Engineering Applications of Artificial Intelligence 133 (2024) 108081

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

Differential Evolution with fitness-difference based parameter control and hypervolume diversity indicator for numerical optimization

Chongle Ren, Zhenghao Song, Zhenyu Meng *

Institute of Artificial Intelligence, Fujian University of Technology, Fuzhou, China

ARTICLE INFO

Keywords:
Differential evolution
Diversity improvement
Fitness difference
Hypervolume diversity indicator
Parameter control

ABSTRACT

Differential Evolution (DE) is one of the most popular and powerful branches of evolutionary algorithm family. However, even many state-of-the-art DE-based variants still exist weakness such as improper parameter adaptation and population stagnation during the later stage of evolution. To mitigate these deficiencies, differential evolution with fitness-difference based parameter control and hypervolume diversity indicator (FDHD-DE) is proposed in this paper. Firstly, a semi-adaptive adaptation scheme for control parameters is proposed, in which the generation of scale factor and crossover rate is modified by dividing into two stages, thus enhancing the efficiency of parameter adaptation. Secondly, a novel fitness-based weighting strategy is proposed to improve the performance of existing success history-based adaptation by employing a novel approach of utilizing fitness information. Finally, a hypervolume-based diversity indicator and corresponding dimension exchange strategy are proposed to alleviate the problem of population stagnation. The performance of FDHD-DE is verified on the 88 benchmark functions from Congress on Evolutionary Computation (CEC) 2013, CEC 2014, and CEC 2017 test suites on 10D, 30D and 50D and a real-world application. The experiment results are compared with several state-of-art DE variants, and the results show that FDHD-DE has better performance, both in terms of solution accuracy and convergence speed.

1. Introduction

Differential Evolution (DE), firstly proposed by Storn and Price to solve Chebyshev polynomial fitting problem in 1995 (Storn and Price, 1997), has gradually gained popularity owing to its easy of implementation and powerful optimization capacity. Similar to other evolutionary algorithms (EAs) (Meng and Pan, 2016; Wang et al., 2018; Meng and Pan, 2018), DE employs evolutionary operators including mutation, crossover and selection, and only the better individual between parent individual and offspring individual will survive to the next generation. At the beginning of evolution, individuals are initialized randomly within the decision space. In the context of DE, individuals are termed as target vectors. Following initialization, mutation vectors are generated between target vectors according to the guide of mutation strategy. The classic mutation strategy DE/rand/1 is presented below for example

$$V_{i,G} = X_{r_1,G} + F \cdot (X_{r_2,G} - X_{r_3,G}) \tag{1}$$

where r_1, r_2 and r_3 are three different index from $\{1, 2, \ldots, PS\}$. F is termed as scale factor which controls the size of difference vector. Through crossover operation, a trial vector $U_{i,G}$ is generated by exchanging dimension between the donor vector $V_{i,G}$ and the target vector $X_{i,G}$.

$$U_{j,i,G} = \begin{cases} V_{j,i,G}, & if\ rand(0,1) \le CR\ or\ j = j_{rand} \\ X_{j,i,G}, & otherwise \end{cases} \tag{2}$$

where CR is the abbreviation of crossover rate which determine how many components the trial vector is inherited from the mutant vector or the target vector (Li and Meng, 2023). Lastly, the greedy selection is conducted between the two vectors, as shown below:

$$X_{i,G+1} = \begin{cases} U_{i,G}, & if\ f(U_{i,G}) < f(X_{i,G}) \\ X_{i,G}, & otherwise \end{cases} \tag{3}$$

Based on basic operators of DE mentioned above, we can tell that two main operators mutation and crossover is deeply associated with two control parameters F and CR. Therefore, the performance of DE is high sensitive to its parameter setting (Elsayed and Sarker, 2013). During the early stage of research on DE, numerous efforts have been dedicated to find out the appropriate setting of parameters to obtain satisfactory optimization results (Meng et al., 2016; Meng and Pan, 2019; Meng and Zhang, 2023). This parameter tuning methods require plenty of computational resources to obtain a set of parameters which

* Corresponding author.
E-mail address: mzy1314@gmail.com (Z. Meng).

https://doi.org/10.1016/j.engappai.2024.108081
Received 16 February 2023; Received in revised form 27 December 2023; Accepted 10 February 2024
Available online 20 February 2024
0952-1976/

perform well on a type of optimization problems. When applying DE with the parameter setting in another type of optimization problems, it may fail to yield satisfactory results (Dragoi and Dafinescu, 2016).

Researchers have paid more attention to parameter adaptation schemes, which allow to adjust control parameters in an adaptive manner during the evolutionary process (Song et al., 2023b). Brest et al. (2006) presented a parameter self-adaptation method, in which each individual is assigned to a parameter set including F and CR. The vector with better parameters will have a higher possibility of survival and thus pass its parameters to the next generation. Zhang and Sanderson (Zhang and Sanderson, 2009) proposed a powerful DE variant named JADE which won the competition at WCCI2008. In JADE, a novel parameter adaptation strategy is proposed by employing semi-fixed distributions to generate both F and CR based on location parameter set, which are updated using the parameters of successful individual in one-to-one selection. LSHADE, proposed by Tanabe and Fukunaga (Tanabe and Fukunaga, 2014), improved robustness of parameter adaptation in JADE by introducing a fitness-dependent weights of the successful parameters. Based on the framework of LSHADE, jSO (Brest et al., 2017) was proposed by truncating the parameters adaptation to fixed values according to different phases of evolution. To mitigate the problem of improper adaptation of CR in many DE variants, Meng et al. (2019) proposed a grouping strategy by dividing the whole population into several groups with different CR values based on the parameter adaptation of LSHADE.

With the development of artificial intelligence, machine learning and reinforcement learning techniques have been leveraged to adjust the control parameters of DE. For example, Liu et al. (2023) viewed the utilization of past experiences obtained through optimization trajectories as a learning behavior, which can be modeled as a Markov Decision Process. Then, a deep neural network is used as parameter controller which generate control parameters. Sun et al. (2021) modeled the parameter control as a finite-horizon Markov decision process and employed policy gradient algorithm to learn an agent which outputs appropriate control parameters. In Sharma et al. (2019), a double deep Q-learning network(DDQN) was used as parameter controller of DE. In the proposed algorithm, different combination of parameters and mutation strategies are assigned to each parent and each generation under the control of DDQN. Reijnen (Reijnen et al., 2022) employed a deep reinforcement learning for parameter adaptation in a multi-objective DE. Experiment results confirm the effectiveness of the novel deep reinforcement learning based parameter adaptation strategy.

Population diversity also attaches great importance to the overall performance of DE (Nadimi-Shahraki and Zamani, 2022; Zhang and Meng, 2023b; Song et al., 2023a). Two common problems stagnation and premature convergence are closely related to the level of population diversity (Yang et al., 2014). During the later stage of evolution, population diversity will reduce as individuals coalesce into local optima. This phenomenon is inevitable in population-based evolutionary algorithm, however, excessively low-level population diversity will result in stagnation. In order to improve the performance of DE in terms of population diversity, it is a top priority to measure the level of population diversity. Song et al. (2023b) proposed a hybrid diversity indicator which employed covariance matrix and fitness information to evaluate the population diversity. Yang et al. (2014) proposed a diversity measurement method based on spatial distribution. Li et al. (2023) proposed a population state evaluation-based framework which evaluates optimization state of population and distribution state of individuals.

After measuring the population diversity, the stagnant individual detected by diversity indicator should be updated in order to get rid of stagnation state (Song et al., 2023b; Meng et al., 2023). Several intervention methods have been proposed to improve population diversity of DE. Meng et al. (2021) proposed an intervention scheme to update stagnant individual by reinitializing a certain number of dimensions of the individual. In Zhang and Meng (2023a), a rank-based mutation scheme was employed to update stagnant individual and adopted the Cauchy distribution to generate F used in the intervention method. Based on the above considerations, the search ability of DE can be enhanced from the perspective of parameter adaptation and population diversity. In addition, as increasing decision variables and more complex fitness landscape in real-world applications, existing algorithms faces greater challenges with the problems of stagnation and premature convergence. Therefore, a differential evolution with fitness-difference based parameter control and hypervolume diversity indicator known as FDHD-DE is proposed in this paper. The main contributions of FDHD-DE are presented as follows:

1. A novel semi-adaptive adaptation scheme for control parameters is proposed, in which the adaptation of F and CR is enhanced by dividing into two stages, thus achieving a better exploration and exploitation balance.
2. A novel fitness-dependent weighting strategy is proposed to improve the efficiency of success history-based adaptation by employing a novel approach of utilizing fitness improvement information.
3. A hypervolume-based diversity indicator is proposed to measure the level of population diversity and then detect the stagnant individual, which is updated by dimension exchange mechanism. This strategy is able to mitigate the problem of stagnation.
4. To evaluate the performance of the proposed FDHD-DE, a total of 88 benchmark functions from CEC2013, CEC2014 and CEC2017 test suites are employed, thus avoiding the overfitting problem in single test suite.

The rest of paper is arranged as follows: Section 2 introduces several popular DE-based variants, and some techniques are employed in FDHD-DE. Section 3 elaborates components of FDHD-DE. Experiment results on benchmark functions and real-world applications are provided in Section 4. Section 5 contains the conclusion.

2. Related work

In this section, we introduce several well-known DE variants, including LSHADE (Tanabe and Fukunaga, 2014), jSO (Brest et al., 2017), PaDE (Meng et al., 2019), LPalmDE (Meng et al., 2018), and Hip-DE (Meng and Yang, 2021).

2.1. The LSHADE algorithm

LSHADE, proposed by Tanabe and Fukunaga (2014) won the CEC2014 competition with the success-history based parameter adaptation and population reduction technique. In LSHADE, the mutation strategy DE/current-to-best/1 proposed in JADE is used to generate mutant vector, as shown below:

$$V_{i,G+1} = X_{i,G} + F \cdot (X^{p}_{best,G} - X_{i,G}) + F \cdot (X_{r_1,G} - \widetilde{X}_{r_2,G}) \tag{4}$$

where $X^{p}_{best,G}$ is randomly selected with top $p \cdot PS$ individuals, in which PS denotes the population size. $\widetilde{X}_{r_2,G}$ are chosen in the union of current population and external archive A.

For parameter adaptation, a memory pool $< \mu_F, \mu_{CR} >$ is maintained, which improves the robustness of choosing the appropriate control parameters. Based on parameter in memory pool, the generation of F and CR of the ith individual obeys Cauchy distribution and Gaussian distribution, respectively. The detailed process is shown in Eq. (5).

$$\begin{cases} F_i = randc_i(\mu_{F,r_i}, 0.1) \\ CR_i = \begin{cases} 0, & if\ \mu_{CR_i,r_i} = \emptyset \\ randn_i(\mu_{CR,r_i}, 0.1), & otherwise \end{cases} \end{cases} \tag{5}$$

where r_i is a random number between $[1, H]$. At the end of each generation, the parameters in memory pool will be updated based on control parameters of successful individual in one-to-one selection. In addition, the fitness difference is employed as weight in the Lehmer mean.

$$\begin{cases} \Delta f_k = |f(X_{i,G}) - f(U_{i,G})| \\ w_k = \dfrac{\Delta f_k}{\sum_{k=1}^{|S_F|} \Delta f_k} \\ mean_{WL}(S_F) = \dfrac{\sum_{k=1}^{|S_F|} w_k \cdot S_F^2(k)}{\sum_{k=1}^{|S_F|} w_k \cdot S_F(k)} \\ \mu_{F,k,G+1} = \begin{cases} mean_{WL}(S_F), & if\ S_F \neq \emptyset \\ \mu_{F,k,G}, & otherwise \end{cases} \end{cases} \tag{6}$$

$$\begin{cases} w_k = \dfrac{\Delta f_k}{\sum_{k=1}^{|S_{CR}|} \Delta f_k} \\ mean_{WL}(S_{CR}) = \dfrac{\sum_{k=1}^{|S_{CR}|} w_k \cdot S_{CR}^2(k)}{\sum_{k=1}^{|S_{CR}|} w_k \cdot S_{CR}(k)} \\ \mu_{CR,k,G+1} = \begin{cases} mean_{WL}(S_{CR}), & if\ S_{CR} \neq \emptyset \\ \mu_{CR,k,G}, & otherwise \end{cases} \end{cases} \tag{7}$$

where S_F and S_{CR} denote the F and CR of successful individual, respectively. The adaptation strategy has been demonstrated to be highly effective in optimization competitions. However, there still exist a weakness which is the dependence on the fitness difference between target vector and trial vector. Then a population size reduction strategy is adopted to linearly decrease the population size as evolution process. The strategy is shown as follows:

$$PS_{G+1} = round[\frac{PS_{min} - PS_{max}}{nfes_{max}} \cdot nfes + PS_{max}] \tag{8}$$

where PS_{max} and PS_{min} denote the maximum and minimum population size. $nfes_{max}$ and $nfes$ denote the maximum and current number of fitness evaluations.

2.2. The jSO algorithm

Based on LSHADE, Brest et al. (2017) proposed the jSO algorithm (Brest et al., 2017) by employing dynamic adjustment of F in mutation strategy, as shown below:

$$V_{i,G} = X_{i,G} + F_w \cdot (X^p_{best,G} - X_{i,G}) + F \cdot (X_{r_1,G} - \widetilde{X}_{r_2,G}) \tag{9}$$

where F_ω is an inertia weight which is mainly used to change the evolution direction by constrict the difference vector, as shown in Eq. (10). p is used to determine the ratio of top-ranking individual to the whole population. Unlike setting p to a fixed value in LSHADE, jSO updates the value of p according to the evolutionary stages, as shown in Eq. (11).

$$F_w = \begin{cases} 0.7 \cdot F & if\ nfes < 0.2 \cdot nfes_{max} \\ 0.8 \cdot F & if\ nfes < 0.4 \cdot nfes_{max} \\ 1.2 \cdot F & otherwise \end{cases} \tag{10}$$

$$p = (\frac{p_{max} - p_{min}}{max_n fes}) \cdot nfes + p_{min} \tag{11}$$

In jSO, a few modifications are made based on success-history based parameter adaptation strategy including the generation of F and CR, as shown below:

$$F_{i,G} = \begin{cases} min(F_{i,G}, 0.7), & if\ nfes < 0.6 \cdot nfes_{max} \\ 0.7 & otherwise \end{cases} \tag{12}$$

$$CR_{i,G} = \begin{cases} max(CR_{i,G}, 0.7), & if\ nfes < 0.25 \cdot nfes_{max} \\ max(CR_{i,G}, 0.6), & if\ nfes < 0.5 \cdot nfes_{max} \\ CR_{i,G}, & otherwise \end{cases} \tag{13}$$

Based on the above modifications, one can observe that these adjustment attach much intervention on the parameter adaptation process, and it may result in over-tuning problem according to “No free lunch theory” (Wolpert and Macready, 1997). In general, a successful parameter adaptation strategy should pose less intervention to the generation of control parameters with aim to cater for optimization problems with different characteristics. This thought inspires the FDHD-DE in the paper.

2.3. The PaDE algorithm

A novel Parameter adaptive DE (PaDE) (Meng et al., 2019) was proposed to tackle some weaknesses such as improper adjustment schemes for the control parameters and defects of linear population size reduction in some DE-based variants. A grouping-based adaptation method for CR is proposed to deal with improper adjustment schemes of CR. With respect to adaptation of population size, a parabolic population reduction mechanism is introduced.

In PaDE, parameter μ_{CR} and the selection probability $P(\cdot)$ are grouped. The control parameter μ_{CR} is the mean value of the normal distribution, which each control parameter in the group is supposed to obey. The selection probability $P(\cdot)$ represents the probability that individuals in the population are assigned to a certain group $p(1) = p(2) = p(3) = p(4)... = p(1) = \frac{1}{k}$, and updated according to Eq. (14).

$$\begin{cases} ns = \sum_{j=1}^{k} ns_j \\ P(j) = \dfrac{rj}{\sum_{j=1}^{k}(rj)} \\ r_j = \begin{cases} \frac{ns_j^2}{ns \cdot (ns_j + nj_j)}, & if\ ns_j > 0 \\ \epsilon, & otherwise \end{cases} \end{cases} \tag{14}$$

where ns_j and nf_j are the number of “s” vectors and “f” vectors in the jth group, and ns is the total number of “s” vectors. To avoid null values of probability. ϵ is assigned to a small constant value, e.g. ϵ= 0.001, to avoid possible zero values during the iterative process. Besides, a new parabolic population size reduction technique is proposed, as shown in Eq. (15).

$$ps_{G+1} = \begin{cases} [\frac{y - PS_{ini}}{(x - PS_{ini})^2} \cdot (nfes - PS_{ini})^2 + PS_{ini}], & if\ nfes < x \\ [\frac{y - PS_{ini}}{x - PS_{ini}} \cdot (nfes - nfes_{max}) + PS_{min}], & otherwise \end{cases} \tag{15}$$

where the $pivot$ connecting the ellipse and the line segment is $Pivot = (x, y)$. This creates an efficient transition population.

2.4. The LPalmDE algorithm

LPalmDE (Meng et al., 2018) was proposed to address the inconvenience of selection of the control parameters F and CR in many DE variants. The idea A better trial vector may be generated by combination of a good F and a bad CR (or vice versa), resulting in that the bad control parameters are considered as a correct direction of evolution. This phenomenon is defined as “misleading interaction between parameters ”. To address this weakness, a new parameter adaptation with adaptive learning mechanism was proposed in LPalmDE.

$$\begin{cases} ns = \sum_{k=1}^{K} ns_k, \\ r_k = \begin{cases} \dfrac{ns_k^2}{ns \cdot (ns_k + nf_k)}, & if\ ns_k > 0 \\ \epsilon, & otherwise \end{cases} \\ p(k) = \dfrac{r_k}{\sum_{k=1}^{K}(r_k)} \\ \mu_{CR} = \dfrac{\sum_{k=1}^{K} p(k) \cdot CR_k^2}{\sum_{k=1}^{K} p(k) \cdot CR_k} \end{cases} \tag{16}$$

where ns_k and nf_k denotes the number of successful individuals and failed(or unsuccessful) in the k-th group, respectively.

An additional enhancement in the LPalmDE involves the integration of a time-stamp based mechanism into the mutation strategy "DE/target-to-$p$best/1 ". This mechanism is designed to prevent excessively old and inferior solutions from becoming "archive residents" throughout the entire evolution process.

2.5. The Hip-DE algorithm

Based on LPalmDE, Hip-DE (Meng and Yang, 2021) was proposed to tackle the weakness in mutation strategy and parameter adaptation. The mutation strategy employed in the standard DE algorithm, DE/rand/1, solely considered the relationships among individuals in the current generation, without utilizing any historical knowledge in generating trial vectors during the evolutionary process. In contrast, the mutation strategy DE/target-to-$p$best/1, not only took into account the relationships among individuals in the current generation but also considered the connections between individuals in the present generation and inferior individuals from the past. As a result, this mutation strategy is able to maintain a balance between diversity and convergence speed. However, historical information is overlooked in the mutation strategy. Therefore, a novel mutation strategy was proposed to utilize the historical information of the whole population, as shown below:

$$V_{i,G} = X_{i,G} + F \cdot (X_{best,G}^{p} - X_{i,G} + X_{r_1,G} - \widetilde{X}_{r_2,G}) \tag{17}$$

where $X_{r_2,G}$ denotes a randomly selected vector in the $\mathbf{P} \cup \mathbf{H}$ population, where H denotes the historical population and P denotes the current population.

For parameter adaptation, some modifications are also made based on LPalmDE, as shown below:

$$\begin{cases} S = \bigcup_{k=1}^{k} S_k \\ w_k = \dfrac{\Delta f_k}{\sum_{k=1}^{|S|} \Delta f_k} \\ mean_{WL}(S_F) = \dfrac{\sum_{k=1}^{|S_F|} w_k \cdot S_F^2(k)}{\sum_{k=1}^{|S_F|} w_k \cdot S_F(k)} \\ \mu_F = \begin{cases} (1-c) \cdot \mu_F + c \cdot mean_{WL}(S_F), & if\ S \neq \emptyset \\ \mu_F, & otherwise \end{cases} \end{cases} \tag{18}$$

$$\begin{cases} mean_{WL}(S_{CR}) = \dfrac{\sum_{k=1}^{|S_{CR}|} w_k \cdot S_{CR}^2(k)}{\sum_{k=1}^{|S_{CR}|} w_k \cdot S_{CR}(k)} \\ \mu_{CR_{idx}} = \begin{cases} mean_{WL}(S_{CR}), & if\ S \neq \emptyset \\ \mu_{CR_{idx}}, & otherwise \end{cases} \end{cases} \tag{19}$$

$$\begin{cases} F_{tmp} = randc(\mu_F, 0.1) \\ F_{tmp} = \begin{cases} randc(\mu_F, 0.1), & while\ F \leq 0 \\ 1. & if\ F > 1 \end{cases} \\ F_i = \begin{cases} F_{tmp}, & if\ rand_1 < \tau_1 \\ F_{i,G}, & otherwise \end{cases} \end{cases} \tag{20}$$

$$CR_i = \begin{cases} C(\mu_{CR_k}, 0.1), & if\ rand_2 < \tau 2 \& \mu_{Cr} \neq 0 \\ 0, & if\ rand_2 < \tau 2 \& \mu_{Cr} = 0 \\ CR_{i,G}, & otherwise \end{cases} \tag{21}$$

For the new mechanism of parameter control, CR follows the normal distribution and F follows the Cauchy distribution. F is primarily dependent on τ_1, whereas CR is primarily based on τ_2 to determine whether to be altered by setting the variables τ_1 and τ_2.

A new population size reduction method was also proposed in the Hip-DE algorithm, where in the early stages of population the population maintains a fixed value and subsequently its size will decrease at step-down manner based on evaluation of the function. The novel reduction mechanism is shown as follows:

$$PS = \begin{cases} PS_{ini}, & if\ nfes \leq nfes_{st} \\ \lceil \frac{PS_{min} - PS_{ini}}{nfes_{max} - nfe_{st}} \cdot (nfes - nfes_{st}) + PS_{ini} \rceil, & otherwise \end{cases} \tag{22}$$

where $nfes_{st}$ represents the number of fitness evaluations allowed during the early generations of evolution and remains fixed at $PS \cdot PS_{ini}$.

3. The proposed FDHD-DE algorithm

In this section, we present the detailed description of FDHD-DE by dividing it into three parts: the first part gives the semi-adaptation scheme for control parameters; the second part provides the novel dynamic fitness-dependent weighting strategy for updating parameters in memory pool; and the last part presents the hypervolume-based diversity indicator and dimension exchange strategy.

3.1. Semi-adaptation scheme for control parameters

Parameter adaptation attaches a great importance to the overall performance of DE owing to its high sensitivity of parameters. Success-history based parameter adaptation strategy in LSHADE has been confirmed to be effective in tackling optimization problems of different characteristics. Many DE-based variants which employ this strategy yielded excellent performance in real-world applications as well as CEC competition series. However, there still exist shortcomings in the strategy. For example, F and CR of each individual are generated based on probability distributions which improves the randomness of parameter setting. However, during the early stage of evolution, generating parameters solely based probability distribution will waste a large number of fitness evaluations by finding the appropriate parameter through adaptation scheme. This phenomenon is illustrated in Fig. 1. Based on our experiments in adaptation process in LSHADE, we find that the value of F will gradually maintain within the range of [0.4, 0.6] through adaptation. Therefore, to save the number of fitness evaluation and improve optimization accuracy, we propose a semi-adaptation scheme for control parameters. The whole adaptation process is divided into two stages: in the early stage of evolution, F is generated based on uniform distribution within [0.4, 0.6]; in the later stage of evolution, F is generated according to Cauchy distribution. The generation of CR obey Gaussian distribution and its value is truncated to 0.6 during the first stage evolution and kept within [0, 1] during the later stage.

$$F_i = \begin{cases} 0.5 + 0.1 \cdot rand, & if\ nfes < \perp \\ randc_i(\mu_{F,r_i}, 0.1), & otherwise \end{cases} \tag{23}$$

$$CR_i = \begin{cases} 0, & if\ \mu_{CR,r_i} = \emptyset \\ randn_i(\mu_{CR,r_i}, 0.1), & otherwise \end{cases} \tag{24}$$

$$CR_i = \begin{cases} max(CR_i, 0.6) & if\ nfes < \perp \\ max(CR_i, 0), min(CR_i, 1), & otherwise \end{cases} \tag{25}$$

where $\perp$ denotes the threshold between two stages in the semi-adaptive scheme and it is set to $nfes_{max} \cdot 0.3$.

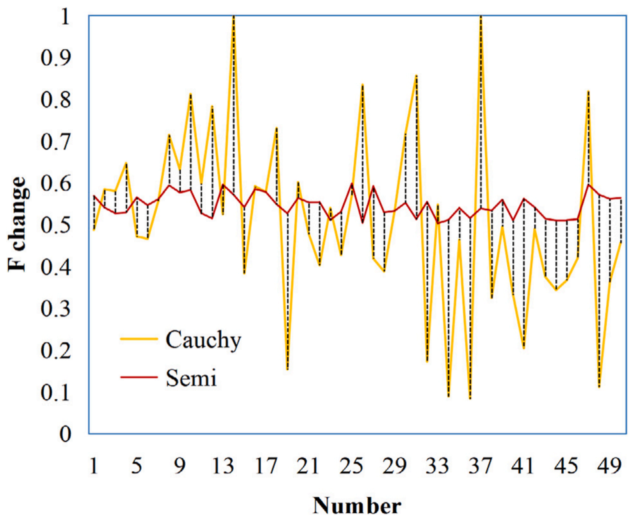


Fig. 1. Change curve of F.

3.2. History memory update with novel fitness-based weighting strategy

The parameter μ_F and μ_{CR} in memory pool will update at the end of each generation to adjust the parameter in the next generation. These memories are used to retain successful values of F and CR used in the mutation and crossover operations and updated based on weighted Lehmer mean of fitness improvement between target vector and trial vector. This weighting strategy focuses on exploitation rather than exploration, which may result in premature convergence and stagnation, especially in high-dimensional decision space (Viktorin et al., 2019; Meng, 2023). To mitigate this deficiency, we employ a counter co to count the number of generations, in which an individual achieve fitness improvement consecutively. If the predefined condition is satisfied, the fitness improvement is calculated as follows:

$$\begin{cases} \Delta f_{k1} = \sqrt{f(X_{i,G}) - f(U_{i,G})} \\ \Delta f_{k2} = \sqrt{f(X_{i,G}) - f(X_{gbest,G})} \\ \Delta f_k = (\Delta f_{k1} + \Delta f_{k2})/2 \end{cases} \tag{26}$$

Otherwise, the fitness difference between target vector and the current best vector is utilized, as shown below:

$$\Delta f_k = \sqrt{f(X_{i,G}) - f(X_{gbest,G})} \tag{27}$$

Subsequently, the parameters μ_F and μ_{CR} in memory pool are updated as follows:

$$\begin{cases} mean_{WL}(S_F) = \dfrac{\sum_{k=1}^{|S_F|} w_k \cdot S_F^2(k)}{\sum_{k=1}^{|S_F|} w_k \cdot S_F(k)} \\ \mu_{F,k,G+1} = \begin{cases} mean_{WL}(S_F) & if\ S_F \neq \emptyset \\ \mu_{F,k,G}, & otherwise \end{cases} \end{cases} \tag{28}$$

$$\begin{cases} mean_{WL}(S_{CR}) = \dfrac{\sum_{k=1}^{|S_{CR}|} w_k \cdot S_{CR}^2(k)}{\sum_{k=1}^{|S_{CR}|} w_k \cdot S_{CR}(k)} \\ \mu_{CR,k,G+1} = \begin{cases} mean_{WL}(S_{CR}), & if\ S_{CR} \neq \emptyset \\ \mu_{CR,k,G}, & otherwise \end{cases} \end{cases} \tag{29}$$

The pseudocode of the fitness-based weighting strategy is shown in Algorithm 1.

Algorithm 1 The weighting strategy

```
co = 0;
for i = 1 to PS do
    if f(X_{i,G}) > f(U_{i,G}) then
        co = co + 1;
    end if
end for
if co > 3 then
    Δf_{k1} = sqrt(f(X_{i,G}) − f(U_{i,G}));
    Δf_{k2} = sqrt(f(X_{i,G}) − f(X_{gbest,G}));
    Δf_k = (Δf_{k1} + Δf_{k2})/2;
else
    Δf_k = sqrt(f(X_{i,G}) − f(X_{gbest,G}));
end if
```

3.3. Hypervolume-based diversity indicator and dimension exchange strategy

Based on our research, we observed that after a certain number of generations, the evolutionary process may encounter stagnation. Even with mutation and crossover, it is nearly impossible to get rid of stagnation. To measure the level of population diversity, we employ hypervolume to calculate the spatial distribution of individuals in a population. The hypervolume is calculated as follows:

$$V_{lim} = \sqrt{\prod_{i=1}^{D} |u_i - l_i|} \tag{30}$$

where u_i and l_i represent the upper and lower boundary of the ith dimension of the search space. After that, the diversity indicator $nVOL$ is calculated according to

$$\begin{cases} V_{pop} = \sqrt{\prod_{i=1}^{D} |(u_{x_i} - l_{x_i})/2|} \\ nVOL = \sqrt{V_{pop}/V_{lim}} \end{cases} \tag{31}$$

where u_{x_i} and l_{x_i} denote the lower and upper bound of ith dimension of the current population.

Then we employed a counter to count the number of generations in which individuals fail to achieve fitness improvement, as presented in Algorithm 2. When $nVOL < \xi$ ($\xi = 0.01$) and $C_n > k \cdot PS \cdot D$, a dimension exchange is performed as intervention strategy to update stagnant individual, as shown below:

$$X_{i,G}(j) = \begin{cases} X_{randi(i),G}(j), & if\ j \in R \\ X_{i,G}, & otherwise \end{cases} \tag{32}$$

The idea behind dimension exchange is simple. As evolution proceeds, individual which is labeled as stagnation may be resulted from some dimensions falling into the local optimum, while some dimensions may also contain useful information. By exchanging dimension, the stagnant individual is expected to escape from stagnation. It is worth noting that we employ a linear reduction strategy to adjust the value of k because stagnation typically occurs in the later stages of evolution. When the value of k is high, the threshold of fitness-based is hard to trigger; when the value of k is small, stagnation can be easily detected. During the iteration process, C_n represents the total number of times that individuals in the population are in a stagnant condition, while *counter* represents the number of times that each person is in a stagnant state.

$$k_G = round[\frac{k_{max} - k_{min}}{nfes_{max}} \cdot nfes + k_{min}] \tag{33}$$

Algorithm 2 Calculate *counter* and C_n

$C_n = 0$
for $i = 1$ **to** PS **do**
  if $f(X_{i,G}) > f(U_{i,G})$ **then**
    $counter(i) = counter(i) + 1$
    $C_n = C_n + counter(i)$
  else
    $counter(i) = 0;$
  end if
end for

where k_{max} and k_{min} denote the maximum and minimum value of k, respectively. Here, k_{max} is set to 6 and k_{min} is set to 0.6. By combining all components, the pseudocode of FDHD-DE is presented in Algorithm 3.

Algorithm 3 Pseudocode of our algorithm

IPSut: Dimension number D, bound constraints $[l^D_{\min}, u^D_{\max}]$, maximum number of function evaluations $nfes_{\max}$, objective $f(X)$.
Output: Global best value $f(X_{gbest})$, global best individual X_{best}, number of function evaluations $nfes$;
Initialize the population $PS = \{X_1, X_2, ..., X_{PS}\}$, $PS = 25 \cdot log(D)\sqrt{D}$, $N = 20 \cdot D$, $\xi = 0.01$, $k_{min} = 0.6, k_{max} = 6, co = 0, C_n = 0$
Find the global best $X_{gbest,G}$ and fitness value $f(X_{gbest,G})$;
while $nfes < nfes_{\max}$ **do**
  Generate control parameters according to Eq. (23)-Eq. (25);
  Generate trial vectors;
  for $i = 1$ **to** PS **do**
    if $f(U_{i,G}) \le f(X_{i,G})$ **then**
      $X_{i,G+1} = U_{i,G};$
    else
      $X_{i,G+1} = X_{i,G}$
    end if
    if $f(U_{i,G} < f(X_{i,G})$ **then**
      $X_{i,G} \to A;$
      $CR_{i,G} \to S_{CR}$, $F_{i,G} \to S_F;$
    end if
  end for
  Update A, μ_F and μ_{CR};
  Calculate hypervolume diversity Eq. (31);
  if $nVOL < \xi$ **then**
    for $i = 1$ **to** PS **do**
      Perform algorithm 2.
      if $C_n > n$ **then**
        Perform dimension exchange mechanism;
        Calculate the fitness value $f(X_{i,G})$;
        $counter(i) = 0;$
        $nfes = nfes + 1;$
      end if
    end for
  end if
  Update population size according to Eq. (8);
end while

4. Experiment analysis

To verify the overall performance and FDHD-DE and avoid overfitting problem, FDHD-DE is tested on the totally 88 benchmark functions from CEC2013, CEC2014 and CEC2017 test suites. For the sake of clarity, we label the benchmark functions from CEC2013 as f_{a_1}-$f_{a_{28}}$, the benchmark functions from CEC2014 as f_{b_1}-$f_{b_{30}}$, and the benchmark functions from CEC2017 as f_{c_1}-$f_{c_{30}}$. For CEC2013 (Liang et al., 2013b), f_{a_1}-f_{a_5} are unimodal functions, f_{a_6}-$f_{a_{20}}$ are multimodal functions, and $f_{a_{21}}$-$f_{a_{28}}$ are composition functions. For CEC2014 (Liang et al., 2013a), f_{b_1}-f_{b_3} are unimodal functions, f_{b_4}-$f_{b_{16}}$ are simple multimodal functions, $f_{b_{17}}$-$f_{b_{22}}$ are hybrid functions, and $f_{b_{23}}$-$f_{b_{30}}$ are composition functions. For CEC2017 (Wu et al., 2017), f_{c_1}-f_{c_3} are in unimodal function group, f_{c_4}-$f_{c_{10}}$ are in simple multimodal function group, $f_{c_{11}}$-$f_{c_{20}}$ are hybrid function group, and $f_{c_{21}}$-$f_{c_{30}}$ are in composition function group. Five state-of-art DE variants LSHADE, jSO, PaDE, LPalmDE, Hip-DE have been used for performance comparison, and all parameters of the above variants are set according to their original papers (see Table 1). A total of 51 runs are conducted for each algorithm on each benchmark functions with the $nfes_{max}$ setting to $10000 \cdot D$. This experiment is carried out on a PC with Inter(R) core(TM) i7-7700HQ CPU Windows 10 operating system and the algorithm mean is implemented in Matlab 2021a version. We assume that the fitness error will be zero if it is less than “eps” (eps=2.2204E−16).

4.1. Algorithm comparison on optimization accuracy

In this subsection, the performance of FDHD-DE is analyzed in terms of optimization accuracy. The mean value (Mean) and standard deviation (Std) of total 51 runs are calculated, and Wilcoxon’s signed rank test is used for performance evaluation. The statistical results on CEC2013, CEC2014 and CEC2017 with dimensions ranging from 10 to 50 are shown in Table 2- Table 10. In order to verify the optimization performance of FDHD-DE on high-dimension optimization, Table 11 presents the comparison results of all algorithms on CEC2017 with 100D. The symbols “>”, “≈ ” and “<” denote that a given algorithm obtains “better performance”, “similar results ” or “worse results ” comparing with FDHD-DE, respectively.

In terms of 10D optimization, as shown in Table 2, Table 5, and Table 8, FDHD-DE perform slightly worse on the 88 benchmark functions than LSHADE, jSO, LPalmDE, PaDE, and Hip-DE, indicating that there exists room for improvement of FDHD-DE on low dimensional functions. Specifically, FDHD-DE obtains better or similar performance on 82 out of 88 function comparing with LSHADE, obtains better or similar performance on 78 functions comparing with jSO, obtains better or similar performance on 74 functions comparing with LPalmDE, obtains better or similar performance on 80 functions comparing with PaDE and 76 functions comparing with Hip-DE. It obtains the best performance on f_{a_7}, f_{a_9}, f_{a_7}, $f_{a_{15}}$, $f_{a_{19}}$ and $f_{a_{23}}$ under CEC2013 test suite, $f_{b_{11}}$, $f_{b_{14}}$, $f_{b_{16}}$ and $f_{b_{19}}$ under CEC2014 test suite and $f_{c_{10}}$, $f_{c_{17}}$, $f_{c_{23}}$ and $f_{c_{29}}$ under CEC2017 test suite.

In terms of 30D optimization, FDHD-DE obtains 63 better and similar performances compared with LSHADE with only 25 worse performances among the 88 function suites from CEC2013, CEC2014 and CEC2017; 76 better and similar performances compared with jSO and only 12 worse performances; 72 better and similar performances compared with LPalmDE and only 16 worse performances; 64 better and similar performances compared with LPalmDE; 73 better and similar ones compared with PaDE; 63 better and similar ones compared with Hip-DE. Moreover, it obtains the best performance on f_{a_1}, f_{a_6}, f_{a_7}, $f_{a_{15}}$, $f_{a_{23}}$ and $f_{a_{25}}$ under CEC2013 test suite, f_{b_1}, f_{b_4}, $f_{b_{14}}$, $f_{b_{16}}$-$f_{b_{21}}$, $f_{b_{24}}$, $f_{b_{25}}$, and $f_{b_{28}}$-$f_{b_{30}}$ under CEC2014 test suite and $f_{c_{13}}$, $f_{c_{14}}$, $f_{c_{15}}$, $f_{c_{17}}$-$f_{c_{19}}$, $f_{c_{23}}$, $f_{c_{24}}$ and $f_{c_{26}}$-$f_{c_{28}}$ under CEC2017 test suite.

In terms of 50D optimization, among comparison results on 88 benchmark functions, there are 69 better and similar performances compared with LSHADE; 77 better and similar ones comparing with jSO; 73 better and similar ones in comparison with LPalmD; 73 better or similar ones in comparison with PaDE; and 63 better and similar ones compared with Hip-DE. As dimensionality increases, FDHD-DE demonstrates obvious performance improvement owing to its semi-adaptive scheme and diversity improvement mechanism. Furthermore, it obtains the best performance on f_{a_2}, f_{a_4}, f_{a_9}, $f_{a_{10}}$, $f_{a_{12}}$, $f_{a_{13}}$ and $f_{a_{20}}$ under CEC2013 test suite, f_{b_1}, f_{b_4}, $f_{b_{14}}$, $f_{b_{16}}$-$f_{b_{18}}$, $f_{b_{20}}$, $f_{b_{21}}$, $f_{b_{24}}$, $f_{b_{25}}$, $f_{b_{27}}$ and $f_{b_{29}}$ under CEC2014 test suite and f_{c_2}-f_{c_4}, $f_{c_{10}}$-$f_{c_{16}}$, $f_{c_{18}}$-$f_{c_{20}}$, $f_{c_{23}}$-$f_{c_{28}}$ and $f_{c_{30}}$ under CEC2017 test suite.

Table 1
Recommended Parameter settings of all these contrasted algorithms.

Algorithm	Parameters initial settings
LSHADE	$\mu_F = 0.5,\ \mu_{CR} = 0.5,\ F\&CR$ same as JADE, $PS = 18 \cdot D \sim 4,\ r^{rac} = 2.6,\ p = 0.11,\ H = 6$
jSO	$F, CR\&r^{rac}$ same as iLSHADE, $\mu_F = 0.3,\ \mu_{CR} = 0.8,\ PS = 25 \cdot log(D) \cdot \sqrt{D} \sim 4,\ p = 0.25 \sim 0.125,\ H = 5$
LPalmDE	$\mu_{F_k} = \mu_{CR_k} = 0.5,\ F \sim CR(\mu_{F_k}, 0.2),\ CR \sim N(\mu_{CR_k}, 0.1),\ p = 0.11,\ k = 6,\ PS = 23 \cdot D \sim K,\ r^{rac} = 1.6,\ T_0 = 70$
PaDE	$\mu_F = 0.8,\ \mu_{CR} = 0.6,\ F\&CR$ same as LSHADE, $p = 0.11,\ PS = 25 \cdot log(D) \cdot \sqrt{D} \sim 4,\ r^{rac} = 1.6,\ r^d = 0.04,\ T_0 = 70,\ K = 4$
Hip-DE	$\mu_{F_k} = 0.6,\ \mu_{CR_k} = 0.8,\ F \sim CR(\mu_{F_k}, 0.1),\ CR \sim N(\mu_{CR_k}, 0.1),\ p = 0.2 \sim 0.05,\ k = 6,\ PS = 15 \cdot D \sim K,\ r^{rac} = 5,\ \tau_F = \tau_{CR} = 0.9,\ C = 0.1,\ N = \lceil r^p \cdot nfes_{max}/ps \rceil$
FDHD-DE	$F = 0.5,\ CR = 0.8,\ p = 0.11,\ r^{rac} = 1.4,\ PS = 25 \cdot log(D) \cdot \sqrt{D} \sim 4,\ H = 4,\ nfes_{max} = D \cdot 1000,\ k_{min} = 0.6,\ k_{max} = 6,\ co = 0,\ C_n = 0$

Table 2
Comparisons between LSHADE, jSO, LPalmDE, PaDE, Hip-DE and FDHD-DE on CEC2013 with 10D.

DE Variants	LSHADE	jSO	LPalmDE	PaDE	Hip-DE	FDHD-DE
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{a_1}	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0*
f_{a_2}	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0*
f_{a_3}	4.1975E−03/1.6957E−02(≈)	**3.5832E−09/1.4064E−08**(≈)	6.9958E−03/2.1430E−02(≈)	6.9958E−03/2.1430E−02(≈)	1.3992E−02/9.9920E−03(<)	1.1193E−02/2.6209E−02
f_{a_4}	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0*
f_{a_5}	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0*
f_{a_6}	4.6176E+00/4.9465E+00(<)	**7.6960E−01/2.6643E+00**(≈)	3.6556E+00/4.7914E+00(<)	1.9240E+00/3.9346E+00(≈)	3.8480E+00/4.8384E+00(≈)	1.9240E+00/3.9346E+00
f_{a_7}	7.2136E−06/1.9848E−05(≈)	8.5429E−06/1.8093E−05(<)	1.1806E−05/4.1382E−05(<)	6.8743E−06/2.2339E−05(<)	5.8196E−06/1.0527E−05(<)	**1.7531E−06/6.6543E−06**
f_{a_8}	2.0219E+01/1.5563E−01(≈)	2.0356E+01/6.7083E−02(<)	**2.0170E+01/1.5794E−01**(≈)	2.0185E+01/1.6226E−01(≈)	2.0190E+01/1.6379E−01(<)	2.0181E+01/1.3539E−01
f_{a_9}	2.7360E+00/1.3627E+00(<)	5.7642E−01/9.5119E−01(<)	6.2155E−01/9.6573E−01(<)	1.1113E+00/1.3708E+00(<)	5.2625E−01/7.5240E−01(≈)	**2.7563E−01/5.0967E−01**
$f_{a_{10}}$	2.3190E−03/6.1405E−03(>)	**7.7336E−04/2.3894E−03**(>)	1.1590E−03/3.7296E−03(>)	4.8761E−03/1.2935E−02(>)	1.9800E−03/5.9716E−03(>)	1.3025E−02/1.2776E−02
$f_{a_{11}}$	*0/0(≈)*	3.9018E−02/1.9505E−01(≈)	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0*
$f_{a_{12}}$	2.0135E+00/1.0228E+00(≈)	2.3021E+00/7.5748E−01(≈)	2.3801E+00/1.0162E+00(≈)	2.5456E+00/1.3641E+00(≈)	**1.6059E+00/7.1994E−01**(>)	2.3454E+00/1.1390E+00
$f_{a_{13}}$	1.6642E+00/8.1532E−01(≈)	2.1423E+00/1.4255E+00(≈)	2.3059E+00/1.6713E+00(≈)	1.7455E+00/1.1427E+00(≈)	**1.4554E+00/6.0909E−01**(≈)	1.6838E+00/8.3131E−01
$f_{a_{14}}$	1.8369E−02/3.8080E−02(≈)	7.1442E−02/7.9916E−02(<)	**3.6738E−03/1.4841E−02**(>)	1.3471E−02/3.3780E−02(≈)	7.3476E−02/2.0322E−02(<)	2.5717E−02/3.9847E−02
$f_{a_{15}}$	3.1495E+02/1.0363E+02(<)	2.7237E+02/1.2418E+02(≈)	3.8940E+02/1.6165E+02(<)	3.2704E+02/1.3075E+02(<)	3.1918E+02/1.5903E+02(<)	**2.4242E+02/1.2524E+02**
$f_{a_{16}}$	2.7225E−01/1.5365E−01(<)	1.0406E+00/2.2261E−01(<)	1.6171E−01/1.5357E−01(≈)	2.7084E−01/1.8882E−01(<)	**1.1344E−01/1.2949E−01**(>)	1.9600E−01/1.3447E−01
$f_{a_{17}}$	*1.0122E+01/1.3510E−14(≈)*	1.0127E+01/2.4195E−02(<)	*1.0122E+01/1.5461E−14(≈)*	*1.0122E+01/1.8501E−14(≈)*	*1.0122E+01/2.3613E−14(≈)*	*1.0122E+01/1.9768E−14*
$f_{a_{18}}$	**1.3629E+01/9.6040E−01**(≈)	1.4179E+01/1.7076E+00(≈)	1.4739E+01/1.8785E+00(<)	1.5038E+01/2.0778E+00(<)	1.3952E+01/1.4635E+00(>)	1.3814E+01/1.1604E+00
$f_{a_{19}}$	2.2444E−01/3.8321E−02(≈)	2.6474E−01/5.0905E−02(<)	2.7790E−01/8.1843E−02(<)	**2.2391E−01/3.2526E−02**(≈)	2.2487E−01/3.7121E−02(≈)	2.2745E−01/4.8866E−02
$f_{a_{20}}$	2.0010E+00/3.4096E−01(<)	1.7192E+00/4.0347E−01(<)	1.7308E+00/3.2635E−01(<)	1.6365E+00/2.5202E−01(<)	1.4847E+00/3.3121E−01(≈)	**1.3848E+00/3.0521E−01**
$f_{a_{21}}$	*4.0019E+02/0(≈)*	*4.0019E+02/0(≈)*	*4.0019E+02/0(≈)*	*4.0019E+02/0(≈)*	*4.0019E+02/0(≈)*	*4.0019E+02/0*
$f_{a_{22}}$	9.6797E+00/1.9383E+01(<)	1.2013E+01/1.9897E+01(<)	**3.2694E+00/3.7856E+00**(≈)	3.7927E+00/3.9378E+00(≈)	3.8246E+00/1.4344E+01(≈)	5.3366E+00/1.4011E+01
$f_{a_{23}}$	2.9199E+02/1.3540E+02(<)	2.5189E+02/1.3910E+02(<)	3.0156E+02/1.6852E+02(<)	2.5629E+02/1.3776E+02(<)	2.2225E+02/1.3975E+02(<)	**1.3414E+02/9.1911E+01**
$f_{a_{24}}$	2.0131E+02/1.4447E+01(<)	2.0161E+02/2.9314E+00(<)	2.0000E+02/0(≈)	**1.9895E+02/7.4720E+00**(≈)	2.0000E+02/6.9040E−06(≈)	2.0000E+02/0
$f_{a_{25}}$	2.0089E+02/1.8205E+00(<)	**1.9839E+02/1.1516E+01**(≈)	2.0000E+02/0(≈)	2.0000E+02/3.2155E−14(≈)	2.0000E+02/1.1191E−04(≈)	2.0000E+02/0
$f_{a_{26}}$	1.3861E+02/4.7794E+01(<)	1.0722E+02/2.0162E+01(≈)	1.0824E+02/2.3217E+01(≈)	1.1020E+02/2.6558E+01(≈)	**1.0535E+02/1.9328E+01**(>)	1.1437E+02/3.1947E+01
$f_{a_{27}}$	3.0356E+02/2.5404E+01(≈)	*3.0000E+02/0(≈)*	*3.0000E+02/0(≈)*	*3.0000E+02/0(≈)*	*3.0000E+02/0(≈)*	*3.0000E+02/0*
$f_{a_{28}}$	**2.9608E+02/2.8006E+01**(≈)	3.0000E+02/0(≈)	3.0000E+02/0(≈)	3.0000E+02/0(≈)	3.0000E+02/0(≈)	3.0000E+02/0
>/≈/<	1/17/10	1/16/11	2/18/8	1/20/7	5/17/6	−/−/−

Table 3
Comparisons between LSHADE, jSO, LPalmDE, PaDE, Hip-DE and FDHD-DE on CEC2013 with 30D.

DE Variants	LSHADE	jSO	LPalmDE	PaDE	Hip-DE	FDHD-DE
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{a_1}	*0.0000E+00/0.0000E+00(≈)*	*0.0000E+00/0.0000E+00(≈)*	*0.0000E+00/0.0000E+00(≈)*	*0.0000E+00/0.0000E+00(≈)*	*0.0000E+00/0.0000E+00(≈)*	*0.0000E+00/0.0000E+00*
f_{a_2}	3.1654E−13/2.5756E−13(<)	1.3390E−10/2.5586E−10(<)	8.2033E−13/1.4032E−12(<)	3.7004E−13/2.6501E−13(<)	6.6429E−13/2.4539E−12(<)	**4.9041E−14/9.4449E−14**
f_{a_3}	1.2898E−01/6.9024E−01(<)	**8.3236E−09/5.4693E−08**(>)	1.0880E+00/7.4563E+00(<)	6.5204E−01/4.6545E+00(<)	5.9494E−01/2.5762E+00(<)	1.4497E−06/6.1140E−06
f_{a_4}	5.7958E−14/1.0008E−13(<)	1.5961E−12/9.2470E−13(<)	1.0700E−13/1.1462E−13(<)	6.2416E−14/1.0248E−13(<)	**2.2292E−14/6.8286E−14**(>)	3.5666E−14/8.3512E−14
f_{a_5}	1.1369E−13/0.0000E+00(≈)	*1.1146E−13/1.5919E−14*(>)	*1.1146E−13/1.5919E−14*(>)	1.1369E−13/0.0000E+00(≈)	1.1592E−13/1.5919E−14(<)	1.1369E−13/0.0000E+00
f_{a_6}	8.3077E−09/4.4229E−08(<)	7.2368E−10/2.4841E−09(<)	6.3286E−12/4.0944E−11(<)	1.6946E−09/1.2084E−08(<)	6.4917E−11/4.4844E−10(<)	**1.3375E−13/4.3771E−14**
f_{a_7}	6.0879E−01/4.4764E−01(<)	1.8790E−02/4.4427E−02(<)	1.8771E−01/2.8314E−01(<)	1.1863E−01/1.5182E−01(<)	1.2862E−01/1.6020E−01(<)	**6.8211E−03/1.2138E−02**
f_{a_8}	2.0842E+01/1.4562E−01(<)	2.0937E+01/5.9054E−02(<)	2.0800E+01/1.3129E−01(>)	2.0884E+01/1.9294E−01(<)	**2.0713E+01/1.7341E−01**(>)	2.0816E+01/1.4339E−01
f_{a_9}	2.6654E+01/1.4454E+00(<)	2.3500E+01/3.5501E+00(<)	**1.8694E+01/4.0730E+00**(>)	2.5580E+01/2.4595E+00(<)	2.5445E+01/2.1807E+00(<)	2.2517E+01/5.4305E+00
$f_{a_{10}}$	5.8008E−04/2.0082E−03(<)	*0.0000E+00/0.0000E+00(≈)*	7.2472E−04/2.6666E−03(<)	4.8347E−04/1.9738E−03(<)	1.4502E−04/1.0357E−03(<)	*0.0000E+00/0.0000E+00*
$f_{a_{11}}$	7.4677E−14/4.1756E−14(>)	1.6161E−13/5.9480E−14(<)	3.4552E−14/2.8029E−14(>)	7.0218E−14/4.3422E−14(>)	**1.8948E−14/2.7063E−14**(>)	1.2706E−13/6.1840E−14
$f_{a_{12}}$	**5.5397E+00/1.7675E+00**(>)	9.0362E+00/2.5876E+00(<)	9.5214E+00/2.8088E+00(<)	6.3250E+00/1.4416E+00(>)	7.9854E+00/1.2339E+00(<)	7.2306E+00/2.0541E+00
$f_{a_{13}}$	6.0977E+00/3.2031E+00(>)	9.9228E+00/4.7778E+00(<)	1.3452E+01/6.0810E+00(<)	7.4727E+00/3.5582E+00(>)	**5.2605E+00/2.5633E+00**(>)	9.4042E+00/4.0041E+00
$f_{a_{14}}$	2.4901E−02/2.3191E−02(>)	1.1759E+01/5.5189E+00(<)	**6.1233E−03/1.1992E−02**(>)	2.8167E−02/2.2355E−02(>)	2.3677E−02/2.6668E−02(>)	4.7430E−02/3.2516E−02
$f_{a_{15}}$	2.6611E+03/2.7629E+02(<)	2.7096E+03/2.9775E+02(<)	2.8517E+03/4.0277E+02(<)	2.8716E+03/3.5794E+02(<)	2.6543E+03/3.0470E+02(<)	**2.5777E+03/2.6880E+02**
$f_{a_{16}}$	7.5627E−01/1.4254E−01(<)	2.2904E+00/3.1035E−01(<)	5.6364E−01/3.0153E−01(>)	6.1399E−01/4.1010E−01(>)	**3.1942E−01/2.5024E−01**(>)	6.9552E−01/2.0987E−01
$f_{a_{17}}$	*3.0434E+01/9.4299E−07(≈)*	3.0673E+01/1.1707E−01(<)	*3.0434E+01/1.8285E−06(≈)*	*3.0434E+01/3.9825E−14(≈)*	*3.0434E+01/5.4965E−14(≈)*	*3.0434E+01/1.8283E−06*
$f_{a_{18}}$	5.1891E+01/3.5319E+00(>)	5.6899E+01/4.7627E+00(<)	**4.5282E+01/3.9113E+00**(>)	5.5458E+01/4.3863E+00(<)	4.9676E+01/3.2152E+00(>)	5.5130E+01/3.8280E+00
$f_{a_{19}}$	1.1673E+00/8.6400E−02(>)	1.2583E+00/9.2718E−02(<)	**1.1089E+00/1.6100E−01**(>)	1.2157E+00/8.4206E−02(<)	1.2068E+00/9.1206E−02(<)	1.1921E+00/1.0355E−01
$f_{a_{20}}$	1.0739E+01/1.7551E+00(<)	9.5726E+00/3.9013E−01(<)	9.1786E+00/4.0029E−01(<)	1.0609E+01/1.9673E+00(<)	**9.0488E+00/3.5762E−01**(>)	9.0893E+00/3.3421E−01
$f_{a_{21}}$	**2.9608E+02/1.9604E+01**(>)	3.0342E+02/4.9361E+01(>)	2.9804E+02/1.4003E+01(>)	3.0481E+02/2.0100E+01(<)	3.0685E+02/2.4726E+01(<)	3.0367E+02/3.1788E+01
$f_{a_{22}}$	1.0831E+02/2.5461E+00(<)	1.1945E+02/3.8193E+00(<)	1.0595E+02/3.2666E−01(>)	**1.0589E+02/4.4673E−01**(>)	1.0590E+02/3.7285E−01(>)	1.0676E+02/1.3394E+00
$f_{a_{23}}$	2.5258E+03/2.8207E+02(<)	2.5135E+03/3.2438E+02(<)	2.7840E+03/4.7290E+02(<)	2.6533E+03/3.6075E+02(<)	2.5666E+03/3.0098E+02(<)	**2.4546E+03/3.1022E+02**
$f_{a_{24}}$	2.0047E+02/1.1076E+00(<)	2.0004E+02/4.9193E−02(>)	2.0003E+02/8.6633E−02(>)	*2.0001E+02/3.5342E−02*(>)	*2.0001E+02/2.4049E−02*(>)	2.0011E+02/1.1008E−01
$f_{a_{25}}$	2.4096E+02/6.9199E+00(<)	2.3860E+02/4.7024E+00(<)	2.2094E+02/2.2883E+01(<)	2.2318E+02/2.2352E+01(<)	2.2045E+02/2.3039E+01(<)	**2.0514E+02/1.4088E+01**
$f_{a_{26}}$	*2.0000E+02/0.0000E+00(≈)*	*2.0000E+02/1.4171E−13(≈)*	*2.0000E+02/1.5891E−13(≈)*	*2.0000E+02/3.2155E−14(≈)*	*2.0000E+02/3.2155E−14(≈)*	*2.0000E+02/3.2155E−14*
$f_{a_{27}}$	3.0136E+02/3.0731E+00(>)	3.0092E+02/1.1319E+00(>)	3.0079E+02/1.4309E+00(>)	**3.0026E+02/5.1130E−01**(>)	3.0034E+02/6.8807E−01(>)	3.0241E+02/2.1984E+00
$f_{a_{28}}$	*3.0000E+02/0.0000E+00(≈)*	*3.0000E+02/6.4311E−14(≈)*	*3.0000E+02/1.0168E−13(≈)*	*3.0000E+02/0.0000E+00(≈)*	*3.0000E+02/0.0000E+00(≈)*	*3.0000E+02/0.0000E+00*
>/≈/<	8/5/15	5/4/19	12/4/12	8/5/15	11/4/13	−/−/−

In order to verify the optimization accuracy of our algorithm for high-dimensional scenario, FDHD-DE is tested on 30 benchmark functions from CEC2017 with 100 dimensions. The comparison results are shown in Table 11. From Table 11, one can see that FDHD-DE perform better on high-dimensional optimization problems. Comparing with LSHADE, FDHD-DE obtains better or similar performances on 21 out of 30 benchmark functions; comparing with jSO, FDHD-DE exhibits better or similar performance on 22 benchmark functions; in comparison

Table 4
Comparisons between LSHADE, jSO, LPalmDE, PaDE, Hip-DE and FDHD-DE on CEC2013 with 50D.

DE Variants	LSHADE	jSO	LPalmDE	PaDE	Hip-DE	FDHD-DE
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{a_1}	4.9041E−14/9.4449E−14(≈)	5.7958E−14/1.0008E−13(≈)	6.6875E−14/1.0463E−13(≈)	2.6750E−14/7.3986E−14(>)	**0/0**(>)	6.6875E−14/1.0463E−13
f_{a_2}	6.7423E+02/8.6125E+02(<)	4.0540E+01/7.2598E+01(<)	3.9424E+03/5.1849E+03(<)	2.0307E+03/2.3340E+03(<)	2.1364E+03/2.2367E+03(<)	**2.5656E−09/1.0370E−08**
f_{a_3}	1.2671E+04/6.5125E+04(<)	**2.9443E+00/1.4549E+01**(>)	2.8044E+03/8.3476E+03(≈)	1.7408E+03/5.0009E+03(≈)	2.2823E+03/1.5217E+04(≈)	2.1091E+02/4.6605E+02
f_{a_4}	7.9086E−11/2.0827E−10(<)	1.5703E−08/2.5171E−08(<)	3.1994E−10/6.4786E−10(<)	1.1436E−11/9.6351E−12(<)	8.1498E−12/5.0062E−12(<)	**3.1030E−12/1.6577E−12**
f_{a_5}	**1.4044E−13/4.8704E−14**(≈)	1.4489E−13/5.1240E−14(≈)	1.6050E−13/5.6508E−14(≈)	1.8056E−13/5.6508E−14(≈)	2.2069E−13/3.5311E−14(<)	1.6273E−13/5.6866E−14
f_{a_6}	*4.3447E+01/0*(≈)	*4.3447E+01/1.6078E−14*(≈)	*4.3447E+01/2.7019E−14*(≈)	*4.3447E+01/0*(≈)	*4.3447E+01/2.7019E−14*(≈)	*4.3447E+01/1.6078E−14*
f_{a_7}	2.1970E+00/1.5443E+00(<)	**1.1096E−01/9.8286E−02**(>)	1.3414E+00/1.1895E+00(<)	1.1727E+00/9.9422E−01(<)	5.9187E−01/5.3267E−01(<)	1.9974E−01/1.0966E−01
f_{a_8}	2.1081E+01/1.0125E−01(<)	2.1137E+01/3.6389E−02(<)	2.1034E+01/1.0688E−01(≈)	2.0986E+01/1.5825E−01(≈)	**2.0889E+01/1.8848E−01**(>)	2.1025E+01/1.3231E−01
f_{a_9}	5.3181E+01/1.7825E+00(<)	4.7390E+01/6.0204E+00(<)	4.1004E+01/5.4078E+00(<)	4.8642E+01/7.9741E+00(<)	4.0702E+01/1.2230E+01(<)	**2.6300E+01/1.2441E+01**
$f_{a_{10}}$	9.9034E−03/8.2053E−03(<)	1.3536E−03/2.9715E−03(<)	1.1206E−02/9.8886E−03(<)	1.1783E−02/9.7117E−03(<)	8.3560E−03/8.5167E−03(<)	**4.9041E−14/1.9755E−14**
$f_{a_{11}}$	6.9846E−11/8.9419E−11(>)	6.6023E−09/1.0054E−08(<)	**8.5822E−14/4.6001E−14**(>)	3.5444E−13/7.9246E−14(>)	2.0731E−13/9.2934E−14(>)	2.1223E−10/1.9721E−10
$f_{a_{12}}$	1.4404E+01/2.5423E+00(≈)	1.8127E+01/4.0285E+00(<)	2.2598E+01/5.1639E+00(<)	1.7637E+01/2.3633E+00(<)	1.4549E+01/2.9729E+00(≈)	**1.3472E+01/2.7012E+00**
$f_{a_{13}}$	2.1767E+01/8.3792E+00(<)	2.6687E+01/1.2368E+01(<)	4.8402E+01/1.5148E+01(<)	3.2929E+01/1.1175E+01(<)	2.5718E+01/6.9772E+00(<)	**1.7068E+01/5.9493E+00**
$f_{a_{14}}$	2.2927E−01/1.1510E−01(>)	6.2689E+01/1.6273E+01(<)	1.2305E−01/6.1027E−02(>)	**6.2610E−02/2.1402E−02**(>)	1.1777E−01/3.2191E−02(>)	5.4905E−01/3.3251E−01
$f_{a_{15}}$	6.3644E+03/4.5097E+02(≈)	6.3182E+03/3.9725E+02(≈)	**6.0941E+03/5.7480E+02**(>)	6.3949E+03/4.0486E+02(≈)	6.3772E+03/4.0331E+02(≈)	6.3402E+03/3.9270E+02
$f_{a_{16}}$	1.1599E+00/2.0453E−01(≈)	3.1041E+00/5.1843E−01(<)	8.5795E−01/3.5467E−01(>)	1.0205E+00/5.2379E−01(≈)	**6.1986E−01/4.3286E−01**(>)	1.1536E+00/2.3193E−01
$f_{a_{17}}$	5.0788E+01/4.4648E−03(>)	5.2455E+01/3.4576E−01(<)	*5.0786E+01/2.7668E−10*(>)	*5.0786E+01/4.1392E−09*(>)	*5.0786E+01/2.6464E−03*(>)	5.0802E+01/1.3266E−02
$f_{a_{18}}$	1.0282E+02/6.5621E+00(>)	1.1094E+02/7.7117E+00(≈)	**7.5292E+01/8.8673E+00**(>)	1.0465E+02/6.7557E+00(>)	9.9416E+01/4.6798E+00(>)	1.0985E+02/7.6487E+00
$f_{a_{19}}$	2.5539E+00/1.3281E−01(≈)	2.6398E+00/1.8963E−01(<)	2.3650E+00/3.0153E−01(>)	**2.3119E+00/1.4224E−01**(>)	2.4978E+00/1.4018E−01(≈)	2.5482E+00/1.9320E−01
$f_{a_{20}}$	1.8164E+01/5.1610E−01(<)	1.8836E+01/4.8032E−01(<)	1.7691E+01/5.7769E−01(≈)	1.7824E+01/7.7552E−01(<)	1.7773E+01/4.4979E−01(<)	**1.7567E+01/4.6694E−01**
$f_{a_{21}}$	7.6054E+02/4.5472E+02(≈)	**7.6004E+02/3.9949E+02**(≈)	9.8441E+02/3.2088E+02(≈)	9.6632E+02/3.3846E+02(≈)	8.1606E+02/4.2597E+02(≈)	8.3975E+02/4.2075E+02
$f_{a_{22}}$	1.3713E+01/1.3369E+00(>)	5.5648E+01/1.0764E+01(<)	**1.1485E+01/7.7771E−01**(>)	1.1799E+01/2.9569E−01(>)	1.1718E+01/5.6189E−01(>)	1.4539E+01/1.3144E+00
$f_{a_{23}}$	5.8224E+03/3.3537E+02(≈)	**5.5551E+03/5.6912E+02**(>)	6.0379E+03/5.7537E+02(≈)	6.2467E+03/4.9915E+02(<)	6.1175E+03/4.6767E+02(<)	5.8955E+03/4.7938E+02
$f_{a_{24}}$	2.1102E+02/4.9491E+00(<)	**2.0055E+02/6.3083E−01**(>)	2.0505E+02/3.6720E+00(<)	2.0357E+02/3.0209E+00(<)	2.0116E+02/1.0748E+00(>)	2.0197E+02/6.7757E−01
$f_{a_{25}}$	2.7714E+02/6.1710E+00(>)	**2.7652E+02/7.5487E+00**(>)	2.8794E+02/6.6974E+00(≈)	2.8668E+02/7.1483E+00(≈)	2.8497E+02/5.9786E+00(>)	2.8906E+02/4.9657E+00
$f_{a_{26}}$	2.5410E+02/5.3639E+01(≈)	**2.1214E+02/3.3582E+01**(>)	2.6044E+02/5.3225E+01(≈)	2.5842E+02/4.7252E+01(≈)	2.5451E+02/5.1935E+01(≈)	2.5795E+02/5.3066E+01
$f_{a_{27}}$	4.1463E+02/8.2186E+01(<)	3.3410E+02/6.3233E+01(≈)	3.6513E+02/3.2188E+01(<)	3.4896E+02/3.7368E+01(<)	**3.2085E+02/1.3853E+01**(≈)	3.3310E+02/6.3233E+01
$f_{a_{28}}$	*4.0000E+02/0*(≈)	*4.0000E+02/2.8433E−13*(≈)	*4.0000E+02/2.8705E−13*(≈)	*4.0000E+02/0*(≈)	*4.0000E+02/7.1902E−14*(≈)	*4.0000E+02/2.8433E−13*
>/≈/<	6/11/11	6/8/14	8/11/9	7/10/11	10/9/9	−/−/−

Table 5
Comparisons between LSHADE, jSO, LPalmDE, PaDE, Hip-DE and FDHD-DE on CEC2014 with 10D.

DE Variants	LSHADE	jSO	LPalmDE	PaDE	Hip-DE	FDHD-DE
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{b_1}	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*
f_{b_2}	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*
f_{b_3}	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*
f_{b_4}	3.0689E+01/1.1317E+01(≈)	**2.3272E+01/1.6448E+01**(>)	2.8046E+01/1.3785E+01(≈)	2.5915E+01/1.5308E+01(≈)	3.0689E+01/1.1317E+01(≈)	3.1370E+01/1.0445E+01
f_{b_5}	**1.3141E+01/9.0364E+00**(≈)	1.5412E+01/8.3545E+00(<)	1.5641E+01/8.2875E+00(≈)	1.3163E+01/8.5960E+00(≈)	1.4699E+01/8.6864E+00(≈)	1.6209E+01/7.7239E+00
f_{b_6}	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*
f_{b_7}	1.7420E−03/4.4667E−03(>)	1.6431E−03/3.6776E−03(>)	**4.3468E−04/2.2947E−03**(>)	4.6365E−03/1.1938E−02(>)	4.3507E−04/1.7576E−03(>)	9.8526E−03/1.1103E−02
f_{b_8}	*0/0*(≈)	1.1705E−01/3.8002E−01(<)	*0/0*(≈)	*0/0*(≈)	*0/0*(≈)	*0/0*
f_{b_9}	2.4208E+00/7.5098E−01(≈)	2.5171E+00/9.6207E−01(≈)	2.7898E+00/1.4352E+00(≈)	2.0756E+00/7.8311E−01(≈)	**1.8555E+00/8.4424E−01**(≈)	2.3631E+00/1.7565E+00
$f_{b_{10}}$	4.8984E−03/1.6958E−02(≈)	4.1477E+00/1.7626E+01(<)	**0/0**(>)	1.2246E−03/8.7454E−03(>)	1.2246E−03/8.7454E−03(>)	1.3471E−02/2.8793E−02
$f_{b_{11}}$	2.4558E+01/2.6676E+01(<)	6.1856E+01/8.1500E+01(<)	6.2484E+01/6.8591E+01(<)	2.4930E+01/3.6758E+01(<)	3.5211E+01/5.9586E+01(<)	**1.9588E+01/3.5924E+01**
$f_{b_{12}}$	6.8479E−02/1.7223E−02(≈)	2.5648E−01/1.3359E−01(<)	**3.4043E−02/3.7980E−02**(>)	9.7508E−02/3.0445E−02(<)	5.0180E−02/3.5362E−02(≈)	6.2531E−02/2.9320E−02
$f_{b_{13}}$	5.0051E−02/1.1646E−02(≈)	7.9091E−02/1.8238E−02(<)	**3.5755E−02/1.7542E−02**(>)	4.9104E−02/1.7107E−02(≈)	4.8281E−02/1.9252E−02(<)	4.7642E−02/1.3907E−02
$f_{b_{14}}$	7.5209E−02/2.6312E−02(<)	6.1650E−02/1.6238E−02(<)	8.9253E−02/3.0849E−02(<)	8.1607E−02/2.7056E−02(<)	9.3147E−02/3.3580E−02(<)	**5.1307E−02/1.8735E−02**
$f_{b_{15}}$	3.6179E−01/7.3429E−02(>)	4.9525E−01/1.0105E−01(<)	4.4600E−01/1.3391E−01(<)	4.2420E−01/9.2191E−02(≈)	**3.3934E−01/8.2343E−02**(>)	3.9764E−01/6.7111E−02
$f_{b_{16}}$	1.2714E+00/2.7142E−01(<)	1.1900E+00/3.5823E−01(<)	1.0983E+00/5.2778E−01(≈)	1.0588E+00/3.1668E−01(≈)	1.0054E+00/3.2978E−01(≈)	**9.4673E−01/3.1490E−01**
$f_{b_{17}}$	9.7500E−01/9.2903E−01(≈)	3.5317E+00/4.2597E+00(<)	**7.9561E−01/6.5280E−01**(≈)	9.5655E−01/7.6236E−01(<)	1.2062E+00/1.0777E+00(<)	9.5477E−01/2.2002E+00
$f_{b_{18}}$	2.1998E−01/1.8881E−01(<)	1.4116E−01/1.1412E−01(<)	**5.3948E−02/6.4774E−02**(≈)	1.1391E−01/1.3267E−01(≈)	1.2709E−01/2.3341E−01(≈)	9.2258E−02/1.3280E−01
$f_{b_{19}}$	1.0115E−01/1.3503E−01(<)	8.0269E−02/1.7647E−01(≈)	5.6308E−02/3.4264E−02(<)	6.6296E−02/1.3412E−01(≈)	7.5280E−02/5.5976E−02(<)	**4.3473E−02/2.0841E−02**
$f_{b_{20}}$	**1.2792E−01/1.0690E−01**(≈)	2.4207E−01/1.7670E−01(<)	1.4108E−01/1.5277E−01(≈)	1.4734E−01/1.3548E−01(≈)	2.4634E−01/1.9335E−01(≈)	1.6503E−01/1.7956E−01
$f_{b_{21}}$	3.6798E−01/3.0177E−01(≈)	5.4473E−01/2.8940E−01(<)	**1.8875E−01/2.5821E−01**(>)	2.3427E−01/2.6289E−01(>)	2.9136E−01/2.8283E−01(≈)	3.8774E−01/3.1316E−01
$f_{b_{22}}$	8.4093E−02/3.9896E−02(≈)	1.1172E+00/3.9532E+00(<)	**3.4310E−02/4.1181E−02**(>)	7.3644E−02/3.0057E−02(>)	8.9447E−02/3.3048E−02(≈)	9.1325E−02/3.4488E−02
$f_{b_{23}}$	*3.2946E+02/0*(≈)	*3.2946E+02/0*(≈)	*3.2946E+02/2.8705E−13*(≈)	*3.2946E+02/0*(≈)	*3.2946E+02/0*(≈)	*3.2946E+02/0*
$f_{b_{24}}$	1.0767E+02/1.5689E+00(≈)	1.0774E+02/2.6073E+00(<)	1.0741E+02/3.1532E+00(≈)	1.0734E+02/2.4027E+00(≈)	**1.0522E+02/3.2771E+00**(>)	1.0659E+02/2.8634E+00
$f_{b_{25}}$	1.3657E+02/3.9957E+01(≈)	**1.2222E+02/2.3627E+01**(>)	1.3970E+02/3.5902E+01(≈)	1.2341E+02/3.1761E+01(>)	1.4733E+02/3.3534E+01(<)	1.3862E+02/3.5755E+01
$f_{b_{26}}$	1.0005E+02/1.4497E−02(≈)	1.0008E+02/1.4378E−02(<)	**1.0003E+02/1.3563E−02**(>)	1.0005E+02/1.9188E−02(≈)	1.0005E+02/2.2598E−02(≈)	1.0005E+02/1.7263E−02
$f_{b_{27}}$	7.9525E+01/1.4408E+02(<)	6.3837E+01/1.2903E+02(≈)	**7.2412E+00/4.1816E+01**(≈)	3.4553E+01/1.0290E+02(≈)	5.9910E+01/1.1975E+02(≈)	7.9321E+01/1.5230E+02
$f_{b_{28}}$	3.8581E+02/3.9603E+01(<)	**3.7530E+02/3.9807E+01**(≈)	3.8991E+02/5.3111E+01(≈)	3.9377E+02/5.3559E+01(≈)	3.9583E+02/5.3116E+01(≈)	3.8187E+02/4.7127E+01
$f_{b_{29}}$	2.2201E+02/4.6507E−01(≈)	*2.2181E+02/2.3190E−01*(>)	*2.2181E+02/2.2898E−01*(>)	2.2184E+02/2.9099E−01(<)	2.2383E+02/2.6392E−01(<)	2.2203E+02/4.4498E−01
$f_{b_{30}}$	4.6624E+02/1.1572E+01(≈)	4.6649E+02/9.9155E+00(≈)	4.6974E+02/1.1657E+01(≈)	4.7097E+02/1.5935E+01(≈)	**4.6372E+02/2.6359E+00**(≈)	4.6846E+02/1.3582E+01
>/≈/<	2/21/7	4/10/16	8/17/5	5/20/5	4/19/7	−/−/−

with LPalmDE, FDHD-DE obtains better or similar performance on 25 functions; comparing with PaDE, FDHD-DE obtains better or similar performance on 23 function; comparing with Hip-DE, FDHD-DE obtains better or similar performances on 25 functions. Specifically, FDHD-DE secures the best performance on f_{c_7}, $f_{c_{10}}$-$f_{c_{15}}$, $f_{c_{17}}$-$f_{c_{19}}$, $f_{c_{22}}$ and $f_{c_{24}}$-$f_{c_{30}}$. The statistic results demonstrate that FDHD-DE can obtain satisfactory performance for high-dimensional optimization.

Table 12 summarizes the comparison results between FDHD-DE with five powerful DE variants. From the 12, we can observe that FDHD-DE secures performance improvement on 142 out of 294 cases comparing with LSHADE; secures performance improvement on 182 out of 294 cases comparing with jSO; 148 cases in comparison with LPalmDE; 152 cases in comparison with PaDE; and 141 cases in comparison with Hip-DE.

To conclude, the superiority of FDHD-DE may be contributed to that the semi-adaptive scheme can mitigate the waste of computational resources in traditional parameter adaptation strategies which employ probability distributions during the early stage of evolution. Besides, the hypervolume-based diversity indicator and dimension exchange mechanism can alleviate the problem of stagnation when dealing with complicated or high-dimensional optimization problem.

As one of the reviewers suggested, we also examined the performance of FDHD-DE with advanced optimization algorithms including LSHADE-cnEpsin Awad et al. (2017), EA4eig (Bujok and Kolenovsky, 2022), DE-EXP (Meng and Chen, 2023) and ISDE (Tian and Gao, 2019) under CEC2017 with dimensions ranging from 10 to 50. The comparison results are provided in Table S-I to S-III of the supplementary

Table 6
Comparisons between LSHADE, jSO, LPalmDE, PaDE, Hip-DE and FDHD-DE on CEC2014 with 30D.

DE Variants	LSHADE	jSO	LPalmDE	PaDE	Hip-DE	FDHD-DE
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{b_1}	1.0588E−14/6.2548E−15(<)	3.5388E−14/1.8281E−14(<)	1.4211E−14/5.6843E−15(<)	1.4211E−14/8.5265E−15(<)	1.0310E−14/7.0072E−15(<)	**3.0651E−15/5.9030E−15**
f_{b_2}	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0*
f_{b_3}	*0/0(≈)*	*0/0(≈)*	2.2292E−15/1.1144E−14(<)	*0/0(≈)*	*0/0(≈)*	*0/0*
f_{b_4}	6.5760E−14/3.2896E−14(<)	5.2385E−14/2.5019E−14(<)	3.5666E−14/2.7756E−14(<)	4.5698E−14/2.5471E−14(<)	5.2385E−14/2.7481E−14(<)	**8.9166E−15/2.0878E−14**
f_{b_5}	2.0115E+01/3.0837E−02(>)	2.0714E+01/1.9588E−01(<)	**2.0066E+01/7.4238E−02**(>)	2.0166E+01/3.7188E−02(<)	2.0068E+01/5.1551E−02(>)	2.0127E+01/4.2801E−02
f_{b_6}	9.0055E−03/6.4312E−02(>)	0/0(>)	0/0(>)	0/0(>)	0/0(>)	1.5670E−02/4.6746E−02
f_{b_7}	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0*
f_{b_8}	1.2483E−13/4.1021E−14(>)	3.5124E+00/5.0276E+00(<)	5.5729E−14/5.7398E−14(>)	1.5381E−13/1.0123E−13(>)	**3.3437E−14/5.2316E−14**(>)	1.9394E−13/9.1782E−14
f_{b_9}	**6.5226E+00/1.5823E+00**(>)	1.0458E+01/3.3085E+00(<)	1.1782E+01/3.1407E+00(<)	8.2457E+00/1.8120E+00(<)	8.1198E+00/1.1622E+00(<)	7.9690E+00/1.5348E+00
$f_{b_{10}}$	4.8986E−03/9.8431E−03(<)	4.6594E+01/9.2690E+01(<)	**1.3197E−12/1.2643E−12**(>)	8.1644E−04/4.0814E−03(>)	1.6329E−03/7.0209E−03(>)	2.8577E−03/8.3480E−03
$f_{b_{11}}$	**1.1895E+03/1.9751E+02**(>)	1.8678E+03/4.8178E+02(<)	1.4012E+03/2.8354E+02(<)	1.2152E+03/2.1827E+02(>)	1.2380E+03/2.0173E+02(>)	1.2560E+03/1.6680E+02
$f_{b_{12}}$	1.5703E−01/2.5925E−02(>)	8.4901E−01/4.2778E−01(<)	1.4456E−01/6.7727E−02(>)	1.8211E−01/4.0776E−02(<)	**1.3268E−01/3.7064E−02**(>)	1.6993E−01/2.8298E−02
$f_{b_{13}}$	1.1837E−01/1.6323E−02(>)	1.5996E−01/2.3439E−02(<)	**1.0907E−01/2.6222E−02**(>)	1.1825E−01/1.8391E−02(>)	1.1358E−01/1.7614E−02(>)	1.3726E−01/1.8994E−02
$f_{b_{14}}$	2.4013E−01/3.0229E−02(<)	2.0324E−01/3.1668E−02(<)	2.1999E−01/2.8498E−02(<)	2.1345E−01/2.3818E−02(<)	2.2812E−01/2.5689E−02(<)	**1.5866E−01/1.9020E−02**
$f_{b_{15}}$	2.1503E+00/2.0440E−01(>)	3.0787E+00/6.8228E−01(<)	**2.0946E+00/4.8430E−01**(>)	2.2001E+00/2.2639E−01(>)	2.1643E+00/2.2567E−01(>)	2.2451E+00/2.8122E−01
$f_{b_{16}}$	8.5443E+00/3.5656E−01(<)	9.4446E+00/7.1276E−01(<)	8.7497E+00/6.5272E−01(<)	8.6542E+00/5.3550E−01(<)	8.6613E+00/4.4900E−01(<)	**8.3849E+00/4.2071E−01**
$f_{b_{17}}$	1.7901E+02/1.0071E+02(<)	7.9633E+01/3.4190E+01(<)	2.2414E+02/1.0103E+02(<)	2.0448E+02/9.8168E+01(<)	2.1157E+02/9.7936E+01(<)	**5.6147E+01/2.0448E+01**
$f_{b_{18}}$	6.3908E+00/2.6048E+00(<)	2.1065E+00/1.2114E+00(<)	6.8491E+00/3.3113E+00(<)	5.8067E+00/2.3911E+00(<)	8.7687E+00/4.1251E+00(<)	**1.8380E+00/1.1198E+00**
$f_{b_{19}}$	3.6246E+00/5.9933E−01(<)	2.3229E+00/6.1148E−01(<)	3.1043E+00/5.2631E−01(<)	3.4699E+00/5.7841E−01(<)	3.5779E+00/5.3213E−01(<)	**1.6091E+00/4.7142E−01**
$f_{b_{20}}$	2.5529E+00/1.2824E+00(<)	2.6717E+00/1.1825E+00(<)	3.5070E+00/1.2706E+00(<)	3.1565E+00/1.4108E+00(<)	2.8417E+00/1.1984E+00(<)	**2.0720E+00/8.7160E−01**
$f_{b_{21}}$	9.4834E+01/8.9766E+01(<)	2.3030E+01/3.2689E+01(<)	9.2864E+01/7.7546E+01(<)	7.0374E+01/6.8848E+01(<)	6.4661E+01/6.3208E+01(<)	**7.7065E+00/6.0875E+00**
$f_{b_{22}}$	**2.5596E+01/6.2954E+00**(>)	6.7967E+01/5.5891E+01(<)	8.3125E+01/6.1693E+01(<)	6.1663E+01/4.9281E+01(>)	7.4355E+01/5.7613E+01(<)	6.5290E+01/5.4941E+01
$f_{b_{23}}$	*3.1524E+02/6.4311E−14(≈)*	*3.1524E+02/0(≈)*	*3.1524E+02/4.0186E−13(≈)*	*3.1524E+02/0(≈)*	*3.1524E+02/0(≈)*	*3.1524E+02/6.4311E−14*
$f_{b_{24}}$	2.2422E+02/1.1854E+00(<)	2.0043E+02/3.0397E+00(<)	2.2420E+02/1.9249E+00(<)	2.2348E+02/8.8410E−01(<)	2.2309E+02/9.8241E−01(<)	**2.0000E+02/1.5612E−07**
$f_{b_{25}}$	2.0261E+02/7.1958E−02(<)	2.0258E+02/3.7166E−02(<)	2.0274E+02/1.8059E−01(<)	2.0270E+02/1.6398E−01(<)	2.0269E+02/1.3055E−01(<)	**2.0255E+02/2.6493E−02**
$f_{b_{26}}$	1.0011E+02/1.7449E−02(>)	1.0016E+02/2.3491E−02(<)	**1.0010E+02/2.7974E−02**(>)	1.0012E+02/1.6644E−02(>)	1.0015E+02/1.4342E−02(<)	1.0013E+02/1.5907E−02
$f_{b_{27}}$	*3.0000E+02/1.1139E−13(≈)*	*3.0000E+02/1.3713E−13(≈)*	*3.0000E+02/9.0949E−14(≈)*	3.0196E+02/1.4003E+01(<)	*3.0000E+02/6.4311E−14(≈)*	*3.0000E+02/7.3958E−03*
$f_{b_{28}}$	8.2446E+02/2.1174E+01(<)	8.2500E+02/2.3905E+01(<)	8.5985E+02/1.5568E+01(<)	8.5756E+02/1.7105E+01(<)	8.6139E+02/2.0977E+01(<)	**8.1833E+02/1.6434E+01**
$f_{b_{29}}$	7.1670E+02/3.8165E+00(<)	6.3786E+02/2.0030E+02(<)	6.5187E+02/1.8294E+02(<)	7.1834E+02/4.9874E+00(<)	7.1769E+02/3.2011E+00(<)	**2.4861E+02/2.2095E+02**
$f_{b_{30}}$	1.2373E+03/4.5329E+02(<)	4.4331E+02/3.9894E+01(<)	6.5393E+02/2.5669E+02(<)	5.5537E+02/1.7305E+02(<)	5.7850E+02/2.1047E+02(<)	**3.7330E+02/2.0860E+01**
>/≈/<	10/5/15	1/5/24	8/4/18	8/4/18	8/5/17	−/−/−

Table 7
Comparisons between LSHADE, jSO, LPalmDE, PaDE, Hip-DE and FDHD-DE on CEC2014 with 50D.

DE Variants	LSHADE	jSO	LPalmDE	PaDE	Hip-DE	FDHD-DE
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{b_1}	5.5642E+02/1.0496E+03(<)	1.2027E+01/2.8475E+01(<)	3.1719E+03/3.3603E+03(<)	6.5768E+02/1.0684E+03(<)	1.3129E+03/1.6470E+03(<)	**7.8477E−11/2.2965E−10**
f_{b_2}	**3.3995E−14/1.1397E−14**(≈)	5.9073E−14/1.8718E−14(<)	3.9010E−14/1.3878E−14(≈)	4.2354E−14/1.5434E−14(<)	4.4026E−14/1.5373E−14(<)	3.5666E−14/1.2510E−14
f_{b_3}	5.6843E−14/0(≈)	6.5760E−14/2.0878E−14(<)	5.6843E−14/1.1369E−14(≈)	*5.5729E−14/7.9597E−15(≈)*	*5.5729E−14/7.9597E−15(≈)*	*5.5729E−14/7.9597E−15*
f_{b_4}	4.5694E+01/4.7144E+01(<)	5.4053E+01/4.9086E+01(<)	7.8758E+00/2.6585E+01(>)	1.1724E+01/3.1855E+01(>)	1.4624E+01/3.2171E+01(<)	**4.8791E−01/2.7636E−01**
f_{b_5}	2.0263E+01/2.7757E−02(>)	2.1107E+01/5.0869E−02(<)	2.0161E+01/1.4449E−01(>)	2.0280E+01/7.9639E−02(<)	**2.0154E+01/1.0827E−01**(>)	2.0258E+01/4.9711E−02
f_{b_6}	3.6242E−01/6.2490E−01(≈)	**3.7560E−03/4.6333E−03**(>)	7.7033E−01/2.8885E+00(≈)	2.8863E−01/6.1836E−01(>)	5.1826E−03/2.2898E−02(>)	3.0579E−01/1.7206E−01
f_{b_7}	2.2479E−13/5.1240E−14(<)	1.0700E−13/2.7016E−14(≈)	1.0254E−13/3.4143E−14(≈)	9.5854E−14/4.1756E−14(≈)	**6.6875E−14/5.6508E−14**(>)	1.0254E−13/3.4143E−14
f_{b_8}	7.9284E−11/9.8589E−11(>)	2.3198E−08/2.7249E−08(<)	**1.5827E−13/7.2183E−14**(>)	7.4454E−13/1.3889E−13(>)	3.9679E−13/1.8663E−13(>)	3.6270E−10/7.2753E−10
f_{b_9}	**1.1465E+01/2.6291E+00**(>)	1.5494E+01/2.9837E+00(<)	2.2467E+01/4.3700E+00(<)	1.5527E+01/2.4750E+00(<)	1.2618E+01/2.0666E+00(>)	1.4266E+01/2.3707E+00
$f_{b_{10}}$	4.4679E−02/2.1633E−02(<)	1.0802E+01/3.8287E+00(<)	**6.1685E−03/9.1693E−03**(>)	8.6409E−03/9.4747E−03(>)	1.0672E−02/1.1276E−02(>)	9.1574E−02/2.8460E−02
$f_{b_{11}}$	3.2623E+03/2.9551E+02(≈)	3.4223E+03/3.0385E+02(<)	3.5727E+03/4.5657E+02(<)	**3.1578E+03/3.0637E+02**(≈)	3.2697E+03/2.7049E+02(≈)	3.2686E+03/3.2915E+02
$f_{b_{12}}$	2.2262E−01/2.7472E−02(≈)	3.1244E−01/2.0356E−01(<)	1.9923E−01/7.0786E−02(≈)	2.2825E−01/4.6021E−02(<)	**1.9121E−01/3.5817E−02**(>)	2.1567E−01/3.2064E−02
$f_{b_{13}}$	**1.6050E−01/2.0376E−02**(>)	1.9423E−01/2.7096E−02(≈)	1.6605E−01/2.7013E−02(>)	1.8438E−01/2.2301E−02(>)	1.7280E−01/1.5395E−02(>)	2.0097E−01/2.3762E−02
$f_{b_{14}}$	3.0886E−01/2.5633E−02(<)	3.1173E−01/3.5911E−02(<)	3.0930E−01/2.5282E−02(<)	2.8821E−01/2.3309E−02(<)	3.0511E−01/2.0451E−02(<)	**1.5407E−01/1.8666E−02**
$f_{b_{15}}$	5.1025E+00/5.0576E−01(≈)	5.3561E+00/6.8352E−01(<)	**4.3985E+00/9.4224E−01**(>)	5.1300E+00/4.0739E−01(≈)	5.1936E+00/4.5943E−01(≈)	5.0299E+00/3.5132E−01
$f_{b_{16}}$	1.6903E+01/4.0371E−01(<)	1.6887E+01/6.1698E−01(<)	1.7034E+01/7.8666E−01(<)	1.6750E+01/5.0161E−01(≈)	1.7005E+01/4.8695E−01(<)	**1.6619E+01/4.5414E−01**
$f_{b_{17}}$	1.4621E+03/4.1337E+02(<)	3.1812E+02/1.5496E+02(<)	1.6354E+03/4.4952E+02(<)	1.6538E+03/3.8183E+02(<)	1.7422E+03/4.4036E+02(<)	**1.3062E+02/8.6386E+01**
$f_{b_{18}}$	9.9579E+01/1.5870E+01(<)	1.1822E+01/4.0249E+00(<)	1.0063E+02/1.3176E+01(<)	1.0100E+02/1.5585E+01(<)	1.0408E+02/1.3951E+01(<)	**3.9423E+00/1.6834E+00**
$f_{b_{19}}$	8.6246E+00/1.9597E+00(<)	9.3137E+00/7.8068E−01(<)	8.1634E+00/1.6954E+00(≈)	7.8896E+00/1.8827E+00(≈)	**7.5140E+00/1.7928E+00**(≈)	7.8037E+00/8.3075E−01
$f_{b_{20}}$	1.3132E+01/3.0502E+00(<)	6.3233E+00/1.9576E+00(<)	1.7466E+01/4.2271E+00(<)	1.4524E+01/5.9873E+00(<)	1.4155E+01/4.8204E+00(<)	**3.4513E+00/1.1149E+00**
$f_{b_{21}}$	4.5891E+02/1.1899E+02(<)	2.6303E+02/6.9748E+01(<)	6.0434E+02/1.6932E+02(<)	5.2876E+02/1.4034E+02(<)	5.1903E+02/1.5543E+02(<)	**2.2351E+02/6.0833E+01**
$f_{b_{22}}$	**1.2574E+02/7.7151E+01**(≈)	1.5245E+02/9.4119E+01(≈)	2.0456E+02/8.9644E+01(<)	1.3086E+02/5.7998E+01(≈)	1.4872E+02/5.9586E+01(≈)	1.3777E+02/6.1285E+01
$f_{b_{23}}$	*3.4400E+02/1.7667E−13(≈)*	*3.4400E+02/1.8955E−13(≈)*	*3.4400E+02/5.1724E−13(≈)*	*3.4400E+02/1.7667E−13(≈)*	*3.4400E+02/9.0949E−14(≈)*	*3.4400E+02/2.1024E−13*
$f_{b_{24}}$	2.7538E+02/6.1256E−01(<)	2.7146E+02/2.0460E+00(<)	2.7513E+02/9.8034E−01(<)	2.7508E+02/1.0715E+00(<)	2.7491E+02/7.0369E−01(<)	**2.6614E+02/2.7536E+00**
$f_{b_{25}}$	2.0534E+02/3.4951E−01(<)	2.0498E+02/1.7154E−01(<)	2.0584E+02/4.4908E−01(<)	2.0586E+02/5.5987E−01(<)	2.0568E+02/3.4811E−01(<)	**2.0488E+02/4.1654E−02**
$f_{b_{26}}$	**1.0016E+02/1.6733E−02**(>)	1.0020E+02/2.5658E−02(≈)	1.1582E+02/3.6669E+01(≈)	1.0996E+02/2.9982E+01(≈)	1.1583E+02/3.6666E+01(≈)	1.0019E+02/2.4098E−02
$f_{b_{27}}$	3.3179E+02/3.2336E+01(<)	3.0941E+02/1.8366E+01(≈)	3.1599E+02/2.5979E+01(≈)	3.2435E+02/3.1123E+01(<)	3.1243E+02/2.4083E+01(≈)	**3.0267E+02/1.4672E+00**
$f_{b_{28}}$	1.2135E+03/3.2218E+01(<)	**1.0814E+03/3.3966E+01**(>)	1.2747E+03/6.1095E+01(<)	1.2621E+03/4.5445E+01(<)	1.2504E+03/5.4025E+01(<)	1.1956E+03/4.7098E+01
$f_{b_{29}}$	8.0963E+02/4.1615E+01(<)	8.1853E+02/5.4245E+01(<)	6.1499E+02/1.2041E+02(<)	5.7430E+02/1.2837E+02(<)	6.1881E+02/1.0444E+02(<)	**4.0744E+02/2.5152E+01**
$f_{b_{30}}$	8.7388E+03/3.8142E+02(≈)	**8.3345E+03/3.4494E+02**(>)	9.6906E+03/6.4480E+02(<)	9.4812E+03/6.4061E+02(<)	9.2567E+03/5.7401E+02(<)	8.6187E+03/3.7470E+02
>/≈/<	5/10/15	3/6/21	6/9/15	5/9/16	8/8/14	−/−/−

file with a summary shown in Table 13. From the table, one can observe that FDHD-DE obtains better performance on 69 out of 90 cases comparing with LSHADE-cnEpsin, 71 cases comparing with ISDE, 47 cases comparing with EA4eig and 62 cases comparing with DE-EXP. Based on experiment results, we can conclude that FDHD-DE is able to secure highly competitive performance by comparing with advanced algorithms.

4.2. Optimization comparison on convergence speed

We also verify the performance of FDHD-DE from the perspective of convergence speed. The convergence curves of all algorithm are illustrated in Figs. 2– 3, which are derived from median values of 51 runs on each benchmark function.

From the above figure, we can clearly see that FDHD-DE is better than LSHADE on f_{a_2}, f_{a_4}, f_{a_6}, f_{a_7}, f_{a_9}, f_{13}, $f_{a_{15}}$, $f_{a_{20}}$, $f_{a_{22}}$-$f_{a_{23}}$, $f_{a_{25}}$;

Table 8
Comparisons between LSHADE, jSO, LPalmDE, PaDE, Hip-DE and FDHD-DE on CEC2017 with 10D.

DE Variants	LSHADE	jSO	LPalmDE	PaDE	Hip-DE	FDHD-DE
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{c_1}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
f_{c_2}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
f_{c_3}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
f_{c_4}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
f_{c_5}	2.6551E+00/9.0465E−01(<)	**1.7753E+00/7.7871E−01**(>)	2.6337E+00/1.1226E+00(≈)	2.3242E+00/9.0518E−01(≈)	1.9525E+00/8.1945E−01(≈)	2.2067E+00/1.2299E+00
f_{c_6}	0/0(≈)	0/0(≈)	1.7833E−14/4.1756E−14(<)	2.2292E−15/1.5919E−14(≈)	0/0(≈)	0/0
f_{c_7}	1.1997E+01/5.6580E−01(≈)	1.1970E+01/6.4821E−01(≈)	1.2762E+01/1.3630E+00(≈)	1.2272E+01/1.1740E+00(≈)	**1.1763E+01/6.2667E−01**(>)	1.2413E+01/1.4807E+00
f_{c_8}	2.5766E+00/9.3722E−01(≈)	2.2241E+00/8.5801E−01(≈)	2.5362E+00/1.1656E+00(≈)	2.4509E+00/8.8651E−01(≈)	**2.1087E+00/7.8737E−01**(≈)	2.5762E+00/1.3517E+00
f_{c_9}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
$f_{c_{10}}$	3.1179E+01/4.4309E+01(<)	4.5575E+01/5.9859E+01(<)	8.6440E+01/8.1759E+01(<)	3.1724E+01/4.6507E+01(<)	5.4977E+01/1.0564E+02(<)	**1.9015E+01/4.3783E+01**
$f_{c_{11}}$	2.4171E−01/5.6734E−01(<)	1.9509E−02/1.3932E−01(≈)	0/0(≈)	6.9781E−01/8.1319E−01(<)	2.8376E−01/6.5957E−01(<)	0/0
$f_{c_{12}}$	3.8569E+01/5.6410E+01(≈)	**2.7110E+00/1.6775E+01**(≈)	3.5466E+01/5.5162E+01(≈)	2.3887E+01/4.8343E+01(≈)	3.3247E+01/5.8590E+01(≈)	2.3609E+01/4.7824E+01
$f_{c_{13}}$	3.2334E+00/2.4100E+00(<)	3.2823E+00/2.2204E+00(<)	**1.5215E+00/2.1529E+00**(≈)	1.8480E+00/2.3046E+00(≈)	2.1097E+00/2.5678E+00(≈)	2.3944E+00/2.4443E+00
$f_{c_{14}}$	2.4035E−01/4.3312E−01(<)	1.5607E−01/3.6544E−01(≈)	**6.0134E−02/2.3631E−01**(≈)	1.8461E−01/4.1138E−01(<)	8.5840E−01/7.9645E−01(<)	6.1036E−02/2.3624E−01
$f_{c_{15}}$	1.4522E−01/2.0202E−01(≈)	3.4994E−01/1.9239E−01(<)	**1.1735E−01/1.8595E−01**(>)	1.5620E−01/2.0004E−01(≈)	1.3775E−01/2.0125E−01(≈)	1.8330E−01/2.0792E−01
$f_{c_{16}}$	3.3860E−01/1.5560E−01(>)	6.0458E−01/2.7309E−01(<)	**1.9013E−01/1.6622E−01**(>)	3.8704E−01/1.8271E−01(≈)	4.4220E−01/2.4157E−01(≈)	4.4429E−01/2.4188E−01
$f_{c_{17}}$	1.3218E−01/1.5785E−01(≈)	9.4763E−01/2.8235E+00(<)	2.1111E−01/3.4588E−01(≈)	1.5382E−01/1.4829E−01(<)	7.9989E−01/2.7181E+00(<)	**1.0381E−01/1.7050E−01**
$f_{c_{18}}$	1.6893E−01/1.9206E−01(>)	2.5521E−01/2.1589E−01(≈)	2.3845E−01/2.2158E−01(≈)	1.9680E−01/1.9414E−01(>)	**1.5345E−01/1.6948E−01**(>)	2.8993E−01/2.0866E−01
$f_{c_{19}}$	**4.6555E−03/8.1753E−03**(≈)	1.1902E−02/1.7241E−02(≈)	1.3548E−02/1.2753E−02(≈)	1.0679E−02/1.0843E−02(<)	1.4125E−02/2.5601E−02(≈)	8.4350E−03/1.9529E−02
$f_{c_{20}}$	0/0(≈)	8.2791E−01/2.8369E+00(<)	0/0(≈)	0/0(≈)	6.1210E−03/4.3713E−02(≈)	0/0
$f_{c_{21}}$	1.6548E+02/5.0371E+01(≈)	**1.4247E+02/5.1274E+01**(>)	1.5289E+02/5.2388E+01(≈)	1.4947E+02/5.1905E+01(≈)	1.6097E+02/5.0211E+01(≈)	1.6480E+02/5.0438E+01
$f_{c_{22}}$	1.0001E+02/4.8645E−02(≈)	1.0000E+02/0(>)	1.0000E+02/1.2368E−13(>)	1.0001E+02/4.0631E−02(>)	1.0000E+02/0(>)	1.0005E+02/1.2339E−01
$f_{c_{23}}$	3.0297E+02/1.6366E+00(<)	3.0109E+02/1.5983E+00(≈)	3.0135E+02/1.6340E+00(<)	3.0157E+02/1.5799E+00(<)	3.0122E+02/1.4469E+00(<)	**3.0066E+02/1.3507E+00**
$f_{c_{24}}$	2.9898E+02/8.0166E+01(<)	**2.6650E+02/1.0345E+02**(≈)	2.7346E+02/9.2398E+01(>)	2.8226E+02/9.1087E+01(≈)	2.9062E+02/8.2896E+01(≈)	2.9377E+02/7.8513E+01
$f_{c_{25}}$	4.1227E+02/2.1324E+01(≈)	**4.0331E+02/1.4773E+01**(>)	4.1752E+02/2.2713E+01(≈)	4.1931E+02/2.2889E+01(≈)	4.1663E+02/2.2574E+01(≈)	4.1751E+02/2.2727E+01
$f_{c_{26}}$	3.0000E+02/0(≈)	3.0000E+02/0(≈)	3.0000E+02/0(≈)	3.0000E+02/0(≈)	3.0000E+02/0(≈)	3.0000E+02/0
$f_{c_{27}}$	3.8948E+02/1.6825E−01(>)	**3.8942E+02/2.0549E−01**(>)	3.9305E+02/1.7129E+00(≈)	3.9306E+02/1.7282E+00(≈)	3.9358E+02/1.2076E+00(≈)	3.9351E+02/1.3230E+00
$f_{c_{28}}$	3.1168E+02/5.8444E+01(≈)	3.2568E+02/8.1963E+01(≈)	3.1113E+02/5.5625E+01(≈)	**3.0556E+02/3.9732E+01**(≈)	3.3950E+02/1.0007E+02(≈)	3.2225E+02/7.7044E+01
$f_{c_{29}}$	2.3322E+02/3.1577E+00(<)	2.3458E+02/3.1148E+00(<)	2.3188E+02/3.3384E+00(≈)	2.3331E+02/3.4771E+00(<)	2.3325E+02/3.7920E+00(≈)	**2.3178E+02/2.4971E+00**
$f_{c_{30}}$	4.0641E+02/2.2822E+01(≈)	**3.9454E+02/6.2039E−02**(≈)	3.9733E+02/1.1444E+01(≈)	3.9639E+02/9.4408E+00(≈)	3.9828E+02/1.3075E+01(≈)	3.9922E+02/1.4463E+01
>/≈/<	3/19/8	5/18/7	4/23/3	2/21/7	3/22/5	−/−/−

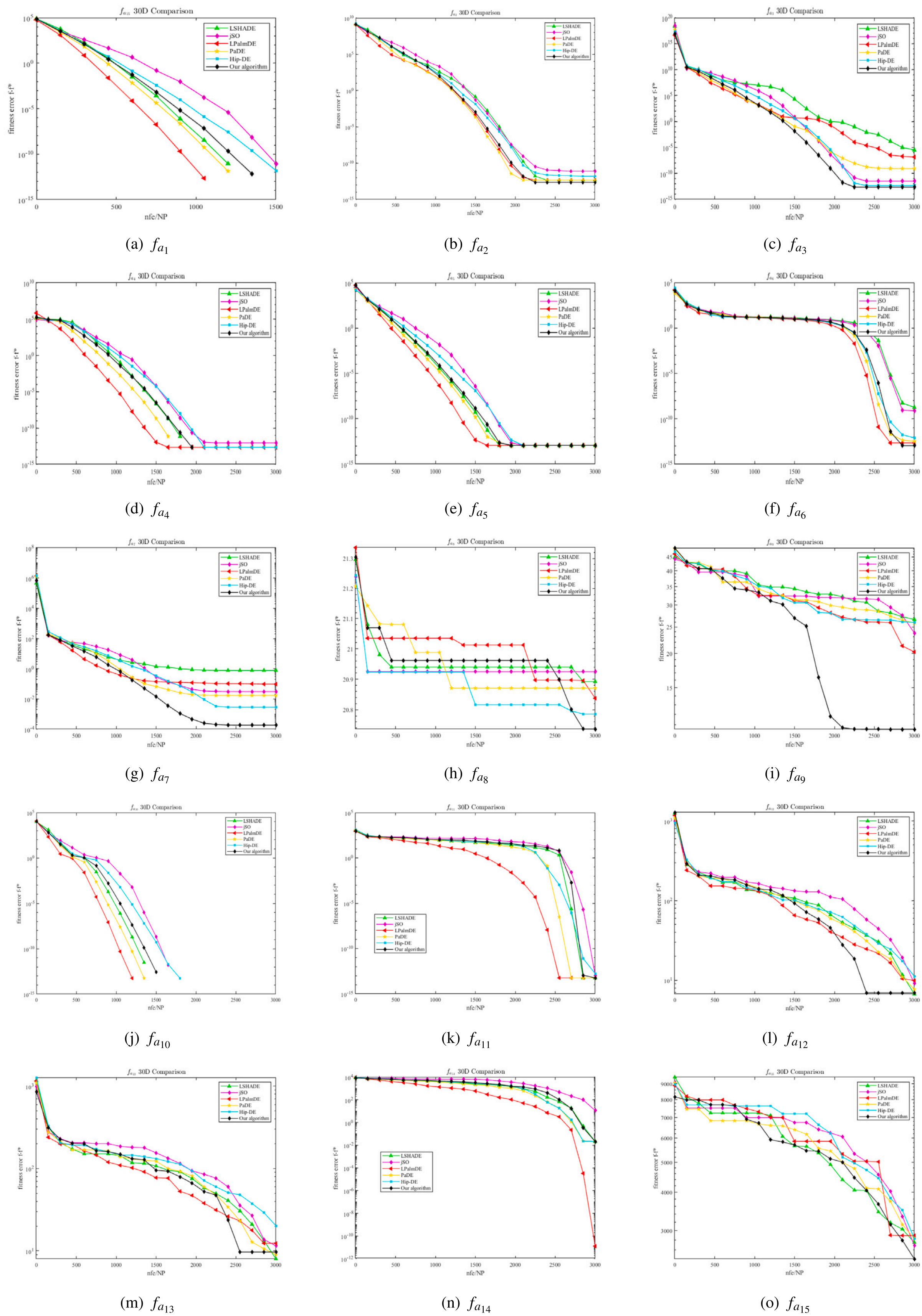
Table 9
Comparisons between LSHADE, jSO, LPalmDE, PaDE, Hip-DE and FDHD-DE on CEC2017 with 30D.

DE Variants	LSHADE	jSO	LPalmDE	PaDE	Hip-DE	our algorithm
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{c_1}	0/0(≈)	2.2292E−15/5.2195E−15(<)	8.3593E−16/3.3770E−15(<)	0/0(≈)	5.5729E−16/2.7859E−15(<)	0/0
f_{c_2}	5.5729E−15/1.1397E−14(<)	**0/0**(>)	6.6875E−15/1.3437E−14(<)	5.5729E−15/1.1397E−14(<)	8.9166E−15/1.3319E−14(<)	2.2292E−15/7.7172E−15
f_{c_3}	7.8020E−15/1.9755E−14(<)	4.3468E−14/2.6875E−14(<)	1.2260E−14/2.3612E−14(<)	2.2292E−15/1.1144E−14(<)	3.3437E−15/1.3508E−14(<)	**0/0**
f_{c_4}	5.8562E+01/2.6869E−14(≈)	5.8670E+01/7.7797E−01(<)	5.3225E+01/1.7514E+01(>)	**5.1443E+01/2.0364E+01**(>)	5.8670E+01/7.7797E−01(<)	5.8562E+01/2.7464E−14
f_{c_5}	**6.1430E+00/1.3759E+00**(>)	8.6048E+00/1.9887E+00(<)	1.0554E+01/2.7722E+00(<)	8.7278E+00/1.5651E+00(<)	7.2833E+00/1.4568E+00(>)	7.9022E+00/1.5457E+00
f_{c_6}	6.0557E−09/2.7126E−08(>)	3.2808E−08/1.5573E−07(>)	1.2037E−13/2.7016E−14(>)	**1.1369E−13/0**(>)	6.0409E−09/2.7124E−08(>)	1.5304E−06/1.0890E−05
f_{c_7}	3.7532E+01/1.4094E+00(>)	3.8443E+01/2.0003E+00(<)	4.2350E+01/3.7840E+00(<)	3.8705E+01/1.6217E+00(<)	**3.5880E+01/9.6925E−01**(>)	3.8068E+01/1.5111E+00
f_{c_8}	**7.1969E+00/1.5862E+00**(>)	8.9693E+00/2.1989E+00(<)	1.2915E+01/3.5624E+00(<)	9.4846E+00/1.7721E+00(<)	7.3763E+00/1.4044E+00(>)	8.1685E+00/1.5466E+00
f_{c_9}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
$f_{c_{10}}$	**1.4229E+03/2.2749E+02**(>)	1.5725E+03/2.3732E+02(<)	1.6161E+03/2.8206E+02(<)	1.5063E+03/2.7206E+02(<)	1.4894E+03/2.2988E+02(>)	1.4908E+03/2.4287E+02
$f_{c_{11}}$	2.5271E+01/2.7455E+01(<)	**4.2944E+00/8.2863E+00**(>)	9.1092E+00/1.4165E+01(<)	1.7092E+01/2.3230E+01(<)	1.4561E+01/2.1375E+01(<)	5.3327E+00/1.2015E+01
$f_{c_{12}}$	1.0507E+03/3.0860E+02(<)	1.9033E+02/1.3062E+02(<)	1.1286E+03/4.1334E+02(<)	1.1150E+03/3.7339E+02(<)	1.0934E+03/3.7777E+02(<)	**1.9607E+01/3.8425E+01**
$f_{c_{13}}$	1.5593E+01/5.7355E+00(<)	1.5355E+01/5.8308E+00(<)	1.6681E+01/5.8629E+00(<)	1.4909E+01/6.2006E+00(<)	1.5258E+01/6.3961E+00(<)	**1.1739E+01/6.5841E+00**
$f_{c_{14}}$	2.1999E+01/1.2871E+00(<)	2.1319E+01/4.0986E+00(<)	**1.9551E+01/6.7897E+00**(>)	2.2486E+01/3.2953E+00(<)	2.2518E+01/1.1845E+00(<)	2.0078E+01/6.2832E+00
$f_{c_{15}}$	3.3133E+00/1.7254E+00(<)	1.2194E+00/7.1566E−01(<)	3.8024E+00/2.0026E+00(<)	3.0850E+00/1.5985E+00(<)	2.8528E+00/1.4990E+00(<)	**7.4165E−01/5.0467E−01**
$f_{c_{16}}$	**5.5403E+01/5.7858E+01**(>)	7.2027E+01/8.4558E+01(>)	1.8316E+02/1.1987E+02(<)	8.8102E+01/8.3072E+01(>)	1.1182E+02/9.1892E+01(<)	9.3967E+01/8.8341E+01
$f_{c_{17}}$	3.2856E+01/5.9989E+00(<)	3.2809E+01/9.0650E+00(<)	3.4552E+01/1.1333E+01(<)	3.1150E+01/7.6777E+00(<)	3.1126E+01/6.7203E+00(<)	**2.7207E+01/7.1334E+00**
$f_{c_{18}}$	2.1789E+01/1.2468E+00(<)	2.0855E+01/4.2168E−01(<)	2.2592E+01/1.6834E+00(<)	2.1732E+01/1.1010E+00(<)	2.2606E+01/1.5667E+00(<)	**2.0570E+01/7.1895E−02**
$f_{c_{19}}$	4.7771E+00/1.2409E+00(<)	4.4298E+00/1.9901E+00(<)	5.0615E+00/1.4097E+00(<)	4.9323E+00/1.5112E+00(<)	5.1407E+00/1.7925E+00(<)	**4.3411E+00/1.5674E+00**
$f_{c_{20}}$	3.0906E+01/6.0065E+00(<)	**3.0435E+01/5.0635E+00**(>)	3.3118E+01/2.7299E+01(<)	3.7630E+01/1.7708E+01(<)	3.9194E+01/2.4154E+01(<)	3.0818E+01/5.0378E+00
$f_{c_{21}}$	**2.0738E+02/1.1102E+00**(>)	2.0936E+02/2.4318E+00(<)	2.1128E+02/2.5408E+00(<)	2.0807E+02/1.7554E+00(<)	2.0747E+02/1.6222E+00(>)	2.0780E+02/1.6903E+00
$f_{c_{22}}$	1.0000E+02/1.4352E−14(≈)	1.0000E+02/1.4352E−14(≈)	1.0000E+02/1.0047E−13(≈)	1.0000E+02/1.4352E−14(≈)	1.0000E+02/1.4352E−14(≈)	1.0000E+02/1.4352E−14
$f_{c_{23}}$	3.5026E+02/2.5652E+00(<)	3.5104E+02/2.6622E+00(<)	3.5013E+02/5.4847E+00(<)	3.4559E+02/3.0598E+00(<)	3.4463E+02/4.2466E+00(<)	**3.4423E+02/2.6603E+00**
$f_{c_{24}}$	4.2603E+02/1.8950E+00(<)	4.2705E+02/2.0143E+00(<)	4.2357E+02/2.9915E+00(<)	4.2130E+02/2.0215E+00(<)	4.2094E+02/2.2052E+00(<)	**4.2016E+02/2.5495E+00**
$f_{c_{25}}$	3.8674E+02/2.2972E−02(<)	3.8670E+02/6.3724E−03(≈)	3.8678E+02/3.2896E−02(<)	3.8677E+02/2.7835E−02(<)	3.8678E+02/2.7130E−02(<)	3.8670E+02/7.0208E−03
$f_{c_{26}}$	9.2373E+02/3.9674E+01(<)	9.3945E+02/4.8108E+01(<)	9.2447E+02/4.7704E+01(<)	8.6890E+02/5.5770E+01(<)	8.7924E+02/3.1269E+01(<)	**8.4964E+02/5.7392E+01**
$f_{c_{27}}$	5.0338E+02/5.1879E+00(<)	4.9451E+02/5.9813E+00(<)	5.0712E+02/4.4763E+00(<)	5.0792E+02/4.8855E+00(<)	5.0545E+02/5.8970E+00(<)	**4.8641E+02/9.0517E+00**
$f_{c_{28}}$	3.3465E+02/5.2231E+01(<)	3.0670E+02/2.7084E+01(<)	3.2898E+02/4.7641E+01(<)	3.3279E+02/5.1725E+01(<)	3.2842E+02/4.9133E+01(<)	**3.0447E+02/2.2344E+01**
$f_{c_{29}}$	4.3466E+02/7.7833E+00(<)	4.3694E+02/1.2304E+01(<)	**4.2772E+02/9.7203E+00**(>)	4.3185E+02/7.8449E+00(>)	4.3469E+02/7.4613E+00(<)	4.3293E+02/6.5621E+00
$f_{c_{30}}$	1.9821E+03/6.2047E+01(<)	**1.9682E+03/9.5995E+00**(>)	2.0690E+03/6.8697E+01(<)	2.0457E+03/5.1264E+01(<)	2.0514E+03/5.5328E+01(<)	1.9810E+03/1.1183E+01
>/≈/<	7/4/19	6/3/21	4/2/24	4/3/23	6/2/22	−/−/−

better than jSO on f_{a_2}, f_{a_4}, f_{a_6}-f_{a_9}, $f_{a_{12}}$-$f_{a_{18}}$, $f_{a_{20}}$-$f_{a_{27}}$; better than LPalmDE on f_{a_2}, f_{a_4}, f_{a_5}-f_{a_8}, $f_{a_{12}}$-$f_{a_{15}}$, $f_{a_{20}}$, $f_{a_{23}}$; better than PaDE on f_{a_2}-$f_{a_{15}}$, $f_{a_{18}}$-$f_{a_{25}}$; better than Hip-DE on benchmarks f_{a_2}-f_{a_8}, f_{a_9}, $f_{a_{13}}$, $f_{a_{15}}$, $f_{a_{16}}$, $f_{a_{20}}$, $f_{a_{23}}$. Furthermore, our algorithm outperforms all other algorithms on f_{a_2}, f_{a_4}, f_{a_6}, $f_{a_{13}}$, $f_{a_{16}}$, $f_{a_{20}}$, $f_{a_{23}}$ with respect to convergence speed. Based on the analysis above, we can find that our algorithm is able to locate the global optima on f_{a_1}, f_{a_4}, $f_{a_{10}}$ and $f_{a_{28}}$. Based on these above, FDHD-DE demonstrates high competitiveness with these powerful DE variants in terms of convergence speed.

4.3. Parameter sensitivity analysis

In order to mitigate the problem of stagnation, we employ a hypervolume-based diversity indicator and dimension exchange strategy to detect and update the stagnant individual, respectively. The two schemes are combined to improve population diversity during the later stage of evolution, so the combination of the two schemes is also termed as diversity enhancement mechanism (Meng and Yang, 2022). The population enhancement mechanism we suggest includes adjusting two parameters: one is the variable k which indicates that the population



(a) f_{a_1} (b) f_{a_2} (c) f_{a_3}

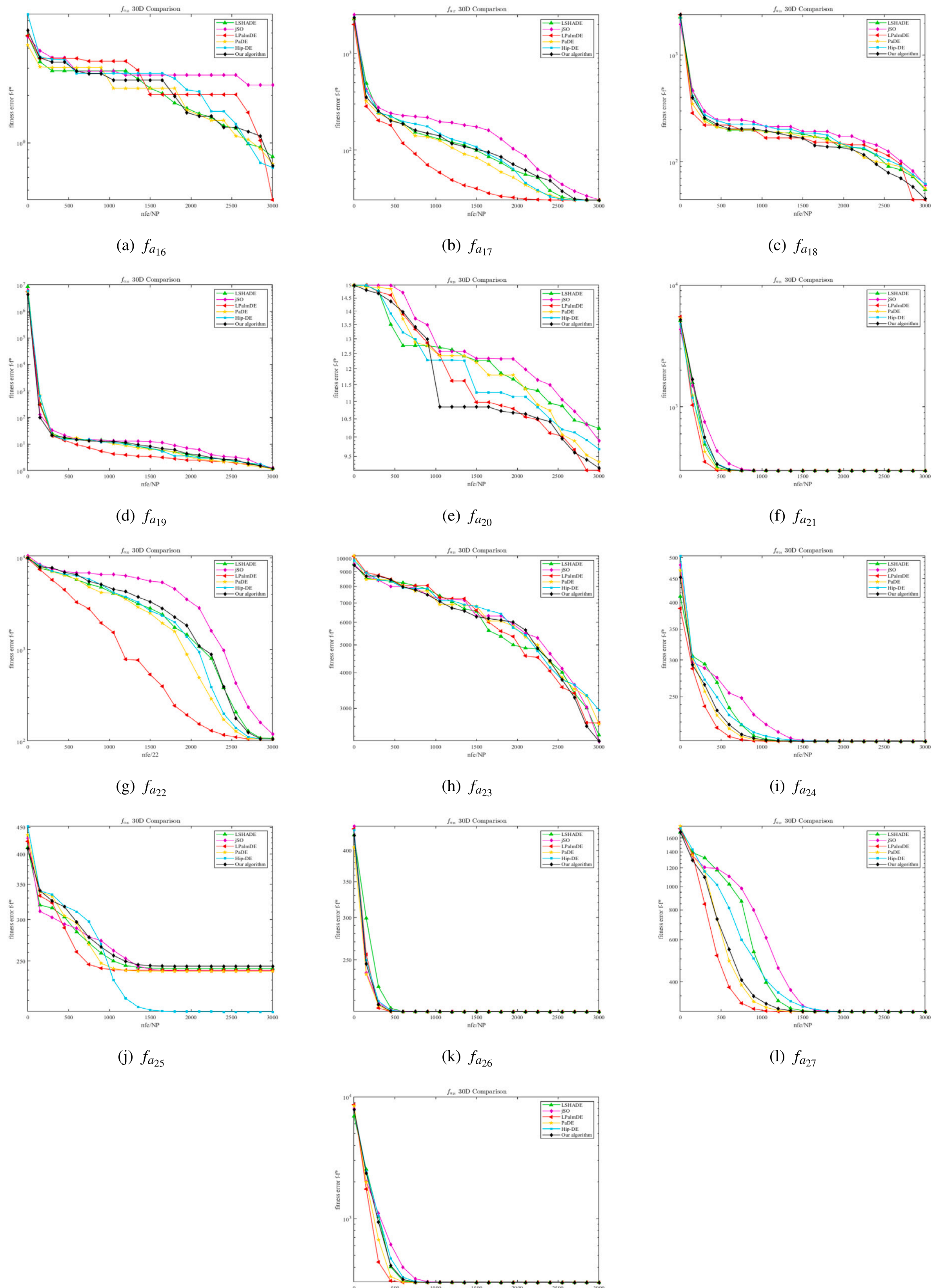
(d) f_{a_4} (e) f_{a_5} (f) f_{a_6}

(g) f_{a_7} (h) f_{a_8} (i) f_{a_9}

(j) $f_{a_{10}}$ (k) $f_{a_{11}}$ (l) $f_{a_{12}}$

(m) $f_{a_{13}}$ (n) $f_{a_{14}}$ (o) $f_{a_{15}}$

Fig. 2. Here we present the convergence comparison by employing the median value of 51 runs on f_{a_1}-$f_{a_{15}}$ of CEC2013 with 30 dimensions.



(a) $f_{a_{16}}$ (b) $f_{a_{17}}$ (c) $f_{a_{18}}$

(d) $f_{a_{19}}$ (e) $f_{a_{20}}$ (f) $f_{a_{21}}$

(g) $f_{a_{22}}$ (h) $f_{a_{23}}$ (i) $f_{a_{24}}$

(j) $f_{a_{25}}$ (k) $f_{a_{26}}$ (l) $f_{a_{27}}$

(m) $f_{a_{28}}$

Fig. 3. Here we present the convergence comparison by employing the median value of 51 runs on $f_{a_{16}}$-$f_{a_{28}}$ of CEC2013 with 30 dimensions.

Table 10
Comparisons between LSHADE, jSO, LPalmDE, PaDE, Hip-DE and FDHD-DE on CEC2017 with 50D.

DE Variants	LSHADE	jSO	LPalmDE	PaDE	Hip-DE	our algorithm
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{c_1}	*1.7833E−14/6.2548E−15(≈)*	3.0930E−14/7.8897E−15(<)	2.2570E−14/7.6139E−15(<)	2.3963E−14/7.7785E−15(<)	2.5914E−14/6.7891E−15(<)	*1.7833E−14/6.2548E−15*
f_{c_2}	5.1271E−14/9.4610E−14(<)	6.2416E−14/6.8689E−14(<)	6.8546E−14/1.7159E−13(<)	4.5140E−14/3.3714E−14(<)	4.5140E−14/2.9633E−14(<)	**2.6750E−14/2.0026E−14**
f_{c_3}	1.6384E−13/4.3510E−14(<)	2.8310E−13/8.9148E−14(<)	1.5381E−13/4.7278E−14(<)	1.6496E−13/4.5836E−14(<)	1.4824E−13/4.9657E−14(<)	**1.1703E−13/3.0951E−14**
f_{c_4}	7.4295E+01/5.0578E+01(<)	5.6418E+01/4.7040E+01(<)	6.9341E+01/5.1779E+01(<)	6.8887E+01/5.2925E+01(<)	8.0188E+01/4.7197E+01(<)	**4.0389E+01/3.4538E+01**
f_{c_5}	**1.1627E+01/2.0398E+00**(>)	1.6199E+01/2.4788E+00(<)	2.3334E+01/4.6458E+00(<)	1.7319E+01/2.1194E+00(<)	1.4647E+01/1.9140E+00(>)	1.5377E+01/2.6901E+00
f_{c_6}	1.0565E−04/5.3863E−04(>)	**4.7353E−07/6.6856E−07**(>)	1.0171E−03/2.1880E−03(<)	8.8203E−04/2.8155E−03(<)	2.5672E−05/1.8260E−04(>)	7.2496E−04/1.0870E−03
f_{c_7}	6.3588E+01/1.6952E+00(>)	6.6238E+01/3.2214E+00(<)	7.1044E+01/5.4122E+00(<)	6.4881E+01/2.3306E+00(>)	**6.1420E+01/1.3780E+00**(>)	6.4884E+01/2.0923E+00
f_{c_8}	**1.2679E+01/1.8320E+00**(>)	1.6906E+01/2.6186E+00(<)	2.4214E+01/5.7974E+00(<)	1.7476E+01/2.4663E+00(<)	1.5243E+01/2.0900E+00(>)	1.5716E+01/2.0658E+00
f_{c_9}	5.1271E−14/5.7132E−14(>)	8.6937E−14/4.8704E−14(<)	1.7555E−03/1.2536E−02(<)	5.5729E−14/5.7398E−14(>)	**1.1146E−14/3.4143E−14**(>)	6.6875E−14/5.6508E−14
$f_{c_{10}}$	3.1805E+03/2.6012E+02(<)	3.1902E+03/3.3435E+02(<)	3.2257E+03/3.8735E+02(<)	3.1929E+03/3.0149E+02(<)	3.1438E+03/3.1760E+02(<)	**3.1275E+03/3.0493E+02**
$f_{c_{11}}$	4.7248E+01/8.2365E+00(<)	2.7776E+01/3.1155E+00(<)	7.2471E+01/1.3860E+01(<)	6.7047E+01/1.4750E+01(<)	4.7692E+01/7.6651E+00(<)	**2.3163E+01/1.8042E+00**
$f_{c_{12}}$	2.1391E+03/5.2201E+02(<)	1.7865E+03/4.8589E+02(<)	2.2169E+03/4.9276E+02(<)	2.1967E+03/4.9929E+02(<)	2.3693E+03/6.1342E+02(<)	**4.0421E+02/2.1629E+02**
$f_{c_{13}}$	5.2539E+01/2.2188E+01(<)	2.8161E+01/2.4834E+01(<)	7.1488E+01/2.9691E+01(<)	5.4129E+01/2.5159E+01(<)	6.5626E+01/2.2012E+01(<)	**7.7834E+00/7.1684E+00**
$f_{c_{14}}$	2.9317E+01/3.1041E+00(<)	2.4916E+01/1.9607E+00(<)	3.0943E+01/3.0322E+00(<)	2.9987E+01/3.1172E+00(<)	3.2397E+01/4.2989E+00(<)	**2.4699E+01/1.6254E+00**
$f_{c_{15}}$	3.9549E+01/9.2048E+00(<)	2.2984E+01/2.5989E+00(<)	4.6376E+01/1.6567E+01(<)	4.0096E+01/9.4288E+00(<)	5.1579E+01/1.5661E+01(<)	**1.8296E+01/1.0259E+00**
$f_{c_{16}}$	4.1417E+02/1.2248E+02(<)	4.2207E+02/1.2573E+02(<)	4.0800E+02/1.1478E+02(<)	3.7213E+02/9.4108E+01(<)	3.7666E+02/1.0247E+02(<)	**3.5331E+02/1.4065E+02**
$f_{c_{17}}$	**2.6699E+02/7.8663E+01**(>)	2.8670E+02/1.0996E+02(<)	3.1111E+02/1.1558E+02(<)	2.8785E+02/7.2687E+01(<)	2.9637E+02/5.9646E+01(<)	2.7037E+02/5.9969E+01
$f_{c_{18}}$	3.7638E+01/1.1413E+01(<)	2.4906E+01/2.2626E+00(<)	5.2376E+01/2.9651E+01(<)	3.9903E+01/1.1006E+01(<)	5.3481E+01/1.5632E+01(<)	**2.0820E+01/3.5861E−01**
$f_{c_{19}}$	2.4046E+01/6.9657E+00(<)	1.3194E+01/2.5710E+00(<)	3.4783E+01/1.6103E+01(<)	3.0483E+01/1.2472E+01(<)	4.0751E+01/1.4438E+01(<)	**9.5265E+00/1.7884E+00**
$f_{c_{20}}$	1.7277E+02/7.5183E+01(<)	1.4218E+02/8.1732E+01(<)	1.6196E+02/1.1808E+02(<)	1.8497E+02/7.0427E+01(<)	1.6931E+02/6.9511E+01(<)	**1.2596E+02/2.8760E+01**
$f_{c_{21}}$	**2.1326E+02/2.3263E+00**(>)	2.1644E+02/3.0197E+00(>)	2.2331E+02/3.7662E+00(<)	2.1770E+02/2.3717E+00(<)	2.1625E+02/2.2651E+00(>)	2.1681E+02/2.5746E+00
$f_{c_{22}}$	2.3684E+03/1.6602E+03(<)	1.4131E+03/1.7393E+03(<)	7.7586E+02/1.4763E+03(<)	5.3051E+02/1.1744E+03(<)	**1.0261E+02/1.5993E+01**(>)	1.1369E+02/6.4229E+01
$f_{c_{23}}$	4.3009E+02/4.0681E+00(<)	4.3220E+02/6.6582E+00(<)	4.3736E+02/7.8816E+00(<)	4.2979E+02/6.4815E+00(<)	4.2759E+02/6.9648E+00(<)	**4.1941E+02/6.5959E+00**
$f_{c_{24}}$	5.0676E+02/3.0050E+00(<)	5.0801E+02/3.9418E+00(<)	5.0459E+02/4.9137E+00(<)	5.0409E+02/5.4055E+00(<)	5.0766E+02/5.8008E+00(<)	**4.9948E+02/5.5718E+00**
$f_{c_{25}}$	4.8301E+02/1.1900E+01(<)	4.8173E+02/4.1139E+00(<)	4.9531E+02/2.3657E+01(<)	4.9807E+02/2.6669E+01(<)	4.8183E+02/4.0653E+00(<)	**4.8019E+02/3.9841E−01**
$f_{c_{26}}$	1.1373E+03/4.9051E+01(<)	1.1565E+03/6.2196E+01(<)	1.1605E+03/7.1254E+01(<)	1.1269E+03/6.4635E+01(<)	1.1307E+03/7.1604E+01(<)	**1.0245E+03/6.9479E+01**
$f_{c_{27}}$	5.3009E+02/1.6504E+01(<)	5.1181E+02/9.6220E+00(<)	5.4156E+02/1.3098E+01(<)	5.3813E+02/1.1589E+01(<)	5.3516E+02/7.5460E+00(<)	**5.0123E+02/5.6118E+00**
$f_{c_{28}}$	4.7401E+02/2.2677E+01(<)	4.5981E+02/6.8398E+00(<)	4.9429E+02/2.2015E+01(<)	4.9812E+02/1.9586E+01(<)	4.8758E+02/2.4279E+01(<)	**4.5885E+02/2.3561E−13**
$f_{c_{29}}$	**3.4906E+02/1.0963E+01**(>)	3.6272E+02/1.4434E+01(>)	3.5149E+02/1.7455E+01(>)	3.5220E+02/1.0171E+01(>)	3.7017E+02/1.2459E+01(<)	3.6299E+02/1.0906E+01
$f_{c_{30}}$	6.6328E+05/9.1980E+04(<)	6.1103E+05/4.1635E+04(<)	6.2226E+05/4.2168E+04(<)	6.2748E+05/3.9594E+04(<)	6.1904E+05/3.5079E+04(<)	**5.8875E+05/1.8175E+04**
>/≈/<	8/1/21	3/0/27	1/0/29	3/0/27	7/0/23	−/−/−

Table 11
Comparisons between LSHADE, jSO, LPalmDE, PaDE, Hip-DE and FDHD-DE on CEC2017 with 100D.

DE Variants	LSHADE	jSO	LPalmDE	PaDE	Hip-DE	FDHD-DE
NO.	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std	Mean/Std
f_{c_1}	7.3896E−13/8.9686E−13(>)	1.7047E−08/7.6147E−12(<)	2.8255E−13/8.0178E−13(>)	**2.0536E−13/2.6495E−13**(>)	2.3256E−12/5.4658E−12(>)	2.2114E−09/3.6006E−09
f_{c_2}	3.0065E+02/1.6475E+03(>)	9.0966E+01/3.7725E+01(>)	5.1674E+00/3.1154E+01(>)	**1.8276E−06/1.0739E−05**(>)	6.2828E+02/2.4471E+03(<)	3.2287E+02/1.2461E+03
f_{c_3}	8.5931E−07/1.0218E−06(<)	3.1320E−07/3.1046E−07(<)	1.8297E−07/2.2898E−07(<)	**7.6206E−08/1.4680E−07**(>)	1.9368E−07/4.1047E−07(<)	1.3714E−07/2.7126E−07
f_{c_4}	1.9668E+02/1.1574E+01(>)	1.9378E+02/2.1805E+01(>)	**1.1581E+02/6.8223E+01**(>)	1.2463E+02/6.9534E+01(>)	1.9590E+02/3.3743E+00(>)	2.0242E+02/3.2077E+01
f_{c_5}	3.7352E+01/5.6033E+00(>)	**2.9772E+01/6.0453E+00**(>)	5.5712E+01/8.9897E+00(<)	4.5139E+01/5.1004E+00(<)	3.9906E+01/4.3826E+00(>)	4.0306E+01/4.5031E+00
f_{c_6}	6.5851E−03/3.8162E−03(>)	**2.0632E−04/4.8850E−04**(>)	1.5295E−02/1.1984E−02(>)	1.5603E−02/1.6361E−02(>)	1.0858E−03/1.1299E−03(>)	1.7939E−01/6.3388E−02
f_{c_7}	1.4008E+02/3.7571E+00(<)	1.3125E+02/8.0417E+00(<)	1.5164E+02/7.7896E+00(<)	1.4014E+02/3.9842E+00(<)	1.3220E+02/3.7489E+00(<)	**1.3075E+02/3.7542E+00**
f_{c_8}	3.9456E+01/4.5198E+00(>)	**3.0340E+01/4.4281E+00**(>)	5.7677E+01/7.9064E+00(<)	4.5875E+01/4.8086E+00(<)	4.3640E+01/4.3073E+00(<)	4.3212E+01/4.5065E+00
f_{c_9}	5.3220E−01/5.6448E−01(>)	**1.2037E−13/2.7016E−14**(>)	1.3239E+00/8.5952E−01(>)	9.6747E−01/6.2137E−01(>)	3.8620E−02/7.6503E−02(>)	2.5559E+00/1.1356E+00
$f_{c_{10}}$	1.0346E+04/4.9641E+02(<)	1.1346E+04/7.7446E+02(<)	9.6959E+03/6.4184E+02(<)	9.7105E+03/5.0214E+02(<)	1.0383E+04/6.5123E+02(<)	**9.1354E+03/5.3384E+02**
$f_{c_{11}}$	4.6550E+02/1.1355E+02(<)	1.1919E+02/3.2361E+01(<)	6.3934E+02/1.1335E+02(<)	5.7635E+02/8.6944E+01(<)	4.5567E+02/7.8064E+01(<)	**4.5808E+01/2.8063E+01**
$f_{c_{12}}$	2.1519E+04/6.7812E+03(<)	1.8260E+04/8.8971E+03(<)	1.8717E+04/6.9982E+03(<)	2.2884E+04/8.8568E+03(<)	2.3315E+04/8.6249E+03(<)	**5.8871E+03/1.8654E+03**
$f_{c_{13}}$	4.9293E+02/3.9935E+02(<)	2.1862E+02/6.2157E+01(<)	1.2089E+03/8.1210E+02(<)	7.2575E+02/3.9125E+02(<)	1.8580E+03/8.0657E+02(<)	**1.6740E+02/6.4207E+01**
$f_{c_{14}}$	2.5249E+02/3.2490E+01(<)	7.0393E+01/1.2821E+01(<)	2.6310E+02/3.8153E+01(<)	2.7586E+02/4.0309E+01(<)	2.5379E+02/2.9448E+01(<)	**3.1299E+01/3.2314E+00**
$f_{c_{15}}$	2.5019E+02/5.5026E+01(<)	1.9962E+02/4.2506E+01(<)	2.4976E+02/4.9758E+01(<)	2.5913E+02/4.3604E+01(<)	2.3688E+02/4.6663E+01(<)	**6.1358E+01/3.1734E+01**
$f_{c_{16}}$	1.7080E+03/2.5313E+02(<)	1.7876E+03/3.7649E+02(<)	1.6725E+03/2.9246E+02(<)	**1.4224E+03/2.6291E+02**(>)	1.8912E+03/3.0987E+02(<)	1.4711E+03/2.7780E+02
$f_{c_{17}}$	1.1030E+03/1.7351E+02(<)	1.2518E+03/2.8538E+02(<)	1.2486E+03/2.6139E+02(<)	1.1762E+03/1.7418E+02(<)	1.2923E+03/1.7638E+02(<)	**1.0630E+03/2.0533E+02**
$f_{c_{18}}$	2.2707E+02/5.1504E+01(<)	1.9851E+02/4.0267E+01(<)	2.2742E+02/5.2416E+01(<)	2.1779E+02/4.7641E+01(<)	2.1770E+02/4.6402E+01(<)	**3.4789E+01/5.8100E+00**
$f_{c_{19}}$	1.7210E+02/2.5433E+01(<)	1.4435E+02/1.7947E+01(<)	1.7882E+02/2.6508E+01(<)	1.8357E+02/3.0051E+01(<)	1.7315E+02/2.2927E+01(<)	**4.1051E+01/4.6744E+00**
$f_{c_{20}}$	1.5994E+03/1.8845E+02(<)	**1.3490E+03/2.5210E+02**(>)	1.4440E+03/2.3029E+02(<)	1.5452E+03/2.1643E+02(<)	1.6243E+03/1.7776E+02(<)	1.3519E+03/2.0335E+02
$f_{c_{21}}$	**2.5871E+02/6.5915E+00**(>)	2.5955E+02/4.6235E+00(>)	2.8439E+02/9.7938E+00(<)	2.6978E+02/5.9218E+00(<)	2.6713E+02/5.9443E+00(<)	2.6119E+02/5.4083E+00
$f_{c_{22}}$	1.1167E+04/1.5373E+03(<)	1.1442E+04/8.0037E+02(<)	1.0523E+04/1.6274E+03(<)	1.1011E+04/5.3478E+02(<)	1.1635E+04/5.5910E+02(<)	**1.0177E+04/5.0395E+02**
$f_{c_{23}}$	**5.6892E+02/8.6729E+00**(>)	5.9110E+02/1.0543E+01(<)	5.8529E+02/1.2811E+01(<)	5.9339E+02/1.9521E+01(<)	6.0766E+02/1.2630E+01(<)	5.7635E+02/1.1058E+01
$f_{c_{24}}$	9.1118E+02/7.4354E+00(<)	9.0300E+02/1.1805E+01(<)	9.3472E+02/1.6291E+01(<)	9.2374E+02/1.2208E+01(<)	9.2928E+02/1.3172E+01(<)	**8.6821E+02/1.5401E+01**
$f_{c_{25}}$	7.4622E+02/2.8359E+01(<)	7.1784E+02/4.2077E+01(<)	7.2855E+02/3.7814E+01(<)	7.3038E+02/4.5545E+01(<)	7.3344E+02/4.1661E+01(<)	**6.6712E+02/3.9445E+01**
$f_{c_{26}}$	3.3042E+03/8.9139E+01(<)	3.1735E+03/1.1047E+02(<)	3.4736E+03/1.4629E+02(<)	3.3906E+03/1.1060E+02(<)	3.3343E+03/9.3665E+01(<)	**2.9161E+03/9.8939E+01**
$f_{c_{27}}$	6.2777E+02/1.9876E+01(<)	6.0566E+02/1.7111E+01(<)	6.6124E+02/2.2129E+01(<)	6.5690E+02/2.1078E+01(<)	6.3851E+02/1.3582E+01(<)	**5.7123E+02/1.4944E+01**
$f_{c_{28}}$	5.2971E+02/3.2387E+01(<)	5.3478E+02/3.0677E+01(<)	5.3301E+02/3.3932E+01(<)	5.2179E+02/2.9053E+01(<)	5.2982E+02/2.8022E+01(<)	**5.1571E+02/3.8763E+01**
$f_{c_{29}}$	1.2654E+03/1.7396E+02(<)	1.4931E+03/2.3702E+02(<)	1.3722E+03/2.0603E+02(<)	1.1774E+03/1.9076E+02(<)	1.2337E+03/1.8583E+02(<)	**1.0844E+03/1.4248E+02**
$f_{c_{30}}$	2.3863E+03/1.5950E+02(<)	2.4451E+03/1.4975E+02(<)	2.5425E+03/1.6137E+02(<)	2.5845E+03/1.5411E+02(<)	2.5758E+03/1.6872E+02(<)	**2.3002E+03/1.1575E+02**
>/≈/<	9/0/21	8/0/22	5/0/25	7/0/23	5/0/25	−/−/−

Table 12
Summary of the comparison results from Tables 2 to 11.

Test suit:	CEC2013			CEC2014			CEC2017				All
>/≈/<	D = 10	D = 30	D = 50	D = 10	D = 30	D = 50	D = 10	D = 30	D = 50	D = 100	Σ
LSHADE	1/17/10	8/5/15	6/11/11	2/21/7	10/5/15	5/10/15	3/19/8	7/4/19	8/1/21	9/0/21	59/93/142
jSO	1/16/11	5/4/19	6/8/14	4/10/16	1/5/24	3/6/21	5/18/7	6/3/21	3/0/27	8/0/22	42/70/182
LPalmDE	2/18/8	12/4/12	8/11/9	8/17/5	8/4/18	6/9/15	4/23/3	4/2/24	1/0/29	5/0/25	58/88/148
PaDE	1/20/7	8/5/15	7/10/11	5/20/5	8/4/18	5/9/16	2/21/7	4/3/23	3/0/27	7/0/23	50/92/152
Hip-DE	5/17/6	11/4/13	10/9/9	4/19/7	8/5/17	8/8/14	3/22/5	6/2/22	7/0/23	5/0/25	67/86/141

Table 13
Summary of comparison results between LSHADE-cnEpsin, ISDE, EA4eig, DE-EXP and FDHD-DE on CEC2017.

Summary of the comparison results				
>/≈/<	D = 10	D = 30	D = 50	*sum*
LSHADE-cnEpsin	4/8/18	4/2/24	2/1/27	10/11/69
ISDE	2/11/17	1/4/25	1/0/29	4/15/71
EA4eig	4/10/16	2/14/14	4/9/17	10/33/47
DE-EXP	4/16/10	3/1/26	4/0/26	11/17/62

Table 14
The analysis of k in reduce population stagnation on 30D optimization under benchmarks f_{c_1}-$f_{c_{30}}$ of our test suite.

n	$k \in (0.6, 5)$	$k \in (0.6, 7)$	$k \in (0.6, 8)$	$k \in (0.6, 9)$	$k \in (0.6, 6)$
f_{c_1}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
f_{c_2}	1.1146E−15/5.5718E−15(<)	0/0(≈)	5.5729E−16/3.9798E−15(<)	2.2292E−15/1.2510E−14(<)	0/0
f_{c_3}	1.1146E−15/7.9597E−15(<)	1.1146E−15/7.9597E−15(<)	2.2292E−15/1.1144E−14(<)	2.2292E−15/1.1144E−14(<)	**0/0**
f_{c_4}	5.7413E+01/8.2003E+00(<)	5.8779E+01/1.0892E+00(<)	5.7413E+01/8.2003E+00(<)	5.8670E+01/7.7797E−01(<)	**5.5335E+01/1.4014E+01**
f_{c_5}	6.4149E+00/1.6016E+00(<)	6.8681E+00/1.4898E+00(<)	7.2453E+00/1.4413E+00(<)	9.0709E+00/1.7744E+00(<)	**5.5823E+00/1.6470E+00**
f_{c_6}	**1.2098E−08/3.7378E−08**(>)	4.5800E−08/1.6091E−07(>)	1.2214E−08/3.7550E−08(>)	2.0206E−08/4.7654E−08(>)	2.0916E−06/5.2833E−06
f_{c_7}	3.7855E+01/1.4582E+00(<)	3.7182E+01/1.2580E+00(<)	3.9623E+01/1.5668E+00(<)	3.9861E+01/1.7487E+00(<)	**3.6685E+01/9.2988E−01**
f_{c_8}	6.4181E+00/1.6954E+00(<)	7.2150E+00/1.6765E+00(<)	7.1102E+00/1.3499E+00(<)	9.0224E+00/1.6698E+00(<)	**5.5329E+00/1.5354E+00**
f_{c_9}	0/0(≈)	0/0(≈)	0/0(≈)	0/0(≈)	0/0
$f_{c_{10}}$	1.4663E+03/2.3575E+02(<)	1.4050E+03/2.0300E+02(<)	**1.3313E+03/2.0936E+02**(>)	1.4474E+03/1.7209E+02(<)	1.3746E+03/2.0947E+02
$f_{c_{11}}$	4.7410E+00/8.4775E+00(>)	5.7844E+00/1.1611E+01(>)	3.5262E+00/1.9671E+00(>)	**3.3122E+00/2.0181E+00**(>)	5.8486E+00/1.1403E+01
$f_{c_{12}}$	5.3554E+01/5.6676E+01(<)	4.6136E+01/5.6066E+01(<)	6.2798E+01/6.0446E+01(<)	4.3880E+01/5.5087E+01(<)	**2.7273E+01/4.6722E+01**
$f_{c_{13}}$	1.3383E+01/6.0729E+00(<)	1.3081E+01/6.2372E+00(<)	1.3972E+01/6.1861E+00(<)	1.4517E+01/5.8243E+00(<)	**1.2986E+01/6.2801E+00**
$f_{c_{14}}$	**1.9446E+01/7.0275E+00**(>)	2.0700E+01/4.9461E+00(>)	2.1009E+01/4.1493E+00(>)	2.0825E+01/4.7579E+00(>)	2.1434E+01/2.4350E+00
$f_{c_{15}}$	1.0373E+00/7.7470E−01(>)	9.1804E−01/6.3986E−01(>)	1.0376E+00/6.6093E−01(>)	**9.0677E−01/6.0061E−01**(>)	1.1361E+00/9.8402E−01
$f_{c_{16}}$	**3.4111E+01/3.9910E+01**(>)	4.8478E+01/6.9608E+01(<)	5.4736E+01/5.6377E+01(<)	4.5313E+01/5.1060E+01(<)	3.5257E+01/5.4709E+01
$f_{c_{17}}$	2.5665E+01/6.3969E+00(<)	2.3125E+01/6.4739E+00(<)	2.4924E+01/6.6233E+00(<)	2.6215E+01/6.6498E+00(<)	**2.2671E+01/7.0836E+00**
$f_{c_{18}}$	2.0612E+01/1.3777E−01(<)	**1.9444E+01/4.7053E+00**(>)	2.0566E+01/6.1520E−02(>)	2.0564E+01/5.9004E−02(>)	2.0608E+01/1.8331E−01
$f_{c_{19}}$	4.7518E+00/1.9636E+00(<)	4.3939E+00/1.7470E+00(>)	**4.2150E+00/1.3190E+00**(>)	4.3964E+00/1.3592E+00(>)	4.5242E+00/1.7821E+00
$f_{c_{20}}$	**2.3831E+01/7.9243E+00**(>)	2.6221E+01/9.4105E+00(<)	2.9042E+01/1.8680E+01(<)	2.8752E+01/7.9558E+00(<)	2.4016E+01/6.8972E+00
$f_{c_{21}}$	2.0712E+02/1.7987E+00(<)	2.0710E+02/1.7913E+00(<)	2.0687E+02/1.7979E+00(<)	2.0966E+02/1.6496E+00(<)	**2.0626E+02/1.6904E+00**
$f_{c_{22}}$	1.0000E+02/1.4352E−14(≈)	1.0000E+02/1.4352E−14(≈)	1.0000E+02/1.4352E−14(≈)	1.0000E+02/1.4352E−14(≈)	1.0000E+02/6.3901E−14
$f_{c_{23}}$	3.4467E+02/2.9357E+00(<)	3.4531E+02/3.1163E+00(<)	3.4506E+02/3.1790E+00(<)	3.4610E+02/2.9342E+00(<)	**3.4451E+02/2.4913E+00**
$f_{c_{24}}$	4.2072E+02/2.6457E+00(<)	**4.1998E+02/2.1682E+00**(>)	4.2065E+02/2.2184E+00(<)	4.2136E+02/2.1429E+00(<)	4.2060E+02/3.1966E+00
$f_{c_{25}}$	3.8670E+02/5.8195E−03(≈)	3.8670E+02/5.1284E−03(≈)	3.8670E+02/8.2477E−03(≈)	3.8670E+02/5.9880E−03(≈)	3.8670E+02/6.6889E−03
$f_{c_{26}}$	8.2825E+02/4.7099E+01(>)	8.3566E+02/4.3116E+01(>)	8.3420E+02/4.0849E+01(>)	**8.2536E+02/3.9386E+01**(>)	8.4099E+02/5.3211E+01
$f_{c_{27}}$	4.9018E+02/9.1109E+00(<)	4.8944E+02/8.8317E+00(<)	4.8848E+02/9.4776E+00(<)	4.9074E+02/9.5550E+00(<)	**4.8625E+02/8.4478E+00**
$f_{c_{28}}$	3.0447E+02/2.2344E+01(>)	3.0649E+02/2.6267E+01(>)	3.0670E+02/2.7084E+01(>)	3.0447E+02/2.2344E+01(>)	3.0831E+02/2.8800E+01
$f_{c_{29}}$	4.2899E+02/6.1220E+00(<)	**4.2686E+02/5.9844E+00**(>)	4.2895E+02/6.5106E+00(<)	4.3148E+02/6.5261E+00(<)	4.2838E+02/7.5324E+00
$f_{c_{30}}$	1.9921E+03/2.5700E+01(<)	1.9817E+03/1.8775E+01(<)	1.9849E+03/2.0698E+01(<)	1.9861E+03/1.2638E+01(<)	**1.9749E+03/1.6276E+01**
>/≈/<	8/4/18	10/5/15	9/4/17	8/4/18	−/−/−

does not evolve during continuous iteration, and the other ξ is the threshold of hypervolume-based diversity indicator.

If the parameter k is too small, the population will perform more times of intervention strategy, which may result in population deviations, and if it is too big, it will fail to detect stagnation individuals. To investigate the setting of k, we test five ranges including $k \in (0.6, 5)$, $k \in (0.6, 6)$, $k \in (0.6, 7)$, $k \in (0.6, 8)$ and $k \in (0.6, 9)$. The experiment results are presented in Table 14. The proper setting of ξ is also analyzed, as shown in Table 15.

From Table 14, it can be concluded that setting $k \in (0.6, 6)$ in the FDHD-DE algorithm is the best among the five settings. From the Table 15, it can be concluded that setting $\xi = 0.5$ in the FDHD-DE algorithm is the best among the 5 setting. Therefore, through the above conclusions, we recommend the settings of $k \in (0.6, 6)$ and $\xi = 0.01$.

4.4. Time complexity analysis

In this section, we compare the time complexity of our proposed method with that of comparing DE variants. The time complexity comparison is conducted using the CEC2013 competition's recommended method, and the results are provided in Table 16. The time complexity is determined by $(\hat{T}_2 - T_1)/T_0$, where the meanings of T_0, T_1 and $\hat{T}_2$ are defined in Liang et al. (2013b). The comparison results are presented in Table 16. From the table, one can see that FDHD-DE has less time complexity than jSO, PaDE, LPalmDE and Hip-DE indicating that the significant performance improvement obtained by FDHD-DE does not come at the cost of time complexity.

As a reviewer suggested, we also provide a further analysis of the computational complexity in terms of $\mathbf{O}(\cdot)$. The computational complexity of FDHD-DE in each run is $\mathbf{O}(PS \cdot D \cdot (U - L))$, in which PS is the population size, D is the dimension, and U and L are the upper and lower boundaries. The total computational complexity of the algorithm under the fixed cost criterion is $\mathbf{O}(nfes_{max} \cdot D \cdot (U - L))$. Obviously, the algorithms in the comparison are of no difference in terms of **O** notation. Moreover, the fitness evaluation usually consumes much more time in comparison with the time costed by the algorithm itself. That is also why CEC2013 competitions give their recommendations rather than using **O** notation for complexity analysis.

4.5. Real-world application

In order to verify the feasibility of FDHD-DE, it is applied to the pressure vessel design. The objective of this design problem is to minimize the total cost while satisfying constraints. There are four design variables: shell thickness (z_1), head thickness (z_2), inner radius (x_3), and length of the vessel without including the head (x_4), where shell thickness (z_1) and head thickness (z_2) are integer multiples of 0.625, and inner radius and length of the vessel are continuous variables. The specific mathematical model is shown as follows:

Minimize:

$$f(x) = 1.7781 z_2 x_3^2 + 0.6224 z_1 x_3 x_4 + 3.1661 z_1^2 x_4 + 19.84 z_1^2 x_3 \tag{34}$$

Constraints:

$$\begin{cases} g_1(x) = 0.00954 x_3 - z_2 \le 0 \\ g_2(x) = 0.00193 x_3 - z_1 \le 0 \\ g_3(x) = x_4 - 240 \le 0 \\ g_4(x) = -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^2 + 1296000 \le 0 \end{cases} \tag{35}$$

Boundary:

$$10 \le x_4, x_3 \le 200, 1 \le x_2, x_1 \le 99 \tag{36}$$

Table 15
The analysis of stagnation indicator $nVOL$ in population diversity enhancement on 30D optimization under benchmarks f_{c_1}–$f_{c_{30}}$ of our test suite.

ξ	$\xi = 0.005$	$\xi = 0.02$	$\xi = 0.03$	$\xi = 0.04$	$\xi = 0.01$
f_{c_1}	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0*
f_{c_2}	1.6719E−15/6.7540E−15(≈)	1.6719E−15/6.7540E−15(≈)	**5.5729E−16/3.9798E−15**(>)	3.3437E−15/9.2483E−15(<)	1.6719E−15/6.7540E−15
f_{c_3}	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0*
f_{c_4}	5.8670E+01/7.7797E−01(<)	5.7413E+01/8.2003E+00(<)	**5.3968E+01/1.5901E+01**(>)	5.7522E+01/8.2526E+00(<)	5.6701E+01/1.1667E+01
f_{c_5}	**5.6167E+00/1.5738E+00**(>)	5.7269E+00/1.2663E+00(>)	6.3276E+00/1.7174E+00(<)	6.4262E+00/1.5511E+00(<)	6.0897E+00/1.5586E+00
f_{c_6}	2.4699E−06/6.4608E−06(<)	5.5094E−06/1.7026E−05(<)	6.6562E−06/1.5867E−05(<)	6.2930E−06/1.9366E−05(<)	**1.9956E−06/5.7025E−06**
f_{c_7}	**3.6648E+01/9.2451E−01**(>)	3.6717E+01/1.0001E+00(<)	3.6691E+01/1.3045E+00(<)	3.6711E+01/1.1322E+00(<)	3.6666E+01/9.5127E−01
f_{c_8}	6.1301E+00/1.6696E+00(>)	**6.0210E+00/1.6895E+00**(>)	6.5399E+00/1.6350E+00(<)	6.4807E+00/1.4989E+00(<)	6.1659E+00/1.9405E+00
f_{c_9}	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0(≈)*	*0/0*
$f_{c_{10}}$	1.3781E+03/2.4262E+02(<)	1.4067E+03/1.8986E+02(<)	1.4060E+03/2.1738E+02(<)	1.4105E+03/2.4444E+02(<)	**1.3055E+03/2.4648E+02**
$f_{c_{11}}$	7.9097E+00/1.5799E+01(<)	**5.0175E+00/1.1653E+01**(>)	5.9465E+00/1.1567E+01(<)	5.4941E+00/1.1613E+01(<)	5.3680E+00/1.2054E+01
$f_{c_{12}}$	**1.9809E+01/3.8729E+01**(>)	2.7476E+01/4.6108E+01(>)	2.6577E+01/4.5108E+01(>)	3.3036E+01/5.0261E+01(>)	3.4837E+01/5.0669E+01
$f_{c_{13}}$	1.3431E+01/5.4456E+00(<)	1.3623E+01/6.0233E+00(<)	1.2916E+01/5.8807E+00(<)	1.4767E+01/5.0464E+00(<)	**1.2823E+01/5.7271E+00**
$f_{c_{14}}$	2.0271E+01/4.5826E+00(>)	2.0233E+01/4.7440E+00(>)	2.1024E+01/4.0878E+00(<)	**2.0034E+01/5.1106E+00**(>)	2.0466E+01/5.0788E+00
$f_{c_{15}}$	9.6770E−01/7.5048E−01(<)	9.4870E−01/7.0261E−01(<)	9.4798E−01/7.2323E−01(<)	**8.2101E−01/5.9062E−01**(>)	9.2883E−01/6.0812E−01
$f_{c_{16}}$	2.5191E+01/4.3219E+01(<)	4.0472E+01/5.6848E+01(<)	3.3943E+01/5.2746E+01(<)	2.9037E+01/3.8970E+01(<)	**2.1991E+01/1.9718E+01**
$f_{c_{17}}$	2.3693E+01/6.4696E+00(<)	2.3406E+01/6.6431E+00(<)	2.4574E+01/7.3442E+00(<)	2.2763E+01/6.6244E+00(<)	**2.1718E+01/7.7198E+00**
$f_{c_{18}}$	2.0213E+01/2.8183E+00(>)	**1.9023E+01/5.4512E+00**(>)	1.9805E+01/3.9173E+00(>)	2.0224E+01/2.8058E+00(>)	2.0613E+01/2.1854E−01
$f_{c_{19}}$	4.4771E+00/1.8785E+00(<)	4.6069E+00/1.8619E+00(<)	**4.0621E+00/1.3078E+00**(>)	4.4044E+00/1.4967E+00(<)	4.3448E+00/1.6637E+00
$f_{c_{20}}$	2.5206E+01/5.7625E+00(<)	2.2813E+01/6.4889E+00(<)	2.5075E+01/6.9208E+00(<)	**2.2572E+01/6.7394E+00**(>)	2.2690E+01/6.5789E+00
$f_{c_{21}}$	**2.0647E+02/1.6504E+00**(>)	2.0683E+02/1.5023E+00(>)	2.0687E+02/1.8133E+00(>)	2.0693E+02/1.7409E+00(>)	2.0695E+02/1.5747E+00
$f_{c_{22}}$	*1.0000E+02/6.3901E−14(≈)*	*1.0000E+02/1.4352E−14(≈)*	*1.0000E+02/1.4352E−14(≈)*	*1.0000E+02/1.4352E−14(≈)*	*1.0000E+02/1.4352E−14*
$f_{c_{23}}$	**3.4366E+02/3.0744E+00**(>)	3.4455E+02/3.4611E+00(>)	3.4509E+02/3.8689E+00(<)	3.4496E+02/3.7145E+00(<)	3.4490E+02/3.5301E+00
$f_{c_{24}}$	**4.2078E+02/2.5108E+00**(>)	4.2084E+02/2.6375E+00(>)	4.2181E+02/2.5577E+00(<)	4.2161E+02/3.1179E+00(<)	4.2120E+02/2.4336E+00
$f_{c_{25}}$	*3.8670E+02/5.4098E−03(≈)*	*3.8670E+02/6.1016E−03(≈)*	*3.8670E+02/6.0408E−03(≈)*	*3.8670E+02/5.9098E−03(≈)*	*3.8670E+02/6.9632E−03*
$f_{c_{26}}$	8.6136E+02/6.7571E+01(<)	8.6527E+02/4.8845E+01(<)	8.5323E+02/5.7156E+01(<)	**8.4308E+02/8.2112E+01**(>)	8.4647E+02/5.1296E+01
$f_{c_{27}}$	4.8763E+02/9.6722E+00(<)	4.8756E+02/8.9716E+00(<)	4.8878E+02/9.8656E+00(<)	4.8754E+02/9.6473E+00(<)	**4.8483E+02/9.9243E+00**
$f_{c_{28}}$	3.0873E+02/3.0250E+01(<)	3.0629E+02/2.5421E+01(<)	3.0894E+02/3.0947E+01(<)	3.1299E+02/3.5970E+01(<)	**3.0223E+02/1.5960E+01**
$f_{c_{29}}$	**4.2792E+02/9.9858E+00**(>)	4.2880E+02/5.3330E+00(>)	4.3029E+02/6.8836E+00(<)	4.2909E+02/6.9398E+00(>)	4.2934E+02/6.1298E+00
$f_{c_{30}}$	**1.9727E+03/1.5720E+01**(>)	1.9733E+03/1.3381E+01(>)	1.9757E+03/1.4987E+01(<)	1.9748E+03/1.5643E+01(<)	1.9735E+03/1.4684E+01
>/≈/<	11/3/16	7/6/17	11/6/13	11/5/14	−/−/−

Table 16
Time complexity of FDHD-DE and comparing algorithms.

Algorithms	T_0	T_1	$\hat{T}_2$	$\frac{\hat{T}_2 - T_1}{T_0}$
LSHADE	0.0583	0.6117	1.4834	14.9520
jSO			1.5076	15.3671
PaDE			1.7714	19.8919
LPalmDE			1.7435	29.7050
Hip-DE			1.9200	22.4401
FDHD-DE			1.5041	15.3070

Table 17
Comparison between FDHD-DE with five algorithms on pressure vessel design.

Algorithm	Max	Min	Mean	Std	p_S-value(α = 0.05)
FDHD-DE	**6.0597189E+03**	**6.0597120E+03**	6.0597162E+03	3.3431389E+01	–
EnMODE	6.0597142E+03	6.0597138E+03	**6.0597141E+03**	**9.2825565E−13**	<0.001
BPMAgES	6.0905916E+03	6.0597426E+03	6.0671363E+03	1.3431389E+01	<0.001
COLSHADE	6.0905262E+03	6.0597143E+03	6.0621792E+03	8.3590593E+01	<0.001
sCMAgES	6.0897175E+03	6.0597253E+03	6.0705769E−02	4.5648948E+01	<0.001
SASS	6.0797198E+03	6.0597135E+03	6.0706125E−02	5.8979797E+01	<0.001

To show the effectiveness of FDHD-DE, we select five evolutionary algorithms, including EnMODE (Sallam et al., 2020), BPMAgES (Hellwig and Beyer, 2020), COLSHADE (Gurrola-Ramos et al., 2020), sCMAgES (Kumar et al., 2020a) and SASS (Kumar et al., 2020b), which are specially proposed for real-world application. Each algorithm is run 30 times independently, and the results are listed in Table 17, where the Wilcoxon rank sum test is used to verify the superiority of FDHD-DE with the five comparing algorithm. From Table 17, we can observe that FDHD-DE obtains the best performance in terms of Max and Min. Therefore, we can conclude that FDHD-DE is a promising algorithm for solving real-world application problems.

5. Conclusion

In this paper, differential evolution with fitness-difference based parameter control and hypervolume diversity indicator (FDHD-DE) is proposed to alleviate the problem of premature convergence and stagnation in numerical optimization. In FDHD-DE, a novel semi-adaptive scheme for control parameters is proposed to improve the efficiency of success-history based parameter adaptation by separating the adaptation process into two stages. Besides, a novel fitness-based weighting strategy is proposed to update parameters in memory pool. Specially, fitness improvement between target vector and trial vector and fitness difference between successful trial vector and the current best vector contribute to the calculation of weight in Lehmer mean, thus achieve a better balance between exploration and exploitation. To mitigate the problem of stagnation during the later stages of evolution, a novel hypervolume-based diversity indicator is employed to measure the level of diversity throughout evolution. The detected stagnant individual will be update through a dimension exchange mechanism. To evaluate the performance of FDHD-DE, a total of 88 benchmark functions from CEC2013, CEC2014, and CEC2017 test suites are used and five powerful DE-based variants are selected for comparison. In addition, FDHD-DE is tested on a real-world application to verify its feasibility. Experiment results on benchmark functions and pressure vessel design demonstrate that FDHD-DE is effective both in benchmark functions and real-world optimization problems.

Although the proposed strategies in this paper have made significant progress compared to previous ones, they still fail to accurately assess the search characteristics of individuals and the evolution state of the population. Therefore, future research should focus on developing more effective evaluation mechanism for search characteristics and evolution state.

CRediT authorship contribution statement

Chongle Ren: Methodology, Software, Writing – original draft. **Zhenghao Song:** Methodology, Software, Writing – original draft.

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

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgments

This work is supported by the Natural Science Foundation of Fujian Province with the Grant No. 2021J05227.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.engappai.2024.108081.

References

Awad, N.H., Ali, M.Z., Suganthan, P.N., 2017. Ensemble sinusoidal differential covariance matrix adaptation with euclidean neighborhood for solving CEC2017 benchmark problems. In: 2017 IEEE Congress on Evolutionary Computation. CEC, IEEE, pp. 372–379.

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

Brest, J., Maučec, M.S., Bošković, B., 2017. Single objective real-parameter optimization: Algorithm jSO. In: 2017 IEEE Congress on Evolutionary Computation. CEC, IEEE, pp. 1311–1318.

Bujok, P., Kolenovsky, P., 2022. Eigen crossover in cooperative model of evolutionary algorithms applied to CEC 2022 single objective numerical optimisation. In: 2022 IEEE Congress on Evolutionary Computation. CEC, IEEE, pp. 1–8.

Dragoi, E.-N., Dafinescu, V., 2016. Parameter control and hybridization techniques in differential evolution: A survey. Artif. Intell. Rev. 45, 447–470.

Elsayed, S.M., Sarker, R.A., 2013. Differential evolution with automatic population injection scheme for constrained problems. In: 2013 IEEE Symposium on Differential Evolution. SDE, IEEE, pp. 112–118.

Gurrola-Ramos, J., Hernàndez-Aguirre, A., Dalmau-Cedeño, O., 2020. COLSHADE for real-world single-objective constrained optimization problems. In: 2020 IEEE Congress on Evolutionary Computation. CEC, IEEE, pp. 1–8.

Hellwig, M., Beyer, H.-G., 2020. A modified matrix adaptation evolution strategy with restarts for constrained real-world problems. In: 2020 IEEE Congress on Evolutionary Computation. CEC, IEEE, pp. 1–8.

Kumar, A., Das, S., Zelinka, I., 2020a. A modified covariance matrix adaptation evolution strategy for real-world constrained optimization problems. In: Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion. pp. 11–12.

Kumar, A., Das, S., Zelinka, I., 2020b. A self-adaptive spherical search algorithm for real-world constrained optimization problems. In: Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion. pp. 13–14.

Li, J., Meng, Z., 2023. Differential evolution with exponential crossover: An experimental analysis on numerical optimization. IEEE Access 11, 131677–131707.

Li, C., Sun, G., Deng, L., Qiao, L., Yang, G., 2023. A population state evaluation-based improvement framework for differential evolution. Inform. Sci. 629, 15–38.

Liang, J., Qu, B., Suganthan, P., 2013a. Problem Definitions and Evaluation Criteria for the CEC 2014 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization. Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore 635.

Liang, J., Qu, B., Suganthan, P., Hernández-Díaz, A.G., 2013b. Problem Definitions and Evaluation Criteria for the CEC 2013 Special Session on Real-Parameter Optimization. Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Nanyang Technological University, Singapore, Technical Report 201212, 34, pp. 281–295.

Liu, X., Sun, J., Zhang, Q., Wang, Z., Xu, Z., 2023. Learning to learn evolutionary algorithm: A learnable differential evolution. IEEE Trans. Emerg. Top. Comput. Intell..

Meng, Z., 2023. Dimension improvements based adaptation of control parameters in differential evolution: A fitness-value-independent approach. Expert Syst. Appl. 223, 119848.

Meng, Z., Chen, Y., 2023. Differential evolution with exponential crossover can be also competitive on numerical optimization. Appl. Soft Comput. 146, 110750.

Meng, Z., Pan, J.-S., 2016. Monkey king evolution: A new memetic evolutionary algorithm and its application in vehicle fuel consumption optimization. Knowl.-Based Syst. 97, 144–157.

Meng, Z., Pan, J.-S., 2018. Quasi-affine transformation evolution with external archive (QUATRE-EAR): An enhanced structure for differential evolution. Knowl.-Based Syst. 155, 35–53.

Meng, Z., Pan, J.-S., 2019. HARD-DE: hierarchical archive based mutation strategy with depth information of evolution for the enhancement of differential evolution on numerical optimization. IEEE Access 12832–12854.

Meng, Z., Pan, J.-S., Kong, L., 2018. Parameters with Adaptive Learning Mechanism (PALM) for the enhancement of differential evolution. Knowl.-Based Syst. 141, 92–112.

Meng, Z., Pan, J.-S., Tseng, K.-K., 2019. PaDE: An enhanced differential evolution algorithm with novel control parameter adaptation schemes for numerical optimization. Knowl.-Based Syst. 168, 80–99.

Meng, Z., Pan, J.-S., Xu, H., 2016. Quasi-Affine TRansformation Evolutionary (QUATRE) algorithm: A cooperative swarm based algorithm for global optimization. Knowl.-Based Syst. 109, 104–121.

Meng, Z., Song, Z., Shao, X., Zhang, J., Xu, H., 2023. FD-DE: Differential evolution with fitness deviation based adaptation in parameter control. ISA Trans 139, 272–290.

Meng, Z., Yang, C., 2021. Hip-DE: Historical population based mutation strategy in differential evolution with parameter adaptive mechanism. Inform. Sci. 562, 44–77.

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

Meng, Z., Zhang, J., 2023. QUATRE-EMS: QUATRE algorithm with novel adaptation of evolution matrix and selection operation for numerical optimization. Inform. Sci. 651, 119714.

Meng, Z., Zhong, Y., Yang, C., 2021. CS-DE: Cooperative strategy based differential evolution with population diversity enhancement. Inform. Sci. 577, 663–696.

Nadimi-Shahraki, M.H., Zamani, H., 2022. DMDE: Diversity-maintained multi-trial vector differential evolution algorithm for non-decomposition large-scale global optimization. Expert Syst. Appl. 198, 116895.

Reijnen, R., Zhang, Y., Bukhsh, Z., Guzek, M., 2022. Deep reinforcement learning for adaptive parameter control in differential evolution for multi-objective optimization. In: 2022 IEEE Symposium Series on Computational Intelligence. SSCI, IEEE, pp. 804–811.

Sallam, K.M., Elsayed, S.M., Chakrabortty, R.K., Ryan, M.J., 2020. Improved multi-operator differential evolution algorithm for solving unconstrained problems. In: 2020 IEEE Congress on Evolutionary Computation. CEC, IEEE, pp. 1–8.

Sharma, M., Komninos, A., López-Ibáñez, M., Kazakov, D., 2019. Deep reinforcement learning based parameter control in differential evolution. In: Proceedings of the Genetic and Evolutionary Computation Conference. pp. 709–717.

Song, Z., Ren, C., Meng, Z., 2023a. An adaptive differential evolution with opposition-learning based diversity enhancement. Expert Syst. Appl. 122942.

Song, Z., Ren, C., Meng, Z., 2023b. Differential evolution with perturbation mechanism and covariance matrix based stagnation indicator for numerical optimization. Swarm Evol. Comput. 101447.

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

Sun, J., Liu, X., Bäck, T., Xu, Z., 2021. Learning adaptive differential evolution algorithm from optimization experiences by policy gradient. IEEE Trans. Evol. Comput. 25 (4), 666–680.

Tanabe, R., Fukunaga, A.S., 2014. Improving the search performance of SHADE using linear population size reduction. In: 2014 IEEE Congress on Evolutionary Computation. CEC, IEEE, pp. 1658–1665.

Tian, M., Gao, X., 2019. An improved differential evolution with information intercrossing and sharing mechanism for numerical optimization. Swarm Evol. Comput. 50, 100341.

Viktorin, A., Senkerik, R., Pluhacek, M., Kadavy, T., Zamuda, A., 2019. Distance based parameter adaptation for success-history based differential evolution. Swarm Evol. Comput. 50, 100462.

Wang, H., Wang, W., Cui, Z., Zhou, X., Zhao, J., Li, Y., 2018. A new dynamic firefly algorithm for demand estimation of water resources. Inform. Sci. 438, 95–106.

Wolpert, D.H., Macready, W.G., 1997. No free lunch theorems for optimization. IEEE Trans. Evol. Comput. 1 (1), 67–82.

Wu, G., Mallipeddi, R., Suganthan, P.N., 2017. Problem Definitions and Evaluation Criteria for the CEC 2017 Competition on Constrained Real-Parameter Optimization. Technical report Zhengzhou, China, National University of Defense Technology, Changsha, Hunan, PR China and Kyungpook National University, Daegu, South Korea and Nanyang Technological University, Singapore, Technical Report.

Yang, M., Li, C., Cai, Z., Guan, J., 2014. Differential evolution with auto-enhanced population diversity. IEEE Trans. Cybern. 45 (2), 302–315.

Zhang, Q., Meng, Z., 2023a. Adaptive differential evolution algorithm based on deeply-informed mutation strategy and restart mechanism. Eng. Appl. Artif. Intell. 126, 107001.

Zhang, J., Meng, Z., 2023b. QUATRE-PM: Quasi-affine transformation evolution with perturbation mechanism. IEEE Access.

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