Information Sciences 506 (2020) 174–195

Contents lists available at ScienceDirect

Information Sciences

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

Individual-dependent feasibility rule for constrained differential evolution

Bing-Chuan Wang[a], Yun Feng[a,*], Han-Xiong Li[a,b]

[a] Department of Systems Engineering and Engineering Management, City University of Hong Kong, Hong Kong
[b] State Key Laboratory of High Performance Complex Manufacturing, School of Mechanical and Electrical Engineering, Central South University, Changsha, China

ARTICLE INFO

Article history:
Received 6 December 2018
Revised 11 July 2019
Accepted 21 July 2019
Available online 1 August 2019

Keywords:
Constrained evolutionary optimization
Individual-dependent
Feasibility rule
Diversity strategy

ABSTRACT

Benefiting from its efficiency and quick convergence, the feasibility rule (FR) is well-known for its ability to solve constrained optimization problems (COPs). However, it is highly criticized for its heavy preference for constraints. Thus, an individual-dependent feasibility rule (IDFR) was designed by alleviating the preference from two aspects. Some information of constraints, which might be nonsignificant, is depressed. By contrast, some promising information of objective function is leveraged. The extent of information is individual-dependent. To further enhance the diversity, a two-phase diversity strategy was developed. Due to their numerous merits, two differential evolution (DE) operators were selected as components of the search algorithm. By the above process, we proposed a constrained DE (i.e., IDFRDE). To the best of our knowledge, we made the first attempt to improve FR from the individual perspective. IDFR is more robust than FR while keeping the same computational time complexity. However, it would converge slower than FR on some easy COPs. Experiments on three widely used benchmarks show that: 1) IDFRDE outperforms or gets similar results comparing with other known algorithms; 2) IDFR is more effective than FR on complex COPs; 3) both, the search algorithm and the diversity strategy are important to IDFRDE.

1. Introduction

Optimization problems are ubiquitous across the engineering as well as the scientific area [20,26]. In most cases, these problems unavoidably deal with constraints. Without loss of generality, a constrained optimization problem (COP) is formulated as follows [18]:

$$\begin{aligned} &\text{minimize } f(\vec{x}),\ \vec{x} = (x_1, \ldots, x_D) \in S,\ L_i \le x_i \le U_i \\ &\text{subject to}: g_j(\vec{x}) \le 0,\ j = 1, \ldots, l \\ &\qquad\qquad h_j(\vec{x}) = 0,\ j = l+1, \ldots, m, \end{aligned}$$

where $f(\vec{x})$ is the objective function to be optimized; $\vec{x}$ is a D-dimensional decision vector (individual/solution); L_i and U_i are the lower and upper bounds of the ith decision variable (i.e., x_i), respectively; $S = \prod_{i=1}^{D}[L_i, U_i]$ is the decision space; $g_j(\vec{x})$ is the jth inequality constraint; $h_j(\vec{x})$ is the $(j-l)$th equality constraint; l and $(m-l)$ are the numbers of inequality and equality constraints, respectively. When an optimization algorithm is used to solve a COP, the degree of constraint violation

* Corresponding author.
E-mail address: yun.feng@my.cityu.edu.hk (Y. Feng).

https://doi.org/10.1016/j.ins.2019.07.076

is calculated as follows [3]:

$$G(\vec{x}) = \sum_{j=1}^{m} G_j(\vec{x}) \tag{1}$$

where $G_j(\vec{x})$ is the degree of constraint violation on the jth constraint:

$$G_j(\vec{x}) = \begin{cases} \max(0, g_j(\vec{x})) & 1 \le j \le l \\ \max(0, |h_j(\vec{x})| - \delta) & l+1 \le j \le m \end{cases} \tag{2}$$

where δ is a positive tolerance value to relax an equality constraint to two inequality ones. The ultimate goal of solving a COP is to seek its feasible optimum.

Due to their powerful search ability, evolutionary algorithms (EAs) [2] are widely used to tackle COPs. When an EA is employed for constrained optimization, a constraint-handling technique (CHT) is necessary to be integrated [18]. Consequently, a large collection of CHTs are developed [16,17,41]. According to the manner of treating objective function and constraints, CHTs can be classified into four categories: 1) CHTs based on penalty function [13,29], 2) CHTs based on treating constraints and objective function separately [7,36,37], 3) CHTs based on multiobjective optimization [9,38], and 4) hybrid CHTs [14,40]. CHTs based on penalty function aggregate objective function and constraints together by a penalty factor. Its name indicates that the second kind of CHTs exploits constraints and objective function separately. Similarly, CHTs based on multiobjective function treat degree of constraint violation as additional objective function(s). Based on the “no free lunch theorem” [42], hybrid CHTs try to combine the advantages of different CHTs.

Among these CHTs, the feasibility rule (FR) [7] is a representative one. As its name implies, this method prefers feasible solutions. To be specific, FR is described by three rules as follows:

- A feasible solution is considered to be better than an infeasible one.
- Two feasible solutions are compared based on their objective function values.
- Two infeasible solutions are compared according to the degree of constraint violation.

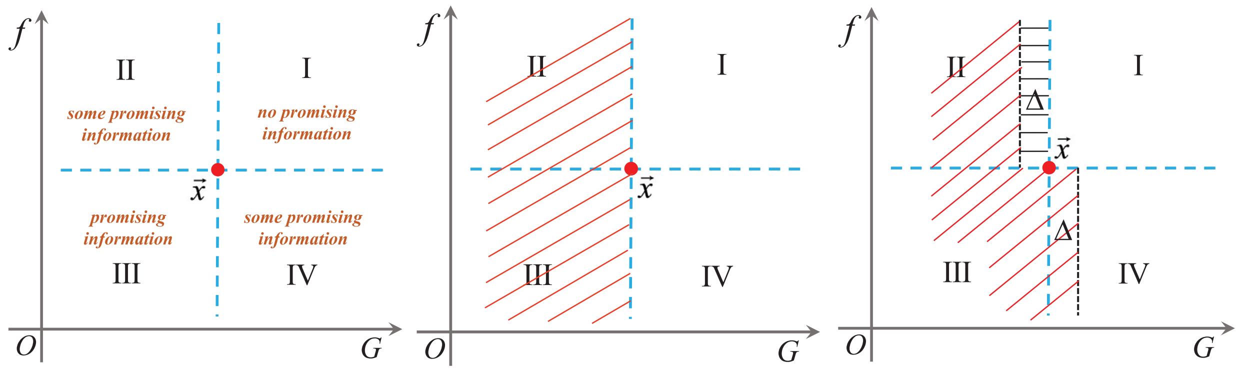
As described above, FR has a strong preference for feasible solutions. As a result, it is capable to motivate the population to the feasible region promptly. That is to say, the convergence can be accelerated. Additionally, it is easy to be implemented and does not involve additional parameters. These advantages facilitate its successful applications to diverse fields [10,24]. However, everything has its pros and cons. Due to its preference for feasible solutions, FR runs a high risk of premature convergence. The population prone to be stuck in a local optimum. When a COP exhibits complex constraints, the situation would be even worse. The population would be stuck into a local optimum in the infeasible region. Consequently, even a feasible solution cannot be obtained. Although this disadvantage is described in some papers [4,41], most methods use FR directly without modification and put their emphasis on designing search algorithms [25,30,33]. Few research studies focus on improving FR.

In the feasibility rule with the incorporation of objective function information (FROFI) [41], the greedy property of FR is addressed by incorporating the information of objective function from three aspects. First, some infeasible solutions with small objective function values discarded by FR are introduced into the population by a replacement mechanism. Additionally, the comparison of individuals is based on the objective function value in the mutation strategy. Furthermore, the information of objective function is also used to generate offspring in the search algorithm. Although FROFI can remedy the shortcomings of FR to a great extent, additional storage space is needed to archive solutions. Besides, since a sorting procedure is involved in the replacement strategy, FROFI is less efficient than FR.

Based on the above observations, we proposed an individual-dependent feasibility rule (IDFR) in this paper. Different from FROFI, IDFR addressed the greedy property of FR from two perspectives. First, some information of constraints, which might be nonsignificant, was depressed. In addition, some information of objective function, which may be promising, was leveraged. The extent of information depressed/leveraged was dependent on the situation of each individual. Additionally, to further enhance the diversity, a two-phase strategy was designed. As we know, the search algorithm is another critical component of a constrained EA (COEA). In this paper, a simple yet effective differential evolution (DE) was developed to generate offspring. By the above process, we proposed a constrained DE (i.e., IDFRDE). Compared with FROFI, IDFRDE is more efficient due to the fact that the replacement strategy is omitted. Besides, without an additional archive, IDFRDE is more compact than FROFI. The main contributions of this paper are summarized as follows:

- An individual-dependent feasibility rule was proposed to handle constraints.
- A two-phase diversity strategy was proposed to enhance the diversity.
- A simple yet effective search algorithm was designed to generate solutions.
- Experiments on three widely used benchmark sets were conducted to investigate the effectiveness of IDFRDE.

The rest of this paper is organized as follows. Section 2 briefly introduces DE which is used to design the search algorithm. IDFRDE is detailed in Section 3. Extensive experiments and discussions are summarized in Section 4. Section 5 concludes the remarks of this paper.



(a) The objective space of a COP (b) Illustration of FR in the objective space (c) Illustration of IDFR in the objective space

Fig. 1. Methodology of individual-dependent feasibility rule (IDFR).

2. Preliminary knowledge–differential evolution

Differential evolution (DE) is a simple yet effective search algorithm in the community of evolutionary computation [5,6,19]. It is named as "differential" evolution because it uses differential vectors to generate solutions. Conventionally, DE includes four stages: initialization, mutation, crossover, and selection.

At the initialization stage, a population with the size of P_S is uniformly generated from the decision space: $P = \{\vec{x}_1, \cdots, \vec{x}_{P_S}\}$. At the mutation stage, for a solution (i.e., target vector) $\vec{x}_i$, a mutant vector $\vec{v}_i$ is generated by a mutation operator. A widely used mutation operator, that is, DE/rand/1, is given as follows [6,19]:

$$\vec{v}_i = \vec{x}_{r1} + F \cdot (\vec{x}_{r2} - \vec{x}_{r3}) \tag{3}$$

where F is the scaling factor; $\vec{x}_{r1}$, $\vec{x}_{r2}$, and $\vec{x}_{r3}$ are three mutually different solutions, which are randomly selected from the population. Note that they are all distinct from $\vec{x}_i$. Other two mutation operators used in this paper are given as follows:

1) DE/rand-to-best/1

$$\vec{v}_i = \vec{x}_{r1} + F \cdot (\vec{x}_{best} - \vec{x}_{r1}) + F \cdot (\vec{x}_{r2} - \vec{x}_{r3}) \tag{4}$$

2) DE/current-to-rand/1

$$\vec{v}_i = \vec{x}_i + rand \cdot (\vec{x}_{r1} - \vec{x}_i) + F \cdot (\vec{x}_{r2} - \vec{x}_{r3}) \tag{5}$$

where $\vec{x}_{best}$ is the solution with the best performance, and *rand* is a random value sampled from a uniform distribution between 0 and 1.

At the crossover stage, for each mutant vector $\vec{v}_i$, a trial vector $\vec{u}_i$ is generated by a crossover operator. The popular binomial crossover operator [23,39] is mathematically formulated as follows:

$$u_{i,j} = \begin{cases} v_{i,j}, & \text{if } rand < CR \text{ or } j = rand_j \\ x_{i,j}, & \text{otherwise} \end{cases} \tag{6}$$

where $u_{i,j}$, $v_{i,j}$, and $x_{i,j}$ are the jth dimension of $\vec{u}_i$, $\vec{v}_i$, and $\vec{x}_i$, respectively; $rand_j$ is an integer randomly chosen from $\{1, \cdots, D\}$, and CR is the crossover control parameter.

Finally, a selection operator is performed on each pair of $\vec{x}_i$ and $\vec{u}_i$. If $\vec{u}_i$ is better than $\vec{x}_i$, it would replace $\vec{x}_i$ and enter into the next generation.

3. Proposed algorithm–IDFRDE

3.1. Motivation

In fact, the mechanism of a CHT can be analyzed from the perspective of balancing constraints and objective function. As shown in Fig. 1(a), from the perspective of an individual $\vec{x}$, it can divide the objective space, which uses the objective function f and the degree of constraint violation G as two coordinates, into four subspaces. Intuitively, f and G of all solutions in subspace I are worse than those of $\vec{x}$. In other words, subspace I contains no promising information. A solution in subspace II owns better G than $\vec{x}$ while its f is worse than that of $\vec{x}$. That is to say, subspace II contains some promising information of constraints. Inversely, solutions in subspace IV have better f but worse G than $\vec{x}$. Thus, we recognize that subspace IV contains some promising information of objective function. Additionally, f and G of solutions in subspace III are better than those of $\vec{x}$. Subspace III includes promising information of both constraints and objective function. Without a doubt, solutions in subspace I are worse than $\vec{x}$, while solutions in subspace III are better than $\vec{x}$. The key task of a CHT is to decide whether a solution in subspace II or IV is better than $\vec{x}$.

As shown in Fig. 1(b), FR considers all solutions in subspace II to be better than $\vec{x}$. All solutions in subspace IV are discarded. Such a preference for the information of constraints leads to the capability to rapidly motivate the population toward the feasible region. However, this greedy property would introduce some disadvantages. First, the population would converge to some big feasible regions promptly. Consequently, the small ones containing the optimum would be lost. Additionally, when a COP with easy constraints is encountered, the population would enter into a feasible region quickly. As a result, the optimum on the boundary between the infeasible region and feasible region would be missed [16,27,28]. When a COP with complex constraints is solved, multiple local optima would exist in the infeasible regions. The population would be stuck easily. These issues would limit the FR's performance to a great extent.

To address the greedy property, some information in subspace II should be depressed and some promising information in subspace IV should be leveraged. In view of this, an individual-dependent feasibility rule (IDFR) was proposed. To further improve the diversity, a two-phase diversity strategy was designed. Additionally, a DE based search algorithm was developed to generate solutions. By the above process, we proposed a constrained DE called IDFRDE. Different from FROFI which incorporates the information of objective function at the population level, IDFRDE introduces the information of objective function at the individual level. In FROFI, some solutions use FR for comparison and some solutions enter the population based on the objective function value. Thus, we do not analyze FROFI from the individual perspective. Instead, IDFRDE uses IDFR to compare solutions, which can be analyzed from the individual perspective as shown in Fig. 1(c).

3.2. Framework

IDFRDE maintains a population of P_S solutions, that is, $P = \{\vec{x}_1, \cdots, \vec{x}_{P_S}\}$. The corresponding objective function values and degree of constraint violation are $FV = \{f(\vec{x}_1), \cdots, f(\vec{x}_{P_S})\}$ and $GV = \{G(\vec{x}_1), \cdots, G(\vec{x}_{P_S})\}$, respectively. The framework of IDFRDE is detailed as follows.

Step 1) Initialization:

Step 1.1) Generate a random population with P_S individuals: $P = \{\vec{x}_1, \ldots, \vec{x}_{P_S}\}$, and evaluate P: $PV = \{(f(\vec{x}_1), G(\vec{x}_1)), \cdots, (f(\vec{x}_{P_S}), G(\vec{x}_{P_S}))\}$.

Step 2) Population updating:

Step 2.1) Generate the offspring population O by using the evolutionary operators in the **search algorithm**.

Step 2.2) Evaluate the offspring population.

Step 2.3) Apply the **individual-dependent feasibility rule (IDFR)** to select solutions.

Step 2.4) Execute the **two-phase diversity strategy**.

Step 3) Stopping criteria: If the stopping criterion is satisfied, then stop the procedure and output the best solution; otherwise, go to **Step 2**).

As shown in the framework, IDFRDE includes three key ingredients: IDFR, the two-phase diversity strategy, and the search algorithm. We will illustrate them next elaborately.

3.3. Individual-dependent feasibility rule

As described in Section 3.1, to address the shortcomings of FR, some information in subspace II should be depressed while some information in subspace IV should be used. In other words, some solutions in subspace II are considered to be worse than $\vec{x}$ while some solutions in subspace IV are considered to be better than $\vec{x}$. To be specific, in subspace II, solutions with relatively bigger degree of constraint violation (i.e., G) are regarded to be worse. If solutions with relatively smaller G are discarded, it would have a seriously negative impact on finding a feasible solution. Similarly, in subspace IV, solutions with smaller G are considered to be better. As a consequence, it would have little negative impact on seeking a feasible solution. In summary, a solution $\vec{y}$ is considered to be better than $\vec{x}$, when one of the following conditions is satisfied:

1. $G(\vec{y}) \leq G(\vec{x}) - \Delta \& f(\vec{y}) > f(\vec{x})$
2. $G(\vec{y}) \leq G(\vec{x}) \& f(\vec{y}) < f(\vec{x})$
3. $G(\vec{x}) \leq G(\vec{y}) \leq G(\vec{x}) + \Delta \& f(\vec{y}) < f(\vec{x})$

For ease of understanding, IDFR is illustrated in Fig. 1(c). As shown in the figure, solutions in the red shaded area are considered to be better than $\vec{x}$. Compared with FR, the black shaded area in subspace II is replaced by the red shaded area in subspace IV. In this manner, some information of constraints in subspace II can be depressed while some information of objective function in subspace IV can be made use of. Note that IDFR will degenerate to FR when Δ is set to 0. The reasons for using two vertical lines to define the better region are threefold. First, in fact, the solutions in the red shaded area are better than $\vec{x}$ in terms of f or G. It is promising to find a solution with smaller f or G. Additionally, in this manner, the red shaded area can be described by simple addition and subtraction equations, and only a parameter Δ is involved. Moreover, experimental results show that IDFR is effective to solve COPs. Of course, other sophisticated methods, such as using nonvertical lines, may be more effective than the proposed method. Though they can take into consideration the improvement of f, these methods are more complicated. Besides, it is not easy to decide the initial value of the threshold similar to Δ. Thus, nonvertical lines could be taken into consideration in our future work.

Intuitively, Δ is a threshold value which decides the amount of information of objective function used. It should be elaborately adjusted due to the fact that different amount of information of objective function is needed at different evolving stages. Some valuable knowledge [13,28,31] summarized in the community of constrained evolutionary optimization can help to tune this parameter. At the early stage of optimization, exploration in the infeasible region is necessary. In this manner, discrete feasible regions and the optimum on the boundary between the infeasible and feasible regions can be located more easily. To this end, more information of objective function should be used at the early stage. At the later stage, the population needs to enter into the feasible region promptly. Thus, less information of objective function should be used. Based on the above analysis, Δ is dynamically adjusted as follows:

$$\Delta = \begin{cases} \Delta_0(1-\frac{t}{T})^{cp}, & \text{if } t \le Tc \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

$$cp = -\frac{\log \Delta_0 + \lambda}{\log(1-\frac{Tc}{T})} \tag{8}$$

where Δ_0 is the initial threshold which is set to the maximum difference between the degree of constraint violation of the initial population and that of their trial vectors; t and T are the current and maximum generation numbers, respectively; Tc is a parameter to truncate the value of Δ, and $10^{-\lambda}$ is the threshold when t is equal to Tc. Similar to [32], in order to improve the usability, if the feasible proportion (i.e., *FeaPro*) exceeds *FP* (i.e., 0.85), Δ would be set to 0. As shown in (7), Δ decreases as generation increases, which is in line with the above analysis.

In summary, IDFR can address the greedy property of FR to some extent. Because $(G(\vec{x})-\Delta)$ and $(G(\vec{x})+\Delta)$ are different in case of different individuals, each individual can decide which information to be used based on its own situation. That is to say, the extent of information depressed/leveraged is individual-dependent.

3.4. Two-phase diversity strategy

In order to further enhance the diversity, we proposed a two-phase diversity strategy, which is composed of a mutation scheme and a restart scheme. When the population is not stuck in a local optimum in the infeasible region completely, the mutation scheme, which would not introduce many disturbances, is used to increase the diversity. When the population is stuck completely, the restart scheme, which regenerates the population, is adopted. First, an indicator should be proposed to judge whether the population is stuck completely or not. Intuitively, if the population is stuck in the infeasible region, the similarity between every two individuals would be tiny. In this case, the standard deviation of the degree of constraint violation (i.e., $std(G(\vec{x}_1),\cdots,G(\vec{x}_{P_S})))$ would be tiny. Thus, the standard deviation is chosen as the indicator. To be specific, if it is bigger than a given threshold (i.e., μ), the mutation scheme would be executed; otherwise, the restart scheme would be triggered.

In the mutation scheme, the individual with the smallest $G(\vec{x})$ (i.e., $\vec{x}_{Gbest}$) is mutated to generate a new individual, under the assumption that an individual generated around $\vec{x}_{Gbest}$ may be close to the feasible region. First, $\vec{x}_{Gbest}$ is copied to a new individual $\vec{x}_t$. Afterward, $\vec{x}_t$ is mutated through a Gaussian distribution:

$$x_{t,d} = \mathcal{N}\left(x_{t,d}, \frac{U_d - L_d}{20}\right) \tag{9}$$

where $x_{t,d}$ is the dth dimension of $\vec{x}_t$; $\mathcal{N}(x_{t,d}, \frac{U_d-L_d}{20})$ is a random value obeying the Gaussian distribution with mean value $x_{t,d}$ and standard deviation $\frac{U_d-L_d}{20}$, and d is an integer selected from $\{1, \cdots, D\}$ as follows:

$$d = \underset{i}{\operatorname{argmin}}\, std(x_{1,i},\cdots,x_{P_S,i}) \tag{10}$$

where std is a function to calculate the standard deviation, and $x_{j,i}$ is the ith dimension of $\vec{x}_j,\ j=1,\cdots,P_S$. By this way, the dimension with the smallest standard deviation is disturbed. Thus, diversity can be enhanced. Additionally, only one dimension is regenerated, which would not introduce too many disturbances. Subsequently, $\vec{x}_t$ is compared with the individual with the biggest $G(\vec{x})$ (i.e., $\vec{x}_{G\max}$). If $f(\vec{x}_t) < f(\vec{x}_{G\max})$ or $G(\vec{x}_t) < G(\vec{x}_{G\max})$, $\vec{x}_{G\max}$ would be replaced by $\vec{x}_t$. In fact, even if $\vec{x}_{G\max}$ is replaced, it would not have a severe impact on finding a feasible solution. As a result, the mutation scheme can introduce diversity to some extent and it would not cause a negative impact on seeking a feasible solution.

In the restart scheme, once the population is stuck completely, it will be regenerated. The reason for regenerating all solutions is as follows. When the population is stuck in the infeasible region, the solutions in the population are close to the local optimum. If we keep some solutions in the population, they would attract other solutions to the local optimum quickly. In summary, the details of the two-phase diversity strategy are described in Algorithm 1 .

Remark 1. It is necessary to point out that similar strategies were proposed to enhance diversity in our previous works [37,41]. The main differences are that the method in [41] only uses a mutation scheme and the method in [37] only uses a restart scheme. However, in the proposed diversity strategy, both a mutation scheme and a restart scheme are used. Additionally, in [41], a uniform distribution is used for mutation and the comparison is only based on $f(\vec{x})$. In the proposed diversity strategy, a Gaussian distribution is used and the comparison is based on $f(\vec{x})$ and $G(\vec{x})$.

Algorithm 1: Two-phase diversity strategy.

if $std(\{G(\vec{x}_1), \cdots, G(\vec{x}_{P_S})\}) > \mu$ **then**
  $d = \underset{i}{\operatorname{argmin}}\, std(x_{1,i}, \cdots, x_{P_S,i})$;
  $\vec{x}_t = \vec{x}_{Gbest}$;
  $x_{t,d} = \mathcal{N}(x_{t,d}, \frac{U_d - L_d}{20})$;
  if $x_{t,d} < L_d$ **then**
    $x_{t,d} = L_d$;
  else if $x_{t,d} > U_d$ **then**
    $x_{t,d} = U_d$;
  Evaluate $\vec{x}_t$;
  if $G(\vec{x}_t) < G(\vec{x}_{G_{\max}}) || f(\vec{x}_t) < f(\vec{x}_{G_{\max}})$ **then**
    Replace $\vec{x}_{G_{\max}}$ with $\vec{x}_t$;
else
  Regenerate the population from the decision space randomly;

3.5. Search algorithm

As we know, search algorithm also plays a critical role in a COEA. Conventionally, when designing the search algorithm, we should consider two issues: 1) balancing the diversity and the convergence, and 2) balancing the constraints and the objective function [36,37]. Due to its numerous advantages, DE is adopted to design the search algorithm. As described in Section 2, different DE operators have distinct properties, which implies that they are potential to address the above two issues satisfactorily. To this end, "DE/current-to-rand/1" and "DE/rand-to-best/1/bin" are adopted.

With respect to (5), each individual $\vec{x}_i$ learns from an individual $\vec{x}_{r1}$, which is randomly selected from the population. Thus, "DE/current-to-rand/1" can improve the diversity to some extent. As shown in Eq. (4), the individual with the best performance is used to generate mutant vectors. Therefore, "DE/rand-to-best/1/bin" is able to promote the convergence. By combining these two operators together, the tradeoff between the diversity and the convergence can be achieved. For the sake of simplicity, they are used with the same probability (i.e., 0.5).

The tradeoff between the constraints and the objective function is another core issue to be addressed. In fact, how to select the "best" individual in the "DE/rand-to-best/1/bin" is closely related to this issue. In view of this, the "best" is selected according to the weighted distance value, which is described as follows:

$$DTC(\vec{x}_i) = \sqrt{p_f \cdot f_n^2(\vec{x}_i) + (1 - p_f) \cdot G_n^2(\vec{x}_i)} \tag{11}$$

$$f_n(\vec{x}_i) = \frac{f(\vec{x}_i) - f_{\min}}{f_{\max} - f_{\min}} \tag{12}$$

$$G_n(\vec{x}_i) = \frac{G(\vec{x}_i) - G_{\min}}{G_{\max} - G_{\min}} \tag{13}$$

where $f_{\min}$ and $f_{\max}$ are the smallest and biggest objective function values in the population, respectively; $G_{\min}$ and $G_{\max}$ are the smallest and biggest degree of constraint violation, respectively. Additionally, a parameter p_f $(0 \le p_f \le 1)$ is used to balance the constraints and the objective function. As described in Eq. (11), a bigger p_f will introduce more information of objective function while a smaller one will introduce more information of constraints. Thus, it should be tuned elaborately.

As discussed in Section 3.3, at the early stage, more information of the objective function is needed to explore the infeasible region. Thus, a relatively bigger p_f is needed. On the contrary, at the later stage, to converge into the feasible region and locate the optimum, the population needs more information of the constraints. Consequently, a relatively smaller p_f is beneficial. In summary, similar to Eq. (7), p_f is dynamically adjusted as follows:

$$p_f = \begin{cases} (1 - \frac{t}{T})^{cp}, & \text{if } t \le Tc \\ 10^{-50}, & \text{otherwise} \end{cases} \tag{14}$$

$$cp = -\frac{\lambda}{\log(1 - \frac{Tc}{T})}. \tag{15}$$

In fact, Eqs. (14) and (15) can be considered as the special case of Eqs. (7) and (8), where Δ_0 is set to 1. By this way, the search algorithm can be matched with IDFR. The tradeoff between the constraints and the objective function can be achieved. The effectiveness of these two equations is validated in the experimental section. In addition, two control parameters in DE, i.e., F and CR, are set in the same way as in [37,41]. In summary, the detials of the search algorithm are described in Algorithm 2.

Algorithm 2: Search algorithm.

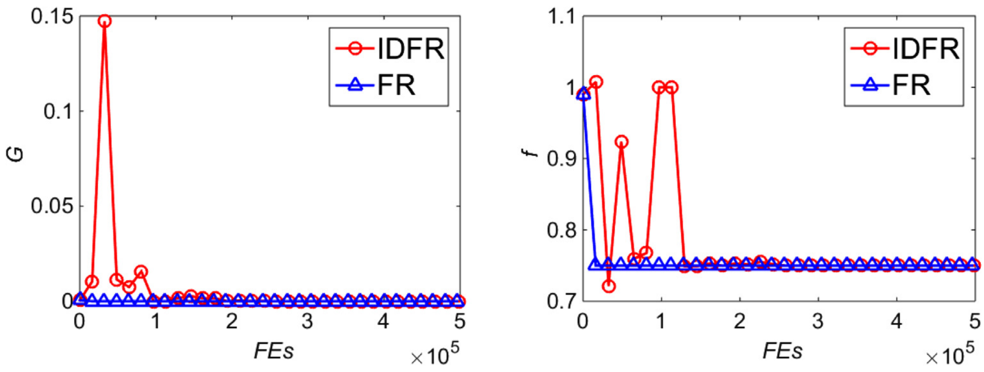
Set $OP = \varnothing$;
for $i = 1$ *to* P_S **do**
  $f_n(\vec{x}_i) = \frac{f(\vec{x}_i) - f_{\min}}{f_{\max} - f_{\min}}$;
  $G_n(\vec{x}_i) = \frac{G(\vec{x}_i) - G_{\min}}{G_{\max} - G_{\min}}$;
Adjust p_f according to Equations (14) and (15);
for $i = 1$ *to* P_S **do**
  $DTC(\vec{x}_i) = \sqrt{p_f \cdot f_n^2(\vec{x}_i) + (1 - p_f) \cdot G_n^2(\vec{x}_i)}$;
$best = \underset{i}{\operatorname{argmin}}\, DTC(\vec{x}_i)$;
for $i = 1$ *to* P_S **do**
  Randomly select a F value from the pool $\{0.6, 0.8, 1.0\}$;
  Randomly select a CR value from the pool $\{0.1, 0.2, 1.0\}$;
  if $rand < 0.5$ **then**
    Select $\vec{x}_{r_1}$, $\vec{x}_{r_2}$, and $\vec{x}_{r_3}$ from the population;
    $\vec{v}_i = \vec{x}_{r1} + F \cdot (\vec{x}_{best} - \vec{x}_{r1}) + F \cdot (\vec{x}_{r2} - \vec{x}_{r3})$;
    for $j = 1$ *to* D **do**
      $u_{i,j} = \begin{cases} v_{i,j}, & \text{if } rand < CR \text{ or } j = rand_j \\ x_{i,j}, & \text{otherwise} \end{cases}$;
  else
    Select $\vec{x}_{r_1}$, $\vec{x}_{r_2}$, and $\vec{x}_{r_3}$ from the population;
    $\vec{u}_i = \vec{x}_i + rand \cdot (\vec{x}_{r1} - \vec{x}_i) + F \cdot (\vec{x}_{r2} - \vec{x}_{r3})$;
  $OP = OP \cup \vec{u}_i$;

Remark 2. It is interesting to find that the search algorithm in this paper is similar to that of the algorithms in [37] and [41]. Similarly, "DE/current-to-rand/1" is used for exploration while DE operators which can use the information of the best solution are used for exploitation. The main difference lies in how to select the best solution. In [37], two DE operators are used for exploitation and the best solutions are selected based on $f(\vec{x})$ and $G(\vec{x})$, respectively. In [41], the best solution is selected according to $f(\vec{x})$. Whereas, in the proposed search algorithm, the best solution is selected based on $DTC(\vec{x})$.

Remark 3. IDFRDE has three key ingredients, those are IDFR, the diversity strategy, and the search algorithm. Although the diversity strategy and the search algorithm have some similarities to that of the algorithms in [37] and [41], the differences have been clarified in Remarks 1 and 2. Moreover, the main motivation of IDFRDE is to improve FR for constraint-handling, which is fundamentally different from that of the algorithms in [37] and [41]. The main aim of the algorithm in [37] is to extend a global algorithm to solve COPs. Its main emphasis is put on the search algorithm design. The aim of the algorithm in [41] is to introduce the information of objective function for constrained optimization. As described in Section 3.1, the information is incorporated at the population level. In fact, it is different from IDFR which uses the information from the individual perspective. Thus, the main contribution of this paper is different from that of [37] and [41]. In summary, IDFRDE is a new method but not a trivial combination of the algorithms in [37] and [41].

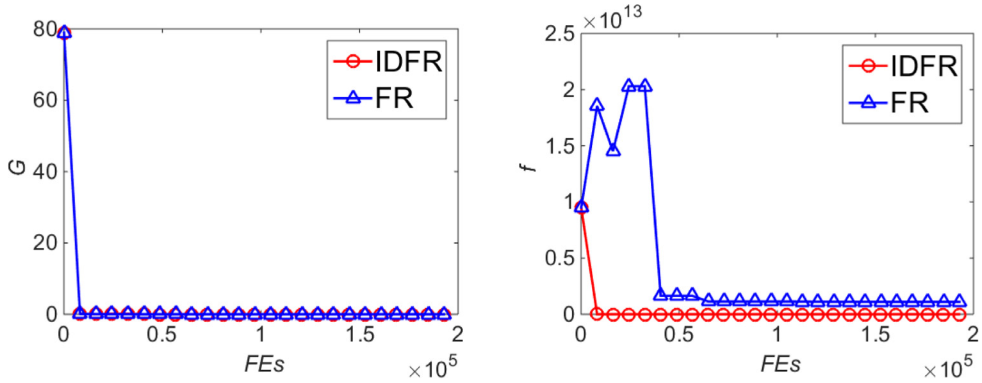
3.6. Computational time complexity analysis

As shown in Section 3.2, IDFRDE includes four main steps, those are, the initialization, IDFR, the two-phase diversity strategy, and the search algorithm. At the initialization stage, P_S solutions are generated. The computational time complexity is $O(P_S)$. In the IDFR, the pair-wise selection is conducted for P_S pairs of parents and offsprings. The computational time complexity is $O(P_S)$. As shown in Algorithm 1, in the mutation scheme, only one solution is mutated. The computational time complexity is $O(1)$. In the restart scheme, P_S solutions are regenerated. The computational time complexity is $O(P_S)$. As shown in Algorithm 2, the search algorithm consists of three single loops. Thus, its computational time complexity is $O(P_S)$. In summary, the computational time complexity of IDFRDE is $O(P_S)$. As described in [37], the computational time complexity of the composite constrained differential evolution (C^2oDE) is $O(P_S)$. As described in [41], the computational time complexity of FROFI is $O(P_S \log(P_S))$. Thus, the computational time complexity of IDFRDE is competitive against that of C^2oDE and FROFI.



(a) (b)

Fig. 2. Evolution of IDFRDE and FRDE on g11: (a) mean degree of constraint violation, (b) mean objective function value.



(a) (b)

Fig. 3. Evolution of IDFRDE and FRDE on C09 with 10D: (a) mean degree of constraint violation, (b) mean objective function value.

4. Experimental study

4.1. Proof-of-principle results

4.1.1. Significance of IDFR

First of all, we used two representative test functions (i.e., g11 from CEC2006 [11] and C09 with 10 dimensions (10D) from IEEE CEC2010 [15]) to test the significance of IDFR. To investigate the advantages and disadvantages of IDFR, we implemented a variant called FRDE where IDFR was replaced by the original FR. Note that other ingredients of FRDE were kept the same as that of IDFRDE. Then, we compared the performance of IDFRDE with that of FRDE on these two test functions. To investigate the efficiency of IDFRDE, we compared the computational time of IDFRDE with that of FROFI on these two test functions.

As shown in [11], g11 has a single-modal constraint. Thus, it is easy to find a feasible solution for g11. The evolution of IDFRDE and FRDE was described in Fig. 2, which describes the convergence curves of the best feasible solution.[1] Note that the convergence curves of FROFI are similar to those of IDFRDE. Thus, they were not given in the figure. As shown in Fig. 2, both IDFRDE and FRDE can find the optimum of g11. Additionally, FRDE converges faster than IDFRDE on this test function. The computational time of IDFRDE and FROFI on this test function is 7.92 s and 8.10 s, respectively.

As shown in [15], C09 includes a multi-modal equality constraint, which is more complex than that of g11. Thus, it is difficult to solve. Similarly, the evolution of IDFRDE and FRDE on this test function was described in Fig. 3. It can be observed that both IDFRDE and FRDE can find a feasible solution efficiently. IDFRDE can locate the feasible optimum finally. However, FRDE obtains a local optimum. In addition, the computational time of IDFRDE and FROFI on this test function is 3.92 s and 6.63 s, respectively.

The above experimental results show that:

[1] The best feasible solution is the best solution in the population based on the feasibility rule.

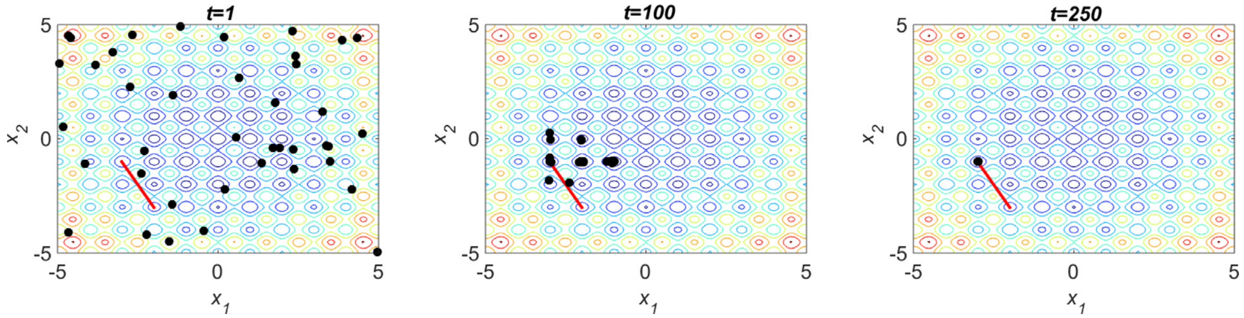


Fig. 4. The evolution of IDFRDE over a typical run on ATF1.

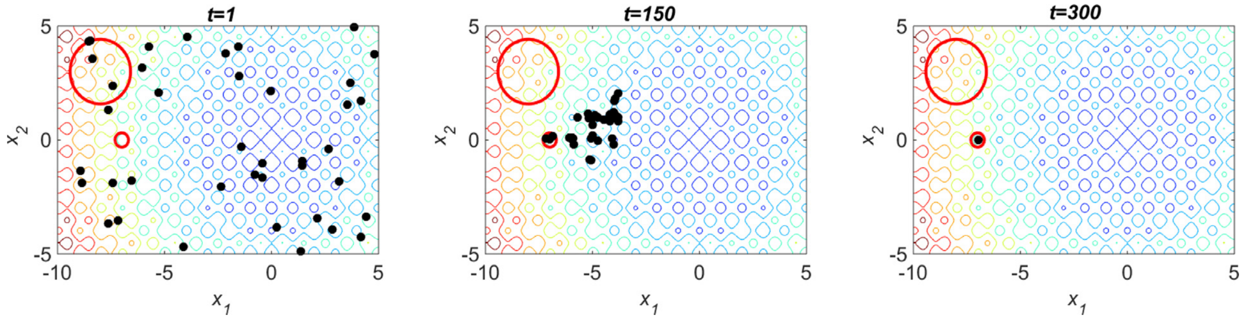


Fig. 5. The evolution of IDFRDE over a typical run on ATF2.

- For COPs with complicated constraints, the proposed method is better than the method with the feasibility rule. Because it can introduce more information of objective function, the proposed method can enter into the feasible region which contains the optimum. However, the feasibility rule would miss the region easily.
- For COPs with relatively simple constraints, the proposed method converges slower than the method with the feasibility rule. Because the feasibility rule prefers constraints. It is suitable to solve COPs owning relatively simple constraints. Whereas, the information of objective function would cause some disturbances.

The performance of IDFRDE and that of FRDE was further compared based on benchmark test functions in Section 4.7. Additionally, IDFRDE is better than FROFI in terms of the computational efficiency. Similarly, the computational efficiency was further analyzed based on benchmark test functions in Section 4.4.

4.1.2. Capability to solve challenging COPs

In this section, we tested the capability of IDFRDE to solve challenging COPs, which include COPs with disjoint feasible regions, COPs with equality constraints, and COPs with the optimum located on the boundary between feasible and infeasible regions. To this end, the performance of IDFRDE was evaluated on three artificial test functions (ATFs) designed in [41]. Specifically, ATF1 has a linear constraint, ATF2 has two disjoint feasible regions, and the optimum of ATF3 is located on the boundary between feasible and infeasible regions. To be noticed, the population size and the maximum number of function evaluations of IDFRDE were set to 40 and 40000, respectively. Besides, the other parameters were set as the same in Section 4.2.

The experimental results were described in Figs. 4–6, where t represents the generation number. As shown in Fig. 4, IDFRDE is able to approach the feasible optimum from diverse directions. As shown in Fig. 5, although some feasible solutions are initialized in the big feasible region, IDFRDE can enter into the small feasible region which contains the optimum. As shown in Fig. 6, IDFRDE has the ability to locate the optimum on the boundary accurately. In summary, IDFRDE is able to solve these challenging COPs successfully.

4.2. Benchmark test functions and parameter settings

To evaluate the performance of IDFRDE, three widely used benchmark sets were adopted, which contain 24 test functions from IEEE CEC2006, 18 test functions with 10D and 30D from IEEE CEC2010, and 28 test functions with 50D and 100D from IEEE CEC2017. The details of these test functions can be found in [11,15,43].

The population size P_S and the maximum function evaluations *MaxFEs* for different benchmark sets were described in Table 1. Additionally, according to [11,15,43], the number of independent runs and the tolerance value δ were set to 25 and

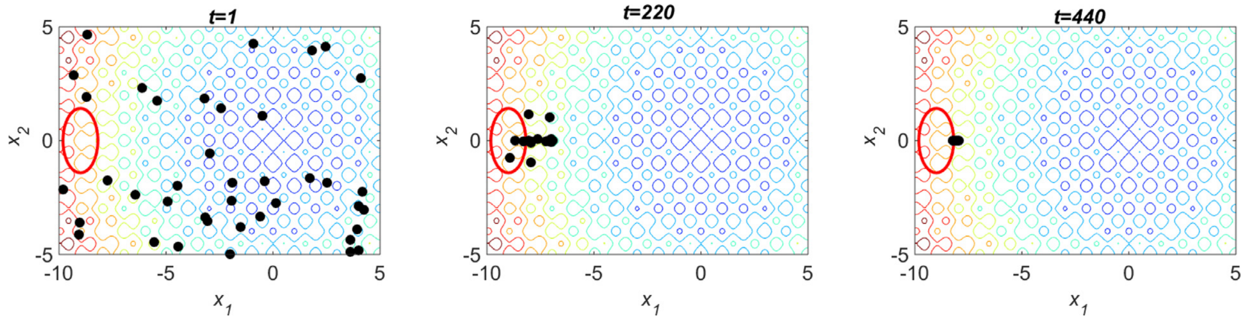


Fig. 6. The evolution of IDFRDE over a typical run on ATF3.

Table 1
Maximum number of function evaluations *MaxFEs* and population size P_S.

Test Functions	*MaxFEs*	P_S
24 test functions from IEEE CEC2006	5.0E+05	80
18 test functions with 10D from IEEE CEC2010	2.0E+05	80
18 test functions with 30D from IEEE CEC2010	6.0E+05	80
28 test functions with 50D from IEEE CEC2017	1.0E+06	100
28 test functions with 100D from IEEE CEC2017	2.0E+06	100

0.0001, respectively. They were kept the same in all compared algorithms. The algorithm-specific parameters, those are, Tc and μ were set to $0.5T$ and 10^{-8}, respectively.

Both the Wilcoxon's rank sum test and multi-problem tests were used to test the statistical significance [8]. Note that the multi-problem tests were implemented by the KEEL software [1].

4.3. Experiments on IEEE CEC2006 benchmark

In this subsection, 24 test functions from IEEE CEC2006 were used to evaluate the performance of IDFRDE. Subsequently, we compared it with five state-of-the-art COEAs which uses different constraint-handling techniques: the fuzzy rule-based penalty method (fpenalty) [29], the novel selection based evolutionary strategy (NSES) [9], the method based on dynamic weights (DW) [21], FROFI [41], and the adaptive penalty function based genetic algorithm (APF-GA) [34], whose experimental results were obtained from their original papers.

The experimental results of these six methods were collected in Table 2. In the table, "Mean OFV" and "Std Dev" denote the average value and standard deviation of the objective function values over 25 independent runs, respectively. As we know, the true optima of these 24 test functions are provided in [11]. Thus, a successful run can be defined as follows [11,44]. In a successful run, the best output obtained by a COEA (i.e., $\vec{x}_{best}$) satisfies that $f(\vec{x}_{best}) - f(\vec{x}^*) < 10^{-4}$, where $\vec{x}^*$ is the true optimum provided in [11]. Based on this definition, a COEA is successful on a test function if and only if it can be successful over 25 runs consistently. In this case, a "*" would be marked at the corresponding position in the table. Due to the fact that there are no feasible solutions for g20 and it is extremely difficult to locate a feasible solution for g22, these two test functions were excluded from comparisons. Note that fpenalty, NSES, and APF-GA do not provide the results of these two test functions. FROFI and DW cannot find a feasible solution for these two test functions. As shown in Table 2, NSES, FROFI, and IDFRDE are successful on 22 test functions. DW, fpenalty, and APF-GA fail to find a feasible solution consistently on one, seven, and eight test functions, respectively. In summary, IDFRDE can achieve competitive performance against other COEAs.

4.4. Experiments on IEEE CEC2010 benchmark

To further validate the performance of IDFRDE, 36 more complicated test functions from CEC2010 were adopted. In fact, these test functions include 18 test functions with 10D and 18 test functions with 30D. Since the optimal solutions of these test functions are not given in [15], all comparisons in this subsection were based on the average objective function values over 25 runs. Similarly, five state-of-the-art methods were selected as the competitors: the improved teaching-learning-based optimization (ITLBO) [36], FROFI [41], CACDE [44], AIS-IRP [45], and fpenalty [29]. The experimental results of these methods were obtained from their original papers. For each test function, we ranked IDFRDE and five competitors based on the average objective function value. Afterward, by suming the rank values of all test functions, the total rank value of a method was obtained. Additionally, the multi-problem Wilcoxon's test and the Friedman's test were implemented by the KEEL software [1] to test statistical significance.

In terms of the test functions with 10D, the experimental results were collected in Table 3, where the rank values are highlighted in bold. As shown in the table, the total rank value of IDFRDE is smaller than that of five competitors. Additionally, IDFRDE achieves the first rank on 11 test functions while other competitors cannot achieve the first rank on more than

Table 2
Experimental results of IDFRDE and other five selected methods over 25 independent runs on 22 test functions from IEEE CEC2006.

IEEE CEC2006	fpenalty Mean OFV ± Std Dev	NSES Mean OFV ± Std Dev	DW Mean OFV ± Std Dev	FROFI Mean OFV ± Std Dev	APF-GA Mean OFV ± Std Dev	IDFRDE Mean OFV ± Std Dev
g01	−1.50000E+01 ± 9.12E-06*	−1.5000E+01 ± 4.21E-30*	−1.5000E+01 ± 5.02E-14*	−1.5000E+01 ± 0.00E+00*	−1.5000E+01 ± 0.00E+00*	−1.5000E+01 ± 0.00E+00*
g02	−8.03560E-01 ± 6.32E-03	−8.0362E-01 ± 2.41E-32*	−8.0362E-01 ± 9.99E-08*	−8.0362E-01 ± 1.78E-07*	−8.0361E-01 ± 1.00E-04	−8.0362E-01 ± 3.12E-09*
g03	−1.0005E+00 ± 0.00E+00*	−1.0005E+00 ± 5.44E-19*	−1.0005E+00 ± 4.27E-12*	−1.0005E+00 ± 4.49E-16*	−1.0004E+00 ± 0.00E+00	−1.0005E+00 ± 4.60E-16*
g04	−3.06655E+04 ± 0.00E+00*	−3.0666E+04 ± 2.22E-24*	−3.066553E+04 ± 0.00E+00*	−3.066553E+04 ± 3.71E-12*	−3.066553E+04 ± 1.00E-04*	−3.066553E+04 ± 3.71E-12*
g05	5.126498E+03 ± 6.80E+00	5.1265E+03 ± 0.00E+00*	5.1264967E+03 ± 4.22E-10*	5.1264967E+03 ± 2.78E-12*	5.12754E+03 ± 1.43E+00	5.1265E+03 ± 2.78E-12*
g06	−6.96181E+03 ± 0.00E+00*	−6.9618E+03 ± 0.00E+00*	−6.961813E+03 ± 0.00E+00*	−6.961813E+03 ± 0.00E+00*	−6.961813E+03 ± 0.00E+00*	−6.961813E+03 ± 0.00E+00*
g07	2.43064E+01 ± 2.80E-03*	2.4306E+01 ± 7.37E-09*	2.430621E+01 ± 5.28E-10*	2.430621E+01 ± 6.32E-15*	2.430621E+01 ± 0.00E+00*	2.4306E+01 ± 6.61E-15*
g08	−9.58250E-02 ± 0.00E+00*	−9.5825E+02 ± 2.01E-34*	−9.5825E+02 ± 2.78E-18*	−9.5825E+02 ± 1.42E-17*	−9.5825E+02 ± 0.00E+00*	−9.5825E+02 ± 1.42E-17*
g09	6.80630E+02 ± 2.21E-08*	6.8063E+02 ± 1.10E-25*	6.8063006E+02 ± 2.23E-11*	6.8063006E+02 ± 2.23E-11*	6.8063006E+02 ± 0.00E+00*	6.8063006E+02 ± 2.94E-13*
g10	7.04917E+03 ± 6.19E+01*	7.0492480E+03 ± 2.07E-24*	7.0492480E+03 ± 4.43E-08*	7.0492480E+03 ± 3.26E-12*	7.077682E+03 ± 5.12E+01	7.0492480E+03 ± 3.62E-10*
g11	7.4990E-01 ± 0.00E+00*	7.499E-01 ± 0.00E+00*	7.499E-01 ± 1.06E-16*	7.499E-01 ± 1.13E-16*	7.499E-01 ± 0.00E+00*	7.499E-01 ± 1.13E-16*
g12	−1.00E+00 ± 0.00E+00*	−1.00E+00 ± 0.00E+00*	−1.00E+00 ± 0.00E+00*	−1.00E+00 ± 0.00E+00*	−1.00E+00 ± 0.00E+00*	−1.00E+00 ± 0.00E+00*
g13	5.39415E-02 ± 3.91E-12*	5.3942E-02 ± 1.98E-34*	5.3942E-02 ± 6.03E-14*	5.3942E-02 ± 2.41E-17*	5.4042E-02 ± 0.00E+00	5.3942E-02 ± 3.05E-17*
g14	−4.69325E+01 ± 2.30E-01	−4.776489E+01 ± 0.00E+00*	−4.776489E+01 ± 3.47E-10*	−4.776489E+01 ± 2.34E-14*	−4.776489E+01 ± 1.00E-04*	−4.776489E+01 ± 2.90E-14*
g15	9.61715E+02 ± 0.00E+00*	9.617150E+02 ± 0.00E+00*	9.617150E+02 ± 4.47E-13*	9.617150E+02 ± 5.80E-13*	9.617150E+02 ± 0.00E+00*	9.617150E+02 ± 5.80E-13*
g16	−1.90516E+00 ± 1.12E-10*	−1.90516E+00 ± 2.62E-30*	−1.90516E+00 ± 0.00E+00*	−1.90516E+00 ± 4.53E-16*	−1.90510E+00 ± 0.00E+00*	−1.90516E+00 ± 4.53E-16*
g17	8.8535397E+03 ± 2.50E+00	8.853533E+03 ± 2.51E-23*	8.880233E+03 ± 3.63E+01	8.853533E+03 ± 0.00E+00*	8.888481E+03 ± 2.90E+01	8.853533E+03 ± 8.30E-13*
g18	−8.66023E-01 ± 2.13E-07*	−8.66025E-01 ± 4.62E-33*	−8.66025E-01 ± 3.30E-07*	−8.66025E-01 ± 6.94E-16*	−8.66025E-01 ± 0.00E+00*	−8.66025E-01 ± 3.81E-13*
g19	3.27001E+01 ± 3.20E-04	3.265559E+01 ± 1.52E-05*	3.265559E+01 ± 3.37E-07*	3.265559E+01 ± 2.18E-14*	3.265559E+01 ± 0.00E+00*	3.265559E+01 ± 5.43E-15*
g21	1.93681E+02 ± 2.50E+01	1.937245E+02 ± 1.62E-22*	1.937245E+02 ± 3.66E-09*	1.937245E+02 ± 2.95E-11*	1.99516E+02 ± 2.35E+00	1.937245E+02 ± 4.82E-10*
g23	−3.81924E+02 ± 8.01E+01	−4.000551E+02 ± 9.08E-26*	−4.000551E+02 ± 6.49E-06*	−4.000551E+02 ± 1.71E-13*	−3.947627E+02 ± 3.87E+00	−4.000551E+02 ± 1.66E-05*
g24	−5.5080E+00 ± 0.00E+00*	−5.50801E+00 ± 0.00E+00*	−5.50801E+00 ± 0.00E+00*	−5.50801E+00 ± 9.06E-16*	−5.50801E+00 ± 0.00E+00*	−5.50801E+00 ± 9.06E-16*
/	**15**	**22**	**21**	**22**	**14**	**22**

Table 3
Experimental results of IDFRDE and other five selected methods over 25 independent runs on the 18 test functions with 10D from IEEE CEC2010.

IEEE CEC2010 with 10D	ITLBO Mean OFV ± Std Dev (rank)	FROFI Mean OFV ± Std Dev (rank)	CACDE Mean OFV ± Std Dev (rank)	AIS-IRP Mean OFV ± Std Dev (rank)	fpenalty Mean OFV ± Std Dev (rank)	IDFRDE Mean OFV ± Std Dev (rank)
C01	−7.47E-01 ± 1.87E-03(**1**)	−7.47E-01 ± 1.35E-03(**1**)	−7.47E-01 ± 1.88E-03 (**1**)	−7.47E-01 ± 1.30E-03(**1**)	−7.47E-01 ± 5.46E-03 (**1**)	−7.47E-01 ± 1.87E-03(**1**)
C02	−2.03E+00 ± 8.14E-02(**4**)	−2.02E+00 ± 1.41E-01(**5**)	−2.26E+00 ± 6.57E-02(**2**)	−2.27E+00 ± 2.00E-03(**1**)	−1.83E+00 ± 9.93E+00 (**6**)	−2.25E+00 ± 5.48E-02(**3**)
C03	0.00E+00 ± 0.00E+00(**1**)	0.00E+00 ± 0.00E+00(**1**)	0.00E+00 ± 0.00E+00 (**1**)	3.75E-09 ± 4.81E-04(**5**)	2.51E+00 ± 7.47E+00 (**6**)	0.00E+00 ± 0.00E+00(**1**)
C04	−1.00E-05 ± 3.39E-15(**1**)	−1.00E-05 ± 0.00E+00(**1**)	−1.00E-05 ± 0.00E+00(**1**)	−9.97E-06 ± 4.28E-03(**5**)	−8.85E-06 ± 8.67E-03(**6**)	−1.00E-05 ± 7.64E-13(**1**)
C05	−4.84E+02 ± 1.11E-11(**1**)	−4.84E+02 ± 8.09E-07(**1**)	−4.84E+02 ± 3.48E-13(**1**)	−4.80E+02 ± 6.30E+00(**6**)	−4.84E+02 ± 3.43E+01(**1**)	−4.84E+02 ± 3.03E-13(**1**)
C06	−5.79E+02 ± 2.39E-04(**2**)	−5.79E+02 ± 5.04E-04(**2**)	−5.79E+02 ± 1.68E-02 (**2**)	−5.80E+02 ± 7.30E-08(**1**)	−5.03E+02 ± 5.91E+01 (**6**)	−5.79E+02 ± 5.02E-07(**2**)
C07	0.00E+00 ± 0.00E+00(**1**)	0.00E+00 ± 0.00E+00(**1**)	0.00E+00 ± 0.00E+00(**1**)	1.17E-08 ± 2.70E+00(**5**)	3.79E+00 ± 2.14E+00(**6**)	0.00E+00 ± 0.00E+00(**1**)
C08	8.47E+00 ± 4.09E+00(**6**)	7.11E+00 ± 4.79E+00(**3**)	7.01E+00 ± 5.01E+00(**4**)	4.09E+00 ± 1.46E+00(**2**)	3.25E-02 ± 9.38E+05(**1**)	8.34E+00 ± 4.47E+00(**5**)
C09	0.00E+00 ± 0.00E+00(**1**)	2.50E+01 ± 3.92E+01(**4**)	2.10E+01 ± 3.51E+01(**3**)	2.70E+01 ± 7.50E+01(**5**)	4.32E+01 ± 8.45E+04(**6**)	0.00E+00 ± 0.00E+00(**1**)
C10	1.92E-01 ± 9.62E-01(**2**)	4.17E+01 ± 8.69E-06(**3**)	6.59E+01 ± 4.40E+01(**4**)	1.62E+03 ± 5.00E+02(**5**)	5.33E+04 ± 3.43E+02(**6**)	0.00E+00 ± 0.00E+00(**1**)
C11	−1.51E-03 ± 1.30E-05(**4**)	−1.52E-03 ± 5.63E-14(**1**)	−1.52E-03 ± 1.30E-06(**1**)	−9.20E-04 ± 8.23E-04(**5**)	2.54E+00 ± 7.53E+00(**6**)	−1.52E-03 ± 5.07E-11(**1**)
C12	−2.39E+01 ± 1.14E+02(**5**)	−3.84E+02 ± 2.17E+02(**4**)	−4.34E+02 ± 2.49E+02(**2**)	−4.36E+02 ± 6.02E+01(**1**)	−4.24E+02 ± 5.43E+02(**3**)	−4.24E-01 ± 2.90E+00(**6**)
C13	−6.52E+01 ± 1.78E+00(**6**)	−6.84E+01 ± 2.52E-09(**1**)	−6.72E+01 ± 1.04E+00(**4**)	−6.79E+01 ± 3.11E-01(**3**)	−6.68E+01 ± 9.28E+00(**5**)	−6.84E+01 ± 2.90E-14(**1**)
C14	0.00E+00 ± 0.00E+00(**1**)	0.00E+00 ± 0.00E+00(**1**)	0.00E+00 ± 0.00E+00(**1**)	1.22E-04 ± 2.90E-08(**5**)	4.39E+04 ± 4.82E+05(**6**)	0.00E+00 ± 0.00E+00(**1**)
C15	3.54E+00 ± 4.97E+00(**5**)	3.09E+00 ± 1.37E+00(**3**)	3.38E+00 ± 1.02E+00 (**4**)	5.19E-09 ± 1.10E-08(**1**)	3.03E+13 ± 8.69E+03(**6**)	2.94E-01 ± 1.02E+00(**2**)
C16	2.27E-01 ± 3.11E-01(**5**)	1.19E-02 ± 2.07E-02(**3**)	4.52E-02 ± 1.03E-01(**4**)	9.96E-18 ± 6.27E-15(**1**)	2.20E+00 ± 5.11E-01(**6**)	1.11E-02 ± 1.92E-02(**2**)
C17	3.91E-01 ± 6.71E-01(**5**)	7.83E-02 ± 2.25E-01(**3**)	1.23E-33 ± 2.52E-33(**1**)	2.93E+00 ± 2.29E+00(**6**)	3.75E-01 ± 9.72E+01(**4**)	1.44E-20 ± 1.37E-20(**2**)
C18	0.00E+00 ± 0.00E+00(**1**)	5.23E-26 ± 1.71E-25(**4**)	0.00E+00 ± 0.00E+00(**1**)	1.66E+00 ± 1.27E+00(**6**)	5.35E-03 ± 1.78E+01(**5**)	0.00E+00 ± 0.00E+00(**1**)
Total Rank	**52**	**42**	**38**	**64**	**86**	**33**

Table 4
Results of the multiple-problem Wilcoxon's test for IDFRDE and other five selected methods on the 18 test functions with 10D from IEEE CEC2010.

Algorithm	R^+	R^-	p-value	$\alpha = 0.1$	$\alpha = 0.05$
ITLBO	142.0	29.0	1.2032E-02	Yes	Yes
FROFI	108.0	45.0	1.4544E-01	No	No
CACDE	103.5	67.5	≥ 0.2	No	No
AIS-IRP	121.0	32.0	3.4800E-02	Yes	Yes
fpenalty	157.0	14.0	8.3920E-04	Yes	Yes

Table 5
Ranking of IDFRDE and other five selected methods by the Friedman's Test on the 18 test functions with 10D from IEEE CEC2010.

Algorithm	Ranking
IDFRDE	**2.6111**
CACDE	2.9167
FROFI	3.0833
ITLBO	3.6667
AIS-IRP	3.6944
fpenalty	5.0278

10 test functions. The results of multi-problem Wilcoxon's test and Friedman's test were summarized in Tables 4 and 5, respectively. In Table 4, all R^+ values are larger than R^- values. A significance level of $\alpha = 0.05$ can be achieved in three cases. Besides, IDFRDE ranks the first place in the Friedman's test. All experimental results reflect that IDFRDE is more competitive than its competitors on the test functions with 10D. To test the computational efficiency of IDFRDE, its computational time was compared with that of FROFI. The computational time provided by IDFRDE for the 18 test functions with 10D over 25 runs is 911.61 s. The corresponding computational time of FROFI is 1335.46 s. Thus, IDFRDE is more efficient than FROFI on these test functions.

The experimental results of the test functions with 30D were summarized in Table 6. Since fpenalty was not evaluated on these test functions in its original paper, a constrained particle swarm optimization (Co-CLPSO) [12] was used instead. In addition, when a method cannot find a feasible solution over 25 runs consistently, a "∇" was marked. As shown in the table, IDFRDE obtains the smallest total rank value among all methods. What's more, IDFRDE can achieve the first rank on 13 test functions. Similarly, the results of multi-problem Wilcoxon's test and Friedman's test were summarized in Tables 7 and 8, respectively. In Table 7, all R^+ values are larger than R^- values. Besides, a significance level of $\alpha = 0.05$ can be achieved in four cases. In the Friedman's test, IDFRDE achieves the first rank. In summary, IDFRDE is better than five competitors on the test functions with 30D. Additionally, it seems that the advantage of IDFRDE on 30D test functions is more significant. The reason may be as follows. In the high-dimensional space, the landscape of the constraints of these test functions might be much more complex than that of the objective functions. Since IDFRDE can introduce the information of objective function, the population can explore the infeasible solution adequately. Thus, IDFRDE can locate the feasible region which contains the optimum easily. As a result, a better feasible solution can be found. Note that the computational time of IDFRDE and FROFI on the test functions with 30D is 3348.82 s and 5034.29 s, respectively. Thus, IDFRDE is more efficient than FROFI.

4.5. Experiments on IEEE CEC2017 benchmark

In order to test IDFRDE's performance on high-dimensional COPs, we further evaluated it on the 28 test functions with 50D and 100D from IEEE CEC2017. Next, we compared it with LSHADE44 [22] and UDE [35], which rank the first place and the second place in the IEEE CEC2017 competition, respectively. The experimental results were recorded in Table 9, where "*voi*" and "*FR*" represent the average degree of constraint violation and feasible rate, respectively. We first ranked IDFRDE, LSHADE44, and UDE on each test function according to the procedure provided in [43]. Afterward, the total rank of each algorithm was calculated. To test the statistical significance, the multi-problem Wilcoxon's test and the Friedman's test were implemented by the KEEL software. The results were summarized in Tables 10 and 11.

As shown in Table 9, IDFRDE achieves the lowest rank on the 28 test functions with both 50D and 100D. As shown in Table 10, all R^+ values are larger than R^- values. Besides, a significance level of $\alpha = 0.05$ can be achieved in all cases. As shown in Table 11, IDFRDE ranks the first place. All experimental results show that IDFRDE is able to solve high-dimensional COPs.

4.6. Convergence analysis

The convergence performance of IDFRDE was further investigated on the 18 test functions with 30D from IEEE CEC2010. For each test function, we recorded the degree of constraint violation G and objective function value f of the best feasible

Table 6
Experimental results of IDFRDE and other five selected methods over 25 independent runs on the 18 test functions with 30D from IEEE CEC2010.

IEEE CEC2010 with 30D	ITLBO Mean OFV ± Std Dev (rank)	FROFI Mean OFV ± Std Dev (rank)	CACDE Mean OFV ± Std Dev (rank)	AIS-IRP Mean OFV ± Std Dev (rank)	Co-CLPSO Mean OFV ± Std Dev (rank)	IDFRDE Mean OFV ± Std Dev (rank)
C01	−8.20E-01 ± 8.95E-04(**2**)	−8.21E-01 ± 2.36E-03(**1**)	−8.20E-01 ± 2.67E-03(**2**)	−8.20E-01 ± 3.25E-04(**2**)	−7.16E-01 ± 5.03E-02(**6**)	−8.19E-01 ± 2.66E-03(**5**)
C02	−2.03E+00 ± 7.64E-02(**4**)	−2.00E+00 ± 4.35E-02(**6**)	−2.01E+00 ± 7.78E-02(**5**)	−2.21E+00 ± 2.84E-03(**2**)	−2.20E+00 ± 1.93E-01(**3**)	−2.27E+00 ± 2.31E-02(**1**)
C03	7.84E+01 ± 6.31E+01(**5**)	2.87E+01 ± 6.24E-08(**2**)	3.08E+01 ± 3.50E+01(**3**)	6.68E+01 ± 4.26E+02(**4**)	3.51E+01 ± 3.31E+01▽(**6**)	0.00E+00 ± 0.00E+00(**1**)
C04	1.69E-03 ± 1.14E-03(**3**)	−3.33E-06 ± 4.13E-10(**1**)	3.54E+00 ± 7.62E+00(**5**)	1.98E-03 ± 1.61E-03(**4**)	1.13E-01 ± 5.63E-01▽(**6**)	−3.32E-06 ± 2.52E-08(**2**)
C05	−4.82E+02 ± 1.73E+00(**2**)	−4.81E+02 ± 2.84E+00(**3**)	−3.41E+02 ± 8.69E+01(**5**)	−4.36E+02 ± 2.51E+01(**4**)	−3.12E+02 ± 8.83E+01(**6**)	−4.84E+02 ± 7.02E-11(**1**)
C06	−5.30E+02 ± 4.80E-01(**2**)	−5.29E+02 ± 5.71E-01(**3**)	−5.22E+02 ± 2.92E+00(**4**)	−4.54E+02 ± 4.79E+01(**5**)	−2.45E+02 ± 3.95E+01(**6**)	−5.31E+02 ± 1.10E-02(**1**)
C07	1.59E-01 ± 7.97E-01(**3**)	0.00E+00 ± 0.00E+00(**1**)	9.57E-01 ± 1.74E+00(**4**)	1.07E+00 ± 1.61E+00(**5**)	1.12E+00 ± 1.83E+00(**6**)	0.00E+00 ± 0.00E+00(**1**)
C08	1.14E+01 ± 2.79E+01(**5**)	0.00E+00 ± 0.00E+00(**1**)	9.76E+00 ± 3.20E+01(**4**)	1.65E+00 ± 6.41E-01(**3**)	4.75E+01 ± 1.13E+02(**6**)	0.00E+00 ± 0.00E+00(**1**)
C09	2.86E+00 ± 1.43E+01(**3**)	4.30E+01 ± 3.27E+01(**4**)	9.23E+03 ± 1.26E+04(**5**)	1.57E+00 ± 1.96E+00(**2**)	1.48E+08 ± 2.45E+08(**6**)	3.08E-01 ± 1.54E+00(**1**)
C10	3.29E+01 ± 1.41E+01(**2**)	3.13E+01 ± 8.22E-02(**3**)	8.20E+10 ± 3.91E+11(**5**)	1.78E+01 ± 1.88E+01(**1**)	1.40E+09 ± 5.84E+09(**6**)	3.13E+01 ± 1.76E-01(**3**)
C11	−3.86E-04 ± 1.14E-05(**3**)	−3.92E-04 ± 2.64E-06(**1**)	2.99E-03 ± 7.14E-03(**5**)	−1.58E-04 ± 4.67E-05(**4**)	2.82E-02 ± 3.21E-02(**6**)	−3.92E-04 ± 3.65E-09(**1**)
C12	−1.98E-01 ± 2.39E-03(**4**)	−1.99E-01 ± 1.42E-06(**1**)	−1.99E-01 ± 2.35E-04(**1**)	4.29E-06 ± 4.52E-04(**5**)	−1.99E-01 ± 1.18E-04▽(**6**)	−1.99E-01 ± 2.12E-04(**1**)
C13	−5.05E+01 ± 1.18E+00(**6**)	−6.83E+01 ± 1.95E-01(**1**)	−6.77E+01 ± 6.88E-01(**2**)	−6.62E+01 ± 2.27E-01(**4**)	−6.08E+01 ± 1.12E+00(**5**)	−6.63E+01 ± 3.00E+00(**3**)
C14	4.78E-01 ± 1.32E+00(**5**)	9.80E-29 ± 4.90E-28(**2**)	7.37E-26 ± 1.79E-25(**3**)	8.68E-07 ± 3.14E-07(**4**)	1.28E+00 ± 1.90E+00(**6**)	0.00E+00 ± 0.00E+00(**1**)
C15	2.38E+01 ± 2.51E+01(**4**)	2.16E+01 ± 8.03E-05(**1**)	2.17E+01 ± 2.45E-01 (**2**)	3.41E+01 ± 3.82E+01(**5**)	5.11E+01 ± 9.18E+01(**6**)	2.21E+01 ± 1.58E+00(**3**)
C16	0.00E+00 ± 0.00E+00(**1**)	0.00E+00 ± 0.00E+00(**1**)	6.03E-04 ± 3.02E-03(**5**)	8.21E-02 ± 1.12E-01(**6**)	5.24E-16 ± 4.67E-16(**4**)	0.00E+00 ± 0.00E+00(**1**)
C17	9.65E-01 ± 1.73E+00(**4**)	1.59E-01 ± 3.82E-01(**2**)	8.24E-01 ± 6.85E-01(**3**)	3.61E+00 ± 2.54E+00(**6**)	1.39E+00 ± 4.26E+00(**5**)	7.46E-02 ± 2.62E-01(**1**)
C18	9.07E-17 ± 3.18E-16(**2**)	4.87E-01 ± 1.25E+00(**4**)	2.35E-05 ± 8.46E-05(**3**)	4.02E+01 ± 1.80E+01(**6**)	1.09E+01 ± 3.72E+01(**5**)	2.28E-29 ± 7.14E-29(**1**)
Total Rank	**60**	**38**	**66**	**72**	**100**	**29**

Table 7
Results of the multiple-problem Wilcoxon's test for IDFRDE and other five selected methods on the 18 test functions with 30D from IEEE CEC2010.

Algorithm	R^+	R^-	p-value	$\alpha = 0.1$	$\alpha = 0.05$
ITLBO	151.0	20.0	2.8080E-03	Yes	Yes
FROFI	96.0	57.0	$\geq$ 0.2	No	No
CACDE	141.5	11.5	9.5370E-04	Yes	Yes
AIS-IRP	155.5	15.5	1.1673E-03	Yes	Yes
Co-CLPSO	171.0	0.0	7.6300E-06	Yes	Yes

Table 8
Ranking of IDFRDE and other five selected methods by the Friedman's test on the 18 test functions with 30D from IEEE CEC2010.

Algorithm	Ranking
IDFRDE	**1.8889**
FROFI	2.3333
ITLBO	3.3333
CACDE	3.8889
AIS-IRP	4.0556
fpenalty	5.5000

solution at each generation. Additionally, the experiment on each test function was repeated 25 times. For each generation, the mean G and f were calculated, which were then plotted in the convergence graphs. We used ITLBO and FROFI for comparison because we can obtain their source codes easily.

The convergence curves of IDFRDE, ITLBO, and FROFI were described in Figures S1-S18 in the supplementary file. Since the true optima are not provided in [15], f^* means the minimum objective function value found by IDFRDE, ITLBO, and FROFI. It can be observed that IDFRDE converges faster than FROFI and ITLBO on most of the test functions in terms of f. However, with regard to G, FROFI converges faster than IDFRDE on most of the test functions. Because much information of objective function is used at the early stage, IDFRDE cannot find a feasible solution quickly. However, it can locate a satisfied feasible solution finally.

4.7. Comparing individual-dependent feasibility rule with the original feasibility rule

To validate the advantages of IDFR over the original FR, we compared the performance of IDFRDE with that of FRDE. Test functions from IEEE CEC2010 were used to evaluate FRDE. We compared their performance by the Wilcoxon's rank-sum test at a 0.05 significance level. The corresponding results were collected in Table 12. When a method cannot find a feasible solution consistently, the feasible rate, i.e., the percentage of runs where at least one feasible solution is found, was recorded. Additionally, "−", "+", and "≈" represent that the compared method performs worse than, better than, and similarly to IDFRDE, respectively. As shown in the table, IDFRDE performs better than FRDE on 22 test functions. However, FRDE is better than IDFRDE on only two test functions. In summary, IDFR would be better than FR for constraint-handling.

4.8. Effectiveness of the two-phase diversity strategy

As described in Section 3.4, the aim of the two-phase diversity strategy is to handle COPs with complex constraints. To validate its effectiveness experimentally, we implemented three variants (i.e., IDFRDE-WoD, IDFRDE-WoM, and IDFRDE-WoR). In IDFRDE-WoD, both the mutation scheme and the diversity scheme were removed. In IDFRDE-WoM and IDFRDE-WoR, the mutation scheme and the restart scheme were removed, respectively. Subsequently, these three variants were evaluated on the 36 test functions from IEEE CEC2010. We compared their performance with that of IDFRDE by the Wilcoxon's rank-sum test at a 0.05 significance level. Experimental results with a significant difference were listed in Table 13. As shown in the table, IDFRDE-WoD cannot find a feasible solution consistently on four test functions, those are, C11 with 10D, C12 with 10D, C11 with 30D, and C17 with 30D. To be specific, IDFRDE-WoD cannot find a feasible solution on these test functions over 11, 25, one, and two runs, respectively. IDFRDE-WoM cannot find a feasible solution consistently on C11 with 30D and it performs worse than IDFRDE on C17 with 30D. IDFRDE-WoR cannot achieve a feasible solution on C12 with 10D and C11 with 30D, respectively. In addition, it is worse than IDFRDE on C17 with 30D. IDFRDE is able to achieve the best results on these four test functions. The experimental results imply that the two-phase diversity strategy can improve the performance of IDFRDE for handling complex constraints. Additionally, both the mutation scheme and the restart scheme are significant to the diversity strategy.

Table 9
Experimental results of IDFRDE, LSHADE44, and UDE over 25 independent runs on 56 test functions from IEEE CEC2017.

Instance	50D			100D		
	LSHADE44 Mean OFV/*voi*/*FR* (rank)	UDE Mean OFV/*voi*/*FR* (rank)	IDFRDE Mean OFV/*voi*/*FR* (rank)	LSHADE44 Mean OFV/*voi*/*FR* (rank)	UDE Mean OFV/*voi*/*FR* (rank)	IDFRDE Mean OFV/*voi*/*FR* (rank)
C01	7.79E-29/0.00E+00/1(**1**)	3.18E-11/0.00E+00/1(**1**)	4.64E-22/0.00E+00/1(**1**)	1.03E-25/0.00E+00/1(**1**)	1.79E-03/0.00E+00/1(**3**)	2.87E-10/0.00E+00/1(**1**)
C02	9.79E-29/0.00E+00/1(**1**)	1.60E-11/0.00E+00/1(**1**)	7.98E-22/0.00E+00/1(**1**)	8.47E-26/0.00E+00/1(**1**)	1.56E-03/0.00E+00/1(**3**)	1.89E-10/0.00E+00/1(**1**)
C03	8.95E+05/0.00E+00/1(**3**)	1.09E+02/0.00E+00/1(**2**)	4.33E-21/0.00E+00/1(**1**)	2.73E+06/0.00E+00/1(**3**)	7.42E+02/0.00E+00/1(**2**)	4.68E-10/0.00E+00/1(**1**)
C04	1.36E+01/0.00E+00/1(**1**)	1.47E+02/0.00E+00/1(**3**)	1.77E+01/0.00E+00/1(**2**)	1.37E+01/0.00E+00/1(**1**)	4.01E+02/0.00E+00/1(**3**)	9.89E+01/0.00E+00/1(**2**)
C05	1.68E-28/0.00E+00/1(**1**)	1.34E+01/0.00E+00/1(**3**)	1.59E-01/0.00E+00/1(**2**)	3.28E-05/0.00E+00/1(**1**)	7.54E+01/0.00E+00/1(**3**)	3.85E-01/0.00E+00/1(**2**)
C06	7.51E+03/1.17E-02/0(**3**)	7.43E+02/0.00E+00/1(**2**)	1.37E+02/0.00E+00/1(**1**)	1.56E+04/9.81E-03/0(**3**)	2.53E+03/4.28E-06/0.96(**2**)	3.32E+02/0.00E+00/1(**1**)
C07	−1.79E+02/0.00E+00/1(**3**)	−9.78E+02/0.00E+00/1(**2**)	−1.83E+03/0.00E+00/1(**1**)	−3.02E+02/0.00E+00/1(**3**)	−1.64E+03/0.00E+00/1(**2**)	−3.02E+03/0.00E+00/1(**1**)
C08	−1.30E-04/0.00E+00/1(**1**)	1.45E-04/0.00E+00/1(**3**)	−1.34E-04/0.00E+00/1(**1**)	−4.81E-05/0.00E+00/1(**1**)	2.97E-03/4.16E-06/0.92(**3**)	1.60E-03/0.00E+00/1(**2**)
C09	−2.04E-03/0.00E+00/1(**1**)	−2.04E-03/0.00E+00/1(**1**)	−2.04E-03/0.00E+00/1(**1**)	−1.43E-03/0.00E+00/1(**1**)	2.46E-01/2.52E-24/0.84(**3**)	0.00E+00/0.00E+00/1(**2**)
C10	−4.83E-05/0.00E+00/1(**1**)	3.04E-05/0.00E+00/1(**3**)	−4.81E-05/0.00E+00/1(**2**)	−1.72E-05/0.00E+00/1(**1**)	5.57E-04/0.00E+00/1(**3**)	3.49E-04/0.00E+00/1(**2**)
C11	−1.76E+00/0.00E+00/1(**1**)	−1.77E+02/4.36E-01/0(**3**)	−4.39E+01/1.57E-01/0(**2**)	−3.65E+00/5.25E-43/0.88(**1**)	−1.84E+02/2.03E-01/0(**2**)	−6.38E+03/1.39E+02/0(**3**)
C12	4.98E+01/0.00E+00/1(**3**)	2.09E+01/0.00E+00/1(**2**)	5.43E+00/0.00E+00/1(**1**)	3.25E+01/0.00E+00/1(**3**)	1.07E+01/0.00E+00/1(**2**)	3.98E+00/0.00E+00/1(**1**)
C13	2.67E+01/0.00E+00/1(**2**)	1.12E+03/0.00E+00/1(**3**)	7.73E-24/0.00E+00/1/1(**1**)	8.07E+01/0.00E+00/1(**2**)	3.38E+04/2.69E+01/0(**3**)	4.69E+01/0.00E+00/1(**1**)
C14	1.40E+00/0.00E+00/1(**3**)	1.23E+00/0.00E+00/1(**2**)	1.10E+00/0.00E+00/1(**1**)	9.72E-01/0.00E+00/1(**3**)	9.14E-01/0.00E+00/1(**2**)	7.84E-01/0.00E+00/1(**1**)
C15	1.78E+01/0.00E+00/1(**3**)	1.05E+01/0.00E+00/1(**2**)	2.36E+00/0.00E+00/1(**1**)	1.81E+01/0.00E+00/1(**3**)	1.80E+01/0.00E+00/1(**2**)	2.36E+00/0.00E+00/1(**1**)
C16	2.53E+02/0.00E+00/1(**3**)	1.21E+01/0.00E+00/1(**2**)	0.00E+00/0.00E+00/1(**1**)	5.35E+02/0.00E+00/1(**3**)	3.37E+01/0.00E+00/1(**2**)	5.03E-01/0.00E+00/1(**1**)
C17	1.03E+00/2.55E+01/0(**1**)	1.05E+00/2.55E+01/0(**1**)	4.08E-01/2.51E+01/0(**1**)	1.09E+00/5.05E+01/0(**1**)	1.10E+00/5.05E+01/0(**1**)	2.27E-01/5.05E+01/0(**1**)
C18	5.67E+03/2.24E+05/0(**2**)	4.06E+03/6.32E+07/0(**3**)	4.88E+02/1.85E+03/0(**1**)	3.44E+03/3.34E+06/0(**2**)	8.33E+03/1.37E+08/0(**3**)	1.00E+03/3.94E+03/0(**1**)
C19	1.21E-05/3.61E+04/0(**1**)	4.66E+00/3.61E+04/0(**1**)	9.39E-04/3.61E+04/0(**1**)	4.68E-05/7.30E+04/0(**1**)	3.25E+01/7.30E+04/0(**1**)	1.57E-03/7.30E+04/0(**1**)
C20	3.20E+00/0.00E+00/1(**1**)	7.59E+00/0.00E+00/1(**3**)	3.88E+00/0.00E+00/1(**2**)	9.36E+00/0.00E+00/1(**2**)	1.89E+01/0.00E+00/1(**3**)	8.21E+00/0.00E+00/1(**1**)
C21	6.29E+01/0.00E+00/1(**3**)	6.43E+00/0.00E+00/1(**1**)	1.01E+01/0.00E+00/1(**2**)	3.16E+01/0.00E+00/1(**3**)	1.48E+01/0.00E+00/1(**2**)	1.37E+01/0.00E+00/1(**1**)
C22	8.39E+03/1.01E-02/0.96(**2**)	2.90E+03/6.74E-02/0.84(**3**)	1.01E+01/0.00E+00/1(**1**)	5.04E+04/6.46E+00/0.04(**2**)	5.58E+04/4.27E+02/0(**3**)	1.44E+05/4.45E+01/0.4(**1**)
C23	1.34E+00/0.00E+00/1(**3**)	1.10E+00/0.00E+00/1(**1**)	1.10E+00/0.00E+00/1(**1**)	9.69E-01/0.00E+00/1(**3**)	7.85E-01/0.00E+00/1(**1**)	7.86E-01/0.00E+00/1(**2**)
C24	1.43E+01/0.00E+00/1(**3**)	1.13E+01/0.00E+00/1(**2**)	2.36E+00/0.00E+00/1(**1**)	1.72E+01/0.00E+00/1(**2**)	1.81E+01/0.00E+00/1(**3**)	6.13E+00/0.00E+00/1(**1**)
C25	2.49E+02/0.00E+00/1(**3**)	2.24E+01/0.00E+00/1(**2**)	4.09E-14/0.00E+00/1(**1**)	5.44E+02/0.00E+00/1(**3**)	1.65E+02/0.00E+00/1(**2**)	5.69E+01/0.00E+00/1(**1**)
C26	1.04E+00/2.55E+01/0(**1**)	1.05E+00/2.55E+01/0(**1**)	8.60E-01/2.55E+01/0(**1**)	1.10E+00/5.05E+01/0(**1**)	1.10E+00/5.05E+01/0(**1**)	9.94E-01/5.05E+01/0(**1**)
C27	2.17E+04/1.34E+07/0(**2**)	1.04E+04/2.58E+08/0(**3**)	5.59E+02/3.60E+03/0(**1**)	3.69E+04/4.78E+08/0(**2**)	4.22E+04/2.03E+09/0(**3**)	1.16E+03/1.87E+04/0(**1**)
C28	2.65E+02/3.63E+04/0(**1**)	1.25E+02/3.63E+04/0(**1**)	9.32E+01/3.62E+04/0(**1**)	5.84E+02/7.34E+04/0(**1**)	3.20E+02/7.33E+04/0(**1**)	2.91E+02/7.33E+04/0(**1**)
Total Rank	**54**	**57**	**34**	**54**	**64**	**36**

Table 10
Results of the multiple-problem Wilcoxon's test for IDFRDE and other two selected methods on 56 test functions from IEEE CEC2017

Algorithm	R^+	R^-	p-value	$\alpha = 0.1$	$\alpha = 0.05$
SHADE44	1067.0	473.0	1.2681E-02	Yes	Yes
UDE	1405.5	134.5	0	Yes	Yes

Table 11
Ranking of IDFRDE and other two selected methods by the Friedman's test on 56 test functions from IEEE CEC2017.

Algorithm	Ranking
IDFRDE	**1.5625**
SHADE44	2.0982
UDE	2.3393

Table 12
Experimental results of IDFRDE and FRDE over 25 independent runs on 36 test functions from IEEE CEC2010.

Instance	10D		30D	
	IDFRDE Mean OFV ± Std Dev (feasible rate)	FRDE Mean OFV ± Std Dev (feasible rate)	IDFRDE Mean OFV ± Std Dev (feasible rate)	FRDE Mean OFV ± Std Dev (feasible rate)
C01	−7.47E-01 ± 1.87E-03	−7.47E-01 ± 2.24E-03 ≈	−8.19E-01 ± 2.66E-03	−8.18E-01 ± 3.97E-03 ≈
C02	−2.25E+00 ± 5.48E-02	5.78E-01 ± 1.65E+00−	−2.27E+00 ± 2.31E-02	2.53E+00 ± 7.38E-01−
C03	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00 ≈	0.00E+00 ± 0.00E+00	2.96E+11 ± 1.41E+12−
C04	−1.00E-05 ± 7.64E-13	−1.00E-05 ± 0.00E+00 ≈	−3.32E-06 ± 2.52E-08	2.60E-02 ± 1.30E-01−
C05	−4.84E+02 ± 3.03E-13	1.59E+02 ± 1.41E+02−	−4.84E+02 ± 7.02E-11	2.90E+02 ± 9.87E+01−
C06	−5.79E+02 ± 5.02E-07	1.35E+02 ± 1.88E+02−	−5.31E+02 ± 1.10E-02	3.60E+02 ± 1.07E+02−
C07	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00 ≈	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00 ≈
C08	8.34E+00 ± 4.47E+00	7.80E+00 ± 4.58E+00 ≈	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00 ≈
C09	0.00E+00 ± 0.00E+00	3.60E+12 ± 4.48E+12−	3.08E-01 ± 1.54E+00	9.00E+12 ± 9.55E+12−
C10	0.00E+00 ± 0.00E+00	1.55E+12 ± 2.19E+12−	3.13E+01 ± 1.76E-01	9.04E+12 ± 5.16E+12−
C11	−1.52E-03 ± 5.07E-11	−1.52E-03 ± 3.34E-18 ≈	−3.92E-04 ± 3.65E-09	−3.92E-04 ± 1.16E-10 ≈
C12	−4.24E-01 ± 2.90E+00	−2.46E+02 ± 2.42E+02+	−1.99E-01 ± 2.12E-04	−1.99E-01 ± 1.33E-05 ≈
C13	−6.84E+01 ± 2.90E-14	−6.84E+01 ± 3.04E-14 ≈	−6.63E+01 ± 3.00E+00	−6.73E+01 ± 1.19E+00+
C14	0.00E+00 ± 0.00E+00	4.67E+08 ± 1.57E+09−	0.00E+00 ± 0.00E+00	3.98E+01 ± 1.99E+02−
C15	2.94E-01 ± 1.02E+00	3.92E+13 ± 5.06E+13−	2.21E+01 ± 1.58E+00	1.07E+14 ± 7.79E+13 −
C16	1.11E-02 ± 1.92E-02	4.81E-01 ± 1.91E-01−	5.91E-04 ± 2.95E-03	9.82E-01 ± 1.11E-01−
C17	1.44E-20 ± 1.37E-20	7.53E+01 ± 3.99E+01−	7.46E-02 ± 2.62E-01	3.65E+02 ± 2.01E+02−
C18	0.00E+00 ± 0.00E+00	2.75E+03 ± 2.58E+03−	2.28E-29 ± 7.14E-29	8.29E+03 ± 3.06E+03 −
−	/	**10**	/	**12**
+	/	**1**	/	**1**
≈	/	**7**	/	**5**

Table 13
Experimental results of IDFRDE, IDFRDE-WoD, IDFRDE-WoM, and IDFRDE-WoR over 25 independent runs on four test functions.

Instance	IDFRDE Mean OFV ± Std Dev (feasible rate)	IDFRDE-WoD Mean OFV ± Std Dev (feasible rate)	IDFRDE-WoM Mean OFV ± Std Dev (feasible rate)	IDFRDE-WoR Mean OFV ± Std Dev (feasible rate)
C11 with 10D	−1.52E-03 ± 5.07E-11	56%	−1.52E-03 ± 7.94E-09	1.52E-03 ± 7.09E-11
C12 with 10D	−4.24E-01 ± 2.90E+00	0%	−5.69E+00 ± 2.29E+01	0%
C11 with 30D	−3.92E-04 ± 3.65E-09	96%	92%	96%
C17 with 30D	7.46E-02 ± 2.62E-01	92%	3.14E+02 ± 6.71E+02	2.31E-01 ± 8.55E-01

4.9. Effectiveness of the search algorithm

As indicated in Section 3.5, the search algorithm plays a significant role in IDFRDE. To analyze the advantages of the search algorithm, we implemented six variants (i.e., IDFRDE-CC, IDFRDE-CO, IDFRDE-Con, IDFRDE-Obj, IDFRDE-Div, and IDFRDE-rand1). In IDFRDE-CC and IDFRDE-CO, both "DE/rand-to-best/1/bin" and "DE/current-to-rand/1" were used. Besides, the "best" was selected according to $G(\vec{x})$ in IDFRDE-CC while the "best" was selected according to $f(\vec{x})$ in IDFRDE-CO. In IDFRDE-Con and IDFRDE-Obj, only "DE/rand-to-best/1/bin" was used. Besides, the "best" was selected according to $G(\vec{x})$ in IDFRDE-Con while the "best" was selected according to $f(\vec{x})$ in IDFRDE-Obj. In IDFRDE-Div, only "DE/current-to-rand/1" was used. In IDFRDE-rand1, "DE/rand/1/bin" was used as the search algorithm. All these variants were evaluated on the 18 test functions with 30D from IEEE CEC2010 and the experimental results were summarized in Table 14. IDFRDE reveals better

Table 14
Experimental results of IDFRDE and other six variants with different search algorithms over 25 independent runs on the 18 test functions with 30D from IEEE CEC2010.

Test Functions	IDFRDE-CC Mean OFV ± Std Dev (feasible rate)	IDFRDE-CO Mean OFV ± Std Dev (feasible rate)	IDFRDE-Con Mean OFV ± Std Dev (feasible rate)	IDFRDE-Obj Mean OFV ± Std Dev (feasible rate)	IDFRDE-Div Mean OFV ± Std Dev (feasible rate)	IDFRDE-rand1 Mean OFV ± Std Dev (feasible rate)	IDFRDE Mean OFV ± Std Dev (feasible rate)
C01	−8.18E-01 ± 3.76E-03 ≈	−8.20E-01 ± 2.04E-03 ≈	−8.18E-01 ± 3.87E-03 ≈	−8.20E-01 ± 1.82E-03 ≈	−6.61E-01 ± 1.30E-01−	−8.22E-01 ± 7.87E-04 ≈	−8.19E-01 ± 2.66E-03
C02	−2.27E+00 ± 1.83E-02 ≈	−2.27E+00 ± 9.52E-03 ≈	−2.28E+00 ± 3.64E-03 ≈	−2.28E+00 ± 3.95E-03 ≈	−2.23E+00 ± 3.68E-02−	−2.21E+00 ± 2.47E-02−	−2.27E+00 ± 2.31E-02
C03	3.49E-26 ± 1.74E-25−	0.00E+00 ± 0.00E+00 ≈	1.58E+01 ± 1.61E+01−	96%−	0%−	1.26E+02 ± 4.79E+01−	0.00E+00 ± 0.00E+00
C04	−3.31E-06 ± 2.21E-08 ≈	−3.31E-06 ± 2.96E-08 ≈	−3.33E-06 ± 0.00E+00 ≈	24%−	0%−	5.12E-03 ± 8.22E-03 −	−3.32E-06 ± 2.52E-08
C05	−4.54E+02 ± 7.22E+01−	−4.84E+02 ± 3.07E-10 ≈	−2.51E+02 ± 9.31E+00−	−4.84E+02 ± 5.01E-13 ≈	76%−	88%−	−4.84E+02 ± 7.02E-11
C06	−5.31E+02 ± 5.73E-02 ≈	−5.31E+02 ± 7.34E-04 ≈	−2.41E+02 ± 5.51E+01−	−5.31E+02 ± 3.85E-06 ≈	92%−	−2.67E+02 ± 4.66E+02−	−5.31E+02 ± 1.10E-02
C07	1.59E-01 ± 7.97E-01−	3.19E-01 ± 1.10E+00−	7.97E-01 ± 1.63E+00−	1.59E-01 ± 7.97E-01−	2.75E+06 ± 6.10E+06−	1.51E-01 ± 2.86E-01−	0.00E+00 ± 0.00E+00
C08	1.26E+01 ± 3.53E+01−	1.22E+01 ± 3.06E+01−	1.78E+01 ± 5.03E+01−	6.38E-01 ± 1.49E+00−	4.52E+06 ± 8.52E+06−	4.52E-02 ± 4.12E-02−	0.00E+00 ± 0.00E+00
C09	9.06E+00 ± 4.53E+01−	0.00E+00 ± 0.00E+00+	8.80E-01 ± 1.80E+00 ≈	1.26E-26 ± 1.71E-26+	1.65E+03 ± 4.37E+03 −	1.14E+02 ± 3.44E+01−	3.08E-01 ± 1.54E+00
C10	3.13E+01 ± 6.17E-06 ≈	4.17E+00 ± 8.40E+00+	3.13E+01 ± 6.91E-06 ≈	4.24E-01 ± 1.51E+00+	6.00E+03 ± 1.06E+04−	3.31E+01 ± 1.57E+00 ≈	3.13E+01 ± 1.76E-01
C11	−3.92E-04 ± 1.05E-09 ≈	80%−	−3.92E-04 ± 6.74E-10 ≈	0%−	0%−	0%−	−3.92E-04 ± 3.65E-09
C12	−1.99E-01 ± 1.18E-06 ≈	84%−	−1.99E-01 ± 3.80E-07 ≈	0%−	0% −	52%−	−1.99E-01 ± 2.12E-04
C13	−6.77E+01 ± 1.56E+00+	−6.66E+01 ± 2.71E+00 ≈	−6.81E+01 ± 6.88E-01+	−6.51E+01 ± 2.63E+00−	−4.17E+01 ± 7.89E+00−	−6.40E+01 ± 9.82E-01−	−6.63E+01 ± 3.00E+00
C14	0.00E+00 ± 0.00E+00 ≈	3.35E+00 ± 1.16E+01−	1.59E-01 ± 7.97E-01−	6.58E+01 ± 2.46E+02−	1.96E+08 ± 4.26E+08−	1.51E-01 ± 2.37E-01−	0.00E+00 ± 0.00E+00
C15	2.18E+01 ± 1.14E+00 ≈	1.92E+01 ± 7.34E+00+	2.16E+01 ± 2.31E-07 ≈	1.81E+01 ± 8.08E+00+	9.24E+03 ± 3.44E+04−	2.16E+01 ± 7.14E-07 ≈	2.21E+01 ± 1.58E+00
e C16	0.00E+00 ± 0.00E+00 ≈	0.00E+00 ± 0.00E+00 ≈	1.21E-02 ± 4.20E-02−	0.00E+00 ± 0.00E+00 ≈	1.05E-02 ± 1.53E-02−	2.32E-03 ± 8.02E-03 −	0.00E+00 ± 0.00E+00
C17	1.13E-02 ± 5.65E-02 ≈	3.54E+00 ± 8.81E+00−	8.57E+01 ± 4.28E+02−	2.14E+01 ± 7.82E+01−	3.98E+01 ± 7.18E+01−	1.87E-02 ± 3.69E-02 ≈	7.46E-02 ± 2.62E-01
C18	8.60E-22 ± 4.30E-21−	2.46E+00 ± 9.44E+00−	6.17E+00 ± 2.48E+01−	1.27E-01 ± 3.35E-01−	2.34E-02 ± 5.55E-02−	4.30E-01 ± 2.01E+00−	2.28E-29 ± 7.14E-29
−	**6**	**7**	**9**	**10**	**18**	**14**	/
+	**1**	**3**	**1**	**3**	**0**	**0**	/
≈	**11**	**8**	**8**	**5**	**0**	**4**	/

Table 15
Experimental results of IDFRDE and IDFRDE-linear over 25 independent runs on four test functions.

Instance	IDFRDE Mean OFV ± Std Dev (feasible rate)	IDFRDE-linear Mean OFV ± Std Dev (feasible rate)
C12 with 10D	−4.24E-01 ± 2.90E+00	48%
C17 with 10D	1.44E-20 ± 1.37E-20	2.09E-02 ± 1.04E-01
C17 with 30D	7.46E-02 ± 2.62E-01	9.31E-01 ± 3.41E+00
C18 with 30D	2.28E-29 ± 7.14E-29	2.46E-08 ± 1.15E-07

Table 16
Experimental results of IDFRDE with five varying *Tc* over 25 independent runs on the 18 test functions with 30D from IEEE CEC2010.

IEEE CEC2010 with 30D	$Tc = 0.1T$ Mean OFV ± Std Dev (feasible rate)	$Tc = 0.3T$ Mean OFV ± Std Dev (feasible rate)	$Tc = 0.7T$ Mean OFV ± Std Dev (feasible rate)	$Tc = 0.9T$ Mean OFV ± Std Dev (feasible rate)	$Tc = 0.5T$ Mean OFV ± Std Dev (feasible rate)
C01	−8.19E-01 ± 2.61E-03 ≈	−8.20E-01 ± 2.34E-03 ≈	−8.19E-01 ± 2.43E-03 ≈	−8.20E-01 ± 2.67E-03 ≈	−8.19E-01 ± 2.66E-03
C02	−2.06E+00 ± 1.20E-01−	−2.25E+00 ± 3.47E-02 ≈	−2.28E+00 ± 1.10E-02 ≈	−2.28E+00 ± 2.25E-03 ≈	−2.27E+00 ± 2.31E-02
C03	2.87E+01 ± 4.96E-10−	1.15E+00 ± 5.73E+00−	0.00E+00 ± 0.00E+00 ≈	6.98E-26 ± 2.42E-25−	0.00E+00 ± 0.00E+00
C04	2.60E-02 ± 1.30E-01−	−3.33E-06 ± 2.94E-10 ≈	2.75E-06 ± 8.24E-06−	1.31E-03 ± 6.22E-04−	−3.32E-06 ± 2.52E-08
C05	−2.07E+02 ± 1.28E+02−	−4.84E+02 ± 5.08E-04 ≈	−4.84E+02 ± 3.54E-08 ≈	−4.84E+02 ± 1.24E-04 ≈	−4.84E+02 ± 7.02E-11
C06	−2.31E+02 ± 9.27E+01−	−5.31E+02 ± 3.28E-02 ≈	−5.31E+02 ± 1.81E-02 ≈	−5.31E+02 ± 2.06E-02 ≈	−5.31E+02 ± 1.10E-02
C07	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
C08	9.29E+00 ± 3.16E+01−	1.57E+01 ± 3.66E+01−	1.84E+01 ± 4.44E+01−	9.92E+00 ± 3.64E+01−	0.00E+00 ± 0.00E+00
C09	2.88E+05 ± 7.66E+05−	0.00E+00 ± 0.00E+00 ≈	0.00E+00 ± 0.00E+00 ≈	0.00E+00 ± 0.00E+00 ≈	3.08E-01 ± 1.54E+00
C10	4.37E+10 ± 2.19E+11−	3.13E+01 ± 2.99E-06 ≈	3.13E+01 ± 2.01E-05 ≈	2.76E+01 ± 7.10E+00 ≈	3.13E+01 ± 1.76E-01
C11	−3.92E-04 ± 5.75E-09 ≈	−3.92E-04 ± 3.27E-10 ≈	96%−	0%−	−3.92E-04 ± 3.65E-09
C12	−1.99E-01 ± 1.11E-09 ≈	−1.99E-01 ± 1.15E-08 ≈	5.76E-03 ± 1.02E+00−	0%−	−1.99E-01 ± 2.12E-04
C13	−6.74E+01 ± 1.68E+00+	−6.72E+01 ± 1.83E+00+	−6.65E+01 ± 2.31E+00 ≈	−6.62E+01 ± 1.89E+00 ≈	−6.63E+01 ± 3.00E+00
C14	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
C15	7.04E+08 ± 2.40E+09 −	2.16E+01 ± 1.59E-07 ≈	2.16E+01 ± 1.30E-07 ≈	2.18E+01 ± 1.14E+00 ≈	2.21E+01 ± 1.58E+00
C16	0.00E+00 ± 0.00E+00 ≈	0.00E+00 ± 0.00E+00 ≈	8.00E-04 ± 4.00E-03 −	6.78E-04 ± 3.39E-03 −	0.00E+00 ± 0.00E+00
C17	1.52E+01 ± 7.61E+01−	1.11E+00 ± 5.18E+00−	9.24E-03 ± 4.62E-02 ≈	1.48E+00 ± 4.12E+00−	7.46E-02 ± 2.62E-01
C18	6.93E-02 ± 3.47E-01−	2.33E-26 ± 1.16E-25−	2.08E-21 ± 8.89E-21−	3.97E-10 ± 1.78E-09 −	2.28E-29 ± 7.14E-29
−	**11**	**4**	**6**	**8**	/
+	**1**	**1**	**0**	**0**	/
≈	**6**	**13**	**12**	**10**	/

results than IDFRDE-CC, IDFRDE-CO, IDFRDE-Con, IDFRDE-Obj, IDFRDE-Div, and IDFRDE-rand1 on six, seven, nine, 10, 18, and 14 test functions, respectively. However, all these variants cannot perform better than IDFRDE on more than three test functions. By comparing IDFRDE with IDFRDE-CC and IDFRDE-CO, we can find that properly utilizing $f(\vec{x})/G(\vec{x})$ is critical to a search algorithm. That is to say, the tradeoff between constraints and objective function is critical. By comparing IDFRDE with IDFRDE-Con, IDFRDE-Obj, DeCODE-Div, and IDFRDE-rand1, we can find that the tradeoff between diversity and convergence is also crucial to a search algorithm.

To investigate the effectiveness of Eqs. (14) and (15), we implemented a variant (i.e., IDFRDE-linear), where p_f is set as $p_f = 1 - \frac{t}{T}$. Both IDFRDE and IDFRDE-linear were evaluated on 36 test functions from IEEE CEC2010. The test functions on which two algorithms have a significant difference were summarized in Table 15. As shown in the table, IDFRDE performs better than IDFRDE-linear on four test functions. Moreover, IDFRDE-linear cannot find a feasible solution consistently on C12 with 10D. As we know, locating a feasible solution is critical to a COEA. Thus, Eqs. (14) and (15) are significant to the search algorithm.

4.10. Parameter sensitivity analysis

As shown in Section 3.3, parameter *Tc* decides how much information of objective function is used. A bigger *Tc* means that more information of objective function would be used. As described in Section 3.3, using the information of objective function properly is critical to constrained optimization. Thus, *Tc* would have a significant impact on the performance of IDFRDE. In this section, the parameter sensitivity of *Tc* was investigated experimentally. To this end, five variants with different *Tc* ($Tc = 0.1T$, $Tc = 0.3T$, $Tc = 0.5T$, $Tc = 0.7T$, and $Tc = 0.9T$) were implemented. Note that in the original IDFRDE, *Tc* is equal to 0.5*T*. All these variants were evaluated on the 18 test functions with 30D from IEEE CEC2010 and the experimental results were collected in Table 16. As shown in the table, IDFRDE with $Tc = 0.5T$ performs better than those with $Tc = 0.1T$, $Tc = 0.3T$, $Tc = 0.7T$, and $Tc = 0.9T$ on 11, four, six, and eight test functions, respectively. Inversely, IDFRDE with $Tc = 0.1T$, $Tc = 0.3T$, $Tc = 0.7T$, and $Tc = 0.9T$ reveal better results than that with $Tc = 0.5T$ on one, one, zero, and zero test function, respectively. Moreover, the Friedman's test in Table 17 shows that IDFRDE with $Tc = 0.5T$ ranks the first place. In summary, $Tc = 0.5T$ was recommended in this paper.

Table 17
Ranking of IDFRDE wiht five varying Tc by the Friedman's test on the 18 test functions with 30D from IEEE CEC2010.

Algorithm	Ranking
$Tc = 0.5T$	**2.2500**
$Tc = 0.3T$	2.5556
$Tc = 0.7T$	3.0556
$Tc = 0.9T$	3.3333
$Tc = 0.1T$	3.8056

Table 18
Experimental results of IDFRDE with five varying μ over 25 independent runs on the 18 test functions with 30D from IEEE CEC2010.

IEEE CEC2010 with 30D	$\mu = 10^{-4}$ Mean OFV ± Std Dev (feasible rate)	$\mu = 10^{-6}$ Mean OFV ± Std Dev (feasible rate)	$\mu = 10^{-10}$ Mean OFV ± Std Dev (feasible rate)	$\mu = 10^{-12}$ Mean OFV ± Std Dev (feasible rate)	$\mu = 10^{-8}$ Mean OFV ± Std Dev (feasible rate)
C01	−8.19E-01 ± 2.30E-03 ≈	−8.19E-01 ± 2.71E-03 ≈	−8.19E-01 ± 3.40E-03 ≈	−8.19E-01 ± 3.55E-03 ≈	−8.19E-01 ± 2.66E-03
C02	−2.27E+00 ± 1.37E-02 ≈	−2.27E+00 ± 1.89E-02 ≈	−2.27E+00 ± 1.78E-02 ≈	−2.27E+00 ± 1.40E-02 ≈	−2.27E+00 ± 2.31E-02
C03	0.00E+00 ± 0.00E+00 ≈	0.00E+00 ± 0.00E+00 ≈	3.49E-26 ± 1.74E-25−	0.00E+00 ± 0.00E+00 ≈	0.00E+00 ± 0.00E+00
C04	80%−	−3.32E-06 ± 2.08E-08 ≈	−3.32E-06 ± 1.15E-08 ≈	−3.32E-06 ± 1.26E-08 ≈	−3.32E-06 ± 2.52E-08
C05	−4.84E+02 ± 1.64E-10 ≈	−4.84E+02 ± 6.08E-11 ≈	−4.84E+02 ± 1.10E-10 ≈	−4.84E+02 ± 1.29E-10 ≈	−4.84E+02 ± 7.02E-11
C06	−5.31E+02 ± 7.06E-03 ≈	−5.31E+02 ± 4.65E-03 ≈	−5.31E+02 ± 5.77E-03 ≈	−5.31E+02 ± 4.22E-03 ≈	−5.31E+02 ± 1.10E-02
C07	0.00E+00 ± 0.00E+00 ≈	1.37E-27 ± 6.86E-27 −	0.00E+00 ± 0.00E+00 ≈	0.00E+00 ± 0.00E+00 ≈	0.00E+00 ± 0.00E+00
C08	2.33E+01 ± 6.01E+01−	4.03E+00 ± 1.93E+01−	1.13E+01 ± 4.57E+01−	2.08E+01 ± 3.86E+01−	0.00E+00 ± 0.00E+00
C09	3.11E+00 ± 1.56E+01−	1.76E-01 ± 8.80E-01 ≈	5.28E-01 ± 1.46E+00 ≈	1.76E-01 ± 8.80E-01 ≈	3.08E-01 ± 1.54E+00
C10	3.13E+01 ± 1.76E-01 ≈	3.13E+01 ± 5.60E-06 ≈	3.13E+01 ± 2.84E-06 ≈	3.13E+01 ± 1.17E-05 ≈	3.13E+01 ± 1.76E-01
C11	0%−	92%−	92%−	92%−	−3.92E-04 ± 3.65E-09
C12	32%−	−1.99E-01 ± 1.58E-07 ≈	−1.99E-01 ± 1.33E-08 ≈	−1.99E-01 ± 2.68E-07 ≈	−1.99E-01 ± 2.12E-04
C13	−6.62E+01 ± 2.73E+00 ≈	−6.65E+01 ± 2.12E+00 ≈	−6.60E+01 ± 3.57E+00 ≈	−6.63E+01 ± 3.38E+00 ≈	−6.63E+01 ± 3.00E+00
C14	0.00E+00 ± 0.00E+00 ≈	0.00E+00 ± 0.00E+00 ≈	0.00E+00 ± 0.00E+00 ≈	5.57E-29 ± 2.78E-28−	0.00E+00 ± 0.00E+00
C15	2.07E+01 ± 4.32E+00 ≈	2.16E+01 ± 1.34E-07 ≈	2.18E+01 ± 1.14E+00 ≈	2.16E+01 ± 1.01E-07 ≈	2.21E+01 ± 1.58E+00
C16	0.00E+00 ± 0.00E+00+	0.00E+00 ± 0.00E+00+	6.09E-04 ± 3.04E-03 ≈	1.05E-03 ± 3.64E-03 ≈	5.91E-04 ± 2.95E-03
C17	7.25E-02 ± 3.63E-01 ≈	1.68E-01 ± 7.62E-01 ≈	9.63E-02 ± 4.41E-01 ≈	1.72E-01 ± 5.54E-01 ≈	7.46E-02 ± 2.62E-01
C18	1.52E-06 ± 4.60E-06−	1.55E-10 ± 5.71E-10−	1.19E-28 ± 5.22E-28 ≈	8.36E-30 ± 1.88E-29 ≈	2.28E-29 ± 7.14E-29
−	**6**	**4**	**3**	**3**	/
+	**1**	**1**	**0**	**0**	/
≈	**11**	**13**	**15**	**15**	/

Table 19
Ranking of IDFRDE wiht five varying μ by the Friedman's test on the 18 test functions with 30D from IEEE CEC2010.

Algorithm	Ranking
$\mu = 10^{-8}$	**2.6111**
$\mu = 10^{-6}$	2.8889
$\mu = 10^{-10}$	2.9722
$\mu = 10^{-12}$	3.0278
$\mu = 10^{-4}$	3.5000

As described in Section 3.4, in the diversity strategy, parameter μ decides which phase to be executed. It would have a significant effect on the diversity strategy. To select it elaborately, five variants with different μ ($\mu = 10^{-4}$, $\mu = 10^{-6}$, $\mu = 10^{-8}$, $\mu = 10^{-10}$, and $\mu = 10^{-12}$) were implemented. Note that $\mu = 10^{-8}$ is used in the original IDFRDE. Similarly, the 18 test functions with 30D from IEEE CEC2010 were used to evaluate these variants. All experimental results were summarized in Table 18. IDFRDE with $\mu = 10^{-8}$ performs better than those with $\mu = 10^{-4}$, $\mu = 10^{-6}$, $\mu = 10^{-10}$, and $\mu = 10^{-12}$ on six, four, three, and three test functions, respectively. Inversely, IDFRDE with $\mu = 10^{-4}$, $\mu = 10^{-6}$, $\mu = 10^{-10}$, and $\mu = 10^{-12}$ reveal better results than that with $\mu = 10^{-8}$ on one, one, zero, and zero test function, respectively. Moreover, the Friedman's test in Table 19 shows that IDFRDE with $\mu = 10^{-8}$ ranks the first place. In summary, $\mu = 10^{-8}$ was recommended in this paper.

5. Conclusions and future directions

This paper tried to address the greedy property of the feasibility rule. For each individual, some information of constraints, which may be nonsignificant, was depressed. In addition, some promising information of objective function was leveraged. The extent of information depressed/leveraged is individual-dependent. To handle complex constraints, a two-phase diversity strategy was developed. Finally, by designing a DE-based search algorithm, we proposed a constrained DE, i.e., IDFRDE. Extensive and systematic experiments verified that:

- The individual-dependent feasibility rule outperforms the original feasibility rule for constraint-handling.
- The two-phase diversity strategy improves IDFRDE's ability to find feasible solutions for COPs with complicated constraints.
- The proposed search algorithm is effective to solve constrained optimization problems.
- IDFRDE shows superior performance against the selected competitors on 24 test functions from IEEE CEC2006, 36 test functions from IEEE CEC2010, and 56 test functions from IEEE CEC2017.

In the future, we would like to extend IDFRDE to solve constrained multiobjective optimization problems. We will also use surrogate models to enhance IDFRDE for solving expensive constrained optimization problems. Moreover, we will reduce the algorithm-specific parameters.

Declaration of Competing Interest

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

References

[1] J. Alcalá-Fdez, L. Sánchez, S. Garcia, M.J. del Jesus, S. Ventura, J.M. Garrell, J. Otero, C. Romero, J. Bacardit, V.M. Rivas, et al., Keel: a software tool to assess evolutionary algorithms for data mining problems, Soft Comput. 13 (3) (2009) 307–318.
[2] T. Back, Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms, Oxford university press, 1996.
[3] C.A.C. Coello, Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art, Comput. Methods Appl. Mech. Eng. 191 (11–12) (2002) 1245–1287.
[4] C.A.C. Coello, Constraint-handling techniques used with evolutionary algorithms, in: Proceedings of the Genetic and Evolutionary Computation Conference Companion, ACM, 2018, pp. 773–799.
[5] S. Das, S.S. Mullick, P.N. Suganthan, Recent advances in differential evolution–an updated survey, Swarm Evol. Comput. 27 (2016) 1–30.
[6] S. Das, P.N. Suganthan, Differential evolution: a survey of the state-of-the-art, IEEE Trans. Evol. Comput. 15 (1) (2011) 4–31.
[7] K. Deb, An efficient constraint handling method for genetic algorithms, Comput. Methods Appl. Mech. Eng. 186 (2–4) (2000) 311–338.
[8] J. Derrac, S. García, D. Molina, F. Herrera, A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms, Swarm Evol. Comput. 1 (1) (2011) 3–18.
[9] L. Jiao, L. Li, R. Shang, F. Liu, R. Stolkin, A novel selection evolutionary strategy for constrained optimization, Inf. Sci. 239 (2013) 122–141.
[10] R. Kicinger, T. Arciszewski, K. De Jong, Evolutionary computation and structural design: a survey of the state-of-the-art, Comput. Struct. 83 (23–24) (2005) 1943–1978.
[11] J. Liang, T.P. Runarsson, E. Mezura-Montes, M. Clerc, P.N. Suganthan, C.C. Coello, K. Deb, Problem definitions and evaluation criteria for the CEC 2006 special session on constrained real-parameter optimization, J. Appl. Mech. 41 (8) (2006) 8–31.
[12] J.J. Liang, S. Zhigang, L. Zhihui, Coevolutionary comprehensive learning particle swarm optimizer, in: Evolutionary Computation (CEC), 2010 IEEE Congress on, IEEE, 2010, pp. 1–8.
[13] J. Liu, K.L. Teo, X. Wang, C. Wu, An exact penalty function-based differential search algorithm for constrained global optimization, Soft Comput. 20 (4) (2016) 1305–1313.
[14] R. Mallipeddi, P.N. Suganthan, Ensemble of constraint handling techniques, IEEE Trans. Evol. Comput. 14 (4) (2010) 561–579.
[15] R. Mallipeddi, P.N. Suganthan, Problem Definitions and Evaluation Criteria for the CEC 2010 Competition on Constrained Real-Parameter Optimization, Nanyang Technological University, Singapore, 2010.
[16] E. Mezura-Montes, C.A.C. Coello, Constraint-handling in nature-inspired numerical optimization: past, present and future, Swarm Evol. Comput. 1 (4) (2011) 173–194.
[17] Z. Michalewicz, A survey of constraint handling techniques in evolutionary computation methods., Evol. Program. 4 (1995) 135–155.
[18] Z. Michalewicz, M. Schoenauer, Evolutionary algorithms for constrained parameter optimization problems, Evol. Comput. 4 (1) (1996) 1–32.
[19] F. Neri, V. Tirronen, Recent advances in differential evolution: a survey and experimental analysis, Artif. Intell. Rev. 33 (1–2) (2010) 61–106.
[20] J. Nocedal, S. Wright, Numerical Optimization, Springer Science & Business Media, 2006.
[21] C. Peng, H.-L. Liu, F. Gu, A novel constraint-handling technique based on dynamic weights for constrained optimization problems, Soft Comput. 22 (12) (2018) 3919–3935.
[22] R. Poláková, L-shade with competing strategies applied to constrained optimization, in: 2017 IEEE Congress on Evolutionary Computation (CEC), IEEE, 2017, pp. 1683–1689.
[23] X. Qiu, K.C. Tan, J.-X. Xu, Multiple exponential recombination for differential evolution, IEEE Trans. Cybern. 47 (4) (2016) 995–1006.
[24] R.V. Rao, V.J. Savsani, D. Vakharia, Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems, Comput.-Aided Des. 43 (3) (2011) 303–315.
[25] R.V. Rao, G. Waghmare, A new optimization algorithm for solving complex constrained design optimization problems, Eng. Optim. 49 (1) (2017) 60–83.
[26] S.S. Rao, Engineering Optimization: Theory and Practice, John Wiley & Sons, 2009.
[27] T.P. Runarsson, X. Yao, Stochastic ranking for constrained evolutionary optimization, IEEE Trans. Evol. Comput. 4 (3) (2000) 284–294.
[28] T.P. Runarsson, X. Yao, Search biases in constrained evolutionary optimization, IEEE Trans. Syst. Man Cybern.Part C 35 (2) (2005) 233–243.
[29] C. Saha, S. Das, K. Pal, S. Mukherjee, A fuzzy rule-based penalty function approach for constrained evolutionary optimization, IEEE Trans. Cybern. 46 (12) (2016) 2953–2965.
[30] R.A. Sarker, S.M. Elsayed, T. Ray, Differential evolution with dynamic parameters selection for optimization problems., IEEE Trans. Evol. Comput. 18 (5) (2014) 689–707.
[31] T. Takahama, S. Sakai, Constrained optimization by the ε constrained differential evolution with an archive and gradient-based mutation, in: Evolutionary Computation (CEC), 2010 IEEE Congress on, IEEE, 2010, pp. 1–9.
[32] T. Takahama, S. Sakai, Efficient constrained optimization by the ε constrained adaptive differential evolution, in: Evolutionary Computation (CEC), 2010 IEEE Congress on, IEEE, 2010, pp. 1–8.
[33] J.H. Tam, Z.C. Ong, Z. Ismail, B.C. Ang, S.Y. Khoo, A new hybrid GA- ACO- PSO algorithm for solving various engineering design problems, Int. J. Comput. Math. (2018) 1–37.
[34] B. Tessema, G.G. Yen, An adaptive penalty formulation for constrained evolutionary optimization, IEEE Trans. Syst. ManCybern.-Part A 39 (3) (2009) 565–578.
[35] A. Trivedi, K. Sanyal, P. Verma, D. Srinivasan, A unified differential evolution algorithm for constrained optimization problems, in: 2017 IEEE Congress on Evolutionary Computation (CEC), IEEE, 2017, pp. 1231–1238.
[36] B.-C. Wang, H.-X. Li, Y. Feng, An improved teaching-learning-based optimization for constrained evolutionary optimization, Inf. Sci. 456 (2018) 131–144.

[37] B.-C. Wang, H.-X. Li, J.-P. Li, Y. Wang, Composite differential evolution for constrained evolutionary optimization, IEEE Trans. Syst. Man Cybern. (2018), doi:10.1109/TSMC.2018.2807785.
[38] B.-C. Wang, H.-X. Li, Q. Zhang, Y. Wang, Decomposition-based multiobjective optimization for constrained evolutionary optimization, IEEE Trans. Syst. Man Cybern. (2018).
[39] Y. Wang, Z. Cai, Q. Zhang, Enhancing the search ability of differential evolution through orthogonal crossover, Inf. Sci. 185 (1) (2012) 153–177.
[40] Y. Wang, Z. Cai, Y. Zhou, W. Zeng, An adaptive tradeoff model for constrained evolutionary optimization, IEEE Trans. Evol. Comput. 12 (1) (2008) 80.
[41] Y. Wang, B.-C. Wang, H.-X. Li, G.G. Yen, Incorporating objective function information into the feasibility rule for constrained evolutionary optimization, IEEE Trans. Cybern. 46 (12) (2016) 2938–2952.
[42] D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, IEEE Trans. Evol. Comput. 1 (1) (1997) 67–82.
[43] G. Wu, R. Mallipeddi, P. Suganthan, Problem Definitions and Evaluation Criteria for the CEC 2017 Competition on Constrained Real-Parameter Optimization, National University of Defense Technology, Changsha, Hunan, PR China and Kyungpook National University, Daegu, South Korea and Nanyang Technological University, Singapore, Technical Report, 2017.
[44] B. Xu, X. Chen, L. Tao, Differential evolution with adaptive trial vector generation strategy and cluster-replacement-based feasibility rule for constrained optimization, Inf. Sci. 435 (2018) 240–262.
[45] W. Zhang, G.G. Yen, Z. He, Constrained optimization via artificial immune system, IEEE Trans. Cybern. 44 (2) (2014) 185–198.