

# An Opposition-based Repair Operator for Multi-objective Evolutionary Algorithm in Constrained Optimization Problems

Zhun Fan

Department of Electronic Engineering  
Shantou University  
Guangdong, Shantou 515063

Wenji Li

Department of Electronic Engineering  
Shantou University  
Guangdong, Shantou 515063

Xinye Cai

College of Computer Science and  
Technology, Nanjing University of  
Aeronautics and Astronautics  
Jiangsu, Nanjing 210016

Han Huang

School of Software Engineering  
South China University of  
Technology, Guangdong, Guangzhou 510006

Shuxiang Xie

Department of Electronic Engineering  
Shantou University  
Guangdong, Shantou 515063

Erik Goodman

BEACON Centre for the Study of  
Evolution in Action, Michigan State  
University, East Lansing, MI, 48824

**Abstract**—In this paper, we design a set of multi-objective constrained optimization problems (MCOPs) and propose a new repair operator to address them. The proposed repair operator is used to fix the solutions that violate the box constraints. More specifically, it employs a reversed correction strategy that can effectively avoid the population falling into local optimum. In addition, we integrate the proposed repair operator into two classical multi-objective evolutionary algorithms MOEA/D and NSGA-II. The proposed repair operator is compared with other two kinds of commonly used repair operators on benchmark problems CTPs and MCOPs. The experiment results demonstrate that our proposed approach is very effective in terms of convergence and diversity.

**Index Terms**—Multi-objective Evolutionary Algorithm, Repair Operator, Constrained Optimization.

## I. INTRODUCTION

Multi-objective optimization problems (MOPs) consist of more than one objectives, which are usually conflicting with each other. In other words, improvements in one objective may lead to the degradation of other objectives. It is impossible to make all of the objectives to be optimal at the same time. Instead, a set of solutions that represent the trade-off among multiple objectives exist for MOPs. In addition, different types of constraints are often unavoidable in MOPs. Such MOPs with constraints are usually termed multi-objective constrained optimization problem. Constraints can be roughly divided into two categories, equality and inequality constraints. Without loss of generality, a multi-objective constrained optimization problem can be defined as follows:

$$\begin{aligned} & \text{minimize} && F(x) = (f_1(x), \dots, f_m(x)) \\ & \text{subject to} && g_i(x) \geq 0, i = 1, \dots, q \\ & \text{subject to} && h_j(x) = 0, j = 1, \dots, p \end{aligned} \quad (1)$$

Where  $x = (x_1, x_2, \dots, x_n) \subset R^n$  is  $n$ -dimensional design variables,  $F(x) = (f_1(x), f_2(x), \dots, f_m(x)) \subset R^m$  is  $m$ -

dimensional objective vector.  $g_i(x) \geq 0$  define  $q$  inequality constraints,  $h_j(x) = 0$  define  $p$  equality constraints.

The existing multi-objective constrained evolutionary algorithms combine the multi-objective evolutionary algorithms with the mechanisms of constraint handling [1]. At present, NSGA-II [2] and MOEA/D [3] are the two classical multi-objective evolutionary algorithms representing two categories of fitness assignment methods, namely fitness assignment based on domination and decomposition. In fitness assignment based on domination, the fitness is decided by non-dominated sorting and crowding distance. Representative algorithms using this type of fitness assignment method include MOGA [4], PAES-II [5], SPEA-II [6] and NSGA-II [2]. In fitness assignment based on decomposition, comparison and sorting of individuals are made via aggregation function with weights allocated specifically to all individuals. Typical algorithms of this category include IMMOGLS [7], UGA [8], cMOGA [9], MOGLS [10], and MOEA/D [3].

The existing constraints handling mechanism can be divided into four categories. They are feasibility maintenance, penalty function, separation of constraint violation and objective value, and multi-objective evolutionary algorithms (MOEAs). The methods of feasibility maintenance are usually applied to the discrete optimization problems, such as the job shop scheduling problems and the vehicle routing problems. They either design appropriate coding and decoding methods to ensure that the individuals are feasible, or apply some mechanisms to repair the infeasible individuals. Unlike the feasibility maintenance methods, the penalty function method is adding one penalty term to the objective functions and transforming the constrained optimization problem into an unconstrained one. Typical methods of this category include segregated penalty functions [17], death penalty functions [18], co-evolutionary penalty functions [19] and adaptive penalty functions [20,21].

Finding an ideal punishment factor is usually very difficult for the penalty function method. The mechanism of separation of constraint violation and objective value treats the objective and constraints separately, and does not need any extra parameters. Typical methods of this category include stochastic ranking (SR) [11], infeasible driven evolutionary algorithm (IDEA) [12] and constraint dominate principle (CDP) [13]. The main feature of MOEAs is to transform a multi-objective constrained optimization problem to another multi-objective optimization problem with an additional objective, which regards the constraint condition as another objective and uses the existing MOEAs to optimize the transformed problem. Typical methods of this category include COMOGA [14], CW [15] and ATMES [16]. It is noteworthy that the penalty function method needs to tune the punishment factor, and the MOEAs method brings additional objective. In this paper, CDP method is used to handle constraints, which requires no additional parameters.

The remainder of the paper is organized as follows. Section II designs a set of multi-objective constrained optimization problems (MCOPs). Section III introduces the repair operator. Section IV gives the experimental results of the CTP and MCOP optimization problems, and Section V concludes the paper.

## II. DESIGN OF MCOPs

The existing multi-objective constrained optimization problems mainly consist of CTP [22,23] and CF[24]. CTP benchmark problems can be defined as follows:

$$\begin{aligned} & \text{minimize} && f_1(x) = x_1 && (2) \\ & \text{minimize} && f_2(x) = g(x)(1 - \sqrt{f_1}/g) \\ & \text{subject to} && C(x) \equiv \cos(\theta)(f_2(x) - e) - \sin(\theta)f_1(x) \geq \\ & && a|\sin(b\pi(\sin(\theta)(f_2(x) - e) + \cos(\theta)f_1(x))^c)|^d \end{aligned}$$

It is important to note that the problem can be made harder by setting  $g(x)$  function with various local extreme. The inequality constraint  $C(x)$  has six parameters  $(\theta, a, b, c, d, e)$ . In fact, the above problem can be used as a constrained test problem generator by tuning these six parameters. Deb et al designed seven benchmark problems named CTP2-CTP8 by setting those six parameters. The original CTP2-CTP8 instances have only 2 decision variables and they are easy to solve. Hence, we extend the CTP2-CTP8 problems to ten decision variables and variable bounds are given by  $0 \leq x_i \leq 1, i = 1, \dots, 10$ . The six constraint parameters are the same as those used in [22].

According to the final report on CEC09 MOEA competition, MOEA/D and NSGA-II are not quite suitable for solving CF instances. Even though it is very easy to search feasible solutions for CF, finding the true Pareto front turns out to be very difficult. This paper mainly focuses on applying the repair operators in the framework of MOEA/D and NSGA-II. Because CF is not a suitable test suite for MOEA/D and NSGA-II, we design a new set of multi-objective constrained optimization problems (MCOPs) to validate the proposed

repair operator in the framework of MOEA/D and NSGA-II. Unlike CTP2-CTP8 instances which have the same multi-objectives and each problem has different constraint conditions by selecting six different parameters, we design a set of problems that have different multi-objectives but share the same constraint conditions. In terms of objective functions, we adopt ZDT test problems [28] and make some changes. In addition, nine ellipses are established in the objective space as the constraint conditions. The general form of constraint conditions are as follows:

$$\begin{aligned} C(x) \equiv & ((f_1(x) - c_x)\cos(\theta) - (f_2(x) - c_y)\sin(\theta))^2/a^2 & (3) \\ & + ((f_1(x) - c_x)\sin(\theta) + (f_2(x) - c_y)\cos(\theta))^2/b^2 \geq 1 \end{aligned}$$

The constraint  $C(x)$  has five parameters  $(\theta, a, b, C_x, C_y)$ , which can be used to further adjust the difficulty levels of the constraint conditions as needed. Among them  $\theta$  denotes the counterclockwise rotation angle of the ellipse.  $a$  and  $b$  control the lengths of the long axis and minor axis of the ellipse respectively.  $C_x$  and  $C_y$  are two vectors representing the centers of the ellipses. For example, if we define the following parameters:

$$\begin{aligned} c_x &= [0, 1, 0, 1, 2, 0, 1, 2] & (4) \\ c_y &= [1.5, 0.5, 2.5, 1.5, 0.5, 3.5, 2.5, 1.5, 0.5] \\ a^2 &= 0.1, b^2 = 0.2, \theta = -0.25\pi \end{aligned}$$

The distribution of constraining ellipses in the objective space is shown in Figure 1.

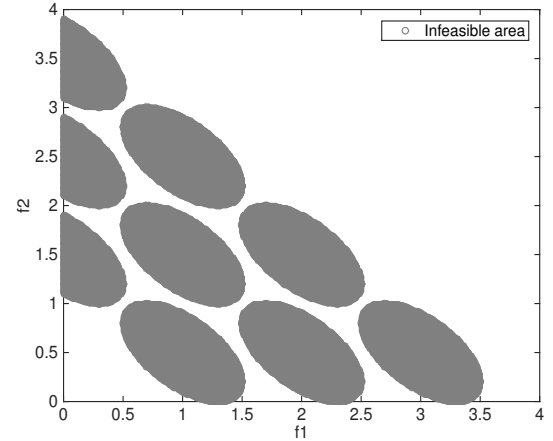


Fig. 1. The distribution of constraint functions.

Combining the constraint functions with objective functions, we design seven multi-objective constraint optimization problems, namely MCOP1-MCOP7. The objective functions of them are listed in Table 1. It is noteworthy that we use constraint dominate principle (CDP) method to handle the constraints defined in formula (2) and formula (3), and use repair operators to fix box constraint.

TABLE I  
OBJECTIVE FUNCTIONS OF MCOP1-MCOP7.

Function Name	Function Definition
MCOP1 PF convex	$\begin{cases} \text{minimize} & f_1(x) = g(x)x_1 \\ \text{minimize} & f_2(x) = g(x)(1 - \sqrt{f_1/g}) \\ & g(x) = 1 + 9 \sum_{i=2}^m x_i/(m-1) \\ & m = 30, x_i \in [0, 1] \end{cases}$
MCOP2 PF discrete	$\begin{cases} \text{minimize} & f_1(x) = g(x)x_1 \\ \text{minimize} & f_2(x) = g(x)(1 - (f_1/g)^2) \\ & g(x) = 1 + 9 \sum_{i=2}^m x_i/(m-1) \\ & m = 30, x_i \in [0, 1] \end{cases}$
MCOP3 PF discrete	$\begin{cases} \text{minimize} & f_1(x) = x_1 \\ \text{minimize} & f_2(x) = g(x)(1 - \sqrt{f_1/g}) \\ & -f_1 \sin(10\pi f_1) \\ & g(x) = 1 + 9(\sum_{i=2}^m \frac{x_i}{m-1})^{0.25} \\ & m = 10, x_i \in [0, 1] \end{cases}$
MCOP4 PF convex	$\begin{cases} \text{minimize} & f_1(x) = g(x)x_1 \\ \text{minimize} & f_2(x) = g(x)(1 - \sqrt{f_1/g}) \\ & g(x) = 1 + 10(m-1) \\ & + \sum_{i=2}^m (x_i^2 - 10\cos(4\pi x_i)) \\ & m = 10, x_i \in [0, 1] \end{cases}$
MCOP5 PF discrete	$\begin{cases} \text{minimize} & f_1(x) = g(x)x_1 \\ \text{minimize} & f_2(x) = g(x)(1 - (f_1/g)^2) \\ & g(x) = 1 + 10(m-1) \\ & + \sum_{i=2}^m (x_i^2 - 10\cos(4\pi x_i)) \\ & m = 10, x_i \in [0, 1] \end{cases}$
MCOP6 PF discrete	$\begin{cases} \text{minimize} & f_1(x) = 1 - \exp(-4x_1) \sin^6(6\pi x_1) \\ \text{minimize} & f_2(x) = g(x)(1 - (f_1/g)^2) \\ & g(x) = 1 + 10(m-1) \\ & + \sum_{i=2}^m (x_i^2 - 10\cos(4\pi x_i)) \\ & m = 10, x_i \in [0, 1] \end{cases}$
MCOP7 PF convex	$\begin{cases} \text{minimize} & f_1(x) = 1 - \exp(-4x_1) \sin^6(6\pi x_1) \\ \text{minimize} & f_2(x) = g(x)(1 - \sqrt{f_1/g}) \\ & g(x) = 1 + 10(m-1) \\ & + \sum_{i=2}^m (x_i^2 - 10\cos(4\pi x_i)) \\ & m = 10, x_i \in [0, 1] \end{cases}$

### III. REPAIR OPERATOR

Repair operators are used to fix the infeasible solutions that violate the box constraints. A lot of research concentrates on repairing the infeasible solutions for discrete multi-objective constrained optimization problems. However, very few researchers have paid attention to the repair operators for continuous multi-objective constrained optimization problems. In fact, there are very large proportion of infeasible solutions which violate the box-constraint during the differential evolutionary process. Fixing the infeasible solutions properly will significantly help multi-objective evolutionary algorithms to find the ideal Pareto front. At present, there are two commonly used repair operators. One of the most commonly used repair operator can be defined as follows:

$$x_{i,j} = \begin{cases} L_j, & \text{if } x_{i,j} < L_j \\ U_j, & \text{if } x_{i,j} > U_j \end{cases} \quad (5)$$

Where  $x_{i,j}$  represents the value of  $j$ -th component of individual  $i$ .  $L_j$  denotes the lower bound of  $j$ -th component of the decision variables.  $U_j$  denotes the upper bound of  $j$ -th

component of the decision variables. Another commonly used repair operator proposed by Wang etc. [26] can be defined as follows:

$$x_{i,j} = \begin{cases} \min\{U_j, 2L_j - x_{i,j}\}, & \text{if } x_{i,j} < L_j \\ \max\{L_j, 2U_j - x_{i,j}\}, & \text{if } x_{i,j} > L_j \end{cases} \quad (6)$$

In order to facilitate discussion, we denote the formula (5) and formula (6) as Repair-A and Repair-B respectively. In this paper, we propose an opposition-based repair operator denoted as Repair-C. The formula of our proposed repair operator can be defined as follows:

$$x_{i,j} = \begin{cases} U_j, & \text{if } x_{i,j} < L_j \\ L_j, & \text{if } x_{i,j} > U_j \end{cases} \quad (7)$$

This repair operator is inspired in part by the concept of opposition-based learning (OBL)[29]. The main idea of OBL is, for finding a better candidate solution, simultaneous consideration of an estimate and its corresponding opposite estimate has a potential to help search towards the global optimum in a more efficient way, due to an arguably better preservation of diversity in the searching population. For example, the differential evolution process can be defined as follows:

$$x'_{i,j} = x_{i,j} + F(x_{r_1,j} - x_{r_2,j}) \quad (8)$$

Where  $r_1$  and  $r_2$  are two unequal random integers and not equal to  $i$ .  $F$  denotes the factor of differential evolution, here we set  $F = 0.5$ . If  $x'_{i,j}$  is less than its lower bound  $L_j$ , it can be inferred that  $x_{i,j}$  has a higher probability of getting a value close to its lower bound  $L_j$ . In this case if we fix  $x'_{i,j}$  to its upper bound  $U_j$ , which can be approximately considered as an opposite estimate of  $x_{i,j}$ , then this operator has a potential to increase the diversity of the population according to the philosophy of OBL. Even though this choice is a bit counterintuitive because normally people think fixing  $x'_{i,j}$  to its lower bound  $L_j$  is a better choice, but we shall also not ignore the possibility that fixing  $x'_{i,j}$  to its lower bound, which is a value with a lot loss of potential after many previous search attempts, the search may have a higher likelihood to be stuck in local minima. To verify this hypothesis, we conduct a lot of experiments which are described in detail in the Section of Experimental Study.

### IV. EXPERIMENTAL STUDY

#### A. Experimental Settings

In order to evaluate the performance of repair operators mentioned in section 3, we combined these three repair operators with NSGA-II and MOEA/D and then studied the experimental results on CTP2-CTP8 and MCOP1-MCOP7. Thirty independent runs with the six algorithms are conducted. The detailed parameter settings of these six algorithms are summarized as follows.

1) Setting for reproduction operators: The mutation probability  $P_m = 1/n$  ( $n$  is the number of decision variables) and its distribution index is set to be 20. For the DE operator, we set  $CR = 1.0$  and  $F = 0.5$  as recommended in [27].

- 2) Population size:  $N = 200$ .
- 3) Number of runs and stopping condition: Each algorithm runs 30 times independently on each test problems. The algorithm stops until 500 000 function evaluations.
- 4) Neighborhood size:  $T = 20$ .
- 5) Probability use to select in the neighborhood:  $\delta = 0.9$ .
- 6) The maximal number of solutions replaced by a child:  $nr = 2$ .

### B. Performance Metric

In this work, performance of a constrained multi-objective evolutionary algorithm is evaluated in two aspects convergence and distribution. Convergence describes the closeness of the obtained Pareto front to the true Pareto front. Distribution on the other hand depicts how the solutions in the obtained Pareto are distributed. We select two metrics - inverted generation distance (IGD)[30] and hypervolume (HV)[31]. Detailed definitions of them are given as follows:

Inverted Generational Distance (IGD): Let  $p^*$  is the ideal Pareto front set,  $A$  is an approximate Pareto front set achieved by evolutionary multi-objective algorithm. IGD metric denotes the distance between  $p^*$  and  $A$ . It is defined as follows:

$$\begin{cases} IGD(P^*, A) = \frac{\sum_{y^* \in P^*} d(y^*, A)}{\|P^*\|} \\ d(y^*, A) = \min_{y \in A} \left\{ \sqrt{\sum_{i=1}^m (y_i^* - y_i)^2} \right\} \end{cases} \quad (9)$$

Hypervolume (HV): HV simultaneously considers the distribution of the obtained Pareto front  $A$  and its vicinity to the true Pareto front. HV is defined as the volume enclosed by  $A$  and the reference vector  $r = (r_1, \dots, r_m)$ . HV can be defined as:

$$HV(P) = vol_{i \in P}(i) \quad (10)$$

Here,  $vol(i)$  represents the volume enclosed by solution  $i \in A$  and the reference vector  $r$ . The maximum value of each objective in the ideal Pareto front set gives the value of each dimension of the reference point  $r$ , and thus constructs the reference point.

### C. Experimental Result

In order to demonstrate the effectiveness of the proposed repair operator, we first compared it with the other two repair operators Operator-A and Operator-B (discussed in section III) in the framework of MOEA/D on CTP2-CTP8 and MCOP1-MCOP7 problems. The final populations with the best hypervolume metric in 30 independent runs with the three repair operators are shown in Figure 2.

From Figure 2, it is clear that MOEA/D-Repair-C has obtained best Pareto fronts on CTP3, CTP7, MCOP5, MCOP6 and MCOP7. For CTP2, CTP5 and MCOP4, MOEA/D-Repair-A and MOEA/D-Repair-C have a similar Pareto front, which is better than the Pareto front obtained by MOEA/D-Repair-B. MOEA/D-Repair-B and MOEA/D-Repair-C have a better Pareto front than MOEA/D-Repair-A on CTP4. For CTP6, CTP8, MCOP1, MCOP2 and MCOP3, the three methods have similar Pareto fronts. Overall MOEA/D-Repair-C

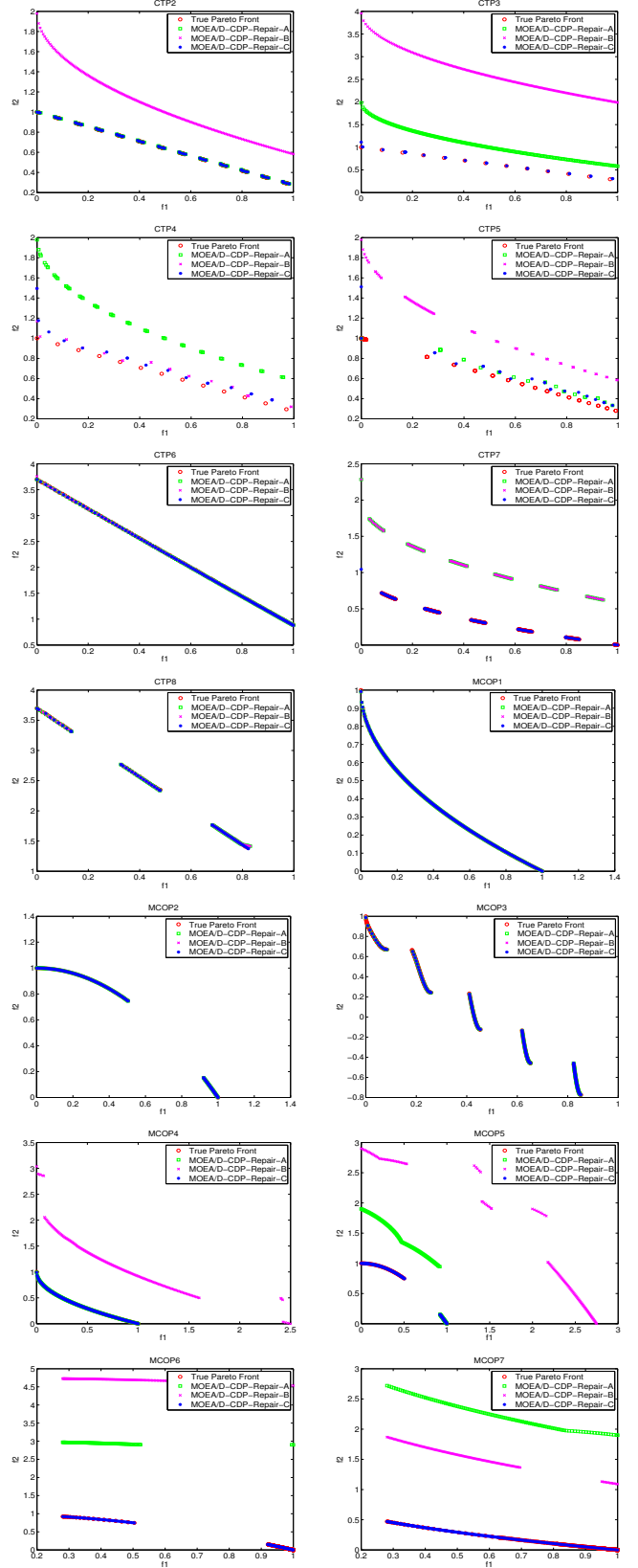


Fig. 2. The final populations with the best hypervolume metric in 30 independent runs

TABLE II  
IGD VALUES OF MOEA/D-REPAIR-A, MOEA/D-REPAIR-B AND MOEA/D-REPAIR-C.

Instance	MOEA/D-Repair-A		MOEA/D-Repair-B		MOEA/D-Repair-C	
	Mean	Std.	Mean	Std.	Mean	Std.
CTP2	6.81E-02	3.86E-02	1.46E-01	7.84E-02	<b>1.70E-04</b>	3.15E-06
CTP3	1.63E-01	9.45E-02	3.37E-01	1.71E-01	<b>1.40E-03</b>	1.71E-04
CTP4	4.68E-01	2.21E-01	5.88E-01	4.67E-01	<b>4.41E-02</b>	2.83E-02
CTP5	8.08E-02	3.70E-02	1.38E-01	9.20E-02	<b>6.85E-03</b>	3.88E-03
CTP6	1.29E-02	9.65E-03	1.28E-01	7.51E-02	<b>5.04E-04</b>	1.99E-05
CTP7	1.17E-01	4.86E-02	1.58E-01	7.13E-02	<b>1.39E-04</b>	2.77E-07
CTP8	3.77E-02	5.80E-02	2.72E-01	1.06E-01	<b>1.12E-03</b>	1.32E-04
MCOP1	2.39E-04	2.33E-06	2.40E-04	3.84E-06	<b>2.37E-04</b>	3.68E-06
MCOP2	2.56E-04	1.47E-07	2.56E-04	1.71E-07	2.56E-04	4.65E-07
MCOP3	2.71E-04	4.94E-07	2.71E-04	1.82E-06	2.71E-04	3.03E-06
MCOP4	8.96E-02	3.01E-02	1.21E-01	2.85E-02	<b>1.55E-02</b>	1.62E-02
MCOP5	1.67E-01	6.14E-02	2.88E-01	9.92E-02	<b>3.27E-02</b>	3.49E-02
MCOP6	1.03E-01	2.05E-02	1.23E-01	2.47E-02	<b>2.89E-02</b>	2.08E-02
MCOP7	1.09E-01	3.09E-02	1.28E-01	3.80E-02	<b>4.63E-02</b>	1.60E-02

TABLE III  
WILCOXON'S RANK SUM TEST AT A 0.05 SIGNIFICANCE LEVEL VALUES OF IGD AMONG MOEA/D-REPAIR-A, MOEA/D-REPAIR-B AND MOEA/D-REPAIR-C.

Instance	Repair-C vs Repair-A		Repair-C vs Repair-B	
	h-value	p-value	h-value	p-value.
CTP2	<b>1.00E+00</b>	4.98E-11	<b>1.00E+00</b>	3.02E-11
CTP3	<b>1.00E+00</b>	3.02E-11	<b>1.00E+00</b>	3.02E-11
CTP4	<b>1.00E+00</b>	4.98E-11	<b>1.00E+00</b>	1.86E-09
CTP5	<b>1.00E+00</b>	5.07E-10	<b>1.00E+00</b>	3.02E-11
CTP6	<b>1.00E+00</b>	1.07E-07	<b>1.00E+00</b>	3.96E-09
CTP7	<b>1.00E+00</b>	3.02E-11	<b>1.00E+00</b>	3.02E-11
CTP8	<b>1.00E+00</b>	6.20E-04	<b>1.00E+00</b>	2.94E-08
MCOP1	0.00E+00	8.50E-02	<b>1.00E+00</b>	1.25E-04
MCOP2	0.00E+00	9.63E-02	0.00E+00	9.05E-02
MCOP3	0.00E+00	1.45E-01	0.00E+00	2.84E-01
MCOP4	<b>1.00E+00</b>	8.10E-10	<b>1.00E+00</b>	3.02E-11
MCOP5	<b>1.00E+00</b>	2.23E-09	<b>1.00E+00</b>	3.02E-11
MCOP6	<b>1.00E+00</b>	8.99E-11	<b>1.00E+00</b>	3.69E-11
MCOP7	<b>1.00E+00</b>	1.52E-09	<b>1.00E+00</b>	5.49E-11

outperforms or is at least competitive with MOEA/D-Repair-A and MOEA/D-Repair-B in all test cases.

Table II presents the average values of IGD over 30 independent runs in the framework of MOEA/D with three repair operators described in section III. Table III presents the Wilcoxon's rank sum test values of IGD among these three different repair operators at a 0.05 significance level. It can be observed that MOEA/D-Repair-C performs significantly better than other two methods on all the instances except for MCOP1, MCOP2 and MCOP3, and almost the same as the other two kinds of repair operators on MCOP1, MCOP2 and MCOP3 (with slightly bigger standard deviation). The main cause is that MCOP1, MCOP2 and MCOP3 are relatively easy to solve that these three different repair operators can not reflect differences on them.

Table IV presents the average values of HV over 30 independent runs in the framework of MOEA/D. Table V presents the Wilcoxon's rank sum test values of HV among these three repair operators. From Table IV and Table V, it can be observed that MOEA/D-Repair-C also performs significantly better than the other two kinds of methods on all the instances except for MCOP1, MCOP2 and MCOP3, which means that our proposed repair operator can effectively avoid

TABLE V  
WILCOXON'S RANK SUM TEST AT A 0.05 SIGNIFICANCE LEVEL VALUES OF HV AMONG MOEA/D-REPAIR-A, MOEA/D-REPAIR-B AND MOEA/D-REPAIR-C.

Instance	Repair-C vs Repair-A		Repair-C vs Repair-B	
	h-value	p-value	h-value	p-value.
CTP2	<b>1.00E+00</b>	1.11E-10	<b>1.00E+00</b>	2.37E-12
CTP3	<b>1.00E+00</b>	9.40E-12	<b>1.00E+00</b>	1.21E-12
CTP4	<b>1.00E+00</b>	4.60E-12	<b>1.00E+00</b>	9.04E-10
CTP5	<b>1.00E+00</b>	7.82E-10	<b>1.00E+00</b>	5.91E-10
CTP6	<b>1.00E+00</b>	1.07E-07	<b>1.00E+00</b>	3.96E-09
CTP7	<b>1.00E+00</b>	5.22E-12	<b>1.00E+00</b>	2.37E-12
CTP8	<b>1.00E+00</b>	1.95E-03	<b>1.00E+00</b>	2.94E-08
MCOP1	1.00E+00	3.02E-11	1.00E+00	3.02E-11
MCOP2	1.00E+00	3.02E-11	1.00E+00	2.15E-10
MCOP3	1.00E+00	3.02E-11	1.00E+00	4.50E-11
MCOP4	<b>1.00E+00</b>	5.58E-10	<b>1.00E+00</b>	2.37E-12
MCOP5	<b>1.00E+00</b>	8.78E-09	<b>1.00E+00</b>	5.77E-11
MCOP6	<b>1.00E+00</b>	5.77E-11	<b>1.00E+00</b>	5.77E-11
MCOP7	<b>1.00E+00</b>	4.19E-02	<b>1.00E+00</b>	4.19E-02

TABLE VII  
WILCOXON'S RANK SUM TEST AT A 0.05 SIGNIFICANCE LEVEL VALUES OF IGD AMONG NSGAI-REPAIR-A, NSGAI-REPAIR-B AND NSGAI-REPAIR-C.

Instance	Repair-C vs Repair-A		Repair-C vs Repair-B	
	h-value	p-value	h-value	p-value.
CTP2	<b>1.00E+00</b>	6.01E-08	<b>1.00E+00</b>	3.02E-11
CTP3	<b>1.00E+00</b>	2.15E-10	<b>1.00E+00</b>	3.02E-11
CTP4	<b>1.00E+00</b>	5.00E-09	<b>1.00E+00</b>	5.49E-11
CTP5	<b>1.00E+00</b>	4.98E-11	<b>1.00E+00</b>	4.08E-11
CTP6	<b>1.00E+00</b>	4.08E-11	<b>1.00E+00</b>	6.70E-11
CTP7	<b>1.00E+00</b>	8.99E-11	<b>1.00E+00</b>	3.02E-11
CTP8	<b>1.00E+00</b>	4.18E-09	<b>1.00E+00</b>	1.25E-07
MCOP1	1.00E+00	3.18E-03	1.00E+00	2.60E-05
MCOP2	1.00E+00	2.38E-03	1.00E+00	6.57E-02
MCOP3	1.00E+00	1.96E-10	1.00E+00	4.80E-07
MCOP4	<b>1.00E+00</b>	3.02E-11	<b>1.00E+00</b>	3.02E-11
MCOP5	<b>1.00E+00</b>	1.21E-10	<b>1.00E+00</b>	3.02E-11
MCOP6	<b>1.00E+00</b>	1.86E-09	<b>1.00E+00</b>	3.02E-11
MCOP7	<b>1.00E+00</b>	1.96E-10	<b>1.00E+00</b>	3.02E-11

the population falling into local optimum in the framework of MOEA/D. For MCOP1, MCOP2 and MCOP3, the three methods obtain almost the same hypervolume(HV) results.

From Table VI and and Table VII, It can be observed the similar results obtained in the framework of MOEA/D. NSGA-II-Repair-C performs better than other two algorithms NSGA-II-Repair-A and NSGA-II-Repair-B in terms of IGD metric on

TABLE IV  
HV VALUES OF MOEA/D-REPAIR-A, MOEA/D-REPAIR-B AND MOEA/D-REPAIR-C.

Instance	MOEA/D-Repair-A		MOEA/D-Repair-B		MOEA/D-Repair-C	
	Mean	Std.	Mean	Std.	Mean	Std.
CTP2	5.00E-02	1.02E-01	9.76E-03	3.71E-02	<b>4.77E-01</b>	9.83E-05
CTP3	3.83E-02	6.45E-02	0.00E+00	0.00E+00	<b>4.45E-01</b>	1.90E-03
CTP4	8.43E-03	3.21E-02	3.84E-02	8.43E-02	<b>3.08E-01</b>	8.97E-02
CTP5	2.16E-02	7.48E-02	1.83E-02	4.75E-02	<b>2.52E-01</b>	1.03E-01
CTP6	4.04E-01	8.34E-02	1.14E-01	1.89E-01	<b>4.99E-01</b>	2.21E-04
CTP7	2.43E-03	5.53E-03	9.74E-04	3.71E-03	<b>5.46E-01</b>	2.56E-05
CTP8	3.08E-01	1.40E-01	5.59E-02	1.46E-01	<b>4.41E-01</b>	4.72E-04
MCOP1	6.64E-01	1.78E-05	6.64E-01	2.05E-05	6.64E-01	6.06E-05
MCOP2	2.21E-01	8.54E-06	2.21E-01	1.50E-05	2.21E-01	4.60E-05
MCOP3	<b>5.16E-01</b>	3.43E-06	5.15E-01	2.23E-05	5.15E-01	2.98E-05
MCOP4	3.37E-02	1.22E-01	2.53E-04	9.64E-04	<b>4.93E-01</b>	1.70E-01
MCOP5	3.55E-03	1.40E-02	0.00E+00	0.00E+00	<b>1.14E-01</b>	1.09E-01
MCOP6	0.00E+00	0.00E+00	0.00E+00	0.00E+00	<b>5.90E-02</b>	1.02E-01
MCOP7	0.00E+00	0.00E+00	0.00E+00	0.00E+00	<b>4.22E-02</b>	1.20E-01

TABLE VI  
IGD VALUES OF NSGA-II-REPAIR-A, NSGA-II-REPAIR-B AND NSGA-II-REPAIR-C.

Instance	NSGA-II-Repair-A		NSGA-II-Repair-B		NSGA-II-Repair-C	
	Mean	Std.	Mean	Std.	Mean	Std.
CTP2	4.66E-02	3.23E-02	1.17E-01	6.41E-02	<b>1.13E-04</b>	4.49E-05
CTP3	1.03E-01	6.32E-02	3.12E-01	1.45E-01	<b>2.95E-03</b>	9.83E-04
CTP4	2.90E-01	1.67E-01	6.18E-01	3.26E-01	<b>7.46E-02</b>	2.32E-02
CTP5	6.14E-02	2.32E-02	1.12E-01	5.13E-02	<b>1.01E-02</b>	4.84E-03
CTP6	1.08E-02	7.22E-03	8.48E-02	7.72E-02	<b>3.02E-04</b>	1.16E-04
CTP7	7.86E-02	4.44E-02	1.61E-01	8.43E-02	<b>5.25E-05</b>	1.54E-06
CTP8	1.98E-02	9.15E-03	1.61E-01	1.47E-01	<b>5.05E-04</b>	8.50E-05
MCOP1	<b>2.74E-04</b>	1.83E-05	4.52E-04	4.21E-05	7.20E-03	1.61E-02
MCOP2	<b>1.07E-02</b>	2.04E-02	1.60E-02	1.05E-02	2.56E-02	2.14E-02
MCOP3	<b>1.18E-04</b>	4.38E-06	2.57E-04	4.12E-05	1.45E-02	1.80E-02
MCOP4	1.01E-01	3.14E-02	1.36E-01	6.00E-02	<b>2.72E-04</b>	1.75E-05
MCOP5	2.27E-01	8.64E-02	3.36E-01	1.05E-01	<b>1.83E-04</b>	1.54E-05
MCOP6	1.02E-01	4.07E-02	1.57E-01	5.35E-02	<b>4.31E-05</b>	1.03E-05
MCOP7	1.13E-01	3.51E-02	1.70E-01	4.87E-02	<b>9.72E-05</b>	1.92E-05

TABLE VIII  
HV VALUES OF NSGA-II-REPAIR-A, NSGA-II-REPAIR-B AND NSGA-II-REPAIR-C.

Instance	NSGA-II-Repair-A		NSGA-II-Repair-B		NSGA-II-Repair-C	
	Mean	Std.	Mean	Std.	Mean	Std.
CTP2	1.01E-01	1.45E-01	2.42E-02	5.49E-02	<b>4.77E-01</b>	5.21E-04
CTP3	7.15E-02	9.84E-02	9.47E-03	3.61E-02	<b>4.31E-01</b>	9.21E-03
CTP4	4.51E-02	7.97E-02	8.42E-03	3.21E-02	<b>1.50E-01</b>	5.88E-02
CTP5	2.29E-02	5.21E-02	1.37E-02	4.19E-02	<b>1.54E-01</b>	6.62E-02
CTP6	4.24E-01	6.14E-02	1.82E-01	2.02E-01	<b>5.00E-01</b>	8.20E-04
CTP7	4.14E-02	1.38E-01	1.97E-02	9.96E-02	<b>5.47E-01</b>	4.14E-05
CTP8	3.63E-01	3.61E-02	1.80E-01	1.94E-01	<b>4.44E-01</b>	2.12E-04
MCOP1	<b>6.64E-01</b>	5.97E-05	6.59E-01	6.29E-04	5.85E-01	1.78E-01
MCOP2	<b>1.92E-01</b>	3.45E-02	1.27E-01	5.24E-02	8.03E-02	8.81E-02
MCOP3	<b>5.17E-01</b>	2.29E-05	5.10E-01	1.57E-03	3.15E-01	2.08E-01
MCOP4	8.14E-03	1.21E-02	1.15E-02	4.91E-02	<b>6.64E-01</b>	1.91E-04
MCOP5	7.38E-03	4.04E-02	0.00E+00	0.00E+00	<b>2.21E-01</b>	4.78E-05
MCOP6	2.38E-02	7.26E-02	0.00E+00	0.00E+00	<b>2.38E-01</b>	1.55E-05
MCOP7	1.83E-02	1.00E-01	0.00E+00	0.00E+00	<b>5.48E-01</b>	1.86E-04

all the instances except for MCOP1, MCOP2 and MCOP3.

Table VIII presents the average values of HV over 30 independent runs in the framework of NSGA-II. Table IX presents the Wilcoxon's rank sum test values of HV among three repair operators at a 0.05 significance level. From Table VIII and Table IX, it can be observed that NSGA-II-Repair-C performs significantly better than other two kinds of methods on all the instances except for MCOP1, MCOP2 and MCOP3. For MCOP1, MCOP2 and MCOP3, the three methods obtain almost the same results. It can be therefore concluded that the

proposed repair operator can also work well in the framework of NSGA-II.

## V. CONCLUSION

This paper proposes a new repair operator which employs a reversed correction strategy to fix the solutions that violate the box-constraint. In order to validate its performance on convergence and diversity, a new set of constrained multi-objective optimization problems is designed, to complement the well-known CTP test suite. Experimental results show

TABLE IX

WILCOXON'S RANK SUM TEST AT A 0.05 SIGNIFICANCE LEVEL VALUES OF HV AMONG NSGAI-REPAIR-A, NSGAI-REPAIR-B AND NSGAI-REPAIR-C.

Instance	Repair-C vs Repair-A		Repair-C vs Repair-B	
	h-value	p-value	h-value	p-value
-				
CTP2	<b>1.00E+00</b>	6.28E-08	<b>1.00E+00</b>	5.22E-12
CTP3	<b>1.00E+00</b>	1.33E-10	<b>1.00E+00</b>	2.37E-12
CTP4	<b>1.00E+00</b>	1.05E-07	<b>1.00E+00</b>	2.47E-10
CTP5	<b>1.00E+00</b>	5.83E-10	<b>1.00E+00</b>	1.59E-10
CTP6	<b>1.00E+00</b>	4.50E-11	<b>1.00E+00</b>	3.41E-11
CTP7	<b>1.00E+00</b>	4.31E-10	<b>1.00E+00</b>	4.11E-12
CTP8	<b>1.00E+00</b>	5.46E-09	<b>1.00E+00</b>	1.16E-10
MCOP1	1.00E+00	1.61E-10	1.00E+00	3.37E-05
MCOP2	1.00E+00	1.19E-06	1.00E+00	1.84E-02
MCOP3	1.00E+00	8.94E-11	1.00E+00	3.24E-07
MCOP4	<b>1.00E+00</b>	1.27E-11	<b>1.00E+00</b>	7.88E-12
MCOP5	<b>1.00E+00</b>	7.31E-12	<b>1.00E+00</b>	1.21E-12
MCOP6	<b>1.00E+00</b>	1.05E-09	<b>1.00E+00</b>	1.21E-12
MCOP7	<b>1.00E+00</b>	4.56E-11	<b>1.00E+00</b>	1.21E-12

that the proposed repair operator outperforms the other repair operators in terms of both convergence and diversity. The future work includes combining the proposed repair operator with other state-of-the-art algorithms to further validate the repair operator and improve the performance of the algorithms, and testing them in real-world applications.

#### ACKNOWLEDGMENT

This research work was supported by Guangdong Key Laboratory of Digital Signal and Image Processing, the National Natural Science Foundation of China under Grant (61175073, 61300159, 61332002, 51375287, 61370102, 61170193), Jiangsu Natural Science Foundation (BK20130808), Guangdong Natural Science Foundation (2014A030306050), the Fundamental Research Funds for the Central Universities, SCUT(2014ZG0043, 2015PT022), and the Ministry of Education - China Mobile Research Funds (MCM20130331).

#### REFERENCES

- [1] Y. Wang, Z.X. Cai, Y. R. Zhou, C. X. Xiao. Constrained Optimization Evolutionary Algorithms[J]. Journal of Software, 2009, 20(1):11-29.
- [2] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II[J]. Evolutionary Computation, IEEE Transactions on, 2002, 6(2): 182-197.
- [3] Zhang Q, Li H. MOEA/D: A multiobjective evolutionary algorithm based on decomposition [J]. Evolutionary Computation, IEEE Transactions on, 2007, 11(6): 712-731.
- [4] Fonseca C M, Fleming P J. Genetic Algorithms for Multiobjective Optimization: Formulation Discussion and Generalization[C]. In Proceedings of 5th International Conference on Genetic Algorithms, San Francisco, CA, USA, 1993, pp. 416-423.
- [5] Corne D W, Jerram N R, Knowles J D, et al. PESA-II: Region-based selection in evolutionary multiobjective optimization[C]. In Proceedings of the Genetic and Evolutionary Computation Conference, San Francisco, CA, USA, 2001, pp. 283-290.
- [6] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization[C]. Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems, Athens, Greece, 2001, pp. 19-26.
- [7] Ishibuchi H, Murata T. A multi-objective genetic local search algorithm and its application to flowshop scheduling[J]. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, 1998, 28(3): 392-403.
- [8] Leung Y W, Wang Y. Multiobjective programming using uniform design and genetic algorithm [J]. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, 2000, 30(3): 293-304.
- [9] Murata T, Ishibuchi H, Gen M. Specification of genetic search directions in cellular multi-objective genetic algorithms[C]. Evolutionary Multi-Criterion Optimization. Springer Berlin Heidelberg, 2001, pp. 82-95.
- [10] Jaskiewicz A. On the performance of multiple-objective genetic local search on the 0/1 knapsack problem-A comparative experiment [J]. Evolutionary Computation, IEEE Transactions on, 2002, 6(4): 402-412.
- [11] Runarsson T P, Yao X. Stochastic ranking for constrained evolutionary optimization [J]. Evolutionary Computation, IEEE Transactions on, 2000, 4(3): 284-294.
- [12] Ray T, Singh H K, Isaacs A, et al. Infeasibility driven evolutionary algorithm for constrained optimization[M]. Constraint-handling in evolutionary optimization. Springer Berlin Heidelberg, 2009, pp. 145-165.
- [13] Deb K. An efficient constraint handling method for genetic algorithms [J]. Computer methods in applied mechanics and engineering, 2000, 186(2): 311-338.
- [14] P. D. Surry and N. J. Radcliffe. The COMOGA method: Constrained optimization by multiobjective genetic algorithm[J]. Control And Cybernetics, 1997, 26(3):391-412.
- [15] Cai Z, Wang Y. A multiobjective optimization-based evolutionary algorithm for constrained optimization [J]. Evolutionary Computation, IEEE Transactions on, 2006, 10(6): 658-675.
- [16] Wang Y, Cai Z, Zhou Y, et al. An adaptive tradeoff model for constrained evolutionary optimization [J]. Evolutionary Computation, IEEE Transactions on, 2008, 12(1): 80-92.
- [17] R. Le. Riche, C. Knopf-Lenoir and R. T. Haftka. A Segregated Genetic Algorithm for Constrained structural Optimization[C]. Proceedings of the 6th International Conference on GA, San Francisco, USA., 1995, pp. 558-565.
- [18] F. Hoffmeister and J. Sprave. Problem-independent handling of constraints by use of metric penalty functions[C]. Proceedings of the 5th Annual Conference on Evolutionary Programming, San Diego, CA, USA., 1996, pp. 289-294.
- [19] Huang F, Wang L, He Q. An effective co-evolutionary differential evolution for constrained optimization [J]. Applied Mathematics and computation, 2007, 186(1): 340-356.
- [20] Ben Hamida S, Schoenauer M. ASCHEA: new results using adaptive segregational constraint handling[C]. In Proceedings of the 2002 Congress on Evolutionary Computation, Hawaii, USA., 2002, pp. 884-889.
- [21] Coit D W, Smith A E, Tate D M. Adaptive penalty methods for genetic optimization of constrained combinatorial problems[J]. INFORMS Journal on Computing, 1996, 8(2): 173-182.
- [22] K. Deb, Multi-Objective Optimization using Evolutionary Algorithms. John Wiley[M]. & Sons, Ltd, Chichester, 2001.
- [23] K. Deb, A. Pratap, and T. Meyarivan. Constrained test problems for multi-objective evolutionary optimization[C]. In Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization. Zurich, Switzerland, 2001, pp. 284-298.
- [24] Zhang Q, Zhou A, Zhao S, et al. Multiobjective optimization test instances for the CEC 2009 special session and competition[J]. University of Essex, Colchester, UK and Nanyang Technological University, Singapore, Special Session on Performance Assessment of Multi-Objective Optimization Algorithms, Technical Report, 2008.
- [25] Zitzler E, Deb K, Thiele L. Comparison of multiobjective evolutionary algorithms: Empirical results [J]. Evolutionary computation, 2000, 8(2): 173-195.
- [26] Wang Y, Cai Z, Zhang Q. Differential evolution with composite trial vector generation strategies and control parameters [J]. Evolutionary Computation, IEEE Transactions on, 2011, 15(1): 55-66.
- [27] Li H, Zhang Q. Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II[J]. Evolutionary Computation, IEEE Transactions on, 2009, 13(2): 284-302.
- [28] Deb K. Multi-objective genetic algorithms: Problem difficulties and construction of test problems [J]. Evolutionary computation, 1999, 7(3): 205-230.
- [29] Rahnemayan S, Tizhoosh H R, Salama M. Opposition-based differential evolution [J]. Evolutionary Computation, IEEE Transactions on, 2008, 12(1): 64-79.
- [30] Bosman P A N, Thierens D. The balance between proximity and diversity in multiobjective evolutionary algorithms[J]. Evolutionary Computation, IEEE Transactions on, 2003, 7(2): 174-188.
- [31] Zitzler E, Thiele L. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach[J]. evolutionary computation, IEEE transactions on, 1999, 3(4): 257-271.