ICIC Express Letters
Volume **10**, Number **11**, November **2016**

ICIC International ©2016 ISSN 1881-803X
pp. 2547–2553

RANK-BASED DIFFERENTIAL EVOLUTION WITH EIGENVECTOR-BASED CROSSOVER OPERATOR

Jun-ichi Kushida, Akira Hara and Tetsuyuki Takahama

Graduate School of Information Sciences
Hiroshima City University
3-4-1 Ozukahigashi, Asaminami-ku, Hiroshima-shi 731-3194, Japan
{ kushida; ahara; takahama }@hiroshima-cu.ac.jp

Received April 2016; accepted July 2016

Abstract. *Differential evolution (DE) is an efficient algorithm for solving optimization problems in a continuous space. In recent years, many studies have reported modification and improvement for DE. Rank-based DE (RDE) is one of the modified DE algorithms, which allocates different control parameter values for each individual based on the ranking information in the current population. We attempt to improve the search ability of RDE by using an eigenvector-based (EIG) crossover. The EIG crossover is a rotationally invariant operator which provides superior performance on non-separable problems. In the EIG crossover, population is rotated to an appropriate coordinate system, and then a crossover operator is executed on the rotated population. In this paper, the performance of the RDE with EIG crossover is evaluated on the basic benchmark functions. Through the experiments, we show that the EIG crossover can enhance the search ability of RDE.*
Keywords: Differential evolution, Crossover, Rotationally invariant

1. **Introduction.** Differential evolution (DE) [1] is one of the most powerful global numerical optimization algorithms in the field of evolutionary algorithm. It has been successfully applied to various standard benchmark problems and has found several real-world applications. However, the performance of DE mainly depends on mutation strategies and crossover operators and their associated control parameters (i.e., population size NP, scaling factor F, and crossover rate CR). Due to this, much research has been conducted to analyze the effects of these control parameters and proposed various parameter adaptive DE variants [2, 3, 4]. Most of them use the binomial crossover operator. However, rank-based DE (RDE) [5], which is one of the adaptive parameter controlling methods, uses the exponential crossover. Rank-based DE (RDE) assigns different F and CR for each individual by taking into account the diversity-convergence balance using ranking information in the current population.

In DE algorithm, the mutation operation is rotation invariant but the binomial crossover and the exponential crossover with $CR \neq 1$ are not rotationally invariant. The performance of DE with both crossover operators is sensitive to rotation on the coordinate system, which represents highly correlated parameters in real-world optimization problems [6, 7]. Due to this, the rotationally invariant arithmetic recombination operator is proposed [8, 9]. However, the pure arithmetic recombination approaches may lose the diversity of population, and suffer from the problem of premature convergence.

To deal with this problem, the eigenvector-based (EIG) crossover operator is proposed as an alternative crossover [6]. The EIG crossover utilizes eigenvectors of covariance matrix of individual solutions, which makes the crossover rotationally invariant. To avoid losing diversity of population, the eigenvector-based ratio P is introduced as a new control parameter, which determines the ratio of the EIG crossover operator and the other crossover operator. In [6], the binomial crossover operator was replaced with the EIG crossover in original DE algorithm, and it could significantly improve the performance of DE. Additionally, the EIG crossover can be applied to any crossover strategy with minimal changes.

In this paper, we incorporate the EIG crossover to the RDE/rand/exp algorithm and therefore introduce a new algorithm, the RDE with eigenvector-based crossover (RDE-EIG). Moreover, by applying the concept of RDE, we propose a control scheme of parameters P based on the ranking information. This paper is organized as follows. A brief description of the DE algorithm is given in Section 2. Section 3 describes the eigenvector-based crossover operator. In Section 4, we present the proposed RDE with the eigenvector-based crossover operator. In Section 5 we present the results of experiment. Finally, the conclusion is given in Section 6.

2. **Differential Evolution.** DE is one of the variants of evolutionary algorithms that use a population. Similar to other evolutionary algorithms, DE searches for a global optimum in the search space with a population of vectors

$$\vec{x}_{i,G} = \{x_{1,i,G}, x_{2,i,G}, \ldots, x_{D,i,G}\}, \quad i = 1, 2, \ldots, NP \tag{1}$$

where G denotes the current generation, D is the dimension of the search space, and NP is the population size. In generation $G = 0$, the jth component of the ith vector can be initialized as

$$x_{j,i,0} = x_{j,\min} + \text{rand}_{i,j}[0, 1] \cdot (x_{j,\max} - x_{j,\min}) \tag{2}$$

where $\text{rand}_{i,j}[0, 1]$ is a uniform random number on the interval $[0, 1]$, and $x_{j,\min}, x_{j,\max}$ are the prescribed minimum and maximum bounds of the jth dimension, respectively. After initialization, DE employs the mutation and crossover operations to produce a trial vector for each target vector $\vec{x}_{i,G}$ in the current population. The main procedure of DE is briefly explained in the following subsections.

2.1. **Mutation.** In mutation, DE creates a mutant vector $\boldsymbol{v}_i = (v_{i1}, v_{i2}, \cdots, v_{iD})$ for each target vector $\boldsymbol{x}_i$ by certain mutation strategy. Some well-known mutation operations are listed as follows.

"rand/1":

$$\vec{v}_{i,G} = \vec{x}_{r1,G} + F(\vec{x}_{r2,G} - \vec{x}_{r3,G}) \tag{3}$$

"best/1":

$$\vec{v}_{i,G} = \vec{x}_{best,G} + F(\vec{x}_{r2,G} - \vec{x}_{r3,G}) \tag{4}$$

"current-to-best/1":

$$\vec{v}_{i,G} = \vec{x}_{i,G} + F(\vec{x}_{best,G} - \vec{x}_{i,G}) + F(\vec{x}_{r2,G} - \vec{x}_{r3,G}) \tag{5}$$

In the above equation, $\vec{x}_{best,G}$ is the best individual in the current population, and the indices $r1$, $r2$ and $r3$ are distinct integers uniformly chosen from the set $\{1, 2, \cdots, NP\} \setminus \{i\}$. The parameter F is called the scaling factor, which is a positive real number.

2.2. **Crossover.** After mutation, DE generates a trial vector $\vec{u}_{i,G} = \{u_{1,i,G}, u_{2,i,G}, \ldots, u_{D,i,G}\}$ by crossover operation to increase the potential diversity of the population. The commonly used crossover operators in DE are the binomial (bin) crossover and the exponential (exp) crossover [1]. In the crossover, CR is the crossover rate within the range $[0, 1)$ and presents the probability of generating genes for a trial vector from a mutant vector. Exponential crossover involves representational bias (dependence of ordering of parameters within a vector) [10]. In this paper, we modify exponential crossover to shuffled exponential crossover [11] by the following operation. First, the variable indices of the parents are randomly shuffled. Next, exponential crossover is applied to the shuffled parameter vectors. Finally, the indices are restored to their pre-shuffled. By this operation, the representational bias can be eliminated.

2.3. **Selection.** The selection operator is performed to select a better one from the target vector $\vec{x}_{i,G}$ and the trial vector $\vec{u}_{i,G}$ according to their fitness values $f(\cdot)$. For a minimization problem, the vector with the lower objective function value survives the next generation, which can be expressed as follows:

$$\vec{x}_{i,G+1} = \begin{cases} \vec{u}_{i,G} & \text{if } f(\vec{u}_{i,G}) < f(\vec{x}_{i,G}) \\ \vec{x}_{i,G} & \text{otherwise} \end{cases} \tag{6}$$

where $f(\cdot)$ is the objective function to be optimized.

The stopping criterion for the DE in general is usually the generation number or the number of objective function evaluations.

3. **Eigenvector-Based Crossover Operator.** The EIG crossover operator makes the binomial crossover operator become a rotationally invariant operator by rotating the coordinate system to a proper one [6]. In each generation, we compute the covariance matrix of the population, and decompose the matrix into a set of eigenvectors.

The covariance between the ith and jth dimensions of the population in the Gth generation is defined as

$$\text{cov}(i,j) = \frac{\sum_{k=1}^{NP}(x_{i,k,G} - \bar{x}_{i,k,G})(x_{i,j,G} - \bar{x}_{i,j,G})}{NP - 1} \tag{7}$$

where $\bar{x}_{i,G} = (1/NP)\sum_{k=1}^{NP} x_{i,k,G}$ denotes the mean value of the variables in the ith dimension. The covariance matrix $\mathbf{C}_G$ can be defined in terms of the covariance as

$$\mathbf{C}_G = (c_{i,j}, c_{i,j} = \text{cov}(i,j)) \tag{8}$$

To compute the eigenvector basis, we factorize the covariance matrix $\mathbf{Q}_G$ into a canonical form as

$$\mathbf{C}_G = \mathbf{Q}_G \Lambda_G (\mathbf{Q}_G)^{-1} \tag{9}$$

where $\mathbf{Q}_G$ is the square matrix $(D \times D)$ whose ith column is the eigenvector $\vec{q}_{i,G}$ of $\mathbf{C}_G$ and Λ_G is the diagonal matrix whose diagonal elements are the corresponding eigenvalues. The factorization of a matrix into a canonical form is called eigen decomposition.

In the eigenvector basis, the ith target vectors $\vec{x}_{i,G}$ can be expressed by $(\mathbf{Q}_G \cdot \vec{x}_{i,G})$; the ith mutant vectors can be expressed by $(\mathbf{Q}_G \cdot \vec{v}_{i,G})$. Then, some of the elements of the mutant vector $\vec{v}_{i,G}$ will be exchanged with some of the elements of its target vector to form a trial vector by a predefined crossover operator, such as binomial crossover or exponential crossover. The trial vector is given by

$$u_{i,j} = \begin{cases} \mathbf{Q}_G^* \cdot \text{xover}(\mathbf{Q}_G \cdot \vec{x}_{i,G}, \mathbf{Q}_G \cdot \vec{v}_{i,G}), & \text{if rand}_i[0,1] \leq P \\ \text{xover}(\vec{x}_{i,G}, \vec{v}_{i,G}), & \text{otherwise} \end{cases} \tag{10}$$

where $\mathbf{Q}_G^*$ is the conjugate transpose of the eigenvector basis Q_G, and $\text{xover}\left(\vec{a}, \vec{b}\right)$ is a crossover operator on two vectors $\vec{a}$ and $\vec{b}$, where P is an eigenvector ratio between 0% and 100% to determine the ratio of the eigenvector-based crossover operator and the other crossover operator.

4. **RDE with Eigenvector-Based Crossover Operator.** The rank-based DE (RDE) is one of the DE variants that adopts an observation-based control of algorithm parameters. In RDE, different parameter values are assigned based on goodness of base vector. When the base vector is good, a small scaling factor and a large crossover rate are selected and convergence is realized. Also, when the base vector is bad, a large scaling factor and a small crossover rate are selected and the divergence is realized.

At the beginning of each generation, the ranks R_i of the individual vectors $\vec{x}_{i,G}$ are given according to the fitness. First, the population is sorted in ascending order (i.e., from the best to the worst) based on the fitness of each individual. Then, the ranking of a vector is assigned as $R_i = i$ $(i = 1, 2, \cdots, NP)$, where the best vector in the current population will obtain the highest ranking ($R_i = 1$). Before mutation, different values of F and CR are assigned to each target vector according to the rank of the base vector. Let a target vector be denoted by $\vec{x}_{i,G}$, the base vector be denoted by $\vec{x}_{r1,G}$ and the rank of the base vector be denoted by R_{r1}. The scaling factor F_i and the crossover rate CR_i for $\vec{x}_{i,G}$ can be defined by the following equations:

$$F_i = F_{\min} + (F_{\max} - F_{\min})\frac{R_{r1} - 1}{NP - 1} \tag{11}$$

$$CR_i = CR_{\max} - (CR_{\max} - CR_{\min})\frac{R_{r1} - 1}{NP - 1} \tag{12}$$

where $F_{\min}$, $F_{\max}$ are parameters to specify the minimum and maximum values of F, and $CR_{\min}$, $CR_{\max}$ are parameters to specify the minimum and maximum values of CR. If the base vector is the best individual, F becomes the minimum value and CR becomes the maximum value. If the base vector is the worst individual, F becomes the maximum value and CR becomes the minimum value.

In this paper, we incorporate the EIG crossover to the RDE algorithm and therefore introduce a new algorithm, the RDE with eigenvector-based crossover (RDE-EIG). The strategy of RDE-EIG is RDE/rand/1/exp (eigenvector-based mixed with exponential crossover). After the parameter assignment, mutant vector is generated by Equation (3). Then, EIG crossover is performed by Equation (10) and trial vector is generated.

As described in Section 3, new algorithm parameter P is introduced in the EIG crossover. The performance of DEs with EIG crossover is dependent on the selection of P, which makes a great impact on the population diversity. To preserve the diversity of population and prevent premature convergence, the EIG crossover requires a suitable setting of P. For this reason, we attempt to automatically adjust P during the search in RDE-EIG. Here we propose a control scheme for parameter P based on the ranking information. Similar to F and CR, the eigenvector ratio P_i for $\vec{x}_{i,G}$ can be defined by the following equations:

$$P_i = P_{\max} - (P_{\max} - P_{\min})\frac{R_{r1} - 1}{NP - 1} \tag{13}$$

where $P_{\min}$, $P_{\max}$ are parameters to specify the minimum and maximum values of P. When the base vector is of a higher rank, the eigenvector basis is used with a high probability to guide the evolution process toward more successful solutions. In contrast, when the base vector is of a lower rank, the natural basis is used in order to increase the population diversity.

5. **Experiment.**

5.1. **Setup.**
In this section, we evaluate the performance of RDE-EIG on the benchmark functions. The mathematical formulas and properties of these functions are shown in Table 1, where dimension $D = 40$. All functions are chosen for the minimization problems and their optimal values are all 0.

Table 1. Benchmark functions

Name	Expression	Domain
F_1: Sphere	$f(\boldsymbol{x}) = \sum_{i=1}^{D} x_i^2$	$[-5.12, 5.12]^D$
F_2: Ellipsoid	$f(\boldsymbol{x}) = \sum_{i=1}^{D} \left(1000^{\frac{i-1}{D-1}} x_i^2\right)$	$[-5.12, 5.12]^D$
F_3: Ackley	$f(\boldsymbol{x}) = 20 - 20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right) + e - \exp\left(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i)\right)$	$[-32.768, 32.768]^D$
F_4: Griewank	$f(\boldsymbol{x}) = 1 + \sum_{i=1}^{D} \frac{x_i^2}{4000} - \prod_{i=1}^{D}\left(\cos\left(\frac{x_i}{\sqrt{i}}\right)\right)$	$[-512, 512]^D$
F_5: Rastrigin	$f(\boldsymbol{x}) = 10D + \sum_{i=1}^{D} x_i^2 - 10\cos(2\pi x_i)$	$[-5.12, 5.12]^D$
F_6: Schaffer	$f(\boldsymbol{x}) = \sum_{i=1}^{D-1} \left(x_i^2 + x_{i+1}^2\right)^{0.25} \times \left\{\sin^2\left(50\left(x_i^2 + x_{i+1}^2\right)^{0.1}\right) + 1\right\}$	$[-100, 100]^D$
F_7: Ridge	$f(\boldsymbol{x}) = \sum_{i=1}^{D}\left(\sum_{j=1}^{i} x_j\right)^2$	$[-64, 64]^D$
F_8: Rosenbrock (chain)	$f(\boldsymbol{x}) = \sum_{i=1}^{D-1}\left\{100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right\}$	$[-2.048, 2.048]^D$
F_9: Rosenbrock (star)	$f(\boldsymbol{x}) = \sum_{i=2}^{D}\left\{100\left(x_1 - x_i^2\right)^2 + (x_i - 1)^2\right\}$	$[-2.048, 2.048]^D$

To investigate the effect of the EIG crossover on RDE, we run RDE-EIG with fixed P and RDE-EIG with proposed control scheme for P for each function. The parameters settings for RDE-EIG are as follows – the population size $NP = 80$, $F_{\min} = 0.5$, $F_{\max} = 1.0$, $CR_{\min} = 0.1$, $CR_{\max} = 1.0$. Each algorithm was run 20 times and the maximum generation is $G_{\max} = 10^5$. If the DE algorithm can reach an error value, defined as $(f(x) - f(x^*))$ where x^* is the global optimum of f, less than $\varepsilon = 10^{-7}$ then we assume it has found the global optimum and stopped the DE.

5.2. **Results.** Figure 1 shows the result of RDE-EIG in each function as P is varied over the set $\{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$. Note that $P = 0$ means original RDE (RDE/rand/1/exp). Here, the plots are averaged number of function evaluations (NFE) required for finding the global optima. When the solution search failed even once, there are no plots in the figure.

In separable unimodal functions (F_1 and F_2) and multimodal functions with small peaks and valleys (F_3 and F_4), as the P increases, the performance of RED-EIG is gradually improved. In contrast, in multimodal functions with large peaks and valleys (F_5 and F_6), large P value significantly degrades the search ability of RDE-EIG. For non-separable unimodal functions (F_7, F_8, and F_9), we can see that $P = 0.4$ or $P = 0.5$ is a suitable value. These characteristics of P in eigenvector-based mixed with exponential crossover are similar to the experimental results of eigenvector-based mixed with binomial crossover mentioned in [6]. From these results, EIG crossover with adequate setting of P can enhance the search ability of RDE/rand/1/exp by making the crossover process rotationally invariant.

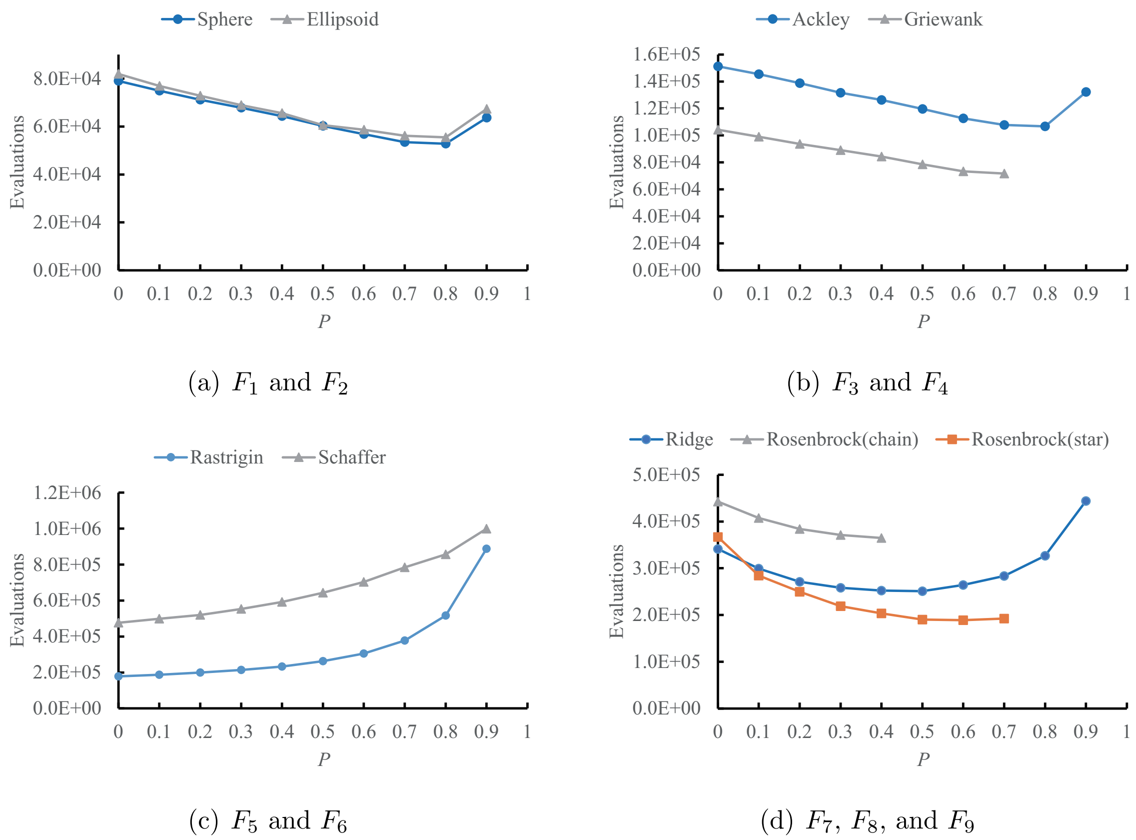

FIGURE 1. The average NFE of RDE-EIG with different Ps

TABLE 2. The average NFE of RDE-EIG with various combinations of P_{min} and P_{max}

	P_{min}, P_{max}			
Function	[0, 0]	[0, 0.5]	[0, 1.0]	[0.5, 1.0]
F_1: Sphere	7.90E+04 [20]	6.68E+04 [20]	5.44E+04 [20]	**5.00E+04** [20]
F_2: Ellipsoid	8.19E+04 [20]	6.87E+04 [20]	5.66E+04 [20]	**5.24E+04** [20]
F_3: Ackley	1.51E+05 [20]	1.30E+05 [20]	1.09E+05 [20]	**1.00E+05** [20]
F_4: Griewank	1.04E+05 [20]	8.79E+04 [20]	7.16E+04 [20]	**6.64E+04** [20]
F_5: Rastrigin	**1.78E+05** [20]	1.92E+05 [20]	2.22E+05 [20]	3.64E+05 [20]
F_6: Schaffer	**4.76E+05** [20]	5.42E+05 [20]	6.57E+05 [20]	– [7]
F_7: Ridge	3.41E+05 [20]	2.54E+05 [20]	**2.46E+05** [20]	2.98E+05 [20]
F_8: Rosenbrock (chain)	4.43E+05 [20]	3.59E+05 [20]	**3.56E+05** [20]	4.22E+05 [20]
F_9: Rosenbrock (star)	3.67E+05 [20]	2.17E+05 [20]	**1.83E+05** [20]	2.03E+05 [20]

Next, Table 2 shows the average NFE of RDE-EIG with control scheme for P. In this experiment, we investigate several combinations of $P_{\min}$ and $P_{\max}$. Note that $P_{\min} = 0$ and $P_{\max} = 0$ means original RDE. For each function, the average NFE required to find the global optima when an algorithm solves the problem with 100% success rate is shown in the top row. The number of success runs is shown in the bottom row. Except for F_5 and F_6, each combination of $P_{\min}$ and $P_{\max}$ demonstrates relatively better performance. In particular, for F_1-F_4, the combination of $P_{\min} = 0.5$ and $P_{\max} = 1.0$ is suitable. For F_7-F_9, the combination of $P_{\min} = 0.0$ and $P_{\max} = 1.0$ is suitable. However, the combination of $P_{\min} = 0.5$ and $P_{\max} = 1.0$ failed to find the global minimum in F_6. Consequently, it seems that the relatively large $P_{\min}$ may lead to the premature convergence for complex multimodal functions.

6. **Conclusion.** We have introduced a new DE algorithm, namely, the RDE-EIG, which extends the RDE with the eigenvector-based crossover operator. In the EIG crossover, new algorithm parameter, called eigenvector ratio P, is introduced and it affects significantly the performance of DEs. Therefore, we proposed a control scheme for P based on the concept of RDE. In the original paper of the EIG crossover [6], binomial crossover was employed and the effectiveness was shown through the numerical simulations. In the paper, we combined exponential crossover with an eigenvector-based crossover operator on RDE algorithm. From the experimental results using basic benchmark functions, we confirmed that the EIG crossover with an adequate setting of P can enhance the search ability of RDE/rand/1/exp, except multimodal functions with large peaks and valleys. Due to this, not only binomial crossover, the mixing of exponential crossover and eigenvector-based crossover is also effective. Finally, for our future work, we will try to improve the control scheme for P by taking into account landscape modality during the search.

REFERENCES

[1] R. Storn and K. V. Price, Differential evolution: A simple and efficient heuristic for global optimization over continuous spaces, *Journal of Global Optimization*, vol.11, no.4, pp.341-359, 1997.

[2] J. Zhang and A. C. Sanderson, JADE: Adaptive differential evolution with optional external archive, *IEEE Trans. Evolutionary Computation*, vol.13, no.5, pp.945-958, 2009.

[3] A. K. Qin, V. L. Huang and P. N. Suganthan, Differential evolution algorithm with strategy adaptation for global numerical optimization, *IEEE Trans. Evolutionary Computation*, vol.13, no.2, pp.398-417, 2009.

[4] J. Liu and J. Lampinen, A fuzzy adaptive differential evolution algorithm, *Soft Computing*, vol.9, no.6, pp.448-462, 2005.

[5] T. Takahama and S. Sakai, Efficient constrained optimization by the epsilon constrained rank-based differential evolution, *Proc. of 2012 IEEE Congress on Evolutionary Computation*, pp.62-69, 2012.

[6] S.-M. Guo and C.-C. Yang, Enhancing differential evolution utilizing eigenvector-based crossover operator, *IEEE Trans. Evolutionary Computation*, vol.19, no.1, pp.31-49, 2015.

[7] S. Das and P. N. Suganthan, Differential evolution: A survey of the state-of-the-art, *IEEE Trans. Evolutionary Computation*, vol.15, no.1, pp.4-31, 2011.

[8] K. V. Price, An introduction to differential evolution, *New Ideas in Optimization*, McGraw-Hill, London, pp.79-108, 1999.

[9] K. V. Price, Eliminating drift bias from the differential evolution algorithm, *Advances in Differential Evolution*, Springer Berlin Heidelberg, pp.33-88, 2008.

[10] K. V. Price, R. N. Storn and J. A. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*, Natural Computing Series, Springer, 2005.

[11] R. Tanabe and F. Alex, Reevaluating exponential crossover in differential evolution, *Parallel Problem Solving from Nature-PPSN XIII*, pp.201-210, 2014.