas getting worse in 3 functions.

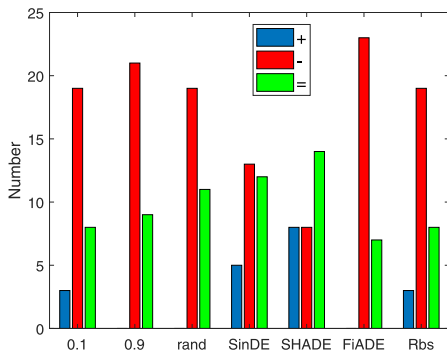


FIGURE 8. Comparisons with  $F$  in literatures on IEEE CEC 2014 problems for  $D = 30$ .

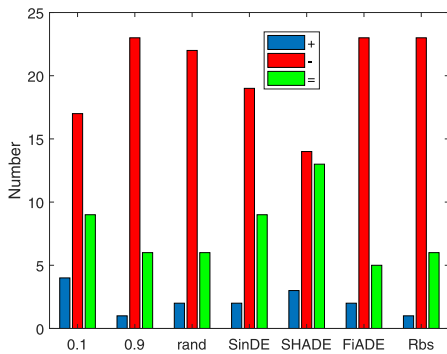


FIGURE 9. Comparisons with  $F$  in literatures on IEEE CEC 2014 problems for  $D = 50$ .

Figure 10 presents the convergence history in log scale with different  $F$ . Several benchmark functions are taken from the 50-D version of IEEE CEC 2014 problems, namely,  $f_1$ ,  $f_3$ ,  $f_{11}$ ,  $f_{12}$ ,  $f_{22}$ , and  $f_{23}$ . These functions include all the aforementioned types, namely, the unimodal functions ( $f_1$ ,  $f_3$ ), simple multimodal functions ( $f_{11}$ ,  $f_{12}$ ), hybrid functions ( $f_{22}$ ) and composition functions ( $f_{23}$  to  $f_{30}$ ), the time history fitness values with different  $F$  are all similar. In fact, the initial fitness values with (17) are the worst in this simulation. However, its performance catches up its peers very quickly; it is among the best only after first 100 generations, as can be seen from Fig. 10 (f). The strength of (17) can be better revealed by checking other functions. For  $f_1$  and  $f_3$ , the exploration capability of (17) is demonstrated.

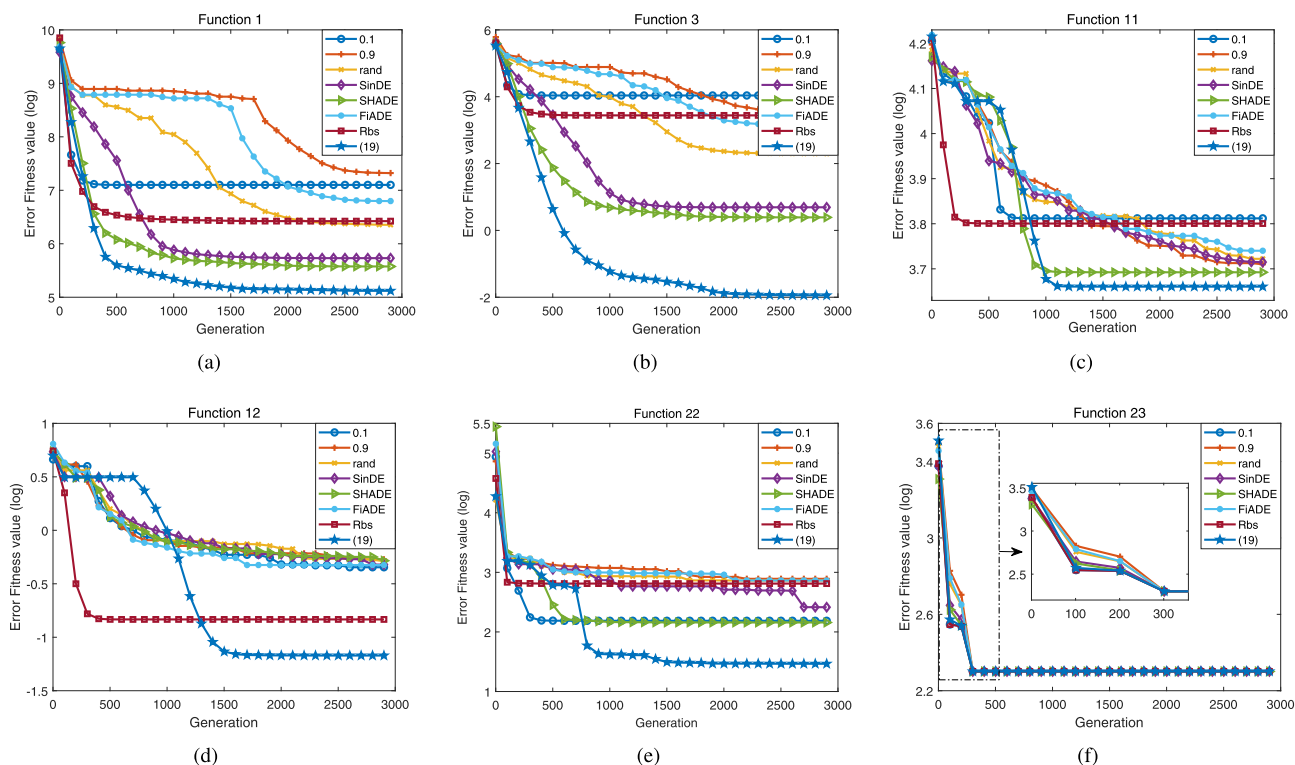
TABLE 6. Wilcoxon’s test between (19) and  $F$  in literatures using CEC2017 problems for  $D= 10, 30, 50$  and  $100$ , benchmark: (19).

$D$	$F$	$R^+$	$R^-$	p-value	+	=	-	Dec
10	0.1	0	435	<b>0.00</b>	0	2	28	-
	0.9	120	286	0.06	2	6	21	-
	rand	169	156	0.86	4	12	13	=
	SinDE	200	125	0.31	8	16	5	=
	SHADE	211	140	0.37	8	14	7	=
	FiADE	154	281	0.17	3	5	21	=
	Rbs	0	435	<b>0.00</b>	0	1	28	-
	EsDE <sub>r</sub> -NR	186	139	0.53	10	13	6	=
30	0.1	19	416	<b>0.00</b>	0	2	27	-
	0.9	0	435	<b>0.00</b>	0	0	29	-
	rand	0	435	<b>0.00</b>	0	1	28	-
	SinDE	151	284	0.15	8	2	19	=
	SHADE	158	248	0.31	8	8	13	=
	FiADE	0	435	<b>0.00</b>	0	0	29	-
	Rbs	0	435	<b>0.00</b>	0	0	29	-
	EsDE <sub>r</sub> -NR	144	291	0.11	7	6	16	=
50	0.1	82	353	<b>0.00</b>	7	1	21	-
	0.9	0	435	<b>0.00</b>	0	0	29	-
	rand	0	435	<b>0.00</b>	0	0	29	-
	SinDE	98	337	<b>0.01</b>	5	4	20	-
	SHADE	258	177	0.38	11	8	10	=
	FiADE	0	435	<b>0.00</b>	0	0	29	-
	Rbs	0	435	<b>0.00</b>	0	0	29	-
	EsDE <sub>r</sub> -NR	142	293	0.10	6	11	12	=
100	0.1	132	303	0.06	9	4	16	=
	0.9	7	428	<b>0.00</b>	0	0	29	-
	rand	32	403	<b>0.00</b>	0	0	29	-
	SinDE	60	375	<b>0.00</b>	0	5	24	-
	SHADE	341	94	<b>0.01</b>	15	6	8	+
	FiADE	15	420	<b>0.00</b>	0	1	28	-
	Rbs	106	329	<b>0.02</b>	9	0	20	-
	EsDE <sub>r</sub> -NR	286	149	0.14	13	4	12	=

The fitness value decreases rapidly during early generations. The exploitation capability of (17) can be seen by examining  $f_{11}$ ,  $f_{12}$  and  $f_{22}$ . Though the error fitness values with (17) in the early stage is not the best, its strength is evident after the generation exceeds 1000.

2) TESTING  $F$  IN (19) ON IEEE CEC 2017 PROBLEMS

Table 12 to Table 15 (See Appendix) are the simulation results for testing on IEEE CEC 2017 problems of dimensions of 10, 30, 50 and 100. Here, a slightly different statistical analysis, namely, the Wilcoxon signed rank test [38] is used to evaluate the performance. The comparison results are collected in Table 6. The benchmark is set as (19).  $R^+$  represents the sum of ranks for the test problems in which the aforementioned previous  $F$  performs better than (19);  $R^-$  represents the sum of ranks for the test problems in which the aforementioned previous  $F$  performs worse than (19). In this subsection, the technique of designing  $F$  that used in the original EsDE<sub>r</sub>-NR algorithm is also employed as a comparison. Setting the significance level to be 0.05, it can be found that the performance of most previous  $F$  are no better than that of (19). Specifically, In most of the dimensions, (19) performs better than 0.1, 0.9, rand, FiADE and Rbs. For SinDE, it is competitive to (19) if the dimension is low. However, when the dimension is high, such as 50 and 100, SinDE can no longer catch up with (19). For SHADE and



**FIGURE 10.** Convergence graph (error of best value (log) and optimal value curves) for different  $F$  on 50-D  $f_1, f_3, f_{11}, f_{12}, f_{22},$  and  $f_{23}$  in CEC 2014 benchmark functions. (a)  $f_1$ . (b)  $f_3$ . (c)  $f_{11}$ . (d)  $f_{12}$ . (e)  $f_{22}$ . (f)  $f_{23}$ .

EsDE<sub>r</sub>-NR, their performances are similar to that of (19). If the dimension is 100, SHADE becomes better.

It should be noted that the combination method of the correlations in (16) is preliminary. More advanced combination strategies may lead to better optimization performance. Studying different combination strategies of  $F$  with fitness values of different vectors are beyond the scope of this work. However, the primary purpose of this subsection is to demonstrate the potentials of these relations. Learning from the comparison results above, it can be concluded that the performance of the proposed  $F$  in (19) with the relations is competitive.

Limited by the classical mutation operator in (4) used in this work, the performance of the new  $F$  in (19) is occasionally less powerful than the original EsDE<sub>r</sub>-NR [22]. Nevertheless, this paper provides a new perspective to design fitness-based  $F$ . For complicated mutation operators such as in [22],  $F$  usually controls the scale of more than one difference vector. Predictably, the correlations between  $F$  and the vectors are more complicated. For example,  $F$  is likely to be related not only to the fitness values but also to the angles of these difference vectors. The detailed discussions about these correlations are beyond the research of the paper. However, the simulation results in this section already indicate the existence of correlations between  $F$  and the fitness values of many vectors. Thus, it is beneficial to exploit this phenomenon. As for specified results for complex mutation operators, it will be the focus of future work.

## V. CONCLUSION

This paper presents a novel method to investigate the correlations between  $F$  and fitness values of target vectors, base vectors and difference vectors in classical DE. By testing on the single-objective-optimization problems in IEEE CEC 2014 Competitions, the qualitative and quantitative dependency is obtained. It is found that  $F$  is in proportion to the fitness values of base vectors whereas it has the opposite dependence on that of target vectors and difference vectors. Compared with target vectors,  $F$  is more sensitive to the fitness values of the base vector and difference vector. To verify the potential of these correlations, a new  $F$  is designed that comprehensively combines these relations. The expression involves a second order power function of the arithmetic mean. Simulation results show that the proposed  $F$  outperforms most current schemes of  $F$ .

This work provides a new way to tune fitness-based adaptive parameters. It can be extended to design  $F$  for newly-developed mutation operators, or to design  $Cr$  for general metaheuristic algorithms. More advanced combination of fitness-based  $F$ -scheme, or correlation between scaling factor and recent mutation operators, may also be future research directions.

## APPENDIX NUMERICAL RESULTS

See Tables 7–15.

TABLE 7. Comparison results for different powers of the target vector on IEEE CEC 2014 problems for  $D = 10$ .

$f_i$	1			1/2			2			1/3			3		
	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=
1	7.31E-05	4.69E-04	-	<b>0.00E+00</b>	0.00E+00	=	9.60E+04	1.92E+05	-	3.37E+00	1.62E+01	-	<b>0.00E+00</b>	0.00E+00	-
2	0.00E+00	0.00E+00	=	<b>0.00E+00</b>	0.00E+00	=	4.82E+03	2.54E+04	-	2.44E-10	1.74E-09	=	<b>0.00E+00</b>	0.00E+00	=
3	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00	=	<b>3.79E+02</b>	7.90E+02	-	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00	=
4	7.50E+00	1.37E+01	=	<b>5.80E+00</b>	1.27E+01	=	1.19E+01	1.58E+01	-	9.55E+00	1.50E+01	-	7.16E+00	1.38E+01	-
5	1.63E+01	5.86E+00	=	<b>1.47E+01</b>	6.35E+00	=	1.56E+01	6.28E+00	=	1.63E+01	5.71E+00	=	1.73E+01	4.89E+00	=
6	<b>0.00E+00</b>	0.00E+00	=	1.62E-01	1.99E-01	-	9.08E-03	4.16E-02	-	3.68E-03	2.01E-02	-	<b>0.00E+00</b>	0.00E+00	-
7	2.18E-02	1.05E-02	-	2.39E-02	1.20E-02	-	<b>1.45E-02</b>	1.05E-02	=	2.39E-02	1.15E-02	-	1.67E-02	8.80E-03	-
8	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	1.41E-03	-	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00	=
9	3.20E+00	8.94E-01	=	3.71E+00	1.00E+00	-	3.58E+00	3.04E+00	-	2.99E+00	9.12E-01	=	<b>2.97E+00</b>	6.75E-01	-
10	8.57E-03	2.17E-02	=	6.12E-03	1.88E-02	=	1.32E-01	8.95E-02	-	1.59E+02	3.27E-02	-	<b>4.90E-03</b>	1.70E-02	-
11	9.74E+01	5.78E+01	-	9.01E+01	5.27E+01	-	<b>6.80E+01</b>	4.55E+01	=	1.13E+02	6.40E+01	-	7.25E+01	4.82E+01	-
12	<b>2.56E-01</b>	5.31E-02	=	2.68E-01	4.38E-02	=	2.67E-01	4.45E-02	=	2.60E-01	5.75E-02	=	2.67E-01	4.87E-02	=
13	1.04E-01	3.30E-02	=	1.35E-01	1.91E-02	-	<b>2.33E-02</b>	1.21E-02	+	4.44E-02	3.21E-02	+	1.16E-01	3.65E-02	-
14	1.34E-01	4.50E-02	-	<b>1.09E-01</b>	2.28E-02	=	1.37E-01	5.25E-02	=	1.19E-01	4.89E-02	=	1.15E-01	2.53E-02	=
15	6.52E-01	8.78E-02	=	6.85E-01	1.03E-01	=	7.32E-01	1.91E-01	-	6.91E-01	1.20E-01	-	<b>6.50E-01</b>	1.03E-01	-
16	1.48E+00	2.53E-01	=	1.57E+00	3.34E-01	-	<b>1.36E+00</b>	3.37E-01	=	1.50E+00	2.21E-01	=	1.42E+00	2.62E-01	=
17	9.63E+00	2.84E+01	-	<b>1.12E+00</b>	7.99E-01	+	4.27E+03	1.33E+04	-	2.87E+01	5.08E+01	-	2.83E+00	1.67E+01	-
18	7.14E-01	6.50E-01	=	5.86E-01	6.03E-01	=	2.03E+03	4.71E+03	-	1.59E+00	1.57E+00	-	<b>4.63E-01</b>	5.04E-01	-
19	<b>1.15E-01</b>	4.67E-02	=	1.80E-01	7.16E-02	-	1.83E-01	2.80E-01	=	1.54E-01	8.67E-02	-	1.17E-01	4.39E-02	-
20	1.85E-01	1.85E-01	=	3.11E-01	1.32E-01	-	5.20E+02	1.70E+03	-	7.72E-01	8.35E-01	-	<b>1.59E-01</b>	1.26E-01	-
21	1.46E+00	8.11E+00	=	3.83E-01	4.10E-01	=	3.85E+02	1.62E+03	-	7.84E+00	2.79E+01	-	<b>3.03E-01</b>	2.66E-01	-
22	1.55E-01	1.45E-01	=	1.89E-01	1.02E-01	-	1.45E+00	4.80E+00	=	<b>9.17E-02</b>	1.14E-01	+	1.30E-01	8.77E-02	-
23	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=
24	1.10E+02	1.35E+00	-	1.12E+02	1.46E+00	-	1.13E+02	4.38E+00	-	1.11E+02	2.44E+00	-	<b>1.09E+02</b>	1.37E+00	-
25	1.20E+02	1.66E+01	=	1.20E+02	1.20E+01	-	1.24E+02	1.94E+01	-	<b>1.19E+02</b>	1.23E+01	=	<b>1.19E+02</b>	1.68E+01	-
26	<b>1.00E+02</b>	4.53E-02	+	<b>1.00E+02</b>	3.41E-02	=	<b>1.00E+02</b>	1.36E-02	+	<b>1.00E+02</b>	2.76E-02	+	<b>1.00E+02</b>	3.55E-02	-
27	2.10E+01	5.96E+01	+	<b>6.28E+00</b>	2.77E+01	+	4.64E+01	7.99E+01	-	3.23E+01	7.10E+01	=	2.50E+01	6.26E+01	-
28	<b>2.00E+02</b>	3.25E-12	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=
29	1.85E+02	3.06E+01	=	1.84E+02	3.50E+01	=	1.92E+02	1.48E+01	=	1.79E+02	3.18E+01	=	<b>1.76E+02</b>	3.82E+01	-
30	3.52E+02	1.47E+02	=	3.50E+02	1.31E+02	=	4.12E+02	2.33E+02	=	3.86E+02	1.77E+02	=	<b>3.48E+02</b>	1.37E+02	-
	Number of (+/-/=) 2/6/22			Number of (+/-/=) 2/11/17			Number of (+/-/=) 2/16/12			Number of (+/-/=) 3/14/13					

**TABLE 8.** Comparison results for different powers of the base vector on IEEE CEC 2014 problems for  $D = 10$ .

$f_i$	1			1/2			2			1/3			3		
	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=
1	1.59E-08	5.46E-08	+	6.63E-05	1.79E-04	-	3.33E-03	1.99E-02	-	2.75E-05	8.77E-05	-	0.00E+00	0.00E+00	-
2	0.00E+00	0.00E+00	=	1.02E-04	7.29E-04	-	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=
3	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=
4	1.70E+00	6.85E+00	=	1.70E-01	8.50E-01	+	4.11E+00	1.03E+01	-	2.39E+00	8.26E+00	=	3.75E+00	1.04E+01	=
5	1.55E+01	5.86E+00	=	1.70E+01	5.65E+00	=	1.72E+01	4.89E+00	=	1.63E+01	5.70E+00	=	1.49E+01	6.52E+00	=
6	0.00E+00	0.00E+00	=	2.62E-02	1.21E-01	-	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=	2.40E-05	1.71E-04	=
7	8.93E-03	1.13E-02	=	7.75E-03	1.16E-02	-	2.76E-02	1.77E-02	-	1.61E-02	1.31E-02	-	4.81E-03	8.15E-03	-
8	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=
9	2.79E+00	8.38E-01	=	3.03E+00	9.62E-01	=	3.55E+00	1.40E+00	-	3.50E+00	1.41E+00	=	2.79E+00	1.10E+00	=
10	6.12E-03	1.88E-02	=	2.45E-03	1.22E-02	=	7.35E-03	2.03E-02	=	0.00E+00	0.00E+00	=	2.45E-03	1.22E-02	=
11	6.33E+01	5.01E+01	=	7.19E+01	4.73E+01	-	5.43E+01	6.43E+01	-	5.59E+01	5.73E+01	=	5.42E+01	4.57E+01	=
12	2.45E-01	5.41E-02	=	2.55E-01	4.18E-02	=	2.41E-01	4.63E-02	+	2.50E-01	5.92E-02	=	2.64E-01	4.77E-02	=
13	4.27E-02	2.66E-02	-	1.40E-01	2.97E-02	-	3.01E-02	1.18E-02	+	3.73E-02	1.40E-02	+	6.84E-02	4.05E-02	+
14	9.84E-02	2.61E-02	-	1.01E-01	2.38E-02	+	1.04E-01	4.01E-02	=	1.01E-01	3.66E-02	=	1.12E-01	2.23E-02	=
15	6.62E-01	1.93E-01	=	7.29E-01	1.28E-01	-	6.23E-01	2.12E-01	=	6.53E-01	1.40E-01	=	6.31E-01	1.33E-01	=
16	1.17E+00	4.18E-01	=	1.34E+00	4.00E-01	-	1.32E+00	4.70E-01	=	1.41E+00	4.50E-01	=	1.20E+00	3.93E-01	=
17	7.87E-01	6.76E-01	=	1.88E+00	2.43E+00	-	2.66E+00	3.88E+00	-	1.17E+00	1.76E+00	-	5.37E-01	4.68E-01	-
18	3.74E-01	4.64E-01	=	8.95E-01	7.74E-01	-	4.31E-01	4.75E-01	=	5.21E-01	5.29E-01	=	3.52E-01	4.26E-01	=
19	1.49E-01	6.96E-02	+	1.75E-01	7.68E-02	-	1.29E-01	1.14E-01	=	1.78E-01	2.07E-01	=	1.10E-01	4.59E-02	=
20	2.42E-01	1.30E-01	=	3.90E-01	2.08E-01	-	2.13E-01	1.65E-01	=	2.12E-01	1.55E-01	=	1.91E-01	1.10E-01	=
21	3.83E-01	4.03E-01	=	5.20E-01	3.91E-01	=	2.66E-01	2.43E-01	=	2.67E-01	2.59E-01	=	3.63E-01	2.78E-01	=
22	1.17E-01	5.00E-02	=	1.44E-01	5.95E-02	-	8.95E-02	8.71E-02	+	1.18E-01	3.73E-02	=	1.33E-01	9.73E-02	=
23	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00	=
24	1.09E+02	2.12E+00	=	1.10E+02	1.49E+00	-	1.09E+02	2.98E+00	=	1.10E+02	2.81E+00	=	1.09E+02	1.76E+00	=
25	1.16E+02	1.39E+01	=	1.14E+02	4.76E+00	=	1.30E+02	2.12E+01	-	1.21E+02	1.93E+01	-	1.16E+02	1.53E+01	-
26	1.00E+02	1.65E-02	-	1.00E+02	5.93E-02	-	1.00E+02	1.53E-02	+	1.00E+02	1.40E-02	+	1.00E+02	2.94E-02	+
27	5.69E+00	2.78E+01	=	2.47E+00	3.84E-01	-	1.52E+01	5.63E+01	=	1.88E+00	4.30E-01	=	1.78E+00	4.57E-01	=
28	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00	=	2.00E+02	4.49E-03	=
29	1.99E+02	5.58E+00	+	1.94E+02	2.10E+01	-	1.98E+02	8.64E+00	-	2.00E+02	0.00E+00	-	1.80E+02	3.48E+01	-
30	3.50E+02	1.32E+02	=	3.61E+02	1.31E+02	-	3.44E+02	1.37E+02	-	3.52E+02	1.32E+02	-	3.29E+02	1.32E+02	-
	Number of (+/-/=) 3/3/24			Number of (+/-/=) 2/18/10			Number of (+/-/=) 4/9/17			Number of (+/-/=) 2/8/20					

TABLE 9. Comparison results for different powers of the difference vector on IEEE CEC 2014 problems for  $D = 10$ .

$f_i$	1			1/2			2			1/3			3		
	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=
1	6.04E-07	3.75E-06	-	3.77E-04	1.02E-03	-	3.44E-06	2.25E-05	-	7.01E-06	3.72E-05	-	2.37E-09	1.69E-08	-
2	0.00E+00	0.00E+00	=	1.38E-08	3.47E-08	-	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=
3	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=
4	1.45E+00	6.83E+00	=	6.82E-01	4.87E+00	=	4.09E+00	1.03E+01	-	3.40E-01	1.18E+00	=	1.62E+00	6.84E+00	=
5	1.46E+01	6.74E+00	=	1.60E+01	6.00E+00	=	1.47E+01	6.82E+00	+	1.58E+01	5.93E+00	=	1.71E+01	5.71E+00	=
6	5.31E-02	1.39E-01	-	4.46E-01	2.20E-01	-	0.00E+00	0.00E+00	=	1.13E-01	1.86E-01	-	0.00E+00	0.00E+00	=
7	1.86E-02	1.04E-02	=	3.29E-02	1.45E-02	-	3.75E-02	2.61E-02	-	3.10E-02	1.74E-02	-	1.85E-02	1.59E-02	-
8	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	=
9	3.17E+00	9.39E-01	=	3.44E+00	9.27E-01	-	3.02E+00	1.15E+00	=	3.45E+00	9.59E-01	-	3.11E+00	7.76E-01	-
10	1.22E-03	8.75E-03	=	0.00E+00	0.00E+00	=	6.12E-03	1.88E-02	=	0.00E+00	0.00E+00	=	3.67E-03	1.48E-02	=
11	9.24E+01	5.52E+01	-	8.63E+01	5.32E+01	=	8.12E+01	8.13E+01	=	7.37E+01	4.30E+01	=	7.09E+01	4.17E+01	=
12	2.66E-01	5.86E-02	-	2.51E-01	5.52E-02	=	2.43E-01	4.44E-02	=	2.43E-01	4.65E-02	=	2.42E-01	4.56E-02	=
13	1.27E-01	4.35E-02	=	1.40E-01	2.48E-02	-	4.91E-02	2.79E-02	+	1.14E-01	4.12E-02	=	1.22E-01	4.21E-02	=
14	1.07E-01	2.21E-02	=	1.05E-01	2.10E-02	=	1.00E-01	2.70E-02	=	1.08E-01	2.53E-02	=	1.04E-01	3.27E-02	=
15	6.96E-01	1.15E-01	-	7.13E-01	1.25E-01	-	6.85E-01	1.90E-01	=	7.31E-01	1.39E-01	-	6.31E-01	1.20E-01	-
16	1.46E+00	3.13E-01	=	1.61E+00	3.16E-01	-	1.41E+00	3.51E-01	=	1.54E+00	3.18E-01	-	1.42E+00	3.38E-01	-
17	1.16E+00	8.20E-01	-	1.82E+00	1.91E+00	-	1.14E+00	8.77E-01	-	1.42E+00	2.02E+00	-	5.87E-01	5.06E-01	-
18	5.42E-01	5.54E-01	=	7.28E-01	6.49E-01	=	6.60E-01	5.77E-01	=	6.82E-01	5.63E-01	=	4.91E-01	4.69E-01	=
19	1.45E-01	5.40E-02	=	1.91E-01	7.52E-02	-	1.43E-01	1.01E-01	=	1.98E-01	8.23E-02	-	1.47E-01	8.69E-02	-
20	2.87E-01	1.51E-01	-	4.40E-01	2.16E-01	-	2.12E-01	1.84E-01	=	2.34E-01	1.55E-01	=	2.22E-01	1.09E-01	=
21	2.94E-01	3.01E-01	=	7.95E-01	9.44E-01	-	2.80E-01	2.42E-01	=	2.96E-01	2.47E-01	=	2.66E-01	2.52E-01	=
22	1.70E-01	7.44E-02	=	1.60E-01	5.35E-02	=	1.32E-01	7.86E-02	=	1.61E-01	4.93E-02	-	1.51E-01	7.74E-02	-
23	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00	=
24	1.11E+02	1.66E+00	-	1.11E+02	1.36E+00	-	1.12E+02	3.33E+00	-	1.11E+02	1.85E+00	-	1.10E+02	1.72E+00	-
25	1.19E+02	1.23E+01	=	1.18E+02	3.48E+00	+	1.21E+02	1.71E+01	=	1.19E+02	1.31E+01	=	1.21E+02	2.03E+01	=
26	1.00E+02	4.65E-02	+	1.00E+02	4.87E-02	=	1.00E+02	1.60E-02	+	1.00E+02	4.23E-02	+	1.00E+02	6.70E-02	+
27	2.31E+00	4.50E-01	=	2.73E+00	4.17E-01	-	5.50E+00	2.78E+01	-	2.30E+00	4.51E-01	=	2.23E+00	4.70E-01	=
28	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00	=	2.00E+02	0.00E+00	=
29	1.99E+02	1.01E+01	-	1.88E+02	3.04E+01	=	1.96E+02	1.62E+01	-	1.94E+02	2.13E+01	=	1.89E+02	2.67E+01	=
30	3.71E+02	1.25E+02	=	3.46E+02	1.31E+02	=	3.18E+02	1.32E+02	=	3.47E+02	1.31E+02	=	3.60E+02	1.30E+02	=
	Number of (+/-/=) 1/9/20			Number of (+/-/=) 1/14/15			Number of (+/-/=) 3/7/20			Number of (+/-/=) 1/10/19					



**TABLE 10. Comparisons with  $F$  in literatures on IEEE CEC 2014 problems for  $D = 30$ .**

(a) 0.1, 0.9, rand and SinDE

$f_i$	0.1			0.9			rand			SinDE		
	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=
1	1.80E+07	8.09E+06	-	1.72E+06	9.98E+05	-	1.45E+05	8.92E+04	-	<b>5.73E+03</b>	5.02E+03	+
2	3.03E+05	4.98E+05	-	1.06E+03	1.34E+03	-	1.49E+02	3.13E+02	-	2.34E-04	9.31E-04	-
3	5.87E+02	5.00E+02	-	1.40E+01	1.34E+01	-	1.74E-01	3.87E-01	-	1.08E-06	3.58E-06	-
4	1.34E+02	2.28E+01	-	8.37E+01	1.14E+01	-	1.00E+02	1.46E+01	-	1.84E+01	2.70E+01	+
5	2.03E+01	9.90E-02	=	2.03E+01	3.56E-02	-	2.03E+01	3.61E-02	=	2.03E+01	8.88E-02	=
6	6.65E+00	2.06E+00	-	1.24E+01	1.01E+00	-	1.20E+01	1.55E+00	-	1.37E+00	3.29E+00	-
7	5.68E-02	7.54E-02	-	3.94E-03	5.28E-03	-	4.37E-04	6.41E-04	-	<b>0.00E+00</b>	0.00E+00	=
8	8.22E-05	1.30E-04	-	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00	=
9	2.82E+01	6.88E+00	-	4.62E+01	4.09E+00	-	3.87E+01	4.30E+00	-	3.11E+01	3.29E+00	-
10	2.18E-02	2.37E-02	-	2.04E-03	6.25E-03	=	8.16E-04	4.08E-03	=	4.86E-03	8.35E-03	=
11	1.82E+03	5.70E+02	=	2.06E+03	2.15E+02	-	2.01E+03	1.87E+02	-	1.88E+03	2.45E+02	-
12	3.69E-01	4.89E-02	=	3.97E-01	4.83E-02	-	3.85E-01	5.02E-02	-	3.74E-01	4.91E-02	=
13	<b>1.14E-01</b>	2.74E-02	+	3.70E-01	3.47E-02	-	3.46E-01	3.83E-02	-	2.96E-01	3.42E-02	-
14	2.59E-01	3.18E-02	-	2.48E-01	2.48E-02	-	2.26E-01	2.10E-02	=	<b>1.91E-01</b>	2.34E-02	+
15	4.12E+00	8.08E-01	+	8.07E+00	8.11E-01	-	6.96E+00	6.50E-01	-	5.28E+00	1.02E+00	+
16	9.76E+00	7.03E-01	=	9.34E+00	4.00E-01	=	9.29E+00	6.56E-01	=	<b>9.20E+00</b>	8.16E-01	=
17	9.61E+05	6.45E+05	-	1.29E+03	6.69E+02	-	9.61E+02	5.29E+02	-	3.33E+02	2.48E+02	-
18	1.22E+03	2.30E+03	-	2.17E+01	1.68E+01	-	1.60E+01	1.34E+01	-	1.20E+01	8.03E+00	-
19	8.26E+00	3.99E+00	-	7.52E+00	4.23E-01	-	4.44E+00	8.54E-01	-	3.36E+00	6.47E-01	-
20	4.02E+03	3.49E+03	-	2.15E+01	1.05E+01	-	1.35E+01	5.21E+00	-	1.02E+01	4.50E+00	-
21	1.07E+05	9.52E+04	-	9.00E+02	5.44E+02	-	5.47E+02	3.96E+02	-	1.27E+02	1.28E+02	-
22	4.60E+01	3.95E+01	-	1.29E+02	4.52E+01	-	1.03E+02	6.05E+01	-	5.26E+01	2.60E+01	-
23	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=
24	<b>2.00E+02</b>	3.44E-12	-	<b>2.00E+02</b>	7.64E-13	=	<b>2.00E+02</b>	2.49E-12	=	<b>2.00E+02</b>	0.00E+00	=
25	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=
26	<b>1.00E+02</b>	2.94E-02	+	<b>1.00E+02</b>	4.09E-02	-	<b>1.00E+02</b>	3.69E-02	-	<b>1.00E+02</b>	2.84E-02	-
27	2.39E+02	9.22E+01	-	2.18E+02	6.10E+01	=	2.08E+02	3.94E+01	=	2.16E+02	5.43E+01	=
28	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=
29	2.00E+02	0.00E+00	-	2.00E+02	0.00E+00	-	2.00E+02	0.00E+00	-	1.80E+02	2.19E+01	+
30	<b>2.00E+02</b>	5.71E-09	=	<b>2.00E+02</b>	1.14E-04	=	2.06E+02	4.05E+01	=	2.05E+02	3.28E+01	=
	Number of (+/-/=) 3/19/8			Number of (+/-/=) 0/21/9			Number of (+/-/=) 0/19/11			Number of (+/-/=) 5/13/12		

(b) SHADE, FiADE, Rbs and (19)

$f_i$	SHADE			FiADE			Rbs			(19)	
	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std
1	1.02E+04	1.01E+04	+	7.06E+05	3.93E+05	-	1.88E+06	1.34E+06	-	1.75E+04	1.83E+04
2	<b>3.53E-09</b>	1.71E-08	+	3.01E+02	3.50E+02	-	1.83E+04	5.27E+04	-	5.35E-05	3.77E-04
3	<b>0.00E+00</b>	0.00E+00	+	5.76E+00	7.33E+00	-	7.36E+01	1.39E+02	-	2.74E-07	9.31E-07
4	<b>6.79E+00</b>	1.80E+01	+	9.04E+01	1.43E+01	-	1.00E+02	2.91E+01	-	6.09E+01	2.26E+01
5	2.03E+01	2.65E-02	-	2.03E+01	8.50E-02	-	<b>2.02E+01</b>	1.01E-01	=	2.03E+01	3.35E-02
6	1.07E+00	2.93E+00	-	1.21E+01	1.06E+00	-	8.47E+00	3.01E+00	-	<b>1.86E-02</b>	1.25E-01
7	<b>0.00E+00</b>	0.00E+00	=	3.26E-04	4.98E-04	-	1.86E-02	3.50E-02	-	<b>0.00E+00</b>	0.00E+00
8	<b>0.00E+00</b>	0.00E+00	=	1.48E-06	3.35E-06	-	1.69E+01	4.18E+00	-	<b>0.00E+00</b>	0.00E+00
9	3.21E+01	8.12E+00	-	4.34E+01	5.50E+00	-	4.43E+01	1.28E+01	-	<b>2.31E+01</b>	2.56E+00
10	8.16E-04	4.08E-03	=	1.22E-03	4.95E-03	-	2.76E+02	2.08E+02	-	<b>4.09E-04</b>	2.92E-03
11	1.95E+03	3.44E+02	-	2.08E+03	1.58E+02	-	2.66E+03	7.17E+02	-	<b>1.74E+03</b>	1.83E+02
12	3.47E-01	9.14E-02	=	3.93E-01	4.62E-02	-	<b>1.18E-01</b>	7.01E-02	+	3.59E-01	5.08E-02
13	1.29E-01	3.13E-02	+	3.50E-01	3.79E-02	-	1.77E-01	4.28E-02	+	2.32E-01	5.47E-02
14	2.14E-01	2.60E-02	+	2.39E-01	2.77E-02	-	2.44E-01	3.41E-02	-	2.26E-01	3.04E-02
15	<b>3.14E+00</b>	6.32E-01	+	7.57E+00	8.51E-01	-	4.47E+00	1.15E+00	=	5.34E+00	2.14E+00
16	9.53E+00	7.59E-01	=	9.49E+00	6.54E-01	=	1.09E+01	7.19E-01	=	1.01E+01	1.66E+00
17	1.24E+02	9.93E+01	=	1.21E+03	6.97E+02	-	9.33E+03	1.99E+04	-	<b>9.39E+01</b>	3.69E+01
18	9.84E+00	3.40E+00	-	2.05E+01	1.71E+01	-	1.25E+02	1.62E+02	-	<b>6.81E+00</b>	2.52E+00
19	3.04E+00	5.22E-01	-	7.09E+00	6.16E-01	-	5.72E+00	1.05E+00	-	<b>2.86E+00</b>	5.49E-01
20	6.26E+00	1.93E+00	=	2.33E+01	1.38E+01	-	1.16E+02	1.80E+02	-	<b>5.61E+00</b>	1.91E+00
21	3.78E+01	4.82E+01	=	1.02E+03	4.05E+02	-	6.46E+02	3.58E+02	-	<b>3.49E+01</b>	2.79E+01
22	2.95E+01	2.39E+01	=	1.17E+02	5.48E+01	-	1.94E+02	1.22E+02	-	<b>2.28E+01</b>	3.61E+00
23	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00
24	<b>2.00E+02</b>	1.04E-12	-	<b>2.00E+02</b>	2.11E-11	=	<b>2.00E+02</b>	6.77E-13	-	<b>2.00E+02</b>	0.00E+00
25	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00
26	<b>1.00E+02</b>	2.65E-02	+	<b>1.00E+02</b>	3.29E-02	-	<b>1.00E+02</b>	4.09E-02	+	<b>1.00E+02</b>	5.11E-02
27	<b>2.01E+02</b>	4.07E+00	=	2.23E+02	7.09E+01	=	2.19E+02	6.06E+01	=	2.08E+02	3.93E+01
28	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00
29	1.96E+02	9.55E+00	-	2.00E+02	0.00E+00	-	2.00E+02	0.00E+00	-	<b>1.88E+02</b>	1.75E+01
30	<b>2.00E+02</b>	4.52E-12	=	<b>2.00E+02</b>	2.80E-04	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	2.74E-08
	Number of (+/-/=) 8/8/14			Number of (+/-/=) 0/23/7			Number of (+/-/=) 3/19/8				

TABLE 11. Comparisons with  $F$  in literatures on IEEE CEC 2014 problems for  $D = 50$ .

(a) 0.1, 0.9, rand and SinDE

$f_i$	0.1			0.9			rand			SinDE		
	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=
1	2.60E+07	1.11E+07	-	2.47E+07	6.69E+07	-	2.55E+06	1.04E+06	-	6.02E+05	2.15E+05	-
2	6.50E+04	1.78E+05	-	4.35E+05	2.86E+05	-	5.98E+04	4.42E+04	-	3.77E+03	2.38E+03	=
3	1.34E+04	4.69E+03	-	3.54E+03	1.47E+03	-	2.43E+02	1.68E+02	-	2.20E+00	2.01E+00	-
4	1.48E+02	1.85E+01	-	9.98E+01	6.88E-01	-	9.89E+01	5.42E-01	-	9.61E+01	1.47E+00	-
5	2.05E+01	1.04E-01	-	2.05E+01	2.06E-01	-	2.05E+01	3.21E-02	-	2.05E+01	1.35E-01	-
6	1.49E+01	3.66E+00	-	2.79E+01	1.52E+00	-	2.64E+01	1.17E+00	-	8.14E+00	9.63E+00	-
7	8.16E-02	8.68E-02	-	7.69E-02	3.07E-02	-	2.89E-02	1.10E-02	-	2.60E-04	4.30E-04	-
8	2.61E-04	3.00E-04	+	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00	=
9	<b>5.61E+01</b>	1.33E+01	=	1.25E+02	7.54E+00	-	1.09E+02	1.05E+01	-	8.72E+01	7.71E+00	-
10	2.00E-02	1.30E-02	-	1.93E-02	1.53E-02	-	1.04E-02	1.05E-02	-	1.19E-02	1.16E-02	=
11	4.45E+03	8.74E+02	=	5.36E+03	3.35E+02	-	5.53E+03	2.04E+03	-	4.90E+03	3.17E+02	-
12	4.57E-01	5.94E-02	-	5.11E-01	5.02E-02	-	5.49E-01	3.69E-01	-	5.76E-01	5.25E-01	-
13	<b>2.05E-01</b>	4.92E-02	+	4.95E-01	3.54E-02	-	4.86E-01	3.21E-02	-	4.36E-01	3.83E-02	-
14	3.20E-01	2.23E-02	-	2.84E-01	2.00E-02	-	2.65E-01	2.00E-02	-	2.44E-01	3.72E-02	-
15	9.13E+00	1.47E+00	-	2.15E+01	1.49E+00	-	1.90E+01	1.50E+00	-	1.34E+01	3.32E+00	-
16	1.91E+01	8.83E-01	=	1.81E+01	3.65E-01	+	1.82E+01	1.07E+00	+	1.84E+01	1.66E+00	+
17	2.06E+06	1.37E+06	-	3.58E+05	7.65E+05	-	3.58E+03	2.31E+03	-	1.12E+03	4.66E+02	-
18	4.89E+02	3.25E+02	-	6.31E+04	5.25E+04	-	5.80E+01	1.86E+01	-	<b>2.30E+01</b>	9.00E+00	+
19	4.85E+01	8.24E+00	-	2.79E+01	4.58E+00	-	1.48E+01	2.68E+00	-	<b>9.27E+00</b>	6.94E-01	=
20	1.08E+04	5.44E+03	-	1.23E+03	2.50E+03	-	6.21E+01	2.12E+01	-	2.74E+01	7.58E+00	-
21	1.09E+06	5.83E+05	-	1.65E+04	1.35E+04	-	1.19E+03	4.98E+02	-	8.62E+02	3.74E+02	-
22	<b>9.69E+01</b>	7.07E+01	=	6.86E+02	1.30E+02	-	5.91E+02	1.48E+02	-	5.35E+02	1.59E+02	-
23	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=
24	<b>2.00E+02</b>	8.34E-12	=	<b>2.00E+02</b>	2.35E-11	=	<b>2.00E+02</b>	1.87E-11	=	<b>2.00E+02</b>	3.34E-11	=
25	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=
26	<b>1.00E+02</b>	5.51E-02	+	<b>1.00E+02</b>	3.06E-02	-	<b>1.00E+02</b>	4.12E-02	-	<b>1.00E+02</b>	3.45E-02	-
27	2.08E+02	5.70E+01	+	2.16E+02	1.12E+02	-	<b>2.00E+02</b>	8.04E-02	+	2.10E+02	4.10E+01	-
28	<b>2.00E+02</b>	0.00E+00	-	<b>2.00E+02</b>	0.00E+00	-	<b>2.00E+02</b>	0.00E+00	-	<b>2.00E+02</b>	7.64E-13	-
29	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=
30	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=
Number of (+/-/=) 4/1/7/9			Number of (+/-/=) 1/2/3/6			Number of (+/-/=) 2/2/2/6			Number of (+/-/=) 2/19/9			

(b) SHADE, FiADE, Rbs and (19)

$f_i$	SHADE			FiADE			Rbs			(19)	
	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std
1	4.55E+05	2.02E+05	-	7.59E+06	3.19E+06	-	3.66E+06	1.32E+06	-	<b>2.45E+05</b>	1.11E+05
2	4.29E+03	2.26E+03	=	2.05E+05	1.35E+05	-	6.12E+04	1.33E+05	-	<b>4.03E+03</b>	3.57E+03
3	2.94E+00	4.21E+00	-	1.27E+03	6.02E+02	-	3.69E+03	1.94E+03	-	<b>2.08E-02</b>	3.06E-02
4	9.56E+01	2.06E+00	-	9.92E+01	5.80E-01	-	1.13E+02	2.19E+01	-	<b>9.39E+01</b>	4.25E+00
5	2.04E+01	1.97E-01	-	2.05E+01	6.42E-02	-	2.00E+01	7.70E-02	+	<b>2.02E+01</b>	2.10E-01
6	2.22E+00	4.08E+00	-	2.64E+01	1.89E+00	-	1.98E+01	4.56E+00	-	<b>1.37E-01</b>	3.76E-01
7	1.26E-08	3.76E-08	-	2.25E-02	1.22E-02	-	6.31E-02	5.50E-02	-	<b>0.00E+00</b>	0.00E+00
8	<b>0.00E+00</b>	0.00E+00	=	2.40E-08	3.71E-08	+	3.94E+01	8.53E+00	-	1.95E-02	1.39E-01
9	5.13E+01	1.35E+01	=	1.19E+02	7.70E+00	-	8.78E+01	1.83E+01	-	5.62E+01	1.36E+01
10	<b>5.65E-03</b>	8.77E-03	=	1.50E-02	1.47E-02	-	8.88E+02	3.45E+02	-	9.42E-03	1.83E-02
11	4.40E+03	8.08E+02	=	5.34E+03	2.76E+02	-	5.30E+03	7.19E+02	-	<b>4.33E+03</b>	9.39E+02
12	3.45E-01	2.07E-01	-	5.14E-01	5.59E-02	-	1.74E-01	7.69E-02	-	<b>1.09E-01</b>	5.97E-02
13	2.13E-01	3.59E-02	+	4.72E-01	3.02E-02	-	2.75E-01	4.38E-02	-	2.47E-01	4.00E-02
14	2.48E-01	2.70E-02	-	2.69E-01	2.39E-02	-	3.06E-01	3.50E-02	-	<b>2.09E-01</b>	2.62E-02
15	<b>5.71E+00</b>	1.11E+00	=	1.99E+01	1.36E+00	-	1.04E+01	2.54E+00	-	6.00E+00	9.87E-01
16	1.86E+01	8.70E-01	=	<b>1.80E+01</b>	3.13E-01	+	2.02E+01	6.35E-01	-	1.88E+01	8.98E-01
17	9.20E+02	2.65E+02	-	1.78E+05	6.03E+05	-	5.85E+04	5.05E+04	-	<b>7.79E+02</b>	3.25E+02
18	3.81E+01	1.21E+01	-	4.21E+03	2.82E+04	-	4.13E+02	2.65E+02	-	2.94E+01	9.62E+00
19	9.95E+00	6.49E-01	-	2.57E+01	4.03E+00	-	2.57E+01	1.07E+01	-	9.38E+00	6.75E-01
20	2.46E+01	6.94E+00	-	1.58E+02	4.89E+01	-	7.60E+02	5.55E+02	-	<b>1.80E+01</b>	5.26E+00
21	4.84E+02	1.66E+02	=	2.63E+03	1.12E+03	-	1.16E+04	1.29E+04	-	<b>4.25E+02</b>	1.92E+02
22	2.09E+02	1.31E+02	-	6.51E+02	1.60E+02	-	6.60E+02	2.44E+02	-	1.48E+02	1.23E+02
23	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00
24	<b>2.00E+02</b>	6.52E-11	=	<b>2.00E+02</b>	3.89E-11	=	<b>2.00E+02</b>	1.44E-11	=	<b>2.00E+02</b>	1.61E-11
25	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00
26	<b>1.00E+02</b>	3.29E-02	+	<b>1.00E+02</b>	3.26E-02	-	<b>1.00E+02</b>	3.70E-02	=	<b>1.00E+02</b>	3.57E-02
27	<b>2.00E+02</b>	3.06E+00	+	2.53E+02	1.89E+02	-	2.51E+02	1.56E+02	-	2.08E+02	3.31E+01
28	<b>2.00E+02</b>	0.00E+00	-	<b>2.00E+02</b>	0.00E+00	-	<b>2.00E+02</b>	0.00E+00	-	<b>2.00E+02</b>	5.74E-14
29	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00
30	<b>2.00E+02</b>	5.71E-09	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00	=	<b>2.00E+02</b>	0.00E+00
Number of (+/-/=) 3/14/13			Number of (+/-/=) 2/2/3/5			Number of (+/-/=) 1/2/3/6					

**TABLE 12. Comparisons with  $F$  in literatures on IEEE CEC 2017 problems for  $D = 10$ .**

(a) 0.1, 0.9, rand and SinDE

$f_i$	0.1			0.9			rand			SinDE		
	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=
1	5.84E+05	2.67E+06	-	1.71E+02	6.41E+02	-	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00	=
3	2.37E+03	1.41E+03	-	1.15E+02	5.93E+02	-	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00	=
4	6.13E+00	1.42E+00	-	1.22E-01	6.10E-01	+	<b>0.00E+00</b>	0.00E+00	+	1.95E-02	4.27E-02	+
5	4.75E+00	1.72E+00	-	3.57E+00	8.60E-01	-	3.23E+00	1.06E+00	-	2.66E+00	8.90E-01	-
6	4.49E-05	2.29E-04	-	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00	=
7	1.51E+01	1.94E+00	-	1.50E+01	1.28E+00	-	1.45E+01	1.22E+00	-	1.32E+01	7.85E-01	-
8	5.14E+00	1.70E+00	-	4.38E+00	1.04E+00	-	3.68E+00	9.21E-01	-	2.97E+00	7.53E-01	-
9	1.09E-01	2.56E-01	-	1.50E-07	1.06E-06	=	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00	=
10	6.33E+01	7.57E+01	-	9.36E+01	5.98E+01	-	6.98E+01	6.15E+01	-	5.60E+01	4.55E+01	-
11	4.86E+00	1.94E+00	-	2.38E+00	9.71E-01	-	2.12E+00	9.05E-01	-	1.27E+00	9.46E-01	=
12	1.57E+05	3.00E+05	-	1.10E+01	3.01E+01	-	9.66E-01	2.62E+00	+	1.21E+00	2.97E+00	+
13	2.48E+03	3.88E+03	-	6.28E+00	4.38E+00	-	3.89E+00	2.52E+00	-	1.61E+00	1.95E+00	=
14	1.32E+01	1.25E+01	-	7.42E-01	7.21E-01	-	7.46E-01	5.57E-01	-	4.84E-01	5.80E-01	=
15	7.17E+00	5.43E+00	-	8.53E-01	7.57E-01	-	4.41E-01	4.86E-01	-	1.80E-01	2.25E-01	=
16	2.55E+00	1.75E+00	-	6.73E-01	3.28E-01	=	4.78E-01	2.63E-01	=	4.75E-01	2.88E-01	=
17	1.87E+00	3.91E+00	-	3.61E-01	2.99E-01	-	2.89E-01	2.71E-01	-	2.26E-01	2.10E-01	-
18	1.01E+03	1.27E+03	-	3.10E-01	2.68E-01	-	1.10E-01	1.81E-01	=	9.94E-02	1.68E-01	=
19	4.63E+01	1.83E+02	-	4.11E-02	1.86E-02	-	2.65E-02	9.89E-03	-	4.91E-03	7.91E-03	+
20	4.05E+01	4.95E-01	-	0.00E+00	0.00E+00	=	<b>0.00E+00</b>	0.00E+00	=	4.28E-02	1.08E-01	=
21	1.48E+02	4.12E+01	-	1.26E+02	3.71E+01	-	1.29E+02	3.92E+01	-	1.33E+02	4.64E+01	=
22	1.01E+02	7.06E-01	-	7.28E+01	4.52E+01	=	<b>8.85E+01</b>	3.26E+01	=	9.65E+01	1.80E+01	=
23	3.06E+02	8.49E+00	-	3.06E+02	1.34E+00	-	3.05E+02	1.40E+00	-	3.04E+02	1.10E+00	=
24	2.68E+02	6.29E+01	=	<b>2.25E+02</b>	1.01E+02	=	2.36E+02	1.02E+02	=	2.33E+02	9.29E+01	=
25	4.31E+02	1.75E+01	-	3.98E+02	3.18E-02	-	3.98E+02	1.07E-01	=	3.99E+02	6.36E+00	=
26	3.38E+02	5.53E+01	-	3.00E+02	2.69E-12	-	3.00E+02	2.56E-13	=	3.00E+02	6.43E-14	+
27	3.95E+02	1.77E+00	-	<b>3.89E+02</b>	6.83E-01	+	<b>3.89E+02</b>	9.53E-01	+	<b>3.89E+02</b>	4.95E-01	+
28	4.08E+02	9.27E+01	-	<b>3.00E+02</b>	4.10E-13	-	<b>3.00E+02</b>	3.01E-13	=	<b>3.00E+02</b>	9.09E-14	+
29	2.47E+02	8.08E+00	-	2.45E+02	1.32E+01	-	2.35E+02	5.20E+00	-	<b>2.30E+02</b>	2.98E+00	+
30	1.77E+04	1.45E+04	-	4.22E+02	2.39E+01	-	3.95E+02	1.81E+00	+	3.95E+02	2.50E+00	+
Number of (+/-/=) 0/28/1			Number of (+/-/=) 2/21/6			Number of (+/-/=) 4/13/12			Number of (+/-/=) 8/5/16			

(b) SHADE, FiADE, Rbs, EsDE<sub>r</sub>-NR and (19)

$f_i$	SHADE			FiADE			Rbs			EsDE <sub>r</sub> -NR			(19)	
	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std
1	<b>0.00E+00</b>	0.00E+00	=	9.42E-04	6.09E-03	-	5.04E+02	1.27E+03	-	<b>0.00E+00</b>	0.00E+00	=	2.61E-09	1.36E-08
3	<b>0.00E+00</b>	0.00E+00	=	1.15E-05	9.44E-06	-	1.07E+01	3.16E+01	-	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00
4	<b>0.00E+00</b>	0.00E+00	+	7.70E-06	5.30E-06	+	4.87E+00	1.07E+00	-	<b>0.00E+00</b>	0.00E+00	+	3.50E+00	7.90E-01
5	4.10E+00	1.68E+00	-	3.69E+00	8.54E-01	-	7.82E+00	3.44E+00	-	2.71E+00	7.61E-01	-	<b>2.22E+00</b>	7.35E-01
6	<b>0.00E+00</b>	0.00E+00	=	6.65E-06	5.00E-06	-	1.15E-04	3.41E-04	-	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00
7	1.45E+01	1.96E+00	-	1.51E+01	1.17E+00	-	1.96E+01	4.48E+00	-	1.34E+01	9.12E-01	-	<b>1.25E+01</b>	7.46E-01
8	4.17E+00	1.67E+00	-	4.27E+00	9.32E-01	-	9.67E+00	3.84E+00	-	3.08E+00	9.07E-01	-	<b>2.19E+00</b>	7.95E-01
9	<b>0.00E+00</b>	0.00E+00	=	3.53E-06	2.38E-06	-	1.42E-02	6.64E-02	-	<b>0.00E+00</b>	0.00E+00	=	<b>0.00E+00</b>	0.00E+00
10	5.68E+01	5.27E+01	-	9.33E+01	6.65E+01	-	4.13E+02	2.20E+02	-	3.66E+01	3.36E+01	=	<b>2.95E+01</b>	4.09E+01
11	<b>3.46E-01</b>	6.55E-01	+	2.52E+00	8.21E-01	-	4.33E+00	1.78E+00	-	7.20E-01	9.31E-01	+	1.26E+00	7.11E-01
12	<b>3.47E-01</b>	1.75E-01	+	5.16E+00	1.71E+01	-	1.63E+03	2.86E+03	-	9.76E-01	2.63E+00	+	3.18E+00	1.69E+01
13	<b>1.45E+00</b>	1.83E+00	=	5.31E+00	3.55E+00	-	7.98E+00	4.68E+00	-	2.00E+00	2.06E+00	=	2.10E+00	2.37E+00
14	1.33E+00	9.91E-01	-	9.92E-01	5.50E-01	-	5.80E+00	6.56E+00	-	8.99E-01	8.95E-01	-	<b>4.71E-01</b>	6.07E-01
15	<b>9.77E-02</b>	1.66E-01	=	7.55E-01	5.14E-01	-	2.01E+00	1.14E+00	-	1.99E-01	3.22E-01	=	1.31E-01	1.88E-01
16	3.92E-01	2.70E-01	+	5.77E-01	2.71E-01	=	2.34E+00	3.30E+00	-	<b>3.81E-01</b>	2.43E-01	+	5.79E-01	2.92E-01
17	2.30E-01	2.30E-01	-	4.13E-01	3.63E-01	-	6.63E+00	7.75E+00	-	1.54E-01	1.40E-01	-	<b>1.04E-01</b>	1.27E-01
18	2.17E-01	3.32E-01	=	2.24E-01	2.54E-01	=	2.34E+01	1.36E+02	-	<b>9.70E-02</b>	1.64E-01	=	1.41E-01	1.69E-01
19	2.15E-02	1.63E-02	=	3.51E-02	1.63E-02	-	8.37E-01	9.12E-01	-	<b>1.59E-02</b>	9.68E-03	+	1.68E-02	9.37E-03
20	<b>0.00E+00</b>	0.00E+00	=	4.61E-06	2.19E-06	-	2.14E+00	4.40E+00	-	3.06E-02	9.38E-02	=	1.84E-02	7.42E-02
21	1.27E+02	3.99E+01	=	1.29E+02	3.60E+01	-	1.32E+02	4.48E+01	-	1.18E+02	3.20E+01	=	<b>1.16E+02</b>	3.49E+01
22	9.23E+01	2.72E+01	=	8.86E+01	3.27E+01	=	1.01E+02	8.43E+00	-	9.51E+01	2.14E+01	-	1.00E+02	1.75E-01
23	3.04E+02	1.24E+01	-	3.05E+02	1.35E+00	-	3.04E+02	4.33E+01	-	3.04E+02	1.63E+00	=	<b>3.03E+02</b>	2.14E+00
24	2.41E+02	9.76E+01	=	2.46E+02	9.48E+01	=	2.45E+02	9.75E+01	=	2.59E+02	9.53E+01	=	2.39E+02	9.82E+01
25	<b>3.98E+02</b>	1.13E-01	=	<b>3.98E+02</b>	3.29E-02	-	4.13E+02	2.17E+01	-	<b>3.98E+02</b>	1.36E-01	=	4.03E+02	1.48E+01
26	<b>2.94E+02</b>	4.20E+01	+	3.00E+02	7.64E-06	-	3.00E+02	1.91E+00	-	3.00E+02	1.37E-13	+	3.00E+02	2.88E-13
27	<b>3.89E+02</b>	7.60E-01	+	<b>3.89E+02</b>	4.72E-01	+	3.93E+02	1.97E+00	-	<b>3.89E+02</b>	5.72E-01	+	3.92E+02	2.10E+00
28	<b>3.00E+02</b>	1.85E-13	+	<b>3.00E+02</b>	9.05E-06	-	3.09E+02	5.25E+01	-	3.06E+02	3.97E+01	+	3.06E+02	4.37E+01
29	2.32E+02	2.90E+00	=	2.42E+02	1.15E+01	-	2.53E+02	1.26E+01	-	2.31E+02	3.01E+00	+	2.32E+02	2.90E+00
30	3.96E+02	3.83E+00	+	3.98E+02	4.65E+00	+	<b>3.57E+03</b>	4.00E+03	-	3.95E+02	1.84E+00	+	4.13E+02	2.29E+01
Number of (+/-/=) 8/7/14			Number of (+/-/=) 3/21/5			Number of (+/-/=) 0/28/1			Number of (+/-/=) 10/6/13					

TABLE 13. Comparisons with  $F$  in literatures on IEEE CEC 2017 problems for  $D = 30$ .

(a) 0.1, 0.9, rand and SinDE

$f_i$	0.1			0.9			rand			SinDE		
	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=
1	1.96E+05	4.31E+05	-	1.25E+04	1.05E+04	-	4.36E+03	2.94E+03	-	1.88E+00	3.37E+00	+
3	3.65E+04	8.57E+05	-	6.36E+04	2.86E+04	-	9.54E+03	2.60E+04	-	1.89E-01	4.22E-01	+
4	1.25E+02	1.04E+01	-	8.80E+01	5.20E-01	-	8.78E+01	1.23E+00	-	7.95E+01	8.30E+00	+
5	2.69E+01	5.63E+00	-	4.28E+01	6.12E+00	-	3.69E+01	4.00E+00	-	2.86E+01	3.50E+00	-
6	2.49E-06	5.27E-06	-	3.42E-05	1.49E-05	-	3.26E-06	2.41E-06	-	5.48E-08	1.77E-07	-
7	5.82E+01	7.43E+00	=	9.22E+01	6.39E+00	-	8.13E+01	6.78E+00	-	6.82E+01	4.77E+00	-
8	2.90E+01	6.83E+00	-	4.60E+01	6.28E+00	-	4.08E+01	4.50E+00	-	3.23E+01	3.73E+00	-
9	1.65E+00	1.51E+00	-	4.22E+01	1.64E+01	-	1.88E+01	8.18E+00	-	9.51E-09	6.38E-08	=
10	<b>1.81E+03</b>	6.39E+02	=	2.39E+03	2.11E+02	-	2.31E+03	6.39E+02	-	2.10E+03	2.65E+02	-
11	1.39E+02	4.66E+01	-	6.35E+01	2.42E+01	-	2.02E+01	1.51E+01	-	1.16E+01	4.36E+00	-
12	8.34E+05	6.47E+05	-	1.83E+06	8.93E+05	-	1.20E+04	9.60E+03	-	<b>3.51E+02</b>	1.70E+02	+
13	1.58E+04	9.13E+03	-	1.54E+02	5.81E+01	-	4.36E+01	1.95E+01	-	2.96E+01	1.20E+01	-
14	1.09E+04	2.49E+04	-	5.82E+01	2.65E+01	-	5.76E+01	2.11E+01	-	3.44E+01	1.01E+01	-
15	5.82E+03	4.88E+03	-	2.08E+01	8.60E+00	-	1.43E+01	6.33E+00	-	1.05E+01	2.47E+00	-
16	1.21E+02	9.33E+01	-	4.56E+02	1.24E+02	-	4.37E+02	1.96E+02	-	2.66E+02	1.18E+02	-
17	4.42E+01	8.49E+00	-	9.46E+01	1.96E+01	-	8.20E+01	1.71E+01	-	6.02E+01	1.04E+01	-
18	1.56E+05	1.00E+05	-	1.36E+02	9.25E+01	-	2.79E+01	2.30E+00	-	2.37E+01	4.42E+00	-
19	6.51E+03	6.40E+03	-	1.49E+01	5.00E+00	-	1.25E+01	3.55E+00	-	9.55E+00	2.53E+00	-
20	4.04E+01	9.47E+00	-	8.66E+01	3.81E+01	-	6.82E+01	2.93E+01	-	4.65E+01	1.39E+01	-
21	2.31E+02	6.41E+00	-	2.45E+02	5.47E+00	-	2.41E+02	5.59E+00	-	2.32E+02	4.10E+00	-
22	1.04E+02	5.22E+00	-	<b>1.00E+02</b>	6.95E-06	-	<b>1.00E+02</b>	3.95E-07	-	<b>1.00E+02</b>	9.51E-08	-
23	3.83E+02	8.79E+00	-	3.89E+02	5.11E+00	-	3.82E+02	4.52E+00	-	3.75E+02	4.36E+00	-
24	4.54E+02	8.00E+00	-	4.63E+02	5.69E+00	-	4.56E+02	4.46E+00	-	4.47E+02	3.53E+00	-
25	3.97E+02	6.95E+00	-	<b>3.87E+02</b>	1.88E-01	-	<b>3.87E+02</b>	2.08E-01	-	<b>3.87E+02</b>	2.04E-02	+
26	1.32E+03	1.08E+02	-	1.42E+03	7.00E+01	-	1.34E+03	5.47E+01	-	1.21E+03	5.25E+01	-
27	5.17E+02	6.28E+00	-	5.07E+02	4.12E+00	-	4.89E+02	8.80E+00	=	<b>4.77E+02</b>	8.35E+00	+
28	4.85E+02	1.64E+01	-	4.25E+02	4.41E+00	-	4.08E+02	1.55E+01	-	<b>3.20E+02</b>	3.97E+01	+
29	5.06E+02	3.85E+01	-	5.69E+02	2.02E+02	-	5.46E+02	1.52E+02	-	4.76E+02	2.47E+01	-
30	3.56E+04	2.94E+04	-	5.08E+03	1.50E+03	-	2.40E+03	2.11E+02	-	<b>2.03E+03</b>	4.04E+01	+
	Number of (+/-/=) 0/27/2			Number of (+/-/=) 0/29/0			Number of (+/-/=) 0/28/1			Number of (+/-/=) 8/19/2		

(b) SHADE, FiADE, Rbs, EsDE<sub>r</sub>-NR and (19)

$f_i$	SHADE			FiADE			Rbs			EsDE <sub>r</sub> -NR			(19)	
	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std
1	<b>5.75E-03</b>	1.62E-02	+	5.20E+03	3.27E+03	-	2.50E+03	2.74E+03	-	7.85E+00	3.61E+01	+	7.15E+01	1.72E+02
3	<b>2.95E-03</b>	5.72E-03	+	3.53E+04	3.58E+04	-	2.78E+03	1.45E+03	-	5.68E-02	7.97E-02	+	1.48E+01	1.93E+01
4	<b>7.43E+01</b>	1.13E+01	+	8.84E+01	6.28E-01	-	1.01E+02	1.75E+01	-	8.01E+00	9.09E+00	+	8.53E+01	7.72E-01
5	3.03E+01	8.11E+00	-	4.17E+01	4.01E+00	-	4.36E+01	1.24E+01	-	3.40E+01	5.79E+00	-	<b>2.33E+01</b>	3.78E+00
6	1.16E-07	3.09E-07	-	5.92E-06	3.98E-06	-	1.38E-02	2.19E-02	-	6.46E-07	1.67E-06	-	<b>2.69E-09</b>	1.92E-08
7	5.89E+01	8.55E+00	=	8.59E+01	6.17E+00	-	8.84E+01	1.83E+01	-	7.08E+01	7.07E+00	-	<b>5.76E+01</b>	3.26E+00
8	3.07E+01	9.75E+00	-	4.50E+01	4.33E+00	-	4.48E+01	1.13E+01	-	3.62E+01	7.41E+00	-	<b>2.27E+01</b>	3.35E+00
9	<b>0.00E+00</b>	0.00E+00	=	2.96E+01	1.12E+01	-	5.14E+00	5.37E+00	-	1.76E-03	1.25E-02	=	<b>0.00E+00</b>	0.00E+00
10	2.09E+03	4.08E+02	=	2.32E+03	2.26E+02	-	2.94E+03	5.33E+02	-	2.13E+03	4.15E+02	-	1.97E+03	2.30E+02
11	1.30E+01	1.29E+01	-	5.14E+01	3.06E+01	-	6.24E+01	2.86E+01	-	1.04E+01	4.03E+00	=	<b>8.60E+00</b>	3.09E+00
12	8.06E+02	7.09E+02	-	2.55E+05	4.72E+05	-	1.17E+05	1.02E+05	-	2.28E+03	2.05E+03	-	5.02E+02	2.64E+02
13	2.55E+01	9.81E+00	-	1.05E+02	5.04E+01	-	4.51E+03	4.22E+03	-	2.53E+01	9.57E+00	-	<b>2.16E+01</b>	9.48E+00
14	<b>1.38E+01</b>	9.32E+00	+	6.98E+01	2.16E+01	-	4.96E+01	1.58E+01	-	1.77E+01	1.08E+01	+	2.66E+01	5.21E+00
15	9.05E+00	2.88E+00	-	1.85E+01	9.84E+00	-	5.70E+01	3.03E+01	-	8.09E+00	2.38E+00	=	<b>7.86E+00</b>	1.70E+00
16	1.61E+02	1.40E+02	-	4.45E+02	1.26E+02	-	6.06E+02	2.35E+02	-	1.27E+02	1.38E+02	-	<b>4.03E+01</b>	5.70E+01
17	3.38E+01	1.47E+01	=	8.53E+01	1.86E+01	-	1.20E+02	9.77E+01	-	4.96E+01	2.16E+01	-	<b>3.12E+01</b>	6.63E+00
18	<b>2.20E+01</b>	6.43E+00	=	3.67E+01	8.80E+00	-	5.05E+03	4.88E+03	-	2.32E+01	5.02E+00	-	2.34E+01	1.15E+00
19	5.48E+00	1.54E+00	+	1.79E+01	9.15E+00	-	2.09E+02	6.25E+02	-	<b>5.09E+00</b>	1.45E+00	+	6.49E+00	1.71E+00
20	2.82E+01	2.80E+01	=	7.93E+01	3.10E+01	-	1.73E+02	1.02E+02	-	2.77E+01	1.95E+01	-	<b>2.00E+01</b>	7.42E+00
21	2.30E+02	9.80E+00	-	2.46E+02	3.87E+00	-	2.47E+02	1.16E+01	-	2.36E+02	8.05E+00	-	<b>2.25E+02</b>	5.45E+00
22	<b>1.00E+02</b>	1.62E-12	=	<b>1.00E+02</b>	3.48E-04	-	1.01E+02	1.62E+00	-	<b>1.00E+02</b>	2.18E-12	=	<b>1.00E+02</b>	1.01E-10
23	3.78E+02	1.09E+01	-	3.87E+02	4.62E+00	-	3.93E+02	1.22E+01	-	3.80E+02	7.73E+00	-	<b>3.67E+02</b>	8.48E+00
24	4.52E+02	9.25E+00	-	4.61E+02	4.26E+00	-	4.69E+02	1.21E+01	-	4.53E+02	9.25E+00	-	<b>4.42E+02</b>	8.45E+00
25	<b>3.87E+02</b>	6.35E-02	-	<b>3.87E+02</b>	1.80E-01	-	3.88E+02	1.02E+00	-	<b>3.87E+02</b>	8.06E-02	-	<b>3.87E+02</b>	2.22E-02
26	1.24E+03	1.12E+02	-	1.40E+03	6.16E+01	-	1.50E+03	1.54E+02	-	1.27E+03	9.64E+01	-	<b>1.08E+03</b>	7.99E+01
27	4.92E+02	1.02E+01	-	5.03E+02	5.18E+00	-	5.16E+02	6.69E+00	-	4.89E+02	5.86E+00	=	4.87E+02	8.69E+00
28	3.33E+02	5.03E+01	+	4.24E+02	4.87E+00	-	4.18E+02	1.66E+01	-	3.37E+02	4.91E+01	+	3.68E+02	3.52E+01
29	<b>4.21E+02</b>	3.41E+01	+	5.31E+02	1.26E+02	-	5.98E+02	1.10E+02	-	4.31E+02	3.82E+01	=	4.42E+02	3.21E+01
30	<b>2.03E+03</b>	4.65E+01	+	3.77E+03	9.17E+02	-	3.94E+03	9.98E+02	-	2.06E+03	5.32E+01	+	2.13E+03	7.95E+01
	Number of (+/-/=) 8/13/8			Number of (+/-/=) 0/29/0			Number of (+/-/=) 0/29/0			Number of (+/-/=) 7/16/6				



**TABLE 14. Comparisons with  $F$  in literatures on IEEE CEC 2017 problems for  $D = 50$ .**

(a) 0.1, 0.9, rand and SinDE

$f_i$	0.1			0.9			rand			SinDE		
	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=
1	5.11E+05	1.64E+06	-	6.59E+05	4.76E+05	-	9.02E+04	6.17E+04	-	4.59E+03	3.73E+03	-
3	9.69E+04	1.66E+04	-	1.62E+05	4.45E+04	-	8.31E+04	7.21E+04	-	2.72E+03	1.09E+03	+
4	2.19E+02	2.61E+01	-	1.61E+02	1.43E+01	-	1.58E+02	1.16E+01	-	1.18E+02	3.54E+01	=
5	5.55E+01	1.12E+01	+	1.25E+02	8.03E+00	-	1.11E+02	9.21E+00	-	8.76E+01	6.89E+00	-
6	9.74E-06	1.36E-05	-	3.42E-03	9.62E-04	-	9.60E-04	3.29E-04	-	4.33E-06	2.63E-06	-
7	1.07E+02	1.39E+01	+	2.15E+02	1.12E+01	-	2.01E+02	1.10E+01	-	1.65E+02	9.52E+00	-
8	5.63E+01	1.21E+01	+	1.20E+02	8.69E+00	-	1.06E+02	8.39E+00	-	8.52E+01	7.87E+00	-
9	7.68E+00	5.16E+00	-	3.94E+02	9.49E+01	-	1.98E+02	6.00E+01	-	3.37E+01	1.78E+01	-
10	4.61E+03	1.05E+03	=	5.16E+03	3.64E+02	-	4.98E+03	3.35E+02	-	4.74E+03	2.88E+02	-
11	5.37E+02	3.13E+02	-	1.90E+02	8.45E+01	-	7.80E+01	4.34E+01	-	4.24E+01	6.90E+00	-
12	7.18E+06	4.82E+06	-	1.95E+07	5.33E+06	-	1.56E+06	6.81E+05	-	9.38E+04	7.28E+04	=
13	5.90E+03	6.50E+03	-	2.12E+06	2.05E+06	-	5.17E+02	1.41E+02	-	<b>1.18E+02</b>	2.61E+01	+
14	8.82E+04	9.29E+04	-	7.98E+01	2.83E+01	-	7.92E+01	4.09E+01	-	6.06E+01	2.06E+01	-
15	4.61E+03	1.95E+03	-	2.30E+02	7.97E+01	-	6.48E+01	1.94E+01	-	3.44E+01	5.99E+00	=
16	<b>4.48E+02</b>	1.49E+02	+	1.19E+03	1.71E+02	-	1.10E+03	1.63E+02	-	1.03E+03	3.91E+02	-
17	<b>1.73E+02</b>	7.30E+01	+	9.21E+02	1.51E+02	-	8.39E+02	1.24E+02	-	6.93E+02	1.66E+02	-
18	8.67E+05	6.28E+05	-	3.86E+04	2.15E+04	-	3.26E+02	2.71E+02	-	<b>3.90E+01</b>	5.28E+00	+
19	1.50E+04	4.54E+03	-	1.73E+02	1.54E+02	-	3.09E+01	6.85E+00	-	2.32E+01	4.92E+00	-
20	<b>1.46E+02</b>	5.93E+01	+	6.33E+02	1.18E+02	-	5.85E+02	1.25E+02	-	5.12E+02	1.57E+02	-
21	<b>2.56E+02</b>	1.39E+01	+	3.28E+02	1.00E+01	-	3.09E+02	8.56E+00	-	2.90E+02	6.44E+00	-
22	1.63E+03	2.12E+03	-	3.57E+03	2.70E+03	-	2.14E+03	2.58E+03	-	1.05E+03	2.01E+03	-
23	4.90E+02	1.25E+01	-	5.50E+02	9.84E+00	-	5.30E+02	8.66E+00	-	5.04E+02	7.59E+00	-
24	5.63E+02	1.37E+01	-	6.19E+02	1.03E+01	-	6.00E+02	8.68E+00	-	5.77E+02	7.26E+00	-
25	5.91E+02	2.04E+01	-	5.23E+02	9.76E+00	-	5.23E+02	6.55E+00	-	<b>4.84E+02</b>	7.78E+00	+
26	1.79E+03	1.39E+02	-	2.42E+03	8.41E+01	-	2.20E+03	1.05E+02	-	1.93E+03	7.65E+01	-
27	6.80E+02	3.51E+01	-	5.88E+02	1.48E+01	-	5.63E+02	3.87E+01	-	5.15E+02	1.29E+01	-
28	6.89E+02	4.86E+01	-	4.73E+02	4.41E+00	-	4.67E+02	2.28E+00	-	<b>4.59E+02</b>	3.06E-02	+
29	5.08E+02	8.09E+01	-	6.89E+02	9.00E+01	-	7.17E+02	4.27E+02	-	5.99E+02	1.79E+02	-
30	2.14E+06	6.10E+05	-	9.97E+05	2.01E+05	-	6.87E+05	4.50E+04	-	6.03E+05	2.05E+04	=
	Number of (+/-/=) 7/21/1			Number of (+/-/=) 0/29/0			Number of (+/-/=) 0/29/0			Number of (+/-/=) 5/20/4		

(b) SHADE, FiADE, Rbs, EsDE<sub>r</sub>-NR and (19)

$f_i$	SHADE			FiADE			Rbs			EsDE <sub>r</sub> -NR			(19)	
	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std
1	3.93E+03	3.09E+03	-	2.99E+05	1.76E+05	-	2.70E+05	1.41E+06	-	5.20E+03	3.68E+03	-	<b>2.37E+03</b>	2.07E+03
3	3.78E+03	2.60E+04	+	1.54E+05	5.01E+04	-	1.55E+04	3.84E+03	-	<b>4.53E+02</b>	2.85E+02	+	6.58E+03	1.85E+03
4	1.29E+02	4.93E+01	-	1.57E+02	1.15E+01	-	1.73E+02	3.91E+01	-	1.37E+02	3.89E+01	-	<b>1.06E+02</b>	2.91E+01
5	<b>5.19E+01</b>	1.33E+01	+	1.17E+02	8.85E+00	-	9.14E+01	2.10E+01	-	6.87E+01	2.01E+01	=	6.40E+01	6.83E+00
6	<b>3.58E-06</b>	7.92E-06	-	7.12E-04	2.56E-04	-	1.19E-01	8.44E-02	-	1.36E-05	1.71E-05	-	4.01E-08	6.89E-08
7	<b>1.02E+02</b>	1.17E+01	+	2.13E+02	1.02E+01	-	1.47E+02	1.87E+01	-	1.08E+02	2.17E+01	+	1.29E+02	7.01E+00
8	<b>4.92E+01</b>	1.11E+01	+	1.17E+02	6.35E+00	-	9.20E+01	1.98E+01	-	6.90E+01	2.07E+01	=	6.45E+01	1.12E+01
9	7.63E-02	1.98E-01	-	2.87E+02	7.23E+01	-	3.78E+01	2.55E+01	-	3.03E-01	4.40E-01	-	<b>5.31E-08</b>	2.53E-07
10	4.59E+03	7.11E+02	=	5.13E+03	2.71E+02	-	5.57E+03	7.18E+02	-	<b>4.37E+03</b>	7.86E+02	=	4.91E+03	1.67E+03
11	4.30E+01	7.45E+00	-	1.22E+02	4.52E+01	-	1.22E+02	3.85E+01	-	4.41E+01	6.68E+00	-	<b>3.46E+01</b>	4.59E+00
12	<b>7.46E+04</b>	4.19E+04	+	2.12E+07	8.01E+06	-	1.91E+06	9.21E+05	-	1.47E+05	1.03E+05	-	1.11E+05	9.05E+04
13	1.30E+02	3.41E+01	=	3.10E+03	2.29E+03	-	1.54E+03	1.38E+03	-	1.42E+02	4.07E+01	=	1.31E+02	3.61E+01
14	4.08E+01	6.91E+00	=	8.07E+01	4.01E+01	-	1.20E+02	3.26E+01	-	3.88E+01	5.50E+00	=	<b>3.87E+01</b>	6.12E+00
15	4.48E+01	1.04E+01	-	1.37E+02	3.97E+01	-	1.77E+03	1.42E+03	-	4.41E+01	1.04E+01	-	<b>3.29E+01</b>	4.89E+00
16	4.87E+02	2.30E+02	+	1.25E+03	1.52E+02	-	1.12E+03	3.57E+02	-	5.50E+02	2.24E+02	=	5.95E+02	3.22E+02
17	2.48E+02	1.42E+02	+	9.27E+02	1.38E+02	-	7.50E+02	2.12E+02	-	3.43E+02	1.80E+02	+	4.65E+02	1.65E+02
18	4.37E+01	8.10E+00	=	1.26E+04	5.90E+03	-	1.53E+04	1.50E+04	-	5.33E+01	1.29E+01	-	4.34E+01	7.69E+00
19	1.87E+01	3.95E+00	=	5.50E+01	2.27E+01	-	9.24E+03	4.91E+03	-	1.89E+01	4.10E+00	-	<b>1.73E+01</b>	2.94E+00
20	2.10E+02	1.93E+02	+	5.78E+02	1.17E+02	-	5.52E+02	2.19E+02	-	1.64E+02	1.20E+02	+	3.61E+02	1.30E+02
21	2.60E+02	1.33E+01	+	3.22E+02	7.49E+00	-	2.93E+02	2.29E+01	-	2.73E+02	2.12E+01	=	2.66E+02	7.27E+00
22	1.85E+03	2.31E+03	-	2.95E+03	2.66E+03	-	4.52E+03	2.48E+03	-	1.09E+03	1.92E+03	=	<b>3.64E+02</b>	1.07E+03
23	<b>4.78E+02</b>	1.44E+01	=	5.39E+02	8.23E+00	-	5.25E+02	2.12E+01	-	4.86E+02	2.00E+01	=	4.80E+02	9.67E+00
24	5.58E+02	1.54E+01	-	6.12E+02	9.42E+00	-	5.91E+02	2.27E+01	-	5.61E+02	1.55E+01	-	<b>5.44E+02</b>	1.16E+01
25	4.95E+02	2.56E+01	=	5.22E+02	8.73E+00	-	5.50E+02	2.34E+01	-	4.89E+02	2.29E+01	=	4.86E+02	9.81E+00
26	1.64E+03	1.44E+02	-	2.32E+03	8.20E+01	-	2.19E+03	2.62E+02	-	1.79E+03	2.33E+02	-	<b>1.53E+03</b>	1.47E+02
27	5.26E+02	1.69E+01	-	6.01E+02	2.46E+01	-	6.32E+02	4.26E+01	-	5.29E+02	1.74E+01	-	<b>5.11E+02</b>	1.02E+01
28	4.62E+02	1.16E+01	+	4.69E+02	2.61E+00	-	5.37E+02	2.38E+01	-	4.60E+02	6.84E+00	+	<b>4.59E+02</b>	6.77E-02
29	<b>3.59E+02</b>	5.42E+01	+	6.46E+02	1.02E+02	-	8.12E+02	2.24E+02	-	3.85E+02	7.37E+01	+	3.91E+02	3.00E+01
30	<b>5.98E+05</b>	1.54E+04	=	8.63E+05	1.12E+05	-	8.24E+05	8.70E+04	-	6.06E+05	2.50E+04	=	5.99E+05	1.70E+04
	Number of (+/-/=) 11/10/8			Number of (+/-/=) 0/29/0			Number of (+/-/=) 0/29/0			Number of (+/-/=) 6/12/11				



TABLE 15. Comparisons with  $F$  in literatures on IEEE CEC 2017 problems for  $D = 100$ .

(a) 0.1, 0.9, rand and SinDE

$f_i$	0.1			0.9			rand			SinDE		
	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=
1	7.81E+04	2.81E+05	-	7.16E+06	2.56E+06	-	1.51E+06	7.72E+05	-	2.06E+04	1.18E+04	-
3	2.65E+05	2.35E+04	-	4.00E+05	5.45E+04	-	3.39E+05	9.16E+04	-	1.39E+05	9.41E+04	-
4	3.90E+02	4.23E+01	-	2.99E+02	1.41E+01	-	2.77E+02	9.76E+00	-	2.49E+02	1.98E+01	-
5	1.19E+02	2.16E+01	+	4.58E+02	1.70E+01	-	4.11E+02	2.05E+01	-	3.47E+02	1.70E+01	-
6	4.00E-05	5.73E-05	=	2.19E-01	3.04E-02	-	1.09E-01	1.79E-02	-	8.89E-03	1.45E-03	-
7	2.37E+02	3.25E+01	+	7.53E+02	2.64E+01	-	6.95E+02	2.74E+01	-	5.91E+02	1.89E+01	-
8	1.19E+02	2.67E+01	+	4.56E+02	1.71E+01	-	4.10E+02	1.81E+01	-	3.48E+02	1.51E+01	-
9	3.63E+01	2.07E+01	-	5.78E+03	6.75E+02	-	3.97E+03	4.73E+02	-	1.59E+03	2.12E+02	-
10	1.20E+04	2.15E+03	+	1.58E+04	2.14E+03	-	1.53E+04	2.11E+03	-	1.46E+04	5.59E+02	=
11	2.13E+04	6.32E+03	-	3.30E+04	7.07E+03	-	1.36E+03	3.03E+02	-	3.44E+02	5.16E+01	-
12	4.41E+07	1.71E+07	-	8.99E+07	1.74E+07	-	6.71E+07	2.99E+07	-	4.01E+06	1.63E+06	-
13	2.37E+03	7.71E+02	-	1.71E+06	1.42E+06	-	5.05E+05	9.25E+05	-	2.37E+03	1.05E+03	-
14	1.92E+06	1.05E+06	-	2.42E+05	1.49E+05	-	4.66E+02	2.07E+02	-	1.00E+02	1.73E+01	-
15	6.44E+02	3.08E+02	-	1.18E+06	1.27E+06	-	1.70E+03	1.35E+03	-	2.64E+02	3.91E+01	-
16	<b>1.88E+03</b>	5.08E+02	+	4.13E+03	2.96E+02	-	4.00E+03	2.76E+02	-	3.78E+03	3.18E+02	-
17	<b>9.98E+02</b>	2.30E+02	+	3.25E+03	2.37E+02	-	3.08E+03	2.41E+02	-	2.85E+03	2.21E+02	=
18	1.84E+06	7.16E+05	-	3.34E+06	8.61E+06	-	1.04E+05	4.17E+04	-	6.75E+03	3.35E+03	-
19	7.82E+02	5.03E+02	-	1.10E+06	1.44E+06	-	3.68E+03	3.19E+03	-	1.10E+02	1.84E+01	-
20	<b>1.04E+03</b>	2.50E+02	+	2.96E+03	2.83E+02	-	2.92E+03	3.47E+02	-	2.72E+03	3.86E+02	=
21	3.53E+02	2.18E+01	+	6.92E+02	1.63E+01	-	6.53E+02	1.68E+02	-	5.78E+02	1.82E+01	-
22	<b>1.23E+04</b>	1.30E+03	+	1.66E+04	2.15E+03	-	1.61E+04	4.76E+02	-	1.62E+04	2.95E+03	=
23	6.60E+02	2.47E+01	=	7.52E+02	1.22E+01	-	7.28E+02	1.15E+01	-	6.90E+02	1.13E+01	-
24	1.04E+03	2.55E+01	=	1.28E+03	2.08E+01	-	1.24E+03	1.47E+01	-	1.16E+03	1.52E+01	-
25	9.84E+02	3.70E+01	-	9.44E+02	2.73E+01	-	9.02E+02	2.68E+01	-	7.88E+02	3.77E+01	=
26	4.63E+03	2.92E+02	-	7.73E+03	1.99E+02	-	7.16E+03	2.05E+02	-	6.35E+03	1.45E+02	-
27	7.99E+02	4.85E+01	-	7.42E+02	1.79E+01	-	7.78E+02	3.70E+01	-	6.54E+02	2.41E+01	-
28	1.13E+03	1.34E+02	-	6.99E+02	1.81E+01	-	6.72E+02	1.84E+01	-	5.88E+02	2.71E+01	-
29	1.89E+03	3.20E+02	=	3.25E+03	2.03E+02	-	3.20E+03	2.13E+02	-	2.94E+03	4.30E+02	-
30	7.78E+04	4.77E+04	-	1.02E+05	4.12E+04	-	2.35E+04	1.95E+04	-	4.38E+03	9.89E+02	-
	Number of (+/-/=) 9/16/4			Number of (+/-/=) 0/29/0			Number of (+/-/=) 0/29/0			Number of (+/-/=) 0/24/5		

(b) SHADE, FiADE, Rbs, EsDE<sub>r</sub>-NR and (19)

$f_i$	SHADE			FiADE			Rbs			EsDE <sub>r</sub> -NR			(19)	
	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std	+/-/=	Mean	Std
1	5.22E+03	2.95E+03	=	3.39E+06	1.95E+06	-	8.47E+06	3.05E+07	-	5.39E+03	2.67E+03	=	<b>4.85E+03</b>	3.08E+03
3	<b>2.79E+04</b>	5.43E+04	+	3.97E+05	5.25E+04	-	8.19E+04	1.22E+04	-	2.93E+04	6.68E+04	+	7.67E+04	1.09E+04
4	2.31E+02	3.07E+01	=	2.87E+02	1.35E+01	-	3.19E+02	3.61E+01	-	2.36E+02	2.65E+01	-	<b>2.27E+02</b>	1.42E+01
5	<b>1.09E+02</b>	2.05E+01	+	4.33E+02	1.87E+01	-	1.91E+02	4.26E+01	+	9.92E+01	2.16E+01	+	2.22E+02	2.68E+01
6	1.07E-04	6.74E-05	-	7.69E-02	1.50E-02	-	8.78E-01	3.96E-01	-	4.55E-04	3.73E-04	-	<b>3.79E-05</b>	3.85E-05
7	2.06E+02	2.07E+01	+	7.23E+02	2.41E+01	-	3.25E+02	4.50E+01	+	<b>1.99E+02</b>	1.98E+01	+	4.35E+02	3.82E+01
8	<b>1.10E+02</b>	1.86E+01	+	4.33E+02	1.95E+01	-	1.93E+02	3.59E+01	+	9.61E+01	2.06E+01	+	2.31E+02	2.33E+01
9	7.08E+00	5.58E+00	-	4.55E+03	7.10E+02	-	3.50E+02	1.23E+02	-	1.03E+01	6.13E+00	-	<b>8.83E-02</b>	1.89E-01
10	<b>1.10E+04</b>	1.42E+03	+	1.59E+04	2.81E+03	-	1.25E+04	1.29E+03	+	1.14E+04	1.64E+03	+	1.58E+04	4.57E+03
11	3.06E+02	6.00E+01	-	1.11E+04	1.13E+04	-	1.87E+03	8.08E+02	-	3.34E+02	7.36E+01	-	<b>2.78E+02</b>	4.70E+01
12	<b>9.42E+05</b>	4.07E+05	+	7.23E+07	1.73E+07	-	1.39E+07	5.70E+06	-	1.53E+06	5.48E+05	+	2.28E+06	1.04E+06
13	<b>1.13E+03</b>	8.26E+02	+	7.63E+05	6.71E+05	-	2.78E+03	8.78E+02	-	1.81E+03	1.77E+03	=	1.81E+03	5.81E+02
14	1.52E+02	2.67E+01	-	4.28E+04	2.62E+04	-	1.30E+04	1.66E+04	-	1.75E+02	2.96E+01	-	<b>9.15E+01</b>	1.21E+01
15	2.94E+02	5.39E+01	-	5.77E+05	6.09E+05	-	6.30E+02	4.19E+02	-	3.11E+02	7.17E+01	-	<b>2.30E+02</b>	5.04E+01
16	1.99E+03	5.98E+02	+	4.10E+03	3.15E+02	-	3.08E+03	5.81E+02	+	2.07E+03	5.83E+02	+	3.40E+03	8.65E+02
17	1.03E+03	3.41E+02	+	3.18E+03	2.11E+02	-	2.16E+03	4.25E+02	+	1.19E+03	3.57E+02	+	2.87E+03	5.98E+02
18	<b>2.51E+03</b>	1.32E+03	+	5.56E+05	8.09E+05	-	9.83E+04	4.04E+04	-	6.93E+03	2.64E+03	-	4.75E+03	2.76E+03
19	1.35E+02	3.14E+01	-	1.91E+06	1.76E+06	-	9.16E+02	7.84E+02	-	1.53E+02	3.58E+01	-	<b>8.98E+01</b>	1.40E+01
20	1.44E+03	5.73E+02	+	2.85E+03	2.51E+02	=	2.10E+03	4.01E+02	+	1.31E+03	3.99E+02	+	2.77E+03	7.23E+02
21	3.38E+02	2.06E+01	+	6.72E+02	1.76E+01	-	4.24E+02	5.16E+01	+	<b>3.24E+02</b>	2.33E+01	+	4.55E+02	2.99E+01
22	1.26E+04	1.80E+03	+	1.63E+04	3.94E+02	-	1.39E+04	1.23E+03	+	1.25E+04	1.39E+03	+	1.67E+04	4.51E+03
23	6.56E+02	2.37E+01	=	7.34E+02	1.29E+01	-	7.61E+02	4.13E+01	-	7.11E+02	1.08E+01	-	<b>6.53E+02</b>	9.68E+00
24	9.93E+02	2.41E+01	+	1.26E+03	1.88E+01	-	1.12E+03	5.02E+01	-	<b>9.84E+02</b>	2.42E+01	+	1.04E+03	5.21E+01
25	7.86E+02	4.63E+01	=	9.17E+02	2.39E+01	-	8.81E+02	4.11E+01	-	<b>7.68E+02</b>	5.03E+01	=	7.80E+02	3.64E+01
26	4.12E+03	2.61E+02	=	7.45E+03	1.58E+02	-	5.49E+03	4.69E+02	-	<b>4.11E+03</b>	2.74E+02	=	4.15E+03	6.91E+02
27	6.44E+02	2.41E+01	-	7.54E+02	1.87E+01	-	7.78E+02	3.92E+01	-	6.51E+02	2.67E+01	-	<b>6.18E+02</b>	1.68E+01
28	6.09E+02	3.36E+01	-	6.80E+02	1.67E+01	-	7.50E+02	3.50E+01	-	5.91E+02	3.30E+01	-	<b>5.65E+02</b>	2.11E+01
29	<b>1.51E+03</b>	3.22E+02	+	3.22E+03	2.07E+02	-	2.86E+03	4.67E+02	-	<b>1.51E+03</b>	3.04E+02	+	1.86E+03	4.90E+02
30	<b>3.27E+03</b>	7.56E+02	=	7.33E+04	3.00E+04	-	1.46E+04	1.10E+04	-	3.93E+03	1.18E+03	-	3.36E+03	5.15E+02
	Number of (+/-/=) 15/8/6			Number of (+/-/=) 0/28/1			Number of (+/-/=) 9/20/0			Number of (+/-/=) 13/12/4				

## REFERENCES

- [1] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim.*, vol. 11, no. 4, pp. 341–359, Dec. 1997.
- [2] D. Izzo, M. Märten, and B. Pan, "A survey on artificial intelligence trends in spacecraft guidance dynamics and control," *Astrodynamics*, vol. 3, no. 4, pp. 287–299, Dec. 2019.
- [3] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 4–31, Feb. 2011.
- [4] S. Das, S. S. Mullick, and P. N. Suganthan, "Recent advances in differential evolution—An updated survey," *Swarm Evol. Comput.*, vol. 27, pp. 1–30, Apr. 2016.
- [5] S. Li, X. Huang, and B. Yang, "Review of optimization methodologies in global and China trajectory optimization competitions," *Progr. Aerosp. Sci.*, vol. 102, pp. 60–75, Oct. 2018.
- [6] D. R. Penas, A. Gómez, B. B. Fraguera, M. J. Martín, and S. Cervino, "Enhanced global optimization methods applied to complex Fisheries stock assessment models," *Appl. Soft Comput.*, vol. 77, pp. 50–66, Apr. 2019.
- [7] R. D. Al-Dabbagh, F. Neri, N. Idris, and M. S. Baba, "Algorithmic design issues in adaptive differential evolution schemes: Review and taxonomy," *Swarm Evol. Comput.*, vol. 43, pp. 284–311, Dec. 2018.
- [8] R. Gämperle, S. D. Müller, and P. Koumoutsakos, "A parameter study for differential evolution," *Adv. Intell. Syst., Fuzzy Syst., Evol. Comput.*, vol. 10, no. 10, pp. 293–298, Aug. 2002.
- [9] J. Ronkkonen, S. Kukkonen, and K. V. Price, "Real-parameter optimization with differential evolution," in *Proc. IEEE Congr. Evol. Comput.*, vol. 1, Dec. 2005, pp. 506–513.
- [10] Z.-J. Wang, Z.-H. Zhan, Y. Lin, W.-J. Yu, H.-Q. Yuan, T.-L. Gu, S. Kwong, and J. Zhang, "Dual-strategy differential evolution with affinity propagation clustering for multimodal optimization problems," *IEEE Trans. Evol. Comput.*, vol. 22, no. 6, pp. 894–908, Dec. 2018.
- [11] Y. Wang, B. Xu, G. Sun, and S. Yang, "A two-phase differential evolution for uniform designs in constrained experimental domains," *IEEE Trans. Evol. Comput.*, vol. 21, no. 5, pp. 665–680, Oct. 2017.
- [12] S. Das, A. Konar, and U. K. Chakraborty, "Two improved differential evolution schemes for faster global search," in *Proc. Conf. Genetic Evol. Comput. (GECCO)*. New York, NY, USA: ACM, 2005, pp. 991–998.
- [13] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 398–417, Apr. 2009.
- [14] S. Biswas, S. Kundu, S. Das, and A. V. Vasilakos, "Teaching and learning best Differential Evolution with self adaptation for real parameter optimization," in *Proc. IEEE Congr. Evol. Comput.*, Jun. 2013, pp. 1115–1122.
- [15] D. Zou, J. Wu, L. Gao, and S. Li, "A modified differential evolution algorithm for unconstrained optimization problems," *Neurocomputing*, vol. 120, pp. 469–481, Nov. 2013.
- [16] J. Zhang and A. C. Sanderson, "JADE: Adaptive differential evolution with optional external archive," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 945–958, Oct. 2009.
- [17] J. Brest, S. Greiner, B. Boskovic, M. Mernik, and V. Zumer, "Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems," *IEEE Trans. Evol. Comput.*, vol. 10, no. 6, pp. 646–657, Dec. 2006.
- [18] R. Tanabe and A. Fukunaga, "Success-history based parameter adaptation for differential evolution," in *Proc. IEEE Congr. Evol. Comput.*, Jun. 2013, pp. 71–78.
- [19] A. P. Piotrowski and J. J. Napiorkowski, "Step-by-step improvement of jade and shade-based algorithms: Success or failure?" *Swarm Evol. Comput.*, vol. 43, pp. 88–108, Dec. 2018.
- [20] A. Draa, S. Bouzoubia, and I. Boukhalifa, "A sinusoidal differential evolution algorithm for numerical optimisation," *Appl. Soft Comput.*, vol. 27, pp. 99–126, Feb. 2015.
- [21] N. H. Awad, M. Z. Ali, P. N. Suganthan, and R. G. Reynolds, "An ensemble sinusoidal parameter adaptation incorporated with L-SHADE for solving CEC2014 benchmark problems," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2016, pp. 2958–2965.
- [22] N. H. Awad, M. Z. Ali, and P. N. Suganthan, "Ensemble of parameters in a sinusoidal differential evolution with Niching-based population reduction," *Swarm Evol. Comput.*, vol. 39, pp. 141–156, Apr. 2018.
- [23] M. M. Ali and A. Törn, "Population set-based global optimization algorithms: Some modifications and numerical studies," *Comput. Oper. Res.*, vol. 31, no. 10, pp. 1703–1725, Sep. 2004.
- [24] A. Ghosh, S. Das, A. Chowdhury, and R. Giri, "An improved differential evolution algorithm with fitness-based adaptation of the control parameters," *Inf. Sci.*, vol. 181, no. 18, pp. 3749–3765, Sep. 2011.
- [25] L. Tang, Y. Dong, and J. Liu, "Differential evolution with an individual-dependent mechanism," *IEEE Trans. Evol. Comput.*, vol. 19, no. 4, pp. 560–574, Aug. 2015.
- [26] A. W. Mohamed and P. N. Suganthan, "Real-parameter unconstrained optimization based on enhanced fitness-adaptive differential evolution algorithm with novel mutation," *Soft Comput.*, vol. 22, no. 10, pp. 3215–3235, May 2018.
- [27] K. V. Price, R. Storn, and J. Lampinen, *Differential Evolution—A Practical Approach to Global Optimization*, vol. 141. Berlin, Germany: Springer-Verlag, 2005.
- [28] S. M. Islam, S. Das, S. Ghosh, S. Roy, and P. N. Suganthan, "An adaptive differential evolution algorithm with novel mutation and crossover strategies for global numerical optimization," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 482–500, Apr. 2012.
- [29] A. W. Mohamed, A. A. Hadi, and K. M. Jambi, "Novel mutation strategy for enhancing SHADE and LSHADE algorithms for global numerical optimization," *Swarm Evol. Comput.*, vol. 50, Nov. 2019, Art. no. 100455.
- [30] A. W. Mohamed and A. K. Mohamed, "Adaptive guided differential evolution algorithm with novel mutation for numerical optimization," *Int. J. Mach. Learn. Cyber.*, vol. 10, no. 2, pp. 253–277, Feb. 2019.
- [31] J. Teo, "Exploring dynamic self-adaptive populations in differential evolution," *Soft Comput.*, vol. 10, no. 8, pp. 673–686, Jun. 2006.
- [32] R. Tanabe and A. S. Fukunaga, "Improving the search performance of SHADE using linear population size reduction," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2014, pp. 1658–1665.
- [33] R. Poláková, J. Tvrđík, and P. Bujok, "Differential evolution with adaptive mechanism of population size according to current population diversity," *Swarm Evol. Comput.*, vol. 50, Nov. 2019, Art. no. 100519.
- [34] L.-B. Deng, S. Wang, L.-Y. Qiao, and B.-Q. Zhang, "DE-RCO: Rotating crossover operator with multiangle searching strategy for adaptive differential evolution," *IEEE Access*, vol. 6, pp. 2970–2983, 2018.
- [35] J. J. Liang, B. Y. Qu, and P. N. Suganthan, "Problem definitions and evaluation criteria for the cec 2014 special session and competition on single objective real-parameter numerical optimization," School Elect. Eng., Nanyang Technol. Univ., Singapore, Tech. Rep. 201311, 2014.
- [36] N. H. Awad, M. Z. Ali, P. N. Suganthan, J. J. Liang, and B. Y. Qu, "Problem definitions and evaluation criteria for the cec 2017 special session and competition on single objective real-parameter numerical optimization," Nanyang Technol. Univ., Jordan Univ. Sci. Technol., Irbid, Jordan, Tech. Rep., 2017, pp. 5–27.
- [37] F. Wilcoxon, *Individual Comparisons by Ranking Methods*. New York, NY, USA: Springer, 1992.
- [38] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 3–18, Mar. 2011.



**XIAOLONG ZHU** received the B.Sc. degree from the Technology and Engineering Center for Space Utilization, Chinese Academy of Sciences, China, where he is currently pursuing the Ph.D. degree. His current research interests include intelligent computing and spacecraft trajectory optimization.



distributed space systems, astrodynamics, and GN&C.

**HAO ZHANG** received the B.Sc. and Ph.D. degrees in aerospace engineering from Beihang University, China, in 2006 and 2012, respectively. From 2012 to 2017, he was with the Asher Space Research Institute, Technion-Israel Institute of Technology, where he was a Postdoctoral Fellow. He is currently an Associate Research Fellow with the Technology and Engineering Center for Space Utilization, Chinese Academy of Sciences, Beijing, China. His research interests include dis-



**YANG GAO** received the B.Sc. degree in automation from Beihang University, China, in 1997, and the Ph.D. degree in aerospace engineering from the University of Missouri, Columbia, in 2003. He was a Postdoctoral Fellow with the University of Missouri. Since 2005, he has been with the Chinese Academy of Sciences, where he is currently a Research Professor with the Technology and Engineering Center for Space Utilization. His research interests include space mission analysis and trajectory optimization.

• • •