

# An Elite Gene Guided Reproduction Operator for Many-objective Optimization

Qingling Zhu, Qiuzhen Lin, Jianqiang Li, Carlos A. Coello Coello, *Fellow, IEEE*,  
Zhong Ming, Jianyong Chen, *Member, IEEE*, and Jun Zhang, *Fellow, IEEE*

**Abstract**—Traditional reproduction operators in many-objective evolutionary algorithms (MaOEAs) seem not so effective to tackle many-objective optimization problems (MaOPs). This is mainly because the population size cannot be set to an arbitrarily large value if the computational efficiency is concerned. In such case, the distance between parents becomes remarkably large and, consequently, it is not easy to reproduce a superior offspring in high dimensional objective space. To alleviate this problem, an elite gene guided (EGG) reproduction operator is proposed for tackling MaOPs in this paper. In this operator, an elite gene pool is built by collecting the knee points from the current population. Then, the offspring is produced by exchanging genes with this elite gene pool under an exchange rate, aiming to reserve more promising genes into the next generation. In order to provide new genes for the population, other genes will be disturbed under a disturbance rate. The settings and functional analysis of exchange rate and disturbance rate are studied using several experiments. The proposed EGG operator is easy for implementation and can be embedded to any MaOEA. As examples, we show the embedding of the proposed EGG operator into four competitive MaOEAs, i.e., MOEA/D, NSGA-III,  $\theta$ -DEA and SPEA2-SDE, provides some advantages over simulated binary crossover, differential evolution and an evolutionary path based reproduction operator on solving a number of benchmark problems with 3 to 15 objectives.

**Index Terms**—Evolutionary operator, Recombination operator, Crossover operator, Many-objective optimization, Evolutionary algorithm.

## I. INTRODUCTION

**M**ULTI-OBJECTIVE optimization problems (MOPs) contain several objectives to be optimized simultaneously. As the objectives often conflict with each other in MOPs, there exists a set of equally optimal solutions termed Pareto-optimal solutions (PS), rather than a single optimal solution.

Manuscript received XX. XX., 2018; revised XX. XX., 2018; accepted XX. XX. 2018. This work was supported by the National Natural Science Foundation of China under Grants 61876110, 61672358 and 61836005, the Joint Funds of the National Natural Science Foundation of China under Key Program Grant U1713212 and the Fundamental Research Project in the Science and Technology Plan of Shenzhen under Grant JCYJ20170817102218122, GGF2018021118145859 and JSGG20180507182904693. Prof. C.A. Coello Coello gratefully acknowledges support from CONACyT project 1920 (Fronteras de la Ciencia) and from a SEP-Cinvestav 2018 project (application no. 4). (Corresponding author: Qiuzhen Lin and Zhong Ming)

Q.L. Zhu, Q.Z. Lin, J.Q. Li, Z. Ming and J.Y. Chen are with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China (e-mails of Qiuzhen Lin and Zhong Ming: qiuzhenlin@szu.edu.cn and mingz@szu.edu.cn).

C.A. Coello Coello is with the Department of Computer Science, CINVESTAV-IPN (Evolutionary Computation Group), México, D.F. 07300, MÉXICO.

J. Zhang is currently a visiting scholar with Victoria University, Melbourne, VIC, Australia.

Due to the powerful population-based search nature in multi-objective evolutionary algorithms (MOEAs), they are effective and efficient to tackle MOPs with two or three objectives. However, when encountering MOPs with more than three objectives (often called many-objective optimization problems (MaOPs)), the performance of MOEAs quickly deteriorates due to the curse of dimensionality [1]. To better solve MaOPs, some research efforts have been conducted to develop new algorithms and technologies.

When tackling MaOPs, three main challenges are often faced. First, the effect of Pareto-based ranking becomes insignificant for MaOPs, as the ratio of non-dominated individuals in the population raises rapidly with the number of objectives. Second, under the above case, the diversity maintenance mechanism begins to play a key role for MaOPs. However, the existing diversity criteria, such as the crowding distance [2], tend to favor some dominance resistant solutions [3] (i.e., solutions with superior performance on at least one of the objectives but with especially poor performance on the rest objectives). Third, the search capabilities of evolutionary operators are considerably weakened for MaOPs, as the parents are usually far away from each other in a high dimensional search space. As observed from the empirical studies reported in [1], an offspring obtained from two nearly converged parents is contrarily far away from the true Pareto-optimal front (PF).

To overcome the first challenge, some relationships modified from the original Pareto dominance have been proposed, such as grid dominance [4], fuzzy Pareto dominance [5],  $\theta$ -dominance [6], reference point dominance [7] and strengthened dominance relation [8], to make them more effective to solve MaOPs. Other performance indicators, such as hypervolume [9],  $I_\epsilon$  [10], and S-metric [11], were reported to substitute Pareto optimality when tackling MaOPs. However, the computation of these indicators is often costly and aggravated with the increase of objectives. To solve the low efficiency problem, computationally efficient indicators such as  $R_2$  [12], IGD [13], IGD-NS [14], and  $I_{SDE+}$  [15] have been studied and used in indicator-based MaOEAs. Other kind of MOEAs [16] [17] based on decomposition seems more effective for MaOPs, by decomposing an MaOP into a set of sub-problems and then optimizing them on a cooperative manner. Some enhanced strategies have been reported for this kind of MOEAs to better solve MaOPs, such as DrEA [18], and PAEA [19]. In order to effectively solve problems with irregular Pareto-optimal fronts in decomposition-based MaOEAs, some adaptive adjustment methods for search directions were proposed [20] [21].

In order to address the second challenge, an extra selec-

tion criterion has been often introduced to maintain diversity when tackling MaOPs. For example, a set of reference points was introduced into NSGA-II [2] with an efficient niching methodology to ensure diversity in NSGA-III [1]. To balance convergence and diversity, a Pareto-based dominance and a decomposition method were combined in MOEA/DD [22], a reference points-based selection strategy was proposed in PREA [23], an angle-penalized distance was proposed in RVEA [24], an one-by-one selection strategy was designed in lby1EA [25], a balanceable fitness estimation method was designed in NMPSO [26], and a meta-objective approach was introduced in MeO [27]. Similar to the use of reference points, the incorporation of preferences into MOEAs was adopted to maintain diversity in [28] [29]. Moreover, an adaptive sorting-based selection strategy was developed in SAEA [30] in order to balance convergence and diversity when handling problems with irregular PF.

On the third challenge, there are very few studies available in the specialized literature. In most many-objective evolutionary algorithms (MaOEAs) [1]-[4], [22]-[31], simulated binary crossover (SBX) [32] has been often used with a large distribution index, leading to a high probability of sampling offspring close to their parents. The reason for this setting of SBX is not fully studied in these MaOEAs and there are few research studies that show the behavior of crossover when tackling MaOPs. To fill this research gap, a novel elite gene guided reproduction operator (EGG) is proposed in this paper. In this approach, an elite gene pool will be built by only including the knee point solutions selected from the current population, showing promising properties on convergence and diversity. Such that, the child solution will be produced by exchanging genes of parents selected from this pool under a high probability (i.e., the exchange rate ( $Er$ )). Moreover, some new genes will be produced by disturbing the allele genes of parents with a disturbance rate ( $Dr$ ). In summary, the main contributions of this paper are listed as follows.

- A generalized pattern of real-coded crossover is summarized. Most of the existing real-coded crossover operators can be realized using this generalized pattern. This approach uses  $Dr$  to manage the disturbance in order to generate new genes for offspring. Otherwise, it retains the original genes from the basic parent.
- An exchanged gene model is introduced and embedded into the above generalized pattern of real-coded crossover. Different from the existing real-coded crossover, this model will contain a recombination between the basic parent and the mating parent based on an  $Er$  probability. That is to say, the child solution will inherit from both the basic parent and mating parents.
- Based on the new pattern, an elite gene guided reproduction operator (EGG) is proposed. In EGG, the mating parents are only selected from the knee points, which can be a better representative for the current population. This mechanism will reserve more elite genes to the next generation. To implement EGG, the disturbance method and the setting of parameters are studied in this paper.

The rest of this paper is organized as follows. Section

II provides the related background of existing real-coded crossover used in most MOEAs and the motivations. The proposed EGG and its details are introduced in Section III. In Section IV, the experimental setup of EGG is clarified and the experimental results are provided in Section V on solving a large number of benchmark problems. Finally, our conclusions are drawn in Section VI along with some pertinent observations.

## II. BACKGROUND AND MOTIVATIONS

In this section, some commonly used crossover operators for real-coded MOEAs are introduced and analyzed. Then the related crossover operators for solving MaOPs and the motivations for this paper are given.

### A. Reproduction Operators in MOEAs

Crossover and mutation are two commonly used reproduction operators in MOEAs. The mutation operator is normally run after the crossover operator. Generally, mutation is applied on a single parent. Some examples are: polynomial-based mutation (PM) [33], Gaussian mutation [34], and Cauchy mutation [35]. In contrast, crossover involves the use of more than one parent. Some examples are: blend crossover (BLX) [36], SBX [32], differential evolution (DE) [37], simplex crossover (SPX) [38], and parent centric crossover (PCX) [39]. A detail survey on crossover operators can be found in [40][41]. When running crossover, the currently evolved parent is often treated as the basic parent, while the other selected parents are regarded as the mating parents in this paper. Two widely used real-coded crossover operators (SBX and DE) are introduced below.

1) *SBX*: Simulated binary crossover was designed by Deb and Agrawal [32], with the aim of mimicking the behavior of one-point crossover in binary representation. In a classical implementation of SBX in NSGA-II, the basic parent is denoted by  $x^{bsc} = (x_1^{bsc}, \dots, x_n^{bsc})$ , and the mating parent is selected as denoted by  $x^{mat} = (x_1^{mat}, \dots, x_n^{mat})$  (here,  $n$  is the number of variables). The disturbed genes (or variables)  $d_i^{SBX}$  ( $i = 1, 2, \dots, n$ ) are then generated based on the genes of  $x^{bsc}$  and  $x^{mat}$ , by

$$d_i^{SBX} = \begin{cases} 0.5[(1 + \beta)x_i^{bsc} + (1 - \beta)x_i^{mat}]; & \text{if } r < 0.5 \\ 0.5[(1 - \beta)x_i^{bsc} + (1 + \beta)x_i^{mat}]; & \text{otherwise} \end{cases} \quad (1)$$

where  $\beta$  follows a polynomial probability distribution and is calculated as follows.

$$\beta = \begin{cases} (2 \times r)^{1/(1+\eta)}; & \text{if } r \leq 0.5 \\ (\frac{1}{2 - 2 \times r})^{1/(1+\eta)}; & \text{otherwise} \end{cases} \quad (2)$$

where  $r$  is a uniformly generated real number in  $[0,1]$ , and the distribution index  $\eta$  is a predefined non-negative real number. A larger value of  $\eta$  will have a higher probability to sample the disturbed genes closer to those of its parents, e.g., in (1),  $d_i^{SBX}$  is close to  $x_i^{bsc}$  or  $x_i^{mat}$ . Then, after the disturbance gene is generated, an offspring is reproduced by inheriting the

gene  $x_i^{bsc}$  from the basic parent or selecting the disturbed gene  $d_i^{SBX}$  controlled by  $Dr$ , as follows.

$$y_i = \begin{cases} d_i^{SBX}; & \text{if } r \leq Dr \\ x_i^{bsc}; & \text{otherwise} \end{cases} \quad (3)$$

where  $y_i$  is the  $i$ -th gene of the offspring.

2) *DE*: Differential evolution is a direct search algorithm originally proposed by Storn and Price [37]. “DE/rand/1/bin” is one classical realization of DE. The disturbed gene  $d_i^{DE}$  is produced by the gene  $x_i^{bsc}$  from one basic parent and the genes  $x_i^{mat_1}, x_i^{mat_2}$  from two mating parents, by

$$d_i^{DE} = x_i^{bsc} + F \times (x_i^{mat_1} - x_i^{mat_2}) \quad (4)$$

where  $F$  is the scaling factor. Generally, (4) is regarded as a mutation method in DE [42]. In a more general sense, (4) is called a disturbance method in this paper in order to distinguish it from a mutation method (e.g., polynomial-based mutation [33]). Then, an offspring is reproduced by inheriting the gene  $x_i^{bsc}$  from the basic parent or selecting the disturbed gene  $d_i^{DE}$  controlled by  $Dr$ , as follows.

$$y_i = \begin{cases} d_i^{DE}; & \text{if } r \leq Dr \text{ or } i = i_{rand} \\ x_i^{bsc}; & \text{otherwise} \end{cases} \quad (5)$$

where  $i_{rand}$  is a randomly generated integer in  $[1, n]$  to guarantee that at least one gene is different from the parent.

### B. Related Works and Motivations

Generally, most of MaOEAs use traditional crossover operators, like SBX and DE for reproducing child solutions. When tackling real-coded MaOPs, to the best of our knowledge, there have been few research efforts that attempt to study the behavior and the performance of crossover in a high dimensional objective space. Here, two reproduction operators for solving binary-coded MaOPs are reviewed. Then, motivated by their ideas, the design principle for the proposed EGG method is introduced.

In [43], the genetic diversity of PS was analyzed and the effectiveness of the crossover operator was studied when solving many-objective knapsack problems (MaOKPs). The authors showed that the genes in PS become noticeably diverse when the number of objectives is increased [43]. A strategy controlling on the maximum number of crossed genes (CCG) was designed to solve MaOKPs, using two-point crossover and uniform crossover [43]. The experiments showed that this CCG strategy significantly enhanced the performance for MaOKPs. Another crossover for MaOKPs, i.e., a distance-based crossover, was reported by Ishibuchi [44], also showing a clear performance enhancement when the distances between parents and offspring are close.

Due to the consideration of computational cost, the population size cannot be set arbitrarily large as we increase the number of objectives, so that the average distance between each pair of parents is significantly enlarged. These far-away parents will consequently lower the effectiveness of crossover operators. In [45], an evolution path based reproduction operator (EP) was proposed for solving MaOPs. However, EP

is designed only based on DE and is unable to embed other crossover operators, such as SBX, BLX, and PCX. Moreover, EP depends on the reference vectors so that it is unavailable for MaOEAs without using reference vectors, such as SPEA2-SDE [46] and MaOEA/C [47]. To fill the research gap, an EGG operator is designed for real-coded MaOPs in this paper. The proposed EGG is easy for implementation and can be embedded into any MaOEA. The design principle for EGG is based on the local search and elite gene guided strategies, as follows.

- The elite gene guided strategy. Elite solutions are selected from the union of parent population and child population during the environmental selection based on the principle of survival of the fittest. In EGG, the mating parents are selected from knee points in the current population, which can provide elite genes for reproducing child solutions.
- The local search strategy. A small gene exchange probability in [43] and a small parent-offspring distance crossover in [44] showed a clear performance improvement for solving binary-coded MaOKPs. In this paper, the local search strategy is also use in EGG for solving real-coded MaOPs.

The properties and details of this EGG operator are introduced in the following section.

## III. THE PROPOSED METHOD

### A. Generalized Crossover Pattern

Considering the SBX and DE operators, two models in them are generally used to construct the offspring's genes. One is disturbance, which combines the genes from the basic parent and the mating parents to produce new genes for the offspring (e.g., (1) and (4)). The other is inheritance, which gets the original genes from the basic parent with a preset probability. For example, for the  $i$ -th gene of offspring, there is a  $Dr$  probability to select the disturbed gene  $d_i^{SBX}$  and otherwise to inherit the gene  $x_i^{bsc}$  from the basic parent  $x^{bsc}$  in SBX; Similarly, there is a  $Dr$  probability to select the disturbed gene  $d_i^{DE}$  and otherwise to inherit the gene  $x_i^{bsc}$  from the basic parent  $x^{bsc}$  in DE. Therefore, the currently used crossover operators can be generalized by a common pattern, as follows.

$$y_i = \begin{cases} \text{Disturb}(x_i^{bsc}, x_i^{mat}); & \text{if } r \leq Dr \\ x_i^{bsc}; & \text{otherwise} \end{cases} \quad (6)$$

where  $y_i$  is the  $i$ -th gene of the offspring,  $r$  is a uniformly generated real number in  $[0, 1]$ ,  $x_i^{bsc}$  indicates the  $i$ -th gene of the basic parent, and  $x_i^{mat}(i = 1, 2, \dots, n)$  denote the  $i$ -th gene from the mating parents.  $\text{Disturb}(x_i^{bsc}, x_i^{mat})$  means to generate a new gene by disturbance methods, such as (1) and (4);  $Dr$  is the probability to control the ratio of disturbed genes in the offspring. This pattern will be biased towards the basic parent when a smaller value of  $Dr$  is set.

### B. Proposed Crossover Pattern

Regarding the generalized crossover pattern introduced in (6), it is only biased to the basic parent. A smaller value of  $Dr$  indicates a stronger bias from the offspring to the basic parent.

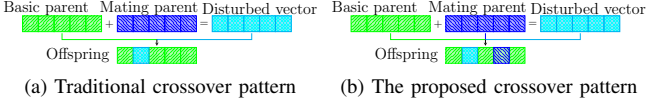


Fig. 1. A comparison between traditional crossover pattern and the proposed crossover pattern.

Particularly, when  $Dr$  is set to zero, the offspring is equal to the basic parent. Here, a more flexible crossover pattern is introduced, as follows.

$$y_i = \begin{cases} \text{Disturb}(x_i^{bsc}, x_i^{mat}); & \text{if } r \leq Dr \\ x_i^{mat}; & \text{else if } r \leq Dr + Er \\ x_i^{bsc}; & \text{otherwise} \end{cases} \quad (7)$$

where  $y_i$  is the  $i$ -th gene of the offspring and  $r$  is a uniformly generated real numbers in  $[0, 1]$ . The disturbance rate  $Dr$  controls the ratio of genes inherited from the disturbed vector, while the exchange rate  $Er$  controls the proportion of genes inherited from the mating parent  $x_i^{mat}$ .

A comparison between the generalized crossover pattern in (6) and the proposed crossover pattern in (7) is illustrated in Fig. 1, from which we can observe that the former only generates child solution with a combination of genes from basic parent and disturbed vector, while the latter can produce child solution with a combination of genes from basic parent, disturbed vector and mating parent. More elite genes can be reserved to the next generation when the mating parents are selected from an elite subset of current population. The proposed EGG operator follows the new proposed crossover pattern in (7), which is introduced in the following subsection.

### C. Implementation of EGG

In order to implement EGG, there are three important issues needed to be addressed. One is the mating parent selection mechanism, which can provide the elite genes for child solution with a ratio of  $Er$ . The other is disturbance method, which is the main source to provide new genes and may significantly affect the crossover's performance. The last one is the settings of  $Dr$  and  $Er$  in (7), which controls the balance between the gene diversity and the recombination of the currently found genes. Here, the detailed implementation of these three parts is introduced in the following subsections in order to better solve MaOPs using EGG.

1) *Mating selection mechanism*: In order to provide good genes for recombination with the basic parent, mating parents are only selected from the knee points [48], as these solutions show some promising properties on convergence and diversity. The knee points have been used in KnEA [48] for mating selection, where the basic parent and mating parents are selected from the knee points using a binary tournament selection strategy with three comparison criteria, i.e., Pareto dominance, knee point criterion, and a weighted distance. However, in EGG, the mating parents are randomly selected from the knee points, while the basic parent is the current solution to be evolved. Moreover, the crossover in KnEA follows the pattern in (6), while EGG realizes the pattern in (7), which has a

combination of genes from basic parent, disturbed vector and mating parent.

A promising adaptive strategy in [48] has been proposed to identify the knee points. With this strategy, a hyperplane  $L$  is first constructed by passing through the extreme points of each objective, and then the solution is deemed to a knee point when it has the maximum distance to  $L$  within its neighborhood. The main steps of finding these knee points are presented in **Algorithm 1**. First, the extreme hyperplane  $L$  is calculated (steps 2-3). Second, the neighborhood region is identified by  $R$ , which is adaptively calculated by steps 4-7. In step 6, the initial niching ratio  $r$  and the rate of knee points  $t$  are set to  $r = 1$  and  $t = 0$ , respectively. Third, the distance between each solution in  $P$  and  $L$  is calculated and then all the solutions are sorted in a descending order (steps 8-9). After that, the extreme solutions are added to the knee point set  $K$  and then removed from  $P$  (steps 10-11). These steps of finding the knee points are slightly different from that of KnEA, as we further consider the extreme points as the knee points. At last, the solution with the largest distance to  $L$  in  $P$  will be added to  $K$  and the neighboring solutions including the solution itself will be removed from  $P$  (steps 13-16). These steps will be run until  $P$  is empty. The ratio of knee points in population is controlled by a parameter  $T$  (step 6), which is set to 0.5 in this paper. Therefore, half of solutions will be selected from the current population as the mating parents pool.

To clearly show this strategy, an illustration of knee points is shown in Fig. 2 on a bi-objective minimization problem. The hyperplane  $L$  connects two extreme solutions  $A$  and  $I$ . Then, the solutions  $C$ ,  $F$ ,  $H$  far away from  $L$  and the two extreme solutions ( $A$  and  $I$ ) are marked as the knee points, with the rectangles in dashed lines to show their neighborhoods.

2) *Disturbance Method*: For the new crossover pattern in (7), a disturbance rate  $Dr$  is used to control the disturbance between the basic parent and the mating parents. This is the main source to provide new genes for the offspring. In our approach, the disturbance method of SBX in (1) with  $\eta = 30$  was used to generate new genes close to the parent's genes, while the disturbance method of DE in (4) with  $F = 0.5$  was adopted to yield a larger step size than SBX.

In [49], we proposed a hybrid gene-level disturbance

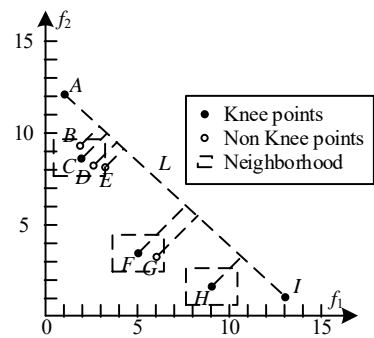


Fig. 2. An illustration of knee points for a bi-objective minimization problem. In this example, solutions  $C$ ,  $F$ ,  $H$  and the two extreme points ( $A$  and  $I$ ) are identified as knee points for the given neighborhood denoted by the rectangles in dashed lines.

---

**Algorithm 1:**  $K = \text{Finding\_knee\_point}(\hat{P})$ 


---

**Input:**  $\hat{P}$  - the non-dominated solutions

**Output:**  $K$  - the knee point set

```

1  $K = \emptyset$ 
2  $E = \text{Find\_extreme\_solution}(\hat{P})$ 
3  $L = \text{Calculate\_extreme\_hyperplane}(E)$ 
4  $f_j^{max} = \max\{f_j(x) | x \in \hat{P}\}, \forall 1 \leq j \leq m$ 
5  $f_j^{min} = \min\{f_j(x) | x \in \hat{P}\}, \forall 1 \leq j \leq m$ 
6 update  $r = r \times e^{-\frac{1-t/T}{m}}$  //initial values  $r=1, t=0$ 
7 calculate  $R$  by  $R_j = (f_j^{max} - f_j^{min}) \times r, \forall 1 \leq j \leq m$ 
8 calculate the distance between each solution in  $\hat{P}$  and  $L$ 
9 sort  $\hat{P}$  in a descending order according to the distances
10 add the extreme solutions  $E$  to knee point set  $K$ 
11  $\hat{P} = \hat{P} \setminus E$ 
12 while  $\hat{P}$  is not empty do
13   set  $p$  to the solution with largest distance to  $L$  in  $\hat{P}$ 
14    $K = K \cup p$ 
15    $NB = \{a | a \in \hat{P} \wedge |f_a^j - f_p^j| \leq R^j, \forall 1 \leq j \leq m\}$ 
16    $\hat{P} = \hat{P} \setminus NB$ 
17 update  $t = \frac{|K|}{N}$ 
18 return  $K$ 

```

---

method, which combines the search power of SBX and DE. Following the idea of the hybrid gene-level disturbance method in [49], the disturbed genes of DE serve as a complement for the disturbed genes of SBX, as follows.

$$d_i^{EGG} = \begin{cases} d_i^{DE}; & \text{if } r \leq Lr \\ d_i^{SBX}; & \text{otherwise} \end{cases} \quad (8)$$

where  $d_i^{EGG}$  are the disturbed genes for EGG, while  $d_i^{SBX}$  and  $d_i^{DE}$  are respectively the disturbed genes from SBX in (1) and from DE in (4). For example, in a 2-dimensional decision space, two parents  $x^1 = (0.3, 0.3)$  and  $x^2 = (0.7, 0.7)$  were used to produce 1000 sampling solutions by (1) with  $\eta = 30$  and the distribution of  $d_i^{SBX}$  in (1) is illustrated in Fig. 3(a). It is clear that the disturbance method in (1) only searches around the four rectangle points  $(0.7, 0.3)$ ,  $(0.7, 0.7)$ ,  $(0.3, 0.7)$ , and  $(0.3, 0.3)$ . After embedding the disturbance method of (4) with a 0.2 probability (i.e.,  $Lr$  is set to 0.2 in (8)), the distribution of 1000 points from  $x^1 = (0.3, 0.3)$  and  $x^2 = (0.7, 0.7)$  is plotted in Fig. 3(b) in 2-D case, while a 3-D case is plotted in Fig. 3(c). Observed from Figs. 3(b) and (c), the disturbance method in (8) can search around a rectangle in 2-D case and a cube in 3-D case, and consequently, in  $n$ -D decision space it will search around a hypercube, whose center is determined by the center of parents. Thus, if a large search step (e.g., the disturbance of (4)) can be added to some genes of the parents, the search capability of EGG can be strengthened, which was experimentally validated in Section V-C2.

The detail of the disturbance method in EGG is provided in **Algorithm 2**, which generates a new disturbed gene  $d_i^{EGG}$  by three alleles  $x_i^{bsc}, x_i^{mat1}, x_i^{mat2}$ . It uses a small probability  $Lr$  (set to 0.2 in this paper) to generate  $d_i^{EGG}$  using (4) with

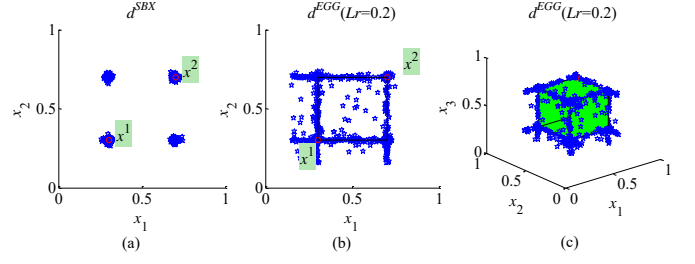


Fig. 3. The disturbance methods of SBX and the hypercube search. (a) The distribution of  $d_i^{SBX}$  in (3); (b) The distribution of  $d_i^{EGG}$  in (8) in 2-D; (c) The distribution of  $d_i^{EGG}$  in (8) in 3-D.

---

**Algorithm 2:**  $d_i^{EGG} = \text{Disturb}(x_i^{bsc}, x_i^{mat1}, x_i^{mat2})$ 


---

**Input:**  $x_i^{bsc}, x_i^{mat1}, x_i^{mat2}$  are three alleles in the basic parent and mating parents

**Output:** the disturbed gene  $d_i^{EGG}$ 

```

1 if  $\text{rand} \leq Lr$  then
2   calculate  $d_i^{DE}$  by (4)
3    $d_i^{EGG} = d_i^{DE}$ 
4 else
5   calculate  $\beta$  by (2)
6   calculate  $d_i^{SBX}$  by (1)
7    $d_i^{EGG} = d_i^{SBX}$ 
8 return  $d_i^{EGG}$ 

```

---

large disturbed step (steps 2-3), otherwise to produce  $d_i^{EGG}$  close to the gene of the parents by (1) (steps 5-7).

3) *The Complete EGG:* The disturbance method and the selection of mating parents in EGG have been introduced in the above subsections. Here, EGG is embedded into the general framework of MOEAs. The pseudo-code of MOEA-EGG is shown in **Algorithm 3**. After the initialization process (steps 1-2), the algorithm turns into the loop of the evolutionary process (steps 3-3), until the function evaluation counter  $fes$  reaches the predefined maximum value  $max\_fes$ . During the evolutionary phase, the selection of mating parents is first run to randomly select two mating parents  $x^{mat1}, x^{mat2}$  from the knee point set (steps 5-7). For each gene, EGG uses a  $Dr$  probability to disturb the current gene (steps 9-10). Otherwise, it exchanges genes with the mating parent by an  $Er$  probability or inherits the same genes from the basic parent by a  $(1 - Dr - Er)$  probability (steps 11-14). After the EGG operator is applied, the mutation operator (e.g., polynomial mutation [33]) is further run to disturb the offspring gene with a small probability (usually set to  $1/n$ ,  $n$  is the number of variables), in order to gain more population diversity (steps 15-16). At last, after the population  $Q$  is generated, the population update process is executed in step 20. The above evolutionary phase will be repeated until  $max\_fes$  is reached. At the end of the algorithm, the non-dominated solutions in  $P$  are reported as the final result.

4) *Settings of the Parameters  $Dr$  and  $Er$ :* It is obvious that the settings of  $Dr$  and  $Er$  significantly affect the performance of EGG. When EGG keeps a large probability of  $Dr$  to disturb parents, the genes in the offspring will be more diverse.

---

**Algorithm 3:** General framework of MOEA-EGG
 

---

**Input:**  $max\_fes$  - maximum number of function evaluations  
 $N$  - population size  
**Output:**  $P$  - the final solutions

```

1 initialize a population  $P = \{x^1, x^2, \dots, x^N\}$ 
2  $fes = 0$ 
3 while  $fes < max\_fes$  do
4   for  $j = 1$  to  $N$  do
5     set  $\hat{P}$  as the first non-dominated front in  $P$ 
6      $K = Finding\_knee\_point(\hat{P})$  (Algorithm 1)
7     randomly select  $x^{mat1}, x^{mat2}$  from  $K$ 
8     for  $i = 1$  to  $n$  do
9       if  $rand \leq Dr$  then
10         $y_i = Disturb(x_i^j, x_i^{mat1}, x_i^{mat2})$ 
11        (Algorithm 2)
12      else if  $rand \leq Dr + Er$  then
13         $y_i = x_i^{mat1}$ 
14      else
15         $y_i = x_i^j$ 
16      if  $rand \leq \frac{1}{n}$  then
17        mutate  $y_i$  by polynomial mutation
18      repair  $y_i$  by box constraints
19     $Q = Q \cup y$ 
20     $fes = fes + 1$ 
21   $P = Population\_Update(P, Q)$ 
22 return  $P$ 

```

---

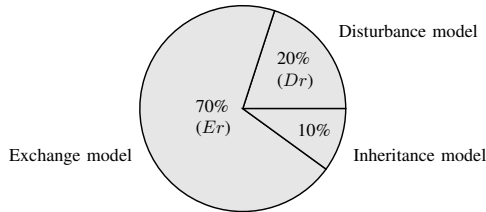


Fig. 4. The configuration for  $Dr$  and  $Er$ .

Intuitively, when  $Dr$  is set to 1.0, all genes in the offspring are new. On the other hand, a large value of  $Er$  lets the offspring be more biased toward the mating parents, while a small  $Er$  makes the offspring more similar to the basic parent. Their impacts on the behavior of EGG (i.e., to favor diversity and convergence) are analyzed experimentally in Section V-B and some suggestions are also given to properly set  $Dr$  and  $Er$ . In this paper, a small  $Dr=0.2$  and a large  $Er=0.7$  are recommended for solving MaOPs, as shown in Fig. 4.

#### D. Features of EGG

After introducing EGG, its features are summarized as follows.

- EGG has three models to generate the offspring genes, i.e., inheritance, disturbance and exchange. The exchanged gene model can yield a recombination of the

genes in basic parents and mating parents. This recombination ratio is controlled by  $Er$  and it can gain diversity for the offspring.

- The mating parents are only selected from the knee points of the first non-dominated front in the population. This lets EGG run the disturbance and the exchanged gene models with some promising individuals, as these knee points show good convergence and diversity, which can be viewed as a better representation for the population.
- A small ratio of running disturbance is used in EGG, as inspired from the studies on recombination operators for MaOEAs [1] [43] [44]. In EGG, a small value of  $Dr$  is applied for each gene in (7), thus the offspring is much closer to the basic parent or mating parent in high dimensional search spaces. Moreover, the effectiveness on the setting for  $Dr$  was also studied in Section V-B.
- On the disturbance model, most genes are disturbed using a small-size step, as a distribution index of 30 for the disturbance model of SBX in (1) is used in EGG, while only a few genes are perturbed with a large-size step using (4). This setting leads to a new search pattern as shown in Figs. 3(b) and (c) for 2-D and 3-D respectively. As validated in Section V-C2, a small ratio of large-size step is helpful to enhance the performance of EGG.

#### IV. EXPERIMENTAL SETUP

This section is devoted to the experimental design for investigating the performance of EGG. In the following sections, the test problems used in our experiments are firstly provided. Then, the quality indicators used for evaluating the performance of an algorithm will be introduced. At last, the experimental settings in this paper are indicated.

##### A. Test Problems

To analyze the effectiveness of EGG, two well-known test suites (i.e., DTLZ [50] and WFG [51]) for MaOPs were used. To reliably compute the quality indicators, DTLZ1-DTLZ4 were adopted, as the nature of PFs of DTLZ5 and DTLZ6 is unclear beyond three objectives [51]. All these problems can be scaled to any number of objectives and decision variables. In this study, the number of objectives is set to  $m \in \{3, 5, 8, 10, 15\}$ . For DTLZ1-DTLZ4, the total number of decision variables is given by  $n = m + k - 1$ , where  $k$  is set to 5 for DTLZ1, and to 10 for DTLZ2-DTLZ4 as recommended in [50]. For all the WFG test problems, the number of decision variables is set to  $2 \times (m - 1) + 20$  composed by two types of parameters (i.e.,  $2 \times (m - 1)$  position related parameters and 20 distance related parameters as suggested in [51]). These test problems have a variety of characteristics, such as linear, mixed (convex&concave), multi-modal, disconnected, degenerate, and differently scaled PFs, which can challenge different abilities of MaOEAs.

##### B. Quality Indicators

In this paper, two indicators, i.e., Hyper-Volume (HV) and Coverage of two sets (C) metric [52], were used to assess the performance of the compared algorithms in solving the DTLZ and WFG test problems.

TABLE I  
THE SETTINGS OF POPULATION SIZE AND MAXIMUM NUMBER OF  
FUNCTION EVALUATIONS FOR DIFFERENT PROBLEMS

$m$	$(H_1, H_2)$	$N$	$max\_fes$		
			DTLZ1	DTLZ2 DTLZ4	DTLZ3 WFG1-WFG9
3	(12,0)	91	18200	9100	45500
5	(6,0)	210	42000	16800	105000
8	(3,2)	156	39000	23400	117000
10	(3,2)	275	55000	27500	206250
15	(2,1)	135	67500	33750	270000

1) *HV metric*: HV assesses the size of the area dominated by the approximated PF, thus a higher value means a better quality of non-dominated set. The reference points for HV are set to 1.1 times the nadir points of the true PFs. Thus, the reference points are set to  $1.1 \times (0.5, \dots, 0.5)$  for DTLZ1, to  $1.1 \times (1, \dots, 1)$  for DTLZ2-DTLZ4, and to  $1.1 \times (2, 4, \dots, 2m)$  for WFG1-WFG9. Regarding the problems with 3 and 5 objectives, a fast calculation method was employed to get the exact HV, while a Monte Carlo sampling approach was applied to approximate HV for the problems with 8, 10, and 15 objectives [9]. It is noted that the solutions dominated by the reference point were not included for the HV calculation.

2) *C metric*: C metric can identify the Pareto-dominance relationship between two compared sets. Moreover, C metric is convenient to evaluate the performance of MaOEAs, because its calculation doesn't need to know the true PF. Assuming that  $A$  and  $B$  are two approximated non-dominated sets. The result of the coverage of two sets  $C(A, B)$  returns the proportion of the individuals in  $B$  that are equal to or dominated by the individuals in  $A$ , which can be formulated as follows.

$$C(A, B) = \frac{|\{b \in B; \exists a \in A : a \succeq b\}|}{|B|} \quad (9)$$

where the symbol " $a \succeq b$ " means  $b$  is equal to or dominated by  $a$ . The value  $C(A, B) = 1$  means that all the individuals in  $B$  are equal to or dominated by the individuals in  $A$ , while  $C(A, B) = 0$  implies no individual in  $B$  is equal to or dominated by the individual in  $A$ . It is noted that both  $C(A, B)$  and  $C(B, A)$  are required to be considered as  $C(A, B)$  is not necessarily equal to  $1 - C(B, A)$ . Therefore,  $A$  is said to be better than  $B$  when we get a large value of  $C(A, B)$  and a smaller value of  $C(B, A)$ .

### C. Experimental Settings

In this paper, to verify the performance of EGG, it was embedded into four competitive MaOEAs, i.e., three representative MaOEAs with reference point (MOEA/D [16], NSGA-III [1],  $\theta$ -DEA [6]) and one representative MaOEA without using reference vectors (SPEA2-SDE [46]). The general settings are introduced as follows.

1) *Setting of the Population Size*: As recommended in [1], a bi-layer generation method was adopted here to generate reference vectors on both the outer layer and the inside layer. The number of reference points  $N$  is determined by the simplex-lattice design factors  $H_1$  and  $H_2$ , respectively for the

TABLE II  
SUMMARIZED HV COMPARISON RESULTS OF SBX, DE, EP AND EGG ON  
FOUR BASELINE MAOEAS

Problems		DTLZ (w/b/s)	WFG (w/b/s)	Total (w/b/s)
Algorithms				
MOEA/D	EGG vs. SBX	0/20/0	2/41/2	2/61/2
	EGG vs. DE	4/9/7	2/34/9	6/43/16
	EGG vs. EP	0/20/0	5/35/5	5/55/5
NSGA-III	EGG vs. SBX	0/18/2	3/34/8	3/52/10
	EGG vs. DE	0/20/0	0/41/4	0/61/4
	EGG vs. EP	0/20/0	2/43/0	2/63/0
SPEA2-SDE	EGG vs. SBX	0/20/0	15/19/11	15/39/11
	EGG vs. DE	8/11/1	1/41/3	9/52/4
$\theta$ -DEA	EGG vs. SBX	0/19/1	2/35/8	2/54/9
	EGG vs. DE	2/12/6	0/38/7	2/50/13

Please note that " $w/b/s$ " indicates that EGG is worse than, better than and similar to the corresponding competitor, respectively on  $w$ ,  $b$ , and  $s$  problems in terms of HV indicator.

inside and outside layers, by  $N = \binom{H_1+m-1}{m-1} + \binom{H_2+m-1}{m-1}$  where  $m$  is the number of objectives. The population size is set the same with the number of reference points. In Table I, the detailed settings of  $H_1$ ,  $H_2$ , and the population sizes are given for all the test problems.

2) *Number of runs and termination criterion*: Each algorithm was run 30 times independently for each test instance. The algorithm will be terminated when the function evaluation counter  $fes$  reaches  $max\_fes$ , which is given in Table I. Please note that different settings of  $max\_fes$  were used for the test problems with different objectives, as they have varied computational complexities.

3) *Significance test*: To test the difference of statistical significance among the experimental results, the Wilcoxon rank sum test [53] at a 5% significance level was carried out on the experimental results obtained by two compared algorithms.

## V. EXPERIMENTAL RESULTS AND DISCUSSIONS

### A. Comparisons with Other Reproduction Operators in MOEA/D, NSGA-III, SPEA2-SDE, and $\theta$ -DEA

The most commonly used reproduction operator for solving MaOPs is SBX, e.g., NSGA-III, SPEA2-SDE and  $\theta$ -DEA all adopt SBX. Recently, an evolutionary path based reproduction operator was proposed for reference vector based MaOEAs. Therefore, in order to study the advantages of EGG, it was firstly incorporated into MOEA/D and NSGA-III in order to compare with SBX, DE and EP<sup>1</sup>. Please note that the probabilities for applying SBX and polynomial-based mutation were respectively set to 1.0 and  $1/n$  (where  $n$  denotes the number of decision variables), and the distribution indexes of SBX and polynomial-based mutation were respectively set to 30 and 20. For the DE operator, the scaling factor  $F$  was set to 0.5 in this paper. The parameter  $Dr$  in (3) and (5) respectively for SBX and DE was analyzed in Section S-B1 of the supplementary file and the best parameter  $Dr = 0.2$  was selected for comparison. The parameters for EP were set as recommended in the original reference [45] as the same baseline algorithm and test problems were used for comparison<sup>2</sup>.

<sup>1</sup>The code of this paper is available at <https://github.com/zhuqingling/EGG>.

<sup>2</sup>The code of EP is downloaded from <https://github.com/hxyokokok/EPDE>.



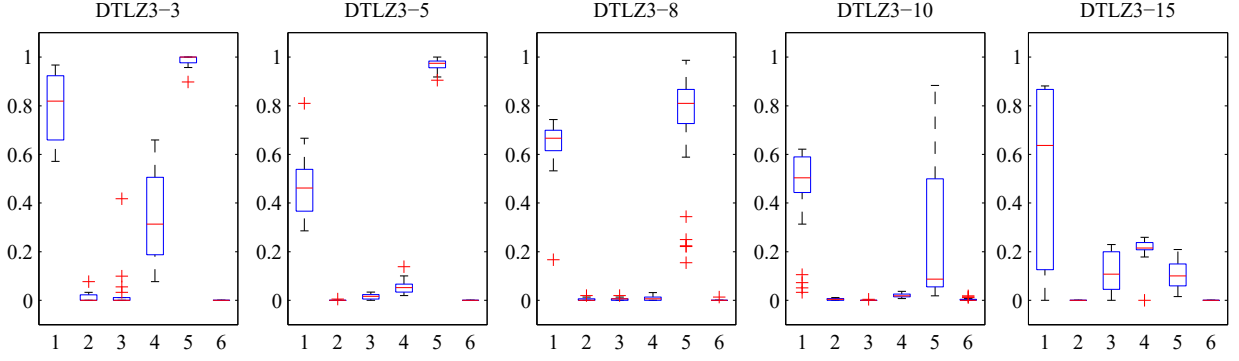


Fig. 5. Box plots of C metric in the framework of MOEA/D in solving DTLZ3 with 3, 5, 8, 10, and 15 objectives. In this plot, “1”, “2”, “3”, “4”, “5”, and “6” in the  $x$ -axis respectively denote  $C(\text{EGG}, \text{SBX})$ ,  $C(\text{SBX}, \text{EGG})$ ,  $C(\text{EGG}, \text{DE})$ ,  $C(\text{DE}, \text{EGG})$ ,  $C(\text{EGG}, \text{EP})$ , and  $C(\text{EP}, \text{EGG})$ . They have the same meanings in Fig. 6 and Figs. S-1 to S-24 of the supplementary file.

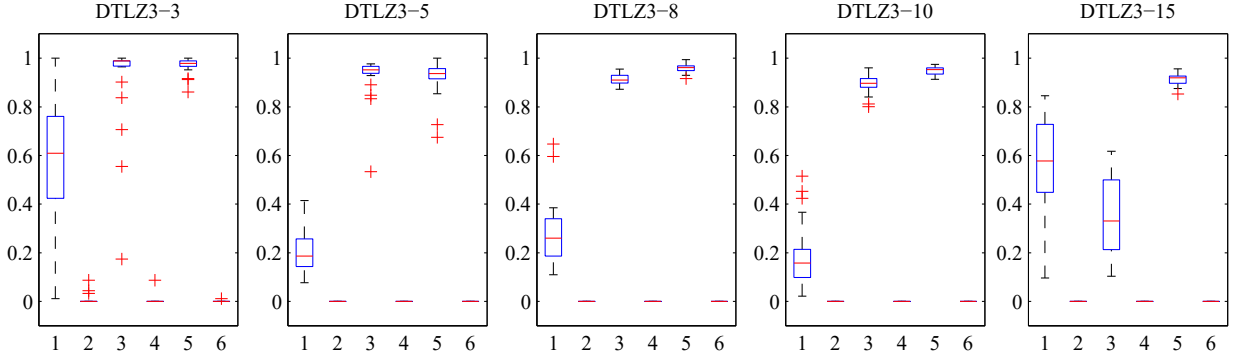


Fig. 6. Box plots of C metric in the framework of NSGA-III in solving DTLZ3 with 3, 5, 8, 10, and 15 objectives.

Due to pages limitations, the detailed HV results (median value and IQR value of 30 runs) of the comparisons between SBX, DE, EP and EGG in the framework of MOEA/D are presented in Table S-3 of the supplementary file. The summarized comparison results of EGG, EP, DE, SBX in the framework of MOEA/D are provided in the “MOEA/D” row of Table II. As observed from the “MOEA/D” row of Table II, EGG performs better than SBX, DE and EP in most cases. More specifically, from the Wilcoxon rank sum test, EGG performs significantly better than SBX, DE and EP in 61, 43, and 55 out of 65 test instances, respectively. In addition, C metric is also calculated and its box plots for all test instances are illustrated in Figs. S-1 to S-12 of the supplementary file. In most cases, the results of C metric are consistent with the HV results shown in Table S-3 of the supplementary file. For example, the box plot results of DTLZ3 with 3, 5, 8, 10, and 15 objectives are plotted in Fig. 5. In Fig. 5, “1”, “2”, “3”, “4”, “5”, and “6” in the  $x$ -axis respectively denote  $C(\text{EGG}, \text{SBX})$ ,  $C(\text{SBX}, \text{EGG})$ ,  $C(\text{EGG}, \text{DE})$ ,  $C(\text{DE}, \text{EGG})$ ,  $C(\text{EGG}, \text{EP})$ , and  $C(\text{EP}, \text{EGG})$ . The larger value of  $C(A, B)$  means there is a large percentage of solutions obtained by algorithm B that are equal to or dominated by solutions obtained by algorithm A. As observed from the subplots in Fig. 5, EGG is better than SBX and EP in 3, 5, 8, 10, and 15-objective DTLZ3 because  $C(\text{EGG}, \text{SBX})$  and  $C(\text{EGG}, \text{EP})$  are respectively large than  $C(\text{SBX}, \text{EGG})$  and  $C(\text{EP}, \text{EGG})$ . However, when compared to

DE on DTLZ3 with 3, 5, 8, 10, and 15 objectives, EGG does not have a superior performance. That is because the neighboring information is not utilized in MOEA/D-EGG since EGG selects the mating parents randomly from the knee points. The neighboring information is helpful for MOEA/D-DE to solve MaOPs due to the fact that two close parents are beneficial for the differential part in DE.

Except the comparison of EGG with SBX, DE, EP in the MOEA/D framework, we also compare them in the framework of NSGA-III. As observed from the summarized results in Table II, EGG shows an overwhelming better performance than SBX, DE and EP. Compare to the cases in the MOEA/D framework, EGG seems more suitable for the NSGA-III framework. There are two reasons for this case. One is that the neighboring information is not considered in the framework of NSGA-III. So that the advantages of utilizing neighboring information in MOEA/D-DE will not be employed in the framework of NSGA-III. On the other hand, the environmental selection mechanism in NSGA-III is better than that in MOEA/D for solving MaOPs, so that it can keep the promising individuals. As a result, EGG can fully utilize the better genes in the archive by exchanging genes with mating parents, which are selected from the knee point set. The box plots of C metric for the experimental results of SBX, DE, EP and EGG in the framework of NSGA-III are plotted in Fig. 6 when solving DTLZ3 with 3, 5, 8, 10, and 15 objectives. As observed from



Fig. 6, more than half of the solutions obtained by SBX, DE, and EP are dominated by that generated from EGG. Whereas, very few solutions obtained by EGG are dominated by that from SBX, DE, and EP, as the mean values of C metric are very close to zero. From this aspect, the convergence of NSGA-III-EGG is much better than that of NSGA-III-SBX, NSGA-III-DE, NSGA-III-EP on DTLZ3 with 3, 5, 8, 10, and 15 objectives. Due to pages limitations, the box plots of C metric for the other test problems are illustrated in Fig. S-13 to Fig. S-24 of the supplementary file.

Moreover, two other frameworks of SPEA2-SDE and  $\theta$ -DEA were also adopted to compare EGG with SBX and DE. The detailed comparison results are collected in Table S-5 of the supplementary file. It is noted that, in [45], EP was only incorporated in the frameworks of MOEA/D and NSGA-III. Thus we do not incorporate it into SPEA2-SDE due to the fact that EP needs to work with reference vectors. From the summarized results in Table II, it can be observed that EGG performs better than SBX and DE on 39 and 52 out of 65 test instances, respectively, in the framework of SPEA2-SDE. The similar observation can also be found in the framework of  $\theta$ -DEA, as EGG gets significantly better performance than SBX and DE on 54 and 50 out of 65 test instances, respectively.

Based on the above analysis, we have the following observations. The performance of EGG is affected by the environmental selection. Recently, there are many methods to improve the performance of environmental selection, trying to reserve promising solutions to the next generation. Here, EGG can better utilize the elite individuals after environmental selection due to the use of exchanged gene model and mating selection mechanism from knee points. Thus, EGG is more suitable for MaOEAs with better environmental selection mechanisms, such as NSGA-III and  $\theta$ -DEA.

### B. An Analysis of $Er$ and $Dr$

In order to analyze the parameter settings of  $Er$  and  $Dr$ , a reference vector based framework  $\theta$ -DEA and a non-reference-vector framework SPEA2-SDE are adopted here to compare EGG with its variants.

The settings of the disturbance rate  $Dr$  and the exchange rate  $Er$  significantly affect the performance of EGG. In this section, five combinations of  $Dr$  and  $Er$  introduced below were included for comparison, with  $Lr$  for the disturbance method set to 0.2 in (8).

- EGG1:  $Dr = 1$  and  $Er = 0$  were selected. All the genes with EGG1 will undergo disturbance as shown in Fig. 7(a). This case provides abundant diversity for the population.
- EGG2:  $Dr = 0.5$  and  $Er = 0$  were selected. EGG2 has a 0.5 probability for disturbance and otherwise inherits the same genes from the basic parent as shown in Fig. 7(b).
- EGG3:  $Dr = 0.5$  and  $Er = 0.25$  were selected. EGG3 has a 0.5 probability for disturbance and otherwise has a 0.5 probability to exchange genes with the mating parent and a 0.5 probability to inherit the same genes from the basic parent, as shown in Fig. 7(c).

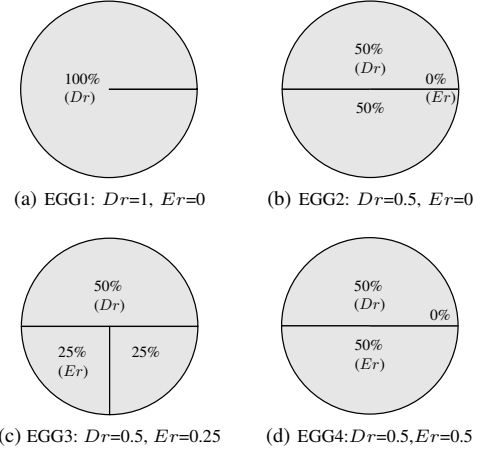


Fig. 7. Different proportions of the three models (disturbance, exchange, and inheritance) using different settings of  $Dr$  and  $Er$  for EGG.

TABLE III  
SUMMARIZED HV COMPARISON RESULTS OF EGG1, EGG2, EGG3, EGG4, AND EGG UNDER THE FRAMEWORKS OF  $\theta$ -DEA AND SPEA2-SDE

Algorithms	Problems	DTLZ (w/b/s)	WFG (w/b/s)	Total (w/b/s)
$\theta$ -DEA	EGG vs. EGG1	0/18/2	2/41/2	2/59/4
	EGG vs. EGG2	2/14/4	1/39/5	3/53/9
	EGG vs. EGG3	2/13/5	1/35/9	3/48/14
	EGG vs. EGG4	1/14/5	2/34/9	3/48/14
SPEA2-SDE	EGG vs. EGG1	3/13/4	1/41/3	4/54/7
	EGG vs. EGG2	4/11/5	0/40/5	4/51/10
	EGG vs. EGG3	1/10/9	5/29/11	6/39/20
	EGG vs. EGG4	2/10/8	0/37/8	4/47/16

- EGG4:  $Dr = 0.5$  and  $Er = 0.5$  were selected as illustrated in Fig. 7(d). When comparing this setting to EGG2 (with  $Dr = 0.5$  and  $Er = 0$ ), EGG4 is more biased to the mating parent, while EGG2 is more biased to the basic parent.
- EGG:  $Dr = 0.2$  and  $Er = 0.7$  were selected as shown in Fig. 4. This setting has a lower ratio for disturbance model and a larger ratio for exchanged gene model when compared to EGG3.

The comparison results of those five settings on solving the DTLZ and WFG test problems are illustrated in Table S-6 of the supplementary file, by embedding them into the framework of  $\theta$ -DEA. Moreover, the summarized comparison results are shown in Table III.

For the DTLZ problems, EGG performed significant better than EGG1, EGG2, EGG3, and EGG4 on 18, 14, 13, and 14 out of 20 test problems according to the Wilcoxon rank sum test. For DTLZ1-DTLZ4 with 3 objectives, EGG was better than EGG2 and EGG4 on DTLZ2 and DTLZ4, but worse on DTLZ1 and DTLZ3. When compared to EGG3, EGG was not advantageous on 3-objective DTLZ1-DTLZ4. However, regarding DTLZ1-DTLZ4 with 5, 8, 10, and 15 objectives, EGG showed overwhelming advantages as it obtained better or similar performance on all cases. EGG was similar to EGG2, EGG3, and EGG4 on DTLZ1 with 5, 8, 10, and 15 objectives according to the Wilcoxon rank sum test. That is because EGG and its variants EGG2, EGG3, EGG4 can well approximate the

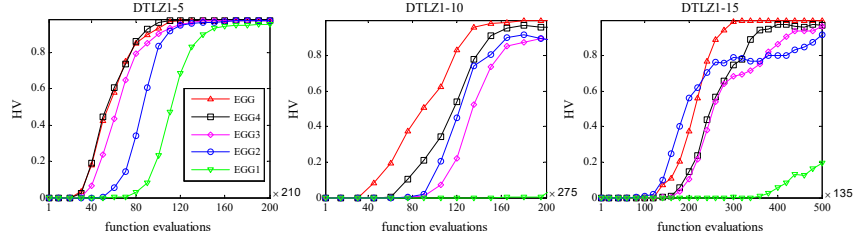


Fig. 8. The HV trend charts (averaged over 30 runs) of EGG, EGG1, EGG2, EGG3, and EGG4 on DTLZ1 with 5, 10 and 15 objectives.

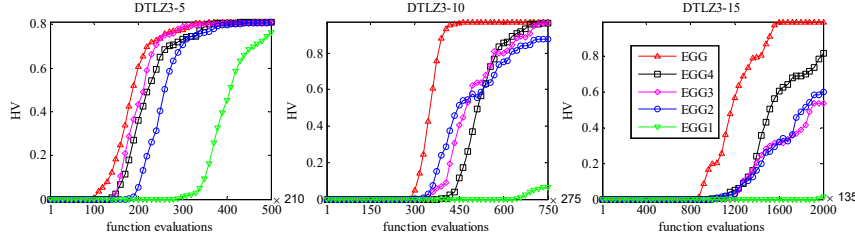


Fig. 9. The HV trend charts (averaged over 30 runs) of EGG, EGG1, EGG2, EGG3, and EGG4 on DTLZ3 with 5, 10, and 15 objectives.

true PF as their HV values are very close to one. The HV trend chart of DTLZ1 with 5, 10, 15 objectives is shown in Fig. 8. As observed from Fig. 8, EGG obtained better HV values for DTLZ1 with 10 and 15 objectives. In addition, the HV trend chart of DTLZ3 with 5, 10, 15 objectives is shown in Fig. 9. DTLZ3 is more difficult than DTLZ1 due to the fact that it has  $(3^k - 1)$  local PFs and all local PFs are parallel to the global PF. As observed from Fig. 9, the convergence rate of EGG is much faster than that of the other four variants, especially for DTLZ3 with 10 and 15 objectives. On the other hand, the superior performance of EGG over EGG1, EGG2, EGG3, and EGG4 can also be found under the framework of SPEA2-SDE. Due to pages limitations, their detail experimental results are provided in Table S-7 of the supplementary file, while the summarized results are shown in Table III. Specifically, EGG performed better than and similarly to EGG1, EGG2, EGG3, and EGG4 on 17, 16, 19, and 18 out of the 20 DTLZ test instances.

For the WFG problems, EGG had a significantly better performance than EGG1, EGG2, EGG3 and EGG4 on 41, 39, 35, and 34 out of 45 test instances. When compared to EGG, EGG1 only performed better on WFG2 with 15 objectives, on WFG3 with 15 objectives, and performed similarly on WFG3 with 8 and 10 objectives. EGG was significantly better than EGG1 on 41 out of 45 test problems. This is mainly because all genes were disturbed in EGG1, leading to a more random evolution for the parents and to a lack of convergence. Such weakness of EGG1 can also be found in the HV trend charts of DTLZ2-DTLZ4 and WFG1-WFG9 in the supplementary file, respectively plotted in Figs. S-25 to S-36. For EGG2, it was better than EGG only on WFG3 with 15 objectives. According to the Wilcoxon rank sum test, EGG2 performed similarly to EGG on WFG3 with 3, 8, and 10 objectives, on WFG2 with 10 objectives, and on WFG9 with 5 objectives. Thus, EGG performed better than EGG2 on 39 out of 45 test problems. This is mainly because the exchanged gene model

was not embedded into EGG2, and this also validated that a recombination of the currently found genes in this model helps to enhance performance. Regarding the comparison of EGG with EGG3, EGG performed worse only on WFG3 with 15 objectives. In summary, EGG performed better than or similarly to EGG3 on 44 out of 45 test problems. The configuration of EGG3 has a larger disturbance rate and much smaller exchange rate than EGG. When compared to EGG4, EGG performed worse on WFG2 with 15 objectives, and on WFG3 with 15 objectives. That is to say, EGG also performed better than or similarly to EGG4 on 43 out of 45 test problems. When compared to EGG, the setting of EGG4 has a larger disturbance rate and is more biased to the mating parent. As EGG3 and EGG4 also contain the exchanged gene model, the superior performance of EGG over EGG3 and EGG4 confirmed the setting of EGG with  $Dr = 0.2$ ,  $Er = 0.7$  was more reasonable and effective.

In summary, EGG performed best on WFG1 and WFG4-WFG9 with 3, 5, 8, 10, and 15 objectives. For WFG2 and WFG3, their HV results were mostly similar to each other, as  $\theta$ -DEA may not be suitable for solving disconnected and degenerated problems using reference points [54]. Thus, EGG was also embedded into the framework of SPEA2-SDE without using the reference points, and the detailed comparison results of the five settings on the DTLZ and WFG problems were provided in Table S-7 of the supplementary file. The summarized comparison results are also collected in Table III. From Table III, it was observed that EGG with  $Dr = 0.2$  and  $Er = 0.7$  performed better than and similarly to EGG1, EGG2, EGG3, and EGG4 on 61, 61, 59, and 63 out of all the 65 test problems, respectively. From all the above analysis, the setting of EGG with  $Dr = 0.2$  and  $Er = 0.7$  performed better than its other four settings. This is mainly due to the fact that EGG disturbs a small proportion of genes to run a local search and also exchanges elite gene provided by the environmental selection and knee-point-based mating

TABLE IV  
SUMMARIZED HV COMPARISON RESULTS OF EGG-NOEXCHANGE,  
EGG-NOKP, EGG ( $Lr=0$ ), EGG ( $Lr=0.5$ ) AND EGG UNDER THE  
FRAMEWORK OF  $\theta$ -DEA

Algorithms	Problems	DTLZ (w/b/s)	WFG (w/b/s)	Total (w/b/s)
EGG vs. EGG-noExchange		1/15/4	3/30/12	4/45/16
EGG vs. EGG-noKP		3/10/7	4/28/13	7/38/20
EGG vs. EGG ( $Lr=0$ )		0/8/12	9/18/18	9/26/30
EGG vs. EGG ( $Lr=0.5$ )		2/2/16	3/21/21	5/23/37

selection. Consequently, EGG overcomes the difficulties as analyzed in Section II. This setting of EGG can fully realize the merits of EGG, as described in Section III-D.

### C. An Analysis of Different Parts in EGG

When compared to the traditional crossover pattern in (6), the proposed new pattern in (7) gains an exchanged gene model, which exchanges gene with the mating parents by an exchange rate  $Er$ . In order to implement EGG in (7) for solving MaOPs, the knee points are randomly selected as mating parents and the disturbance method in (8) is used. In this section, all these three parts (the exchanged gene model, the disturbance method in (8), and the knee points as the mating parents) were experimentally analyzed. Thus, EGG was compared to the three EGG variants, i.e., EGG-noExchange (this variant sets  $Er = 0$  and discards the exchanged gene model), EGG-noKP (this variant only randomly selects the mating parents from the population, but not from the knee points), EGG( $Lr = 0$ ) (this variant sets  $Lr = 0$ ), and EGG( $Lr = 0.5$ ) (this variant sets  $Lr = 0.5$ ). Their median and IQR results on 65 test instances (DTLZ1-DTLZ4, WFG1-WFG9 with 3, 5, 8, 10, 15 objectives) were collected in Table S-8 of the supplementary file. The summaries of statistical test results are shown in Table IV. Based on these experimental results, the analysis for the exchanged gene model, the disturbance method in (8), and the knee points as the mating parents are respectively given below.

1) *An Analysis of the Exchanged Gene Model:* For the analysis of  $Er$  and  $Dr$ , it shows that the EGG variant with a small ratio for disturbance and a large ratio for exchange, i.e.,  $Dr = 0.2$ ,  $Er = 0.7$ , was shown to perform best in solving most of the test problems adopted. Here, the impact of the exchanged gene model is further studied by comparing EGG to EGG-noExchange.

By observing the summarized comparison results of EGG-noExchange and EGG at the “EGG vs. EGG-noExchange” row of Table IV, it was found that the performance of EGG was significantly deteriorated when the exchanged gene model was removed, as EGG-noExchange performed worst on most cases. More specifically, EGG performed better than or similarly to EGG-noExchange on 61 out of 65 (93%) test instances. Although the mating parents were also selected from the knee point set in EGG-noExchange, the elite genes in the mating parents have less chances to be reserved for the next generation if the exchanged gene model is not applied to EGG.

2) *An Analysis of the Disturbance Method:* In the disturbance method of (9) with  $Lr=0.2$ , a probability of 0.8 was

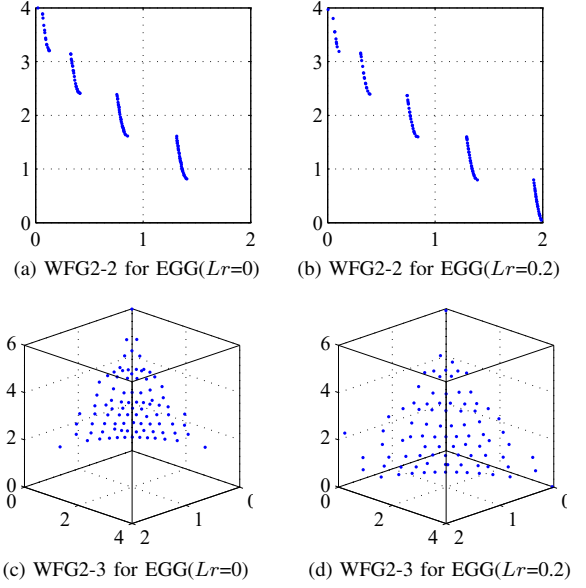


Fig. 10. The final approximate PFs for EGG with different  $Lr$  on WFG2 problem (One run with median HV value is plotted).

used for a small-size step, i.e.,  $d_i^{SBX}$ , while a probability of 0.2 was employed for a large-size step, i.e.,  $d_i^{DE}$ . This search model was shown in Figs. 3(b)-(c). In order to keep the same search model in Figs. 3(b) and 3(c), the ratio of the large-size step  $d_i^{DE}$  should be set small (i.e., around [0.1, 0.3]).

Regarding the WFG2 test instance as an example, which is a disconnected problem and whose last objective function is multi-modal, EGG( $Lr=0.2$ ) showed a significantly better performance than that without a larger step size (i.e., EGG( $Lr=0$ )). In this case, some extreme areas of the true PF of WFG2 may be missed, when a large-size step was removed from disturbance, as shown in Fig. 10, where the final sets from EGG( $Lr=0$ ) and EGG( $Lr=0.2$ ) were plotted on WFG2 with 2 and 3 objectives. Therefore, a small ratio for running a large-size step in EGG was very helpful to enhance its performance.

Here, two extreme cases were further considered for the disturbance method, such as no large-size step ( $Lr=0$ ) and a large ratio ( $Lr=0.5$ ) of large-size step. From the summarized comparison results of EGG( $Lr=0$ ) and EGG( $Lr=0.2$ ) in Table IV, EGG ( $Lr=0.2$ ) performed better than or similarly to EGG( $Lr=0$ ) on 56 out of 64 (87.5%) test instances. Especially for WFG2, as observed from the detailed comparison results in Table S-8 of the supplementary file, EGG performed better than EGG( $Lr=0$ ) on all the WFG2 instances having 3, 5, 8, 10, 15 objectives. On the other hand, when compared to EGG( $Lr=0.5$ ) in Table IV, EGG also performed better or similarly on 60 out of 65 (92%) test instances. This also indicated that the ratio of large-size step should not be set too large, as it will only run a coarse-grained search around the parents.

From the above discussion, it is reasonable to conclude that the disturbance method plotted in Figs. 3(b) and 3(c) was effective for EGG when solving MaOPs. However, a proper setting of  $Lr$  should be suggested, as a disturbance

without large-size step or with too many large-size steps will significantly deteriorate the performance of EGG.

3) *An Analysis of Mating Selection:* In our proposed EGG, the mating parents are only picked out from the knee points of the current population. Please note that the mating parents in EGG not only participate in the exchanged gene model in (7), but also join the disturbance method in (8). Here, EGG was compared to its variant EGG-noKP and the experimental results of EGG-noKP are provided in Table S-8 of the supplementary file.

Based on the comparison of results of EGG-noKP and EGG shown in Table S-8, it can be seen that the performance of EGG deteriorated when the knee points were not used, as EGG-noKP was respectively better than, worse than and similar to EGG on 7, 38, and 20 out of 65 test problems. In other words, EGG performed better than or similarly to EGG-noKP on 58 out of 65 (89%) test instances. These experimental results confirmed that the selection of knee points for mating parents can provide better genes for EGG to participate both of the exchanged gene model in (7) and the disturbance method in (8).

Based on the above analysis of three components of EGG, it can be concluded that the exchanged gene model, the disturbance method in (8) and the knee points as the mating parents have a positive impact on the performance of EGG. Moreover, based on the summarized comparison performance of these EGG variants in Table IV, it is found that EGG performed significantly better than EGG-noExchange, EGG-noKP, EGG( $Lr=0$ ), EGG( $Lr=0.5$ ) on 45, 38, 26 and 23 out of 65 test instances. Therefore, we can conclude that the exchanged gene model is the most important (first) contributor for the performance enhancement of EGG. The knee point as the mating parents was the second contributor for enhancing EGG. Moreover, as the knee point based mating selection mechanism can provide elite genes for the exchanged gene model, the synergy of those two parts can further enhance the performance of EGG. At last, the disturbance method in (8) was the third contributor for enhancing EGG, as EGG performed similarly to EGG( $Lr=0$ ) and EGG( $Lr=0.5$ ) respectively on 30 and 37 out of 65 test instances, and EGG performed significantly better than EGG( $Lr=0$ ) and EGG( $Lr=0.5$ ) respectively on 26 and 23 out of 65 test instances according to the Wilcoxon rank sum test.

## VI. CONCLUSION

In this paper, an elite gene guided reproduction operator (EGG) was designed for tackling MaOPs, by using three models (disturbance, exchange, and inheritance) to generate offspring solutions. The running of these models is controlled by a disturbance rate  $Dr$  and an exchange rate  $Er$ . When tackling MaOPs, our experiments indicated that EGG with a small value of  $Dr$  and a large value of  $Er$  showed a better overall performance. That is because the mating parents are only selected from a sub set of the current population, i.e., the knee point set. This mating selection mechanism cooperated with the exchanged gene model will have more elite genes reserved to the next generation. In addition, a new disturbance

method was proposed for EGG, in which most of the genes are disturbed by a small-size step while a few genes are perturbed using a large-size step. This new disturbance strategy leads to a hypercube search, which was validated to effectively enhance the performance of EGG when solving MaOPs. The experimental results showed the superior performance of our proposed EGG operator when it was embedded into three reference-based MaOEAs (MOEA/D, NSGA-III and  $\theta$ -DEA) and a non-reference-based MaOEA (SPEA2-SDE).

In the framework of MOEA/D, EGG is not so promising when compared with DE operator. That is mainly because EGG randomly selects the mating parents from the knee point set without considering the neighborhood information. In the future, the neighboring knee points around the current solution can be further used to enhance the performance under the MOEA/D framework. Moreover, it would also be interesting to extend our EGG to solve constrained MaOPs by incorporating constraint-handling techniques.

## REFERENCES

- [1] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2014.
- [2] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [3] K. Ikeda, H. Kita, and S. Kobayashi, "Failure of pareto-based MOEAs: Does non-dominated really mean near to optimal?" in *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No. 01TH8546)*, vol. 2. IEEE, 2001, pp. 957–962.
- [4] S. Yang, M. Li, X. Liu, and J. Zheng, "A grid-based evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 5, pp. 721–736, 2013.
- [5] Z. He, G. G. Yen, and J. Zhang, "Fuzzy-based pareto optimality for many-objective evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 2, pp. 269–285, 2014.
- [6] Y. Yuan, H. Xu, B. Wang, and X. Yao, "A new dominance relation-based evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 1, pp. 16–37, 2016.
- [7] M. Elarbi, S. Bechikh, A. Gupta, L. B. Said, and Y.-S. Ong, "A new decomposition-based NSGA-II for many-objective optimization," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 7, pp. 1191–1210, 2018.
- [8] Y. Tian, R. Cheng, X. Zhang, Y. Su, and Y. Jin, "A strengthened dominance relation considering convergence and diversity for evolutionary many-objective optimization," *IEEE Transactions on Evolutionary Computation*, 2018.
- [9] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *Evolutionary computation*, vol. 19, no. 1, pp. 45–76, 2011.
- [10] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2004, pp. 832–842.
- [11] N. Beume, B. Naujoks, and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653–1669, 2007.
- [12] F. Li, R. Cheng, J. Liu, and Y. Jin, "A two-stage R2 indicator based evolutionary algorithm for many-objective optimization," *Applied Soft Computing*, vol. 67, pp. 245–260, 2018.
- [13] Y. Sun, G. G. Yen, and Z. Yi, "IGD indicator-based evolutionary algorithm for many-objective optimization problems," *IEEE Transactions on Evolutionary Computation*, 2018.
- [14] Y. Tian, R. Cheng, X. Zhang, F. Cheng, and Y. Jin, "An indicator-based multiobjective evolutionary algorithm with reference point adaptation for better versatility," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 4, pp. 609–622, 2018.

- [15] T. Pamulapati, R. Mallipeddi, and P. N. Suganthan, "Isde+-an indicator for multi and many-objective optimization," *IEEE Transactions on Evolutionary Computation*, 2018.
- [16] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on evolutionary computation*, vol. 11, no. 6, pp. 712–731, 2007.
- [17] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II," *IEEE transactions on evolutionary computation*, vol. 13, no. 2, pp. 284–302, 2009.
- [18] L. Chen, H.-L. Liu, K. C. Tan, Y.-M. Cheung, and Y. Wang, "Evolutionary many-objective algorithm using decomposition-based dominance relationship," *IEEE transactions on cybernetics*, 2018.
- [19] Y. Zhou, Y. Xiang, Z. Chen, J. He, and J. Wang, "A scalar projection and angle-based evolutionary algorithm for many-objective optimization problems," *IEEE transactions on cybernetics*, no. 99, pp. 1–12, 2018.
- [20] X. Cai, Z. Mei, and Z. Fan, "A decomposition-based many-objective optimization algorithm with two types of adjustments for direction vectors," *IEEE transactions on cybernetics*, vol. 48, no. 8, pp. 2335–2348, 2018.
- [21] X. He, Y. Zhou, Z. Chen, and Q. Zhang, "Evolutionary many-objective optimization based on dynamical decomposition," *IEEE Transactions on Evolutionary Computation*, 2018.
- [22] K. Li, K. Deb, Q. Zhang, and S. Kwong, "An evolutionary many-objective optimization algorithm based on dominance and decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 5, pp. 694–716, 2015.
- [23] Y. Liu, D. Gong, X. Sun, and Y. Zhang, "Many-objective evolutionary optimization based on reference points," *Applied Soft Computing*, vol. 50, pp. 344–355, 2017.
- [24] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A reference vector guided evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 5, pp. 773–791, 2016.
- [25] Y. Liu, D. Gong, J. Sun, and Y. Jin, "A many-objective evolutionary algorithm using a one-by-one selection strategy," *IEEE Transactions on Cybernetics*, vol. 47, no. 9, pp. 2689–2702, 2017.
- [26] Q. Lin, S. Liu, Q. Zhu, C. Tang, R. Song, J. Chen, C. A. C. Coello, K.-C. Wong, and J. Zhang, "Particle swarm optimization with a balanceable fitness estimation for many-objective optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 1, pp. 32–46, 2018.
- [27] D. Gong, Y. Liu, and G. G. Yen, "A meta-objective approach for many-objective evolutionary optimization," *Evolutionary computation*, pp. 1–25, 2018.
- [28] R. Wang, R. C. Purshouse, and P. J. Fleming, "Preference-inspired co-evolutionary algorithms using weight vectors," *European Journal of Operational Research*, vol. 243, no. 2, pp. 423–441, 2015.
- [29] D. Gong, F. Sun, J. Sun, and X. Sun, "Set-based many-objective optimization guided by a preferred region," *Neurocomputing*, vol. 228, pp. 241–255, 2017.
- [30] C. Liu, Q. Zhao, B. Yan, S. Elsayed, T. Ray, and R. Sarker, "Adaptive sorting-based evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, 2018.
- [31] D. Gong, J. Sun, and Z. Miao, "A set-based genetic algorithm for interval many-objective optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 1, pp. 47–60, 2018.
- [32] R. B. Agrawal, K. Deb, and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex systems*, vol. 9, no. 2, pp. 115–148, 1995.
- [33] K. Deb and M. Goyal, "A combined genetic adaptive search (GeneAS) for engineering design," *Computer Science and informatics*, vol. 26, pp. 30–45, 1996.
- [34] R. Hinterding, "Gaussian mutation and self-adaption for numeric genetic algorithms," in *Proceedings of 1995 IEEE International Conference on Evolutionary Computation*, vol. 1. IEEE, 1995, p. 384.
- [35] K. Chellapilla, "Combining mutation operators in evolutionary programming," *IEEE transactions on Evolutionary Computation*, vol. 2, no. 3, pp. 91–96, 1998.
- [36] L. J. Eshelman and J. D. Schaffer, "Real-coded genetic algorithms and interval-schemata," in *Foundations of genetic algorithms*. Elsevier, 1993, vol. 2, pp. 187–202.
- [37] R. Storn and K. 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.
- [38] S. Tsutsui, M. Yamamura, and T. Higuchi, "Multi-parent recombination with simplex crossover in real coded genetic algorithms," in *Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation—Volume 1*. Morgan Kaufmann Publishers Inc., 1999, pp. 657–664.
- [39] K. Deb, A. Anand, and D. Joshi, "A computationally efficient evolutionary algorithm for real-parameter optimization," *Evolutionary computation*, vol. 10, no. 4, pp. 371–395, 2002.
- [40] G. Pavai and T. Geetha, "A survey on crossover operators," *ACM Computing Surveys (CSUR)*, vol. 49, no. 4, p. 72, 2017.
- [41] Q. Zhu, Q. Lin, and J. Chen, "A gene-level hybrid search framework for multiobjective evolutionary optimization," *Neural Computing and Applications*, vol. 30, no. 3, pp. 759–773, 2018.
- [42] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 398–417, 2009.
- [43] H. Sato, H. E. Aguirre, and K. Tanaka, "Genetic diversity and effective crossover in evolutionary many-objective optimization," in *International Conference on Learning and Intelligent Optimization*. Springer, 2011, pp. 91–105.
- [44] H. Ishibuchi, Y. Tanigaki, H. Masuda, and Y. Nojima, "Distance-based analysis of crossover operators for many-objective knapsack problems," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2014, pp. 600–610.
- [45] X. He, Y. Zhou, and Z. Chen, "An evolution path based reproduction operator for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, 2017.
- [46] M. Li, S. Yang, and X. Liu, "Shift-based density estimation for pareto-based algorithms in many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 3, pp. 348–365, 2014.
- [47] Q. Lin, S. Liu, K.-C. Wong, M. Gong, C. A. C. Coello, J. Chen, and J. Zhang, "A clustering-based evolutionary algorithm for many-objective optimization problems," *IEEE Transactions on Evolutionary Computation*, 2018.
- [48] X. Zhang, Y. Tian, and Y. Jin, "A knee point-driven evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 6, pp. 761–776, 2015.
- [49] Q. Zhu, Q. Lin, Z. Du, Z. Liang, W. Wang, Z. Zhu, J. Chen, P. Huang, and Z. Ming, "A novel adaptive hybrid crossover operator for multi-objective evolutionary algorithm," *Information Sciences*, vol. 345, pp. 177–198, 2016.
- [50] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable multi-objective optimization test problems," in *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No. 02TH8600)*, vol. 1. IEEE, 2002, pp. 825–830.
- [51] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 5, pp. 477–506, 2006.
- [52] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach," *IEEE transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.
- [53] J. Derrac, S. Garcia, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, 2011.
- [54] H. Ishibuchi, Y. Setoguchi, H. Masuda, and Y. Nojima, "Performance of decomposition-based many-objective algorithms strongly depends on pareto front shapes," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 2, pp. 169–190, 2017.



**Qingling Zhu** received the B.Sc. degree from the Nanchang Institution of Technology, Nanchang, China, in 2013, the M.Sc. degree from Shenzhen University, Shenzhen, China, in 2016. He is currently pursuing the Ph.D. degree with the Department of Computer Science, City University of Hong Kong, Hong Kong. His current research interests include evolutionary multiobjective optimization and machine learning.





**Qiuzhen Lin** received the B.S. degree from Zhaoqing University, Zhaoqing, China, in 2007, the M.S. degree from Shenzhen University, Shenzhen, China, in 2010, and the Ph.D. degree from the Department of Electronic Engineering, City University of Hong Kong, Hong Kong, in 2014.

He is currently a Lecturer with the College of Computer Science and Software Engineering, Shenzhen University. He has published over ten research papers since 2008. His current research interests include artificial immune system, multiobjective optimization, and dynamic system.

optimization, and dynamic system.



**Jianyong Chen** (M'11) received the Ph.D. degree from the City University of Hong Kong, Hong Kong, in 2003.

He is a Professor with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. He was with ZTE Corporation, Shenzhen, as a Senior Engineer of network technology from 2003 to 2006. Then he joined Shenzhen University. He has published over 30 papers and got over 30 patents in the field of artificial intelligence and information security. His

current research interests include in artificial intelligence and information security.

Dr. Chen was the Vice-Chairman of International Telecommunication Union-Telecommunication SG17 from 2004 to 2012, and an Editor of three recommendations developed in ITU-T SG17.



**Jianqiang Li** received his B.S. and Ph.D. Degree in automation major from South China University of Technology, Guangzhou, China, in 2003 and 2008, respectively.

He is a professor at the College of Computer and Software Engineering of Shenzhen University. He led a project of the National Natural Science Foundation, and a project of the Natural Science Foundation of Guangdong Province, China. His major research interests include embedded systems and Internet of Things.



**Carlos A. Coello Coello** (M'98–SM'04–F'11) received the Ph.D. degree in computer science from Tulane University, New Orleans, LA, USA, in 1996. He is currently a Professor (CINVESTAV-3F Researcher) with the Computer Science Department, CINVESTAV-IPN, Mexico City, Mexico. He has authored and co-authored over 450 technical papers and book chapters. He has also co-authored the book entitled *Evolutionary Algorithms for Solving Multi-Objective Problems* (Second Edition, Springer, 2007). His has over 45 000 Google

Scholar citations with an H-index of 83. His current research interests include evolutionary multiobjective optimization and constraint-handling techniques for evolutionary algorithms. Dr. Coello Coello was a recipient of the 2007 National Research Award from the Mexican Academy of Sciences in the area of Exact Sciences, the 2013 IEEE Kiyo Tomiyasu Award and the 2012 National Medal of Science and Arts in the area of Physical, Mathematical and Natural Sciences. He is currently an Associate Editor of the IEEE Transactions on Evolutionary Computation and serves in the editorial board of 12 other international journals. He is a member of the ACM and the Mexican Academy of Science.



**Jun Zhang** (F'17) received the Ph.D. degree from the City University of Hong Kong, Kowloon, Hong Kong, in 2002.

He is currently a visiting scholar with Victoria University, Melbourne, VIC, Australia. His current research interests include computational intelligence, cloud computing, high performance computing, operations research, and power electronic circuits.

Dr. Zhang was a recipient of the Changjiang Chair Professor from the Ministry of Education, China, in 2013, the China National Funds for Distinguished Young Scientists from the National Natural Science Foundation of China in 2011, and the First-Grade Award in Natural Science Research from the Ministry of Education, China, in 2009. He is currently an Associate Editor of the IEEE Transactions on Evolutionary Computation, the IEEE Transactions on Cybernetics, and the IEEE Transactions on Industrial Electronics.



**Zhong Ming** is a professor at College of Computer and Software Engineering of Shenzhen University. He is a member of a council and senior member of China Computer Federation. His major research interests are AI and cloud computing. He led two projects of National Natural Science Foundation, including one key project (61836005), one normal project (61672358)