Received January 2, 2020, accepted January 16, 2020, date of publication January 20, 2020, date of current version January 28, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2968119

A Hybrid Differential Evolution Algorithm and Its Application in Unmanned Combat Aerial Vehicle Path Planning

JENG-SHYANG PAN[1,2,3], (Senior Member, IEEE), NENGXIAN LIU[1], AND SHU-CHUAN CHU[3]

[1]College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350116, China
[2]Fujian Provincial Key Lab of Big Data Mining and Applications, Fujian University of Technology, Fuzhou 350116, China
[3]College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China

Corresponding author: Nengxian Liu (lylnx@fzu.edu.cn)

ABSTRACT CIPDE and JADE are two powerful and effective Differential Evolution (DE) algorithms with strong exploration and exploitation capabilities. In order to take advantage of these two algorithms, we present a hybrid differential evolution algorithm combining modified CIPDE (MCIPDE) with modified JADE (MJADE) called CIJADE. In CIJADE, the population is first partitioned into two subpopulations according to the fitness value, i.e., superior and inferior subpopulations, to maintain the population diversity. The superior subpopulation evolves using the operation defined in MCIPDE. The MCIPDE adds an external archive to the mutation scheme to enhance the population diversity and exploration capability of original CIPDE. While the inferior subpopulation evolves using the operation defined in MJADE. The MJADE modifies the original JADE by adjusting the parameter p in linear decreasing way to balance the exploration and exploitation ability of original JADE. A new crossover operation is designed to original JADE to deal with the problem of stagnation. Furthermore, the parameters CR and F values of CIJADE are updated according to a modified parameter adaptation strategy in each generation. We validate the performance of the proposed CIJADE algorithm over 28 benchmark functions of the CEC2013 benchmark set. The experimental results indicate that the proposed CIJADE performs better than the eleven popular state-of-the-art DE variants. What's more, we apply the proposed CIJADE to deal with Unmanned Combat Aerial Vehicle (UCAV) path planning problem. The simulation results show that the proposed CIJADE can efficiently find the optimal or near optimal flight path for UCAV.

INDEX TERMS Differential evolution, hybrid algorithm, modified CIPDE, modified JADE, UCAV path planning.

I. INTRODUCTION

Differential evolution(DE) [1] first introduced by Storn and Price in 1995 is an efficient and robust optimization algorithm for dealing with different types of benchmark functions and practical optimization problems [2], [3]. DE is one of the most popular evolutionary algorithms (EAs) because it needs few control parameters, is very powerful, and is easy to implement. As an important paradigm of EAs, DE is also a population-based global stochastic search algorithm. DE utilizes three main operators mutation, crossover and selection to make the population move toward the global optimum gradually [4]. In the last two decades, DE and other meta-heuristic algorithms have been widely utilized to deal with a variety of scientific and real-world optimization problems, such as concept drift detectors in data streams [5], optimum bandwidth allocation [6], time series forecasting [7], job scheduling [8], maritime hybrid energy system [9] and vehicle fuel consumption [10].

The overall performance of DE is significantly affected by its three control parameters (i.e., scale factor F, crossover rate CR and population size NP) and trial vector generation schemes (i.e., mutation and crossover operators) [11], [12]. A lot of evidences show that the most proper

The associate editor coordinating the review of this manuscript and approving it for publication was Sotirios Goudos.

mutation schemes and parameters of DE are usually different when it is used to different types of optimization problems [4], [13], [14]. The reason is that DE has different exploitation and exploration capabilities when it owns different mutation schemes and parameter settings [2]. In fact, it is a time-consuming and challenging task to select proper control parameters and mutation operators for DE because different optimization problems possess different characteristics [15], [16]. It is still an open issue in the field of DE research.

So far, there are more than five mutation schemes (i.e., DE/rand/1, DE/best/1, DE/rand/2, DE/best/2, DE/target-to-rand/1, and DE/target-to-best/1) and more than two crossover operations (i.e., binomial and exponential crossover) in the DE family [3]. Among these mutation schemes, "DE/rand/1" and "DE/rand/2" are random in nature and they have strong exploration ability. While "DE/best/1" and "DE/best/2" are greedy in nature and they have strong exploitation ability. Zhang and Sanderson [17] proposed a very effective and powerful mutation scheme "DE/target-to-pbest/1" which is less greedy than other best individual guided mutation strategy and it can make a good balance between the exploration and exploitation capabilities. Recently, Zheng *et al.* [18] proposed a new powerful mutation scheme "DE/target-to-ci_mbest/1" which adopts collective information of top m good individuals. The experimental results demonstrated this mutation scheme also can make a good trade-off the exploration and exploitation capabilities. These mutations schemes are suitable for different problems. To make better use of these mutations schemes, many researchers have proposed several improved DE variants which adopt multiple mutations schemes adaptively and cooperatively [13], [15], [19]–[21]. Qin *et al.* [13] presented a self-adaptive DE (SaDE) in which four mutation schemes (i.e., "DE/rand/1", "DE/target-to-best/1", "DE/rand/2", "DE/target-to-rand/1") are adaptively selected by each individual according to their previous experiences of successfully generating solutions. Wang et al. presented a composite DE algorithm named CoDE [19] which employs three mutation schemes (i.e., "DE/rand/1", "DE/rand/2", "DE/target-to-rand/1") and three parameter control schemes in a random manner to generate candidate solutions. Mallipeddi *et al.* [20] presented a DE variant with ensemble of mutation schemes and parameters, named EPSDE, which has a strategy pool containing three distinct mutation strategies (i.e., "DE/best/2", "DE/rand/1" and "DE/target-to-rand/1"). Wu *et al.* [21] presented a multi-population ensemble DE, named MPEDE, which consists of three mutation schemes (i.e., "DE/rand/1", "DE/target-to-pbest/1" and "DE/target-to-rand/1") with dynamic resource allocation strategy. All of the above algorithms combine the mutation schemes having strong exploration ability with the mutation schemes having strong exploitation ability.

From a higher level point of view, different DE variants have different characteristics and show different abilities in solving different optimization problems [4]. Such as JADE [17] has strong exploitation ability and is suitable for tackling unimodal and simple multimodal functions, and CoDE [19] has strong exploration ability and is suitable for dealing with complicated multimodal functions [22]. Many powerful and efficient DE variants have been proposed in last few decades. These DE variants have their own advantages and disadvantages. In order to make the full use of the advantages of these DE variants, some researchers have used ensemble and hybrid technology to combine these DE variants, and then put forward even better DE algorithms. Wu *et al.* [4] proposed a high-level ensemble of different DE variants (EDEV) which consists of three highly popular and efficient DE variants, namely JADE, CoDE and ESPDE. Li *et al.* [22] proposed hybrid DE algorithm HMJCDE which uses hybrid technology to combine the two well-known DE variants JADE and CoDE. DE has been also mixed with other swarm intelligence algorithms, such as particle swarm optimization (PSO) [23] and Artificial Bee Colony [24]. These hybrid algorithms can take advantage of different methods and enhance the overall optimization performance. However, there are no work to mix two powerful DE variants JADE and CIPDE [18] to combine their advantages.

Meanwhile, population partitioning approaches for improving the performance of swarm intelligence algorithms, such as DE and PSO, attracted more and more attention [4], [21], [25]–[27]. Dividing the whole population into multiple sub-populations can maintain the population diversity [28]. Zhong *et al.* [25] presented a DE variant DP-DE based on dual populations. In the DP-DE, first population focuses on global search and the second one focuses on local fine tuning. Wu *et al.* [4] proposed multi-population based framework (MPF) to implement the ensemble of several DE variants named EDEV. Yu and Zhang [26] presented a MPDE in which each subpopulation uses the same mutation scheme "DE/best/1" in the evolution.

Based on the above consideration, a hybrid differential evolution algorithm combining modified CIPDE with modified JADE called CIJADE is proposed in this paper. In the CIJADE, the population is first divided into two subpopulation according to the fitness value, i.e., superior and inferior subpopulations. The superior subpopulation evolves using the operation defined in modified CIPDE and the inferior subpopulation evolves using the operation defined in modified JADE. Furthermore, the parameters CR and F values are updated according to a modified parameter adaptation strategy in each generation. CIJADE has been verified on 28 benchmark functions developed for the 2013 IEEE Congress on Evolutionary Computation (IEEE CEC2013). The experimental results indicate that the proposed CIJADE performs better than the comparing algorithms. Moreover, we also apply the CIJADE to the unmanned combat aerial vehicle (UCAV) path planning problem and compare it with other meta-heuristic algorithms.

The rest of this paper is organized as follows. Section 2 briefly introduces the standard DE, JADE and CIPDE. Then, our hybrid DE algorithm combining modified CIPDE with modified JADE, called CIJADE, is given in Section 3. Section 4 presents the experimental results on

CEC2013 benchmark set. Moreover, the application of the CIJADE to the UCAV path planning problem is given in Section 5. Finally, Section 6 concludes this study.

II. RELATED WORKS

In this section, we briefly introduce basic DE, JADE and CIPDE algorithms. They are closely related to our proposed algorithm CIJADE. Some reviews on other state-of-the-art DE variants such as jDE, SaDE, and CoDE can be found in the previous paper of Meng *et al.* [14]. A detailed review of other improved DE variants can be found in the elaborate surveys [2], [3].

A. DIFFERENTIAL EVOLUTION (DE)

Similar to other EAs, DE has three main operators: mutation, crossover, and selection. The DE population is represented by a set of real parameter vectors $X_{i,g} = (x_{i,1,g}, x_{i,2,g}, \ldots, x_{i,D,g})$, $i = 1, 2, \ldots, NP$, $g = 1, 2, \ldots, G_{\max}$, where g is the generation number, NP is the population size, and D is the dimension of the target problem. DE begins with an initial population including NP D-dimensional individuals randomly generated by the following equation.

$$x_{i,j,0} = x_{\min,j} + rand(0,1) \cdot (x_{\max,j} - x_{\min,j}) \quad j = 1, 2, \ldots, D \tag{1}$$

where $i = 1, 2, \ldots, NP$ represents the index of the individual, j represents the variable index in the ith individual at the generation $g = 0$, $rand(0, 1)$ is a uniformly distributed random real number in the interval [0, 1], and $x_{\min,j}$, $x_{\max,j}$ are the lower and the upper bounds of the variable $x_{i,j}$. Then DE enters a cycle of evolutionary operations including mutation, crossover and selection until a termination criterion is met. These three components are presented in detail as follows.

Mutation: DE performs the mutation operation to create a donor vector $V_{i,g} = (v_{i,1,g}, v_{i,2,g}, \ldots, v_{i,D,g})$ for each target vector of the population. Six commonly used mutation schemes are given as follows.

DE/rand/1:

$$V_{i,g} = X_{r1,g} + F \cdot (X_{r2,g} - X_{r3,g}) \tag{2}$$

DE/rand/2:

$$V_{i,g} = X_{r1,g} + F \cdot (X_{r2,g} - X_{r3,g}) + F \cdot (X_{r4,g} - X_{r5,g}) \tag{3}$$

DE/best/1:

$$V_{i,g} = X_{best,g} + F \cdot (X_{r1,g} - X_{r2,g}) \tag{4}$$

DE/best/2:

$$V_{i,g} = X_{best,g} + F \cdot (X_{r1,g} - X_{r2,g}) + F \cdot (X_{r3,g} - X_{r4,g}) \tag{5}$$

DE/target-to-best/1:

$$V_{i,g} = X_{i,g} + F \cdot (X_{best,g} - X_{i,g}) + F \cdot (X_{r1,g} - X_{r2,g}) \tag{6}$$

DE/target-to-rand/1:

$$V_{i,g} = X_{i,g} + F \cdot (X_{r1,g} - X_{i,g}) + F \cdot (X_{r2,g} - X_{r3,g}) \tag{7}$$

where g stands for the generation, $X_{best,g}$ is the best individual in current generation, the indices $r1, \ldots, r5$ are randomly chosen from the interval $[1, NP]$ and $r1 \neq r2 \neq r3 \neq r4 \neq r5 \neq i$. A positive control parameter F is called scaling factor, which is adopted to scale the difference vectors.

Crossover: DE performs the crossover operation to create a trial vector $U_{i,g} = (u_{i,1,g}, u_{i,2,g}, \ldots, u_{i,D,g})$ for each individual by crossing the mutation vector and target vector. The most commonly used binomial crossover operation is formulated as follows.

$$u_{i,j,g} = \begin{cases} v_{i,j,g} & \textit{if } (rand_{i,j}(0,1) \leq CR \textit{ or } j = j_{rand}) \\ x_{i,j,g} & \textit{otherwise} \end{cases} \tag{8}$$

where $CR \in [0, 1]$ is a user-defined parameter called crossover rate, $rand_{i,j}(0, 1) \in [0, 1]$ is a uniformly distributed random number, and j_{rand} is a uniform random integer between 1 and D.

Selection: DE performs the selection operation by using a one-to-one competition to select the trial vector or the target vector to enter the next generation based on the fitness value. For a minimization problem, the selection operation is formulated as follows.

$$X_{i,g+1} = \begin{cases} U_{i,g} & \textit{if } f(U_{i,g}) \leq f(X_{i,g}) \\ X_{i,g} & \textit{otherwise} \end{cases} \tag{9}$$

where $f(.)$ are the fitness values of the trial vector $U_{i,g}$ and the target vector $X_{i,g}$.

B. JADE

JADE [17] has two main features, i.e., a new mutation scheme with an optional external archive and the control parameter values of the F, Cr are updated in adaptive way.

The new mutation scheme is denoted by DE/target-to-pbest/1, which is given in Eqs. (10) and (11). It is a variant of the DE/target-to-best/1.

Mutation without archive:

$$V_{i,g} = X_{i,g} + F \cdot (X_{pbest,g} - X_{i,g}) + F \cdot (X_{r1,g} - X_{r2,g}) \tag{10}$$

Mutation with archive:

$$V_{i,g} = X_{i,g} + F \cdot (X_{pbest,g} - X_{i,g}) + F \cdot (X_{r1,g} - \overline{X_{r2,g}}) \tag{11}$$

This new mutation scheme adopts the optional external archive to reserve the population diversity and give useful information on the promising evolutionary direction. In Eqs. (10) and (11), the $X_{pbest,g}$ represents the randomly selected individual from the top $100p\%(p \in (0, 1])$ individuals at the g generation. The control parameter p determines the greediness of this new mutation scheme and balances exploitation and exploration. The difference between Eq. (10) and Eq. (11) depends on the second difference vector. $X_{r2,g}$ in Eq. (10) is selected from the current population solutions P while $\overline{X_{r2,g}}$ in Eq. (11) is selected from the union $P \cup A$, where A is a set of archived inferior solutions. At the beginning, the archive A is empty and then unsuccessful target vectors (inferior solutions) are added to A at each generation.

The archive size is set to the same as the population size. If the number of inferior solutions exceeds archive size, randomly selected inferior solutions are removed from the archive A to make space for the newly inserted vectors.

In JADE, control parameters F, Cr for each vector are updated in an adaptive way. F is updated according to a Cauchy distribution, which is shown in Eq. (12). Cr is updated according to a Gaussian distribution, which is shown in Eq. (13).

$$F = randc(\mu F, 0.1) \tag{12}$$
$$Cr = randn(\mu Cr, 0.1) \tag{13}$$

where both μF and μCr are initialized to 0.5. When the value of Cr is outside of the range [0, 1], it is replaced by the boundary value (0 or 1) close to the generated value. F is truncated to 1 when $F > 1$, while F is regenerated when $F < 0$. In each generation, successful values for F and Cr are saved in the set of S_F and S_{Cr}, and utilized to update μF and μCr. At the end of the generation, when S_F and S_{Cr} are not empty, μF and μCr are updated as follows.

$$\mu F = (1 - c) \cdot \mu F + c \cdot mean_L(S_F) \tag{14}$$
$$\mu Cr = (1 - c) \cdot \mu Cr + c \cdot mean_A(S_{Cr}) \tag{15}$$

where c is a learning rate, and usually $1/c \in [5, 20]$. $Mean_A(.)$ is the usual arithmetic mean while $mean_L(.)$ is the Lehmer mean which is defined as follows [18].

$$mean_L(S_F) = \sum\nolimits_{F \in S_F} F^2 / \sum\nolimits_{F \in S_F} F \tag{16}$$

C. CIPDE

CIPDE [18] also has two main features, i.e., a new mutation scheme collective information-based mutation (CIM) and a new collective information-based crossover (CIX) to deal with stagnation. CIPDE uses the parameter control scheme proposed in JADE.

The new mutation scheme CIM is denoted by DE/target-to-ci_mbest/1, which is given in Eq. (17). CIM uses the collective information of top m good vectors corresponding to the target vector.

$$V_{i,g} = X_{\mathrm{i},g} + F \cdot (X_{\mathrm{ci\_mbest}^i,g} - X_{\mathrm{i},g}) + F \cdot (X_{\mathrm{r1},g} - X_{\mathrm{r2},g}) \tag{17}$$

where target vector $X_{\mathrm{i},g}$ with a fitness ranking of i, $X_{\mathrm{r1},g}$ and $X_{\mathrm{r2},g}$ are randomly chosen distinct vector and different from base vector. The collective vector $X_{\mathrm{ci\_mbest}^i,g}$ is defined as follows.

$$X_{\mathrm{ci\_mbest}^i,g} = \sum_{k=1}^{m} w_k \cdot X_{k,g} \tag{18}$$

$X_{\mathrm{ci\_mbest}^i,g}$ is a linear weighting combination of the top $m, m \in [1, \mathrm{i}]$, vectors in population with fitness values better than or equal to $X_{\mathrm{i,g}}$. w_k is a weighting factor using to denote the contributions of different vectors, which is defined as Eq. (19).

$$w_k = \frac{(m - k + 1)}{(1 + 2 + \ldots + m)} \quad for\ k = 1, 2, \ldots, m \tag{19}$$

In order to deal with the stagnation problem of DE, the authors proposed the collective information-based crossover (CIX). Stagnation is the situation where the algorithm cannot produce better trivial vectors even though the population still has a certain level of diversity. In the literature [18], they use the consecutive unsuccessful update CUU_i(i = 1, 2, ..., NP) to identify the stagnant individual and it can be defined as Eq. (20).

$$CUU_{i,g+1} = \begin{cases} 0 & if\ f(U_{i,g}) \le f(X_{i,g}) \\ CUU_{i+g} + 1 & otherwise \end{cases} \tag{20}$$

where $CUU_{i,g}$ stands for the consecutive unsuccessful update of the ith population individual, whose initial value is 0. If $CUU_{i,g} > T$, where T represents the user-defined threshold of stagnation, which means that ith individual is stagnant [18], [29], [30]. In the literature [18], the recommended value of T is set to 90. The individual uses the Eq. (21) as crossover operation when the stagnation occurs, otherwise, it uses the Eq. (8) as crossover operation.

$$u_{i,j,g} = \begin{cases} v_{i,j,g} & if\ (rand_{i,j}(0,1) \le CR\ or\ j = j_{rand}) \\ x_{\mathrm{ci\_mbest}^i,j,g} & otherwise \end{cases} \tag{21}$$

where $x_{\mathrm{ci\_mbest}^i,j,g}$ is the jth variable of the $X_{\mathrm{ci\_mbest}^i,g}$.

III. THE PROPOSED ALGORITHM CIJADE

The main idea of our algorithm CIJADE is described in this section. In CIJADE, we adopt a dual-population framework to hybrid the modified CIPDE and modified JADE. CIJADE can combine the advantages of both modified CIPDE and modified JADE. Figure 1 shows an illustration of the main framework of our proposed CIJADE algorithm. From Figure 1, we can see that CIJADE is composed of population initialization, population division with sort strategy, subpopulation evolution through MCIPDE algorithm or MJADE algorithm, and parameters update.

A. POPULATION DIVISION

Dual or multi subpopulations and every subpopulation adopting different evolutionary method is an effective and efficient technology to enhance evolutionary algorithms' performance [4], [21], [25], since this technology can benefit the evolutionary algorithms to make good trade-off between exploration and exploitation ability during evolution process, reserve population diversity and avoid premature convergence. Therefore, in our proposed CIJADE, we divide the whole population into two subpopulations with sort strategy based on the fitness values, name $pop_{superior}$ and $pop_{\inf erior}$. NP_1, NP_2 are the sizes of $pop_{superior}$ and $pop_{\inf erior}$, respectively. λ is the proportion between NP_1 and NP. $NP_1 + NP_2 = NP$. In CIJADE, subpopulation $pop_{superior}$ evolves by employing the operation defined in MCIPDE. The reason is that the individuals in $pop_{superior}$ have better fitness values using the collective information of top m best vectors to guide them can avoid trapping local optima. While, subpopulation

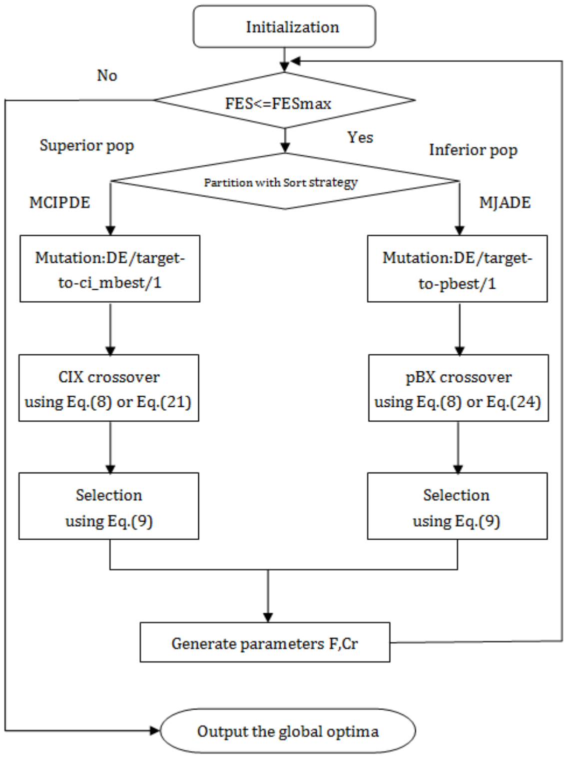


FIGURE 1. The main framework of CIJADE.

$pop_{inferior}$ evolves by employing the operation defined in MJADE. The reason is that the individuals in $pop_{inferior}$ have worse fitness values using one of the top $100p\%$ best vectors to guide them can accelerate convergence speed.

B. MODIFIED CIPDE: MCIPDE

In CIPDE, the authors proposed a new mutation scheme collective information-based mutation (CIM), which is effective and efficient. However, the performance of this mutation scheme still can be improved. An external archive A is integrated into this new mutation scheme for the improvement of population diversity and exploration ability, which saves the inferior solutions defeated by their corresponding trial vectors in selection process. The modified mutation scheme is defined as Eq. (22).

$$V_{i,g} = X_{\mathrm{i},g} + F \cdot (X_{\mathrm{ci\_mbest}^i,g} - X_{\mathrm{i},g}) + F \cdot (X_{\mathrm{r1},g} - \overline{X_{\mathrm{r2},g}}) \tag{22}$$

where $\overline{X_{\mathrm{r2},g}}$ is selected from the union $P \cup A$.

C. MODIFIED JADE: MJADE

In JADE, the authors proposed an effective and powerful mutation scheme DE/target-to-pbest/1 in Eqs. (10) and (11). In this study, we use the DE/target-to-pbest/1 with archive, i.e., Eq. (11). The parameter p is used to balance the greediness of this mutation scheme. However, in original JADE, the value of parameter p is static and set manually. In this study, the value of p is linearly adjusted as follows.

$$p = p_{\max} - (p_{\max} - p_{\min}) \cdot gen/(MaxGen) \tag{23}$$

where $p_{\max}$ and $p_{\min}$ are the maximum value and the minimum value for p, respectively. gen is the number of current generation, $gen = [1, 2, \ldots, MaxGen]$, $MaxGen$ is the maximum number of generations.

In JADE, the Eq. (8) is used to perform the crossover operation. However, the effective of standard crossover operation in original DE can be improved. In paper [18], the authors use the collective vector to perform the crossover operation when stagnation occurs. Their experiment results demonstrate that the proposed crossover operation is an effective and efficient crossover to deal with stagnation problem. Inspired by this, a p-Best crossover proposed in [31] is incorporated in the crossover operation of MJADE, named the modified crossover operation pBX. When the ith target vector traps in stagnation, i.e., $CUU_{i,g} > T$, pBX is defined as follows.

$$u_{i,j,g} = \begin{cases} v_{i,j,g} & if\ rand \le CR\ or\ j = j_{rand} \\ x_{pbest,j,g} & otherwise \end{cases} \tag{24}$$

where $X_{\mathrm{pbest},g}$ represents the randomly selected individual from the top $100p\%$ individuals at the g generation.

D. PARAMETER ADAPTION

The parameter control scheme has an important influence on the performance of DE algorithm. An effective adaptive parameter control scheme is proposed in JADE in which each individual $X_{i,g}$ has its own $F_{i,g}$ and $Cr_{i,g}$. They obey the Cauchy distribution and Gaussian distribution, respectively. Since this parameter control scheme is very effective, it was also adopted by the CIPDE. However, the authors of paper [32] found that this parameter control scheme can be improved by repairing the $Cr_{i,g}$ value. This effective repairing method is also incorporated in our proposed CIJADE. To clearly explain the repair technique, we rewrite the binomial crossover Eq. (8) as follows [32].

$$b_{i,j} = \begin{cases} 1 & if\ rand \le CR\ or\ j = j_{rand} \\ 0 & otherwise \end{cases} \tag{25}$$

$$u_{i,j,g} = b_{i,j} \cdot v_{i,j,g} + (1 - b_{i,j}) \cdot x_{i,j,g} \tag{26}$$

Suppose that the repaired crossover rate is denoted by $\overline{Cr_{i,g}}$, which is calculated as Eq. (27).

$$\overline{Cr_{i,g}} = \frac{\sum_{j=1}^{D} b_{i,j}}{D} \tag{27}$$

The pseudo-code of the proposed CIJADE algorithm is summarized in Algorithm 1.

IV. EXPERIMENTS ON BENCHMARK FUNCTIONS

To assess the efficiency of CIJADE, a test suit with 28 well-benchmarked optimization functions proposed for CEC2013 special session on real-parameter single objective optimization is adopted [33]. These 28 functions can be divided

Algorithm 1 CIJADE Algorithm

Initialization: Initialize the population P randomly, set $FES = 0$, $FESmax = 10000 * D$, $\mu F = 0.5$, $\mu CR = 0.5$, $T = 90, p_{\max} = 0.2, p_{\min} = 0.1, c = 0.1, CUU_{i=1:NP} = 0$, $\lambda = 0.2$

Calculate each individual's fitness value;
$FES = FES + NP$;

while $FES \leq FES$ max**do**

$S_F = \emptyset, S_{CR} = \emptyset$;

Sort the population P based on the fitness value and then partition the population into $pop_{\sup erior}$ and $pop_{\inf erior}$

// $pop_{\sup erior}$ evolve

for $i = 1; i \leq NP_1; i++$**do**

Generate F_i according to Eq (12), generate CR_i according to Eq (13);

Generate mutant vector $V_{i,g}$ according to Eq (22);

if $CUU_i < T$ **then**

Generate trial vector $U_{i,g}$ according to Eq (8);

else

Generate trial vector $U_{i,g}$ according to Eq (21);

end if

Calculate the fitness value $f(U_i)$; $FES = FES + 1$;

if $f(U_i) \leq f(X_i)$ **then**

$X_{i,g+1} = U_{i,g}$; $CUU_i = 0$; $F_i \rightarrow S_F$; $\overline{CR_i} \rightarrow S_{CR}$;

else

$X_{i,g+1} = X_{i,g}$; $CUU_i = CUU_i + 1$;

end if

end for

// $pop_{\inf erior}$ evolve

Update parameter p using Eq (23)

for$i = 1; i \leq NP_2; i++$**do**

Generate F_i according to Eq (12), Generate CR_i according to Eq (13);

Generate mutant vector $V_{i,g}$ according to Eq (11);

if $CUU_i < T$ **then**

Generate trial vector $U_{i,g}$ according to Eq (8);

else

Generate trial vector $U_{i,g}$ according to Eq (24);

end if

Calculate the fitness value $f(U_i)$; $FES = FES + 1$;

if $f(U_i) \leq f(X_i)$ **then**

$X_{i,g+1} = U_{i,g}$; $CUU_i = 0$; $F_i \rightarrow S_F$; $\overline{CR_i} \rightarrow S_{CR}$;

else

$X_{i,g+1} = X_{i,g}$; $CUU_i = CUU_i + 1$;

end if

end for

Update μF using Eq. (14); Update μCR using Eq.(15);

$Gen = Gen + 1$;

end while

into three categories, F1-F5 are unimodal functions, F6-F20 are basic multimodal functions, and F21-F28 are composition functions. All of them are considered as black-box test functions, and their search ranges are $[-100, 100]^D$ (D is the number of decision dimensions). The full description of all these benchmark functions are given in [33]. And these 28 benchmark functions are rotated and shifted to the same global optimum O=$\{o_1, o_2, \ldots, o_D\}^T$ [33].

TABLE 1. Parameter configurations for all of these comparative algorithms.

Algorithms	Parameters settings
jDE	$NP = 100, CR = 0.9,$ $\tau_1 = \tau_2 = 0.1, F_1 = 0.1, F_u = 0.9$
JADE	$NP = 100, c = 0.1, \mu F = 0.5,$ $\mu CR = 0.5, p = 0.05$
MJADE	$NP = 100, c = 0.1, \mu F = 0.5, \mu CR = 0.5,$ $p_{\max} = 0.2, p_{\min} = 0.1, T = 90$
SaDE	$NP = 50, F \sim N(0.5, 0.3), \mu CR = 0.5,$ $CR \sim N(\mu CR, 0.1), LG = 50$
CoDE	$NP = 30, F = 1.0, CR = 0.1, or\ F = 1.0,$ $CR = 0.9\ or\ F = 0.8, CR = 0.2$
Rcr-JADE	$NP = 100, c = 0.1, \mu F = 0.5,$ $\mu CR = 0.5, p = 0.05$
CobiDE	$NP = 100, pb = 0.4, ps = 0.5,$ $\mu F = 0.65\ or\ 1.0, \mu CR = 0.1\ or\ 0.95$
MPEDE	$NP = 250, \lambda_1 = 0.2, ng = 20$
AGDE	$NP = 50, p = 0.1, F \in [0.1, 1],$ $CR_1 \in [0.05, 0.15], CR_2 \in [0.9, 1]$
CIPDE	$NP = 100, c = 0.1, \mu F = 0.7,$ $\mu CR = 0.5, T = 90$
MCIPDE	$NP = 100, c = 0.1, \mu F = 0.7,$ $\mu CR = 0.5, T = 90$
EFADE	$NP = 50, F_i = rand(0, k_i),$ $CR_1 \in [0.05, 0.15], CR_2 \in [0.9, 1]$
EDEV	$NP = 100, \lambda_1 = \lambda_2 = \lambda_3 = 0.1,$ $\lambda_4 = 0.7, ng = 20$
CIJADE	$NP = 100, c = 0.1, \mu F = 0.5, \mu CR = 0.5,$ $p_{\max} = 0.2, p_{\min} = 0.1, T = 90, \lambda = 0.2$

In our experiments, the numbers of variables D for all benchmark functions are set to 50 and 100. The maximal function evaluation (FESmax) is set to $D \times 10^4$ according to the guidelines provided in the special session of CEC2013 [33]. The comparative algorithms are run 20 times independently over the benchmark functions due to the stochastic property of comparative algorithms. Both the mean and the standard deviation of the function error values $\Delta f = f_i - f_i^*$ are reported. The Wilcoxon signed-rank test at a 0.05 significance level is used for comparing CIJADE with each comparative algorithm on each function. Symbols "-" and "+" represent that one comparative algorithm is significantly worse performance and better performance than CIJADE, respectively, while "=" means

TABLE 2. Comparison results of mean and standard deviation of 20-run fitness error of CIJADE with other state-of-the-art DE variants.

50D	jDE Mean/Std	JADE Mean/Std	SaDE Mean/Std	CoDE Mean/Std	Rcr-JADE Mean/Std	CobiDE Mean/Std
1	1.59E-13/1.07E-13(-)	1.82E-13/9.33E-14(-)	0.00E+00/0.00E+00(=)	0.00E+00/0.00E+00(=)	1.82E-13/9.33E-14(-)	1.14E-13/1.17E-13(-)
2	4.76E+05/1.41E+05(-)	2.98E+04/1.59E+04(=)	1.46E+05/5.83E+04(-)	2.57E+05/1.05E+05(-)	2.24E+04/1.35E+04(=)	3.55E+05/1.30E+05(-)
3	9.84E+06/2.59E+07(-)	1.65E+06/3.88E+06(=)	4.43E+06/8.84E+06(-)	1.46E+07/2.62E+07(-)	1.56E+06/3.09E+06(=)	9.46E+05/1.20E+06(=)
4	1.53E+02/1.71E+02(-)	1.02E-04/1.28E-04(+)	1.87E+02/1.11E+02(-)	1.38E-01/1.16E-01(-)	7.19E+03/1.79E+04(=)	3.61E-02/2.91E-02(-)
5	1.31E-13/4.16E-14(=)	1.36E-13/4.67E-14(=)	0.00E+00/0.00E+00(+)	5.68E-14/5.83E-14(+)	1.25E-13/3.50E-14(=)	2.17E-12/8.81E-13(-)
6	4.36E+01/2.40E-01(-)	4.34E+01/8.14E-14(=)	4.38E+01/1.27E+00(-)	4.37E+01/1.27E+00(=)	4.37E+01/1.27E+00(=)	4.34E+01/6.20E-09(=)
7	1.60E+01/8.55E+00(=)	2.26E+01/1.20E+01(=)	2.69E+01/8.02E+00(-)	3.44E+01/9.98E+00(-)	1.99E+01/8.62E+00(=)	1.37E+01/4.54E+00(=)
8	2.11E+01/4.47E-02(=)	2.11E+01/3.86E-02(=)	2.11E+01/4.46E-02(=)	2.10E+01/5.97E-02(+)	2.11E+01/5.24E-02(=)	2.11E+01/4.27E-02(=)
9	5.24E+01/6.06E+00(-)	5.34E+01/2.64E+00(-)	3.23E+01/4.26E+00(+)	3.35E+01/5.52E+00(+)	5.41E+01/2.18E+00(-)	2.08E+01/2.61E+00(+)
10	6.92E-02/5.22E-02(-)	4.44E-02/2.77E-02(=)	8.93E-02/4.44E-02(-)	4.98E-02/2.66E-02(-)	4.72E-02/2.76E-02(-)	2.71E-02/1.98E-02(=)
11	3.69E-14/2.78E-14(-)	2.84E-15/1.27E-14(=)	0.00E+00/0.00E+00(=)	5.47E-01/7.55E-01(-)	0.00E+00/0.00E+00(=)	4.13E+01/4.25E+00(-)
12	1.17E+02/1.60E+01(-)	5.81E+01/7.27E+00(-)	8.33E+01/1.24E+01(-)	9.71E+01/2.63E+01(-)	5.70E+01/8.56E+00(=)	7.41E+01/1.90E+01(-)
13	1.74E+02/2.53E+01(-)	1.23E+02/1.99E+01(=)	1.87E+02/3.25E+01(-)	1.83E+02/2.98E+01(-)	1.32E+02/2.79E+01(-)	1.36E+02/5.14E+01(=)
14	2.52E-03/4.89E-03(+)	5.00E-02/2.14E-02(=)	4.72E+02/9.31E+01(-)	2.76E+01/1.41E+01(-)	4.06E-02/1.66E-02(=)	2.73E+03/2.89E+02(-)
15	9.95E+03/4.48E+02(-)	7.07E+03/4.96E+02(-)	1.20E+04/4.42E+02(-)	6.70E+03/5.60E+02(-)	6.92E+03/3.97E+02(-)	6.08E+03/8.24E+02(=)
16	3.23E+00/2.59E-01(=)	2.12E+00/6.84E-01(=)	3.15E+00/2.59E-01(=)	9.27E-01/3.75E-01(+)	2.07E+00/7.04E-01(=)	1.23E+00/1.40E+00(+)
17	5.08E+01/6.90E-14(=)	5.08E+01/3.91E-14(=)	7.48E+01/2.23E+00(-)	5.20E+01/5.37E-01(-)	5.08E+01/2.93E-14(=)	1.31E+02/5.49E+00(-)
18	2.82E+02/2.26E+01(-)	1.45E+02/8.92E+00(-)	3.26E+02/1.35E+01(-)	1.21E+02/1.56E+01(-)	1.45E+02/1.23E+01(-)	1.07E+02/2.07E+01(-)
19	2.87E+00/1.74E-01(-)	2.80E+00/2.81E-01(-)	9.56E+00/1.14E+00(-)	3.22E+00/6.52E-01(-)	2.70E+00/2.36E-01(-)	1.42E+01/1.13E+00(-)
20	2.14E+01/4.44E-01(-)	1.96E+01/7.91E-01(-)	2.03E+01/4.31E-01(-)	1.96E+01/8.11E-01(-)	1.97E+01/6.37E-01(-)	2.08E+01/8.36E-01(-)
21	6.15E+02/4.71E+02(=)	8.34E+02/3.92E+02(=)	9.30E+02/2.84E+02(=)	7.14E+02/4.04E+02(=)	6.64E+02/4.42E+02(=)	2.92E+02/2.84E+02(+)
22	2.40E+01/2.07E+01(-)	1.22E+01/4.31E+00(=)	4.85E+02/2.52E+02(-)	3.67E+01/1.00E+01(-)	1.15E+01/9.85E-01(-)	3.44E+03/5.28E+02(-)
23	9.86E+03/5.14E+02(-)	7.24E+03/6.87E+02(-)	1.22E+04/5.38E+02(-)	7.32E+03/7.89E+02(-)	7.09E+03/6.52E+02(-)	6.01E+03/7.65E+02(-)
24	2.58E+02/1.12E+01(-)	2.57E+02/2.54E+01(=)	2.50E+02/8.09E+00(=)	2.61E+02/1.21E+01(-)	2.55E+02/2.50E+01(=)	2.20E+02/1.05E+01(+)
25	3.16E+02/2.57E+01(=)	3.54E+02/2.11E+01(-)	3.23E+02/1.21E+01(=)	3.14E+02/1.11E+01(=)	3.62E+02/9.40E+00(-)	2.89E+02/9.59E+00(+)
26	3.04E+02/1.08E+02(=)	3.78E+02/9.13E+01(-)	2.31E+02/6.27E+01(+)	2.75E+02/8.56E+01(+)	3.54E+02/1.03E+02(-)	2.38E+02/6.83E+01(+)
27	1.20E+03/2.68E+02(-)	1.42E+03/2.79E+02(-)	9.25E+02/5.12E+01(=)	1.11E+03/1.25E+02(=)	1.47E+03/2.40E+02(-)	8.04E+02/1.33E+02(+)
28	4.00E+02/1.75E-13(=)	6.95E+02/9.06E+02(=)	4.00E+02/7.38E-14(=)	4.00E+02/1.40E-13(=)	4.00E+02/1.75E-13(=)	4.00E+02/2.57E-11(=)
+/=/-	1/9/18	1/16/11	3/9/16	5/6/17	0/15/13	7/8/13

50D	MPEDE Mean/Std	AGDE Mean/Std	CIPDE Mean/Std	EFADE Mean/Std	EDEV Mean/Std	CIJADE Mean/Std
1	1.93E-13/8.33E-14(-)	2.16E-13/5.08E-14(-)	1.36E-13/1.14E-13(-)	9.09E-14/1.14E-13(=)	0.00E+00/0.00E+00(=)	3.41E-14/8.33E-14
2	9.17E+04/3.73E+04(-)	9.85E+04/4.76E+04(-)	2.54E+04/9.09E+03(=)	1.86E+05/7.59E+04(-)	5.08E+04/2.54E+04(-)	2.57E+04/1.11E+04
3	4.61E+05/7.90E+05(=)	3.24E+07/5.03E+07(-)	4.95E+06/1.06E+07(-)	2.22E+06/2.61E+06(-)	3.93E+06/5.58E+06(=)	1.09E+06/2.13E+06
4	4.21E-01/1.36E+00(=)	4.96E+00/4.43E+00(-)	4.61E+03/9.70E+03(-)	1.21E+01/9.29E+00(-)	1.23E+01/1.71E+01(-)	9.14E+03/1.45E+04
5	1.42E-13/5.05E-14(=)	1.25E-13/3.50E-14(=)	2.22E-13/5.80E-14(-)	1.14E-13/0.00E+00(=)	0.00E+00/0.00E+00(+)	1.25E-13/3.50E-14
6	4.37E+01/1.27E+00(=)	3.95E+01/1.14E+01(=)	4.37E+01/1.27E+00(=)	4.34E+01/6.48E-13(=)	3.77E+01/3.07E+00(+)	4.34E+01/6.25E-14
7	1.39E+01/6.86E+00(=)	5.11E+01/1.42E+01(-)	1.66E+01/6.55E+00(=)	1.74E+01/5.15E+00(=)	2.09E+01/8.28E+00(=)	1.69E+01/6.59E+00
8	2.11E+01/2.39E-02(=)	2.11E+01/2.94E-02(=)	2.11E+01/2.05E-02(=)	2.11E+01/3.71E-02(=)	2.11E+01/2.35E-02(=)	2.11E+01/4.35E-02
9	3.05E+01/4.14E+00(+)	3.91E+01/1.70E+01(=)	4.12E+01/5.54E+00(=)	3.14E+01/4.87E+00(+)	4.65E+01/8.23E+00(-)	3.80E+01/3.95E+00
10	2.09E-02/1.56E-02(=)	6.22E-02/2.85E-02(-)	7.16E-02/3.90E-02(-)	6.29E-02/2.88E-02(-)	3.34E-02/2.77E-02(=)	3.16E-02/2.61E-02
11	5.40E-14/1.27E-14(-)	2.34E+01/2.20E+01(-)	0.00E+00/0.00E+00(=)	1.49E-01/3.65E-01(-)	0.00E+00/0.00E+00(=)	0.00E+00/0.00E+00
12	5.27E+01/1.59E+01(=)	1.55E+02/3.01E+01(-)	4.32E+01/1.26E+01(+)	8.03E+01/2.02E+01(-)	5.54E+01/1.01E+01(=)	5.23E+01/8.16E+00
13	1.32E+02/3.64E+01(-)	2.81E+02/4.48E+01(-)	9.29E+01/2.70E+01(=)	1.62E+02/4.25E+01(-)	1.26E+02/3.09E+01(=)	1.05E+02/2.99E+01
14	4.93E+00/1.91E+00(-)	4.11E+02/6.69E+01(-)	2.43E+00/1.35E+00(-)	8.43E+02/1.50E+02(-)	1.43E+00/8.86E-01(-)	3.96E-02/2.33E-02
15	8.45E+03/5.53E+02(-)	1.29E+04/4.26E+02(-)	6.14E+03/8.79E+02(=)	1.08E+04/3.03E+03(-)	8.19E+03/6.24E+02(-)	6.03E+03/8.88E+02
16	3.11E+00/4.51E-01(=)	3.19E+00/2.78E-01(=)	2.77E+00/9.65E-01(=)	3.25E+00/3.02E-01(=)	3.18E+00/6.76E-01(=)	2.58E+00/1.20E+00
17	5.09E+01/3.21E-02(-)	7.80E+01/1.29E+01(-)	5.10E+01/6.88E-02(-)	9.77E+01/3.73E+00(-)	5.08E+01/1.02E-13(=)	5.08E+01/5.84E-14
18	1.75E+02/2.19E+01(-)	4.21E+02/3.12E+01(-)	7.94E+01/2.95E+01(+)	3.13E+02/1.22E+02(-)	1.73E+02/2.41E+01(-)	8.33E+01/5.91E+00
19	3.41E+00/4.53E-01(-)	1.10E+01/2.25E+00(-)	2.19E+00/2.97E-01(-)	1.02E+01/8.42E-01(-)	3.84E+00/3.40E-01(-)	1.86E+00/2.30E-01
20	1.95E+01/6.47E-01(-)	2.12E+01/4.40E-01(-)	1.89E+01/7.50E-01(=)	2.15E+01/3.32E-01(-)	1.99E+01/4.88E-01(-)	1.88E+01/7.79E-01
21	7.46E+02/3.86E+02(=)	7.46E+02/3.86E+02(=)	7.10E+02/4.39E+02(=)	6.64E+02/4.42E+02(=)	8.03E+02/4.17E+02(=)	7.88E+02/4.10E+02
22	2.02E+01/4.13E+00(-)	1.64E+03/1.01E+03(-)	2.07E+01/2.95E+00(-)	1.06E+03/5.81E+02(-)	4.25E+01/5.75E+01(-)	1.14E+01/6.01E-01
23	8.51E+03/9.19E+02(-)	1.28E+04/4.89E+02(-)	5.93E+03/7.59E+02(=)	1.05E+04/2.99E+03(-)	8.55E+03/8.32E+02(-)	5.45E+03/8.82E+02
24	2.38E+02/5.49E+00(=)	2.89E+02/1.35E+01(-)	2.37E+02/9.82E+00(=)	2.42E+02/1.32E+01(=)	2.90E+02/5.53E+01(-)	2.46E+02/1.76E+01
25	3.06E+02/1.09E+01(+)	3.17E+02/2.03E+01(=)	3.27E+02/1.92E+01(-)	3.17E+02/1.41E+01(=)	3.70E+02/1.36E+01(-)	3.17E+02/1.32E+01
26	2.42E+02/6.65E+01(+)	3.36E+02/5.96E+01(=)	3.16E+02/6.91E+01(=)	2.63E+02/8.77E+01(+)	3.16E+02/8.80E+01(=)	2.87E+02/9.02E+01
27	8.35E+02/1.16E+02(+)	9.77E+02/1.67E+02(=)	8.69E+02/1.88E+02(+)	1.01E+03/1.41E+02(=)	1.29E+03/3.18E+02(-)	1.02E+03/1.89E+02
28	4.00E+02/1.75E-13(=)	5.59E+02/7.13E+02(=)	4.00E+02/1.75E-13(=)	4.00E+02/1.67E-13(=)	5.46E+02/6.53E+02(=)	5.47E+02/6.56E+02
+/=/-	4/12/12	0/10/18	3/15/10	2/11/15	2/13/13	-/-/-

TABLE 3. Comparison results of mean and standard deviation of 20-run fitness error of CIJADE with other state-of-the-art DE variants.

100D	jDE Mean/Std	JADE Mean/Std	SaDE Mean/Std	CoDE Mean/Std	Rcr-JADE Mean/Std	CobiDE Mean/Std
1	2.27E-13/0.00E+00(+)	4.09E-13/1.19E-13(=)	0.00E+00/0.00E+00(+)	0.00E+00/0.00E+00(+)	3.87E-13/1.07E-13(=)	1.71E-13/1.01E-13(+)
2	1.72E+06/4.25E+05(-)	1.48E+05/4.71E+04(=)	6.13E+05/1.39E+05(-)	9.24E+05/2.67E+05(-)	1.75E+05/7.15E+04(=)	1.66E+06/4.28E+05(-)
3	1.44E+08/2.28E+08(+)	2.74E+08/2.30E+08(=)	2.61E+08/2.27E+08(=)	1.45E+08/1.12E+08(=)	2.36E+08/2.01E+08(=)	5.40E+07/4.61E+07(+)
4	3.79E+02/2.18E+02(-)	1.32E+04/3.24E+04(=)	1.54E+03/8.04E+02(-)	4.13E+00/6.05E+00(-)	4.10E-04/3.74E-04(=)	2.69E+00/1.57E+00(-)
5	9.15E-13/1.12E-12(=)	5.06E-13/1.01E-13(=)	8.53E-14/5.05E-14(+)	1.14E-13/0.00E+00(+)	4.83E-13/8.94E-14(=)	1.21E-12/2.19E-13(-)
6	2.16E+02/3.59E+01(-)	9.06E+01/7.36E+01(=)	9.53E+01/5.14E+01(=)	1.18E+02/5.47E+01(=)	1.09E+02/4.87E+01(=)	1.85E+02/2.41E+01(-)
7	5.12E+01/1.26E+01(=)	7.58E+01/2.19E+01(-)	7.15E+01/1.25E+01(-)	7.54E+01/1.54E+01(-)	7.00E+01/1.59E+01(-)	3.08E+01/8.86E+00(+)
8	2.13E+01/3.13E-02(-)	2.13E+01/2.71E-02(-)	2.13E+01/2.19E-02(-)	2.12E+01/5.88E-02(+)	2.13E+01/2.93E-02(-)	2.13E+01/2.68E-02(-)
9	1.29E+02/4.93E+00(-)	1.34E+02/3.06E+00(-)	9.00E+01/8.78E+00(+)	9.16E+01/1.10E+01(+)	1.33E+02/2.94E+00(-)	6.54E+01/6.39E+00(+)
10	1.44E-01/1.00E-01(-)	3.04E-02/1.73E-02(=)	1.19E-01/5.89E-02(-)	6.57E-02/2.54E-02(-)	2.50E-02/1.45E-02(=)	6.67E-02/2.94E-02(-)
11	5.68E-14/0.00E+00(=)	5.68E-14/0.00E+00(=)	1.54E+00/1.27E+00(-)	1.95E+01/4.69E+00(-)	5.68E-14/0.00E+00(=)	1.39E+02/1.17E+01(-)
12	2.32E+02/3.04E+01(-)	1.88E+02/1.75E+01(=)	3.05E+02/2.45E+01(-)	2.94E+02/3.54E+01(-)	1.90E+02/2.35E+01(=)	1.81E+02/3.46E+01(=)
13	4.35E+02/6.84E+01(-)	4.33E+02/2.66E+01(-)	5.56E+02/7.18E+01(-)	5.69E+02/7.26E+01(-)	4.30E+02/4.89E+01(-)	3.71E+02/4.59E+01(=)
14	1.04E-01/1.85E-01(+)	8.45E-02/1.36E-02(+)	2.13E+03/2.48E+02(-)	2.15E+02/1.44E+02(-)	8.88E-02/1.79E-02(+)	8.68E+03/7.21E+02(-)
15	2.06E+04/1.06E+03(-)	1.50E+04/6.88E+02(-)	2.61E+04/8.38E+02(-)	1.44E+04/1.27E+03(-)	1.51E+04/6.13E+02(-)	1.27E+04/1.35E+03(=)
16	3.65E+00/4.10E-01(-)	1.96E+00/4.48E-01(=)	3.68E+00/2.49E-01(-)	1.69E+00/6.09E-01(=)	2.11E+00/3.12E-01(=)	9.43E-01/1.47E+00(+)
17	1.02E+02/1.18E-13(=)	1.02E+02/8.75E-14(=)	1.99E+02/7.46E+00(-)	1.15E+02/2.67E+00(-)	1.02E+02/9.49E-14(=)	3.97E+02/1.28E+01(-)
18	5.55E+02/3.72E+01(-)	3.65E+02/2.56E+01(-)	8.10E+02/3.55E+01(-)	3.43E+02/3.31E+01(-)	3.74E+02/2.53E+01(-)	2.31E+02/2.73E+01(=)
19	5.94E+00/5.22E-01(+)	9.02E+00/1.15E+00(-)	4.38E+01/4.28E+00(-)	9.79E+00/1.30E+00(-)	8.98E+00/1.25E+00(-)	3.28E+01/4.52E+00(-)
20	5.00E+01/2.55E-14(-)	5.00E+01/2.10E-11(-)	4.99E+01/2.18E-01(=)	4.99E+01/2.44E-01(=)	5.00E+01/1.39E-09(-)	5.00E+01/8.61E-12(-)
21	3.30E+02/4.70E+01(+)	3.85E+02/3.66E+01(=)	4.00E+02/2.62E-13(=)	3.60E+02/5.03E+01(=)	3.80E+02/4.10E+01(=)	3.65E+02/4.89E+01(=)
22	1.48E+02/7.21E+01(-)	6.70E+01/6.08E+01(=)	3.09E+03/2.04E+03(-)	2.43E+02/1.41E+02(-)	5.74E+01/4.99E+01(=)	1.05E+04/1.26E+03(-)
23	2.14E+04/1.25E+03(-)	1.68E+04/1.37E+03(-)	2.73E+04/1.43E+03(-)	1.65E+04/1.54E+03(-)	1.67E+04/1.21E+03(-)	1.40E+04/1.18E+03(=)
24	3.21E+02/2.20E+01(=)	3.43E+02/2.42E+01(-)	3.61E+02/1.47E+01(-)	3.68E+02/2.05E+01(-)	3.35E+02/1.55E+01(-)	2.67E+02/1.58E+01(+)
25	4.53E+02/3.69E+01(+)	5.87E+02/3.74E+01(-)	5.19E+02/1.31E+01(+)	4.84E+02/2.03E+01(+)	5.85E+02/3.99E+01(-)	4.19E+02/1.83E+01(+)
26	5.99E+02/3.08E+01(-)	5.96E+02/4.92E+01(-)	4.63E+02/1.39E+01(+)	4.84E+02/2.43E+01(=)	6.08E+02/4.11E+01(-)	4.37E+02/2.53E+01(+)
27	2.85E+03/6.32E+02(-)	2.54E+03/8.29E+02(=)	2.02E+03/2.28E+02(=)	2.34E+03/2.38E+02(=)	3.03E+03/5.83E+02(-)	1.50E+03/3.06E+02(+)
28	3.87E+03/1.11E+03(=)	3.42E+03/1.05E+03(-)	3.00E+03/1.46E+02(-)	3.20E+03/9.50E+02(=)	3.40E+03/1.07E+03(=)	2.73E+03/6.71E+02(+)
+/=/-	6/6/16	1/14/13	5/5/18	5/8/15	1/14/13	10/6/12

100D	MPEDE Mean/Std	AGDE Mean/Std	CIPDE Mean/Std	EFADE Mean/Std	EDEV Mean/Std	CIJADE Mean/Std
1	3.07E-13/1.11E-13(+)	3.18E-13/1.14E-13(=)	4.55E-13/1.28E-13(=)	1.36E-13/1.14E-13(+)	0.00E+00/0.00E+00(+)	3.87E-13/1.07E-13
2	2.11E+05/4.68E+04(-)	6.61E+05/1.46E+05(-)	2.70E+05/5.03E+04(-)	9.96E+05/1.87E+05(-)	2.39E+05/6.84E+04(-)	1.37E+05/4.34E+04
3	6.23E+07/5.90E+07(+)	2.94E+08/3.22E+08(=)	2.79E+08/2.30E+08(=)	6.12E+07/5.94E+07(+)	1.81E+08/1.84E+08(=)	1.93E+08/1.35E+08
4	1.22E+01/2.87E+01(-)	2.10E+01/1.51E+01(-)	1.74E+04/2.76E+04(-)	7.91E+01/5.71E+01(-)	4.10E+00/1.01E+01(-)	1.49E+04/2.68E+04
5	3.52E-13/7.28E-14(+)	3.18E-13/4.67E-14(+)	7.39E-13/1.59E-13(-)	1.14E-13/0.00E+00(+)	3.41E-14/5.35E-14(+)	4.66E-13/9.69E-14
6	8.27E+01/6.14E+01(=)	1.55E+02/4.09E+01(=)	8.83E+01/5.17E+01(=)	1.92E+02/2.50E+01(-)	8.84E+01/7.43E+01(=)	1.14E+02/8.17E+01
7	4.65E+01/1.00E+01(+)	1.01E+02/3.29E+01(-)	5.61E+01/1.43E+01(=)	3.81E+01/7.16E+00(+)	6.24E+01/1.30E+01(=)	5.39E+01/1.31E+01
8	2.13E+01/3.03E-02(-)	2.13E+01/2.42E-02(-)	2.13E+01/3.06E-02(-)	2.13E+01/2.55E-02(-)	2.13E+01/3.88E-02(-)	2.13E+01/4.10E-02
9	9.90E+01/8.49E+00(+)	1.18E+02/2.84E+01(-)	1.12E+02/6.94E+00(=)	9.09E+01/7.11E+00(+)	8.69E+01/1.11E+01(+)	1.09E+02/4.60E+00
10	3.31E-02/2.23E-02(=)	8.30E-02/6.65E-02(-)	7.07E-02/3.61E-02(-)	1.16E-01/4.41E-02(-)	2.62E-02/1.20E-02(=)	2.77E-02/1.95E-02
11	1.44E+00/2.00E+00(-)	3.56E+02/1.86E+02(-)	7.39E-14/2.67E-14(-)	5.22E+00/1.91E+00(-)	0.00E+00/0.00E+00(+)	5.68E-14/0.00E+00
12	2.09E+02/3.28E+01(-)	6.35E+02/1.18E+02(-)	1.73E+02/2.78E+01(=)	1.64E+02/2.01E+01(=)	2.01E+02/2.27E+01(-)	1.73E+02/2.43E+01
13	4.15E+02/6.37E+01(-)	9.01E+02/1.04E+02(-)	3.62E+02/4.25E+01(=)	3.41E+02/5.54E+01(=)	4.39E+02/5.29E+01(-)	3.64E+02/5.34E+01
14	8.27E-01/4.79E-01(-)	6.83E+03/5.92E+02(-)	7.89E+00/2.23E+00(-)	8.05E+03/4.01E+02(-)	1.41E+01/5.65E+00(-)	3.34E-01/5.42E-02
15	1.41E+04/1.49E+03(-)	2.98E+04/4.42E+02(-)	1.29E+04/1.23E+03(=)	2.43E+04/7.68E+03(-)	1.39E+04/8.77E+02(-)	1.30E+04/1.11E+03
16	3.62E+00/6.29E-01(-)	4.00E+00/2.33E-01(-)	3.33E+00/1.09E+00(-)	3.95E+00/2.78E-01(-)	1.46E+00/5.07E-01(=)	2.15E+00/1.42E+00
17	1.02E+02/2.44E-02(-)	3.59E+02/4.95E+01(-)	1.02E+02/1.47E-01(-)	3.54E+02/1.04E+01(-)	1.02E+02/4.15E-02(-)	1.02E+02/2.51E-13
18	2.51E+02/3.10E+01(-)	1.18E+03/1.01E+02(-)	2.13E+02/2.11E+01(=)	6.54E+02/3.01E+02(-)	2.68E+02/2.35E+01(-)	2.23E+02/1.59E+01
19	7.35E+00/9.25E-01(=)	1.07E+02/9.33E+01(-)	9.94E+00/1.55E+00(-)	3.82E+01/1.50E+00(-)	1.29E+01/1.41E+00(-)	7.44E+00/1.20E+00
20	5.00E+01/1.56E-01(=)	5.00E+01/1.97E-10(-)	4.96E+01/4.47E-01(+)	5.00E+01/5.09E-14(-)	5.00E+01/1.52E-01(=)	4.99E+01/1.82E-01
21	3.95E+02/2.24E+01(=)	3.85E+02/3.66E+01(=)	3.65E+02/4.89E+01(=)	3.65E+02/4.89E+01(=)	3.90E+02/3.08E+01(=)	3.85E+02/3.66E+01
22	5.05E+01/4.42E+01(-)	1.04E+04/1.38E+03(-)	4.88E+01/3.73E+01(-)	8.54E+03/1.81E+03(-)	1.32E+02/9.62E+01(-)	4.85E+01/4.81E+01
23	1.49E+04/1.35E+03(-)	3.06E+04/4.50E+02(-)	1.51E+04/1.85E+03(-)	2.55E+04/6.95E+03(-)	1.50E+04/1.12E+03(-)	1.38E+04/1.45E+03
24	3.36E+02/1.70E+01(-)	4.57E+02/3.18E+01(-)	3.20E+02/1.41E+01(=)	2.95E+02/1.38E+01(+)	4.51E+02/1.12E+02(-)	3.25E+02/1.51E+01
25	4.80E+02/2.45E+01(+)	5.06E+02/5.61E+01(=)	5.05E+02/4.11E+01(+)	4.93E+02/1.92E+01(+)	5.71E+02/2.58E+01(-)	5.29E+02/2.77E+01
26	4.46E+02/1.92E+01(+)	4.66E+02/1.59E+01(+)	4.58E+02/3.90E+01(+)	4.76E+02/2.50E+01(+)	4.79E+02/3.68E+01(=)	5.06E+02/5.02E+01
27	1.88E+03/2.84E+02(+)	2.43E+03/3.78E+02(=)	1.84E+03/1.80E+02(+)	1.89E+03/2.80E+02(=)	3.08E+03/8.78E+02(-)	2.26E+03/5.92E+02
28	3.33E+03/1.01E+03(=)	6.87E+03/2.53E+03(-)	3.64E+03/1.12E+03(-)	3.20E+03/1.01E+03(+)	3.49E+03/1.07E+03(-)	3.03E+03/8.21E+02
+/=/-	8/6/14	2/6/20	4/11/13	9/4/15	4/8/16	-/-/-

that the performance achieved by two algorithms is similar. All the experiments were performed on a computer with Intel(R) Core(TM) i5 3.3 GHz dual-core CPU, 8.0 GB RAM and Windows 7 Operating System. All the algorithms were coded using Matlab 2016a.

A. COMPARISON OF CIJADE WITH STATE-OF-THE-ART DE VARIANTS

In this subsection, the proposed CIJADE is contrasted with several state-of-the-art DE variants including jDE [34], JADE [17], SaDE [13], CoDE [19], Rcr-JADE [32], CobiDE [35], MPEDE [21], AGDE [36], CIPDE [18], EFADE [37] and EDEV [4] on F1-F28 benchmark functions with 50D and 100D. These DE variants are selected due to their popularity and competitive performance. SaDE, CoDE, MPEDE, and EDEV are DE variants employing multi-mutation strategies or multi-DE variants. The parameters configurations of these comparative algorithms are listed in Table 1 according to the original papers' recommended values.

1) COMPUTATIONAL RESULTS OF BENCHMARK FUNCTIONS WITH 50D

The experimental results of all the benchmark functions with 50D are reported in Table 2. From Table 2, we can find that CIJADE performs the best among the twelve state-of-the-art DE variants on all the 28 benchmark functions with 50D. Comparing with the jDE algorithm, the CIJADE achieves 18 better performances, 9 similar performances, and 1 worse performance out of the total 28 functions. Comparing with the JADE algorithm, the CIJADE achieves 11 better performances, 16 similar performances, and 1 worse performance out of the total 28 functions. Comparing with the SaDE algorithm, the CIJADE achieves 16 better performances, 9 similar performances, and 3 worse performances out of the total 28 functions. Comparing with the CoDE algorithm, the CIJADE achieves 17 better performances, 6 similar performances, and 5 worse performances out of the total 28 functions. Comparing with the Rcr-JADE algorithm, the CIJADE achieves 13 better performances, 15 similar performances, and 0 worse performance out of the total 28 functions. Comparing with the CobiDE algorithm, the CIJADE achieves 13 better performances, 8 similar performances, and 7 worse performances out of the total 28 functions. Comparing with the MPEDE algorithm, the CIJADE achieves 12 better performances, 12 similar performances, and 4 worse performances out of the total 28 functions. Comparing with the AGDE algorithm, the CIJADE achieves 18 better performances, 10 similar performances, and 0 worse performance out of the total 28 functions. Comparing with the CIPDE algorithm, the CIJADE achieves 10 better performances, 15 similar performances, and 3 worse performances out of the total 28 functions. Comparing with the EFADE algorithm, the CIJADE achieves 15 better performances, 11 similar performances, and 2 worse performances out of the total 28 functions. Comparing with the EDEV algorithm, the CIJADE achieves 13 better performances, 13 similar performances, and 2 worse performances out of the total 28 functions. Overall, CIJADE achieves better performance than the other eleven DE variants in dealing with CEC 2013 benchmark functions with 50D. This is because the CIJADE can make full use of the advantages of CIPDE and JADE algorithms.

2) COMPUTATIONAL RESULTS OF BENCHMARK FUNCTIONS WITH 100D

Table 3 gives the experimental results for all comparative algorithms on each benchmark function with 100D. From Table 3, it is find that CIJADE perform the best among the twelve state-of-the-art DE variants on all the 28 benchmark functions with 100D. Comparing with the jDE algorithm, the CIJADE achieves 16 better performances, 6 similar performances, and 6 worse performance out of the total 28 functions. Comparing with the JADE algorithm, the CIJADE achieves 13 better performances, 14 similar performances, and 1 worse performance out of the total 28 functions. Comparing with the SaDE algorithm, the CIJADE achieves 18 better performances, 5 similar performances, and 5 worse performances out of the total 28 functions. Comparing with the CoDE algorithm, the CIJADE achieves 15 better performances, 8 similar performances, and 5 worse performances out of the total 28 functions. Comparing with the Rcr-JADE algorithm, the CIJADE achieves 13 better performances, 14 similar performances, and 1 worse performance out of the total 28 functions. Comparing with the CobiDE algorithm, the CIJADE achieves 12 better performances, 6 similar performances, and 10 worse performances out of the total 28 functions. Comparing with the MPEDE algorithm, the CIJADE achieves 14 better performances, 6 similar performances, and 8 worse performances out of the total 28 functions. Comparing with the AGDE algorithm, the CIJADE achieves 20 better performances, 6 similar performances, and 2 worse performances out of the total 28 functions. Comparing with the CIPDE algorithm, the CIJADE achieves 13 better performances, 11 similar performances, and 4 worse performances out of the total 28 functions. Comparing with the EFADE algorithm, the CIJADE achieves 15 better performances, 4 similar performances, and 9 worse performances out of the total 28 functions. Comparing with the EDEV algorithm, the CIJADE achieves 16 better performances, 8 similar performances, and 4 worse performances out of the total 28 functions. Overall, our CIJADE also achieves better performance than the other eleven DE variants in dealing with CEC 2013 benchmark functions with 100D. It also can find that when the dimension of the problem is increased from 50 to 100, the performance of CIJADE will not degrade too much.

3) CONVERGENCE CURVES FOR ALL COMPARATIVE ALGORITHMS

The comparisons of convergence curves are given in Figs. 2-5 for algorithm evaluation. The median values of the 20-run

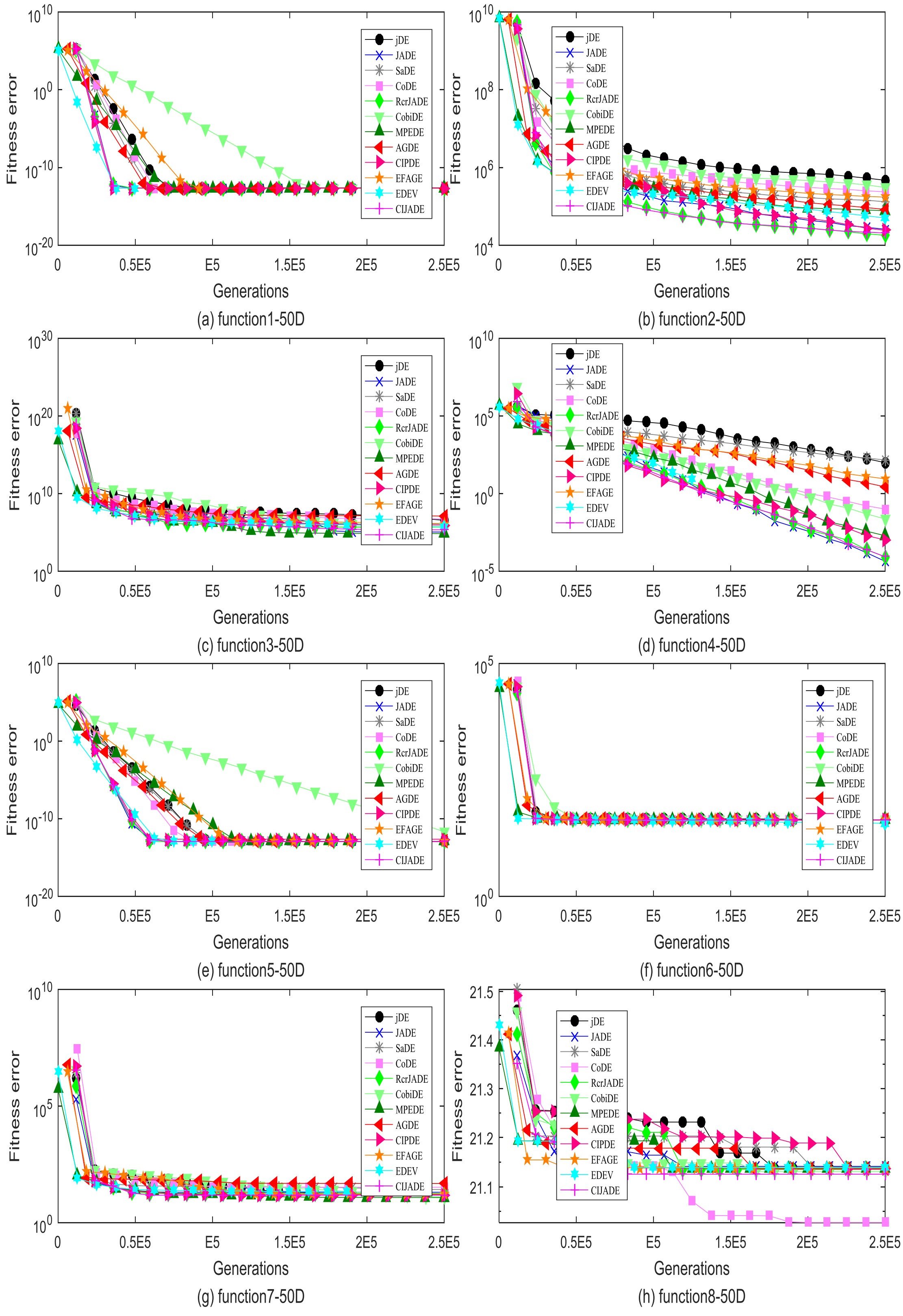


FIGURE 2. Convergence comparison on F1-F8.

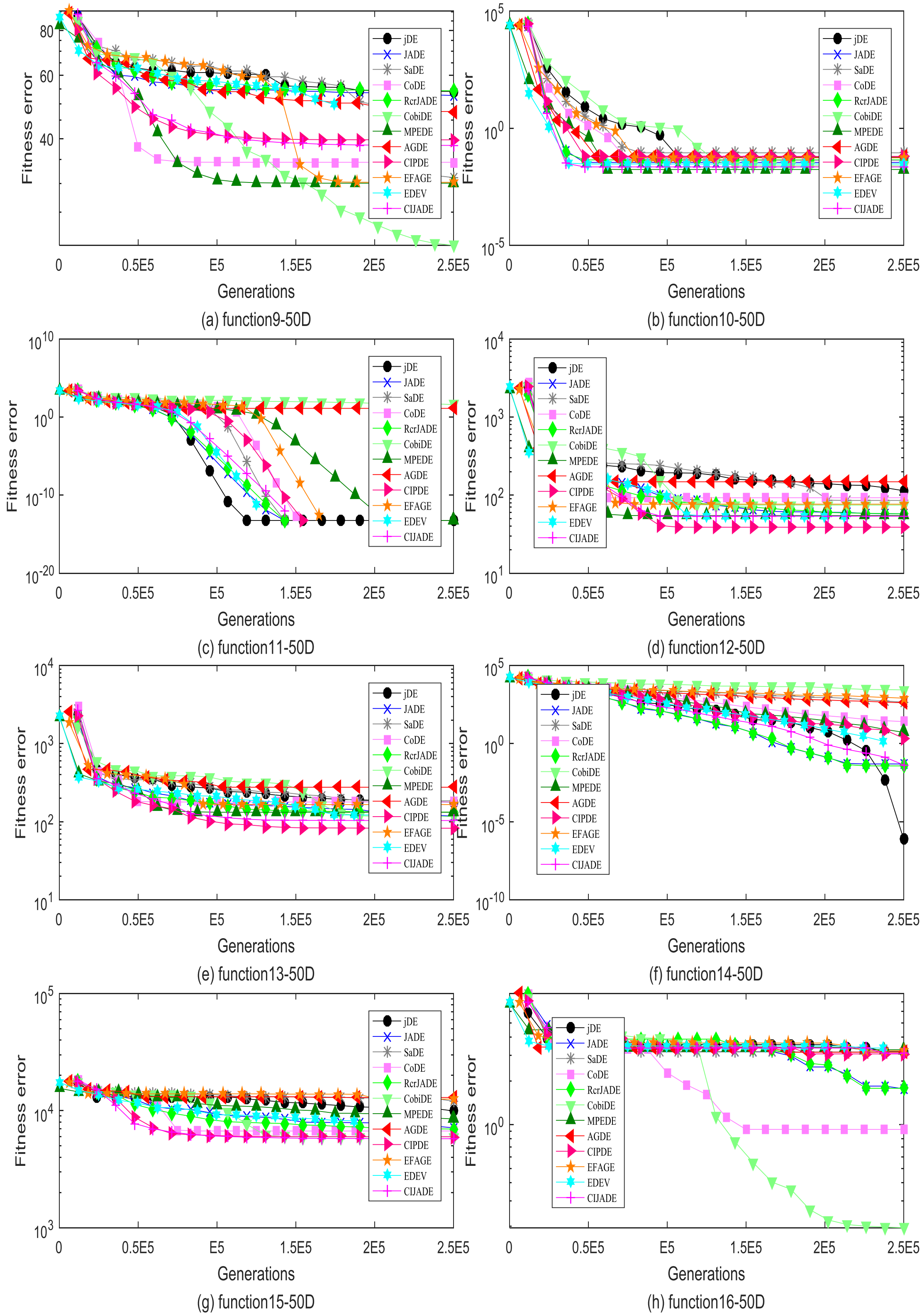


FIGURE 3. **Convergence comparison on F9-F16.**

fitness errors achieved by these comparative algorithms on each benchmark function with 50D are selected for this comparison. Fig. 2 gives the first part including 8 figures, F1-F8, of the total 28-figure comparison, Fig. 3 gives the second part including 8 figures, F9-F16, of the total 28-figure comparison, Fig. 4 gives the third part including 8 figures,

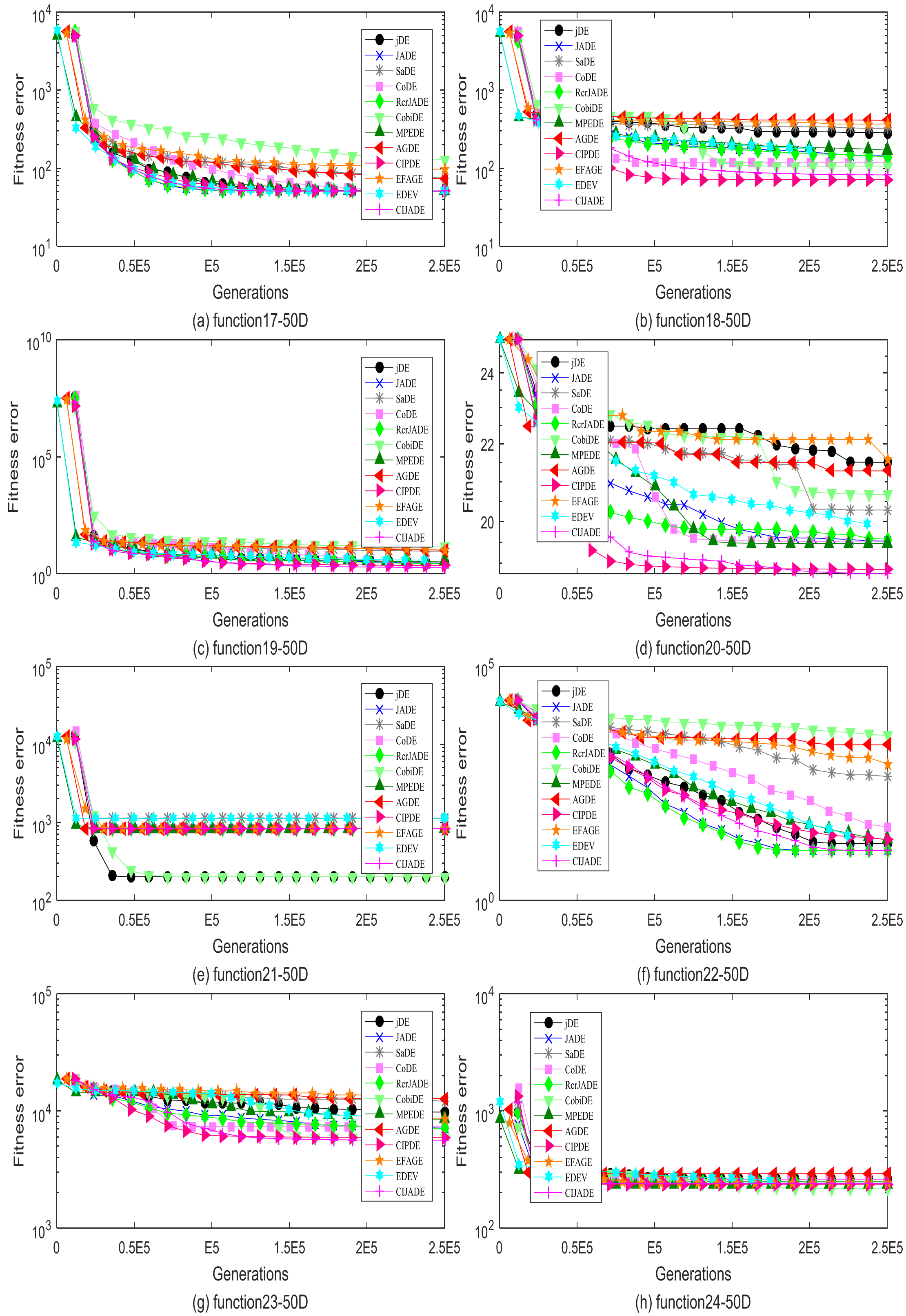


FIGURE 4. Convergence comparison on F17-F24.

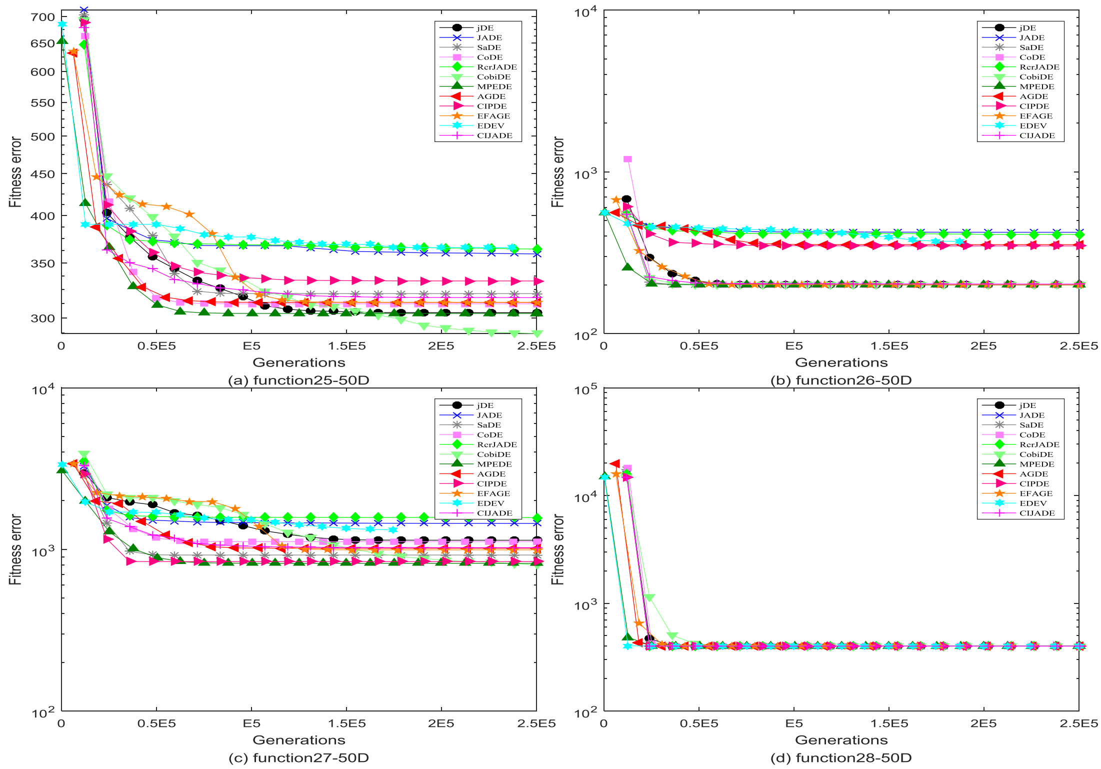


FIGURE 5. Convergence comparison on F25-F28.

F17-F24, of the total 28-figure comparison, and Fig. 5 gives the last part 4 figures from F25 to F28. From all these figures, it can be seen that for convergence speed our proposed CIJADE algorithm outperforms jDE on F2-F4, F8-F10, F13, F15, F19, F20, F22-F24, and F27. It also outperforms JADE on F2, F8-F10, F13, F15, F17, F19-F20, F23, and F25-F27. It also outperforms SaDE on F2-F4, F7-F8, F10, F13-F17, F19-F23, and F26. It also outperforms CoDE on F2-F4, F7, F10- F11, F13-F15, F17, F19-F20, F22-F24, and F27. It also outperforms Rcr-JADE on F3, F8-F10, F13, F15, F17, F19, F20, F23, and F25-F27,. It also outperforms CobiDE on F2-F4, F5, F7-F8, F10-F11, F13-F14, F16-F17, F19-F20, and F22-F23. It also outperforms MPEDE on F2, F4, F8, F11, F13, F14-F15, F17, F19-F20, and F22-F23. It also outperforms AGDE on F2-F4, F7, F8, F9, F10-F20, F22-F23, and F26. It also outperforms CIPDE on F2-F4, F8-F10, F14-F15, F19-F20, F22-F23, and F25-F26. It also outperforms EFADE on F2-F4, F8, F10-F20, and F22-F23. It also outperforms EDEV on F2-F4, F8-F11, F13-F15, F17, F19-F23, and F25-F27. In conclusion, our CIJADE is competitive with the other eleven state-of-the-art DE variants in terms of convergence speed.

B. EFFECTIVENESS OF MCIPDE AND MJADE

In this subsection, in order to assess the effects of MCIPDE and MJADE, we compare the modified algorithms with original algorithms CIPDE and JADE, respectively. The parameter settings of these comparative algorithms are given in Table 1. Table 4 reports the mean and the standard deviation of 20 independent runs of function error values on CEC2013 benchmark functions with 50D.

As shown in Table 4, comparing with the CIPDE algorithm, the MCIPDE achieves 6 better performances, 19 similar performances, and 3 worse performances out of the total 28 benchmark functions from the "Mean/Std" perspective of view. Comparing with the JADE algorithm, the MJADE achieves 15 better performances, 12 similar performances, and 1 worse performance out of the total 28 benchmark functions from the "Mean/Std" perspective of view. These results indicate that these modifications on two original algorithms are effective.

C. ANALYSIS OF THE PARAMETERS λ

In CIJADE, the parameter λ determine the population size of the $pop_{\sup erior}$, in which the individuals evolve using the operation of MCIPDE. When analyzing the impact of λ, $p_{\max}$, $p_{\min}$ and T are fixed to 0.2, 0.1 and 90 respectively. We conduct the proposed CIJADE with different λ values. The other parameter configurations of CIJADE are the same as subsection 4.1. The Mean/Std of function error values of 20 independent runs on CEC2013 functions with 50D are reported in Table 5. From the Table 5, we can see that the performance of CIJADE is not very sensitive to the setting of λ for most of the functions. When $\lambda = 0.2$ or $\lambda = 0.1$, the CIJADE has relative better results, in our study we use the 0.2 as the recommended value.

TABLE 4. Comparison results of the MCIPDE with CIPDE and MJADE with JADE.

50D	MCIPDE Mean/Std	CIPDE Mean/Std	MJADE Mean/Std	JADE Mean/Std
1	0.0000E+00/0.0000E+00	1.3642E-13/1.1428E-13(-)	1.2506E-13/1.1606E-13	1.8190E-13/9.3312E-14(=)
2	1.8628E+04/9.7158E+03	2.5410E+04/9.0917E+03(-)	2.3616E+04/1.4440E+04	2.9800E+04/1.5938E+04(=)
3	2.2625E+06/3.2149E+06	4.9540E+06/1.0637E+07(=)	1.7173E+06/2.5557E+06	1.6484E+06/3.8802E+06(=)
4	5.6877E+03/1.1844E+04	4.6132E+03/9.7023E+03(-)	1.2470E+04/1.7685E+04	1.0227E-04/1.2800E-04(+)
5	1.1369E-13/0.0000E+00	2.2169E-13/5.8028E-14(-)	1.1937E-13/2.5421E-14	1.3642E-13/4.6656E-14(=)
6	4.3447E+01/5.9493E-14	4.3731E+01/1.2701E+00(=)	4.3447E+01/6.8774E-14	4.3447E+01/8.1440E-14(=)
7	1.6512E+01/6.7137E+00	1.6584E+01/6.5520E+00(=)	1.3982E+01/8.2268E+00	2.2552E+01/1.1992E+01(-)
8	2.1141E+01/2.4723E-02	2.1136E+01/2.0505E-02(=)	2.1088E+01/1.1483E-01	2.1136E+01/3.8561E-02(=)
9	4.1986E+01/4.6343E+00	4.1172E+01/5.5385E+00(=)	3.7612E+01/5.5019E+00	5.3424E+01/2.6437E+00(-)
10	3.8051E-02/2.0594E-02	7.1564E-02/3.8991E-02(-)	2.5733E-02/2.0175E-02	4.4446E-02/2.7743E-02(-)
11	0.0000E+00/0.0000E+00	0.0000E+00/0.0000E+00(=)	0.0000E+00/0.0000E+00	2.8422E-15/1.2711E-14(=)
12	5.2584E+01/1.0194E+01	4.3231E+01/1.2571E+01(+)	4.5370E+01/8.7361E+00	5.8071E+01/7.2674E+00(-)
13	1.1017E+02/3.7218E+01	9.2896E+01/2.6959E+01(=)	9.7215E+01/2.5252E+01	1.2318E+02/1.9869E+01(-)
14	2.1965E+00/1.3203E+00	2.4289E+00/1.3516E+00(=)	2.3736E-02/1.3379E-02	4.9967E-02/2.1446E-02(-)
15	6.3140E+03/8.7612E+02	6.1447E+03/8.7943E+02(=)	5.9439E+03/9.5119E+02	7.0696E+03/4.9619E+02(-)
16	2.6939E+00/1.2181E+00	2.7741E+00/9.6472E-01(=)	2.4319E+00/1.3099E+00	2.1230E+00/6.8448E-01(=)
17	5.0992E+01/6.9808E-02	5.0963E+01/6.8849E-02(=)	5.0786E+01/3.6450E-14	5.0786E+01/3.9122E-14(=)
18	7.9597E+01/8.9418E+00	7.9366E+01/2.9459E+01(+)	8.5288E+01/1.1480E+01	1.4462E+02/8.9200E+00(-)
19	2.1848E+00/3.6476E-01	2.1906E+00/2.9652E-01(=)	1.9704E+00/2.7495E-01	2.7997E+00/2.8130E-01(-)
20	1.8678E+01/6.8626E-01	1.8925E+01/7.4989E-01(=)	1.8811E+01/9.8863E-01	1.9624E+01/7.9128E-01(-)
21	9.1242E+02/3.3083E+02	7.1045E+02/4.3877E+02(=)	7.5657E+02/4.3067E+02	8.3449E+02/3.9225E+02(=)
22	3.6376E+01/5.5295E+01	2.0732E+01/2.9472E+00(=)	1.1700E+01/6.9019E-01	1.2244E+01/4.3134E+00(=)
23	6.0407E+03/1.1392E+03	5.9329E+03/7.5937E+02(=)	5.6996E+03/9.1576E+02	7.2355E+03/6.8723E+02(-)
24	2.4384E+02/1.1667E+01	2.3659E+02/9.8174E+00(+)	2.4288E+02/1.5196E+01	2.5738E+02/2.5387E+01(-)
25	3.0560E+02/1.6091E+01	3.2736E+02/1.9248E+01(-)	3.3125E+02/8.1396E+00	3.5376E+02/2.1095E+01(-)
26	3.1322E+02/6.9762E+01	3.1571E+02/6.9149E+01(=)	2.8046E+02/9.1214E+01	3.7764E+02/9.1331E+01(-)
27	9.2316E+02/1.9397E+02	8.6873E+02/1.8770E+02(=)	1.0883E+03/2.2017E+02	1.4175E+03/2.7907E+02(-)
28	5.4869E+02/6.6497E+02	4.0000E+02/1.7496E-13(=)	4.0000E+02/1.7496E-13	6.9451E+02/9.0650E+02(=)
+/=/-	-/-/-	3/19/6	-/-/-	1/12/15

D. ANALYSIS OF THE PARAMETERS p_{max}, p_{min}

In CIJADE, the parameters p_{max} and p_{min} affect the greediness of the "DE/target-to-pbest/1" mutation scheme. To study the effects of these two parameters on the performance of CIJADE, the proposed CIJADE with different p_{max} and p_{min} values are conducted. When analyzing the impact of p_{max} and p_{min}, λ and T are fixed to 0.2, and 90 respectively. The other parameter configurations of CIJADE are the same as subsection 4.1. The Mean/Std of function error values of 20 independent runs on CEC2013 functions with 50D are reported in Table 6.

From Table 6, we can see that when the CIJADE algorithm with the combination of $p_{max} = 0.1$ and $p_{min} = 0.05$, it has relatively poor performance. When the CIJADE algorithm with the combination of $p_{max} = 0.5$ and $p_{min} = 0.4$, and the combination of $p_{max} = 0.4$ and $p_{min} = 0.3$, it also has poor performance.

When the CIJADE algorithm with the combination of $p_{max} = 0.3$ and $p_{min} = 0.2$, and the combination of $p_{max} = 0.2$ and $p_{min} = 0.1$, it has relatively better performance. From these comparison results, we chose the combination of $p_{max} = 0.2$ and $p_{min} = 0.1$ as the recommended values.

E. ANALYSIS OF THE PARAMETER T

In CIJADE, the parameter T is the stagnation threshold which determines the frequency of CIX and pBX are executed. To study the effects of this parameter on the performance of CIJADE, the proposed CIJADE with different T values are conducted. When analyzing the impact of T, p_{max}, p_{min}, and λ are fixed to 0.2, 0.1, and 0.2, respectively. The other parameter configurations of CIJADE are the same as subsection 4.1. The Mean/Std of function error values of 20 independent runs on CEC2013 functions with 50D are reported in Table 7.

From table 7, when T is set to 90, the CIJADE has better performance. If T value is smaller than or lager than 90, the CIJADE has relatively poor performance. The reason is that when T value is too small the CIX and pBX will execute frequently, which makes the algorithm too greedy and deteriorates CIJADE's exploration ability. On the contrary, when the T value is too large the CIX and pBX will be idle and their benefits are diminished. Therefore, from these comparison results, we chose the $T = 90$ as the recommended values.

V. APPLICATION OF THE CIJADE TO THE UCAV PATH PLANNING PROBLEM

We present the proposed CIJADE algorithm for the UCAV path planning problem in this section. In recent years, UCAV have become an important part of the civilian and military fields due to their outstanding abilities to work in remote and extremely dangerous environments [38], [39]. Path planning plays a vital role in the control of UCAV to accomplish the combat missions quickly and reliably, which aims to find an

TABLE 5. Comparison results of the CIJADE with different λ values.

50D	$\lambda = 0.05$ Mean/Std	$\lambda = 0.1$ Mean/Std	$\lambda = 0.2$ Mean/Std	$\lambda = 0.3$ Mean/Std	$\lambda = 0.4$ Mean/Std
1	1.36E-13/1.14E-13 (-)	6.82E-14/1.07E-13 (=)	3.41E-14/8.33E-14	3.41E-14/8.33E-14(=)	3.41E-14/8.33E-14(=)
2	2.54E+04/1.47E+04 (=)	2.06E+04/1.21E+04 (=)	2.57E+04/1.11E+04	2.05E+04/9.35E+03(=)	2.34E+04/1.52E+04(=)
3	1.72E+06/3.96E+06 (=)	3.36E+06/1.15E+07 (=)	1.09E+06/2.13E+06	1.87E+06/2.83E+06(=)	2.01E+06/4.21E+06(=)
4	9.27E+03/1.66E+04 (=)	1.28E+04/1.59E+04 (=)	9.14E+03/1.45E+04	1.06E+04/1.53E+04(=)	1.04E+04/1.32E+04(=)
5	1.25E-13/3.50E-14 (=)	1.14E-13/0.00E+00 (=)	1.25E-13/3.50E-14	1.19E-13/2.54E-14(=)	1.25E-13/3.50E-14(=)
6	4.34E+01/8.94E-14 (=)	4.34E+01/5.72E-14 (=)	4.34E+01/6.25E-14	4.13E+01/9.72E+00(=)	4.34E+01/7.80E-14(=)
7	1.64E+01/1.01E+01 (=)	1.57E+01/8.33E+00 (=)	1.69E+01/6.59E+00	1.48E+01/9.07E+00(=)	1.53E+01/6.55E+00(=)
8	2.11E+01/4.66E-02 (=)	2.11E+01/7.44E-02 (=)	2.11E+01/4.35E-02	2.11E+01/8.75E-02(=)	2.11E+01/8.10E-02(=)
9	3.82E+01/5.52E+00 (=)	3.87E+01/4.56E+00 (=)	3.80E+01/3.95E+00	4.03E+01/4.65E+00(=)	3.87E+01/4.87E+00(=)
10	2.87E-02/1.65E-02 (=)	2.83E-02/2.09E-02 (=)	3.16E-02/2.61E-02	3.32E-02/2.44E-02(=)	2.82E-02/1.83E-02(=)
11	0.00E+00/0.00E+00 (=)	0.00E+00/0.00E+00 (=)	0.00E+00/0.00E+00	0.00E+00/0.00E+00(=)	0.00E+00/0.00E+00(=)
12	5.29E+01/1.19E+01 (=)	5.16E+01/9.71E+00 (=)	5.23E+01/8.16E+00	4.76E+01/8.50E+00(+)	5.01E+01/9.48E+00(=)
13	9.94E+01/2.31E+01 (=)	9.29E+01/2.06E+01 (=)	1.05E+02/2.99E+01	9.42E+01/1.85E+01(=)	1.00E+02/1.84E+01(=)
14	2.62E-02/1.67E-02 (+)	2.77E-02/1.74E-02 (=)	3.96E-02/2.33E-02	1.25E-01/3.26E-02(-)	2.03E-01/3.87E-02(-)
15	5.90E+03/8.95E+02 (=)	5.64E+03/7.16E+02 (=)	6.03E+03/8.88E+02	6.03E+03/8.16E+02(=)	5.86E+03/1.08E+03(=)
16	2.06E+00/1.10E+00 (=)	2.66E+00/1.21E+00 (=)	2.58E+00/1.20E+00	2.21E+00/1.16E+00(=)	2.31E+00/1.38E+00(=)
17	5.08E+01/3.69E-14 (=)	5.08E+01/3.97E-14 (=)	5.08E+01/5.84E-14	5.08E+01/6.78E-14(=)	5.08E+01/1.60E-11(=)
18	8.32E+01/1.04E+01 (=)	8.63E+01/9.38E+00 (=)	8.33E+01/5.91E+00	8.40E+01/1.01E+01(=)	8.01E+01/6.20E+00(=)
19	1.94E+00/2.80E-01 (=)	1.98E+00/2.71E-01 (=)	1.86E+00/2.30E-01	2.00E+00/2.48E-01(=)	2.00E+00/3.35E-01(=)
20	1.85E+01/8.82E-01 (=)	1.90E+01/7.85E-01 (=)	1.88E+01/7.79E-01	1.82E+01/8.20E-01(+)	1.85E+01/7.64E-01(=)
21	8.81E+02/3.67E+02 (=)	7.88E+02/4.10E+02 (=)	7.88E+02/4.10E+02	8.49E+02/3.97E+02(=)	8.52E+02/3.58E+02(=)
22	1.19E+01/7.15E-01 (-)	1.17E+01/1.09E+00 (=)	1.14E+01/6.01E-01	1.14E+01/7.06E-01(-)	1.12E+01/9.10E-01(-)
23	5.79E+03/6.68E+02 (=)	5.72E+03/8.04E+02 (=)	5.45E+03/8.82E+02	5.75E+03/9.46E+02(=)	6.17E+03/1.17E+03(-)
24	2.35E+02/1.03E+01 (=)	2.46E+02/1.69E+01 (=)	2.46E+02/1.76E+01	2.44E+02/9.10E+00(=)	2.44E+02/1.09E+01(=)
25	3.34E+02/8.83E+00 (-)	3.25E+02/1.08E+01 (=)	3.17E+02/1.32E+01	3.30E+02/1.30E+01(-)	3.19E+02/1.27E+01(=)
26	2.80E+02/9.01E+01 (=)	2.93E+02/8.64E+01 (=)	2.87E+02/9.02E+01	2.80E+02/9.00E+01(=)	3.07E+02/7.27E+01(=)
27	1.11E+03/2.03E+02 (=)	1.06E+03/2.29E+02 (=)	1.02E+03/1.89E+02	1.04E+03/2.18E+02(=)	1.04E+03/2.25E+02(=)
28	5.45E+02/6.49E+02 (=)	4.00E+02/1.71E-13 (=)	5.47E+02/6.56E+02	4.00E+02/1.71E-13(=)	6.94E+02/9.04E+02(=)
+/=/-	1/24/3	0/28/0	-/-/-	2/23/3	0/25/3
50D	**$\lambda = 0.5$ Mean/Std**	**$\lambda = 0.6$ Mean/Std**	**$\lambda = 0.7$ Mean/Std**	**$\lambda = 0.8$ Mean/Std**	**$\lambda = 0.9$ Mean/Std**
1	0.00E+00/0.00E+00(=)	0.00E+00/0.00E+00(=)	1.14E-14/5.08E-14(=)	0.00E+00/0.00E+00(=)	1.14E-14/5.08E-14(=)
2	1.99E+04/1.08E+04(=)	2.26E+04/1.24E+04(=)	2.26E+04/1.10E+04(=)	1.82E+04/9.19E+03(+)	1.99E+04/8.94E+03(=)
3	3.48E+06/5.21E+06(=)	2.96E+06/4.98E+06(=)	2.02E+06/2.73E+06(=)	2.14E+06/2.83E+06(=)	3.47E+06/6.78E+06(-)
4	5.62E+03/1.24E+04(=)	6.35E+03/1.15E+04(=)	7.64E+03/1.28E+04(=)	9.24E+03/1.33E+04(=)	1.17E+04/1.55E+04(=)
5	1.14E-13/0.00E+00(=)	1.14E-13/0.00E+00(=)	1.25E-13/3.50E-14(=)	1.25E-13/3.50E-14(=)	1.14E-13/0.00E+00(=)
6	4.37E+01/1.27E+00(=)	4.40E+01/1.75E+00(=)	4.34E+01/8.63E-14(=)	4.40E+01/1.75E+00(=)	4.34E+01/6.88E-14(=)
7	1.71E+01/5.33E+00(=)	1.81E+01/7.22E+00(=)	1.52E+01/5.05E+00(=)	1.51E+01/6.61E+00(=)	1.53E+01/8.23E+00(=)
8	2.11E+01/6.94E-02(=)	2.11E+01/9.33E-02(=)	2.11E+01/4.26E-02(=)	2.11E+01/5.77E-02(=)	2.11E+01/4.15E-02(=)
9	3.94E+01/5.75E+00(=)	4.04E+01/3.70E+00(=)	3.91E+01/5.15E+00(=)	4.15E+01/6.40E+00(-)	4.03E+01/4.40E+00(=)
10	2.79E-02/1.87E-02(=)	3.90E-02/2.41E-02(=)	3.76E-02/2.61E-02(=)	3.89E-02/5.46E-02(=)	3.69E-02/3.76E-02(=)
11	0.00E+00/0.00E+00(=)	0.00E+00/0.00E+00(=)	0.00E+00/0.00E+00(=)	0.00E+00/0.00E+00(=)	0.00E+00/0.00E+00(=)
12	4.94E+01/1.07E+01(=)	5.36E+01/1.16E+01(=)	5.47E+01/1.10E+01(=)	4.96E+01/1.06E+01(=)	5.45E+01/1.22E+01(=)
13	1.08E+02/3.32E+01(=)	1.08E+02/2.81E+01(=)	1.04E+02/2.99E+01(=)	1.07E+02/3.59E+01(=)	1.06E+02/2.70E+01(=)
14	2.78E-01/7.63E-02(-)	3.65E-01/1.06E-01(-)	6.62E-01/4.60E-01(-)	9.07E-01/5.50E-01(-)	1.56E+00/1.26E+00(-)
15	6.20E+03/7.30E+02(=)	6.15E+03/8.57E+02(=)	6.14E+03/7.19E+02(=)	6.10E+03/6.60E+02(=)	5.98E+03/7.83E+02(=)
16	2.82E+00/9.43E-01(=)	2.44E+00/1.16E+00(=)	2.21E+00/1.11E+00(=)	2.80E+00/1.05E+00(=)	2.57E+00/1.12E+00(=)
17	5.08E+01/1.82E-07(=)	5.08E+01/3.08E-03(-)	5.08E+01/2.24E-02(-)	5.09E+01/4.51E-02(-)	5.09E+01/4.70E-02(-)
18	7.93E+01/9.07E+00(+)	7.88E+01/8.79E+00(+)	7.93E+01/7.18E+00(+)	7.70E+01/5.67E+00(+)	7.67E+01/6.48E+00(+)
19	2.06E+00/2.61E-01(-)	2.06E+00/2.85E-01(-)	2.03E+00/2.73E-01(=)	2.01E+00/2.60E-01(=)	2.05E+00/3.57E-01(-)
20	1.85E+01/1.00E+00(=)	1.87E+01/7.79E-01(=)	1.84E+01/1.02E+00(=)	1.88E+01/9.64E-01(=)	1.85E+01/8.79E-01(=)
21	9.12E+02/3.31E+02(=)	7.92E+02/3.73E+02(=)	8.03E+02/4.17E+02(=)	8.20E+02/3.86E+02(=)	9.41E+02/3.36E+02(=)
22	1.19E+01/4.10E-01(-)	1.39E+01/2.46E+00(-)	2.55E+01/4.77E+01(-)	3.52E+01/5.59E+01(-)	2.59E+01/3.26E+01(-)
23	5.91E+03/8.01E+02(=)	5.70E+03/1.03E+03(=)	5.83E+03/8.83E+02(=)	6.07E+03/8.81E+02(-)	5.66E+03/1.00E+03(=)
24	2.44E+02/1.19E+01(=)	2.42E+02/1.19E+01(=)	2.42E+02/1.36E+01(=)	2.46E+02/9.64E+00(=)	2.48E+02/1.46E+01(=)
25	3.21E+02/1.73E+01(=)	3.18E+02/1.94E+01(=)	3.17E+02/1.96E+01(=)	3.14E+02/2.00E+01(=)	3.15E+02/2.35E+01(=)
26	2.94E+02/8.63E+01(=)	3.07E+02/8.15E+01(=)	2.73E+02/7.60E+01(=)	2.54E+02/7.50E+01(=)	3.09E+02/6.61E+01(=)
27	1.04E+03/1.87E+02(=)	1.02E+03/2.25E+02(=)	1.02E+03/2.12E+02(=)	8.96E+02/2.06E+02(+)	9.84E+02/2.09E+02(=)
28	5.50E+02/6.72E+02(=)	5.47E+02/6.58E+02(=)	5.47E+02/6.58E+02(=)	6.96E+02/9.12E+02(=)	5.49E+02/6.67E+02(=)
+/=/-	1/24/3	1/23/4	1/24/3	3/20/5	1/22/5

optimal or near-optimal path between the starting point and the desired destination under the artificial threats and some specific constraints. In general, the methodology of path planning is to find a path which minimizes the flight distance, fuel consumption, and exposure to threat sources. Path planning of UCAV is considered as an NP-complete optimization

TABLE 6. Comparison results of the CIJADE with different $p_{\max}$, $p_{\min}$ values.

50D	pmax=0.1,pmin=0.05 Mean/Std	pmax=0.2,pmin=0.1 Mean/Std	pmax=0.3,pmin=0.2 Mean/Std	pmax=0.4,pmin=0.3 Mean/Std	pmax=0.5,pmin=0.4 Mean/Std
1	1.02E-13/1.16E-13(-)	3.41E-14/8.33E-14	6.82E-14/1.07E-13(=)	6.82E-14/1.07E-13(=)	1.02E-13/1.16E-13(-)
2	2.30E+04/1.07E+04(=)	2.57E+04/1.11E+04	2.83E+04/2.79E+04(=)	2.72E+04/1.39E+04(=)	2.70E+04/1.27E+04(=)
3	1.22E+06/2.16E+06(=)	1.09E+06/2.13E+06	7.80E+05/1.21E+06(=)	1.11E+06/3.50E+06(=)	2.01E+06/3.70E+06(=)
4	7.85E+03/1.32E+04(=)	9.14E+03/1.45E+04	8.61E+03/1.22E+04(=)	1.27E+04/1.72E+04(-)	7.25E+03/1.50E+04(-)
5	1.36E-13/4.67E-14(=)	1.25E-13/3.50E-14	1.14E-13/0.00E+00(=)	1.19E-13/2.54E-14(=)	1.14E-13/0.00E+00(=)
6	4.37E+01/1.27E+00(=)	4.34E+01/6.25E-14	4.34E+01/6.68E-14(=)	4.34E+01/6.81E-14(=)	4.34E+01/5.72E-14(=)
7	1.65E+01/7.50E+00(=)	1.69E+01/6.59E+00	1.69E+01/1.05E+01(=)	1.65E+01/8.00E+00(=)	1.74E+01/6.85E+00(=)
8	2.11E+01/5.55E-02(=)	2.11E+01/4.35E-02	2.11E+01/1.16E-01(=)	2.11E+01/1.07E-01(=)	2.11E+01/8.96E-02(=)
9	3.97E+01/3.99E+00(=)	3.80E+01/3.95E+00	3.71E+01/4.86E+00(=)	3.70E+01/4.24E+00(=)	3.83E+01/3.83E+00(=)
10	3.19E-02/2.07E-02(=)	3.16E-02/2.61E-02	2.56E-02/1.94E-02(=)	2.80E-02/2.08E-02(=)	2.76E-02/2.24E-02(=)
11	0.00E+00/0.00E+00(=)	0.00E+00/0.00E+00	0.00E+00/0.00E+00(=)	0.00E+00/0.00E+00(=)	0.00E+00/0.00E+00(=)
12	5.53E+01/1.08E+01(=)	5.23E+01/8.16E+00	4.77E+01/9.00E+00(+)	5.31E+01/1.25E+01(=)	4.88E+01/1.23E+01(=)
13	1.19E+02/2.90E+01(=)	1.05E+02/2.99E+01	1.06E+02/2.88E+01(=)	9.17E+01/1.68E+01(=)	1.13E+02/2.37E+01(=)
14	3.01E-02/1.73E-02(=)	3.96E-02/2.33E-02	5.27E-02/2.07E-02(=)	8.12E-02/2.85E-02(-)	8.18E-02/2.34E-02(-)
15	6.24E+03/9.21E+02(=)	6.03E+03/8.88E+02	5.81E+03/8.90E+02(=)	5.64E+03/6.94E+02(=)	5.50E+03/5.69E+02(+)
16	1.96E+00/1.29E+00(=)	2.58E+00/1.20E+00	2.13E+00/1.15E+00(=)	2.16E+00/1.28E+00(=)	2.56E+00/1.28E+00(=)
17	5.08E+01/6.19E-14(=)	5.08E+01/5.84E-14	5.08E+01/5.05E-14(=)	5.08E+01/5.49E-14(=)	5.08E+01/6.50E-14(=)
18	8.64E+01/8.76E+00(=)	8.33E+01/5.91E+00	8.45E+01/6.50E+00(=)	8.03E+01/7.83E+00(=)	8.28E+01/8.21E+00(=)
19	1.92E+00/3.24E-01(=)	1.86E+00/2.30E-01	1.89E+00/2.19E-01(=)	1.88E+00/2.59E-01(=)	1.89E+00/2.33E-01(=)
20	1.90E+01/8.26E-01(=)	1.88E+01/7.79E-01	1.89E+01/7.63E-01(=)	1.86E+01/8.95E-01(=)	1.85E+01/9.89E-01(=)
21	7.28E+02/4.14E+02(=)	7.88E+02/4.10E+02	9.44E+02/2.86E+02(=)	9.09E+02/3.74E+02(=)	8.20E+02/3.86E+02(=)
22	2.23E+01/4.75E+01(-)	1.14E+01/6.01E-01	1.18E+01/8.12E-01(=)	1.15E+01/5.33E-01(-)	1.10E+01/1.09E+00(=)
23	5.54E+03/7.34E+02(=)	5.45E+03/8.82E+02	5.63E+03/6.53E+02(=)	5.48E+03/8.21E+02(=)	5.94E+03/6.10E+02(=)
24	2.41E+02/1.22E+01(=)	2.46E+02/1.76E+01	2.46E+02/1.28E+01(=)	2.40E+02/1.21E+01(=)	2.43E+02/1.23E+01(=)
25	3.29E+02/1.39E+01(-)	3.17E+02/1.32E+01	3.27E+02/1.34E+01(-)	3.32E+02/1.37E+01(-)	3.31E+02/8.99E+00(-)
26	2.85E+02/8.02E+01(=)	2.87E+02/9.02E+01	2.54E+02/7.53E+01(=)	2.48E+02/7.51E+01(=)	2.51E+02/7.77E+01(=)
27	1.08E+03/2.18E+02(=)	1.02E+03/1.89E+02	1.01E+03/2.70E+02(=)	1.10E+03/1.98E+02(=)	1.13E+03/2.00E+02(-)
28	4.00E+02/1.71E-13(=)	5.47E+02/6.56E+02	4.00E+02/1.75E-13(=)	4.00E+02/1.75E-13(=)	5.48E+02/6.63E+02(=)
+/=/-	0/25/3	-/-/-	1/26/1	0/24/4	1/22/5

problem [40]. In order to find the optimal flight path, globally optimized path planning algorithm such as meta-heuristic algorithms, can be used to instead of the deterministic algorithm, such as the A * algorithm [41]. Using meta-heuristic algorithm has the advantage of being able to jump out of the local optima. Some popular meta-heuristic algorithms have been proposed to deal with the UCAV path planning problem, such as Particle Swarm Optimization (PSO) [42], Artificial Bee Colony (ABC) algorithm [39], and differential evolution (DE) [43]. In this study, we apply the proposed CIJADE to deal with UCAV path planning problem and compare it with standard PSO, DE, ABC, JADE and CIPDE algorithms.

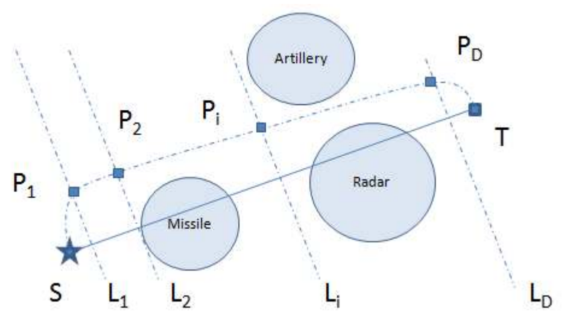


FIGURE 6. UCAV battle field model.

A. MODEL AND INDIVIDUAL ENCODING

In the model for UCAV path planning (see Figure 6), S represents the starting point of the path, and T represents the

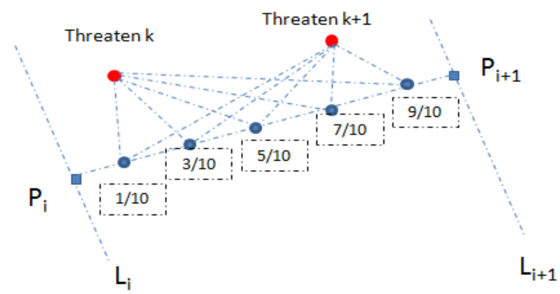


FIGURE 7. Computation of the threat cost.

end point of the path. There are some threatening areas, such as radars, missiles, and artilleries, in the combat field. The effects of threatening areas are presented in terms of circles with different radiuses and threat weights [44]. If a part of its path falls in a threatening areas, the UCAV will be vulnerable to the threat with a certain probability which is proportional to its distance away from the center of threat. While, when its path is outside of the threatening areas, the probability of the UCAV being attacked is zero. The mission of path planning is to find an optimal path between S and T considering all these threatening areas and fuel costs.

We first draw a straight line ST connecting the starting point and the end point and then divide ST into (D+1) equal segments by D vertical dash lines $\{L_1, L_2, \ldots, L_i, \ldots, L_D\}$ as shown in Figure 6. We take these lines as new axes and select

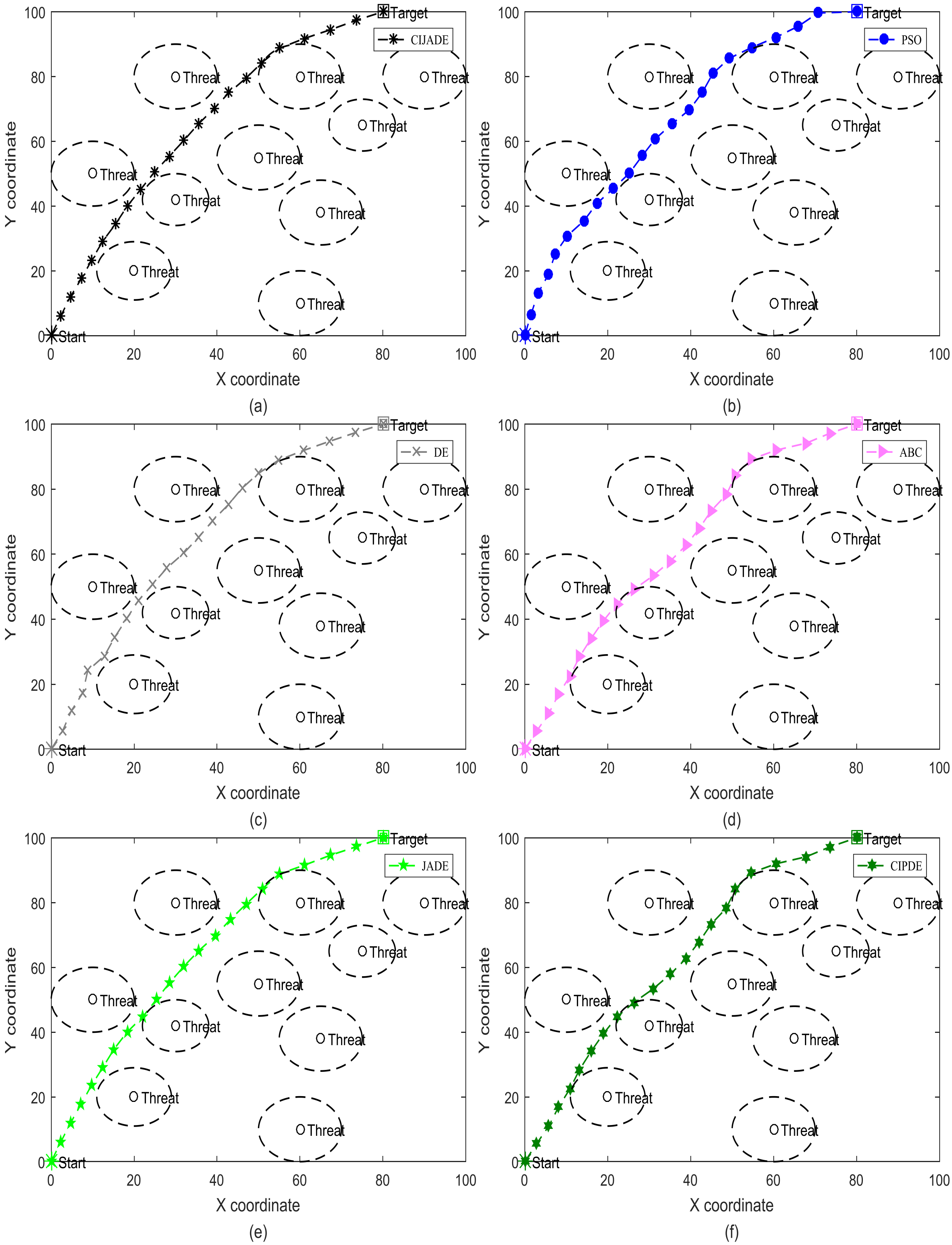


FIGURE 8. Path planning results of (a) CIJADE, (b) PSO, (c) DE, (d) ABC, (e) JADE, and (f) CIPDE when D=20.

TABLE 7. Comparison results of the CIJADE with different T values.

50D	T=10 Mean/Std	T=50 Mean/Std	T=90 Mean/Std	T=130 Mean/Std	T=170 Mean/Std
1	1.25E-13/1.16E-13(-)	2.27E-14/7.00E-14(=)	3.41E-14/8.33E-14	1.25E-13/1.16E-13(-)	4.55E-14/9.33E-14(=)
2	2.01E+04/1.01E+04(=)	2.77E+04/1.49E+04(=)	2.57E+04/1.11E+04	1.85E+04/1.04E+04(+)	1.91E+04/8.55E+03(+)
3	5.00E+06/1.39E+07(=)	2.15E+06/3.84E+06(=)	1.09E+06/2.13E+06	1.17E+06/3.33E+06(=)	3.37E+06/7.44E+06(=)
4	2.03E+04/1.23E+04(-)	1.53E+04/1.48E+04(=)	9.14E+03/1.45E+04	4.36E+03/1.10E+04(=)	5.81E+03/1.21E+04(=)
5	2.05E-13/4.67E-14(-)	1.31E-13/4.16E-14(=)	1.25E-13/3.50E-14	1.19E-13/2.54E-14(=)	1.36E-13/4.67E-14(=)
6	4.34E+01/9.67E-14(=)	4.37E+01/1.27E+00(=)	4.34E+01/6.25E-14	4.34E+01/7.80E-14(=)	4.37E+01/1.27E+00(=)
7	2.26E+01/5.66E+00(-)	1.72E+01/7.95E+00(=)	1.69E+01/6.59E+00	1.43E+01/5.66E+00(=)	1.66E+01/5.90E+00(=)
8	2.11E+01/9.60E-02(=)	2.11E+01/4.42E-02(=)	2.11E+01/4.35E-02	2.11E+01/1.12E-01(=)	2.11E+01/9.73E-02(=)
9	4.12E+01/6.56E+00(=)	3.75E+01/4.60E+00(=)	3.80E+01/3.95E+00	3.91E+01/4.18E+00(=)	4.11E+01/2.93E+00(-)
10	3.46E-02/2.22E-02(=)	3.34E-02/2.15E-02(=)	3.16E-02/2.61E-02	3.60E-02/1.80E-02(=)	3.46E-02/2.19E-02(=)
11	7.31E+00/3.54E+00(-)	0.00E+00/0.00E+00(=)	0.00E+00/0.00E+00	0.00E+00/0.00E+00(=)	0.00E+00/0.00E+00(=)
12	4.66E+01/1.45E+01(+)	4.62E+01/1.32E+01(+)	5.23E+01/8.16E+00	4.27E+01/8.87E+00(+)	4.81E+01/8.43E+00(=)
13	1.16E+02/3.29E+01(=)	9.16E+01/2.31E+01(=)	1.05E+02/2.99E+01	1.07E+02/2.71E+01(=)	1.05E+02/2.53E+01(=)
14	4.75E+02/2.94E+02(-)	8.12E-02/1.79E-02(-)	3.96E-02/2.33E-02	7.58E-02/2.26E-02(-)	7.66E-02/2.09E-02(-)
15	6.49E+03/9.14E+02(=)	6.08E+03/9.12E+02(=)	6.03E+03/8.88E+02	5.75E+03/8.91E+02(=)	5.82E+03/7.78E+02(=)
16	2.75E+00/1.14E+00(=)	2.06E+00/1.17E+00(=)	2.58E+00/1.20E+00	2.67E+00/1.06E+00(=)	2.16E+00/1.16E+00(=)
17	5.91E+01/3.49E+00(-)	5.08E+01/4.75E-14(=)	5.08E+01/5.84E-14	5.08E+01/4.39E-14(=)	5.08E+01/4.66E-14(=)
18	8.97E+01/9.52E+00(-)	7.90E+01/1.05E+01(=)	8.33E+01/5.91E+00	8.79E+01/8.30E+00(=)	8.49E+01/6.69E+00(=)
19	3.86E+00/5.66E-01(-)	2.08E+00/2.71E-01(-)	1.86E+00/2.30E-01	1.93E+00/2.07E-01(=)	1.86E+00/2.62E-01(=)
20	1.94E+01/7.86E-01(-)	1.85E+01/1.05E+00(=)	1.88E+01/7.79E-01	1.88E+01/7.88E-01(=)	1.90E+01/8.90E-01(=)
21	8.09E+02/3.38E+02(=)	8.66E+02/3.63E+02(=)	7.88E+02/4.10E+02	7.10E+02/4.39E+02(=)	8.52E+02/3.58E+02(=)
22	3.15E+02/1.67E+02(-)	1.20E+01/1.39E+00(-)	1.14E+01/6.01E-01	1.15E+01/1.14E+00(-)	2.61E+01/4.26E+01(-)
23	6.65E+03/1.05E+03(-)	5.37E+03/1.14E+03(=)	5.45E+03/8.82E+02	5.38E+03/7.36E+02(=)	5.56E+03/7.12E+02(=)
24	2.41E+02/1.14E+01(=)	2.40E+02/1.54E+01(=)	2.46E+02/1.76E+01	2.41E+02/9.06E+00(=)	2.41E+02/1.53E+01(=)
25	3.16E+02/1.62E+01(=)	3.22E+02/1.27E+01(=)	3.17E+02/1.32E+01	3.24E+02/1.70E+01(=)	3.31E+02/1.15E+01(-)
26	3.02E+02/7.87E+01(=)	2.95E+02/8.25E+01(=)	2.87E+02/9.02E+01	3.12E+02/8.61E+01(=)	2.72E+02/8.15E+01(=)
27	9.83E+02/1.52E+02(=)	9.88E+02/1.48E+02(=)	1.02E+03/1.89E+02	1.08E+03/1.99E+02(=)	1.11E+03/2.25E+02(=)
28	5.49E+02/6.67E+02(=)	8.47E+02/1.09E+03(=)	5.47E+02/6.56E+02	6.94E+02/9.06E+02(=)	4.00E+02/1.75E-13(=)
+/=/-	1/15/12	1/24/3	-/-/-	2/23/3	1/23/4

D pointsP_i from each L_i to form a feasible path from starting point S to end point T, which can be described as follows.

$$path = \{S, P_1, P_2, \ldots, P_i, \ldots, P_D, T\} \tag{28}$$

In this sense, to find optimal path is to determine the location of P_i. Since the location of L_i can be easily obtained, we only need to determine the distance from point P_i to the straight line ST. In other words, we need to determine D parameters in Eq. (28), therefore, it can be considered as D-dimensional optimization problem. To make the problem solving faster, it is encourage to transform the space coordinate system [39]. We take the line ST as the x axis and the straight line perpendicular to ST as the y axis. Let (x, y) stand for the coordinates in the original combat field and (x′, y′) stand for the coordinates after the coordinate transformation, the relationships between these two coordinates are defined in Eq. (29).

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{29}$$

where θ represents the angle between the original x axis and the new x axis line ST.

B. FITNESS EVALUATION

The evaluation of the flight path mainly composes of the threat cost f_{threat} and the fuel consumption f_{fuel}. The objective function (f) can be expressed as follows.

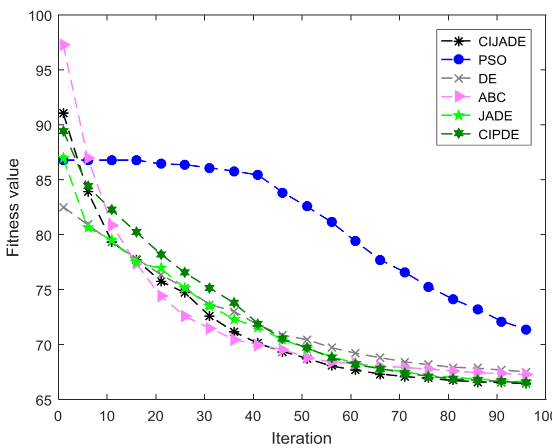


FIGURE 9. Convergence curves of all algorithms when D=20.

$$f = \lambda \cdot f_{threat} + (1 - \lambda) \cdot f_{fuel} \tag{30}$$

$$f_{threat} = \int_S^T w_{threat}\, dl \tag{31}$$

$$f_{fuel} = \int_S^T w_{fuel}\, dl \tag{32}$$

From Eq. (30), the objective function of UCAV flight path is the weighted sum of the threat cost and fuel consumption

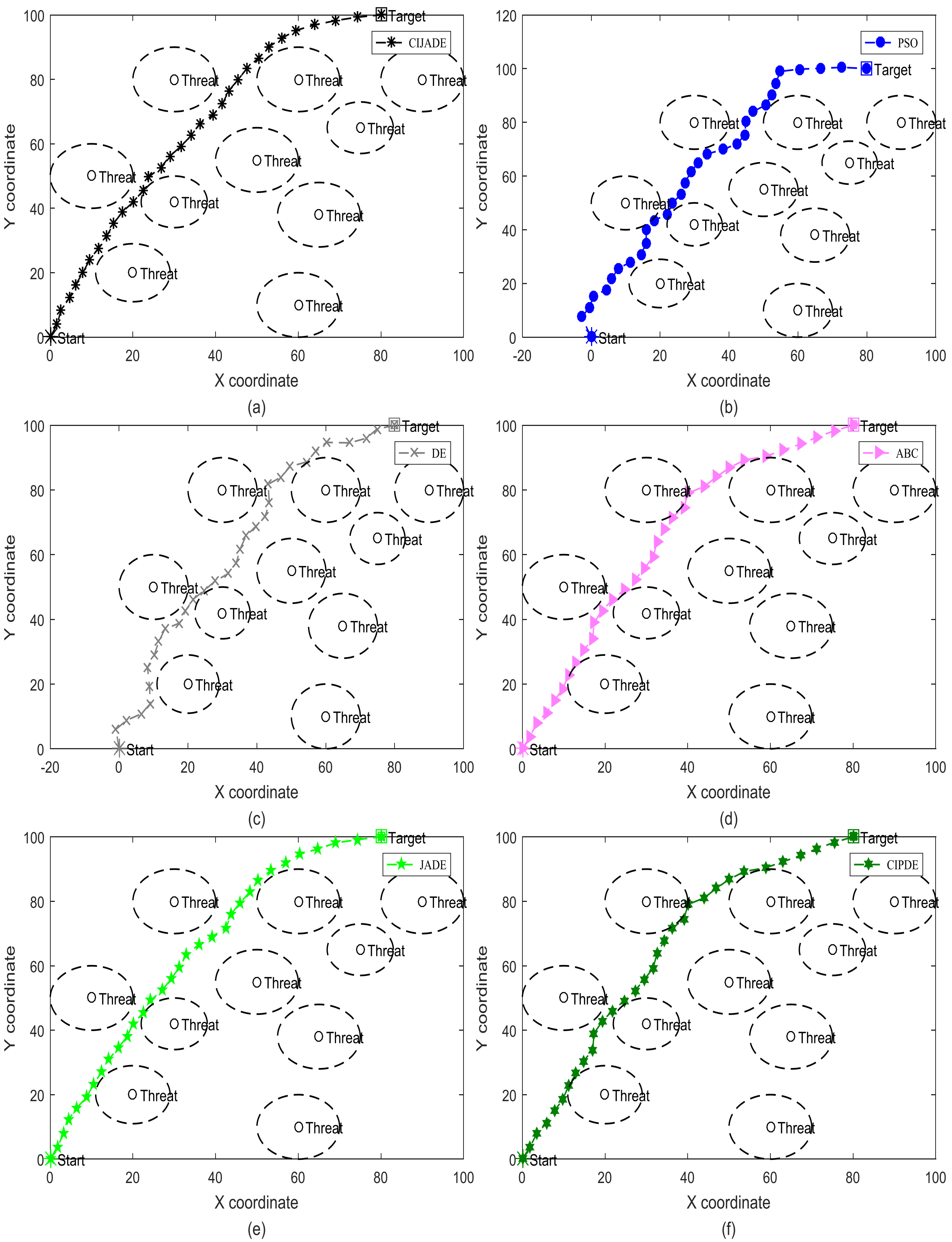


FIGURE 10. **Path planning results of (a) CIJADE, (b) PSO, (c) DE, (d) ABC, (e) JADE, and (f) CIPDE when D=30.**

cost. $\lambda \in [0, 1]$ indicates the weighting parameter. In this study, λ is set to 0.5 according to [45], [39]. w_{threat} and w_{fuel} are variables depend on every instantaneous position on the flight path. In this study, it is assumed that the UCAV flies at a constant speed, then its fuel consumption rate w_{fuel} will be a constant value (i.e., $w_{fuel} \equiv 1$) and the fuel consumption cost f_{fuel} is considered in direct proportion to the length of the flight path.

The threat cost from the point P_i to the point P_{i+1} is calculated by an approximation at five sample points as shown in Figure 7 [46]. If the sub path (P_i, P_{i+1}) falls into a threatening area, the threat cost is calculated as fellows.

$$w_{threat,L_i} = \frac{L_i}{5} \cdot \sum_{k=1}^{N_t} t_k \cdot \left(\frac{1}{d^4_{0.1,i,k}} + \frac{1}{d^4_{0.3,i,k}} + \frac{1}{d^4_{0.5,i,k}} + \frac{1}{d^4_{0.7,i,k}} + \frac{1}{d^4_{0.9,i,k}}\right) \tag{33}$$

where the number of threatening areas is represented by N_t. The length of ith sub path is represented by L_i. The distance between the 1/10 point on the sub path and the kth threat center is represented by $d_{0.1,i,k}$. The degree of kth threat center is denoted by t_k.

TABLE 8. Information of threatening objects.

Index	Threat coordinates	Threat radius	Threat degree
1	(10, 50)	10	8
2	(20, 20)	9	6
3	(30, 42)	8	5
4	(30, 80)	10	4
5	(50, 55)	10	7
6	(60, 10)	10	6
7	(60, 80)	10	7
8	(65, 38)	12	6
9	(75, 65)	8	8
10	(90, 80)	10	10

C. SIMULATION EXPERIMENT

In this section, we verity the performance of the CIJADE for the UCAV path planning problem. The flight scenario of UCAV path planning is the same as the [47] and the parameters of threat areas are listed in Table 8. The coordinates of the start point and end point are set to (0, 0) and (80,100), respectively. We compare the simulation results of the proposed CIJADE algorithm with standard PSO, DE, ABC, JADE, and CIPDE algorithms. The population size is set to 60 for all compared algorithms. The maximum number of iteration is set to 100. The parameters for the CIJADE, JADE, and CIPDE are the same as subsection 4.1. The parameters for the PSO are $c_1 = c_2 = 2$, $w_{\max} = 0.9$, $w_{\min} = 0.4$. The parameters for the DE are $F = 0.5$, $Cr = 0.1$. The parameters for the ABC are $FoodNumber = 30$, limit $= 100$. All compared algorithms are run 20 times with different random seeds.

TABLE 9. The best, worst, mean and standard deviations of fitness values.

Algorithms	D	Best	Worst	Mean	Std
CIJADE	20	**66.29802**	**66.64696**	**66.42659**	**0.087581**
PSO		67.66308	73.96052	70.79475	1.869968
DE		66.73059	72.5714	67.39736	1.249009
ABC		66.63164	68.18436	67.26045	0.512939
JADE		66.32121	67.00737	66.52485	0.193289
CIPDE		66.33105	66.75156	66.46963	0.12627
CIJADE	30	**67.62242**	**75.25686**	**70.29763**	**1.813082**
PSO		73.75303	85.561	78.98284	3.828977
DE		71.76644	81.64503	75.19733	2.434716
ABC		67.84587	76.71826	72.65667	2.147211
JADE		67.52199	81.63118	70.40789	3.026138
CIPDE		68.1706	78.23871	71.03377	2.502887

The experimental results are reported in Table 9. Table 9 lists the best, the worst, the mean and the standard deviation (Std) of the fitness value. The best results are shown in bold. Path planning results optimized by different algorithms are shown in Figures 8 and 10 when D=20 and D=30, respectively. Their corresponding convergence curves of average fitness value are plotted in Figures 9 and 11.

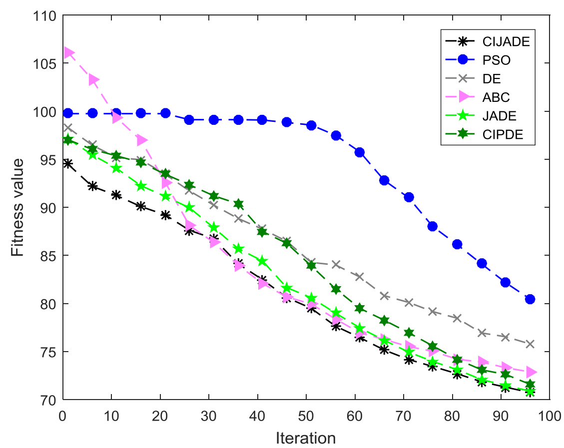


FIGURE 11. Convergence curves of all algorithms when D=30.

As can be seen in Table 9, CIJADE outperforms the other algorithms in terms of the best, the worst, the mean, and the standard deviation on 20D and 30D problem. It should be noted that all algorithms' fitness values become larger when D=30 because a larger D leads to the search space larger. From the figures 8 and 10, we can see that all the compared algorithms can guarantee UCAV avoiding the collision and obstacle. In terms of flight path smooth, from the figure 8, the planning paths optimized by CIJADE and JADE are smoother than the other three algorithms. From the figure 10, the planning path optimized by CIJADE is smoother than the other three algorithms. As can be seen from figures 9 and 11, the convergence rates of CIJADE for 20D and 30D are better than the PSO, DE, and ABC algorithms and competitive to

JADE and CIPDE algorithms. We also can notice that the advantage of CIJADE is more obvious when D=30.

VI. CONCLUSIONS

In this paper, we present a hybrid differential evolution algorithm combining modified CIPDE with modified JADE called CIJADE. Both CIPDE and JADE are powerful and effective DE variant with strong exploration and exploitation capabilities. The goal of our proposed CIJADE algorithm is to take advantage of these two approaches and further improve the optimization performance. In CIJADE, the population is first divided into two subpopulations based on the fitness value, i.e., superior and inferior subpopulations. This dual-population framework can maintain the population diversity. The superior subpopulation evolves using the operation defined in MCIPDE. The MCIPDE adds an external archive to the mutation scheme to enhance the population diversity and exploration capability of original CIPDE. While the inferior subpopulation evolves using the operation defined in MJADE. The MJADE modifies the original JADE by adjusting the parameter p in linear decreasing way, with the aim of balancing the exploration and exploitation ability of JADE. Furthermore, a new crossover operation is added to original JADE to deal with the problem of stagnation. Finally, the parameters CR and F values of CIJADE are updated according to a modified parameter adaptation strategy in each generation.

We use the CEC2013 test suite with 28 benchmark functions to assess the performance of the proposed CIJADE algorithm. The effectiveness of the MCIPDE and MJADE is evaluated by comparisons with original CIPDE and JADE. The parameters of CIJADE are also analyzed. The experimental results demonstrate that the proposed CIJADE algorithm offers better performance than eleven popular state-of-the-art DE variants including jDE, JADE, SaDE, CoDE, Rcr-JADE, CobiDE, MPEDE, AGDE, CIPDE, EFADE and EDEV. Moreover, the proposed CIJADE algorithm is applied to the UCAV path planning problem. The simulation results indicate that the CIJADE can efficiently find the optimal or near optimal flight path for UCAV and find more stability result than the compared algorithms including PSO, DE, ABC, JADE, and CIPDE. In the future work, we will apply the proposed CIJADE algorithm to solve more real-world optimization problems, such as wireless sensor networks [48]–[50], Vehicle Localization and Velocity Estimation [51].

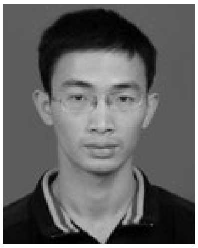
REFERENCES

[1] K. Price and R. Storn, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim.*, vol. 11, no. 4, pp. 341–359, 1997.
[2] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 4–31, Feb. 2011.
[3] S. Das, S. S. Mullick, and P. N. Suganthan, "Recent advances in differential evolution—An updated survey," *Swarm Evol. Comput.*, vol. 27, pp. 1–30, Apr. 2016.
[4] G. Wu, X. Shen, H. Li, H. Chen, A. Lin, and P. Suganthan, "Ensemble of differential evolution variants," *Inf. Sci.*, vol. 423, pp. 172–186, Jan. 2018.
[5] S. G. Santos, R. S. Barros, and P. M. Gonçalves, "A differential evolution based method for tuning concept drift detectors in data streams," *Inf. Sci.*, vol. 485, pp. 376–393, Jun. 2019.
[6] Z. Afzal, P. A. Shah, and K. M. Awan, "Optimum bandwidth allocation in wireless networks using differential evolution," *J. Ambient Intell. Humanized Comput.*, vol. 10, no. 4, pp. 1401–1412, Apr. 2019.
[7] N.-T. Thao, "An improved fuzzy time series forecasting model using the differential evolution algorithm," *J. Intell. Fuzzy Syst.*, vol. 36, no. 2, pp. 1727–1741, Mar. 2019.
[8] S. Nguyen, D. Thiruvady, A. T. Ernst, and D. Alahakoon, "A hybrid differential evolution algorithm with column generation for resource constrained job scheduling," *Comput. Oper. Res.*, vol. 109, pp. 273–287, Sep. 2019.
[9] R. Tang, X. Li, and J. Lai, "A novel optimal energy-management strategy for a maritime hybrid energy system based on large-scale global optimization," *Appl. Energy*, vol. 228, pp. 254–264, Oct. 2018.
[10] Z. Meng and J.-S. Pan, "Monkey king evolution: A new memetic evolutionary algorithm and its application in vehicle fuel consumption optimization," *Knowl.-Based Syst.*, vol. 97, pp. 144–157, Apr. 2016.
[11] Z. Meng, J.-S. Pan, and H. Xu, "QUasi-affine transformation evolutionary (QUATRE) algorithm: A cooperative swarm based algorithm for global optimization," *Knowl.-Based Syst.*, vol. 109, pp. 104–121, Oct. 2016.
[12] Z. Meng and J.-S. Pan, "HARD-DE: Hierarchical ARchive based mutation strategy with depth information of evolution for the enhancement of differential evolution on numerical optimization," *IEEE Access*, vol. 7, pp. 12832–12854, 2019.
[13] A. Qin, V. Huang, and P. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 398–417, Apr. 2009.
[14] Z. Meng, J.-S. Pan, and L. Kong, "Parameters with adaptive learning mechanism (PALM) for the enhancement of differential evolution," *Knowl.-Based Syst.*, vol. 141, pp. 92–112, Feb. 2018.
[15] Q. Fan, X. Yan, and Y. Xue, "Prior knowledge guided differential evolution," *Soft Comput.*, vol. 21, no. 22, pp. 6841–6858, Nov. 2017.
[16] Z. Meng, J.-S. Pan, and K.-K. Tseng, "PaDE: An enhanced Differential Evolution algorithm with novel control parameter adaptation schemes for numerical optimization," *Knowl.-Based Syst.*, vol. 168, pp. 80–99, Mar. 2019.
[17] J. Zhang and A. Sanderson, "JADE: Adaptive differential evolution with optional external archive," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 945–958, Oct. 2009.
[18] L. M. Zheng, S. X. Zhang, K. S. Tang, and S. Y. Zheng, "Differential evolution powered by collective information," *Inf. Sci.*, vol. 399, pp. 13–29, Aug. 2017.
[19] Y. Wang, Z. Cai, and Q. Zhang, "Differential evolution with composite trial vector generation strategies and control parameters," *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 55–66, Feb. 2011.
[20] R. Mallipeddi, P. Suganthan, Q. Pan, and M. Tasgetiren, "Differential evolution algorithm with ensemble of parameters and mutation strategies," *Appl. Soft Comput.*, vol. 11, no. 2, pp. 1679–1696, Mar. 2011.
[21] G. Wu, R. Mallipeddi, P. Suganthan, R. Wang, and H. Chen, "Differential evolution with multi-population based ensemble of mutation strategies," *Inf. Sci.*, vol. 329, pp. 329–345, Feb. 2016.
[22] G. Li, Q. Lin, L. Cui, Z. Du, Z. Liang, J. Chen, N. Lu, and Z. Ming, "A novel hybrid differential evolution algorithm with modified CoDE and JADE," *Appl. Soft Comput.*, vol. 47, pp. 577–599, Oct. 2016.
[23] Y. Chen, L. Li, H. Peng, J. Xiao, Y. Yang, and Y. Shi, "Particle swarm optimizer with two differential mutation," *Appl. Soft Comput.*, vol. 61, pp. 314–330, Dec. 2017.
[24] S. S. Jadon, R. Tiwari, H. Sharma, and J. C. Bansal, "Hybrid artificial bee colony algorithm with differential evolution," *Appl. Soft Comput.*, vol. 58, pp. 11–24, Sep. 2017.
[25] J.-H. Zhong, M. Shen, J. Zhang, H. S.-H. Chung, Y.-H. Shi, and Y. Li, "A differential evolution algorithm with dual populations for solving periodic railway timetable scheduling problem," *IEEE Trans. Evol. Comput.*, vol. 17, no. 4, pp. 512–527, Aug. 2013.
[26] W.-J. Yu and J. Zhang, "Multi-population differential evolution with adaptive parameter control for global optimization," in *Proc. 13th Annu. Conf. Genetic Evol. Comput. (GECCO)*, 2011, pp. 1093–1098.
[27] N. Liu, J. S. Pan, and T. T. Nguyen, "A bi-population QUasi-affine TRansformation evolution algorithm for global optimization and its application to dynamic deployment in wireless sensor networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2019, no. 1, p. 175, 2019.

[28] H. Ma, S. Shen, M. Yu, Z. Yang, M. Fei, and H. Zhou, "Multi-population techniques in nature inspired optimization algorithms: A comprehensive survey," *Swarm Evol. Comput.*, vol. 44, pp. 365–387, Feb. 2019.
[29] L. Cui, Q. Huang, G. Li, S. Yang, Z. Ming, Z. Wen, N. Lu, and J. Lu, "Differential evolution algorithm with tracking mechanism and backtracking mechanism," *IEEE Access*, vol. 6, pp. 44252–44267, 2018.
[30] S.-M. Guo, C.-C. Yang, P.-H. Hsu, and J. S.-H. Tsai, "Improving differential evolution with a successful-parent-selecting framework," *IEEE Trans. Evol. Comput.*, vol. 19, no. 5, pp. 717–730, Oct. 2015.
[31] S. M. Islam, S. Das, S. Ghosh, S. Roy, and P. N. Suganthan, "An adaptive differential evolution algorithm with novel mutation and crossover strategies for global numerical optimization," *IEEE Trans. Syst., Man, Cybern. B. Cybern.*, vol. 42, no. 2, pp. 482–500, Apr. 2012.
[32] W. Gong, Z. Cai, and Y. Wang, "Repairing the crossover rate in adaptive differential evolution," *Appl. Soft Comput.*, vol. 15, pp. 149–168, Feb. 2014.
[33] J. J. Liang, B. Y. Qu, P. N. Suganthan, and A. G. Hernández-Díaz, "Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization," Comput. Intell. Lab., Zhengzhou Univ., Zhengzhou, China, Tech. Rep. 201212, 2013.
[34] J. Brest, S. Greiner, B. Boskovic, M. Mernik, and V. Zumer, "Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems," *IEEE Trans. Evol. Comput.*, vol. 10, no. 6, pp. 646–657, Dec. 2006.
[35] Y. Wang, H.-X. Li, T. Huang, and L. Li, "Differential evolution based on covariance matrix learning and bimodal distribution parameter setting," *Appl. Soft Comput.*, vol. 18, pp. 232–247, May 2014.
[36] A. W. Mohamed and A. K. Mohamed, "Adaptive guided differential evolution algorithm with novel mutation for numerical optimization," *Int. J. Mach. Learn. Cyber.*, vol. 10, no. 2, pp. 253–277, Feb. 2019.
[37] A. W. Mohamed and P. N. Suganthan, "Real-parameter unconstrained optimization based on enhanced fitness-adaptive differential evolution algorithm with novel mutation," *Soft Comput.*, vol. 22, no. 10, pp. 3215–3235, May 2018.
[38] S. Ghambari, J. Lepagnot, L. Jourdan, and L. Idoumghar, "A comparative study of meta-heuristic algorithms for solving UAV path planning," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Nov. 2018, pp. 174–181.
[39] B. Li, L.-G. Gong, and W.-L. Yang, "An improved artificial bee colony algorithm based on balance-evolution strategy for unmanned combat aerial vehicle path planning," *Sci. World J.*, vol. 2014, pp. 1–10, 2014.
[40] J. S. Pan, J. L. Liu, and S. C. Hsiung, "Chaotic cuckoo search algorithm for solving unmanned combat aerial vehicle path planning problems," in *Proc. ACM Int. Conf.*, 2019, pp. 224–230.
[41] X. Chen, X. M. Chen, and J. Zhang, "The dynamic path planning of UAV based on A* Algorithm," *Appl. Mech. Mater.*, vols. 494–495, pp. 1094–1097, Feb. 2014.
[42] Y. Zhang, L. Wu, and S. Wang, "UCAV path planning by fitness-scaling adaptive chaotic particle swarm optimization," *Math. Problems Eng.*, vol. 2013, pp. 1–9, 2013.
[43] K. Y. Kok and P. Rajendran, "Differential-evolution control parameter optimization for unmanned aerial vehicle path planning," *PLoS ONE*, vol. 11, no. 3, Mar. 2016, Art. no. e0150558.
[44] G. Wang, L. Guo, H. Duan, L. Liu, and H. Wang, "A bat algorithm with mutation for UCAV path planning," *Sci. World J.*, vol. 2012, Dec. 2012, Art. no. 418946.
[45] C. Xu, H. Duan, and F. Liu, "Chaotic artificial bee colony approach to uninhabited combat air vehicle (UCAV) path planning," *Aerosp. Sci. Technol.*, vol. 14, no. 8, pp. 535–541, Dec. 2010.
[46] D. Rodić and A. P. Engelbrecht, "Social networks in simulated multi-robot environment," *Int. J. Intell. Comput. Cybern.*, vol. 1, no. 1, pp. 110–127, 2008.
[47] N. Lin, J. Tang, X. Li, and L. Zhao, "A novel improved bat algorithm in UAV path planning," *Comput., Mater. Continua*, vol. 61, no. 1, pp. 323–344, 2019.
[48] J.-S. Pan, L. Kong, T.-W. Sung, P.-W. Tsai, and V. Snasel, "Alpha-fraction first strategy for hierarchical wireless sensor networks," *J. Internet Technol.*, vol. 19, no. 6, pp. 1717–1726, 2018.
[49] J.-S. Pan, L. Kong, T.-W. Sung, P.-W. Tsai, and V. Snášel, "A clustering scheme for wireless sensor networks based on genetic algorithm and dominating set," *J. Internet Technol.*, vol. 19, no. 4, pp. 1111–1118, 2018.
[50] J. Wang, Y. Gao, W. Liu, A. K. Sangaiah, and H.-J. Kim, "Energy efficient routing algorithm with mobile sink support for wireless sensor networks," *Sensors*, vol. 19, no. 7, p. 1494, Mar. 2019.
[51] C.-H. Chen, C.-A. Lee, and C.-C. Lo, "Vehicle localization and velocity estimation based on mobile phone sensing," *IEEE Access*, vol. 4, pp. 803–817, 2016.

JENG-SHYANG PAN (Senior Member, IEEE) received the B.S. degree in electronic engineering from the National Taiwan University of Science and Technology, in 1986, the M.S. degree in communication engineering from National Chiao Tung University, Taiwan, in 1988, and the Ph.D. degree in electrical engineering from the University of Edinburgh, U.K., in 1996.

He is currently the Dean with the College of Information Science and Engineering, Fujian University of Technology. He is also a Ph.D. Supervisor with Fuzhou University. He joined the Editorial Board of LNCS *Transactions on Data Hiding and Multimedia Security*, the *Journal of Computers*, the *Journal of Information Hiding*, and *Multimedia Signal Processing*. His current research interests include evolutionary computation, information security, and signal processing.



NENGXIAN LIU received the B.S. degree in information and computing science from Quanzhou Normal University, in 2006, the M.S. degree in computer science from Fuzhou University, in 2009. He is currently pursuing the Ph.D. degree with Fuzhou University. He is currently an Assistant Researcher with Fuzhou University. His current research interests include evolutionary algorithms and their applications in real-world problems.

SHU-CHUAN CHU received the Ph.D. degree from the School of Computer Science, Engineering and Mathematics, Flinders University, Australia, in 2004. She joined Flinders University, in December 2009, after nine years at Cheng Shiu University, Taiwan. She has been the Research Fellow with the College of Science and Engineering, Flinders University, since December 2009. She has also been the Research Fellow with a Ph.D. Advisor with the College of Computer Science and Engineering, Shandong University of Science and Technology, since September 2019. Her research interests are mainly in swarm intelligence, intelligent computing, and data mining.