Multiple Exponential Recombination for Differential Evolution

Xin Qiu, Kay Chen Tan, *Fellow, IEEE*, and Jian-Xin Xu, *Fellow, IEEE*

***Abstract*—Differential evolution (DE) is a popular population-based metaheuristic approach for solving numerical optimization problems. In recent years, considerable research has been devoted to the development of new mutation strategies and parameter adaptation mechanisms. However, as one of the basic algorithmic components of DE, the crossover operation has not been sufficiently examined in existing works. Most of the main DE variants solely employ traditional binomial recombination, which has intrinsic limitations in handling dependent subsets of variables. To fill this research niche, we propose a multiple exponential recombination that inherits all the main advantages of existing crossover operators while possessing a stronger ability in managing dependent variables. Multiple segments of the involved solutions will be exchanged during the proposed operator. The properties of the new scheme are examined both theoretically and empirically. Experimental results demonstrate the robustness of the proposed operator in solving problems with unknown variable interrelations.**

***Index Terms*—Crossover operator, differential evolution (DE), variable interrelation.**

I. Introduction

DIFFERENTIAL evolution (DE), proposed by Storn and Price [1], has many advantages, e.g., it is simple and straightforward to implement and exhibits good performance on a wide variety of problems. The number of control parameters in DE is also small and the space complexity of DE is low [2]. Because of these superiorities, DE has gained more attention in recent years, and many new variants of DE are emerging [3]–[15]. Among the existing DE variants, most works focus on designing new mutation strategies and adapting control parameters intelligently. However, as one of the basic steps in traditional DE [1], crossover operation has not been sufficiently examined in the main DE variants and most of the algorithms solely applied binomial recombination [6], [16]. In contrast, exponential recombination received less attention during the design of new DE algorithms.

According to some comparative studies [17], [18], binomial recombination is generally more robust and efficient than exponential recombination. To seek the reason, the relation between the mutation probability [16] and the value of control parameter Cr (crossover rate) is linear in binomial recombination while in exponential recombination it is nonlinear. Thus, it may result in difficulties in choosing proper values for Cr during implementations of exponential recombination, especially in solving high-dimensional problems [16]. Moreover, binomial recombination is able to generate all the 2^D possible solutions during crossover (where D is the dimension of decision space of the problem) whereas exponential recombination can only cover part of the 2^D possibilities [17].

Intrinsically, the main difference between the two recombination operators lies in the distribution of the exchanged variables. Exponential recombination leads to a consecutive crossover so that a set of adjacent elements are exchanged. In binomial recombination, the exchanged variables are dispersed randomly after the nonconsecutive crossover. Based on existing empirical investigations [16], [18], consecutive crossover shows promising results in solving nonseparable problems, in which the variables are strongly dependent with each other. The global optima of a nonseparable problem cannot be achieved by optimizing each variable separately, thus maintaining the dependent subsets of variables is very critical in solving this kind of problems [19].

As a consecutive crossover operator, exponential recombination tends to preserve strongly dependent components that are adjacent or physically proximate to each other. For binomial recombination, all the elements are presumed to be linked equally strongly and all the dependent variables stand the same chance to be split up during crossover. Consequently, the efficiency of the two crossover operators depend heavily on the interrelationships among variables. Considering the fact that the variable linkage information is not available in advance for most optimization problems [20], selection of the proper crossover strategy becomes a difficult issue for existing DE algorithms. In addition, even though the highly related variables are arranged adjacent to one another, the current exponential recombination is still inconvenient to use in face of high-dimensional problems and may also be ineffective regardless of the problem dimensionality due to its intrinsic disadvantages mentioned above. A more robust crossover

Manuscript received September 15, 2015; revised January 6, 2016; accepted February 20, 2016. Date of publication March 15, 2016; date of current version March 15, 2017. This work was supported by the Singapore Ministry of Education Academic Research Fund Tier 1 under Project R-263-000-A12-112. This paper was recommended by Associate Editor J. Zhang.

X. Qiu is with the Graduate School for Integrative Sciences and Engineering, National University of Singapore, Singapore 119077 (e-mail: qiuxin@u.nus.edu).

K. C. Tan and J.-X. Xu are with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 119077.

This paper has supplementary downloadable multimedia material available at http://ieeexplore.ieee.org provided by the authors. The supplementary PDF document includes the deduction process for (16), the detailed mathematical derivation for (29)–(32) and the supplementary Fig. S1. The total size of the file is 152KB.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2016.2536167

strategy that can efficaciously handle different types of variable interrelations is highly desirable.

To circumvent the above issue, a new crossover operator named multiple exponential recombination is proposed in this paper. The main innovation of the proposed operator is the semi-consecutive crossover strategy, in which multiple segments of the involved individuals will be combined to form a new solution. In this way, the new operator becomes more robust than the traditional crossover strategies in handling dependent variables. Moreover, the specially designed mechanisms enable the new operator to preserve all the main advantages of binomial recombination and exponential recombination: linear relationship between mutation probability and control parameter value, coverage of all possible offsprings, and strengths in preserving particular types of dependent subsets. The properties of multiple exponential recombination is examined both theoretically and empirically. Implementation in different DE variants demonstrates the robustness of the proposed operator in solving problems with unknown variable interrelations.

The remainder of this paper is organized as follows. Section II introduces the algorithmic elements of basic DE and reviews the existing related work. Section III describes the technical details of the proposed strategy and discusses the underlying rationales. Section IV analyzes the properties of the proposed operator empirically and evaluates its optimization performance via experiments. Section V concludes this paper and highlights the potential future directions.

II. Background

A. Classical DE Algorithm

1) Initialization: The first step of DE is the initialization of the population of N and D over the search space, where N denotes the population size and D denotes the variable dimensions. We symbolize each individual by $X_{i,g} = (x^1_{i,g}, x^2_{i,g}, \ldots, x^D_{i,g})$, where $i = 1, 2, \ldots, N$, $g = 0, 1, \ldots, G_{\max}$ and $G_{\max}$ denotes the maximum number of generations. Furthermore, let us define the lower search bound as $X_{\min} = (x^1_{\min}, x^2_{\min}, \ldots, x^D_{\min})$ and the upper search bound as $X_{\max} = (x^1_{\max}, x^2_{\max}, \ldots, x^D_{\max})$. Finally, the initial value of the ith individual is generated as

$$x^j_{i,0} = x^j_{\min} + \mathrm{rand}(0,1) \cdot \left(x^j_{\max} - x^j_{\min}\right), j = 1, 2, 3, \ldots, D. \tag{1}$$

2) Mutation: In this step, each individual will generate a new individual, called the mutant vector $V_{i,g}$. The most frequently used mutation strategies are listed

"DE/rand/1"

$$V_{i,g} = X_{r_1,g} + F \cdot \left(X_{r_2,g} - X_{r_3,g}\right) \tag{2}$$

"DE/best/1"

$$V_{i,g} = X_{\mathrm{best},g} + F \cdot \left(X_{r_1,g} - X_{r_2,g}\right) \tag{3}$$

"DE/rand/2"

$$V_{i,g} = X_{r_1,g} + F \cdot \left(X_{r_2,g} - X_{r_3,g}\right) + F \cdot \left(X_{r_4,g} - X_{r_5,g}\right) \tag{4}$$

"DE/best/2"

$$V_{i,g} = X_{\mathrm{best},g} + F \cdot \left(X_{r_1,g} - X_{r_2,g}\right) + F \cdot \left(X_{r_3,g} - X_{r_4,g}\right) \tag{5}$$

"DE/current-to-best/2"

$$\begin{aligned} V_{i,g} = X_{i,g} + F \cdot \left(X_{\mathrm{best},g} - X_{i,g}\right) + F \cdot \left(X_{r_1,g} - X_{r_2,g}\right) \\ + F \cdot \left(X_{r_3,g} - X_{r_4,g}\right) \end{aligned} \tag{6}$$

"DE/current-to-rand/2"

$$\begin{aligned} V_{i,g} = X_{i,g} + F \cdot \left(X_{r_1,g} - X_{i,g}\right) + F \cdot \left(X_{r_2,g} - X_{r_3,g}\right) \\ + F \cdot \left(X_{r_4,g} - X_{r_5,g}\right) \end{aligned} \tag{7}$$

where $X_{\mathrm{best},g}$ denotes the individual with the best fitness value in current generation. The indices r_1–r_5 are randomly selected integers from $\{1, 2, \ldots, N\}$ that are distinct from i and mutually different. $F \in [0, 1]$ is a real parameter, called the scaling factor.

3) Crossover: After mutation, crossover operation is employed to generate the trial vectors $U_{i,g}$. During crossover, mutant vectors are recombined with the original members of the current population, called target vectors, to form trial vectors. Two basic crossover schemes of DE are exponential recombination and binomial recombination. Binomial recombination is mostly used in DE literature [6].

Binomial recombination is performed on each variable and it could be outlined as

$$u^j_{i,g} = \begin{cases} v^j_{i,g}, & \text{if } \mathrm{rand}(0,1) \le \mathrm{Cr} \text{ or } j = j_{\mathrm{rand}} \\ x^j_{i,g}, & \text{otherwise} \end{cases} \tag{8}$$

where $j_{\mathrm{rand}} \in \{1, 2, 3, \ldots, D\}$ is a randomly selected index to ensure that the trial vector could get at least one component from the mutant vector. rand(0, 1) is a uniform random number on the interval [0, 1] and independently generated for each i and each j. Cr is called the crossover probability.

In exponential recombination, first two integers n and L are generated separately, where n is the starting point of crossover operation in the involved vectors and L is the number of variables that get exchanged during the crossover operation. More specifically, n is randomly chosen between the interval $[1, D]$, where D denotes the problem dimension. The pseudo-code to obtain L is

```
$L = 0$
WHILE ((rand(0, 1) $\le$ Cr) AND ($L \le D$))
    DO{$L = L + 1$}
```

where Cr is the crossover probability. rand(0, 1) is a uniform random number generator on the interval [0, 1]. If $L \ge 1$, then the trial vectors are generated as

$$u^j_{i,g} = \begin{cases} v^j_{i,g}, & \text{for } j = n, n+1, n+2, \ldots, n+L-1 \\ x^j_{i,g}, & \text{for all other } j \in [1, D]. \end{cases} \tag{9}$$

Otherwise, trial vectors will be identical to target vectors ($L = 0$).

4) Selection: Selection is the last step to generate the population of next generation. The process of selection is to determine whether the target vector or the trial vector survives to the next generation according to their fitness value.

The selection operation in DE is described

$$X_{i,g+1} = \begin{cases} U_{i,g}, & \text{if } f\left(U_{i,g}\right) \leq f\left(X_{i,g}\right) \\ X_{i,g}, & \text{otherwise} \end{cases} \tag{10}$$

where $f(X)$ is the fitness function to be minimized.

B. Related Work

The behavior of the DE algorithm is deeply influenced by selection of mutation scheme, crossover strategy, and control parameters. In order to improve the performance of traditional DE, a good volume of enhanced DE variants were proposed in the past 20 years [3]–[11], [21]–[25]. Among these studies, most of them are related with mutation operators [26]–[31] and control parameters [28], [32], [33]. In comparison, crossover operator attracted much less attention and only few works focused on examining the crossover in DE.

The major role of crossover is to enhance the potential diversity of the population by recombining the mutant vector with one existing solution. Two different crossover strategies were proposed in the original work of Storn and Price [1], [34], namely, the exponential recombination and the binomial recombination. In the later development of new DE variants, the binomial version was much more frequently employed as the crossover scheme [2], [16], [18]. Mezura-Montes *et al.* [17] compared the performances of several DE variants with different mutation and crossover strategies. Based on the results on various types of problems, they concluded that binomial recombination was much better than exponential recombination due to the fact that not all the recombinations of mutant vector and current parent can be sampled in the exponential variant. In Tvrdik's empirical study [35], the use of exponential recombination only increased efficiency in comparison with binomial recombination for small part of the tested problems. The superiority of binomial recombination over exponential one was also claimed by Jeyakumar and Shanmugavelayutham [36], [37] in their two comparative studies. Although different behaviors for the two crossover schemes were observed in the above works, no further investigation was conducted to examine the cause of these differences. Lin *et al.* [18] discovered that the consecutive crossover was more reliable in solving some nonseparable problems but did not provide detailed explanations for this phenomenon. Zaharie [16] analyzed the relationship between parameter Cr and mutation probability for both crossover strategies from a mathematical perspective. It was suggested after implementing empirical tests [16], [38] that the behavior of the algorithm during crossover depended more on the mutation probability.

Besides the comparative studies on existing crossover strategies, a small number of improved crossover operations were proposed in the recent years. Islam *et al.* [8] introduced the p-Best crossover operation, in which a greedy parent selection strategy was incorporated with binomial recombination. Guo and Yang [39] utilized eigenvectors of covariance matrix of individual solutions in binomial recombination. Zhao and Suganthan [40] fixed the length of the crossover in exponential recombination according to the dimensionality of the problems and named it as the linearly scalable exponential crossover operator. It is notable that actually these new strategies are intrinsically identical with the original binomial recombination or exponential recombination in terms of selection strategy for exchanging positions, only the involved solutions or the crossover length are changed. Development of an essentially new crossover operator for DE is still a research niche.

III. Multiple Exponential Recombination

A. Overview

The aim of the proposed new crossover strategy is to inherit all the main advantages of traditional crossover operators while providing a more robust performance in handling different types of variable interrelations.

As the most widely used crossover scheme in DE literatures, binomial recombination offers two main advantages when compared with exponential recombination: 1) it is convenient to control the mutation probability in binomial recombination via adapting the value of parameter Cr and 2) all possible 2^D recombinations of the mutant vector and target vector can be generated during binomial recombination, where D is the dimension of decision space. In order to preserve these two strengths, a linear relationship between the control parameter and the mutation probability has been created during the design of the new operator. In addition, the new operator is developed as a multipoint crossover operation [41] so that all the possible configurations can be covered during recombination.

Besides the above features, the new strategy is expected to overcome the limitations of existing crossover operators in handling dependent variables. More specifically, under similar mutation probability [16], the adjacent or physically proximate variables are more likely to be disrupted in binomial recombination than in exponential recombination. Conversely, the chance to update together the variables that are distributed distantly from each other is much lower in exponential recombination than in binomial recombination.

Fig. 1 illustrates the limitations of the two traditional crossover operators in solving nonseparable problems. Since the crossover operation is conducted in a circular manner in exponential recombination, we use the outer circle and inner circle to represent the target vector and mutant vector, respectively. Each rectangle represents a decision variable and the variables marked with red and blue represent two dependent subsets. Optimizing the elements within one subset simultaneously will lead to more efficient exploration. The two-headed arrows mean that the pointed variable in the target vector will be replaced by the corresponding variable in the mutant vector to form the trial vector. Assuming that both crossover operations are going to change 6 out of 12 variables, two common situations are depicted in Fig. 1 to demonstrate the limitations of each strategies. For binomial recombination, the crossover is performed on each variable independently so that the exchange points are more likely to be distributed uniformly. In this

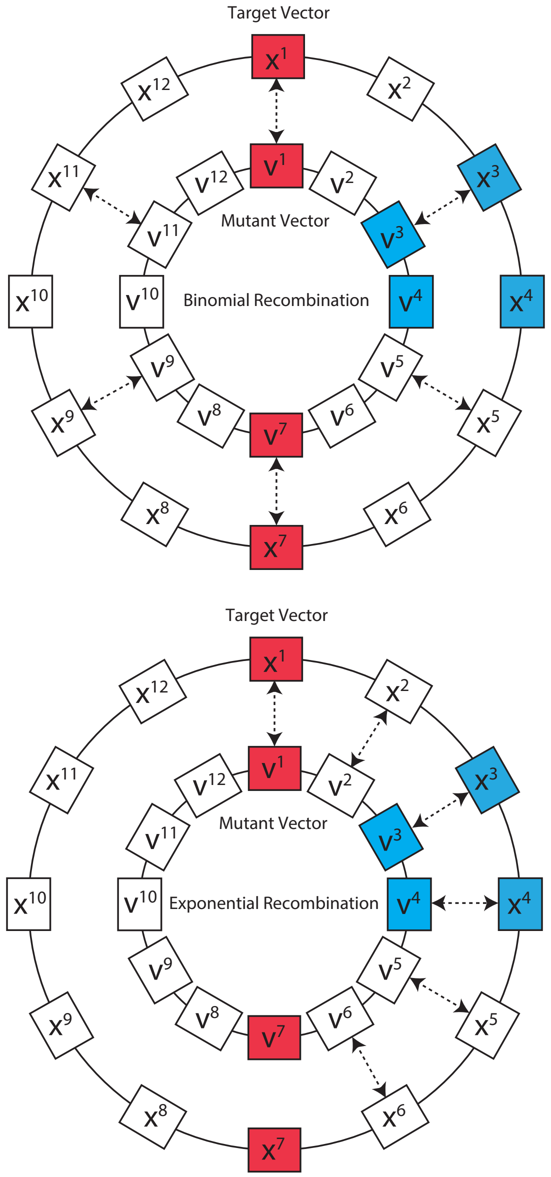


Fig. 1. Limitations of binomial recombination and exponential recombination in handling dependent variables via two examples. The elements with same color (red or blue) are strongly dependent on each other. All the involved solutions are visualized in a circular manner.

case, the dependent subset x^3 and x^4 will be disrupted since they are adjacent to each other. As a consecutive crossover operation, exponential recombination exchanges a segment of the involved solutions at each time. Thus, adjacent variables like x^3 and x^4 have a higher chance to be updated together. However, if the distance between two variables exceeds the length of this segment, it is impossible to update these variables concurrently in exponential recombination. As shown in Fig. 1, given the fact that the length of exchange segment is less than 7, under no circumstances can x^1 and x^7 be replaced by the corresponding elements in mutant vector at the same time. Only binomial recombination is able to update x^1 and x^7 simultaneously. In summary, solely applying a consecutive or nonconsecutive crossover operator is not sufficient to properly handle different types of variable interrelations. To address this issue, a semi-consecutive crossover operator is developed in this paper.

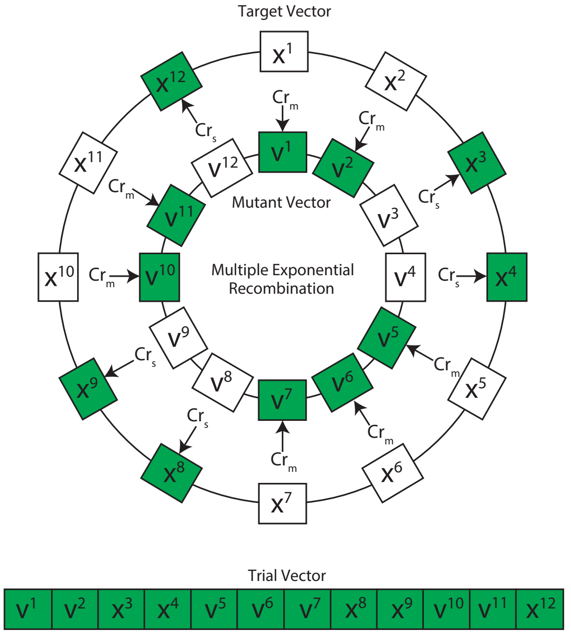


Fig. 2. Detailed behaviors during a multiple exponential recombination. The elements in color are the ones selected to construct the trial vector.

B. Algorithmic Structure

The main idea of the proposed strategy is to perform the basic operations of traditional exponential recombination in a multiple manner. Segments from the mutant vector and the target vector will alternately constitute the trial vector, thereby leading to a semi-consecutive crossover. Fig. 2 and Algorithm 1 demonstrate the standard procedure of multiple exponential recombination.

First, a starting point n is randomly selected from $[1, D]$, where D is the dimension of the optimization problem. Starting from dimension n, a segment of the mutant vector will be extracted to construct the trial vector. The length of this segment is decided by a sequence of Bernoulli trials with success probability Cr_m. Similar to the operations in traditional exponential recombination, each successful event will increase the segment length by 1. This process will be terminated once an unsuccessful trial happened. Please note that the length can be 0 if the first Bernoulli trial fails. Subsequently, starting from the dimension where last unsuccessful event happened, another segment from the target vector will be selected to constitute the trial vector. The procedure to decide this segment length is the same as in last step, and the only difference is that the success probability for each trial is changed to Cr_s. The proposed operator will keep switching between the above two processes until all the components of the trial vector have been decided.

From the above description, it is obvious that the behavior of the operator depends heavily on the selection of Cr_m and Cr_s. However, the relationship between Cr_m (or Cr_s) and the segment length is nonlinear, and the effect of Cr_m and Cr_s on mutation probability is even more complicated. This issue will cause difficulty in controlling the behavior of the proposed operator during applications. To address this issue, another two control parameters are used instead, namely Cr and T.

Algorithm 1 Procedure of Multiple Exponential Recombination

```
Require:
    $X_{i,g} = (x_{i,g}^1, x_{i,g}^2, \dots, x_{i,g}^D)$: Target Vector
    $V_{i,g} = (v_{i,g}^1, v_{i,g}^2, \dots, v_{i,g}^D)$: Mutant Vector
    $Cr$: Parameter to control the mutation probability
    $T$: Parameter to control the lengths of the exchanged segments
Ensure:
    $U_{i,g} = (u_{i,g}^1, u_{i,g}^2, \dots, u_{i,g}^D)$: Trial vector
$E_m = T \cdot Cr$
$E_s = T \cdot (1 - Cr)$
$Cr_m = E_m/(E_m + 1)$
$Cr_s = E_s/(E_s + 1)$
Set $n$ as an integer randomly generated from the interval $[1, D]$
$k = 1, Mutation\_Enable = 1$
while $k \leq D$ do
    if $Mutation\_Enable = 1$ then
        while $k \leq D$ and $rand(0,1) \leq Cr_m$ do
            $d = \langle n \rangle_D$, where $\langle n \rangle_D$ equals to $n$ if $n \leq D$ and equals to $n - D$ if $n > D$
            $u_{i,g}^d = v_{i,g}^d$
            $n = n + 1$
            $k = k + 1$
        end while
        $Mutation\_Enable = 0$
    else
        while $k \leq D$ and $rand(0,1) \leq Cr_s$ do
            $d = \langle n \rangle_D$
            $u_{i,g}^d = x_{i,g}^d$
            $n = n + 1$
            $k = k + 1$
        end while
        $Mutation\_Enable = 1$
    end if
end while
```

The users are only required to set Cr and T, and the value of Cr_m and Cr_s will then be automatically decided as follows:

$$E_m = T \cdot \mathrm{Cr} \tag{11}$$

$$E_s = T \cdot (1 - \mathrm{Cr}) \tag{12}$$

$$\mathrm{Cr}_m = E_m/(E_m + 1) \tag{13}$$

$$\mathrm{Cr}_s = E_s/(E_s + 1). \tag{14}$$

Briefly speaking, E_m and E_s are the approximate average lengths of the segments extracted from the mutant vector and target vector, respectively. Cr decides the mutation probability and T controls the length of exchanged segments. The underlying rationale of this design will be discussed from a mathematical perspective in the next section.

C. Theoretical Analysis

In both traditional binomial and exponential recombination, the value of parameter Cr strongly affects the mutation probability p_m [16], which represents the probability that a component of the trial vector is mutated (means this component is taken from the mutant vector). A higher p_m will lead to more mutant elements in the trial vector, whereas a lower p_m tends to deliver more variables from the target vector to the trial vector. As a result, the convergence speed of the population and even the entire optimization performance are heavily influenced by the value of p_m. Creating a linear relationship between Cr and p_m will make it more convenient for the users to control the behavior of the algorithm. Among the existing crossover operations, this is only achieved by binomial recombination. In order to preserve this main advantage of binomial recombination, the control parameters in multiple exponential recombination are specially designed based on the following mathematical analysis.

In the proposed strategy, multiple segments from both mutant vector and target vector construct the trial vector. Thus, the mutation probability is actually decided by the average lengths of these segments. Following the definition of Cr_m, the length of one segment extracted from the mutant vector (denoted as L_m) is decided via the following process:

$L_m = 0$
WHILE $((\mathrm{rand}(0,1) \leq \mathrm{Cr}_m)$ AND $(L_m \leq D))$
DO$\{L_m = L_m + 1\}$

where $\mathrm{rand}(0,1)$ is a uniform random number generator on the interval $[0, 1]$. L_m is the number of variables contained by the segment and D is the dimension of the decision space.

In such a process, the probability for L_m to be h is given by

$$P(L_m = h) = \begin{cases} \mathrm{Cr}_m^h \cdot (1 - \mathrm{Cr}_m), & \text{if } 0 \leq h < D \\ \mathrm{Cr}_m^D, & \text{if } h = D. \end{cases} \tag{15}$$

The mathematical expectation of this length L_m is computed as follows:

$$\begin{aligned} E(L_m) &= \sum_{h=0}^{D-1} h \cdot \mathrm{Cr}_m^h \cdot (1 - \mathrm{Cr}_m) + D \cdot \mathrm{Cr}_m^D \\ &= \mathrm{Cr}_m \cdot \frac{1 - \mathrm{Cr}_m^D}{1 - \mathrm{Cr}_m}. \end{aligned} \tag{16}$$

The deduction process of the above equation is provided in the supplementary material.

Analogously, given a Cr_s, the expectation of the length of one segment taken from the trial vector is calculated as

$$E(L_s) = \mathrm{Cr}_s \cdot \frac{1 - \mathrm{Cr}_s^D}{1 - \mathrm{Cr}_s} \tag{17}$$

where L_s represents the number of components inside the segment.

Under the assumption that the dimension of the problem D is larger than 50 and both Cr_m and Cr_s are less than 0.9, we will have

$$1 - \mathrm{Cr}_m^D \approx 1 - \mathrm{Cr}_s^D \approx 1. \tag{18}$$

Thus

$$E(L_m) \approx \frac{\mathrm{Cr}_m}{1 - \mathrm{Cr}_m} \tag{19}$$

$$E(L_s) \approx \frac{\mathrm{Cr}_s}{1 - \mathrm{Cr}_s}. \tag{20}$$

In the proposed strategy, E_m and E_s are the expected values of $E(L_m)$ and $E(L_s)$, respectively. By replacing $E(L_m)$ and $E(L_s)$ with E_m and E_s in (19) and (20), we will obtain

$$E_m \approx \frac{\mathrm{Cr}_m}{1-\mathrm{Cr}_m} \tag{21}$$

$$E_s \approx \frac{\mathrm{Cr}_s}{1-\mathrm{Cr}_s}. \tag{22}$$

Consequently, the required values of Cr_m and Cr_s can be computed using (13) and (14), respectively.

Considering the mutation probability p_m is defined as the probability that an element in the trial vector is inherited from the mutant vector, the value of p_m can be approximated using $E(L_m)$ and $E(L_s)$ as

$$p_m \approx \frac{E(L_m)}{E(L_m)+E(L_s)}. \tag{23}$$

It is notable that the higher the dimension D is, the closer the true p_m and the above approximated value will be. Thus, the relationship between p_m and Cr is given by

$$p_m \approx \frac{E_m}{E_m+E_s} = \frac{T\cdot \mathrm{Cr}}{T\cdot\mathrm{Cr}+T\cdot(1-\mathrm{Cr})} = \mathrm{Cr}. \tag{24}$$

An approximate linear relationship is achieved between the mutation probability p_m and control parameter Cr. This is the first advantage of the proposed operator.

In order to fulfill the assumption that both Cr_m and Cr_s are less than 0.9 (0.9 included), we will have

$$\mathrm{Cr}_m = E_m/(E_m+1) \le 0.9 \Rightarrow E_m \le 9 \tag{25}$$

$$\mathrm{Cr}_s = E_s/(E_s+1) \le 0.9 \Rightarrow E_s \le 9. \tag{26}$$

Thus, considering (11), (12) and the fact that the Cr value is commonly chosen from $[0.1, 0.9]$, we will have

$$E_m = T\cdot\mathrm{Cr} \le T\cdot 0.9 \le 9 \Rightarrow T \le 10 \tag{27}$$

$$E_s = T\cdot(1-\mathrm{Cr}) \le T\cdot 0.9 \le 9 \Rightarrow T \le 10. \tag{28}$$

Based on (11) and (12), a relatively higher T value will lead to a longer average length of the exchange segments. This is beneficial to increase the chance of preserving the physically proximate dependent variables. Taking all the above analysis into account, the value of T will be fixed as 10 for all the experiments in this paper.

The second advantage of multiple exponential recombination is that all 2^D possible trial vectors can be generated during crossover. To explain briefly, the number of the exchanged segments is unfixed and the lengths of these segments can vary between 0 to D, thereby leading to 2^D possibilities during the recombination process.

The last and most important strength of the new operator is the robustness in handling different types of variable interrelations. Since preservation of dependent subsets is very critical in solving nonseparable problems, one quantitative criterion to judge the ability of an operator in managing dependent variables is the distribution of probabilities to split up these subsets. Both the physical distance between the dependent variables in the vector representation and the values of other control parameters will affect the disruption probability. We have mathematically derived the probabilities to disrupt two dependent variables with different physical distances while employing multiple exponential recombination. The derivation is based on modeling the crossover process as a discrete-time Markov chain [42]. Interested readers may refer to the supplementary material for details of this mathematical derivation. The final results are presented below.

A transition matrix A is used in the representation of the obtained probability

$$A = \begin{bmatrix} \dfrac{\mathrm{Cr}_m}{1-(1-\mathrm{Cr}_m)(1-\mathrm{Cr}_s)} & \dfrac{(1-\mathrm{Cr}_s)\mathrm{Cr}_m}{1-(1-\mathrm{Cr}_m)(1-\mathrm{Cr}_s)} \\ \dfrac{(1-\mathrm{Cr}_m)\mathrm{Cr}_s}{1-(1-\mathrm{Cr}_m)(1-\mathrm{Cr}_s)} & \dfrac{\mathrm{Cr}_s}{1-(1-\mathrm{Cr}_m)(1-\mathrm{Cr}_s)} \end{bmatrix}. \tag{29}$$

The probability to disrupt two variables during multiple exponential recombination is calculated as

$$P_{\text{disruption}} = p_v\cdot q_x + p_x\cdot q_v \tag{30}$$

where p_v, p_x, q_v, and q_x are given by

$$\begin{bmatrix} p_v \\ p_x \end{bmatrix} = \frac{1}{n}\left(A^{D-1}+\cdots+A+A^0\right)\begin{bmatrix} \dfrac{\mathrm{Cr}_m}{1-(1-\mathrm{Cr}_m)(1-\mathrm{Cr}_s)} \\ \dfrac{(1-\mathrm{Cr}_m)\mathrm{Cr}_s}{1-(1-\mathrm{Cr}_m)(1-\mathrm{Cr}_s)} \end{bmatrix} \tag{31}$$

$$\begin{bmatrix} q_v \\ q_x \end{bmatrix} = A^L\begin{bmatrix} p_v \\ p_x \end{bmatrix} \tag{32}$$

where D is the dimensionality of the optimization problem and L is the dimension difference between the two variables in the circular representation of the solutions (count from the variable that first undergoes the crossover operation).

From the above derivation results, it is not straightforward to interpret the properties of this disruption probability. In order to provide a clearer demonstration, simulation experiments will be conducted in the next section to approximate the distribution of disruption probability using relative frequency. Based on the simulation result (Fig. 3), in comparison with traditional binomial recombination, the proposed operator has a similar probability to preserve the distant dependent subsets while possessing a lower probability to disrupt the physically proximate variables. This characteristic enables the proposed strategy to be more robust in managing variables with unknown interrelations. Details of the simulation experiments will be given in the next section.

IV. Empirical Study

A. Properties of the Proposed Strategy

Two critical properties of the proposed strategy will be examined in this section via experiments. One is the relationship between mutation probability p_m and the control parameter Cr. The other one is the probability to split up two dependent variables during crossover operations.

Based on the theoretical analysis in Section III-C, the relationship between p_m and Cr is approximately linear in multiple exponential recombination. In order to verify this conclusion empirically, the values of p_m under different Cr are approximated using the relative frequency (or empirical probability),

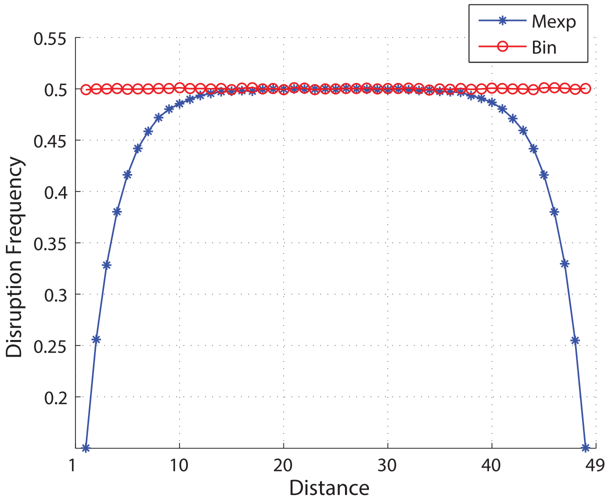


Fig. 3. Disruption frequency for multiple exponential recombination and binomial recombination under different distance settings. The dimensionality of the decision space is 50 and the Cr value is 0.5 for both strategies.

which is defined as the number of times an event occurred normalized by the total number of trials. When the number of trials is large enough, the obtained relative frequency can accurately reveal the true probability. In this paper, T is fixed as 10 for multiple exponential recombination and the value of Cr will change from 0 to 1 gradually at an interval of 0.01. For each value of Cr, 1×10^6 independent trials will be performed to simulate the process of the proposed operation and the traditional binomial recombination, respectively. The relative frequency of the event that a variable is mutated during the crossover is calculated for each Cr value, and we name this relative frequency as "mutation frequency." Fig. S1 (see supplementary material) shows simulation results when the dimensionality of decision space is 50 and 100. "Mexp" refers to multiple exponential recombination and "Bin" refers to binomial recombination. Consistent abbreviations are also used in all the following figures and tables.

From Fig. S1 (see supplementary material), it can be observed that both the crossover operators are able to produce an approximate linear relationship between the mutation frequency and Cr value. More specifically, the value of mutation frequency is very close to that of Cr in 50-D problems and is nearly identical with that of Cr in 100-D situations. Considering that the mutation frequency is a reliable approximation of p_m, this observation is in accordance with our previous discussions in Section III-C. A higher dimensionality will lead to a smaller discrepancy between p_m and Cr and an approximate linear relationship is achieved between them in the proposed strategy.

Next, behavior of multiple exponential recombination in handling dependent variables is investigated via simulations. To demonstrate the robustness of the proposed strategy, dependent variables with different distances are tested in the experiments. The probabilities to split up these dependent subsets is compared between multiple exponential recombination and binomial recombination. The dimension of the decision space is set as 50, T is set as 10 for multiple exponential recombination and Cr is fixed as 0.5 for both operators. Since the value of p_m is similar for the two strategies under same Cr, any difference in the performance is mainly due to the positions of the mutated components instead of the number of mutated variables. Without loss of generality, one variable is always placed in the first dimension and the other dependent variable will change its position from the second dimension till the last. The difference between their dimension indices is defined as the "distance" between them (e.g., a distance of "1" means the two variables are adjacent to each other). In the experiments, the distance between the two dependent variables will change from 1 to 49. 1×10^6 independent trials will be simulated for each setting. The relative frequency for the event that the two dependent variables are disrupted during crossover is computed and referred to as "disruption frequency." Fig. 3 presents the disruption frequency over different distances for both crossover operators.

For binomial recombination, the disruption frequency stays around 0.5 for all the distance settings. Similar disruption frequency is also observed in multiple exponential recombination for distances between 15 and 35. However, when the distance is less than 15 or larger than 35, the performances of the two strategies become vastly different. Multiple exponential recombination exhibits a gradually decreasing disruption frequency when the distance value becomes closer to 1 or 49. Considering the crossover operation in multiple exponential recombination is conducted in a circular manner, actually a distance of 49 also means the two variables are adjacent to each other and a distance of 25 is the most distant situation. Following this consideration, it can be concluded that the closer the two variables are placed, the less probable the two variables will be split up by multiple exponential recombination. In the most extreme case, where the two dependent variables are adjacent to each other, the disruption frequency may drop to around 0.15. Even in the worst case, where the distance between the two dependent variables is 25, the disruption frequency of the proposed strategy will not exceed that of the traditional binomial recombination.

To examine further, we have changed the Cr value from 0 to 1 at an interval of 0.01, and tested each Cr value under different distance settings. Similarly, 1×10^6 independent trials will be conducted for each parametric setup on each crossover operator. Fig. 4 shows the obtained disruption frequencies after aforementioned simulations. Based on Fig. 4, the behaviors of the two operators are almost the same when the distance between the two dependent variables is 25. In contrast, under the "distance=1" setting, multiple exponential recombination is able to provide a much lower disruption frequency under most Cr values. For the "distance=5" case, this lower disruption frequency is observed when the Cr value falls between 0.2 to 0.8. These results validate the conclusion that the proposed strategy is better than binomial recombination in preserving physically proximate variables without deteriorating the performance in handling distant variables.

To summarize, the proposed multiple exponential recombination preserves the advantages of traditional binomial recombination in handling distant dependent variables whilst

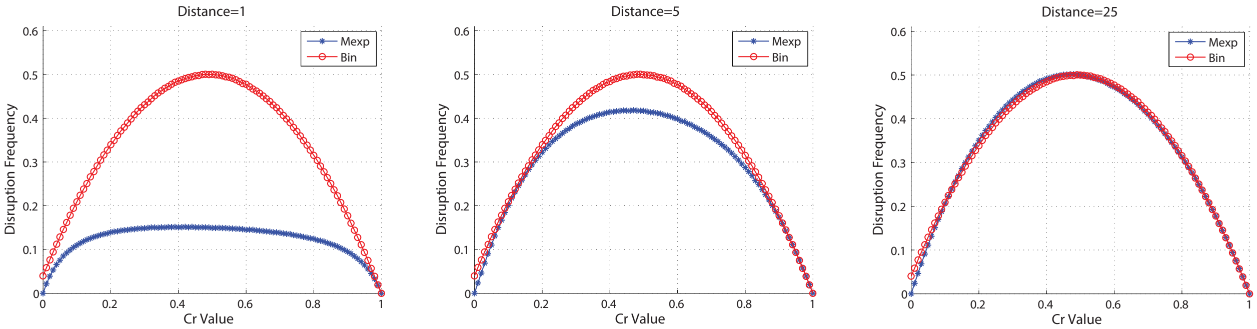


Fig. 4. Disruption frequency for multiple exponential recombination and binomial recombination under different Cr values and distance settings. The dimensionality of the decision space is 50.

TABLE I
MEAN AND STANDARD DEVIATION OF THE BEST-OF-RUN ERRORS FOR 30 INDEPENDENT RUNS OVER 50-D BENCHMARK SET

f_i	DE/rand/1/bin		DE/rand/1/Mexp			DE/best/1/bin		DE/best/1/Mexp			DE/c-t-b/2/bin		DE/c-t-b/2/Mexp		
(50D)	Mean	Std	Mean	Std	+/=/-	Mean	Std	Mean	Std	+/=/-	Mean	Std	Mean	Std	+/=/-
f_1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	=	4.54E+00	2.48E+01	**2.97E-13**	**1.52E-13**	+	0.00E+00	0.00E+00	0.00E+00	0.00E+00	=
f_2	6.43E+04	1.05E+04	**5.38E+03**	**9.83E+02**	+	3.76E+00	1.92E+01	**1.28E-12**	**1.18E-12**	+	2.44E+04	2.76E+03	**3.26E+02**	**7.32E+01**	+
f_3	4.59E+08	5.59E+07	**2.94E+08**	**4.52E+07**	+	1.12E+07	4.85E+06	**3.76E+06**	**1.65E+06**	+	1.84E+08	2.32E+07	**1.09E+08**	**2.35E+07**	+
f_4	8.95E+04	1.14E+04	**1.66E+04**	**2.49E+03**	+	4.19E+03	4.28E+03	**1.95E+03**	**3.79E+03**	+	3.90E+04	7.13E+03	**3.63E+03**	**6.76E+02**	+
f_5	4.90E+03	1.08E+03	**3.43E+03**	**1.20E+03**	+	5.57E+03	1.45E+03	6.13E+03	1.42E+03	=	3.76E+03	1.49E+03	3.79E+03	1.61E+03	=
f_6	3.58E+01	4.52E-01	**2.73E+00**	**1.24E+00**	+	5.66E+00	1.24E+01	**6.64E-01**	**1.51E+00**	+	2.78E+01	2.48E+00	**4.91E-06**	**1.86E-05**	+
f_7	1.36E-02	5.17E-03	**4.48E-03**	**1.99E-03**	+	1.25E-02	1.38E-02	9.23E-03	1.77E-02	=	1.92E-03	4.79E-03	**7.45E-05**	**2.18E-04**	+
f_8	2.11E+01	3.83E-02	2.11E+01	3.84E-02	=	2.11E+01	3.36E-02	2.11E+01	4.72E-02	=	2.11E+01	3.21E-02	2.11E+01	4.51E-02	=
f_9	2.33E+02	1.10E+01	**1.82E+02**	**1.14E+01**	+	8.52E+01	2.42E+01	8.77E+01	2.76E+01	=	2.54E+02	1.15E+01	**2.06E+02**	**1.00E+01**	+
f_{10}	3.85E+02	1.26E+01	3.79E+02	1.32E+01	=	3.06E+02	1.10E+02	**1.89E+02**	**1.08E+02**	+	3.88E+02	1.14E+01	**3.79E+02**	**1.26E+01**	+
f_{11}	7.30E+01	1.07E+00	**7.24E+01**	**1.05E+00**	+	3.95E+01	1.94E+01	3.26E+01	1.17E+01	=	7.21E+01	2.10E+00	7.19E+01	1.43E+00	=
f_{12}	7.93E+05	5.49E-14	7.99E+05	5.49E-14	=	1.12E+05	8.09E+04	1.29E+05	1.16E+05	=	3.60E+05	2.28E+05	4.70E+05	2.07E+05	=
f_{13}	2.98E+01	1.41E+00	**2.23E+01**	**1.10E+00**	+	1.24E+01	5.66E+00	1.43E+01	8.22E+00	=	2.90E+01	1.48E+00	**2.25E+01**	**8.71E-01**	+
f_{14}	2.32E+01	1.44E-01	2.33E+01	1.66E-01	=	2.27E+01	2.89E-01	2.26E+01	2.72E-01	=	2.32E+01	1.57E-01	2.31E+01	2.17E-01	=
	Total Number of (+/=/-)				9/5/0	Total Number of (+/=/-)				6/8/0	Total Number of (+/=/-)				8/6/0
f_i	DE/rand/2/bin		DE/rand/2/Mexp			DE/best/2/bin		DE/best/2/Mexp			DE/c-t-r/2/bin		DE/c-t-r/2/Mexp		
(50D)	Mean	Std	Mean	Std	+/=/-	Mean	Std	Mean	Std	+/=/-	Mean	Std	Mean	Std	+/=/-
f_1	4.90E-03	8.93E-04	**6.92E-04**	**1.17E-04**	+	0.00E+00	0.00E+00	0.00E+00	0.00E+00	=	0.00E+00	0.00E+00	0.00E+00	0.00E+00	=
f_2	1.03E+05	1.33E+04	**3.15E+04**	**2.48E+03**	+	2.93E+04	7.19E+03	**6.55E+02**	**3.56E+02**	+	5.03E+04	5.07E+03	**3.50E+03**	**4.29E+02**	+
f_3	6.96E+08	7.89E+07	**5.29E+08**	**7.22E+07**	+	1.52E+08	4.38E+07	**6.08E+07**	**2.04E+07**	+	3.30E+08	3.19E+07	**2.42E+08**	**3.03E+07**	+
f_4	1.30E+05	1.36E+04	**5.28E+04**	**6.35E+03**	+	4.97E+04	1.03E+04	**5.10E+03**	**2.90E+03**	+	7.17E+04	8.09E+03	**1.58E+04**	**2.10E+03**	+
f_5	1.59E+04	1.35E+03	**1.47E+04**	**1.40E+03**	+	5.28E+03	2.08E+03	**3.73E+03**	**1.59E+03**	+	1.01E+04	7.28E+02	**9.16E+03**	**9.47E+02**	+
f_6	7.42E+03	2.14E+03	**7.19E+02**	**1.60E+02**	+	3.82E+01	2.22E+01	**2.36E-07**	**9.10E-07**	+	3.79E+01	1.33E+00	**1.26E+01**	**3.08E+00**	+
f_7	1.49E+02	3.03E+01	**5.22E+01**	**9.97E+00**	+	2.31E-03	4.75E-03	**1.81E-03**	**4.38E-03**	+	1.18E-01	4.64E-02	**7.73E-02**	**4.21E-02**	+
f_8	2.11E+01	3.74E-02	2.11E+01	3.38E-02	=	2.11E+01	2.78E-02	2.11E+01	2.88E-02	=	2.11E+01	3.98E-02	2.11E+01	4.33E-02	=
f_9	2.84E+02	8.61E+00	**2.29E+02**	**1.01E+01**	+	2.38E+02	1.27E+01	**1.96E+02**	**1.38E+01**	+	2.56E+02	9.21E+00	**2.08E+02**	**1.16E+01**	+
f_{10}	4.33E+02	1.24E+01	4.34E+02	1.40E+01	=	3.88E+02	1.43E+01	3.82E+02	1.70E+01	=	3.93E+02	1.46E+01	3.92E+02	1.24E+01	=
f_{11}	7.28E+01	1.36E+00	7.28E+01	1.24E+00	=	7.27E+01	1.56E+00	**7.13E+01**	**2.10E+00**	+	7.25E+01	1.87E+00	7.18E+01	1.72E+00	=
f_{12}	1.74E+06	1.12E+05	**1.40E+06**	**1.57E+05**	+	9.91E+04	8.64E+04	8.81E+04	7.08E+04	=	1.30E+06	1.37E+05	**1.11E+06**	**1.04E+05**	+
f_{13}	5.87E+01	6.09E+00	**3.47E+01**	**1.60E+00**	+	2.94E+01	1.41E+00	**2.30E+01**	**1.39E+00**	+	3.04E+01	1.41E+00	**2.30E+01**	**1.10E+00**	+
f_{14}	2.33E+01	1.73E-01	2.33E+01	1.31E-01	=	2.31E+01	1.79E-01	2.32E+01	1.38E-01	=	2.32E+01	1.53E-01	2.31E+01	1.55E-01	=
	Total Number of (+/=/-)				10/4/0	Total Number of (+/=/-)				9/5/0	Total Number of (+/=/-)				9/5/0

The symbols "+/=/-" indicate the statistical test results using Wilcoxon rank sum test at the 5% significance level. "+" and "-" mean that the performance of the variant with multiple exponential recombination is significantly superior or inferior than that of the variant with binomial recombination, respectively. "=" means the performance difference between the two variants is not statistically significant. "c-t-r" and "c-t-b" represent "current-to-rand" and "current-to-best", respectively.

has a lower chance to disrupt the physically proximate dependent subsets. A more robust behavior is achieved by the proposed strategy in handling different types of variable interrelations.

B. Performance Evaluation

To evaluate the optimization performance of the new crossover operator, the proposed multiple exponential recombination will be implemented with six classical DE mutation strategies, namely DE/rand/1, DE/best/1, DE/rand/2, DE/best/2, DE/current-to-best/2, and DE/current-to-rand/2. Fourteen representative benchmark problems from the special session and competition on real parameter optimization held under the IEEE CEC 2005 [43] (f_1–f_{14}) are utilized to evaluate the performance of the algorithms. Both the 50-D and 100-D versions of these problems are tested. Numerous types of problems are covered including unimodal, multimodal, shifted, and rotated. All the problems are nonseparable except for f_1 and f_9. The performance of the algorithms with multiple exponential recombination is compared with those variants with binomial recombination. For all algorithms, population size is fixed as 100, F is set as 0.5, Cr is set as 0.5, and the maximum number of function evaluations is set to 50×10^4. T is fixed as 10 for multiple exponential recombination. The performance evaluation of the algorithm is based on the best-of-run error, which corresponds to the absolute difference between the best-of-run fitness value and the actual global optimum. All of the simulations were done on an Intel Core i7 machine with 16-GB RAM and 3.40-GHz speed. Tables I and II show the mean and standard deviation of the best-of-run errors over 30 independent runs for each algorithm on 50-D and 100-D benchmarks, respectively. In order to judge whether the results of multiple exponential variants and binomial variants differ in a statistically significant way, a nonparametric

TABLE II
MEAN AND STANDARD DEVIATION OF THE BEST-OF-RUN ERRORS FOR 30 INDEPENDENT RUNS OVER 100-D BENCHMARK SET

f_i	DE/rand/1/bin		DE/rand/1/Mexp			DE/best/1/bin		DE/best/1/Mexp			DE/c-t-b/2/bin		DE/c-t-b/2/Mexp		
(100D)	Mean	Std	Mean	Std	+/=/-	Mean	Std	Mean	Std	+/=/-	Mean	Std	Mean	Std	+/=/-
f_1	1.61E-05	4.09E-06	**5.28E-06**	**1.41E-06**	+	0.00E+00	0.00E+00	0.00E+00	0.00E+00	=	1.13E-10	2.33E-11	**4.28E-11**	**1.23E-11**	+
f_2	4.31E+05	3.02E+04	**2.46E+05**	**1.94E+04**	+	2.50E+04	1.20E+04	**2.78E+03**	**3.92E+03**	+	2.41E+05	2.26E+04	**1.08E+05**	**1.13E+04**	+
f_3	2.55E+09	2.98E+08	**2.37E+09**	**2.85E+08**	+	4.87E+07	1.48E+07	**3.70E+07**	**1.39E+07**	+	1.09E+09	1.50E+08	**9.39E+08**	**1.41E+08**	+
f_4	5.01E+05	3.65E+04	**3.32E+05**	**2.82E+04**	+	1.35E+05	3.62E+04	**7.53E+04**	**2.02E+04**	+	2.99E+05	3.70E+04	**1.78E+05**	**1.71E+04**	+
f_5	2.48E+04	1.75E+03	2.39E+04	1.55E+03	=	1.62E+04	2.90E+03	1.61E+04	2.88E+03	=	2.01E+04	1.09E+03	**1.93E+04**	**1.52E+03**	+
f_6	3.86E+02	7.63E+01	**1.13E+02**	**7.78E+00**	+	1.09E+02	4.81E+01	**1.19E+00**	**1.86E+00**	+	9.12E+01	3.13E-01	**7.48E+01**	**1.07E+00**	+
f_7	1.08E+01	1.82E+00	**4.85E+00**	**1.11E+00**	+	4.76E-03	6.66E-03	6.89E-03	8.40E-03	=	7.44E-01	3.09E-02	**6.60E-01**	**4.49E-02**	+
f_8	2.13E+01	2.11E-02	2.13E+01	3.54E-02	=	2.13E+01	2.48E-02	2.13E+01	2.70E-02	=	2.13E+01	2.85E-02	2.13E+01	2.32E-02	=
f_9	7.01E+02	1.62E+01	**6.61E+02**	**1.78E+01**	+	2.80E+02	5.34E+01	2.86E+02	5.05E+01	=	7.37E+02	2.11E+01	**6.99E+02**	**2.23E+01**	+
f_{10}	9.69E+02	2.29E+01	9.65E+02	2.65E+01	=	8.21E+02	2.82E+02	**6.85E+02**	**2.94E+02**	+	9.67E+02	1.79E+01	**9.56E+02**	**2.45E+01**	+
f_{11}	1.62E+02	2.15E+00	1.62E+02	1.70E+00	=	1.60E+02	2.67E+00	**8.59E+01**	**2.39E+01**	+	1.62E+02	1.62E+00	1.61E+02	1.87E+00	=
f_{12}	9.37E+06	6.29E+05	**8.93E+06**	**6.18E+05**	+	5.94E+05	3.19E+05	5.62E+05	2.80E+05	=	5.75E+06	1.66E+06	5.84E+06	1.46E+06	=
f_{13}	9.74E+01	4.65E+00	**8.24E+01**	**2.74E+00**	+	8.54E+01	3.17E+01	8.99E+01	2.10E+01	=	7.95E+01	2.13E+00	**7.28E+01**	**1.60E+00**	+
f_{14}	4.79E+01	1.47E-01	4.79E+01	1.65E-01	=	4.75E+01	3.49E-01	4.74E+01	3.19E-01	=	4.78E+01	2.01E-01	4.78E+01	2.55E-01	=
	Total Number of (+/=/-)				9/5/0	Total Number of (+/=/-)				6/8/0	Total Number of (+/=/-)				10/4/0
f_i	DE/rand/2/bin		DE/rand/2/Mexp			DE/best/2/bin		DE/best/2/Mexp			DE/c-t-r/2/bin		DE/c-t-r/2/Mexp		
(100D)	Mean	Std	Mean	Std	+/=/-	Mean	Std	Mean	Std	+/=/-	Mean	Std	Mean	Std	+/=/-
f_1	6.57E+03	4.64E+02	**4.39E+03**	**2.81E+02**	+	0.00E+00	0.00E+00	0.00E+00	0.00E+00	=	4.47E-04	8.26E-05	**2.91E-04**	**4.56E-05**	+
f_2	5.25E+05	3.69E+04	**3.63E+05**	**2.09E+04**	+	2.70E+05	2.66E+04	**1.35E+05**	**1.98E+04**	+	3.44E+05	1.84E+04	**1.99E+05**	**1.04E+04**	+
f_3	3.79E+09	3.56E+08	**3.55E+09**	**3.40E+08**	+	7.51E+08	1.65E+08	6.63E+08	1.73E+08	=	1.89E+09	1.83E+08	**1.76E+09**	**1.58E+08**	+
f_4	6.19E+05	6.01E+04	**4.53E+05**	**4.20E+04**	+	3.37E+05	3.65E+04	**2.17E+05**	**3.64E+04**	+	4.17E+05	3.77E+04	**2.79E+05**	**1.72E+04**	+
f_5	4.11E+04	1.89E+03	4.07E+04	2.15E+03	=	1.72E+04	2.20E+03	1.75E+04	1.78E+03	=	3.17E+04	1.19E+03	**3.09E+04**	**1.69E+03**	+
f_6	1.14E+09	1.36E+08	**6.93E+08**	**9.37E+07**	+	1.17E+02	3.23E+01	**8.11E+01**	**1.51E+01**	+	9.42E+02	1.18E+02	**3.92E+02**	**6.12E+01**	+
f_7	1.75E+04	9.76E+02	**1.49E+04**	**8.68E+02**	+	6.32E-01	9.61E-02	**5.45E-01**	**1.14E-01**	+	4.60E+01	9.81E+00	**2.96E+01**	**5.18E+00**	+
f_8	2.13E+01	2.20E-02	2.13E+01	2.39E-02	=	2.13E+01	2.51E-02	2.13E+01	2.86E-02	=	2.13E+01	2.24E-02	2.13E+01	1.96E-02	=
f_9	8.45E+02	2.15E+01	**7.94E+02**	**1.72E+01**	+	7.17E+02	2.91E+01	**6.76E+02**	**2.43E+01**	+	7.62E+02	1.32E+01	**7.19E+02**	**1.84E+01**	+
f_{10}	1.16E+03	2.57E+01	1.16E+03	2.67E+01	=	9.78E+02	4.05E+01	9.74E+02	3.70E+01	=	1.01E+03	1.42E+01	1.01E+03	2.64E+01	=
f_{11}	1.62E+02	2.12E+00	1.61E+02	1.76E+00	=	1.61E+02	1.95E+00	1.62E+02	1.38E+00	=	1.62E+02	1.57E+00	1.61E+02	2.19E+00	=
f_{12}	1.45E+07	6.02E+05	**1.33E+07**	**7.53E+05**	+	5.68E+05	2.22E+05	5.82E+05	2.56E+05	=	1.26E+07	6.95E+05	**1.17E+07**	**7.92E+05**	+
f_{13}	4.36E+05	6.65E+04	**2.69E+05**	**3.77E+04**	+	8.01E+01	2.83E+00	**7.32E+01**	**2.40E+00**	+	1.08E+02	5.86E+00	**8.79E+01**	**3.09E+00**	+
f_{14}	4.80E+01	1.80E-01	4.79E+01	1.57E-01	=	4.79E+01	1.95E-01	4.79E+01	1.83E-01	=	4.78E+01	1.97E-01	4.79E+01	1.56E-01	=
	Total Number of (+/=/-)				9/5/0	Total Number of (+/=/-)				6/8/0	Total Number of (+/=/-)				10/4/0

The symbols "+/=/-" indicate the statistical test results using Wilcoxon rank sum test at the 5% significance level. "+" and "-" mean that the performance of the variant with multiple exponential recombination is significantly superior or inferior than that of the variant with binomial recombination, respectively. "=" means the performance difference between the two variants is not statistically significant. "c-t-r" and "c-t-b" represent "current-to-rand" and "current-to-best", respectively.

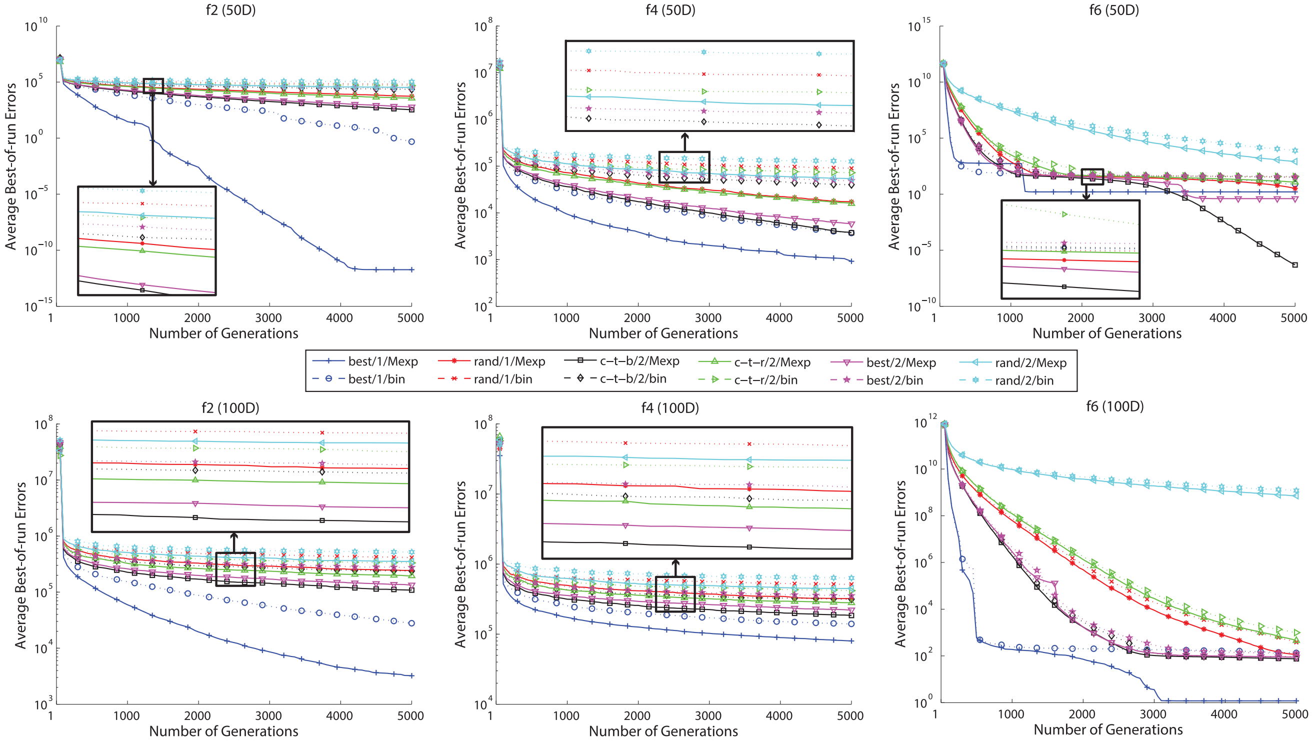


Fig. 5. Average best-of-run errors of 30 independent runs over generations. "c-t-r" and "c-t-b" represent "current-to-rand" and "current-to-best," respectively.

statistical test called Wilcoxon rank-sum test [44] is conducted at the 5% significance level. The entries which are significantly better than the counterparts are marked in boldface in Tables I and II.

From the experimental results, the variants with multiple exponential recombination exhibits superior overall performance in comparison with the binomial variants. When incorporated with DE/rand/1, DE/rand/2, DE/current-to-best/2, and

DE/current-to-rand/2, the proposed crossover operator is able to significantly improve the performance of the algorithm in most benchmark problems under both 50 dimensions and 100 dimensions. For DE/best/2, the Mexp version significantly outperforms the bin version in 9 out of 14 50-D problems and 6 out of 14 100-D problems. For DE/best/1, the proposed strategy also provides significantly better performance on 6 out 14 benchmarks in both 50-D and 100-D cases. As discussed before, the mutation probabilities are almost the same in these two crossover operations under identical Cr values. Thus, the different performances of these two operators only result from their different behaviors in selecting the exchange positions.

It is notable that for all the mutation strategies, in no benchmarks can binomial recombination show a significantly better performance than the proposed operator. To seek the reason, the specially designed mechanism enables multiple exponential recombination to be more robust than binomial recombination in tackling different types of dependent subsets. Compared to binomial recombination, multiple exponential recombination not only handle the distant variables similarly but also possess a higher chance to preserve the physically proximate dependent subsets. Among all tested problems, only f_1 and f_9 are separable problems. A very interesting phenomenon is that even for these two separable problems, multiple exponential recombination is able to significantly outperform traditional binomial recombination in most situations. The explanation is that some favorable variable configurations may form during the optimization process and multiple exponential recombination is more likely to preserve these structures when the involved variables are physically proximate to each other.

To give a clearer picture of the performance difference between the two crossover operators, Fig. 5 plots the change of best-of-run errors over generations for each algorithm on six problems. It can be observed that all the Mexp variants converge toward the global optimum faster than the bin counterparts during the entire optimization process. A more effective search is achieved by multiple exponential recombination in solving these problems. The different convergence speeds of each mutation strategy are beyond the scope of our discussion.

C. Implementation in SaDE

In this section, the proposed multiple exponential recombination is implemented in SaDE [4], which is a very powerful state-of-the-art DE variant. The original SaDE will self-adapt the control parameter values and mutation strategies by learning from the successful experiences in generating promising solutions. However, the crossover operator in the original SaDE is fixed as binomial recombination due to its popularity in other DE literatures [4]. During the experiments, multiple exponential recombination will replace binomial recombination in crossover operation and all the other mechanisms of the original SaDE will be reserved. The performance of the modified SaDE will be compared with that of the original SaDE.

TABLE III
MEAN AND STANDARD DEVIATION OF THE ERRORS (30 RUNS)

Problems	Original SaDE	SaDE/Mexp
(Dimensions)	Mean ± Std	Mean± Std
f_1 (50)	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00
f_2 (50)	3.20E-10 ± 4.39E-10	**4.36E-16 ± 7.91E-16**
f_3 (50)	8.87E+04 ± 3.47E+04	8.97E+04 ± 4.32E+04
f_4 (50)	7.48E+02 ± 7.22E+02	**3.13E+01 ± 4.15E+01**
f_5 (50)	3.57E+03 ± 5.87E+02	3.52E+03 ± 6.40E+02
f_6 (50)	6.05E+00 ± 1.55E+01	**1.99E+00 ± 2.03E+00**
f_7 (50)	5.66E-03 ± 1.04E-02	**3.44E-03 ± 8.16E-03**
f_8 (50)	2.11E+01 ± 4.94E-02	2.11E+01 ± 3.60E-02
f_9 (50)	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00
f_{10} (50)	9.39E+01 ± 1.38E+01	9.28E+01 ± 1.37E+01
f_{11} (50)	3.43E+01 ± 9.58E+00	3.27E+01 ± 9.26E+00
f_{12} (50)	9.60E+03 ± 6.05E+03	**7.17E+03 ± 7.25E+03**
f_{13} (50)	9.34E+00 ± 6.13E-01	**8.46E+00 ± 5.77E-01**
f_{14} (50)	2.24E+01 ± 1.91E-01	2.24E+01 ± 2.38E-01
f_1 (100)	1.68E-30 ± 9.22E-30	1.68E-30 ± 9.22E-30
f_2 (100)	3.28E-03 ± 3.21E-03	**6.59E-05 ± 9.53E-05**
f_3 (100)	9.43E+05 ± 2.48E+05	9.10E+05 ± 3.00E+05
f_4 (100)	3.50E+04 ± 6.25E+03	**1.98E+04 ± 4.94E+03**
f_5 (100)	9.14E+03 ± 1.29E+03	9.48E+03 ± 1.31E+03
f_6 (100)	4.88E+01 ± 3.25E+01	**2.52E+00 ± 1.95E+00**
f_7 (100)	5.25E-03 ± 7.33E-03	4.60E-03 ± 6.65E-03
f_8 (100)	2.13E+01 ± 2.17E-02	2.13E+01 ± 2.50E-02
f_9 (100)	1.33E+00 ± 1.09E+00	1.53E+00 ± 1.22E+00
f_{10} (100)	3.14E+02 ± 3.79E+01	3.12E+02 ± 3.97E+01
f_{11} (100)	8.86E+01 ± 5.96E+00	8.86E+01 ± 7.10E+00
f_{12} (100)	4.20E+04 ± 1.63E+04	**3.67E+04 ± 2.56E+04**
f_{13} (100)	2.94E+01 ± 1.39E+00	**2.58E+01 ± 1.81E+00**
f_{14} (100)	4.67E+01 ± 2.86E-01	4.67E+01 ± 2.87E-01

The purpose of this implementation is to further investigate the behaviors of the proposed operator when incorporated with self-adaptive DE variant. Both the 50-D and 100-D versions of f_1–f_{14} from CEC-05 benchmark suite will be tested. For both algorithms, the population size is fixed as 100 and the maximum number of fitness evaluation is set to 50×10^4 for 50-D problems and 100×10^4 for 100-D problems. T is fixed as 10 for multiple binomial recombination. The Cr value used in the two crossover operators will be adapted via the parameter adaptation mechanism in SaDE. For all the remaining parameters in SaDE, the recommended parametric setup in the original literature [4] is utilized. Table III presents the mean and standard deviation of the best-of-run errors over 30 independent runs for each algorithm on each problem. Wilcoxon rank-sum test is conducted at the 5% significance level for each pair of results. The entries that significantly outperforms the counterparts are marked in boldface.

From Table III, the modified SaDE variant with multiple exponential recombination provides significantly better performance than the original version with binomial recombination in 11 out of 28 benchmarks. In all the remaining 17 problems, the performance difference between the two variants is not statistically significant. This observation indicates that replacing binomial recombination with multiple exponential recombination has enhanced the robustness of SaDE in solving problems with different types of variable interrelations.

Subsequently, the behaviors of the parameter adaptation mechanism in tuning Cr values are studied. In SaDE,

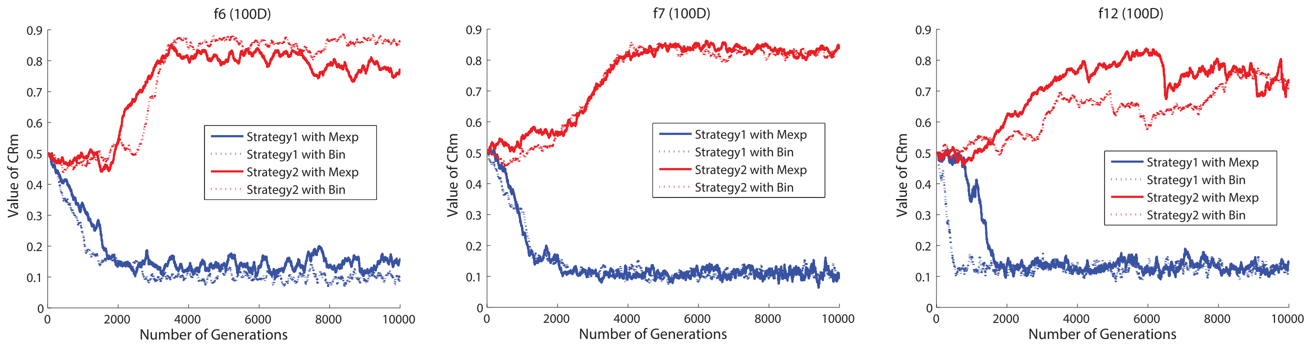


Fig. 6. Adaptation behaviors of the CRm values in SaDE. Strategies 1 and 2 are the first two default candidate mutations strategies in original SaDE, namely, DE/rand/1 and DE/current-to-best/2.

the adaptation of Cr value is based on a normal distribution with mean value CRm (please note this is the control parameter in the original SaDE, it is different from our control parameter Cr_m) and standard deviation 0.1. Each mutation strategy in the candidate pool will have one associated CRm and the value of this CRm is adapted according to previous Cr values that have generated promising trial vectors. Fig. 6 plots the change of CRm values during optimization process for the first two mutation strategies in both modified and original SaDE on 100-D f_6, f_7, and f_{12}. According to Fig. 6, changes of CRm values in both operators show similar pattern. For strategy 1, both the operator tends to preserve higher CRm values whereas in strategy 2, lower CRm values are preferred by both crossover operations. Considering that the Cr values are generated based on CRm, the distribution of actually used Cr values will be similar in both variants. Thus, the performance difference between the two variants are not caused by utilizing different Cr values. All the improvements in the Mexp variants are mainly due to the new mechanisms in selecting crossover positions. Another observation is that either the value above 0.8 or below 0.2 is preferred by the Cr adaptation mechanisms in SaDE. Based on the simulation results in Fig. 4, when the Cr value is close to 0 or 1, the disruption frequency distribution will become similar for both crossover operators. As a result, the performance difference between the two SaDE variants may not be significant in some benchmark problems.

V. Conclusion

This paper has presented a new DE crossover operator, in which multiple segments will be exchanged among involved vectors. The specially designed mechanisms enable the proposed strategy to inherit all the main advantages of existing crossover operators while providing a more robust performance in handling variable interrelations. The properties of the new operator is investigated both theoretically and empirically. Implementation in six classical DE algorithms and one adaptive DE variant demonstrates the superiority of the proposed strategy over binomial recombination in solving problems with unknown variable interrelations.

In future work, a parameter adaptation mechanism exclusively for multiple exponential recombination is expected to be developed. In addition, design of new benchmark problems with different variable dependency configurations will be considered. Finally, the proposed method may be applied to solve some real world optimization problems.

References

[1] R. Storn and K. V. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Golb. Optim.*, vol. 11, no. 4, pp. 341–359, Dec. 1997.

[2] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 4–31, Feb. 2011.

[3] J. Brest, S. Greiner, B. Boskovic, M. Mernik, and V. Zumer, "Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems," *IEEE Trans. Evol. Comput.*, vol. 10, no. 6, pp. 646–657, Dec. 2006.

[4] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 398–417, Apr. 2009.

[5] J. Zhang and A. Sanderson, "JADE: Adaptive differential evolution with optional external archive," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 945–958, Oct. 2009.

[6] S. Das, A. Abraham, U. K. Chakraborty, and A. Konar, "Differential evolution using a neighborhood-based mutation operator," *IEEE Trans. Evol. Comput.*, vol. 13, no. 3, pp. 526–553, Jun. 2009.

[7] M. G. Epitropakis, D. K. Tasoulis, N. G. Pavlidis, V. P. Plagianakos, and M. N. Vrahatis, "Enhancing differential evolution utilizing proximity-based mutation operators," *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 99–119, Feb. 2011.

[8] S. M. Islam, S. Das, S. Ghosh, S. Roy, and P. N. Suganthan, "An adaptive differential evolution algorithm with novel mutation and crossover strategies for global numerical optimization," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 482–500, Apr. 2012.

[9] H. Wang, S. Rahnamayan, H. Sun, and M. G. H. Omran, "Gaussian bare-bones differential evolution," *IEEE Trans. Cybern.*, vol. 43, no. 2, pp. 634–647, Apr. 2013.

[10] Q. Fan and X. Yan, "Self-adaptive differential evolution algorithm with zoning evolution of control parameters and adaptive mutation strategies," *IEEE Trans. Cybern.*, vol. 46, no. 1, pp. 219–232, Jan. 2016.

[11] W. Gong and Z. Cai, "Differential evolution with ranking-based mutation operators," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 2066–2081, Dec. 2013.

[12] R. A. Sarker, S. M. Elsayed, and T. Ray, "Differential evolution with dynamic parameters selection for optimization problems," *IEEE Trans. Evol. Comput.*, vol. 18, no. 5, pp. 689–707, Oct. 2014.

[13] L. Tang, Y. Zhao, and J. Liu, "An improved differential evolution algorithm for practical dynamic scheduling in steelmaking-continuous casting production," *IEEE Trans. Evol. Comput.*, vol. 18, no. 2, pp. 209–225, Apr. 2014.

[14] S. Biswas, S. Kundu, and S. Das, "Inducing niching behavior in differential evolution through local information sharing," *IEEE Trans. Evol. Comput.*, vol. 19, no. 2, pp. 246–263, Apr. 2015.

[15] S. Bandyopadhyay and A. Mukherjee, "An algorithm for many-objective optimization with reduced objective computations: A study in differential evolution," *IEEE Trans. Evol. Comput.*, vol. 19, no. 3, pp. 400–413, Jun. 2015.

[16] D. Zaharie, “Influence of crossover on the behavior of differential evolution algorithms,” *Appl. Soft Comput.*, vol. 9, no. 3, pp. 1126–1138, Jun. 2009.

[17] E. Mezura-Montes, J. Velázquez-Reyes, and C. A. C. Coello, “A comparative study of differential evolution variants for global optimization,” in *Proc. 8th Annu. Conf. Genet. Evol. Comput. (GECCO)*, Seattle, WA, USA, 2006, pp. 485–492.

[18] C. Lin, A. Qing, and Q. Feng, “A comparative study of crossover in differential evolution,” *J. Heuristics*, vol. 17, no. 6, pp. 675–703, Dec. 2011.

[19] D. R. Newman. (Sep. 2006). *The Use of Linkage Learning in Genetic Algorithms*. [Online]. Available: http://www.ecs.soton.ac.uk/~drn05r/ug/irp/.

[20] Y. Chen, T. Yu, K. Sastry, and D. Goldberg, “A survey of linkage learning techniques in genetic and evolutionary algorithms,” Illinois Genet. Algorithms Lab., Univ. Illinois Urbana–Champaign, Champaign, IL, USA, Tech. Rep. 2007014, 2007.

[21] Y.-L. Li *et al.*, “Differential evolution with an evolution path: A deep evolutionary algorithm,” *IEEE Trans. Cybern.*, vol. 45, no. 9, pp. 1798–1810, Sep. 2015.

[22] U. Halder, S. Das, and D. Maity, “A cluster-based differential evolution algorithm with external archive for optimization in dynamic environments,” *IEEE Trans. Cybern.*, vol. 43, no. 3, pp. 881–897, Jun. 2013.

[23] W.-F. Gao, G. G. Yen, and S.-Y. Liu, “A dual-population differential evolution with coevolution for constrained optimization,” *IEEE Trans. Cybern.*, vol. 45, no. 5, pp. 1108–1121, May 2015.

[24] S. Das, A. Mandal, and R. Mukherjee, “An adaptive differential evolution algorithm for global optimization in dynamic environments,” *IEEE Trans. Cybern.*, vol. 44, no. 6, pp. 966–978, Jun. 2014.

[25] M. Yang, C. Li, Z. Cai, and J. Guan, “Differential evolution with auto-enhanced population diversity,” *IEEE Trans. Cybern.*, vol. 45, no. 2, pp. 302–315, Feb. 2015.

[26] Y. Cai and J. Wang, “Differential evolution with neighborhood and direction information for numerical optimization,” *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 2202–2215, Dec. 2013.

[27] J. Wang, J. Liao, Y. Zhou, and Y. Cai, “Differential evolution enhanced with multiobjective sorting-based mutation operators,” *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2792–2805, Dec. 2014.

[28] L. Tang, Y. Dong, and J. Liu, “Differential evolution with an individual-dependent mechanism,” *IEEE Trans. Evol. Comput.*, vol. 19, no. 4, pp. 560–574, Aug. 2015.

[29] W. Gong, Z. Cai, and D. Liang, “Adaptive ranking mutation operator based differential evolution for constrained optimization,” *IEEE Trans. Cybern.*, vol. 45, no. 4, pp. 716–727, Apr. 2015.

[30] S. Biswas, S. Kundu, and S. Das, “An improved parent-centric mutation with normalized neighborhoods for inducing niching behavior in differential evolution,” *IEEE Trans. Cybern.*, vol. 44, no. 10, pp. 1726–1737, Oct. 2014.

[31] Y.-J. Gong, Q. Zhou, Y. Lin, and J. Zhang, “Orthogonal predictive differential evolution,” in *Proc. 18th Asia Pac. Symp. Intell. Evol. Syst.*, vol. 1. Singapore, 2015, pp. 141–154.

[32] W.-J. Yu *et al.*, “Differential evolution with two-level parameter adaptation,” *IEEE Trans. Cybern.*, vol. 44, no. 7, pp. 1080–1099, Jul. 2014.

[33] W. Gao, G. G. Yen, and S. Liu, “A cluster-based differential evolution with self-adaptive strategy for multimodal optimization,” *IEEE Trans. Cybern.*, vol. 44, no. 8, pp. 1314–1327, Aug. 2014.

[34] R. Storn and K. V. Price, “Differential evolution—A simple and efficient adaptive scheme for global optimization over continuous spaces,” Int. Comput. Sci. Inst., Berkeley, CA, USA, Tech. Rep. TR-95-012, 1995.

[35] J. Tvrdik, “Adaptive differential evolution and exponential crossover,” in *Proc. Int. Multiconf. Comput. Sci. Inf. Technol. (IMCSIT)*, Wisła, Poland, Oct. 2008, pp. 927–931.

[36] G. Jeyakumar and C. Shanmugavelayutham, “Convergence analysis of differential evolution variants on unconstrained global optimization functions,” *Int. J. Artif. Intell. Appl.*, vol. 2, no. 2, pp. 116–126, Apr. 2011.

[37] G. Jeyakumar and C. Shanmugavelayutham, “A comparative performance analysis of differential evolution and dynamic differential evolution variants,” in *Proc. World Congr. Nat. Biol. Inspired Comput. (NaBIC)*, Coimbatore, India, Dec. 2009, pp. 463–468.

[38] D. Zaharie, “A comparative analysis of crossover variants in differential evolution,” in *Proc. Int. Multiconf. Comput. Sci. Inf. Technol.*, Wisła, Poland, 2007, pp. 171–181.

[39] S.-M. Guo and C.-C. Yang, “Enhancing differential evolution utilizing eigenvector-based crossover operator,” *IEEE Trans. Evol. Comput.*, vol. 19, no. 1, pp. 31–49, Feb. 2015.

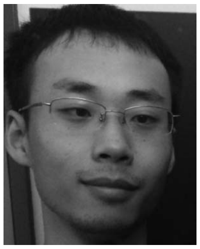
[40] S.-Z. Zhao and P. N. Suganthan, “Empirical investigations into the exponential crossover of differential evolutions,” *Swarm Evol. Comput.*, vol. 9, pp. 27–36, Apr. 2013.

[41] K. De Jong and W. M. Spears, “A formal analysis of the role of multi-point crossover in genetic algorithms,” *Ann. Math. Artif. Intell.*, vol. 5, no. 1, pp. 1–26, 1992. [Online]. Available: http://dx.doi.org/10.1007/BF01530777.

[42] A. A. Markov, “Extension of the limit theorems of probability theory to a sum of variables connected in a chain,” in *Dynamic Probabilistic Systems, Volume 1: Markov Chains*, vol. 1. New York, NY, USA: Wiley, 1971.

[43] P. N. Suganthan *et al.*, “Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization,” School Elect. Electron. Eng., Nanyang Technol. Univ., Singapore, Tech. Rep. 2005005, May 2005.

[44] F. Wilcoxon, “Individual comparisons by ranking methods,” *Biometrics Bull.*, vol. 1, no. 6, pp. 80–83, Dec. 1945.



Xin Qiu received the B.E. degree in electrical engineering from Nanjing University, Nanjing, China, in 2012. He is currently pursuing the Ph.D. degree with the Computational Intelligence Research Group, National University of Singapore, Singapore.

His current research interests include differential evolution, multiobjective optimization, and applications of evolutionary algorithms.

Kay Chen Tan (SM’08–F’14) received the B.Eng. (First Class Hons.) degree in electronics and electrical engineering, and the Ph.D. degree from the University of Glasgow, Glasgow, Scotland, in 1994 and 1997, respectively.

He is an Associate Professor with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. He has published over 100 journal papers, over 100 papers in conference proceedings, and co-authored five books. His current research interest include computational and artificial intelligence, with applications to multiobjective optimization, scheduling, automation, data mining, and games.

Dr. Tan was a recipient of the 2012 IEEE Computational Intelligence Society Outstanding Early Career Award for his contributions to evolutionary computation in multiobjective optimization. He was an Editor-in-Chief of the *IEEE Computational Intelligence Magazine* from 2010 to 2013. He is currently an Editor-in-Chief of the IEEE Transactions on Evolutionary Computation. He serves as an Associate Editor/Editorial Board Member of over 15 international journals, including the IEEE Transactions on Cybernetics, the IEEE Transactions on Computational Intelligence and AI in Games, *Evolutionary Computation* (MIT Press), the *European Journal of Operational Research*, the *Journal of Scheduling*, and the *International Journal of Systems Science*.

Jian-Xin Xu (M’92–SM’98–F’11) received the Ph.D. degree from University of Tokyo, Tokyo, Japan, in 1989.

In 1991, he joined the National University of Singapore, where he is currently a Professor with the Department of Electrical Engineering. He has published over 200 journal papers and six books in the field of system and control, supervised 30 Ph.D. students and 15 research associates, as well as conducted close to 30 research grants. His current research interests include learning theory, intelligent system and control, nonlinear and robust control, robotics, and precision motion control.