



## Article Improving Dual-Population Differential Evolution Based on Hierarchical Mutation and Selection Strategy

Yawei Huang \*, Xuezhong Qian 🕩 and Wei Song

School of Artificial Intelligence and Computer Science, Jiangnan University, Lihu Avenuc, Wuxi 214122, China; xzqian@jiangnan.edu.cn (X.Q.)

\* Correspondence: 6213113063@stu.jiangnan.edu.cn

Abstract: The dual-population differential evolution (DDE) algorithm is an optimization technique that simultaneously maintains two populations to balance global and local search. It has been demonstrated to outperform single-population differential evolution algorithms. However, existing improvements to dual-population differential evolution algorithms often overlook the importance of selecting appropriate mutation and selection operators to enhance algorithm performance. In this paper, we propose a dual-population differential evolution (DPDE) algorithm based on a hierarchical mutation and selection strategy. We divided the population into elite and normal subpopulations based on fitness values. Information exchange between the two subpopulations was facilitated through a hierarchical mutation strategy, promoting a balanced exploration-exploitation trade-off in the algorithm. Additionally, this paper presents a new hierarchical selection strategy aimed at improving the population's capacity to avoid local optima. It achieves this by accepting discarded trial vectors differently compared to previous methods. We expect that the newly introduced hierarchical selection and mutation strategies will work in synergy, effectively harnessing their potential to enhance the algorithm's performance. Extensive experiments were conducted on the CEC 2017 and CEC 2011 test sets. The results showed that the DPDE algorithm offers competitive performance, comparable to six state-of-the-art differential evolution algorithms.

Keywords: mutation strategy; selection strategy; multi-population; differential evolution

#### 1. Introduction

In recent years, nature-inspired optimization algorithms [1–5] have been extensively researched for solving complex optimization problems. Storn and Price, inspired by nature, proposed the differential evolution algorithm, which has been successfully applied to a broad spectrum of optimization challenges, ranging from multi-objective problems [6] to constrained optimization [7] and large-scale optimization [8,9]. Over the past few years, many excellent differential evolution algorithms have been proposed [10–14]. This article primarily focuses on enhancing the performance of DE for single-objective numerical optimization.

The single-population differential evolution algorithm requires the construction of a population composed of multiple individuals and, then, iteratively evolving the individuals through the mutation, crossover, and selection operators to continuously find better solutions. The mutation strategy has a significant impact on the performance of the algorithm [15–18]. Improper setting of the mutation strategy may lead to two extremes: (1) premature convergence and (2) inability to converge. For instance, the mutation strategy DE/rand/1 is recognized for its robust exploration capabilities, because it does not depend on specific individuals, but entirely on randomness to select vectors for driving evolution; this method is conducive to the algorithm exploring different regions of the search space. However, it is incapable of fully exploiting high-quality regions, leading to a need for more iterations to find the global optimal solution. The mutation strategy DE/best/1 operates



Citation: Huang, Y.; Qian, X.; Song, W. Improving Dual-Population Differential Evolution Based on Hierarchical Mutation and Selection Strategy. *Electronics* **2024**, *13*, 62. https://doi.org/10.3390/ electronics13010062

Academic Editor: Stefano Ferilli

Received: 4 December 2023 Revised: 14 December 2023 Accepted: 19 December 2023 Published: 22 December 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). using the best individual in the population as the base vector, thus possessing significant convergence capabilities. It efficiently guides the search process towards the direction of optimized solutions. However, this inevitably leads to a rapid decline in the diversity of the algorithm. In order to have a better balance between exploration and exploitation in these algorithms, some researchers have proposed new mutation strategies. Zhang and Anderson [19] enhanced diversity and achieved a more-balanced exploration and exploitation of evolutionary processes by introducing an external archive to guide the mutation operators. However, this method does not fully leverage the potential of superior individuals. Yintong Li [20] further proposed a new directional mutation strategy to ensure that the evolution progresses in a better direction. Wang [21] introduced the DE/current-to-lpbest/1 mutation strategy, which leverages a blend of the current individual's information and the elite solutions. This approach involves guiding the mutation process with insights from multiple elite individuals, enhancing the algorithm's ability to explore potential areas more effectively. EFADE [22] introduced a new triangular mutation operator to better balance exploration and exploitation tendencies. Some researchers have found that combining multiple mutation strategies can help further enhance the algorithm's performance when faced with different problems. Based on the above discussion, how to choose an appropriate mutation strategy to balance exploration and exploitation is challenging.

Besides the mutation strategy, the survivor selection strategy also significantly impacts the performance of the differential evolution algorithm. Most research tends to use a greedy selection strategy. This strategy, based on the principle of natural selection, accepts individuals with better fitness values and eliminates those with poorer ones. Although this can make the population converge quickly to better values, this method overlooks the information from the discarded trial individuals that could help improve the algorithm's performance, and it is inefficient when facing local optima. Some studies have shown that improving survivor selection strategies can also enhance the performance of differential evolution algorithms. Pravesh Kumar and Millie Pant [23] proposed a new selection strategy where it is no longer a competition between the target vector and its corresponding trial vector, but a screening of all candidate vectors from both, allowing better individuals to enter the next generation. Tian [24] proposed a new selection strategy, in which the selection of individuals is related not only to the fitness values, but also to their positions, enabling us to fully utilize exploration information and enhance diversity. Guo [25] proposed a subsetto-subset selection operator that divides the target population and the trial population into multiple subsets and uses a rank-based selection operator in the corresponding subsets to improve the convergence performance of DE. Zeng [26] introduced a new selection operator designed to aid individuals in stagnation. When an individual is stuck, this operator first uses the best vector previously discarded by the individual for replacement. If that does not lead to evolution, it then uses the second-best vector, and so on. It continues with all the vectors from the individual's past successful updates. However, if none of these methods help the individual improve, a drastic measure is taken. The operator replaces the individual with the best individual of itself from its history. This strategy is employed with the aim of assisting the individual to escape from local optima.

Based on the above discussion, previous studies have typically applied improved mutation and selection strategies to single-population differential evolution algorithms. However, due to the fact that single-population DE algorithms maintain only one population, the inevitable exchange guidance between individuals during the evolution process may lead to a decrease in population diversity, shifting the focus towards local search. This can result in the algorithm prematurely converging to local optima, without fully exploring other potential global optima in the search space.

In response to the limitations of single-population DE algorithms, many researchers have attempted to improve DE algorithms using multi-population techniques, achieving promising results [27–29]. Among them, the performance of dual-population algorithms stands out, in particular. Pan [30] proposed a dual-population differential evolution algorithm based on different combinations of mutation strategies. This approach enhances the

robustness of populations when facing various problems by applying mutation strategies with different exploration and exploitation capabilities to different populations. Li [31] introduced a dual-population DE algorithm based on fitness stratification, utilizing a leader population to guide an adjoint population in exploring more-promising regions while maintaining good convergence properties. Furthermore, Wang [32] presented a DE algorithm that co-evolves a population with an accompanying population. It stores suboptimal solutions discarded by the main population in the previous generation and uses these individuals to assist in the subsequent evolution through mutation strategies, greatly enhancing population diversity in this manner.

The study indicates that existing dual-population differential evolution algorithms can be categorized into three types:

- 1. Dual-population with varied mutation strategies: This approach uses different mutation strategies for exploration and exploitation to achieve a balance between global and local searches. It enhances the algorithm's convergence speed, global search capability, and robustness across various problems, facilitating the discovery of highquality solutions more efficiently. However, it overlooks inter-population information exchange, potentially impairing the algorithm's full perception of target areas and leading to performance decline. Moreover, its tendency to employ a greedy selection strategy can result in stagnation when encountering local optima.
- 2. Hierarchical guidance in dual-populations: In this approach, a hierarchy is established between two populations—an elite population guides a standard population. This guidance aims to facilitate the exchange of information and strategies between the two groups, leveraging the strengths of the elite population to improve overall performance. However, this method has its pitfalls. The over-reliance on elite individuals can lead to a decrease in diversity within the population, as the algorithm may focus too narrowly on the solutions proposed by the elite. This reduced diversity can make the algorithm susceptible to becoming trapped in local optima, where it repeatedly explores suboptimal solutions without finding better alternatives.
- 3. Main and accompanying populations' collaboration: This type involves creating an accompanying population to support the main population's evolution. The accompanying population is used to store individuals that are discarded by the main population, which can later be reintegrated into the mutation strategies. This reintegration aims to increase the diversity within the main population, countering the natural tendency of evolutionary algorithms to converge on similar solutions. However, this approach also faces a significant challenge when dealing with local optima. The presence of too many stagnant individuals, who do not contribute to further evolution, in both the main and auxiliary populations can severely hinder the evolutionary process, making it difficult for the algorithm to progress beyond suboptimal solutions.

Based on the above discussion, we propose a dual-population differential evolution algorithm (DPDE) based on a hierarchical mutation and selection strategy. On the one hand, we divided the population into different subpopulations based on fitness and implemented a hierarchical mutation strategy to better balance exploration and exploitation. On the other hand, we introduced a new selection strategy to assist individuals in enhancing their ability to escape local optima. This selection strategy consists of two main steps: (1) determining individual stagnation using our proposed state definition and (2), if stagnation is detected, applying the appropriate survivor selection strategy based on the individual's population. In this way, we hope to improve the performance of the DE algorithm when facing local optima and to allow better cooperation with the mutation strategy.

This paper's primary contributions are outlined as follows:

 In this study, a hierarchical mutation strategy is proposed for the dual-population differential evolution algorithm. Specifically, the population is divided into elite and normal subpopulations based on the fitness values. The elite subpopulation focuses on optimizing solutions, while the normal subpopulation is dedicated to exploring the search space. By adopting this hierarchical mutation strategy, effective information exchange between subpopulations is facilitated, thereby enhancing the overall performance of the algorithm.

- 2. An innovative selection strategy is proposed in this paper. Firstly, a new criterion is introduced to identify whether an individual has fallen into a local optimum. Subsequently, for individuals in stagnation, survivors in the selection strategy are not limited to only those with superior fitness, but may include different individuals. This approach enhances the performance of the dual-population differential evolution algorithm when facing local optima.
- 3. Comprehensive evaluations were performed using the CEC 2017 and CEC 2011 benchmark suites. The results indicated that our proposed DPDE algorithm exhibited superior performance in comparison to several state-of-the-art algorithms.

The rest of this paper is structured as follows. Section 2 introduces the traditional DE algorithm. Section 3 details our proposed algorithm, DPDE. In Section 4, we present the comparative results of DPDE with other algorithms on the CEC 2017 and CEC 2011 test suites, along with an analysis of the parameter sensitivity and the method's effectiveness. Finally, Section 5 provides a summary of the paper.

#### 2. Basic Differential Evolution

In this section, we will provide a detailed introduction to the structure of the differential evolution algorithm, mainly through the four operations of initialization, mutation, crossover, and selection. The minimization problem considered in this article is as follows:

$$\min(f(x_i)), \quad x_i \in S \tag{1}$$

In this formula, f(.) denotes the value of the objective function and x stands for the individuals of the population. The search space S is defined as

$$S = \prod_{j=1}^{D} [X_{j,\min}, X_{j,\max}]$$
<sup>(2)</sup>

where  $X_{j,\min}$  and  $X_{j,\max}$  correspond to the lower and upper bounds of the *j*-th dimension of the individual *X*, respectively. *D* represents the maximum dimension of the individual.

#### 2.1. Initialization

The initialization operation is used to generate a population *pop* consisting of *NP* individuals,  $pop = (X_1, X_2, ..., X_{NP})$ . The operation for initializing population individuals is as follows:

$$X_{i,j} = X_{j,\min} + \operatorname{rand}(0,1) \times (X_{j,\max} - X_{j,\min})$$
(3)

where rand (0, 1) is a random number uniformly distributed in the range [0, 1] and  $X_{i,j}$  represents the *j*-th dimension of individual *i*.

#### 2.2. Mutation

In the *g*-th generation, the mutation strategy is used to generate a mutant vector  $V_i^g$  based on the target vector  $X_i^g$ . The following are six widely used mutation strategies:

1. DE/rand/1:

$$V_{i}^{g} = X_{r1}^{g} + F \times \left(X_{r2}^{g} - X_{r3}^{g}\right)$$
(4)

2. DE/rand/2:

$$V_{i}^{g} = X_{r1}^{g} + F \times \left(X_{r2}^{g} - X_{r3}^{g}\right) + F \times \left(X_{r4}^{g} - X_{r5}^{g}\right)$$
(5)

3. DE/best/1:

$$V_i^g = X_{best}^g + F \times \left(X_{r1}^g - X_{r2}^g\right) \tag{6}$$

4. DE/best/2:

$$V_i^g = X_{best}^g + F \times \left(X_{r1}^g - X_{r2}^g\right) + F \times \left(X_{r3}^g - X_{r4}^g\right) \tag{7}$$

5. DE/current-to-rand/1:

$$V_i^g = X_i^g + F \times \left( X_{r1}^g - X_{r2}^g \right) + F \times \left( X_{r3}^g - X_{r4}^g \right)$$
(8)

6. DE/current-to-pbest/1:

$$V_i^g = X_i^g + F \times \left(X_{pbest}^g - X_i^g\right) + F \times \left(X_{r1}^g - \overline{X_{ra}^g}\right)$$
(9)

Among them,  $X_{\text{best}}^{g}$  represents the best individual in the *g*-th generation of the population,  $X_{\text{pbest}}^{g}$  represents a randomly selected individual from the top 100*p*% individuals with the best fitness in the *g*-th generation of the population, *ra* is a randomly generated integer from the set {1, 2, . . . , NP + |A|}, *A* is an archive that stores the parent vectors discarded in the selection strategy,  $\bar{X}^{g}$  is the collection of the population and the archive *A*, the subscripts  $r_1, r_2, r_3, r_4$ , and  $r_5$  are randomly generated integers from the set {1, 2, . . . , NP}, and the subscripts are different from each other.

#### 2.3. Crossover

After the mutation operation, the obtained mutation vector and the target vector are crossed to obtain a trial vector. The definitions of commonly used crossover operations are as follows:

$$U_{i,j}^{g} = \begin{cases} V_{i,j}^{g} & \text{if } \operatorname{rand}(0,1) \le CR \text{ or } j = j_{\operatorname{rand}} \\ X_{i,j}^{g} & \text{otherwise} \end{cases}$$
(10)

where CR represents the crossover rate, rand (0, 1) is a random number uniformly distributed in the range [0, 1],  $j_{rand}$  is a random integer randomly generated from the range [1, D], and D is the maximum value of the individual's dimension.

#### 2.4. Selection

Following the crossover process, the selection mechanism decides between the trial vector  $U_i^g$  and the associated target vector  $X_i^g$ , with the superior of the two progressing to the subsequent generation. The formula based on the greedy survivor selection strategy is as follows:

$$X_i^{g+1} = \begin{cases} X_i^g & \text{if } f(X_i^g) < f(U_i^g) \\ U_i^g & \text{otherwise} \end{cases}$$
(11)

# 3. Dual-Population Differential Evolution Based on Hierarchical Mutation and Selection Strategy

In differential evolution algorithms, achieving a balance between the exploration and exploitation of the population, as well as assisting individuals in escaping local optima pose significant challenges. However, traditional dual-population differential evolution algorithms often struggle to effectively address these issues. On the one hand, the selection of appropriate mutation strategies is a challenging task. On the other hand, when individuals become trapped in local optima, traditional greedy selection strategies often struggle to efficiently drive further population evolution.

To overcome these challenges, this paper initially divides the population into two subpopulations, then introduces a hierarchical mutation strategy to balance the exploration and exploitation capabilities of the population. A new hierarchical selection strategy is introduced to assist the dual-population differential evolution algorithm to better address the local optima issues. These two strategies work in tandem to enhance the overall performance of the population. Ultimately, based on these strategies, the paper proposes the DPDE algorithm.

#### 3.1. Dual-Population Strategy

The dual-population differential algorithm has been extensively studied by researchers in recent years. It can adopt different mutation strategies in various populations, thereby facilitating the maintenance of good exploration and exploitation capabilities within the populations. In this paper, we employed fitness-based grouping of the population, where individuals ranking within the top bp% are classified as the elite population (ep), while the remaining individuals are categorized as the normal population (np). Notably, the ep focuses more on refining solution quality, whereas the np specializes in exploring potential solution spaces. We introduce a novel hierarchical mutation strategy aimed at maximizing information exchange between these two subpopulations, enhancing search capabilities and preventing premature convergence. As illustrated in Figure 1, individuals within the ep (depicted by red circles) represent the top-performing fraction, tasked with uncovering superior solutions. In contrast, the np, guided by the ep, explores areas with untapped potential within the solution space. This approach leverages the information from the ep to guide the evolution of the np, thus promoting a balanced exploration-exploitation strategy. Furthermore, we incorporated a novel hierarchical selection strategy designed to improve the individual's ability to escape local optima. Detailed explanations of the hierarchical mutation and selection strategies will follow in subsequent sections. At the end of each generation, individuals in these two subpopulations are reclassified based on their fitness values, facilitating information exchange and ensuring the leadership of the ep.



Figure 1. The framework of DPDE.

The value of the bp influences the size of the np and ep, which, in turn, affects the algorithm's performance. We designed a larger bp value in the early stage, benefiting the normal population in exploring more potential regions and, thus, boosting the exploration capability. Later, we reduced the bp value, allowing both populations to learn from the top-performing individuals in the entire population, thereby enhancing the algorithm's convergence capability. The formula for the bp is as follows:

$$bp = 0.29 \times \left(1 - \frac{nfes}{maxnfes}\right) + 0.11 \tag{12}$$

Here, *nfes* represents the current number of function evaluations and *maxnfes* represents the maximum number of function evaluations.

#### 3.2. Hierarchical Mutation Strategy

In this section, we will introduce the hierarchical mutation strategy used in this paper. The hierarchical mutation strategy consists primarily of two mutation strategies, namely DE/current-to-pbest/1 and improved DE/current-to-pbest/1, employed for the elite population and the normal population, respectively.

For the elite population: We adopted the DE/current-to-pbest/1 mutation strategy from JADE. This strategy selects the top 100p% individuals from the population and utilizes an external archive A to maintain diversity. This approach possesses an impressive convergence rate. Through this strategy, we aimed for the individuals to search within the best regions of the population, thereby finding improved solutions.

$$V_{i}^{g} = X_{i}^{g} + F \times (X_{nhest}^{g} - X_{i}^{g}) + F \times (X_{r1}^{g} - \overline{X_{ra}^{g}})$$
(13)

where  $X_{pbest}^g$  represents an individual randomly selected from the 100*p*% best-fitness individuals of the population in the g-th generation. ra is a randomly generated integer from the set  $\{1, 2, ..., NP + |A|\}$ . *A* is an archive storing the parent vectors that are eliminated in the selection strategy, and  $\overline{X^g}$  is the union of the population and the archive *A*.

For the normal population: We employed the improved DE/current-to-pbest/1. This strategy improves upon DE/current-to-pbest/1 by changing the selection of the top 100p% individuals in the population to the selection from the elite population. Through this strategy, we allow the less-proficient individuals in the population to accept guidance from the superior individuals in the elite population, thereby exploring more-promising regions, enhancing the algorithm's exploration capabilities, while maintaining a certain degree of convergence.

$$V_{i}^{g} = X_{i}^{g} + F \times (X_{ev1}^{g} - X_{i}^{g}) + F \times (X_{r1}^{g} - X_{ra}^{g})$$
(14)

where  $X_{ep1}$  is an individual randomly selected from the elite population.

#### 3.3. Hierarchical Selection Strategy

As mentioned earlier, we propose a dual-population DE algorithm. This approach achieves a balance between exploration and exploitation through multiple mutation strategies. However, during the evolutionary process, individuals often get stuck in local optima. Traditional greedy selection strategies struggle to provide an effective solution. Therefore, to address this issue more effectively, we designed a new selection strategy. This strategy consists of two steps: (1) determining if an individual has fallen into a local optimum; (2) deciding on the selection strategy based on the population to which the individual belongs. We will now provide a detailed explanation of these two steps.

First phase: Determining whether an individual is trapped in a local optimum is challenging. While many studies lean towards a simplified approach, i.e., considering that the individual has stagnated when its fitness has not improved over a certain number of generations, this method overlooks the probability of an individual getting trapped in local optima at different stages of evolution. In the early stages of evolution, individuals with poorer fitness are more likely to improve. If there is no significant improvement in fitness over a long time, it is highly probable that they have fallen into a local optimum. In contrast, in the later stages of evolution, individuals often focus on convergence, and improving fitness becomes challenging. Therefore, a more-stringent stagnation judgment criterion is required. To make a more-accurate judgment, we propose a new formula for determining individual stagnation based on the iteration phase, as shown in Equation (15).

$$T = \begin{cases} T1 & \text{if } nfes \le 0.5 \times \text{maxnfes} \\ T1 + \frac{nfes - 0.5 \times \text{maxnfes}}{0.5 \times \text{maxnfes}} (T2 - T1) & \text{if } nfes > 0.5 \times \text{maxnfes} \end{cases}$$
(15)

In the formula, *T* represents the threshold used for individual stagnation judgment. *T*1 and *T*2 represent the minimum and maximum values of the threshold, respectively. *nfes* 

8 of 22

denotes the current iteration number, while maxnfes stands for the maximum number of iterations. Specific settings will be discussed in detail in the experimental section. As the formula suggests, in the early stages of evolution, we set a relatively small fixed stagnation judgment value for the population. It is noteworthy that this value should not be set too low to reduce misjudgments. In the later stages of evolution, we employed a linearly increasing stagnation judgment, adapting to the increased difficulty in judging stagnation during the evolutionary process and helping the population converge better.

Second phase: When an individual falls into a local optimum, we adopted a targeted selection strategy based on the population's characteristics.

For individuals in the normal population that fall into a local optimum, our first approach was to accept the suboptimal trial individual they generate. If they remain stagnant after accepting this for T3 generations, we concluded that the current individual is in a difficult-to-improve area. At this time, we randomly selected the best discarded trial vector from the historical data of an individual in the elite population. This approach has two advantages. First, it reduces computational resource consumption due to stagnated individuals in the normal population. Second, by incorporating a high-quality trial vector with a good fitness value, we gain insights from superior regions. By employing this strategy, we can better explore potentially promising regions. As illustrated in Figure 2, when individual A from the normal population falls into a local optimum, we first accept the inferior trial vector C, with a certain probability of disturbing to vector D, thereby escaping the local optimum. When the perturbation range of A is small and we accept experimental individual B, it is still difficult to get out of the local optimum. If after T3 generations, the fitness of the individual still has not improved, we then accept the unused optimal discarded vector C from the history of the elite population. We hope to further explore promising regions based on C and assist individual A to escape the local optimum.

Stagnant individuals in the elite population already possess relatively good fitness values, so we focused more on whether the surrounding target landscape can help them escape local optima. Therefore, we selected the discarded trial individuals generated in the current generation. Through this method, we expected to better develop locally. With the HS strategy, the population's exploration and exploitation abilities are further enhanced. It is worth noting that the best individuals still use a greedy selection strategy to improve the algorithm's stability.



Figure 2. Illustration of how the selection strategy helps to escaping from local optima.

#### 3.4. Control Parameter

The efficiency of the differential evolution algorithm is significantly affected by its parameters. For the parameter settings, Success-History Based Parameter Adaptation for Differential Evolution (SHADE) has achieved widespread success. In this paper, we also choose it for the parameter setting. The SHADE method maintains two historical memory

sets,  $S_{CR}$  and  $S_F$ , which store the crossover rate CR and the scaling factor F values used during successful updates, respectively. In the *G*-th generation, the *F* and *CR* values for each individual in the population are generated using the following formulae:

$$F_{i,G} = \operatorname{randc}(uF_r, 0.1) \tag{16}$$

$$CR_{i,G} = \operatorname{randn}(uCR_r, 0.1) \tag{17}$$

where randc represents the Cauchy distribution and randn represents the normal distribution.  $uF_r$  and  $uCR_r$  are randomly selected from  $M_F$  and  $M_{CR}$ , respectively, where r = 1, ...,H. H is the pre-defined size of the set. Initially,  $M_F$  and  $M_{CR}$  are both set to 0.5 and are updated at the end of each generation. The formulas for  $M_F$  and  $M_{CR}$  are as follows:

$$w_k = \frac{\Delta f_k}{\sum_{k=1}^{|S_{CR}|} \Delta f_k} \tag{18}$$

$$\begin{cases} M_{CR} = \text{mean}_{WL}(S_{CR}) \\ \text{mean}_{WL}(S_{CR}) = \frac{\sum_{k=1}^{|S_{CR}|} W_k \times S_{CR,k}^2}{\sum_{k=1}^{|S_{CR}|} W_k \times S_{CR,k}} \end{cases}$$
(19)

$$\begin{cases} M_F = \text{mean}_{WL}(S_F) \\ \text{mean}_{WL}(S_F) = \frac{\sum_{k=1}^{|S_F|} W_k \times S_{F,k}^2}{\sum_{k=1}^{|S_F|} W_k \times S_{F,k}} \end{cases}$$
(20)

where  $\Delta f_k = |f(U_{k,G}) - f(X_{k,G})|$ , represents the improvement magnitude of the fitness value for individual *k* that was successfully updated in the *G*-th generation. This method effectively achieves the adaptive adjustment of the parameters. |S| denotes the size of the set. In this paper, *F* will be truncated to 0.6 in the early stages and extended to the range [0–1] in the later stages.

Reducing the population size during the iteration process can enhance the performance of the algorithm. In this paper, we adopted the Linear Population Size Reduction strategy (LPSR). Under this strategy, we used a linear formula to adjust the population size in each generation. The specific formula for updating the population size is as follows:

$$N_{G+1} = \text{round}\left[\frac{(N^{\min} - N^{\min})}{\max \text{nfes}} \times \text{nfes} + N^{\min}\right]$$
(21)

where  $N_{\text{init}}$  represents the initial population size and  $N_{\text{min}}$  represents the predetermined minimum population size. At the end of each generation, the  $(N_G - N_{G+1})$  individuals with the worst fitness in the population will be removed.

Based on the above discussion, the pseudocode for DPDE is shown as Algorithm 1.

Al	gori	ithm	1	DPDI	E al	lgorit	hm.
----	------	------	---	------	------	--------	-----

- 1: Set NP =  $18 \cdot D$ ,  $N_{\min} = 4$ , and initialize T1, T2, T3
- 2: Initialize  $p_0$  randomly according to Equation (3),  $p_0 = X_1, X_2, \dots, X_{NP}$
- 3: Set  $M_F = 0.5$ ,  $M_{CR} = 0.5$ ,  $A = \emptyset$ , nfes = *NP*, and set all values in the count equal to 0, H = 5

4: while nfes < maxnfes do

- 5:  $S_F = \emptyset, S_{CR} = \emptyset, CR = \emptyset, F = \emptyset$
- 6: **for** i = 1 to NP **do**
- 7: r = randomly select from [1, H]
- 8: **if**  $M_{CR}^r < 0$  **then**
- 9:  $CR_i = 0$
- 10: **else**

Algorithm 1 Cont.  $CR_i = N(M_{CR}^r, 0.1)$ 11: 12: end if  $F_i = \mathcal{C}(M_{F,r}, 0.1)$ 13: if nfes  $< 0.6 \cdot \text{maxnfes}$  and  $F_i \ge 0.6$  then 14:  $F_i = 0.6$ 15: end if 16: Calculate bp according to Equation (12)17: Calculate T according to Equation (15) 18: if  $i \leq bp \cdot NP$  then 19: Compute  $V_i^g$  by mutation Equation (13) 20: else 21: Compute  $V_i^g$  by mutation Equation (14) 22: 23: end if Compute  $U_i^g$  by crossover Equation (10) 24: end for 25: Sort the population by the ascending fitness values 26: 27: for i = 1 to NP do if  $f(U_i^g) < f(x_i^g)$  then 28:  $x_i^{g+1} = u_i^g, count(i) = 0, flag(i) = 0$  $A \leftarrow x_i^g, S_F \leftarrow F_i, CR_i \leftarrow S_{CR}$ 29: 30: else 31: count(i) = count(i) + 132: if  $i < bp \cdot NP$  and  $count(i) \ge T$  then 33:  $x_i^{g+1} = U_i^g$ 34: else if  $i \ge bp \cdot NP$  and  $count(i) \ge T$  then 35: if flag(i) == 0 then 36:  $x_i^{g+1} = U_i^g$ 37: else 38: flag(i) = flag(i) + 139: end if 40: if  $flag(i) \ge T3$  then 41:  $x_i^{g+1}$  = the historical best discarded vector that individual i has never 42: used end if 43: end if 44: end if 45: Update A if necessary 46: 47: end for Calculate NP<sub>new</sub> according to Equation (21) 48: 49:  $NP = NP_{new}$ if  $S_F \neq \emptyset$  then 50: Calculate  $W_k$  according to Equation (18) 51: Update  $M_{F,k}$  and  $M_{CR,k}$  according to Equations (19) and (20) 52: 53: end if nfes = nfes + NP54: 55: end while

#### 4. Experiments' Analysis and Comparison

In this section, we primarily discuss the performance of the proposed DPDE on the CEC 2017 test set [33], comparing it with some of the advanced DE algorithms. Further, this paper verifies the effectiveness of the new selection strategy proposed. Simultaneously, a detailed discussion on some parameter settings involved in the method is conducted. The test dimensions in this paper range from 10, 30, 50, to 100, with a search range between [-100, 100]. For each function, the optimal value is known. Each algorithm was

run independently 51 times, recording the mean and standard deviation of the fitness error value  $f(X_{best}) - f(X_*)$ . Here,  $X_{best}$  represents the optimal solution achieved by the algorithm in a single run, while  $X_*$  denotes the true global optimum of each function. When the fitness error value is less than  $10^{-8}$ , it is considered as 0. For the CEC2011 test set [34], the maximum number of evaluations in this paper was set to 150,000, and each algorithm was run 25 times on each instance. Moreover, to better assess the algorithm's performance, this study conducted non-parametric statistical tests at a 5% significance level, namely the Wilcoxon signed-rank test and the Friedman test.

The programming language used was MATLAB, compiled with MATLAB 2021. The experiments were executed on a system powered by an AMD Ryzen 7 5800H with Radeon Graphics 3.20 GHz and equipped with 16 GB RAM.

#### 4.1. Experimental Setup

In this subsection, we delve into the specifics of the DPDE algorithm on the CEC 2017 test set. The CEC 2017 test suite includes unimodal functions (f1–f3), simple multi-modal functions (f4–f10), hybrid functions (f11–f20), and composition functions (f21–f30). The symbols "+", "-", and "=" are used to indicate whether DPDE performed better, worse, or similarly, respectively, compared to the corresponding algorithm. The algorithms taken for comparison were JADE, LSHADE, jSO [35], MEPDE [36], HIP-DE [37], and PaDE [38]. JADE is a classic differential evolution algorithm. LSHADE was the champion of the CEC 2014 real-parameter single-objective optimization competition. jSO was the top performer in the CEC 2017 competition. MEPDE is a classic multi-population differential evolution algorithms. The specific parameter settings are displayed in Table 1. For DPDE, some parameters were set as follows: T1 was 48 for 10D and 24 for 30D, 50D, and 100D. T2 was 208 for 10D, 30 D, and 50D, while it was 160 for 100D. T3 was consistently set to 16.

Algorithm	Year	Parameters' Initial Settings
MPEDE	2016	$N = 250, ng = 20, \lambda_1 = \lambda_2 = \lambda_3 = 0.2$
JADE	2009	$N = 100, u_F = 0.5, u_{CR} = 0.5, p = 0.05, c = 0.1$
LSHADE	2014	$N_{max} = 18 \times D, N_{min} = 4, H = 6, r^{arc} = 2.6, M_F = 0.5, M_{CR} = 0.5$
jSO	2017	$N_{max} = 25 \times log(D) \times sqrt(D), H = 5, M_F = 0.3, M_{CR} = 0.8, r^{arc} = 2.6, p_{min} = 0.125, p_{max} = 0.25$
PaDE	2019	$25 \times log(D) \times sqrt(D), u_F = 0.8, u_{CR} = 0.6, k = 4, p = 0.11, r^{arc} = 1.6$ $T_0 = 70$
HIP-DE	2021	$N_{max} = 15 \times D, N_{min} = 4, M_F = 0.6, M_{CR} = 0.8, \tau_F = \tau_{CR} = 0.9, K = 6,$
DPDE		$N_{max} = 18 \times D, N_{min} = 4, H = 6, r^{arc} = 2.6, M_F = 0.5, M_{CR} = 0.5$

Table 1. Parameter settings.

#### 4.2. Optimization Accuracy

To thoroughly validate the DPDE algorithm's effectiveness, this study conducted an extensive analysis using the CEC2017 test suite. The results are showcased in Tables A1–A4. In these tables, the lowest average error values for each function are highlighted in bold. For 10D, among the 30 benchmarks, our algorithm performed significantly better than HIP-DE for 11 functions, outperformed PaDE for 8, surpassed jSO for 12, was superior to LSHADE for 10, exceeded JADE for 14, and outshone MPEDE for 19. Our approach achieved the best performance for the functions f5, f7, f10, f17, and f29. For 30D, among the 30 benchmarks, our algorithm outperformed HIP-DE for 12 functions, PaDE for 12, jSO for 15, LSHADE for 17, JADE for 21, and MPEDE for 17. Our method secured the best results for functions f5, f7, f8, f10, f16, f17, f21, and f29. For 50D, out of 30 benchmarks, our technique surpasses HIP-DE in 17 instances, PaDE in 15, jSO in 14, LSHADE in 16, JADE in 26, and MPEDE in 23. Notably, our method achieved optimal performance for functions

f5, f6, f7, f8, f10, f16, f17, f20, f21, f26, and f29. For 100D, among the 30 benchmarks, our algorithm stood out by besting HIP-DE on 16 functions, PaDE on 21, jSO on 14, LSHADE on 19, JADE on 22, and MPEDE on 23. In this dimension, our technique excelled for functions f5, f7, f8, f10, f16, f17, f20, f21, f22, f23, f24, f26, f28, and f29.

As depicted in the Table 2, based on the CEC 2017 benchmark functions, the table lists the rankings of different algorithms under the Friedman test. A lower ranking denotes better algorithmic performance. The top-ranked algorithm is emphasized in bold. It can be observed that, for 10 dimensions, the PaDE algorithm ranked first, while our proposed technique came in a close second. In the 30D, 50D, and 100D settings, our method clinched the top ranking. This underscores the observation that, as the dimensionality increased, the performance of our algorithm became more pronounced.

**Table 2.** Average rankings between DPDE and the other algorithms according to the Friedman test at the 0.05 significance level.

Algorithm	10D	30D	50D	100D
DPDE	3.2333	2.7667	2.5000	2.5000
HIP-DE	3.4500	3.4667	3.7333	3.8000
PaDE	3.1500	3.1833	3.6000	3.7333
jSO	3.5333	3.7333	2.9333	2.7667
LSHADE	3.9000	4.2000	3.2667	3.6000
JADE	5.7167	5.8000	6.2000	5.9000
MPEDE	5.0167	4.8500	5.7667	5.7000

The top-ranked algorithm is emphasized in bold.

#### 4.3. Comparison on Convergence Speed

To better analyze the convergence characteristics of DPDE, this paper presents the convergence curves of the algorithm for functions f5, f10, f21, and f24 across the dimensions 10D, 30D, 50D, and 100D. As shown in Figures 3–6, for the function f24, the convergence curve of our algorithm was fairly comparable to the other algorithms. However, for functions f5, f10, and f21, it can be observed that our algorithm did not converge as quickly during the early iterations. Yet, in the later stages, it managed to converge to a value better than the other algorithms. This indicates that our method can identify promising regions and converge to more-optimal solutions, reflecting its robust exploration and exploitation capabilities.



Figure 3. Convergence curves of the mean fitness on certain test functions for 10D.



Figure 4. Convergence curves of the mean fitness on certain test functions in 30D.



Figure 5. Convergence curves of the mean fitness on certain test functions in 50D.



Figure 6. Convergence curves of the mean fitness on certain test functions in 100D.

#### 4.4. Effectiveness of HS Strategy and New Stagnation Detection Formula

This paper introduces a new selection strategy (HS) and a new stagnation detection formula. To demonstrate the effectiveness of these two strategies, ablation experiments were conducted. As shown in Table 3, DPDE-1 represents the DPDE algorithm without the new selection strategy. The ranking results of the two algorithms after the Friedman test are presented in the figure. The DPDE algorithm showed significant improvements across all four dimensions, with the smallest improvement in the 10-dimensional space. Table 4 illustrates the validity of the new stagnation detection formula. We set up three comparative experiment groups, namely DPDE-fixed24, DPDE-fixed48, and DPDE-fixed96, representing the use of a fixed number of 24, 48, and 96 consecutive non-updated individuals to determine if an individual is stagnant. Our method outperformed the DPDE approach using fixed counts across all four dimensions. Among them, the performance gap was the smallest between DPDE-fixed96 and DPDE in 10D and between DPDE-fixed24 and DPDE in 30D, 50D, and 100D. Therefore, it was deduced that the novel HS strategy and the formula for detecting stagnation contributed positively to improving the differential evolution algorithm's performance.

Table 3. Comparison of average ranks between DPDE and DPDE-1 using the Friedman test.

	10D	30D	50D	100D	Average
DPDE	1.3333	1.1333	1.2000	1.3000	1.2417
DPDE-1	1.6667	1.8667	1.8000	1.7000	1.7583
The ten replied alo	anithma is smarphasi	and in hold			

The top-ranked algorithm is emphasized in bold.

Table 4. Comparison of average ranks between DPDE and its variants using the Friedman test.

	10D	30D	50D	100D	Average
DPDE	2.2167	2.0167	2.0000	1.9667	2.0500
DPDE-fixed24	2.7667	2.3167	2.2000	2.3667	2.4125
DPDE-fixed48	2.6500	2.6167	2.5333	2.7667	2.6417
DPDE-fixed96	2.3667	3.0500	3.2667	2.9000	2.8958

#### 4.5. Parameter Analysis

The settings of T1 and T2 play a significant role in determining whether an individual is stagnant. Specifically, T1 and T2 represent the minimum and maximum values for the stagnation threshold, respectively. To investigate the impact of the settings of T1 and T2 on the algorithm performance, three values were set for each: T1 at 24, 48, and 96 and T2 at 160, 208, and 256. Combining these yielded nine variants, the ranking results based on the Friedman test are displayed in Table 5. It was revealed that that DPDE-T5 performed best in 10D, DPDE-T2 in 30D and 50D, and the DPDE-T1 setting in 100D. T3 played a crucial role in helping individuals stuck in local optima in the normal population. It represents the maximum stagnant count after which an individual can accept a worse trial solution. In our experiments, we tested T3 values of 8, 16, and 32. The results in Table 6 show that T3 = 8 performed best in 10D, while T3 = 16 was optimal in 30D, 50D, and 100D. The chosen value for T3 in this paper was 16.

Table 5. Comparison of average ranks between DPDE and its variants using the Friedman test.

	10D	30D	50D	100D
DPDE-T1	5.7500	3.9500	3.6333	3.0833
DPDE-T2	5.2500	3.6667	3.5333	4.4500
DPDE-T3	5.7500	4.1000	3.7667	3.3167
DPDE-T4	4.3833	4.1833	5.6667	5.3833
DPDE-T5	4.3000	4.5333	4.8000	4.5500
DPDE-T6	4.7833	4.8667	4.6000	5.1000
DPDE-T7	5.0833	6.6667	6.2667	6.2333
DPDE-T8	4.8833	6.2500	6.1333	6.8667
DPDE-T9	4.8167	6.7833	6.6000	6.0167

The top-ranked algorithm is emphasized in bold.

**Table 6.** Comparison of average ranks between DPDE and its variants using the Friedman test under different values of T3.

	10D	30D	50D	100D	Average
DPDE_8	1.9333	2.1000	2.0667	1.9667	2.0167
DPDE_16	1.9500	1.8667	1.8667	1.8667	1.8875
DPDE_32	2.1167	2.0333	2.0667	2.1667	2.0958

The top-ranked algorithm is emphasized in bold.

#### 4.6. Comparison Results for the Real-World Problems on CEC 2011

To validate the effectiveness of the algorithm proposed in this paper in real-world problems, we selected four bounded problems for testing: problem 1, problem 5, problem 6, and problem 7. The dimensions of these problems are 30, 30, 30, and 20, respectively. As shown in Table 7, it can be observed that the DPDE algorithm significantly outperformed or was at least comparable to six of the most-advanced DE variants. Notably, our method outshone all other approaches for problem 7. Moreover, JADE, PaDE, HIP-DE, MPEDE, and jSO performed significantly worse than the DPDE algorithm on half of the problems. The experimental results suggested that the DPDE algorithm had potential for solving real-world problems.

Problem	MPEDE	JADE	LSHADE	jSO	PaDE	HIP-DE	DPDE
problem 2	-2.17E+01 (+)	-2.33E+01 (+)	-2.62E+01 (=)	-2.59E+01 (+)	-2.62E+01 (=)	-2.64E+01 (=)	-2.64E+01
problem 5	-3.41E+01 (+)	-3.59E+01 (=)	-3.62E+01 (=)	-3.57E+01 (=)	-3.61E+01 (=)	-3.61E+01 (=)	-3.59E+01
problem 6	-2.82E+01 (=)	-2.90E+01 (=)	-2.92E+01 (-)	-2.91É+01 (-)	-2.60E+01 (+)	-2.11E+01 (+)	-2.83E+01
problem 7	1.29E+00 (+)	1.17E+00 (+)	1.15E+00 (+)	1.14E+00 (+)	1.08E+00 (+)	1.13E+00 (+)	7.43E-01
Total $+/-/=$	3/0/1	2/0/2	1/1/2	2/1/1	2/0/2	2/0/2	

Table 7. Comparison results for the real-world problems.

The top-ranked algorithm is emphasized in bold.

#### 5. Conclusions

In this paper, we introduced the application of hierarchical mutation and selection strategies to the dual-population differential evolution algorithm. This aimed to address the challenge of balancing exploration and exploitation in the algorithm and mitigating the issue of evolution stagnation. As a result, we proposed the DPDE algorithm. In the hierarchical mutation strategy, the DE/current-to-pbest/1 and improved DE/current-pbest/1 strategies were applied separately to the elite population and the normal population. Due to the guiding role of the elite population in the mutation strategy, we facilitated information exchange between these two subpopulations, further achieving a balance between exploration and exploitation. In the novel hierarchical selection strategy, when individuals were trapped in local optima, elite subpopulation individuals accepted discarded vectors from the current generation, while the normal subpopulation individuals accepted both discarded vectors and historical optimal discarded vectors from the elite subpopulation. This strategy makes full use of historically generated trial vectors, thus enhancing individuals' ability to escape local optima and improving the overall algorithm performance. Additionally, we employed adaptive parameter techniques and population size adjustment strategies to enhance the algorithm's exploration and exploitation capabilities at different stages.

Experimentally, we rigorously tested our approach on the CEC 2017 benchmark suite and compared it with six state-of-the-art differential evolution algorithms, namely HIP-DE, PaDE, jSO, LSHADE, JADE, and MPEDE. The experimental results indicated that the algorithm presented in this paper significantly outperformed the other compared differential evolution algorithms. The ablation study demonstrated the effectiveness of the HS strategy and the new stagnation detection formula introduced in this paper. The results on the CEC 2011 benchmark further demonstrated the potential of our method to solve real-world problems.

Lastly, while the algorithm in this paper primarily targets single-objective parameter problems, it will be meaningful to explore the feasibility of employing the DPDE algorithm for multi-objective optimization challenges in the future.

**Author Contributions:** Methodology: Y.H.; Validation: Y.H., X.Q. and W.S.; Writing—original draft preparation: Y.H.; Writing—review and editing: Y.H. and X.Q.; Project administration: X.Q. and W.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the National Natural Science Foundation of China (62076110) and the Natural Science Foundation of Jiangsu Province (BK20181341).

**Data Availability Statement:** The data used to support the findings of this study are available from the corresponding author upon request.

**Conflicts of Interest:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

### Appendix A

		DPDE	HIP-DE	PaDE	jSO	LSHADE	JADE	MPEDE
F1	Mean	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	1.46E-09
••	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	5.65E-09
F2	Mean	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	6.75E-10	0.00E+00
	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	4.82E-09	0.00E+00
F3	Mean	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F4	Mean	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	1.28E-08
	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	3.33E-08
F5	Mean	3.13E-01	1.68E+00	2.27E+00	2.11E+00	2.89E+00	3.76E+00	6.13E+00
E	Std	4.67E-01	8.78E-01	9.75E-01	7.87E-01	8.02E-01	1.07E+00	1.50E+00
FO	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	2.25E-05
F7	Mean	1 11F+01	1.18E+01	1.25E+01	1.21E+00	1.23E+01	1.35E+01	1.78E+01
17	Std	4.30E-01	5.40E-01	1.22E+00	4.84E-01	7.24E-01	1.85E+00	1.82E+00
F8	Mean	3.71E-01	1.93E+00	2.36E+00	2.28E+00	2.54E+00	3.81E+00	6.41E+00
	Std	5.96E-01	8.99E-01	7.95E-01	7.78E-01	1.04E+00	1.04E+00	1.85E+00
F9	Mean	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F10	Mean	1.18E+00	3.86E+01	2.77E+01	2.89E+01	1.91E+01	1.34E+02	2.74E+02
	Std	1.63E+00	7.55E+01	4.38E+01	4.90E+01	3.34E+01	1.07E+02	1.08E+02
F11	Mean	0.00E+00	4.54E-01	5.29E-01	0.00E+00	5.37E-01	1.50E+00	2.25E+00
<b>E10</b>	Std	0.00E+00	7.77E-01	7.62E-01	0.00E+00	7.46E-01	1.10E+00	6.39E-01
F12	Mean	1.19E+02	2.67E+01	2.3/E+01	1.45E+01	5.22E+01	8.14E+02	1.10E+01
E12	Sta	6.49E+01	5.03E+01	4.79E+01	3.89E+01	7.07E+01	3.24E+03	1.42E+01
F15	Std	4.20E+00 2.02E+00	$1.37 \pm +00$ 2 34 E + 00	1.92E+00 2.37E+00	4.34E+00 1 71E+00	4.40E+00 1 73E+00	4.70E+00 2.93E+00	3.36E+00 2.01E+00
F14	Mean	4 31F-01	7.41E-01	1 72E-01	9.75E-02	3.07E-01	2.95E+00 2.45E+00	4.32E+00
	Std	2.80E+00	7.13E-01	3.59E-01	2.99E-01	4.04E-01	5.96E+00	1.81E+00
F15	Mean	3.67E-01	1.68E-01	1.06E-01	2.21E-01	1.77E-01	2.84E-01	7.72E-01
	Std	1.75E-01	2.04E-01	1.67E-01	2.19E-01	1.96E-01	2.09E-01	2.29E-01
F16	Mean	6.96E-01	4.62E-01	3.18E-01	4.45E-01	3.57E-01	5.22E+00	2.35E+00
	Std	2.09E-01	2.03E-01	1.48E-01	3.80E-01	2.04E-01	2.33E+01	7.68E-01
F17	mean	1.30E-01	2.56E-01	1.62E-01	4.09E-01	1.64E-01	1.67E+00	7.15E+00
<b>F10</b>	Std	1.72E-01	2.90E-01	1.60E-01	3.66E-01	2.02E-01	5.48E+00	2.44E+00
F18	mean	7.64E-01	2.39E-01	2.49E-01	2.64E-01	1.88E-01	6.68E+00	2.75E+00
E10	Sta	2.81E+00 2.76E-02	2.08E-01 1.38E-02	2.05E-01 1.02E-02	2.07E-01 1.33E-02	1.95E-01 1.44E-02	9.38E+00 6.28E-02	1.50E+00 6.05E-01
119	Std	3 28E-02	2.67E-02	1.02E-02	1.35E-02 1.25E-02	1.44E-02 1.12E-02	2.93E-02	1.93E-01
F20	mean	2.75E-01	0.00E+00	0.00E+00	3.55E-01	6.12E-03	4.90E-01	1.05E+00
	Std	1.35E-01	0.00E+00	0.00E+00	1.08E-01	4.37E-02	2.79E+00	6.60E-01
F21	mean	1.38E+02	1.61E+02	1.31E+02	1.59E+02	1.58E+02	1.81E+02	1.21E+02
	Std	4.96E+01	5.01E+01	4.76E+01	5.57E+01	5.14E+01	4.46E+01	4.28E+01
F22	mean	1.00E+02	1.00E+02	1.00E+02	1.00E+02	1.00E+02	9.24E+01	9.41E+01
	Std	0.00E+00	6.43E-14	1.48E-13	1.48E-13	4.85E-02	2.64E+01	2.38E+01
F23	mean	3.03E+02	3.01E+02	3.02E+02	3.03E+02	3.04E+02	3.06E+02	3.06E+02
E24	Std	1.98E+00	1.43E+00	1.63E+00	1.63E+00	1.07E+00	2.05E+00	1.64E+00
Г24	Std	2.41E+02 1 13E+02	5.09E+02 6.08E+01	5.03E+02 6.80E±01	2.93E+02 8.49E±01	7.03E+02 7.31E+01	5.11E+02 7.03E+01	2.33E+02 1 10E+02
F25	mean	4.11E+02	4.19E+02	4 18E+02	4.03E+01	4.15E+02	4.23E+01	4.05E+02
1 20	Std	2.09E+01	2.30E+01	2.28E+01	1.49E+01	2.23E+01	2.33E+01	1.67E+01
F26	Mean	3.00E+02	3.00E+02	3.00E+02	3.00E+02	3.00E+02	3.05E+02	3.00E+02
	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	3.60E+01	1.52E-08
F27	Mean	3.89E+02	3.93E+02	3.93E+02	3.89E+02	3.89E+02	3.91E+02	3.89E+02
	Std	1.39E-01	1.74E+00	1.96E+00	1.67E-01	1.39E-01	7.65E+00	3.65E-01
F28	Mean	3.67E+02	3.56E+02	3.11E+02	3.76E+02	3.82E+02	4.35E+02	3.11E+02
<b>F2</b> 0	Std	1.24E+02	1.15E+02	5.56E+01	1.30E+02	1.35E+02	1.49E+02	5.56E+01
г29	iviean	2.31E+02 3.34E+00	2.33E+02 4.46E+00	2.32E+U2 3.66E+00	2.34E+U2 3.23E+00	2.35E+02 3.41E+00	2.33E+02 1.26E+01	2.31E+02 5.67E+00
E30	Mean	1.64E+04	4.01E+02	4.00E+02	4.85E+04	3.25E+04	1.06E+01	3.96E+02
100	Std	1.14E+05	1.67E+01	1.57E+01	1.94E+05	1.60E+05	2.94E+05	9.56E-01
	+		11	8	12	10	14	19
	-		9	11	4	6	3	7
	=		10	11	14	14	13	4

 Table A1. Mean errors of the algorithms under 10D of CEC 2017.

		DPDE	HIP-DE	PaDE	jSO	LSHADE	JADE	MPEDE
F1	Mean	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F2	Mean	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F3	Mean	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	4.74E+03	0.00E+00
	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F4	Mean	5.86E+01	5.63E+01	5.62E+01	5.86E+01	5.86E+01	5.08E+01	5.35E+01
	Std	2.72E-14	1.15E+01	1.44E+01	2.69E-14	2.72E-14	2.14E+01	1.77E+01
F5	Mean	2.95E+00	7.35E+00	8.05E+00	8.58E+00	6.84E+00	2.68E+01	2.74E+01
	Std	1.87E+00	1.18E+00	1.46E+00	1.91E+00	1.45E+00	4.60E+00	7.32E+00
F6	Mean	0.00E+00	8.75E-09	0.00E+00	4.62E-08	1.38E-09	0.00E+00	0.00E+00
	Std	0.00E+00	3.28E-08	0.00E+00	1.61E-07	6.88E-09	0.00E+00	0.00E+00
F7	Mean	3.48E+01	3.62E+01	3.87E+01	3.93E+01	3.80E+01	5.50E+01	5.62E+01
TO	Std	1.33E+00	9.40E-01	1.64E+00	2.29E+00	1.20E+00	3.53E+00	7.25E+00
F8	Mean	3.35E+00	7.42E+00	9.12E+00	9.55E+00	7.53E+00	2.55E+01	2.72E+01
БО	Sta	1.93E+00	1.23E+00	1.75E+00	1.90E+00	1.52E+00	4.04E+00	7.26E+00
Г9	Mean	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	6.91E-03	1.42E-02
E10	Moon	0.00E+00	1.52E + 02	0.00E+00 1 54E+02	$1.40E \pm 02$	1.44E + 02	0.30E-02	0.04E-02
1.10	Std	1.101+02 1.23E+02	1.52E+0.5 1.78E+0.2	1.34E+0.3 2 89E+0.2	$1.40\pm0.03$ 2.71E+02	1.44L+0.02	1.92E+03 2.01E+02	2.03E+03 3.58E+02
F11	Mean	$2.34E\pm02$	1.70E+02 1.42E+01	2.09E+02	$1.41E\pm01$	1.94E+02 1.76E+01	2.01E+02 3.12E+01	2 30E+02
111	Std	2.34E+01 2 74E+01	2.14E+01	1.27E+01 1 79E+01	2.20E+01	2.41E+01	2.54E+01	1 77E+01
F12	Mean	1.12E+03	1.14E+03	1.00E+03	3.13E+02	1.05E+03	1.25E+03	8.99E+02
	Std	3.73E+02	4.07E+02	3.95E+02	2.42E+02	3.44E+02	4.05E+02	4.01E+02
F13	Mean	1.48E+01	1.52E+01	1.39E+01	1.95E+01	1.85E+01	3.71E+01	2.17E+01
	Std	6.56E+00	7.09E+00	7.04E+00	3.59E+00	6.58E+00	1.55E+01	8.49E+00
F14	Mean	1.93E+01	2.14E+01	2.11E+01	2.16E+01	2.16E+01	5.22E+03	1.66E+01
	Std	4.74E+00	3.99E+00	4.79E+00	2.33E+00	3.16E+00	9.17E+03	1.05E+01
F15	Mean	2.85E+00	3.08E+00	3.08E+00	1.51E+00	3.86E+00	6.50E+02	8.86E+00
	Std	2.28E+00	1.74E+00	1.78E+00	1.15E+00	1.53E+00	2.11E+03	3.28E+00
F16	Mean	1.47E+01	1.16E+02	1.09E+02	4.05E+01	3.89E+01	4.12E+02	4.01E+02
	Std	2.08E+00	9.12E+01	9.92E+01	5.39E+01	3.82E+01	1.55E+02	1.76E+02
F17	Mean	2.19E+01	3.10E+01	2.90E+01	3.23E+01	3.28E+01	7.58E+01	5.38E+01
<b>T</b> 40	Std	5.86E+00	7.28E+00	7.00E+00	6.76E+00	7.11E+00	3.12E+01	1.46E+01
F18	Mean	2.20E+01	2.28E+01	2.14E+01	2.09E+01	2.22E+01	2.14E+04	2.24E+01
<b>E10</b>	Std	1.52E+00	1.64E+00	2.98E+00	4.16E-01	1.43E+00	5.90E+04	8.71E+00
F19	Mean	5.20E+00	5.08E+00	5.22E+00	5.36E+00	6.20E+00	1.39E+03	7.13E+00
E20	Moon	1.04E+00	2.03E+00 3.40E+01	1.90E+00	$1.70\pm+00$ 2.73E+01	1.96E+00 3.13E+01	$1.11E \pm 0.02$	7.94E+00 7.82E+01
120	Std	2.10L+01 3.99F±00	7.35E+00	$2.40E \pm 01$	6 30E±00	4.65E±00	$5.48E\pm01$	$5.66E \pm 01$
F21	Mean	2.05E+02	2.07E+02	2.401+01 2.08E+02	2 10F+02	2.08E+02	2.26E+01	2 28F+02
1 2 1	Std	1.95E+00	1.62E+00	1.43E+00	1.86E+00	1.58E+00	4 77E+00	7.39E+00
F22	Mean	1.00E+02	1.00E+02	1.00E+02	1.00E+02	1.00E+02	1.00E+02	1.00E+02
	Std	1.44E-14	1.44E-14	1.44E-14	6.39E-14	1.44E-14	1.44E-14	1.44E-14
F23	Mean	3.51E+02	3.44E+02	3.45E+02	3.57E+02	3.54E+02	3.73E+02	3.76E+02
	Std	4.69E+00	3.69E+00	3.68E+00	3.73E+00	3.66E+00	6.11E+00	1.02E+01
F24	Mean	4.25E+02	4.18E+02	4.20E+02	4.28E+02	4.26E+02	4.47E+02	4.49E+02
	Std	1.63E+00	1.98E+00	2.49E+00	2.74E+00	2.24E+00	4.84E+00	8.17E+00
F25	Mean	4.50E+02	4.48E+02	4.48E+02	4.49E+02	4.48E+02	4.58E+02	4.59E+02
	Std	1.44E-02	2.05E-02	2.74E-02	7.98E-03	2.46E-02	1.35E-01	1.32E+00
F26	Mean	4.74E+02	4.71E+02	4.72E+02	4.72E+02	4.71E+02	4.79E+02	4.80E+02
	Std	3.54E+01	4.06E+01	4.24E+01	3.94E+01	2.85E+01	6.98E+01	1.07E+02
F27	Mean	4.98E+02	4.94E+02	4.95E+02	4.95E+02	4.94E+02	5.00E+02	5.01E+02
TOC	Std	3.37E+00	4.99E+00	5.09E+00	5.59E+00	4.34E+00	7.49E+00	6.27E+00
F28	Mean	1.90E+03	1.91E+03	1.91E+03	1.90E+03	1.90E+03	1.97E+03	1.98E+03
EOO	Std	5.22E+01	4.90E+01	4.89E+01	4.36E+01	5.40E+01	5.51E+01	4.92E+01
F29	Mean	1.96E+03	1.96E+03	1.96E+03	1.96E+03	1.96E+03	2.02E+03	2.02E+03
E20	Sta	9.05E+00	0.29E+00 2.07E+02	1.06E+01 2.04E+02	1.40E+01 1.06E · 02	9.40E+00 2.00E+02	2./3E+UI	2.77E+01
1.20	Sta	2.01E+03 6.47E+01	2.07 E+03 7 30E+01	2.04E+03 5 22E→01	2 70E+03	2.00E+03 5.08E+01	2.30E+03	2.02E+03 8 67E + 01
	- -	0.47 11+01	12	12	2.79E+01 15	17	21	0.02E+01 17
	_		6	4	6	0	1	4
	=		12	14	9	13	8	9

Table A2. Mean errors of the algorithms under 30D of CEC 2017.

		DPDE	HIP-DE	PaDE	jSO	LSHADE	JADE	MPEDE
F1	Mean	0.00E+00						
	Std	0.00E+00						
F2	Mean	0.00E+00						
	Std	0.00E+00						
F3	Mean	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	4.31E+04	7.77E-04
	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	5.70E+04	3.38E-03
F4	Mean	6.82E+01	9.54E+01	7.51E+01	5.42E+01	6.69E+01	7.24E+01	5.97E+01
	Std	4.75E+01	4.79E+01	5.07E+01	4.97E+01	5.22E+01	4.79E+01	4.47E+01
F5	Mean	3.53E+00	1.48E+01	1.69E+01	1.58E+01	1.18E+01	8.11E+01	5.37E+01
	Std	1.66E+00	2.01E+00	1.98E+00	3.39E+00	2.28E+00	8.11E+00	1.21E+01
F6	Mean	9.64E-09	6.52E-08	8.41E-04	5.07E-07	3.31E-05	5.59E-06	6.97E-04
	Std	3.03E-08	1.06E-07	2.54E-03	9.51E-07	2.36E-04	6.29E-06	2.09E-03
F7	Mean	6.05E+01	6.11E+01	6.47E+01	6.70E+01	6.35E+01	1.36E+02	1.07E+02
	Std	1.98E+00	1.67E+00	2.53E+00	3.19E+00	1.84E+00	7.67E+00	1.13E+01
F8	Mean	3.10E+00	1.53E+01	1.78E+01	1.55E+01	1.23E+01	8.24E+01	5.53E+01
	Std	1.81E+00	1.99E+00	2.43E+00	3.50E+00	2.25E+00	7.36E+00	1.23E+01
F9	Mean	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	5.51E-02	9.35E-01
	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	2.64E-01	8.92E-01
F10	Mean	7.18E+02	3.22E+03	3.09E+03	3.04E+03	3.20E+03	5.40E+03	4.85E+03
	Std	2.79E+02	2.46E+02	2.97E+02	4.46E+02	2.79E+02	2.57E+02	7.28E+02
F11	Mean	4.20E+01	5.03E+01	6.52E+01	2.87E+01	4.95E+01	1.32E+02	1.04E+02
<b>F10</b>	Std	7.73E+00	1.05E+01	1.28E+01	2.82E+00	9.93E+00	7.75E+01	2.51E+01
F12	Mean	2.61E+03	2.28E+03	2.23E+03	2.09E+03	2.37E+03	4.10E+03	9.81E+03
<b>F10</b>	Sta	6.31E+02	5.45E+02	4.12E+02	5.32E+02	5.96E+02	2.33E+03	7.02E+03
F13	Mean	6.04E+01	6.19E+01	5.91E+01	3.72E+01	5.66E+01	1.15E+02	9.52E+01
E14	Sta	3.23E+01	2.33E+01	2.29E+01	1.92E+01	2.81E+01	4.50E+01	5.51E+01
1.14	Std	2.47E+01	3.23E+01	3.01E+01	2.37E+01 2.11E+00	3.13E+01	0.55E+05	1.4E+01
F15	Moon	5.37E+00 5.35E+01	4.03E+00 5 76E+01	3.38E+00 4.14E+01	2.11E+00 2.20E+01	3.80E+00 4.90E±01	2.01E+04 1 54E+02	7.04L+01 7.83E+01
115	Std	1.68E±01	$1.58E\pm01$	4.14L+01 1 17E+01	2.59E+01	4.901+01 1 56E+01	7.93E±02	$4.32E\pm01$
F16	Mean	1.00E+01	3.99F+02	3.59E+02	4.02E+00	3 70F+02	1.00E+03	9.67E+02
110	Std	3.35E+01	1.08E+02	1.12E+02	1.02E+02 1.48E+02	1.35E+02	1.50E+00 1.58E+02	2.01E+02
F17	Mean	5 51E+01	3.07E+02	2.92E+02	2 55E+02	2 15E+02	7.98E+02	5.88E+02
1 17	Std	4.06E+01	7.49E+01	7.34E+01	9.18E+01	7.57E+01	1.43E+02	1.99E+02
F18	Mean	4.96E+01	5.57E+01	4.05E+01	2.54E+01	4.94E+01	1.20E+04	1.21E+02
	Std	2.24E+01	2.20E+01	1.10E+01	2.36E+00	1.81E+01	8.43E+04	9.31E+01
F19	Mean	4.69E+01	4.07E+01	2.75E+01	1.46E+01	3.48E+01	1.01E+02	4.44E+01
	Std	1.95E+01	1.35E+01	8.62E+00	2.56E+00	1.17E+01	3.20E+01	1.86E+01
F20	Mean	4.28E+01	1.80E+02	1.72E+02	1.17E+02	1.55E+02	6.25E+02	3.50E+02
	Std	5.64E+00	6.97E+01	7.29E+01	6.75E+01	5.55E+01	1.17E+02	1.74E+02
F21	Mean	2.05E+02	2.17E+02	2.18E+02	2.19E+02	2.14E+02	2.83E+02	2.53E+02
	Std	3.13E+00	1.92E+00	2.51E+00	2.75E+00	2.58E+00	8.76E+00	1.37E+01
F22	Mean	4.40E+02	1.00E+02	4.69E+02	2.23E+03	2.29E+03	4.31E+03	3.68E+03
	Std	2.51E+02	1.37E+00	1.11E+03	1.53E+03	1.67E+03	2.58E+03	2.67E+03
F23	Mean	4.28E+02	4.29E+02	4.28E+02	4.36E+02	4.32E+02	5.05E+02	4.81E+02
	Std	6.23E+00	6.68E+00	6.76E+00	5.57E+00	3.49E+00	1.28E+01	1.39E+01
F24	Mean	5.05E+02	5.07E+02	5.04E+02	5.19E+02	5.11E+02	5.55E+02	5.43E+02
	Std	2.68E+00	5.73E+00	5.25E+00	4.14E+00	3.17E+00	8.80E+00	1.34E+01
F25	Mean	4.82E+02	4.82E+02	4.98E+02	4.81E+02	4.82E+02	5.18E+02	4.98E+02
<b>TO</b> (	Sta	1.1/E+01	4.26E+00	2.95E+01	3.31E+00	3.77E+00	3.52E+01	3.44E+01
F26	Mean	1.08E+03	1.12E+03	1.12E+03	1.22E+03	1.19E+03	1.79E+03	1.56E+03
<b>T</b> 07	Sta	4.83E+01	5.92E+01	8.05E+01	4.36E+01	4.73E+01	1.0/E+02	1.68E+02
F27	Niean	5.37E+02	5.36E+02	5.40E+02	5.37E+02	5.35E+02	5.54E+02	5.43E+02
EJO	Sta	1.18E+U1 4.70E+02	1.55E+U1 4.01E+02	1.35E+01 4.00E+02	2.05E+01	1.11E+U1 4.64E+02	1.72E+01 4.85E+02	2.41E+01
1.70	Std	4.70E+02	4.71E+U2 5.14E+00	4.77E+U2 5.28E+00	4.37E+02	4.04E+02	4.00E+02	4.07E+U2
E20	Mean	3.00E+00 3.37E±02	3.14E+00 3.65E±02	3.20E+00 3.53E±02	1.37E+01 3.58E±02	3.49F+00	2.03E+01 5.12E±02	1.22E+01 4 39E±02
1.72	Std	3.07E±01	3 32E±01	3.35E+02 3.36E⊥01	0.00±+02 4 78E±01	3 30F±01	5.12E+02 5.40E+01	$4.591\pm02$ 6 45 E ± 01
E30	Mean	$6.46F \pm 05$	$6.19F \pm 05$	6 11 F±05	$6.41$ F $\pm 05$	657E+05	6 59F±05	6 80F±05
100	Std	5.102+0.05	$3.27E \pm 0.04$	3 88F±04	6.87F±04	7 53F+04	6.39F±04	9.69F±04
	+	J.7 4L TUH	17	15	14	16	26	23
	_		2	5	8	1	0	0
	=		11	10	8	13	4	7

Table A3. Mean errors of the algorithms under 50D of CEC 2017.

		DPDE	HIP-DE	PaDE	jSO	LSHADE	JADE	MPEDE
F1	Mean	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
••	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F2	Mean	3.38E+05	4.63E+01	6.85E+00	2.85E+04	4.08E+04	1.72E+03	9.32E+14
	Std	2.10E+06	1.30E+02	4.13E+01	1.27E+05	2.39E+05	1.23E+04	6.66E+15
F3	Mean	2.46E-04	1.65E-07	6.49E-08	3.83E-06	4.66E-06	2.08E+05	1.73E+01
	Std	6.27E-04	1.49E-07	1.02E-07	5.03E-06	6.55E-06	2.08E+05	5.49E+01
F4	Mean	1.99E+02	1.96E+02	1.30E+02	2.01E+02	1.98E+02	1.28E+02	8.84E+01
	Std	6.04E+00	5.40E+00	6.87E+01	9.20E+00	6.58E+00	5.71E+01	6.43E+01
F5	Mean	4.53E+00	4.00E+01	4.60E+01	3.55E+01	2.79E+01	2.55E+02	1.57E+02
	Std	1.82E+00	3.83E+00	5.37E+00	5.58E+00	5.66E+00	1.77E+01	2.75E+01
F6	Mean	1.01E-03	1.18E-03	1.60E-02	2.36E-05	1.52E-03	2.66E-05	1.69E-01
	Std	9.28E-04	1.07E-03	1.58E-02	2.12E-05	1.17E-03	1.66E-04	1.40E-01
F7	Mean	1.22E+02	1.31E+02	1.42E+02	1.40E+02	1.36E+02	3.78E+02	2.98E+02
EO	Sta	8.61E+00	3.14E+00	4.77E+00	6.63E+00	4.35E+00	2.14E+01	3.50E+01
го	Std	3.78E+00	4.20E+01	4.03E+01 4.21E+00	5.54E+01	2.65E+01	2.00E+02	1.40E+02
FO	Moon	1.09E+00 2.30E_02	4.09E+00 1 15E_01	4.31E+00 1.07E+00	0.00E+00 8 78E-03	4.02E+00 1.13E_01	2.10E+01 1.64E+00	2.33E+01 3.59E+01
1.2	Std	2.00E-02 8.06E-02	2 41F-01	9 39E-01	2.69E-02	1.13E-01 1.94E-01	1.04E+00 1.12E+00	2.39E+01 2.36E+01
F10	Mean	3 59E+03	1.04E+04	9.56E+03	9.68E+03	1.94E 01 1.02E+04	1.70E+00	1.11E+04
110	Std	6 55E+02	5.14E+02	5.89E+02	$5.60\pm100$ 5.61E+02	5.12E+02	4.77E+02	1.05E+03
F11	Mean	2.92E+02	4.47E+02	5.79E+02	1.02E+02	3.89E+02	1.31E+04	7.82E+02
	Std	8.17E+01	8.65E+01	9.38E+01	3.29E+01	1.02E+02	1.18E+04	2.24E+02
F12	Mean	3.15E+04	2.20E+04	1.93E+04	1.93E+04	2.53E+04	1.88E+04	3.43E+04
	Std	1.35E+04	8.16E+03	7.46E+03	8.28E+03	1.02E+04	6.88E+03	2.26E+04
F13	Mean	2.46E+03	2.03E+03	9.79E+02	2.23E+02	1.16E+03	2.24E+03	6.46E+02
	Std	5.58E+02	8.66E+02	7.06E+02	5.66E+01	7.54E+02	1.85E+03	7.56E+02
F14	Mean	2.45E+02	2.51E+02	2.73E+02	7.28E+01	2.54E+02	4.24E+02	4.83E+02
	Std	3.35E+01	3.86E+01	3.76E+01	1.25E+01	2.92E+01	1.06E+02	1.26E+02
F15	Mean	2.47E+02	2.45E+02	2.50E+02	2.17E+02	2.58E+02	3.01E+02	3.32E+02
	Std	4.21E+01	4.50E+01	4.20E+01	5.48E+01	4.90E+01	6.15E+01	1.25E+02
F16	Mean	1.96E+02	1.84E+03	1.48E+03	1.68E+03	1.54E+03	3.56E+03	2.83E+03
	Std	1.17E+02	2.46E+02	2.61E+02	3.63E+02	2.80E+02	3.20E+02	5.86E+02
F17	Mean	1.27E+02	1.2/E+03	1.1/E+03	1.14E+0.3	1.06E+03	2.68E+03	1.92E+03
E18	Moon	3.70E+01 2.00E+02	2.03E+02 2.12E+02	1.70E+02 2.27E+02	2.33E+02	1.76E+02 2.10E+02	2.00E+02 3.01E+02	4.37E+02 1 41E+03
1.10	Std	2.00E+02 3.69E+01	2.12E+02 3.02E+01	$4.80E\pm01$	3.96E+01	$4.26E\pm01$	8.51E+02	1.41E+0.03 1.23E+0.03
F19	Mean	1.69E+02	1.76E+02	1.84E+02	1 41E+02	1.20E+01	2.07E+02	2.39E+02
11/	Std	1.96E+01	2.10E+01	3.04E+01	1.93E+01	2.34E+01	5.28E+01	6.49E+01
F20	Mean	1.89E+02	1.60E+03	1.56E+03	1.26E+03	1.43E+03	2.81E+03	1.85E+03
	Std	6.55E+01	1.87E+02	1.81E+02	2.55E+02	2.20E+02	2.29E+02	3.75E+02
F21	Mean	2.32E+02	2.66E+02	2.71E+02	2.62E+02	2.58E+02	4.80E+02	3.64E+02
	Std	3.05E+00	5.17E+00	5.33E+00	5.48E+00	4.83E+00	2.13E+01	2.45E+01
F22	Mean	2.62E+03	1.12E+04	1.08E+04	9.71E+03	1.09E+04	1.81E+04	1.22E+04
	Std	5.64E+02	2.19E+03	6.04E+02	6.55E+02	5.90E+02	4.76E+02	1.11E+03
F23	Mean	5.58E+02	6.12E+02	5.94E+02	5.65E+02	5.62E+02	7.42E+02	6.94E+02
	Std	9.45E+00	1.13E+01	1.78E+01	9.24E+00	7.62E+00	1.12E+01	2.85E+01
F24	Mean	9.04E+02	9.28E+02	9.25E+02	9.25E+02	9.17E+02	1.10E+03	1.03E+03
<b>T</b> 05	Std	6.51E+00	1.53E+01	1.50E+01	1.00E+01	6.32E+00	1.60E+01	2.67E+01
F25	Mean	7.51E+02	7.33E+02	7.30E+02	7.21E+02	7.50E+02	7.42E+02	7.41E+02
E9(	Sta	2.52E+01	3.6/E+01	4.04E+01	4.36E+01	2.95E+01	4.26E+01	6.04E+01
F20	CLA	5.15E+05	3.30E+03	5.30E+03	3.37 E+03	5.40E+05	5.00E+05	4.31E+03
E27	Moon	6.36E+01	6.90E+01	1.21E+02	1.50E+02	1.12E+02	2.12E+02	2.09E+02
1.77	Std	0.30E+02	$1.34E\pm02$	0.50E+02	2.03E±01	0.34E+02	2.00E+02	7.00E+02 3.69E±01
F28	Mean	5.23E+02	5.34E+02	5.27E+02	5.34E+02	5.31E+02	5.37E+02	5.26E+02
1 20	Std	1.60E+01	3.51E+01	3.34E+01	2.87E+01	3.05E+01	3.23E+01	3.59E+01
F29	Mean	9.85E+02	1.25E+03	1.21E+03	1.46E+03	1.42E+03	2.41E+03	2.43E+03
/	Std	1.30E+02	1.55E+02	1.72E+02	2.11E+02	1.64E+02	2.31E+02	4.55E+02
F30	Mean	2.42E+03	2.55E+03	2.58E+03	2.50E+03	2.40E+03	3.39E+03	2.55E+03
	Std	1.42E+02	1.38E+02	1.84E+02	2.20E+02	1.40E+02	1.03E+03	1.93E+02
	+		16	21	14	19	22	23
	-		3	5	10	4	5	2
	=		11	4	6	7	3	5

Table A4. Mean errors of the algorithms under 100D of CEC 2017.

#### References

- 1. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the ICNN'95-International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948.
- 2. Dorigo, M.; Birattari, M.; Stutzle, T. Ant colony optimization. IEEE Comput. Intell. Mag. 2006, 1, 28–39. [CrossRef]
- 3. Mirjalili, S.; Lewis, A. The whale optimization algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [CrossRef]
- 4. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey wolf optimizer. Adv. Eng. Softw. 2014, 69, 46–61. [CrossRef]
- 5. Storn, R.; Price, K. Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [CrossRef]
- Liang, J.; Qiao, K.; Yue, C.; Yu, K.; Qu, B.; Xu, R.; Li, Z.; Hu, Y. A clustering-based differential evolution algorithm for solving multimodal multi-objective optimization problems. *Swarm Evol. Comput.* 2021, 60, 100788. [CrossRef]
- Mezura-Montes, E.; Velázquez-Reyes, J.; Coello, C.C. Modified differential evolution for constrained optimization. In Proceedings of the 2006 IEEE International Conference on Evolutionary Computation, Vancouver, BC, Canada, 16–21 July 2006; pp. 25–32.
- 8. Deng, W.; Shang, S.; Cai, X.; Zhao, H.; Zhou, Y.; Chen, H.; Deng, W. Quantum differential evolution with cooperative coevolution framework and hybrid mutation strategy for large scale optimization. *Knowl.-Based Syst.* **2021**, 224, 107080. [CrossRef]
- Wang, Z.J.; Zhan, Z.H.; Kwong, S.; Jin, H.; Zhang, J. Adaptive granularity learning distributed particle swarm optimization for large-scale optimization. *IEEE Trans. Cybern.* 2020, *51*, 1175–1188. [CrossRef]
- Tanabe, R.; Fukunaga, A. Success-history based parameter adaptation for differential evolution. In Proceedings of the 2013 IEEE Congress on Evolutionary Computation, Cancun, Mexico, 20–23 June 2013; pp. 71–78.
- 11. Tanabe, R.; Fukunaga, A.S. Improving the search performance of SHADE using linear population size reduction. In Proceedings of the 2014 IEEE Congress on Evolutionary Computation (CEC), Beijing, China, 6–11 July 2014; pp. 1658–1665.
- 12. Zhang, X.; Liu, Q.; Qu, Y. An adaptive differential evolution algorithm with population size reduction strategy for unconstrained optimization problem. *Appl. Soft Comput.* **2023**, *138*, 110209. [CrossRef]
- 13. Zeng, Z.; Zhang, M.; Zhang, H.; Hong, Z. Improved differential evolution algorithm based on the sawtooth-linear population size adaptive method. *Inf. Sci.* 2022, 608, 1045–1071. [CrossRef]
- 14. Li, Y.; Wang, S.; Yang, B.; Chen, H.; Wu, Z.; Yang, H. Population reduction with individual similarity for differential evolution. *Artif. Intell. Rev.* **2023**, *56*, 3887–3949. [CrossRef]
- Qin, A.K.; Suganthan, P.N. Self-adaptive differential evolution algorithm for numerical optimization. In Proceedings of the 2005 IEEE Congress on Evolutionary Computation, Edinburgh, UK, 2–4 September 2005; Volume 2, pp. 1785–1791.
- 16. Wang, Y.; Cai, Z.; Zhang, Q. Differential evolution with composite trial vector generation strategies and control parameters. *IEEE Trans. Evol. Comput.* **2011**, *15*, 55–66. [CrossRef]
- 17. Peng, H.; Han, Y.; Deng, C.; Wang, J.; Wu, Z. Multi-strategy co-evolutionary differential evolution for mixed-variable optimization. *Knowl.-Based Syst.* **2021**, *229*, 107366. [CrossRef]
- 18. Zhong, X.; Cheng, P. An elite-guided hierarchical differential evolution algorithm. Appl. Intell. 2021, 51, 4962–4983. [CrossRef]
- 19. Zhang, J.; Sanderson, A.C. JADE: Adaptive differential evolution with optional external archive. *IEEE Trans. Evol. Comput.* 2009, 13, 945–958. [CrossRef]
- 20. Li, Y.; Han, T.; Wang, X.; Zhou, H.; Tang, S.; Huang, C.; Han, B. MjSO: A modified differential evolution with a probability selection mechanism and a directed mutation strategy. *Swarm Evol. Comput.* **2023**, *78*, 101294. [CrossRef]
- 21. Wang, H.B.; Ren, X.N.; Li, G.Q.; Tu, X.Y. APDDE: Self-adaptive parameter dynamics differential evolution algorithm. *Soft Comput.* **2018**, *22*, 1313–1333. [CrossRef]
- 22. Mohamed, A.W.; Suganthan, P.N. Real-parameter unconstrained optimization based on enhanced fitness-adaptive differential evolution algorithm with novel mutation. *Soft Comput.* **2018**, *22*, 3215–3235. [CrossRef]
- 23. Kumar, P.; Pant, M.; Singh, V. Information preserving selection strategy for differential evolution algorithm. In Proceedings of the 2011 World Congress on Information and Communication Technologies, Mumbai, India, 11–14 December 2011; pp. 462–466.
- 24. Tian, M.; Gao, X.; Dai, C. Differential evolution with improved individual-based parameter setting and selection strategy. *Appl. Soft Comput.* **2017**, *56*, 286–297. [CrossRef]
- Guo, J.; Li, Z.; Yang, S. Accelerating differential evolution based on a subset-to-subset survivor selection operator. *Soft Comput.* 2019, 23, 4113–4130. [CrossRef]
- 26. Zeng, Z.; Zhang, M.; Chen, T.; Hong, Z. A new selection operator for differential evolution algorithm. *Knowl.-Based Syst.* 2021, 226, 107150. [CrossRef]
- 27. Zhan, Z.H.; Wang, Z.J.; Jin, H.; Zhang, J. Adaptive distributed differential evolution. *IEEE Trans. Cybern.* **2019**, *50*, 4633–4647. [CrossRef] [PubMed]
- 28. Xia, X.; Gui, L.; Zhang, Y.; Xu, X.; Yu, F.; Wu, H.; Wei, B.; He, G.; Li, Y.; Li, K. A fitness-based adaptive differential evolution algorithm. *Inf. Sci.* 2021, 549, 116–141. [CrossRef]
- 29. Deng, L.; Li, C.; Han, R.; Zhang, L.; Qiao, L. TPDE: A tri-population differential evolution based on zonal-constraint stepped division mechanism and multiple adaptive guided mutation strategies. *Inf. Sci.* **2021**, *575*, 22–40. [CrossRef]
- 30. Pan, J.S.; Liu, N.; Chu, S.C. A hybrid differential evolution algorithm and its application in unmanned combat aerial vehicle path planning. *IEEE Access* 2020, *8*, 17691–17712. [CrossRef]
- Li, Y.; Wang, S.; Yang, H.; Chen, H.; Yang, B. Enhancing differential evolution algorithm using leader-adjoint populations. *Inf. Sci.* 2023, 622, 235–268. [CrossRef]

- 32. Wang, M.; Ma, Y. A differential evolution algorithm based on accompanying population and piecewise evolution strategy. *Appl. Soft Comput.* **2023**, *143*, 110390. [CrossRef]
- 33. Wu, G.; Mallipeddi, R.; Suganthan, P.N. Problem Definitions and Evaluation Criteria for the CEC 2017 Competition on Constrained Real-Parameter Optimization; Technical Report; National University of Defense Technology: Changsha, China; Kyungpook National University: Daegu, Republic of Korea; Nanyang Technological University: Singapore, 2017.
- Das, S.; Suganthan, P.N. Problem Definitions and Evaluation Criteria for CEC 2011 Competition on Testing Evolutionary Algorithms on Real World Optimization Problems; Jadavpur University: Kolkata, India; Nanyang Technological University: Singapore, 2010; pp. 341–359.
- Brest, J.; Maučec, M.S.; Bošković, B. Single objective real-parameter optimization: Algorithm jSO. In Proceedings of the 2017 IEEE Congress on Evolutionary Computation (CEC), Donostia-San Sebastián, Spain, 5–8 June 2017; pp. 1311–1318.
- Wu, G.; Mallipeddi, R.; Suganthan, P.N.; Wang, R.; Chen, H. Differential evolution with multi-population based ensemble of mutation strategies. *Inf. Sci.* 2016, 329, 329–345. [CrossRef]
- 37. Meng, Z.; Yang, C. Hip-DE: Historical population based mutation strategy in differential evolution with parameter adaptive mechanism. *Inf. Sci.* 2021, *562*, 44–77. [CrossRef]
- Meng, Z.; Pan, J.S.; Tseng, K.K. PaDE: An enhanced Differential Evolution algorithm with novel control parameter adaptation schemes for numerical optimization. *Knowl.-Based Syst.* 2019, 168, 80–99. [CrossRef]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.