# Diversity Preservation with Hybrid Recombination for Evolutionary Multiobjective Optimization

Sen Bong Gee Department of Electrical and Computer Engineering National University of Singapore Singapore 117576 Email: a0039834@nus.edu.sg

*Abstract*—Convergence and diversity are two crucial issues in evolutionary multiobjective optimization. To enhance the diversity property of Multiobjective Evolutionary Algorithm (MOEA), a novel selection method is implemented on decomposition-based MOEA (MOEA/D). The selection method incorporates the concept of maximum diversity loss, which quantifies the diversity loss of each individual in every generation. By monitoring tolerance of the diversity loss, the diversity of the solutions in each generation can be preserved. To further enhance the algorithm's search ability, a new hybrid recombination strategy is implemented by taking the advantage of different recombination operators. In terms of Inverted Generational Distance (IGD), the experiment results shown that the proposed algorithm, namely DHRS-MOEA/D, performed significantly better than many state-of-theart MOEAs in most of the CEC-09 and WFG test problems.

#### I. INTRODUCTION

There is a dramatic increase of studies in evolutionary multiobjective optimization (EMO) during the past two decades. One of the main reasons of EMO's popularity is because of its potential application in numerous complex engineering optimization problems. A multiobjective optimization problem (MOP) is an optimization problem which involves more than one objective and generally the objectives are conflicting to each other. The set of objective solutions which provide the best trade-off between different objectives is called Pareto optimal set. The main purpose of a multiobjective evolutionary algorithm (MOEA) is to find the Pareto optimal set of a given MOP.

Diversity and convergence are two important issues in any MOEA. Over the past two decades, a considerable number of research studies have concentrated on these two issues as they directly affect the optimization performance of any MOEA. There is a consensus that balance between these two issues plays a crucial role in an MOEA's optimization performance. Numerous researchers proposed different techniques to control the balance between these two properties of an MOEA [1]. To preserve the diversity of a MOEA's search process, the concepts of Niching [2], Weight Vector [3], Crowding [4]–[6] and others [7]–[10] have been proposed. Apart from diversity preservation, hybridization of metaheuristics is another popular technique used in the design of an MOEA.

Kay Chen Tan Department of Electrical and Computer Engineering National University of Singapore Singapore 117576 Email: eletankc@nus.edu.sg

There are various methods of the hybridization that have been proposed in the literature. Among these methods, hybridization of crossover operator has attracted much of the attention as it takes advantage of different operators during the search process [11].

The main purpose of this paper is to propose a novel MOEA which incorporates a diversity preservation selection operator and a hybrid recombination strategy. The algorithm is based on the decomposition-based MOEA [12] which decomposes an MOP into a set of single objective optimization problem. To evaluate the performance of the algorithm, Walking Fish Group (WFG) [13] and CEC-09 [14] test suite are used as benchmark test problems. Comparative studies are performed by comparing the proposed algorithm with other state-of-the-art algorithms. Section I gives a brief introduction to the field and the focus of the paper. Section II provides some and definitions commonly used in EMO. In Section III, the proposed algorithm design is illustrated. Section IV presents the experiment results of the paper and conclusions are drawn in Section V.

#### II. BACKGROUND

An MOP consists of more than one objective function that needs to be optimized simultaneously. Mathematically, an MOP can be expressed as

$$\begin{array}{ll} \underset{\mathbf{x}}{\text{minimize}} & \mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}) \ f_2(\mathbf{x}) \ \dots \ f_m(\mathbf{x})]^T \\ \text{subject to} & \mathbf{x} \in \Omega \end{array}$$
(1)

where  $f_i$  is the *i*-th objective function; *m* is the number of objective functions;  $\mathbf{f}(\mathbf{x}) \in \mathbb{R}^m$  is the objective vector; *n* is the number of decision variables;  $\mathbf{x} \in \mathbb{R}^n$  is the decision vector and  $\Omega \subset \mathbb{R}^n$  is the feasible decision space. Without loss of generality, a minimization problem is considered here. Generally,  $\Omega$  can be described by

$$\Omega = \{ \mathbf{x} \in \mathbb{R}^n | g_j(\mathbf{x}) \le 0 \text{ for } j = 1, \dots, p; h_k(\mathbf{x}) = 0, \text{ for } k = 1, \dots, q \}$$
(2)

where  $g_j$  is the *j*-th inequality constraints out of total *p* inequality equations and  $h_k$  is the *k*-th equality constraints out of total *q* equality equations. Since (1) involves more than

one objective function, they might be no single point in  $\Omega$  that minimizes all the objectives simultaneously. Therefore, the concepts of Pareto dominance and Pareto optimality are needed to define the solution set that provides the best trade-off between different objective functions.

Let  $\mathbf{u}, \mathbf{v} \in \mathbb{R}^m$  be two objective vectors. For the case of minimization,  $\mathbf{u}$  is said to dominate  $\mathbf{v}$  (or  $\mathbf{u} \leq \mathbf{v}$ ) if and only if  $u_i \leq v_i$  for all i = 1, 2, ..., m and there exists at least one  $u_i < v_i$ . Let  $F \subset \mathbb{R}^m$  be the feasible objective space which is mapped by  $\Omega$ . A decision vector  $\mathbf{x}^* \in \Omega$  is said to be a Pareto optimal solution if there is no vector in F that dominates  $\mathbf{f}(\mathbf{x}^*)$ . The objective vector of the Pareto optimal solution is called Pareto optimal objective vector. All the Pareto optimal solutions in  $\Omega$  jointly form the Pareto optimal set (POS) whereas all the Pareto Optimal Front (POF). A vector whose elements are the lower (upper for maximization problem) bounds of all objective is called the ideal vector. In general, the ideal vector is in infeasible objective space. For more detailed treatment on the concepts, please see [15]–[17].

#### **III. ALGORITHM DESIGN**

This section introduces the proposed algorithm Diversity Preservation with Hybrid Recombination Strategy Multiobjective Evolutionary Algorithm (DHRS-MOEA/D). This algorithm preserves the diversity of solutions by restricting the maximum allowable diversity loss (MRDL) [18]. Moreover, the algorithm enhances the search process by implementing the Hybrid Recombination Strategy (HRS) which incorporates different recombination operator in the algorithm.

#### A. Maximum Relative Diversity Loss

Relative diversity loss (RDL) is a measurement quantity that quantifies the diversity loss of an individual solution over two consecutive generations. To compute this quantity, ratio of the parent and offspring solution to the convergence direction is used to quantify the diversity loss.



Figure 1. Parent B and offspring C form a reference pair which is used to compute the relative diversity loss of parent A to offspring D

To illustrate the way of computing RDL, a simple example is given as shown in Fig. 1. Let's assume that we have a parent-offspring pair, **B** and **C**. For the ease of illustrations, **B** is named reference parent whereas **C** is called reference offspring. The RDL of parent **A** to offspring **D** (with respect to **B** and **C**) is defined as

$$\Gamma^{\mathbf{b} \to \mathbf{d}}_{\mathbf{d}_{\mathrm{conv},1}} = \frac{\triangle ABC}{\triangle BCD} \tag{3}$$

where  $\triangle ABC$  and  $\triangle BCD$  are the area of triangle **ABC** and **BCD** respectively. This equation basically estimates the ratio of spread between **AB** and **CD**. High value of the resulting  $\Gamma_{\mathbf{d}_{\text{conv},1}}^{\mathbf{b}\to\mathbf{d}}$  may indicate the shrink of solution spread in the objective space. Notice this equation (3) only consider two individuals in the population. If there are more than two individuals in the population, we need to modify equation (3) as follow

$$\Gamma^{\mathbf{p}\to\mathbf{c}} = \max_{i=1,\dots,k} \Gamma^{\mathbf{p}\to\mathbf{c}}_{\mathbf{d}_{\mathrm{conv},i}} \tag{4}$$

 $\Gamma^{\mathbf{p} \to \mathbf{c}}$  is called maximum RDL as it takes the highest  $\Gamma^{\mathbf{p} \to \mathbf{c}}_{\mathbf{d}_{\text{conv},i}}$  among the k parent-offspring pairs.

Convergence directions are needed to compute the MRDL. In this paper, the convergence direction is defined as a direction from a given point in the feasible objective space to the closest point on PF. Based on this definition, one offspringparent pair is used to estimate the convergence direction. The condition for the offspring-parent pair requires that offspring to dominate the parent solution and the parent solution must the closest to the offspring solution in the objective space among the parent population. The rationale behind the estimation is to use the domination relationship from the parent and offspring pool to predict the convergence direction.

Algorithm 1 shows the pseudocode of calculating the MRDL. The reference parents objective vector set (P) and reference offsprings objective vector set (C) are used to store the estimated convergence directions. At the start of each generation, the algorithm will clear these two reference sets (P and C). If these two reference sets are empty, the resulting MRDL is zero. If there is more than one objective vector in each reference set, the algorithm will calculate the RDL using different parent-offspring pairs and return the maximum RDL.

During the environmental selection, the MRDL quantity is computed for each individual solution in the population. If the MRDL is higher than a predefined value, the individual is discarded for the next generation. For more detailed descriptions for the MRDL, please refer [18].

#### B. Hybrid Recombination Strategy

As the fitness landscape is not known *a priori*, it is difficult to design a single crossover operator that works well during different evolutionary stage. Even in the same generation, a crossover operator may perform well (in term of generating good offspring solutions) on a subset of parent solutions whereas perform poor on other parent solutions. To circumvent this problem, hybridization of crossover operators is a feasible

# Algorithm 1 Compute the Maximum Relative Diversity Loss

# **Require:**

- P: The reference parents objective vector set.
- C: The reference offsprings objective vector set.
- x: The parent objective vector.
- y : The offspring objective vector.
- Require: |P| = |C|

**Ensure:** 

g: Maximum Relative Diversity Loss

```
1: if P = \emptyset then
2:
        g = 0
        return
3:
4: else
        s = |P|
5:
        max = 0
6:
       for i = 1 \rightarrow s do
Calculate r = \frac{\triangle x P_i C_i}{\triangle y P_i C_i}
 7:
8:
           if r > max then
 9:
10:
               max = r
           end if
11:
12:
        end for
13:
        g = max
14:
        return
15: end if
```

approach as it relaxes the dependency on single crossover that performs well over the generations.

We propose a simple hybrid recombination strategy (HRS) that assigns crossover operator based on its performance. During the initialization stage of the algorithm, each individual is randomly assigned one of the recombination operators from the recombination operator set with equal probability. The recombination operator of each individual may change depending on the performance of the individual's operator. If the recombination operator of a particular weight vector did not produce an offspring solution that surpasses the environmental selection for consecutive  $\gamma$  times, the recombination operator will be substituted by the next recombination operator in the set. In DHRS-MOEA/D, the recombination operator set consists of DE mutation operator and SBX operator.

## C. Main Algorithm

In DHRS-MOEA/D, Tchebycheff Approach [17] is used to decompose an MOP into N scalar optimization sub-problems, where N is the population size of the algorithm. For the sake of completeness, Tchebycheff approach is briefly described. In this approach, an MOP can be decomposed in the form

$$\underset{\mathbf{x}}{\operatorname{minimize}} \quad g^{te}(\mathbf{x}|\lambda, \mathbf{z}^*) = \max_{1 \le i \le m} \{\lambda_i | f_i(\mathbf{x}) - \mathbf{z}^* | \}$$
subject to  $\mathbf{x} \in \Omega$ 
(5)

where  $\lambda_i$  is the *i*-th scalar component of the weight vector  $\boldsymbol{\lambda}$ ,  $\mathbf{z}^* = (z_1^*, z_2^*, \dots, z_m^*)^T$  is the reference point in the objective space, i.e.,  $z_i^* = \min\{f_i(\mathbf{x}) | \mathbf{x} \in \Omega\}$  for each  $i = 1, \dots, m$ . DHRS-MOEA/D incorporates the selection operator [18] and HRS into the decomposition-based MOEA. In each generation, DHRS-MOEA/D maintains:

- a population consists of N individuals, each individual has a weight vector  $(\lambda^i \in [0 \ 1]^m)$ , a decision vector  $(\mathbf{x}^i)$ , a vector that records the objective vector  $(\mathbf{f}^i \in \mathbb{R}^m)$ , a set of integer, B(i), which contains the T nearest weight vector indices, a memory that record the current type of recombination operator, and an integer  $r^i$  that records the number of inferior offspring that is generated by the individual's current recombination operator;
- a vector set, *P*, that stores the reference parents of current generation;
- a vector set, *C*, that stores the reference offspring of current generation;
- an approximated ideal vector,  $\mathbf{z} \in \mathbb{R}^m$ .

The pseudo-code of DHRS-MOEA/D is shown in Algorithm 2. During initialization, each individual is randomly assigned a recombination operator from an operator set. In DHRS-MOEA/D, there are two different recombination operators, namely simulated binary crossover (SBX) and differential evolution (DE) operators. After the initialization is finished, p parents are selected from weight vector neighbourhood index set, B(i) to reproduce an offspring solution. The recombination operator for any individual in the population may change and it depends on the performance of the operator. The hybrid recombination strategy (HRS) is used to decide the operator assignment of each individual. If an individual's operator did not reproduce an offspring solution that surpasses environmental selection for consecutive  $\beta$  times, the algorithm will the next recombination operator to the individual.

During the environmental selection, the algorithm first compares the offspring solution (y) with parent solution  $(x^k)$ . If any of the parent solution is dominated by the newly generated offspring solution, the proposed algorithm will calculate the  $\Gamma^{\mathbf{p} \to \mathbf{c}}$  of replacing the parent solution with the offspring solution. This step is to check whether there is a similar offspring solution that surpasses current environmental selection. If the computed  $\Gamma^{\mathbf{p}\to\mathbf{c}}$  is lower than preset threshold,  $\gamma$ , the offspring solution will replace the parent solution. Otherwise, the parent solution surpasses the environmental selection and the newly generated solution is discarded. This characterizes one of the major differences between DHRS-MOEA/D and MOEA/D. In DHRS-MOEA/D, a non-dominated offspring solution is not necessary to replace the parent solution. Although this characteristic may raise some doubts about the algorithm's convergence at first sight, our empirical results showed that the convergence of DHRS-MOEA/D is not worse than other major MOEAs. Before the parent-offspring replacement happened, the offspring objective vector is stored in the reference offspring set and the closest parent objective vector is stored in the reference parent set. These two vectors later is used to approximate the convergence direction which is one of the most important element in the calculation of  $\Gamma^{\mathbf{p}\to\mathbf{c}}$ .

Algorithm 2 Diversity Preservation with Hybrid Recombination Strategy Multiobjective Evolutionary Algorithm (DHRS-MOEA/D)

# Input:

MOP (1) A stopping criterion

N: Population size

T: The number of the weight vectors in the neighbourhood of each weight vector

- $\beta$ : The parameter used in hybrid recombination strategy
- $\gamma$ : Maximum tolerable  $\Gamma^{\mathbf{p} \to \mathbf{c}}$

## **Output:**

Approximated Pareto Front ( $\mathbf{f}^1, \ldots, \mathbf{f}^N$ )

Approximated Pareto Solutions  $(\mathbf{x}^1, \dots, \mathbf{x}^N)$ 

# Step 1 > Initialization:

- Step 1.1 Generate a set of weight vector and assign each individual a specific weight vector. Find the T closest weight vectors (in terms of Euclidean distance) for each individual. Set B(i) = $i_1, \ldots, i_T$ , where  $\lambda^{i_1}, \ldots, \lambda^{i_T}$  are the T closest weight vectors to  $\lambda^i$ .
- Step 2.2 Generate an initial population,  $\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^N$ , by uniformly random sampling the decision space. Set  $\mathbf{f}^i = \mathbf{f}(\mathbf{x}^i)$  and randomly assign recombination operator for each individual.
- Step 3.3 Initialize z by setting  $z_k = \min_{j=1,...,N} f_k^j$  where k = 1,...,m.

**Step 2** > **Update:** Set  $P = C = \emptyset$ . For i = 1, ..., N, do

- Step 2.1 **Reproduction:** Select the appropriate recombination operator according to the HRS. If  $r_i > \beta$ , change the recombination operator, else keep the current recombination operator. Randomly select p indices from B(i) (where p is the number of parents used in recombination operator) and use the operator to produce a new solution,  $\mathbf{x}'$ . Apply polynomial mutation to the new solution to produce  $\mathbf{y}$ .
- Step 2.2 Update of z: Evaluate y to get f(y). If  $f_j(y) < z_i$  for any  $j \in \{1, ..., m\}$ , set  $z_i = f_j(y)$
- Step 2.3 Selection: Declare a set  $D = \{1, ..., N\}$ and permute the sequence in the set. For each j = 1, ..., N, set  $k = D_j$ , if  $g^{te}(\mathbf{y}|\lambda^k, \mathbf{z}) < g^{te}(\mathbf{x}^k|\lambda^k, \mathbf{z})$ , calculate  $\Gamma^{\mathbf{p} \to \mathbf{c}}$  using Algorithm 1 and set  $r^i = r^i + 1$ . If  $\Gamma^{\mathbf{p} \to \mathbf{c}} < \gamma$ , set  $r^i = 0$ ,  $\mathbf{x}^k = \mathbf{y}, \mathbf{f}^k = \mathbf{f}(\mathbf{y}), C = C \cup \{\mathbf{f}(\mathbf{y})\},$  $P = P \cup \{\mathbf{w}\}$ , where  $\mathbf{w}$  is the nearest parent objective vector to  $\mathbf{y}$ , and then directly go back Step 2.

**Step 3** > **Stopping Criterion:** If the stopping criterion is satisfied, stop the process, output  $\{\mathbf{x}^1, \ldots, \mathbf{x}^N\}$  and  $\{\mathbf{f}^1, \ldots, \mathbf{f}^N\}$ . Otherwise, go to **Step 2**.

## **IV. EXPERIMENT RESULTS**

This section presents the experiment results of DHRS-MOEA/D performance in terms of Inverted Generational Distance (IGD). Two well-known MOEA test suites, namely WFG and CEC-09, are used to evaluate the algorithm's performance. For the WFG test suite, the decision variable are set to be 24, 4 of them are position related parameters (k = 4) whereas the rest are distance related parameters (l = 24). For the CEC-09 test suite, recommended settings in [14] are used in the experiment. The POFs of the CEC-09 test suite are obtained from the CEC-09 competition website. The parameter settings of the algorithm are shown in Table I.

Table I PARAMETER SETTINGS FOR EXPERIMENTS

Parameters	Values
Population size, N	100 or 300 (2 or 3 objectives)
Total number of generation	500
Total number of fitness evaluation	$5\times 10^4$ or $1.5\times 10^5$
Neighbourhood size, $T$	20
Probability in mating selection, $n_r$	0.9
Crossover control in DE, $CR$	1.0
Differential weight in DE, F	0.5
Distribution index in SBX, $\eta_c$	20
Distribution index in mutation, $\eta_m$	20
Mutation rates, $p_m$	1/n (n: decision variables)
Maximum allowable $\Gamma^{\mathbf{p} \rightarrow \mathbf{c}}, \gamma$	20
HBS parameter, $\beta$	2

The proposed algorithm, DHRS-MOEA/D, is compared with other major MOEAs. In the comparative study, NSGA-II [4], NSDE [12], MOEA/D-DE [12] and MOEA/D-SBX [19] are compared with the proposed algorithm. To make a fair comparison, the parameters of different operators are kept the same across different algorithms. Each algorithm performs 30 independent runs for each test problem. The mean and standard deviation of the obtained IGD are recorded down. Student's t-test and Wilcoxon rank sum test are also performed to check whether the IGD values of the test problems with different algorithms are from the same normal distributions, as shown in Table II and Table III. 5% significant level is used to decide whether the null hypothesis (that two algorithm's IGD values are independent random samples from the same distribution) to be rejected or not. If the p value is less than  $5 \times 10^{-2}$ , we are at least 95% confident that the IGD values of the two algorithms are from different probability distributions. If a particular algorithm's average IGD value outperforms other algorithms in the same test problem, as well as the algorithm's IGD value's probability distribution is different with other algorithms, we conclude that the algorithm outperforms other algorithms significantly.

From Table II and III, DHRS-MOEA/D ranks between first and third in all the 19 test problems (12 first ranks, 6 second ranks and 1 third rank). Out of 12 rank one test problems, DHRS-MOEA/D performs significantly better than others on 7 of the benchmark test problems with 95% confidence level (UF2-UF4, UF9-UF10, WFG1, WFG3, WFG6-WFG8). In terms of IGD values, DHRS-MOEA/D generally performs better than others in these two test suites. Fig. 2 shows the IGD over generation for UF2, UF3, UF4 and UF9 problems. From the figure, it can be noticed that the proposed algorithm converges relatively slow at the early generations (generation number < 100). However, the proposed algorithm performs better at the later generation. Fig. 3 and Fig. 4 show the final solutions of UF3 from different algorithms over 30 independent runs. From the figure, we can observe that DHRS-MOEA/D produces well spread solutions without significantly deteriorating its convergence properties. For WFG1 problem, we can see that none of the algorithms reaches the true PF. This is because WFG1 problem is heavily biased where slight deviations from the true POS result in huge different in objective space. UF3 is a difficult MOP as it can cause diversity loss of MOEAs at the early generations. If we refer to Fig. 2(b), it can be noticed that other algorithms (except the proposed DHRS-MOEA/D) converge very fast at the early generations. After 200-th generation, most of the algorithms' IGD values reach the steady state. However, the proposed algorithm is still improving its IGD performance.

# V. CONCLUSION

This paper has proposed a new MOEA which improves the optimization performance of the MOEA/D. By controlling the diversity loss of the population, diversity property of the MOEA has been improved. Hybrid recombination strategy is used to enhance the search ability by alternating the crossover operator based on its optimization performance. From the simulation results, we can conclude that the proposed algorithm improves its optimization performance at the cost of sacrifying its convergence speed at the early generation. If the number of fitness evaluations is very limited and convergence speed is highly desired, the proposed algorithm may not be a good candidate of optimizer. If the optimization performance is the first priority, the proposed algorithm is competitive to other state-of-the-art MOEAs.

#### ACKNOWLEDGEMENT

This work was supported by the Singapore Ministry of Education Academic Research Fund Tier 1.

#### REFERENCES

- P. Bosman and D. Thierens, "The balance between proximity and diversity in multiobjective evolutionary algorithms," *Evolutionary Computation, IEEE Transactions on*, vol. 7, no. 2, pp. 174–188, Apr. 2003.
- [2] C. Fonseca and P. Fleming, "Multiobjective genetic algorithms made easy: selection sharing and mating restriction," in *Genetic Algorithms in Engineering Systems: Innovations and Applications, 1995. GALESIA. First International Conference on (Conf. Publ. No. 414).* IET, 1995, pp. 45–52.
- [3] H. Ishibuchi and T. Murata, "A multi-objective genetic local search algorithm and its application to flowshop scheduling," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions* on, vol. 28, no. 3, pp. 392–403, 1998.
- [4] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *Evolutionary Computation, IEEE Transactions on*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
- [5] S. Mahfoud, "Crowding and preselection revisited," Parallel problem solving from nature, vol. 2, pp. 27–36, 1992.

- [6] S. Kukkonen and K. Deb, "Improved pruning of non-dominated solutions based on crowding distance for bi-objective optimization problems," in *Proceedings of the World Congress on Computational Intelligence (WCCI-2006)(IEEE Press). Vancouver, Canada.* Proceedings of the World Congress on Computational Intelligence (WCCI-2006)(IEEE Press). Vancouver, Canada, 2006, pp. 1179–1186.
- [7] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, "Combining convergence and diversity in evolutionary multiobjective optimization," *Evolutionary Computation*, vol. 10, no. 3, pp. 263–282, 2002.
- [8] H. Lu and G. Yen, "Rank-density based multiobjective genetic algorithm," in *Evolutionary Computation*, 2002. CEC'02. Proceedings of the 2002 Congress on, vol. 1. IEEE, 2002, pp. 944–949.
- [9] O. Schütze, M. Laumanns, E. Tantar, C. A. C. Coello, and E.-G. Talbi, "Computing gap free pareto front approximations with stochastic search algorithms," *Evolutionary Computation*, vol. 18, no. 1, pp. 65–96, 2010.
- [10] G. Rudolph, H. Trautmann, S. Sengupta, and O. Schütze, "Evenly spaced pareto front approximations for tricriteria problems based on triangulation," in *Evolutionary Multi-Criterion Optimization*. Springer, 2013, pp. 443–458.
- [11] L. Tang and X. Wang, "A hybrid multi-objective evolutionary algorithm for multi-objective optimization problems," *Evolutionary Computation, IEEE Transactions on*, 2013.
- [12] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II," *Evolutionary Computation*, *IEEE Transactions on*, vol. 13, no. 2, pp. 284–302, Apr. 2009.
- [13] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *Evolutionary Computation, IEEE Transactions on*, vol. 10, no. 5, pp. 477–506, Oct. 2006.
- [14] Q. Zhang, A. Zhou, S. Zhao, P. Suganthan, W. Liu, and S. Tiwari, "Multiobjective optimization test instances for the CEC 2009 special session and competition," University of Essex, Colchester, UK and Nanyang Technological University, Singapore, Tech. Rep., 2008.
- [15] K. Deb, Multi-Objective Optimization using Evolutionary Algorithms, ser. Wiley-Interscience Series in Systems and Optimization. John Wiley & Sons, Chichester, 2001.
- [16] C. Coello, G. Lamont, and D. Van Veldhuizen, Evolutionary algorithms for solving multi-objective problems. Springer, 2007, vol. 5.
- [17] K. Miettinen, Nonlinear multiobjective optimization. Springer, 1999, vol. 12.
- [18] S. B. Gee, X. Qiu, and K. C. Tan, "A novel diversity maintenance scheme for evolutionary multi-objective optimization," in *Intelligent Data Engineering and Automated Learning–IDEAL 2013.* Springer, 2013, pp. 270–277.
- [19] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *Evolutionary Computation, IEEE Transactions* on, vol. 11, no. 6, pp. 712–731, Dec. 2007.



Figure 2. Plot of average IGD over generation in (a) UF2 (b) UF3 (c) UF4 and (d) UF9 benchmark problem.



Figure 3. Plots of the approximated Pareto front of UF3 with 30 independent runs by using (a) DHRS-MOEA/D (b) MOEA/D-DE (c) MOEA/D-SBX (d) NSDE (e) NSGA-II



Figure 4. Plots of the approximated Pareto front of WFG1 with 30 independent runs by using (a) DHRS-MOEA/D (b) MOEA/D-DE (c) MOEA/D-SBX (d) NSDE (e) NSGA-II

Table II		
STATISTICS OF THE IGD VALUES (CEC-09	Test	SUITE)

		UF1	UF2	UF3	UF4	UF5	UF6	UF7	UF8	UF9	UF10
MOEA/D	Average	0.156953	0.064201	0.306635	0.056051	0.439415	0.437451	0.353306	0.147992	0.133993	0.293651
-SBX	Std. Dev.	0.065312	0.031131	0.029931	0.003418	0.084132	0.150794	0.155061	0.035788	0.062404	0.130362
	Rank	5	5	5	3	3	4	5	3	3	2
	p(t-test)	4.31E-14	9.04E-09	1.78E-41	4.37E-22	3.95E-01	8.92E-06	6.24E-13	3.56E-07	1.62E-06	2.20E-02
	$H_0$ (t-test)	Reject	Reject	Reject	Reject	Accept	Reject	Reject	Reject	Reject	Reject
	p (r. sum)	3.18E-11	1.04E-10	2.87E-11	2.87E-11	2.68E-01	9.85E-06	1.02E-09	3.39E-07	8.02E-08	8.64E-02
	$H_0$ (r. sum)	Reject	Reject	Reject	Reject	Accept	Reject	Reject	Reject	Reject	Accept
MOEA/D	Average	0.047532	0.042727	0.151459	0.08661	0.766799	0.438578	0.101792	0.091078	0.106537	0.582637
-DE	Std. Dev.	0.037332	0.031711	0.068806	0.010382	0.132768	0.220314	0.164595	0.012398	0.04522	0.071635
	Rank	2	2	2	5	4	5	3	1	2	4
	p (t-test)	1.73E-01	5.28E-03	7.06E-09	4.74E-29	5.75E-16	2.37E-04	1.09E-01	6.20E-06	1.59E-04	1.16E-25
	$H_0$ (t-test)	Accept	Reject	Reject	Reject	Reject	Reject	Accept	Reject	Reject	Reject
	p (r. sum)	9.06E-01	9.33E-02	2.89E-07	2.87E-11	5.32E-10	1.37E-04	4.93E-02	1.13E-05	2.08E-06	7.03E-11
	$H_0$ (r. sum)	Accept	Accept	Reject							
NSGA-II	Average	0.123068	0.048146	0.218053	0.053257	0.329483	0.230239	0.235744	0.219417	0.163559	0.323611
	Std. Dev.	0.03187	0.012489	0.066681	0.001766	0.092297	0.06805	0.144534	0.009773	0.049139	0.070307
	Rank	4	4	4	2	1	1	4	5	4	3
	p (t-test)	1.82E-20	2.59E-13	1.31E-17	1.06E-27	3.31E-03	5.86E-01	1.47E-07	1.19E-41	3.91E-13	6.82E-06
	$H_0$ (t-test)	Reject	Reject	Reject	Reject	Reject	Accept	Reject	Reject	Reject	Reject
	p (r. sum)	2.87E-11	2.87E-11	2.87E-11	2.87E-11	2.96E-03	5.25E-01	1.48E-09	2.87E-11	5.32E-10	1.20E-07
	$H_0$ (r. sum)	Reject	Reject	Reject	Reject	Reject	Accept	Reject	Reject	Reject	Reject
NSDE	Average	0.060237	0.042996	0.15138	0.072304	0.848596	0.418027	0.038879	0.151971	0.193795	2.430789
	Std. Dev.	0.016159	0.004701	0.027065	0.007766	0.171164	0.08179	0.042124	0.030039	0.064635	0.184853
	Rank	3	3	3	4	5	3	1	4	5	5
	p (t-test)	8.78E-09	5.86E-23	2.17E-20	8.85E-26	6.75E-17	1.16E-06	7.68E-01	8.21E-10	5.67E-14	6.61E-54
	$H_0$ (t-test)	Reject	Reject	Reject	Reject	Reject	Reject	Accept	Reject	Reject	Reject
	p (r. sum)	4.78E-09	3.51E-11	7.76E-11	2.87E-11	1.04E-10	2.28E-07	2.60E-04	2.49E-10	3.64E-10	2.87E-11
	$H_0$ (r. sum)	Reject									
DHRS-	Average	0.037849	0.02578	0.062131	0.046211	0.409578	0.246909	0.044692	0.107821	0.067304	0.228139
MOEA/D	Std. Dev.	0.008519	0.00352	0.02174	0.00078	0.116556	0.151995	0.098667	0.013652	0.027963	0.078988
	Rank	1	1	1	1	2	2	2	2	1	1
-											

Table III STATISTICS OF THE IGD VALUES (WFG TEST SUITE)

		WFG1	WFG2	WFG3	WFG4	WFG5	WFG6	WFG7	WFG8	WFG9
MOEA/D	Average	1.048133	0.187786	0.020332	0.016714	0.069122	0.082187	0.020597	0.127121	0.060725
-SBX	Std. Dev.	0.045804	0.064425	0.005855	0.001559	0.000568	0.023867	0.011149	0.00975	0.038031
	Rank	2	5	2	1	1	3	4	2	3
	p (t-test)	5.65E-23	1.97E-08	1.24E-05	5.38E-11	6.20E-01	1.06E-07	1.36E-02	2.25E-11	2.55E-01
	$H_0$ (t-test)	Reject	Reject	Reject	Reject	Accept	Reject	Reject	Reject	Accept
	p (r. sum)	5.32E-10	4.29E-11	1.86E-10	3.21E-08	3.39E-02	2.71E-08	3.59E-01	7.04E-10	9.67E-03
	$H_0$ (r. sum)	Reject	Reject	Reject	Reject	Reject	Reject	Accept	Reject	Reject
MOEA/D	Average	1.163333	0.167235	0.020445	0.081227	0.069269	0.107269	0.019025	0.127221	0.059763
-DE	Std. Dev.	0.013848	0.0883	0.001762	0.008085	0.000296	0.031888	0.001118	0.012852	0.028653
	Rank	4	4	3	4	3	4	3	3	2
	p (t-test)	5.74E-61	3.49E-05	8.30E-23	1.96E-44	1.52E-01	5.72E-12	6.38E-25	1.32E-08	2.39E-01
	$H_0$ (t-test)	Reject	Reject	Reject	Reject	Accept	Reject	Reject	Reject	Accept
	p (r. sum)	2.87E-11	3.34E-09	2.87E-11	2.87E-11	2.23E-01	1.62E-09	2.87E-11	7.39E-08	6.73E-04
	$H_0$ (r. sum)	Reject	Reject	Reject	Reject	Accept	Reject	Reject	Reject	Reject
NSGA-II	Average	1.078783	0.161065	0.021071	0.018874	0.070581	0.064023	0.016965	0.137202	0.08435
	Std. Dev.	0.081178	0.027787	0.001555	0.001081	0.000457	0.006819	0.000974	0.006502	0.052919
	Rank	3	3	4	2	4	2	2	4	4
	p (t-test)	3.26E-16	4.49E-08	8.94E-28	5.95E-03	2.95E-22	7.97E-05	3.91E-12	4.41E-25	5.10E-03
	$H_0$ (t-test)	Reject								
	p (r. sum)	8.12E-09	2.66E-02	2.87E-11	1.63E-02	2.87E-11	1.20E-07	7.76E-11	2.87E-11	5.84E-02
	$H_0$ (r. sum)	Reject	Accept							
NSDE	Average	1.218112	0.046113	0.034366	0.092772	0.075478	0.107921	0.030578	0.141382	0.110565
	Std. Dev.	0.005217	0.025469	0.001689	0.003727	0.001746	0.034865	0.00186	0.010109	0.037973
	Rank	5	1	5	5	5	5	5	5	5
	p (t-test)	9.05E-76	3.48E-02	3.67E-54	1.83E-66	2.55E-27	2.89E-11	1.63E-46	1.52E-21	9.65E-08
	$H_0$ (t-test)	Reject								
	p (r. sum)	2.87E-11	2.19E-02	2.87E-11	2.87E-11	2.87E-11	1.26E-08	2.87E-11	3.18E-11	2.29E-08
	$H_0$ (r. sum)	Reject								
DHRS-	Average	0.910717	0.075047	0.015212	0.019849	0.069128	0.045605	0.015382	0.110458	0.048994
MOEA/D	Std. Dev.	0.010472	0.068955	0.000382	0.001522	0.00022	0.022731	0.000209	0.005046	0.040303
	Rank	1	2	1	3	2	1	1	1	1