Improving Differential Evolution with Ring Topology-Based Mutation Operators

Jingliang Liao, Yiqiao Cai*, Yonghong Chen, Tian Wang and Hui Tian

College of Computer Science and Technology
Huaqiao University, Xiamen, China
Email: caiyq@hqu.edu.cn
**corresponding author*

***Abstract*—Differential evolution (DE) has been proven to be a simple and powerful evolutionary algorithm, and obtains many successful applications in scientific and engineering fields. The mutation strategy plays the key role in DE for finding global optimal solutions. In most of the DE algorithms, the base and difference vectors are randomly selected from the current population, and both the neighborhood and direction information are not fully and simultaneously exploited in the evolutionary process. In order to alleviate this drawback and enhance the performance of DE, we use the ring topology to construct neighborhood for each individual and then introduce the direction information with the neighbors into the mutation operator of DE. The proposed DE is named as ring-DE in this paper. By this way, ring-DE can utilize the neighborhood and direction information simultaneously to guide the search of DE. In order to evaluate the effectiveness of the proposed method, ring-DE is incorporated into several original DE algorithms. Experimental results clearly show that ring-DE is able to enhance the performance of the DE algorithms studied.**

Keywords- Differential evolution, neighborhood information, direction information, mutation strategy

I. INTRODUCTION

Differential evolution (DE), which was firstly proposed by Storn and Price [1], is a simple yet powerful evolutionary algorithm for global numerical optimization. It has many attractive characteristics, such as ease to use, simple structure, speediness and robustness. Recently, DE has been extended to handle multi-objective, constrained, large-scale, dynamic, and uncertain optimization problems [2]. DE has been successfully applied in various scientific and engineering fields [2], such as chemical engineering, engineering design, pattern recognition, and so on.

In DE, there exist two main factors which significantly influence the optimization performance of the DE. One is the control parameters, i.e., population size *NP*, scaling factor *F*, and crossover rate *CR*, and the other is the evolutionary operators, i.e., mutation, crossover, and selection. In the mutation operator, a mutant vector can be treated as the lead individual to explore the search space and generated by adding a difference vector to a base vector. We have observed, however, that these two vectors (i.e., base and difference vectors) in most of DE are selected either randomly or locally, which does not utilize the neighborhood or direction information of population to guide the search.

In order to alleviate this drawback and enhance the performance of DE, we propose a new DE framework with ring topology based mutation operator. Using ring topology, we introduce the neighborhood information into DE by constructing neighborhood for each individual. Then, we partition the neighbors of each vector into better set and worse set according to their fitness compared to that of it. Based on the grouping of neighbors, we introduce the direction information into mutation by selecting the vectors from better and worse set respectively to form the difference vector, with respect to the base vector. Finally, a simple and effective DE framework, ring-DE, is proposed. In this way, ring-DE not only utilizes the information of neighboring individuals to exploit the regions of minima and accelerate convergence but also incorporates the direction information of population to prevent individuals from entering an undesired region and move to a promising area.

To evaluate the effectiveness of the proposed method, ring-DE is applied to four famous and widely used original DE algorithms, i.e., DE/rand/1, DE/rand/2, DE/best/1, and DE/current-to-best/1. Extensive experiments have been carried out on a set of benchmark functions. The results show that ring-DE is able to enhance the performance of the DE algorithms studied.

The rest of this paper is organized as follows: In Section II, the original DE is introduced. Section III briefly reviews some related work. The proposed ring-DE is presented in detail in Section IV. In Section V, experimental results are reported. Finally, the conclusions are drawn in Section VI.

II. DE

In this paper, DE is for solving the numerical optimization problem. Without loss of generality, we consider the optimization problem to be minimized is $f(\mathrm{X})$, $X = [\mathrm{x}^1, \mathrm{x}^2, ..., \mathrm{x}^D] \in \mathrm{R}^D$ and D is the dimension of the decision variables. DE evolves a population of *NP* vectors representing the candidate solutions. Each vector is denoted as $X_{i,G} = [\mathrm{x}^1_{i,G}, \mathrm{x}^2_{i,G}, ..., \mathrm{x}^D_{i,G}]$, where i = 1, 2, …, *NP*, *NP* is the size of the population and G is the number of current generation.

Initialization: In DE, the initial population should cover the entire search space as much as possible by uniformly randomizing individuals within the search space constrained by the prescribed minimum and maximum bounds. That is, the *j*th parameter of the *i*th individual is initialized by

$$x^j_{i,G} = L_j + rand(0,1) \times (U_j - L_j) \quad (1)$$

where $rand(0,1)$ represents a uniformly distributed random number within the range [0,1] and L_j and U_j represents the lower and upper bounds of the *j*th variable respectively.

Mutation: After initialization, DE employs the mutation strategy to generate a mutant vector $V_{i,G}$ with respect to each individual $X_{i,G}$ (called target vector) in the current population. For example, the four most frequently used mutation strategies in the literature are listed as follows:

- DE/rand/1

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

- DE/rand/2

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

- DE/best/1

$$V_{i,G} = X_{\text{best},G} + F \times (X_{r1,G} - X_{r2,G}) \tag{4}$$

- DE/current-to-best/1

$$V_{i,G} = X_{\text{i},G} + F \times (X_{\text{best},G} - X_{\text{i},G}) + F \times (X_{r1,G} - X_{r2,G}) \tag{5}$$

The indices *r1*, *r2*, *r3*, *r4*, *r5* are mutually exclusive integers randomly generated within the range [1, *NP*], which are also different from the index i. $X_{\text{best},G}$ is the best individual vector at generation *G*, and the mutation factor *F* is a positive control parameter for scaling the difference vector. Their more details can be found in [1] and [2].

Crossover: After the mutation phase, crossover operator is applied to each pair of $X_{i,G}$ and $V_{i,G}$ to generate a trial vector $U_{i,G}$. There are two kinds of crossover scheme: binomial and exponential. The binomial crossover is widely used, which can be defined as follows:

$$u_{i,G}^{j} = \begin{cases} v_{i,G}^{j} & \text{if } rand(0,1) \le CR \text{ or } j = j_{rand}; \\ x_{i,G}^{j} & \text{otherwise,} \end{cases} \tag{6}$$

where $CR \in [0, 1]$ is called the crossover rate. j_{rand} is a randomly chosen integer in the range [1, *D*]. If $u_{i,G}^{j}$ is out of the boundary, we reinitialized it within the range $[L_j, U_j]$.

Selection: The selection operation selects the better one from each pair of $X_{i,G}$ and $U_{i,G}$ according to their fitness values for the next generation. For example, the selection operator is given by

$$X_{i,G+1} = \begin{cases} U_{i,G} & \text{if } f(\text{U}_{i,G}) \le f(\text{X}_{i,G}); \\ X_{i,G} & \text{otherwise.} \end{cases} \tag{7}$$

III. RELATED WORK

Early researchers realized that the structure of the populations can influence the evolution of the population. In the section, we focus on the related work on how the neighborhood and direction information has been utilized in DE to improve its performance.

A. *Neighborhood Information*

The neighborhood concepts are usually used to improve the performance of DE. There are two main types of neighborhood information: one relies on the population topology and the other on the geographical locations on the fitness landscape. More details about the neighborhood concepts utilized in DE could be found in [3].

In the first one, the neighbors of each individual do not necessary lie in the vicinity of its topological region in the search space. Different from the original DE algorithm, many DE variants utilize the neighborhood information with the structured population. In these DE variants, the individuals for the mutation strategies are selected according to a neighbor list constructed from the population topologies [4]. In [5], the performance of the self-adaptive DE is improved by using a ring neighborhood topology.

The second kind of neighborhood information is derived from the current population during the evolutionary process. In this, we name some individuals be the neighbors of one individual when they locate in the vicinity of its topological region in the search space. In [3], the authors proposed a proximity-based DE framework (ProDE) based on the proximity characteristics among the individual solutions as they evolve. In ProDE, they select the individuals for the mutation strategies by using an affinity matrix based on the Euclidean distance. For improving the performance of DE, the learning-enhanced DE (LeDE) is proposed in [6]. In LeDE, the neighborhoods of each individual involved in the intra-cluster learning strategy are defined based on the identified clusters.

B. *Direction Information*

In DE, the difference vector of the mutation strategies is important to guide the search, and it always be constructed in a random manner. The related work of the direction information used for constructing the difference vector will be briefly introduced below.

In [7], a new mutation strategy, which is identified as DE/rand/±mean, is proposed. In this strategy, the population is partitioned into two sub-populations according to the mean fitness value of all individuals. Then two vectors are randomly selected from the better sub-population and the worse one respectively to generate the different vector. Recently, the direction information is derived from two sources, namely, the best and worst near-neighbor individuals in [8]. Then, three types of direction information based on the direction information with different sources are introduced to guide search. Specifically, directional attraction (DA) enhances the ability of exploitation, directional repulsion (DR) encourages the individuals to explore more areas, and directional convergence (DC) can accelerate convergence of population.

IV. RING–DE

As mentioned above, both neighborhood information and direction information can be utilized to improve the performance of DE, but they are not fully and simultaneously exploited in the evolutionary process of DE. Thus, instead of selecting the base and difference vectors randomly, we employ the neighborhood and direction information simultaneously to design a new mutation operator based on the ring topology for improving the performance of DE.

A. *Ring topology-based neighborhors*

With the ring-topology, all vectors of population are organized on a ring topology with respect to their indices. In this way, X_1 is next to X_{NP}, and X_{NP} is prior to X_1. For each individual X_i, its sub-population is constructed based on the ring topology by selecting the individuals X_{i-R}, …, X_i, …, X_{i+R}, where $R = p * NP$, *p* is the sub-population radius proportion and in this paper we set $p = 0.1$. After that, each individual of population is with 2*R* neighbors.

B. *Difference vector with direction information*

With the ring topology-based neighbors, the direction information is incorporated into mutation by selecting the vectors from the neighbors to form the difference vectors according to the base vector. Specifically, with respect to the fitness of the base vector, we partition the sub-population into a better set and a worse set. Then, the first component vector of the difference vector (e.g. X_{r2} for DE/rand/1 in Eq. (2)) is selected from the better set and the second one (e.g. X_{r3}) is selected from the worse set. Such that, a difference vector with direction information directing at the better solution vector from the worse one can be obtained for guiding search.

In the application of ring-DE, the base and difference vector based on the ring-topology for different mutation operators are constructed as follow.

1) DE/rand/1 and DE/rand/2

X_{r1} will be selected randomly from the sub-population of the target vector, and the sub-population is partitioned according to the fitness value of the individual X_{r1}. Then, the difference vector is constructed with the two defined sets.

2) DE/best/1 and DE/current-to-best/1

These two mutation strategies incorporate the best individual, which have a fast convergence speed but are likely to lose diversity of the population and trap in the local optima. In ring-DE, the best neighbor in the sub-population of the target individual is set as the best individual. For DE/best/1, two vectors are randomly selected from the sub-population and the difference vector is constructed by directing the search from the worse vector to the better vector. Different from DE/best/1, DE/current-to-best/1 partitions the sub-population based on the current individual.

C. *The framework of ring-DE*

We use the notation ring-DE/rand/1 to designate that the proposed ring-DE framework is used in the original strategy DE/rand/1. The pseudocode of ring-DE/rand/1 is shown in **Algorithm 1** where the differences with respect to DE/rand/1 are highlighted with "*". It is clear that the proposed ring-DE framework affects only the mutation stage, hence it could be directly and easily applied to any DE mutation strategy.

V. Simulation Results

In order to evaluate the performance of ring-DE, 25 benchmark functions are chosen from the CEC2005 test suite [9]. In this section, the benchmark functions are presented firstly. Secondly, the experimental setup is shown. Thirdly, the simulation results are analyzed and discussed.

A. *Benchmark Functions*

In this section, 25 benchmark functions are used, which are denoted as F1-F25, which are from the special session on real-parameter optimization of the 2005 IEEE Congress on evolutionary computation (CEC2005) [9]. They can be categorized into four groups: unimodal functions (F1-F5), basic multimodal functions (F6-F12), expanded multimodal functions (F13-F14) and hybrid composition functions (F15-F25). More details of them can be found in [9].

Algorithm 1 ring-DE/rand/1

```
Generate the initial population P and set G = 1, p = 0.1;
Evaluate the fitness for each individual in P;
While the terminated condition is not satisfied do
    For each individual X_{i,G} do
        *Randomly select the base vector X_{r1} from the
         according sub-population;
        *Partition the sub-population into better and worse
         sets, then select X_{r2}, X_{r3} from the better and
         worse sets respectively to construct the
         difference vector;
        Use Eq. (2) to generate a mutant vector
        Use Eq. (6) to generate a trial vector;
        Use Eq. (7) to determine the survived vector;
    End For
    Set G = G + 1
End while
```

B. *Experimental Setup*

In order to compare the performance between ring-based DE and its corresponding original DE, we use the same random initial population, and the parameters for all the experiments are set as follows unless a change is mentioned.

- Dimension of each function: D = 30 and D = 50;
- Population size: NP = 100;
- Mutation factor: F = 0.5;
- Crossover rate: CR = 0.9;
- Sub-population radius proportion: p = 0.1;
- Number of runs: $NumR$ = 25;
- Maximum number of function evaluations: $MNFEs$ $=10000\times D$.

In order to show the significant differences among the algorithms, several nonparametric statistical tests [10] are also carried out by the KEEL software [11].

C. *Comparison of Quality of the Final Solution*

In this section, four mutation operators (see Eq. (2) - (5)) are used in the experimental study. Among these four mutation operators, three of them have one difference vectors, while the rest one has two difference vectors. Normally, the mutation operators with two difference vectors are more explorative. Firstly, we evaluate the quality of the final solution of our proposed ring-based mutation operators in DE.

The results for all functions at D = 30 and D = 50 are shown in Table I and Table II. The better values compared between DE and its corresponding ring-DE are highlighted in boldface. In order to compare the significance between two algorithms the paired Wilcoxon signed-rank test is used. In Table I, according to the Wilcoxon's test, the results are summarized as "*w/t/l*", which denotes that our proposed ring-DE wins in *w* functions, ties in *t* functions, and loses in *l* functions, compared with its corresponding original DE method. In addition, the multiple-problem statistical analysis based on the Wilcoxon's test between ring-based DE and its corresponding original DE is reported for all functions in Tables III and IV, respectively.

For all functions at D = 30, Table I shows that in the majority of the test functions the ring-based DE methods provide significantly better results compared with their corresponding original DE methods. For example, with DE/rand/1/bin strategy, ring-DE significantly improves the performance of DE in 12 out of 25 functions, but only loses in 4 functions. With DE/rand/2/bin strategy, ring-DE wins in 22 functions, ties in 3 functions compared with DE. There is no function that DE can significantly outperform ring-DE.

With DE/best/1/bin strategy, ring-DE wins in 22 functions, ties in 2 functions, and only loses in 1 function compared with DE. With DE/current-to-rand/1/bin strategy, ring-DE wins in 20 functions, ties in 4 functions, and only loses in 1 functions according to the Wilcoxon's test results at α = 0.05. Additionally, according to the results of multiple-problem statistical analysis shown in Table III we can see that ring-based DEs consistently get higher $R+$ values than $R-$ values in all cases compared with the original DEs. This means that the ring-based DE is better than its original DE for all functions.

TABLE I.
ERROR VALUES OF THE ORIGINAL DE MUTATION STRATEGIES AND THEIR CORRESPONDING RING-BASED VARIANTS OVER THE 30-DIMENSIONAL CEC 2005 BENCHMARK SET

	DE/rand/1		**ring-DE/rand/1**	**DE/rand/2**		**ring-DE/rand/2**
F1	**0.00e+000 0.00e+000**	-	1.28e-028 1.35e-028	8.65e-001 3.60e-001	+	**6.20e-016 5.28e-016**
F2	7.13e-005 5.73e-005	+	**6.87e-013 1.81e-012**	7.05e+003 1.38e+003	+	**2.10e-001 1.03e-001**
F3	4.45e+005 2.60e+005	+	**1.13e+005 5.92e+004**	4.94e+007 1.22e+007	+	**4.83e+005 1.73e+005**
F4	2.12e-002 1.70e-002	+	**8.38e-006 2.65e-005**	1.43e+004 2.27e+003	+	**1.88e+001 9.62e+000**
F5	**6.65e+001 7.38e+001**	-	4.75e+002 3.31e+002	7.94e+003 7.54e+002	+	**5.77e+002 1.30e+002**
F6	**2.80e+000 1.63e+000**	-	2.39e+001 1.53e+001	4.53e+003 1.74e+003	+	**7.76e+000 2.70e+000**
F7	**3.94e-004 1.97e-003**	-	2.12e-002 1.65e-002	3.61e+000 8.37e-001	+	**9.86e-004 2.76e-003**
F8	2.10e+001 4.30e-002	=	**2.09e+001 5.47e-002**	2.10e+001 4.74e-002	=	**2.10e+001 3.85e-002**
F9	1.30e+002 2.76e+001	+	**2.86e+001 9.62e+000**	2.13e+002 8.56e+000	+	**1.91e+002 9.88e+000**
F10	1.79e+002 1.18e+001	+	**3.62e+001 1.01e+001**	2.38e+002 1.40e+001	+	**2.03e+002 1.04e+001**
F11	3.96e+001 1.42e+000	+	**2.74e+001 6.53e+000**	3.94e+001 1.03e+000	=	**3.92e+001 1.34e+000**
F12	**1.32e+003 1.79e+003**	=	2.65e+003 3.38e+003	5.17e+005 5.06e+004	+	**1.45e+004 3.41e+004**
F13	1.51e+001 9.53e-001	+	**3.23e+000 8.09e-001**	2.02e+001 1.26e+000	+	**1.62e+001 1.47e+000**
F14	1.33e+001 1.44e-001	+	**1.31e+001 1.54e-001**	**1.34e+001 1.53e-001**	=	1.34e+001 1.72e-001
F15	4.04e+002 2.00e+001	=	**3.93e+002 6.33e+001**	4.09e+002 2.14e+001	+	**3.96e+002 2.00e+001**
F16	2.05e+002 9.51e+000	+	**5.99e+001 1.64e+001**	2.66e+002 1.22e+001	+	**2.27e+002 1.33e+001**
F17	2.28e+002 1.95e+001	+	**5.60e+001 9.96e+000**	2.98e+002 1.71e+001	+	**2.50e+002 1.03e+001**
F18	8.97e+002 2.91e+001	=	**8.80e+002 5.08e+001**	9.41e+002 4.21e+000	+	**9.03e+002 2.15e+001**
F19	8.92e+002 3.46e+001	=	**8.80e+002 5.08e+001**	9.38e+002 4.23e+000	+	**9.08e+002 1.26e+000**
F20	9.00e+002 2.09e+001	=	**8.85e+002 4.86e+001**	9.40e+002 3.71e+000	+	**8.99e+002 2.97e+001**
F21	5.00e+002 0.00e+000	=	5.00e+002 0.00e+000	5.00e+002 9.41e-002	+	**5.00e+002 1.53e-005**
F22	9.08e+002 9.46e+000	=	**9.06e+002 1.14e+001**	1.02e+003 1.87e+001	+	**9.22e+002 7.13e+000**
F23	5.34e+002 9.99e-005	+	**5.34e+002 1.13e-002**	5.35e+002 1.77e+000	+	**5.34e+002 1.41e-004**
F24	2.00e+002 0.00e+000	=	2.00e+002 0.00e+000	2.00e+002 1.41e-001	+	**2.00e+002 0.00e+000**
F25	6.13e+002 1.54e+000	+	**2.16e+002 8.20e+001**	6.48e+002 4.11e+000	+	**6.17e+002 1.98e+000**
w/t/l	12/9/4		-	22/3/0		-
	DE/best/1		**ring-DE/best/1**	**DE/current-to-best/1**		**ring-DE/current-to-best/1**
F1	3.83e+003 1.57e+003	+	**1.46e+002 2.69e+002**	3.20e+003 1.58e+003	+	**4.84e+000 9.78e+000**
F2	7.12e+003 3.38e+003	+	**9.66e+001 2.08e+002**	6.18e+003 1.64e+003	+	**2.56e-003 6.39e-003**
F3	1.66e+007 1.44e+007	+	**9.35e+005 4.65e+005**	4.94e+006 2.22e+006	+	**2.39e+004 1.83e+004**
F4	2.73e+002 5.14e+002	+	**7.41e+001 2.45e+002**	6.29e+002 4.18e+002	+	**3.10e-007 7.37e-007**
F5	8.67e+003 1.94e+003	+	**3.32e+003 5.72e+002**	7.69e+003 1.50e+003	+	**5.57e+002 3.77e+002**
F6	5.09e+008 3.97e+008	+	**1.25e+007 1.86e+007**	2.67e+008 2.86e+008	+	**4.28e+005 8.34e+005**
F7	3.27e+003 5.07e+002	+	**2.15e+001 3.24e+001**	3.92e+003 5.51e+002	+	**1.53e+001 3.01e+001**
F8	2.10e+001 5.01e-002	=	**2.10e+001 4.88e-002**	**2.09e+001 3.95e-002**	=	2.09e+001 4.11e-002
F9	1.13e+002 2.18e+001	+	**6.25e+001 1.57e+001**	7.10e+001 1.33e+001	+	**4.28e+001 1.42e+001**
F10	1.69e+002 3.98e+001	+	**1.07e+002 2.27e+001**	1.04e+002 3.11e+001	+	**5.87e+001 1.13e+001**
F11	**2.17e+001 2.93e+000**	-	2.90e+001 2.62e+000	**1.36e+001 2.22e+000**	-	1.94e+001 3.18e+000
F12	1.01e+005 3.75e+004	+	**1.03e+004 1.06e+004**	4.15e+004 2.66e+004	+	**1.31e+003 1.84e+003**
F13	8.44e+000 2.43e+000	+	**4.40e+000 1.19e+000**	5.48e+000 3.98e+000	=	**3.44e+000 8.37e-001**
F14	**1.18e+001 7.13e-001**	=	1.22e+001 5.39e-001	**1.18e+001 3.78e-001**	=	1.19e+001 3.65e-001
F15	5.17e+002 5.50e+001	+	**4.28e+002 5.42e+001**	4.44e+002 8.91e+001	=	**3.96e+002 9.47e+001**
F16	3.16e+002 1.64e+002	+	**1.74e+002 9.37e+001**	2.95e+002 1.82e+002	+	**8.28e+001 2.95e+001**
F17	3.41e+002 1.51e+002	+	**1.56e+002 5.04e+001**	2.62e+002 1.69e+002	+	**1.18e+002 1.11e+002**
F18	9.90e+002 2.65e+001	+	**9.22e+002 6.18e+001**	9.83e+002 3.00e+001	+	**8.99e+002 5.75e+001**
F19	9.77e+002 2.03e+001	+	**9.24e+002 6.30e+001**	9.91e+002 2.25e+001	+	**8.76e+002 6.89e+001**
F20	9.82e+002 2.38e+001	+	**9.22e+002 5.48e+001**	9.92e+002 2.21e+001	+	**9.06e+002 5.48e+001**
F21	1.11e+003 1.54e+002	+	**7.55e+002 3.23e+002**	1.07e+003 1.63e+002	+	**5.54e+002 1.81e+002**
F22	1.04e+003 6.67e+001	+	**9.73e+002 3.96e+001**	1.02e+003 3.42e+001	+	**9.28e+002 1.97e+001**
F23	1.15e+003 8.48e+001	+	**7.36e+002 2.93e+002**	1.08e+003 1.05e+002	+	**6.60e+002 1.42e+002**
F24	1.08e+003 2.59e+002	+	**2.00e+002 4.11e-003**	9.12e+002 2.00e+002	+	**2.00e+002 0.00e+000**
F25	1.51e+003 3.97e+001	+	**6.30e+002 4.80e+002**	1.45e+003 3.60e+001	+	**5.08e+002 1.96e+002**
w/t/l	22/2/1		-	20/4/1		-

"+", "–", and "=" indicate our approach is respectively better than, worse than, or similar to its competitor according to the Wilcoxon signed-rank test at α = 0.05.

TABLE II.
ERROR VALUES OF THE ORIGINAL DE MUTATION STRATEGIES AND THEIR CORRESPONDING RING-BASED VARIANTS OVER THE 50-DIMENSIONAL CEC 2005 BENCHMARK SET

	DE/rand/1		ring-DE/rand/1	DE/rand/2		ring-DE/rand/2
F1	**1.27e-028 1.31e-028**	-	6.00e-028 3.04e-028	4.22e+002 1.39e+002	+	**1.91e-016 3.72e-016**
F2	4.93e+000 2.97e+000	+	**1.15e-006 8.44e-007**	7.53e+004 9.40e+003	+	**4.17e+002 1.60e+002**
F3	2.82e+006 9.84e+005	+	**2.46e+005 1.04e+005**	4.41e+008 7.98e+007	+	**6.46e+006 1.54e+006**
F4	3.90e+002 2.06e+002	+	**1.49e+001 1.47e+001**	9.71e+004 1.07e+004	+	**6.84e+003 2.20e+003**
F5	**2.07e+003 4.09e+002**	-	3.30e+003 4.06e+002	2.13e+004 1.24e+003	+	**3.19e+003 4.33e+002**
F6	**3.55e+001 2.42e+001**	-	1.13e+002 9.26e+001	6.12e+006 2.90e+006	+	**4.28e+001 2.33e+001**
F7	**2.96e-004 1.48e-003**	-	9.43e-003 1.25e-002	1.07e+002 3.12e+001	+	**2.27e-003 5.45e-003**
F8	2.11e+001 3.49e-002	=	**2.11e+001 3.18e-002**	**2.11e+001 4.91e-002**	=	2.11e+001 6.70e-002
F9	2.08e+002 4.46e+001	+	**8.10e+001 1.64e+001**	4.70e+002 1.62e+001	+	**3.89e+002 1.48e+001**
F10	3.58e+002 1.21e+001	+	**7.85e+001 1.59e+001**	5.18e+002 2.44e+001	+	**3.86e+002 2.03e+001**
F11	7.36e+001 1.26e+000	+	**6.92e+001 7.65e+000**	7.31e+001 1.29e+000	=	**7.23e+001 1.85e+000**
F12	**1.20e+004 1.17e+004**	=	1.25e+004 1.07e+004	2.64e+006 1.96e+005	+	**1.30e+004 9.09e+003**
F13	3.03e+001 1.33e+000	+	**6.21e+000 1.52e+000**	4.30e+001 1.76e+000	+	**3.21e+001 1.36e+000**
F14	2.30e+001 1.67e-001	+	**2.29e+001 2.19e-001**	2.32e+001 1.65e-001	=	**2.32e+001 1.17e-001**
F15	**3.68e+002 7.48e+001**	=	3.77e+002 6.53e+001	4.63e+002 5.53e+001	+	**4.00e+002 3.42e-001**
F16	2.55e+002 8.82e+000	+	**6.18e+001 1.32e+001**	3.58e+002 1.37e+001	+	**2.78e+002 1.11e+001**
F17	2.80e+002 9.05e+000	+	**6.17e+001 1.37e+001**	4.16e+002 2.38e+001	+	**3.09e+002 1.03e+001**
F18	**9.14e+002 2.38e+001**	-	9.25e+002 4.84e+001	1.03e+003 7.63e+000	+	**9.18e+002 2.46e+001**
F19	**9.09e+002 3.30e+001**	-	9.38e+002 3.03e+001	1.03e+003 1.08e+001	+	**8.93e+002 5.33e+001**
F20	**8.70e+002 1.28e+002**	-	9.42e+002 1.58e+001	1.03e+003 1.18e+001	+	**8.98e+002 5.03e+001**
F21	5.00e+002 1.53e-005	=	**5.00e+002 1.08e-005**	6.38e+002 4.96e+001	+	**5.00e+002 1.08e-005**
F22	**9.58e+002 1.12e+001**	=	9.62e+002 8.12e+000	1.14e+003 1.34e+001	+	**9.82e+002 8.12e+000**
F23	**5.39e+002 1.11e-002**	-	5.53e+002 6.96e+001	7.13e+002 5.57e+001	+	**5.39e+002 1.13e-002**
F24	2.00e+002 3.11e-006	=	2.00e+002 3.11e-006	6.40e+002 1.13e+002	+	**2.00e+002 0.00e+000**
F25	6.10e+002 3.25e+000	+	**4.38e+002 1.98e+002**	7.49e+002 1.77e+001	+	**6.21e+002 4.25e+000**
w/t/l	11/6/8		-	22/3/0		-
	DE/best/1		**ring-DE/best/1**	**DE/current-to-best/1**		**ring-DE/current-to-best/1**
F1	2.13e+004 8.24e+003	+	**2.90e+002 2.72e+002**	2.23e+004 5.56e+003	+	**4.56e+001 7.33e+001**
F2	2.57e+004 6.99e+003	+	**1.44e+002 3.26e+002**	2.32e+004 5.97e+003	+	**2.54e-002 3.81e-002**
F3	1.44e+008 5.51e+007	+	**2.01e+006 6.59e+005**	9.89e+007 4.05e+007	+	**4.37e+006 1.51e+006**
F4	6.36e+003 3.96e+003	+	**3.68e+003 3.43e+003**	6.23e+003 1.96e+003	+	**8.88e+001 1.80e+002**
F5	1.80e+004 2.60e+003	+	**8.16e+003 1.40e+003**	1.49e+004 1.69e+003	+	**3.25e+003 6.65e+002**
F6	3.82e+009 1.88e+009	+	**2.00e+008 3.29e+008**	2.81e+009 1.05e+009	+	**6.08e+006 8.84e+006**
F7	6.43e+003 7.75e+002	+	**5.18e+001 6.72e+001**	7.32e+003 6.28e+002	+	**2.31e+001 3.88e+001**
F8	2.11e+001 3.01e-002	=	**2.11e+001 2.61e-002**	2.11e+001 3.13e-002	=	**2.11e+001 2.82e-002**
F9	2.77e+002 3.96e+001	+	**1.74e+002 3.71e+001**	1.93e+002 2.79e+001	+	**1.33e+002 2.55e+001**
F10	4.73e+002 8.14e+001	+	**2.81e+002 5.72e+001**	2.92e+002 4.30e+001	+	**1.91e+002 3.61e+001**
F11	**4.94e+001 4.74e+000**	-	5.85e+001 4.37e+000	**3.62e+001 3.77e+000**	-	4.70e+001 4.96e+000
F12	7.66e+005 2.29e+005	+	**3.65e+004 1.99e+004**	2.77e+005 1.48e+005	+	**1.13e+004 2.18e+004**
F13	2.87e+001 6.49e+000	+	**1.14e+001 2.38e+000**	8.56e+000 1.67e+000	+	**6.54e+000 1.15e+000**
F14	2.12e+001 6.49e-001	=	**2.09e+001 7.02e-001**	**2.13e+001 4.52e-001**	=	2.16e+001 4.82e-001
F15	6.23e+002 7.49e+001	+	**4.35e+002 2.82e+001**	5.46e+002 7.10e+001	+	**4.00e+002 2.77e+001**
F16	3.66e+002 1.03e+002	+	**1.81e+002 4.62e+001**	2.88e+002 1.35e+002	+	**1.44e+002 8.35e+001**
F17	3.18e+002 7.19e+001	+	**1.80e+002 3.85e+001**	2.27e+002 8.89e+001	+	**1.32e+002 7.13e+001**
F18	1.09e+003 3.06e+001	+	**1.05e+003 2.12e+001**	1.05e+003 2.71e+001	+	**1.00e+003 1.88e+001**
F19	1.09e+003 2.84e+001	+	**1.05e+003 2.19e+001**	1.05e+003 2.27e+001	+	**9.97e+002 2.03e+001**
F20	1.09e+003 2.61e+001	+	**1.05e+003 2.34e+001**	1.05e+003 2.61e+001	+	**1.01e+003 1.92e+001**
F21	1.24e+003 2.91e+001	+	**9.13e+002 3.48e+002**	1.21e+003 2.93e+001	+	**8.51e+002 3.48e+002**
F22	1.13e+003 6.93e+001	+	**1.04e+003 2.87e+001**	1.12e+003 2.82e+001	+	**1.01e+003 1.70e+001**
F23	1.25e+003 5.44e+001	+	**9.73e+002 2.80e+002**	1.23e+003 2.70e+001	+	**9.91e+002 1.79e+002**
F24	1.29e+003 2.91e+001	+	**1.26e+003 1.49e+001**	1.25e+003 2.39e+001	+	**1.04e+003 3.74e+002**
F25	1.66e+003 3.11e+001	+	**1.31e+003 3.92e+001**	1.62e+003 4.29e+001	+	**1.09e+003 2.62e+002**
w/t/l	22/2/1		-	22/2/1		-

"+", "–", and "=" indicate our approach is respectively better than, worse than, or similar to its competitor according to the Wilcoxon signed-rank test at $\alpha = 0.05$.

For the Wilcoxon's test whether at $\alpha = 0.05$ or $\alpha = 0.1$ in all four cases there are significant differences for all problems between ring-based DE and original DE. This indicates that ring-based DE is significantly better than its corresponding original DE based on the multiple-problem statistical analysis in these four cases.

With respect to the mutation strategies DE/rand/1/bin and DE/rand/2/bin, from Table I we observe that by using the neighborhood and direction information simultaneously the ring-DE framework efficiently exploits the population structure and guides the evolution towards more promising solutions. Furthermore, compared the result of the Wilcoxon's test of DE/rand/1/bin to the DE/rand/2/bin we conclude that the stronger exploration capability the mutation strategy has the better performance the according ring-DE algorithm would get.

In Table I, the results show that the ring-DE framework influences substantially the performance of DE/best/1/bin and DE/current-to-best/1/bin. The reason might be that the selection we designed for the best vector enhances the diversity of the exploitive mutation strategy (which uses the best individual to guide the search).

TABLE III.
RESULTS OF THE MULTIPLE-PROBLEM WILCOXON'S TEST FOR ORIGINAL DE VARIANTS FOR FUNCTIONS F01 - F25 AT D = 30.

Algorithm	**w/t/l**	**R+**	**R-**	**p-value**	**α = 0.05**	**α = 0.1**
ring-DE/rand/1 vs DE/rand/1	12/9/4	220.5	79.5	4.41E-02	+	+
ring-DE/rand/2 vs DE/rand/2	22/3/0	320	5	5.96E-07	+	+
ring-DE/best/1 vs DE/best/1	22/2/1	296	4	8.34E-07	+	+
ring-DE/current-to-best/1 vs DE/current-to-best/1	20/4/1	296	4	8.34E-07	+	+

TABLE IV.
RESULTS OF THE MULTIPLE-PROBLEM WILCOXON'S TEST FOR ORIGINAL DE VARIANTS FOR FUNCTIONS F01 - F25 AT D = 50.

Algorithm	**w/t/l**	**R+**	**R-**	**p-value**	**α = 0.05**	**α = 0.1**
ring-DE/rand/1 vs DE/rand/1	11/6/8	168.5	131.5	5.87E-01	=	=
ring-DE/rand/2 vs DE/rand/2	22/3/0	323.5	1.5	1.49E-07	+	+
ring-DE/best/1 vs DE/best/1	22/2/1	298	2	3.58E-07	+	+
ring-DE/current-to-best/1 vs DE/current-to-best/1	22/2/1	296	4	8.34E-07	+	+

From Table II and Table IV, similar to the results for all functions at $D = 30$, it is obvious that ring-DE approaches also consistently outperform their original DE methods in the majority of the test functions at $D = 50$. ring-DE/rand/1/bin, ring-DE/rand/2/bin, ring-DE/best/1/bin and ring-DE/current-to-best/1/bin significantly improve their original DE algorithms in 11, 22, 22 and 22 out of 25 functions, respectively. Moreover, with respect to the multiple-problem analysis DE based on ring-based mutation operators obtains significantly better results in 3 cases at $\alpha = 0.05$.

D. Comparison of the Convergence Speed

Finally, in Fig. 1 we present convergence graphs for four functions at 30D, namely, F1, F3, F10 and F14. The graphs illustrate median solution error value curves for all DE variants considered in this section obtained from 25 independent simulations. As previously mentioned the graphs indicate that in most cases the ring-DE framework enhances the convergence of a strategy. Details follow.

From (a) and (b) in Fig.1, it is clear that for each strategy the ring-DE has a higher convergence speed than its according original DE. For example, in (a) we can see that ring-DE/rand/1 obtained a potential solution at about 1200th generation, and DE/rand/1 need about 2800 generations to get a similar solution. This result supports that our ring-DE framework can largely accelerate the convergence of DE for unimodal functions. Similar to the results from (a) and (b), for (c) it is obvious that for all strategies we studied the convergence can be enhanced by the ring-DE framework for basic multimodal functions. In the end, an unexpected result obtained in (d) that the DEs with our ring-DE framework have the same convergence speed with their according original DEs for expanded multimodal function F14. In the future study, we will pay more attention to this problem.

VI. CONCLUSIONS

In this paper, a simple and effective framework, named ring-DE, has been presented. On the one hand, we use ring topology to construct neighborhood for each individual. In this way, the neighborhood information can be utilized for selecting the parent vectors. On the other hand, the direction information with the neighbors of the current individual is introduced into the mutation operator of DE. Thus, the difference vector which contains the neighborhood and direction information can be obtained. In this work we have applied it to four original DE mutation strategies. Through evaluating the effectiveness of ring-DE, the results show us that the ring-DE framework is able to improve the performance of different DE algorithms.

In the future, we will apply ring-DE to other DE variants to test the effectiveness, and the parameters of ring-DE (e.g., sub-population radius proportion) will also be studied.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China (61305085), the Support Program for Innovative Team and Leading Talents of Huaqiao University (2014KJTD13) and the Fundamental Research Funds for the Central Universities (12BS216).

REFERENCES

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

[2] S. Das and P. N. Suganthan, "Differential evolution: a survey of the state-of-the-art," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, Feb. 2011.

[3] M. G. Epitropakis, D. K. Tasoulis, N. G. Pavlidis, V. P. Plagianakos, and M. N. Vrahatis, "Enhancing differential evolution utilizing proximitybased mutation operators," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 99–119, Feb. 2011.

[4] B. Dorronsoro and P. Bouvry, "Improving classical and decentralized differential dvolution with new mutation operator and population topologies," *IEEE Transactions on Evolutionary Computation*, vol.15, no.1, pp.67– 98, Feb. 2011.

[5] M. Omran, A. Engelbrecht, and A. Salman, "Using the ring neighborhood topology with self-adaptive differential evolution," in *Advances in Natural Computation*, L. Jiao, L. Wang, X.-B. Gao, J. Liu, and F. Wu, Eds. Berlin, Germany: Springer-Verlag, 2006, pp. 976–979.

[6] Y. Cai, J. Wang, and J. Yin, “Learning-enhanced differential evolution for numerical optimization,” *Soft Computing*, vol. 16, no. 2, pp. 303–330, Feb. 2012.

[7] Y. X. Wang and Q. L. Xiang, “Exploring new learning strategies in differential evolution algorithm,” in *Proc. IEEE Congr. Evol. Comput.*, 2008, pp. 204–209.

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

[9] P. N. Suganthan, N. Hansen, J. J. Liang, K. Deb, Y.-P. Chen, A. Auger, and S. Tiwari, “Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization,” Nanyang Technol. Univ., Singapore, KanGAL Rep. No. 2005005, May 2005, IIT Kanpur, India.

[10] S. García, A. Fernández, J. Luengo and F. Herrera, “A study of statistical techniques and performance measures for genetics-based machine learning: Accuracy and interpretability,” *Soft Comput*, vol. 13, no. 10, pp. 959 – 977, 2009.

[11] J. Alcalá-Fdez, L. Sánchez and S. García, “KEEL: a software tool to assess evolutionary algorithms for data mining problems,”[Online]. Available: http://www.keel.es/

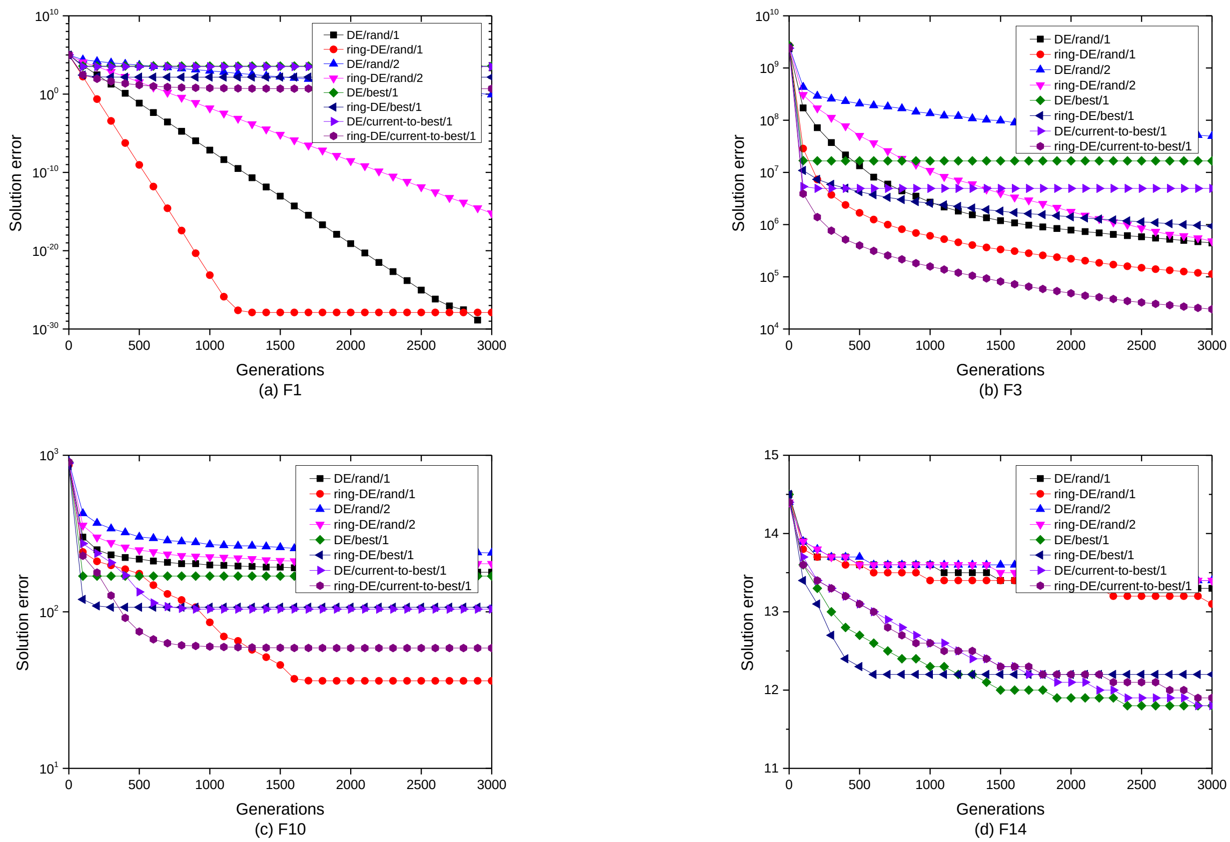

Figure 1. Convergence graphs of different DE variants for the selected functions at D = 30.