This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Adaptive Replacement Strategies for MOEA/D

Zhenkun Wang, Qingfu Zhang, *Senior Member, IEEE*, Aimin Zhou, *Member, IEEE*, Maoguo Gong, *Senior Member, IEEE*, and Licheng Jiao, *Senior Member, IEEE*

***Abstract*—Multiobjective evolutionary algorithms based on decomposition (MOEA/D) decompose a multiobjective optimization problem into a set of simple optimization subproblems and solve them in a collaborative manner. A replacement scheme, which assigns a new solution to a subproblem, plays a key role in balancing diversity and convergence in MOEA/D. This paper proposes a global replacement scheme which assigns a new solution to its most suitable subproblems. We demonstrate that the replacement neighborhood size is critical for population diversity and convergence, and develop an approach for adjusting this size dynamically. A steady-state algorithm and a generational one with this approach have been designed and experimentally studied. The experimental results on a number of test problems have shown that the proposed algorithms have some advantages.**

***Index Terms*—Adaptive scheme, decomposition, multiobjective optimization, replacement.**

I. Introduction

A MULTIOBJECTIVE optimization problem (MOP) has several objectives to be optimized. An MOP can be mathematically formulated as follows:

$$\begin{aligned} &\text{minimize} && F(x) = (f_1(x), \ldots, f_m(x)) \\ &\text{subject to} && x \in \Omega \end{aligned} \tag{1}$$

where $x = (x_1, \ldots, x_n) \in R^n$ is a decision variable vector, Ω is the feasible region of the search space, and $F : \Omega \rightarrow R^m$ consists of m objective functions $f_1(x), \ldots, f_m(x)$. If Ω is a closed and connected region in R^n and all the objectives are continuous of x, this problem can be called a continuous MOP.

Let $x, y \in \Omega$, x is said to dominate y, denoted by $x \prec y$, if and only if $f_i(x) \leq f_i(y)$ for all $i \in \{1, \ldots, m\}$, and $F(x) \neq F(y)$. A point $x^* \in \Omega$ is called Pareto optimal if no other $x \in \Omega$ dominates x^*. The set of all the Pareto optimal points is called the Pareto set (PS) and the Pareto front (PF) is defined as $PF = \{F(x)|x \in PS\}$ [1].

Since the objectives in (1) usually conflict with each other, no single solution can optimize all objectives at the same time. In practice, Pareto optimal solutions are the best tradeoff candidates among different objectives. Most multiobjective evolutionary algorithms (MOEAs) aim at finding a finite number of solutions to approximate the PF [2]–[4]. Selection plays a key role in balancing the population diversity and convergence in MOEAs [5]–[7]. Based on their selection strategies, most existing MOEAs can be roughly classified into the following three categories: 1) Pareto domination-based approach in which the selection operation is mainly based on the Pareto dominance relations among solutions in a population [8]–[11]; 2) indicator-based approach which uses a quality indicator to guide their selection [12]–[16]; and 3) decomposition-based approach which decomposes an MOP into a number of simple subproblems and optimizes them in a collaborative manner [17]–[21].

This paper studies multiobjective evolutionary algorithms based on decomposition (MOEA/D), particularly its replacement. The MOEA/D framework used in this paper optimizes a number of single objective subproblems in a collaborative manner. Each single subproblem i has its current solution x^i in the population. A neighborhood relationship among these subproblems is defined in MOEA/D, and a corresponding neighborhood relationship can then be established for all the current solutions. Each solution has a replacement neighborhood. At each generation in most existing MOEA/D variants, for each current solution x^i to subproblem i, a new solution x^i_{new} is generated by reproduction operators on some of its neighboring solutions, and then x^i_{new} compares with each current solution in the neighborhood of x^i and replaces it if x^i_{new} is better in terms of optimizing the related subproblem. However, the neighboring subproblems of i may not always be suitable for x^i_{new}. To avoid the mismatch between solutions and subproblems, we propose a global replacement (GR) scheme for MOEA/D. This scheme searches for the most suitable subproblem j among all the subproblems for x^i_{new}, and the replacement is conducted for the current solutions of neighbors of subproblem j. This paper demonstrates that the replacement neighborhood size is critical for controlling the balance between population diversity and convergence. To ensure population diversity at the early search stage and make the search focus on exploitation at its late search stage, we propose an adaptive version of GR scheme. This adaptive scheme has been used in a steady-state MOEA/D variant and a

Manuscript received July 29, 2014; revised November 13, 2014 and January 27, 2015; accepted February 8, 2015. This work was supported in part by the Cheung Kong Scholars Programme of China under Grant K5051302050, and in part by the National Natural Science Foundation of China under Grant 61473241, Grant 61273317, Grant 61273313 and Grant 61175063. This paper was recommended by Associate Editor P. N. Suganthan.

Z. Wang, M. Gong, and L. Jiao are with the Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education of China, Xidian University, Xi'an 710071, China.

Q. Zhang is with the Department of Computer Science, City University of Hong Kong, Hong Kong, and also with the School of Computer Science and Electronic Engineering, University of Essex, Colchester CO4 3SQ, U.K. (e-mail: qingfu.zhang@cityu.edu.hk; qzhang@essex.ac.uk).

A. Zhou is with the Shanghai Key Laboratory of Multidimensional Information Processing, East China Normal University, Shanghai 200241, China, and also with the Department of Computer Science and Technology, East China Normal University, Shanghai 200241, China.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2015.2403849

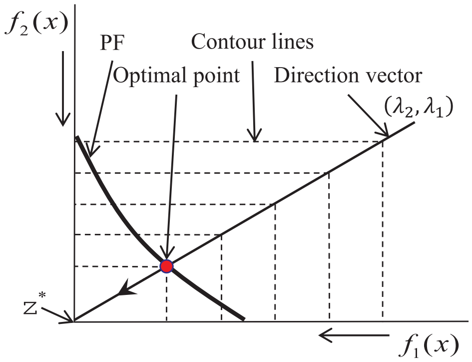


Fig. 1. Illustration of the Tchebycheff decomposition approach.

generational one. Comprehensive and systematic experiments on some test problems have been carried out to study the proposed schemes.

The rest of this paper is organized as follows. MOEA/D is briefly reviewed in Section II. Section III presents the GR scheme. In Section IV, experimental studies have been conducted to show that different MOPs need different replacement neighborhood sizes in the GR scheme. An adaptive GR (AGR) scheme is proposed in Section V, a steady-state variant of MOEA/D and a generational one with this adaptive scheme are also given in this section. Experiments and discussion are presented in Section VI. Finally, this paper is concluded in Section VII.

II. MOEA/D

This section introduces decomposition methods and a neighborhood concept, which are two basic components in MOEA/D [17], [18], and then presents a basic MOEA/D framework and some improvements on it.

A. Decomposition

In MOEA/D, an MOP is decomposed into N subproblems. Each subproblem can be a single optimization problem or a multiobjective one. The set of the optimal solutions of these subproblems approximate the PF [17]–[19]. In this paper, we only consider single objective decomposition. Many decomposition approaches have been used and investigated in MOEA/D frameworks [17], [22], [23]. As shown in Fig. 1, the Tchebycheff approach, used in this paper, defines a subproblem as follows:

$$\begin{aligned} \text{minimize} \quad & g\left(x|\lambda, z^*\right) = \max_{1\le i\le m} \left\{\lambda_i \left|f_i(x) - z_i^*\right|\right\} \\ \text{subject to} \quad & x \in \Omega \end{aligned} \tag{2}$$

where $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m)$ is the weight vector of (2), i.e., $\lambda_i \geq 0$, for all $i = 1, 2, \ldots, m$ and $\sum_{i=1}^{m} \lambda_i = 1$. z^* is a utopian objective vector, i.e., $z_i^* < \min\{f_i(x)|x \in \Omega\}$, for each $i = 1, 2, \ldots, m$. Moreover, $((1/\lambda_1)/(\sum_{i=1}^{m} 1/\lambda_i), (1/\lambda_2)/(\sum_{i=1}^{m} 1/\lambda_i), \ldots, (1/\lambda_m)/(\sum_{i=1}^{m} 1/\lambda_i))$ is called the direction vector of subproblem (2).

To decompose an MOP by the Tchebycheff approach, N weight vectors should be selected for defining N subproblems. Following [19] and [23]–[26], we set the weight vectors in a way such that their corresponding direction vectors are uniformly distributed in the m-dimensional unity simplex. It has been proven that with this setting, the optimal solutions of the N subproblems are uniformly distributed on the PF if the PF is the unity simplex. Therefore, this setting is appropriate when the shape of the PF is similar to an unity simplex.

B. Neighborhood

Since every subproblem in the above decomposition is associated with a direction vector and its objective function is continuous of the direction vector, one can assume that two subproblems have similar optimal solutions if their direction vectors are close. To make use of this assumption, MOEA/D introduces a neighborhood concept [17]. Given a subproblem, its T-neighborhood is the set of all the subproblems with the T closest direction vectors to its direction vector in terms of Euclidean distance. For simplicity, we always let a subproblem be one of its own neighbors [17].

Two different neighborhoods are often used in MOEA/D algorithms [27], [28]. One is the mating neighborhood and the other is the replacement neighborhood. The former is for selecting parent solutions and the latter is used for determining which old solutions should be replaced by a new one.

C. Framework

MOEA/D is an iterative algorithm. At each generation (i.e., iteration), it maintains N solutions $x^1, \ldots, x^N$, where x^i is the current solution to subproblem i. For convenience, we assume that each subproblem is a minimization problem and the objective function of subproblem i is $g^i(x)$. Since each solution in the current population is associated with a different subproblem, we can also define the neighborhood structure of these solutions. The T-neighborhood of solution x^i consists of the solutions of the subproblems in T-neighborhood of subproblem i. A solution in the T-neighborhood of solution x^i is called a T-neighbor of x^i.

A basic steady-state MOEA/D framework works as follows.

1) Determine the mating neighborhood size T_m and the replacement neighborhood size T_r.
2) For each subproblem i, the following occurs.
    a) *Mating:* Conduct reproduction operators on some solutions selected from T_m-neighborhood of x^i to generate a new solution x^i_{new}.
    b) *Replacement:* For each current solution x^j in the T_r-neighborhood of x^i, replace x^j by x^i_{new} if x^i_{new} is better than x^j in terms of $g^j(x)$.

Convergence and diversity have been recognized as two major goals in the design of MOEAs. Ideally, the population should move toward the PF rapidly and maintain good diversity during its evolution [29], [30]. A balance between these two goals should be considered [31]. If an algorithm spends too much effort on convergence, its population may lose diversity at the early search stage. As a result, it would be trapped at local PFs and/or unable to approximate some parts of the PF [32]. On the other hand, too much effort on diversity will slow down the search [33].

The effects of T_m on the performance of MOEA/D largely depends on the distribution of current solutions and reproduction operators. Extensive investigations have been made in [18], [28], and [34] to examine the effects of T_m on the convergence and diversity. This paper mainly consider the effects of T_r on the performance of MOEA/D. The larger the value of T_r is, the more exploitation and less exploration the algorithm will do. A small T_r value is good for exploration and population diversity since a solution may have very few duplicates in the next generation. Ideally, an algorithm should do more exploration at its early stage, since there is no good confidence that the most promising region identified so far contains solutions of high quality. To make the best use of computational resources, it should focus on exploitation at its late stage. Some attempts have been made to improve convergence and diversity in MOEA/D. For example, in MOEA/D-DE [18], the neighborhood size can be set to the population size with a small probability and only a few old solutions are allowed to be replaced by the same new solution. Zhao *et al.* [35] proposed an ensemble learning method for using several different neighborhood sizes in MOEA/D. In the next section, we will present a new replacement approach for enhancing both convergence and diversity.

III. GR Scheme

In the steady-state MOEA/D framework introduced in Section II, the new solution x^i_{new} is only allowed to replace T_r-neighbors of x^i. It can save computational cost since only objective function comparisons with these T_r neighboring solutions are required. In MOEA/D, each solution is associated with a subproblem. We call a subproblem suitable for a solution x if its optimal solution is close to x. It is desirable that each solution is associated with a suitable subproblem. However, the replacement scheme introduced in the above section may discard x^i_{new}, when x^i_{new} is poor for all the T_r neighbors of subproblem i but good for other subproblems. Even if x^i_{new} is not discarded, it is possible that x^i_{new} is not associated with its most suitable subproblem. Mismatch between subproblems and solutions would discourage both diversity and convergence.

To overcome this shortcoming, we propose the following GR scheme. For each new solution x^i_{new}, it works as follows.

1) Find subproblem j, the most suitable subproblem for x^i_{new}.
2) For each current solution x^k in the T_r neighborhood of x^j, replace x^k by x^i_{new} if x^i_{new} is better than x^k in terms of $g^k(x)$.

There are many ways to define the most suitable problem for a given solution. A good definition should promote both diversity and convergence. For simplicity, in this paper, subproblem j is defined as its most suitable subproblem of a given solution x if

$$j = \arg\min_{1\le k\le N}\left\{g^k(x)\right\}. \tag{3}$$

Now, we can make some remarks on the GR scheme.

1) In the GR scheme, x^i_{new} is not used for replacing the neighboring solutions of x^i. Instead, it is to replace the neighboring solutions of x^j, where x^j is the current solution of the most suitable subproblem for x^i_{new}. Therefore, the GR scheme can guarantee that each solution is associated with a suitable subproblem. As a result, mismatching will be reduced, parent solutions selected for reproduction operators in MOEA/D will have good chance to produce good child solutions, and diversity can also be promoted.
2) For simplicity, this GR scheme does not limit the number of solutions replaced by a new solution. It is possible that all the T_r neighboring solutions are replaced by the same new solution. Thus, T_r is a key parameter for controlling the balance of convergence and diversity. A large T_r will allow a good solution to have many copies in the next population, and thus promotes convergence but decreases diversity. On the other hand, a small T_r will be good for diversity but not convergence.

IV. Effects of T_r in GR Scheme

T_r is the only control parameter in the GR scheme. To study how to set it properly, this section experimentally investigates its effect on the algorithm performance. We implement an algorithm, called MOEA/D-GR, which is the same as MOEA/D-DE proposed in [18] except that it uses the GR scheme as its replacement method. Like MOEA/D-DE, MOEA/D-GR selects parent solutions from the whole population instead of the T_m neighborhood with probability $(1-\delta)$.

A. Experimental Settings

We have tested MOEA/D-GR on ZDT and DTLZ problems F1–F9 [18], MOP1–MOP7 [19], [36], [37]. The characteristics of these problems are listed in Table I.

The following is the parameter settings in our experimental study.

1) *The Population Size N:* The population size is set to be 100 for the ZDT problems, 300 for DTLZ problems, F1–F5, F7–F9, and MOP1–MOP5, and 595 for F6, MOP6, and MOP7.
2) *Decomposition:* The Tchebycheff approach is used. The setting of weight vectors is the same as in MOEA/D-DE. The z^*_i value is estimated from the previous search as done in MOEA/D-DE.
3) *Stopping Condition:* The maximal number of function evaluations is set to be 25 000 for ZDT problems, 100 000 for DTLZ problems, 150 000 for F1–F5 and F7–F9, 300 000 for MOP1–MOP5 and F6, and 900 000 for MOP6 and MOP7.
4) *Control Parameters in Reproduction Operators:* As recommended in [18], we set $\text{CR} = 1.0$ and $F = 0.5$ for the DE operator [38] and $\eta = 20$ and $p_{dm} = 1/n$ for the polynomial mutation [39].
5) T_m: $T_m = 0.1 \times N$ as in [18].
6) *Other Control Parameters:* The probability for the T_m neighboring solutions to be a mating neighborhood: $\delta = 0.8$.
7) Number of Independent Runs for Each Test Problem: 30.

TABLE I
TEST INSTANCES

Name	n	m	Range	Inter-variable dependencies	Characteristics
ZDT1	30	2	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Linear	Convex, Unimodal
ZDT2	30	2	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Linear	Nonconvex, Unimodal
ZDT3	30	2	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Linear	Convex, Disconnected, Multimodal
ZDT4	10	2	$x_1 \in [0,1], x_i \in [-5,5],\ 2 \leqslant i \leq n$	Linear	Nonconvex, Multimodal
ZDT6	10	2	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Linear	Nonconvex, Nonuniformly, Multimodal
DTLZ1	10	3	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Linear	Nonconvex, Multimodal
DTLZ2	10	3	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Linear	Nonconvex, Unimodal
F1	30	2	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Nonlinear	Convex, Unimodal
F2	30	2	$x_1 \in [0,1], x_i \in [-1,1],\ 2 \leqslant i \leq n$	Nonlinear	Convex, Multimodal
F3	30	2	$x_1 \in [0,1], x_i \in [-1,1],\ 2 \leqslant i \leq n$	Nonlinear	Convex, Multimodal
F4	30	2	$x_1 \in [0,1], x_i \in [-1,1],\ 2 \leqslant i \leq n$	Nonlinear	Convex, Multimodal
F5	30	2	$x_1 \in [0,1], x_i \in [-1,1],\ 2 \leqslant i \leq n$	Nonlinear	Convex, Multimodal
F6	10	3	$x_1, x_2 \in [0,1],\ x_i \in [-2,2],\ 3 \leqslant i \leq n$	Nonlinear	Nonconvex, Multimodal
F7	10	2	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Nonlinear	Convex, Multimodal
F8	10	2	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Nonlinear	Convex, Multimodal
F9	30	2	$x_1 \in [0,1], x_i \in [-1,1],\ 2 \leqslant i \leq n$	Nonlinear	Nonconvex, Multimodal
MOP1	10	2	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Nonlinear	Convex, Multimodal
MOP2	10	2	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Nonlinear	Nonconvex, Multimodal
MOP3	10	2	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Nonlinear	Nonconvex, Multimodal
MOP4	10	2	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Nonlinear	Convex, Disconnected, Multimodal
MOP5	10	2	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Nonlinear	Convex, Multimodal
MOP6	10	3	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Nonlinear	Nonconvex, Multimodal
MOP7	10	3	$x_i \in [0,1],\ 1 \leqslant i \leq n$	Nonlinear	Nonconvex, Multimodal

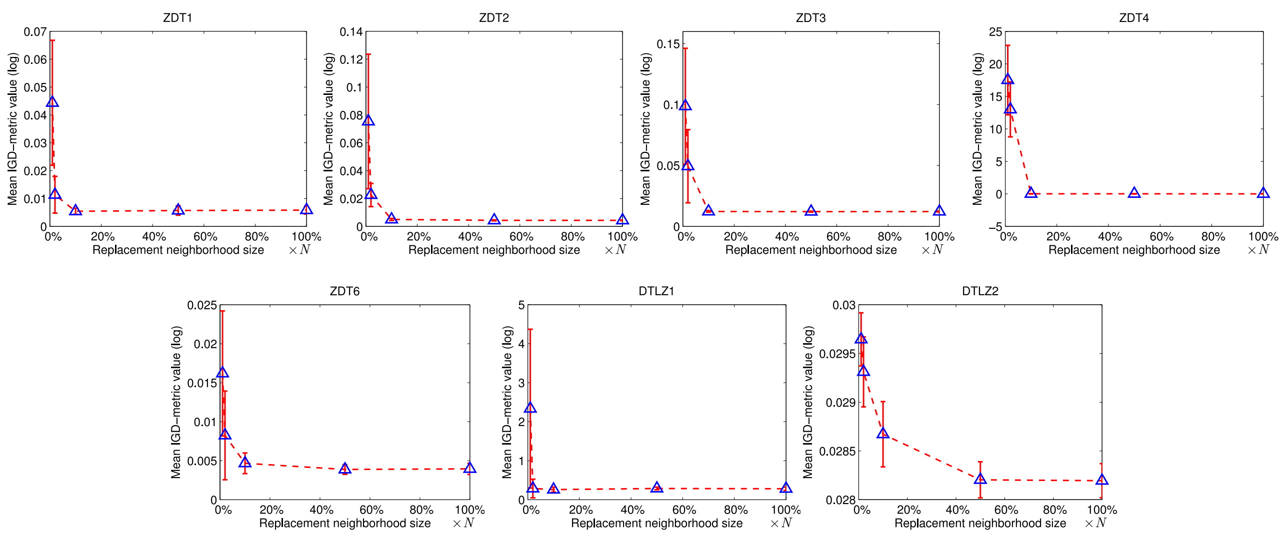


Fig. 2. Mean IGD-metric values of MOEA/D-GR with different replacement neighborhood sizes on ZDT and DTLZ problems.

We have tested five different values of T_r: the integers were rounded up from $\{1\%, 2\%, 10\%, 50\%, 100\%\} \times N$.

B. Performance Metric

The inverted generational distance (IGD) [40], [41] is adopted to assess the algorithm performance. Let P^* be a set of uniformly distributed Pareto optimal points along the PF in the objective space. Let P be an approximate set to the PF obtained by an algorithm. The IGD from P^* to P is defined as

$$\mathrm{IGD}\big(P^*, P\big) = \frac{\sum_{v \in P^*} d(v, P)}{|P^*|} \tag{4}$$

where $d(v, P)$ is the minimum Euclidean distance between v and the points in P. If $|P^*|$ is large enough, $\mathrm{IGD}(P^*, P)$ could measure both convergence and diversity of P. We select 500 evenly distributed points in the PF and let these points be P^* for each test problem with two objectives, and 1000 points for each test problem with three objectives.

C. Results and Discussion

The average IGD values of MOEA/D-GR in 30 independent runs on these test problems are given in Figs. 2–4. From Fig. 2, it can be observed that better results can be achieved when T_r is larger than 10%N on ZDT and DTLZ problems. This is probably due to the fact that the PS shapes of these problems are lines or hyperplanes. Once several different Pareto optimal solutions have been generated, it is easy for the algorithm to obtain other Pareto optimal solutions by genetic operators. Therefore, the algorithm should spend more effort on convergence. This is why a large T_r leads to good performance on ZDT and DTLZ problems.

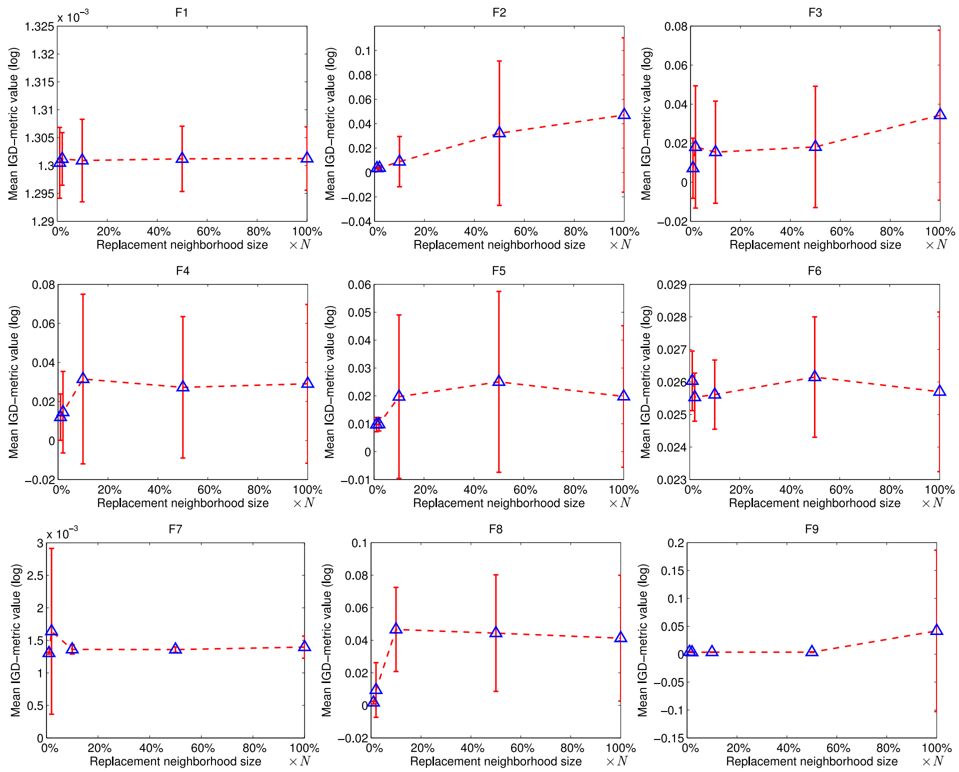


Fig. 3. Mean IGD-metric values of MOEA/D-GR with different replacement neighborhood sizes on F1–F9.

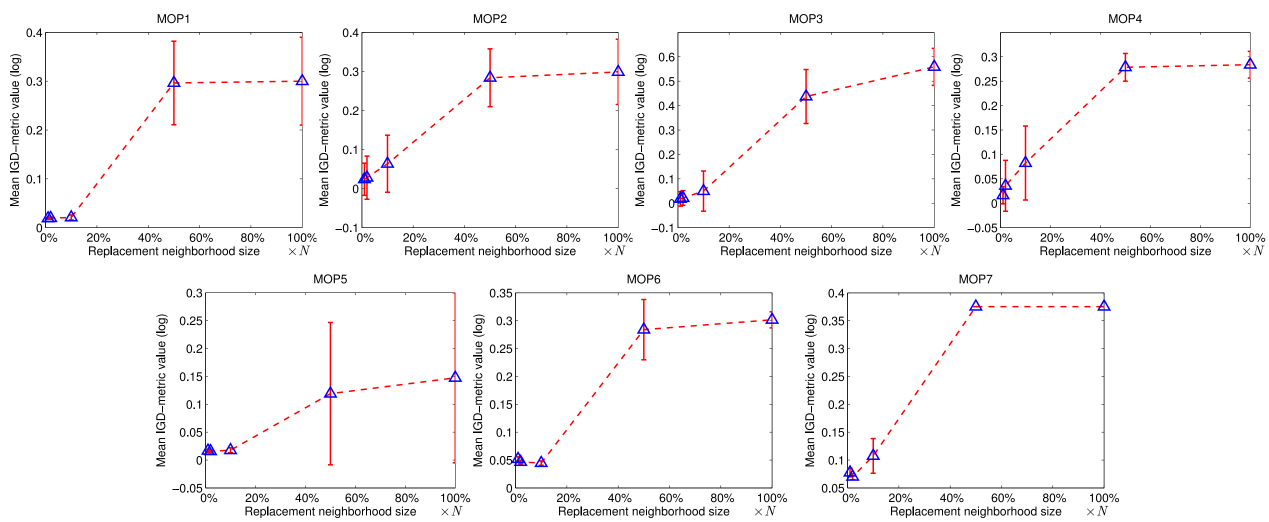
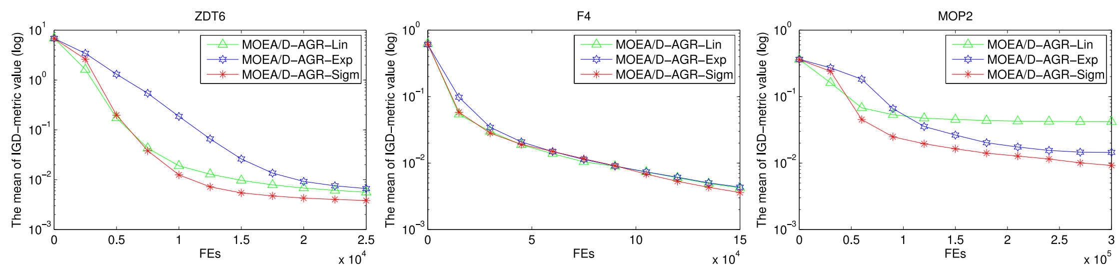


Fig. 4. Mean IGD-metric values of MOEA/D-GR with different replacement neighborhood sizes on MOP1–MOP7.

It is clear from Fig. 3 that a large T_r does not perform well on F2–F5 and F8. A reason could be that these problems have complicated PS shapes, and it is difficult to obtain a good approximation to the PS using genetic operators on very few Pareto optimal solutions. Therefore, population diversity is important to solve these problems. For this reason, large T_r is not good for these problems. It can also be observed from these figures that the effect of T_r is problem dependent. T_r has little impact on algorithm performances on F1 and F7 due to the simple landscapes of these two problems. On F2–F5 and F9, a small T_r (smaller than $10\%N$) is needed for good performances since their PS shapes are more complicated and it is necessary to maintain good population diversity by using small T_r values. F8 has many local PSs, a smaller replacement neighborhood works well. On F6 with three objectives, too large or too small T_r cannot produce good results.

Fig. 4 shows that only a very small T_r can produce acceptable results for MOP1–MOP7. It is because search on most subproblems are easily trapped at locally optimal solutions if a large T_r is used, and a small T_r can promote population diversity and drive the search out of locally optimal solutions.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



(a) (b) (c)

Fig. 5. Evolution of the mean IGD-metric values of MOEA/D-AGR with three different adaptive strategies during the evolutionary process. (a) ZDT6. (b) F4. (c) MOP2.

V. AGR Scheme

As discussed in Section IV, a small T_r value promotes diversity, whereas a large one encourages convergence. Different problems require different T_r values. Thus, one should spend much effort in carefully setting this value if a fixed T_r value is used during the whole search process [42]. To ease this burden, a dynamic adaptation strategies, AGR is proposed in this section. We implement a steady-state MOEA/D which uses AGR as the replacement scheme. We call it MOEA/D-AGR. We also implement a generational MOEA/D with AGR (gMOEA/D-AGR).

A. AGR

As discussed before, T_r should be small at the early search stage to maintain good population diversity, and be large at the late stage to improve the convergence speed. We propose to increase the value of T_r during the search. We consider the following three different adaptive schemes:

$$\text{Linear} : T_r = \left\lceil \frac{k \times T_{\max}}{K} \right\rceil \tag{5}$$

$$\text{Exponential} : T_r = \left\lceil \frac{\left(\exp\left(\frac{5\times k}{K}\right) - 1\right) \times T_{\max}}{\exp(5) - 1} \right\rceil \tag{6}$$

$$\text{Sigmoid} : T_r = \left\lceil \frac{T_{\max}}{1 + \exp\left(-20 \times \left(\frac{k}{K} - \gamma\right)\right)} \right\rceil \tag{7}$$

where $\lceil . \rceil$ is the ceiling function, $T_{\max}$ is the maximal value for T_r, k is the current generation number, K is the maximal generation number and $\gamma \in [0, 1)$ is a control parameter to determine how T_r increases as the search goes. Fig. 6 illustrates how T_r increases in these three strategies.

In the following, we propose two versions of MOEA/D with the above AGR scheme.

B. MOEA/D-AGR

MOEA/D-AGR is the same as MOEA/D-GR except that it use the AGR scheme to adjust T_r. Like MOEA/D-GR, it is a steady-state algorithm.

MOEA/D-AGR with the three different AGR strategies are denoted as MOEA/D-AGR-Lin, MOEA/D-AGR-Exp, and MOEA/D-AGR-Sigm, respectively. To determine which strategy is more effective, we compare them on ZDT4, F4, and MOP2. $T_{\max}$ is set to be $0.4 \times N$ and $\gamma = 0.25$. Fig. 5 shows the evolution of the mean IGD values of the population obtained by the three algorithms versus the number of function evaluations in 30 independent runs. On MOP2, MOEA/D-AGR-Lin is the worst one. On ZDT6, MOEA/D-AGR-Exp is the worst. On F4, three strategies perform very similarly. Overall, MOEA/D-AGR-Sigm is more stable and efficient than others. Therefore, Sigmoid AGR strategy (7) is used in this paper.

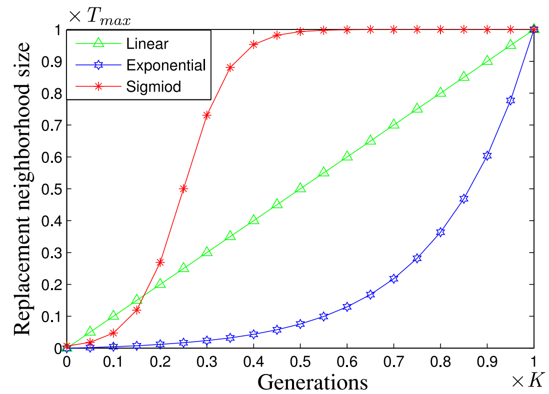


Fig. 6. T_r versus the iterative generations.

C. gMOEA/D-AGR

MOEA/D-AGR is a steady-state algorithm, and at most T_r solutions can be updated at each time. It is not suitable for parallel computing. In the following, we propose a generational version of MOEA/D with AGR (gMOEA/D-AGR). This algorithm maintains one solution x^i for each subproblem i. Its mating neighborhood size T_m is fixed during the search as in MOEA/D-GR and T_r is dynamically adjusted according to (7).

At each generation, it works as follows.

1) Compute the replacement neighborhood size T_r by using (7).
2) *Mating:* For each subproblem i, do:
   a) conduct reproduction operators on some solutions selected from T_m-neighborhood of x^i to generate a new solution.
3) *Replacement:* For each subproblem i, do:
   a) find the T_r closest solutions to the direction vector of subproblem i among the N current solutions and the N new solutions;

TABLE II
RESULT COMPARISONS ON ZDT AND DTLZ PROBLEMS

Problem	IGD	NSGA-II	SMS-EMOA	MOEA/D-DE	ENS-MOEA/D	MOEA/D-M2M	MOEA/D-GR	MOEA/D-AGR	gMOEA/D-AGR
ZDT1	Mean	4.61E-03	**3.61E-03**	1.27E-02	1.35E-02	7.77E-03	5.43E-03	5.37E-03	1.08E-02
	Std	1.47E-04	**2.02E-05**	7.10E-03	4.06E-03	3.01E-03	1.09E-03	1.55E-03	5.43E-03
	Rank	2≈	1+	7−	8−	5−	4≈	3	6−
ZDT2	Mean	2.49E-02	2.45E-02	1.49E-02	5.86E-03	7.26E-03	4.97E-03	**4.93E-03**	6.29E-03
	Std	1.10E-01	1.10E-01	3.79E-03	1.05E-03	3.01E-03	7.06E-04	**6.81E-04**	1.62E-03
	Rank	8≈	7−	6−	3−	5−	2≈	1	4−
ZDT3	Mean	9.57E-03	**4.43E-03**	2.71E-02	1.36E-02	1.35E-02	1.23E-02	1.22E-02	1.25E-02
	Std	1.04E-02	**4.52E-05**	1.32E-02	2.46E-03	2.07E-03	5.71E-04	5.08E-04	5.08E-04
	Rank	2+	1+	8−	7−	6−	4≈	3	5−
ZDT4	Mean	3.46E-02	4.87E-01	3.15E-01	1.44E-02	6.05E-01	1.76E-02	1.10E-02	**7.95E-03**
	Std	1.53E-01	3.98E-01	2.25E-01	9.67E-03	2.61E-01	2.75E-02	4.97E-03	**2.78E-03**
	Rank	5−	7−	6−	3≈	8−	4≈	2	1+
ZDT6	Mean	4.41E-03	**3.01E-03**	1.32E-02	6.08E-03	3.18E-03	4.67E-03	3.82E-03	5.01E-03
	Std	1.65E-04	**1.42E-05**	7.46E-03	1.99E-03	8.57E-05	1.32E-03	6.43E-04	1.72E-03
	Rank	4−	1+	8−	7−	2+	5−	3	6−
DTLZ1	Mean	3.26E-01	2.87E-01	4.85E-01	2.99E-01	1.62	**2.59E-01**	2.66E-01	2.87E-01
	Std	**1.00E-02**	2.24E-01	5.42E-01	3.03E-02	8.51E-01	6.38E-02	1.04E-01	1.23E-01
	Rank	6−	4−	7−	5−	8−	1≈	2	3≈
DTLZ2	Mean	3.97E-02	7.07E-01	2.87E-02	3.73E-02	4.33E-02	2.87E-02	**2.84E-02**	**2.84E-02**
	Std	1.26E-03	8.13E-01	2.75E-04	1.67E-04	3.35E-03	3.35E-04	**1.69E-04**	2.39E-04
	Rank	6−	8−	3−	5−	7−	3−	1	1≈
Total		33	29	45	38	41	23	15	26
Final Rank		5	4	8	6	7	2	1	3

$+$,$-$ and $\approx$ denote that the performance of the corresponding algorithm is significantly better than, worse than, and similar to MOEA/D-AGR respectively by Wilcoxon's rank sum test with $\alpha = 0.05$.

b) replace x^i by the solution with the smallest g^i function value among all the T_r solutions found in the above step.

The above algorithm can generate N new solutions and do its replacement in a parallel way.

VI. EXPERIMENTAL STUDIES

A. Test Problems

Twenty-three test instances are used in our experimental study, and their characteristics are given in Table I.

B. Performance Metric

The IGD indicator is used to assess the algorithm performance.

C. Algorithms in Comparison and Parameter Settings

We compare our three proposed algorithms, MOEA/D-GR, MOEA/D-AGR, and gMOEA/D-AGR with NSGA-II [9], SMS-EMOA [15], MOEA/D-DE [18], ENS-MOEA/D [35], and MOEA/D-M2M [19]. The three proposed algorithms can be regarded as a modification of MOEA/D-DE. NSGA-II is a commonly used Pareto dominance-based MOEA and SMS-EMOA is a well-known indicator-based MOEA. ENS-MOEA/D is another variant of MOEA/D with an ensemble of neighborhood sizes. MOEA/D-M2M decomposes an MOP into a number of multiobjective subproblems and emphasizes population diversity more than convergence. The parameter settings of these algorithms are the same as in [9], [15], [18], [19], and [35]. The detailed parameter settings are summarized as follows.

1) *The Population Size N:* It is set to be 100 for the ZDT problems, 300 for DTLZ problems, F1–F5, F7–F9, and MOP1–MOP5, and 595 for F6 and MOP6-MOP7.
2) *Number of Runs and Stopping Condition:* Each algorithm is run 30 times independently for each test instance. The algorithms stop after a given number of function evaluations. The maximal number of function evaluations is set to be 25 000 for ZDT problems, 100 000 for DTLZ problems, 150 000 for F1–F5 and F7–F9, 300 000 for MOP1–MOP5 and F6, and 900 000 for MOP6 and MOP7.
3) *Control Parameters in Reproduction Operators:* As recommended in [18], we set CR $= 1.0$ and $F = 0.5$ for the DE operator [38] and $\eta = 20$ and $p_{dm} = 1/n$ for the polynomial mutation [39].
4) *The Neighborhood Sizes:* In MOEA/D-DE and MOEA/D-GR, $T_r = T_m = 0.1 \times N$. In MOEA/D-AGR and gMOEA/D-AGR, $T_m = 0.1 \times N$ and T_r is set by (7) with $T_{\max} = 0.4 \times N$ and $\gamma = 0.25$.
5) *Other Control Parameters:* $\delta = 0.8$ for all the variants of MOEA/D algorithms. $n_r = 2$ for MOEA/D-DE.

D. Experimental Results on ZDT and DTLZ Problems

ZDT and DTLZ problems have linear PS shapes. The comparison results are presented in Table II. It is clear that SMS-EMOA, NSGA-II, and our three proposed algorithms (i.e., MOEA/D-GR, MOEA/D-AGR, and gMOEA/D-AGR) perform better than MOEA/D-DE, ENS-MOEA/D, and MOEA/D-M2M. MOEA/D-AGR performs best on ZDT2 and DTLZ2, two unimodal problems with concave PFs. It is the second best on ZDT4 and DTLZ1 and third best on the other problems. MOEA/D-GR performs best on DTLZ1, and gMOEA/D-AGR is the best on DTLZ1 and the multimodal problem ZDT4. SMS-EMOA can achieve the best results on ZDT1, ZDT3, and ZDT6, but its running time is very long and does not work well on other problems.

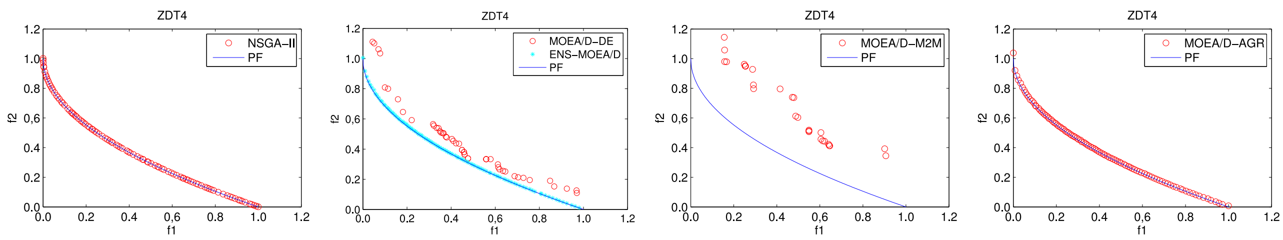


Fig. 7. Plots of the final solutions with the lowest IGD-metric values found by NSGA-II, MOEA/D-DE, ENS-MOEA/D, MOEA/D-M2M, and MOEA/D-AGR in 30 runs in the objective space on ZDT4.

TABLE III
RESULT COMPARISONS ON F1–F9

Problem	IGD	NSGA-II	SMS-EMOA	MOEA/D-DE	ENS-MOEA/D	MOEA/D-M2M	MOEA/D-GR	MOEA/D-AGR	gMOEA/D-AGR
	Mean	2.88E-03	2.64E-03	1.32E-03	1.38E-03	1.50E-03	**1.30E-03**	**1.30E-03**	1.31E-03
F1	Std	6.83E-05	1.49E-04	6.94E-06	8.34E-06	5.20E-05	7.41E-06	7.33E-06	**5.95E-06**
	Rank	8−	7−	4−	5−	6−	1≈	1	3≈
	Mean	7.67E-02	5.90E-02	3.72E-03	8.37E-03	1.70E-02	8.91E-03	**3.15E-03**	3.22E-03
F2	Std	1.54E-02	1.51E-02	3.56E-04	2.13E-02	5.58E-03	2.06E-02	2.87E-04	**1.83E-04**
	Rank	8−	7−	3−	4−	6−	5−	1	2−
	Mean	3.32E-02	2.94E-02	4.26E-03	3.79E-02	5.11E-03	1.54E-02	**3.42E-03**	3.46E-03
F3	Std	3.80E-03	3.29E-03	2.16E-03	4.98E-02	1.46E-03	2.62E-02	**1.16E-03**	1.23E-03
	Rank	7−	6−	3−	8−	4−	5−	1	2≈
	Mean	3.17E-02	4.08E-02	6.70E-03	3.51E-02	8.87E-03	3.15E-02	**3.60E-03**	4.65E-03
F4	Std	1.05E-02	7.28E-03	6.14E-03	3.97E-02	2.05E-03	4.34E-02	**1.06E-03**	1.54E-03
	Rank	6−	8−	3−	7−	4−	5−	1	2−
	Mean	2.62E-02	2.09E-02	1.01E-02	1.69E-02	**6.70E-03**	1.97E-02	9.47E-03	9.27E-03
F5	Std	4.66E-03	1.79E-03	2.00E-03	1.76E-02	**9.60E-04**	2.93E-02	2.51E-03	1.91E-03
	Rank	8−	7−	4≈	5−	1+	6−	3	2≈
	Mean	6.82E-02	1.25E-01	2.54E-02	2.92E-02	3.23E-02	2.56E-02	2.43E-02	**2.42E-02**
F6	Std	7.44E-03	4.76E-02	1.08E-03	2.08E-03	1.36E-03	1.06E-03	6.70E-04	**4.34E-04**
	Rank	7−	8−	3−	5−	6−	4−	2	1≈
	Mean	1.51E-01	3.51E-01	1.31E-03	1.50E-03	2.57E-03	1.36E-03	**1.30E-03**	1.45E-03
F7	Std	2.91E-02	9.82E-02	**1.16E-05**	2.25E-04	3.33E-04	7.25E-05	1.22E-05	5.24E-04
	Rank	7−	8−	2≈	5−	6−	3−	1	4−
	Mean	1.50E-01	3.17E-01	1.75E-03	2.50E-02	6.61E-02	4.66E-02	**1.61E-03**	2.93E-03
F8	Std	2.46E-02	1.09E-01	**8.24E-04**	2.72E-02	2.57E-02	2.58E-02	9.37E-04	4.77E-03
	Rank	7−	8−	2−	3−	6−	5−	1	4−
	Mean	8.99E-02	8.17E-02	4.16E-03	4.96E-03	1.78E-02	3.67E-03	**3.28E-03**	3.80E-03
F9	Std	1.95E-02	1.02E-02	7.29E-04	1.42E-03	7.16E-03	4.50E-04	**3.75E-04**	1.01E-03
	Rank	8−	7−	4−	5−	6−	2−	1	3−
Total		66	66	28	47	45	36	12	23
Final Rank		7	7	3	6	5	4	1	2

$+$,$-$ and $\approx$ denote that the performance of the corresponding algorithm is significantly better than, worse than, and similar to MOEA/D-AGR respectively by Wilcoxon's rank sum test with $\alpha = 0.05$.

Overall, MOEA/D-AGR is the best on ZDT and DTLZ problems, which is confirmed by the Wilcoxon's rank sum test.

Fig. 7 shows that the distribution of the final solutions with the lowest IGD-metric values found by different algorithms in all the 30 runs on ZDT4. Since NSGA-II and SMS-EMOA have very similar performance, we have not plotted the final solutions obtained by SMS-EMOA. Fig. 7 shows that MOEA/D-DE and MOEA/D-M2M cannot approximate the whole PF very well.

E. Experimental Results on F1–F9

F1–F9 [18] have complicated PS shapes. The comparison results on these problems are shown in Table III. It is evident that MOEA/D-AGR significantly outperforms other seven algorithms. It yields the best mean IGD values on all the problems except F5 and F6. F5 has a very complicated PS shape and no algorithm can approximate its PF very well, and F6 has three objectives. It should be pointed out that the performance of MOEA/D-AGR is still acceptable on F5. MOEA/D-AGR is only slightly worse than gMOEA/D-AGR on F6. Although they yield the best result on F5 and F1 respectively, MOEA/D-M2M and MOEA/D-GR do not work so well on other problems.

MOEA/D-DE is the worst on ZDT and DTLZ problems. Table III, however, shows that MOEA/D-DE can produce acceptable results on F1–F9. NSGA-II works well on ZDT and DTLZ problems, but it is poor on F1–F9. MOEA/D-AGR is the best among the eight algorithms in terms of the average rank on F1–F9. The statistics by the Wilcoxon's rank sum tests also indicate that MOEA/D-AGR is significantly better than other seven algorithms. gMOEA/D-AGR also works well.

Fig. 8 plots the final solutions with the lowest IGD-metric values obtained by different algorithms among 30 runs on F8 in the decision and objective spaces respectively. MOEA/D-DE and ENS-MOEA/D have about the same performance on F8, we have not plotted the results of ENS-MOEA/D. It can be seen from these figures

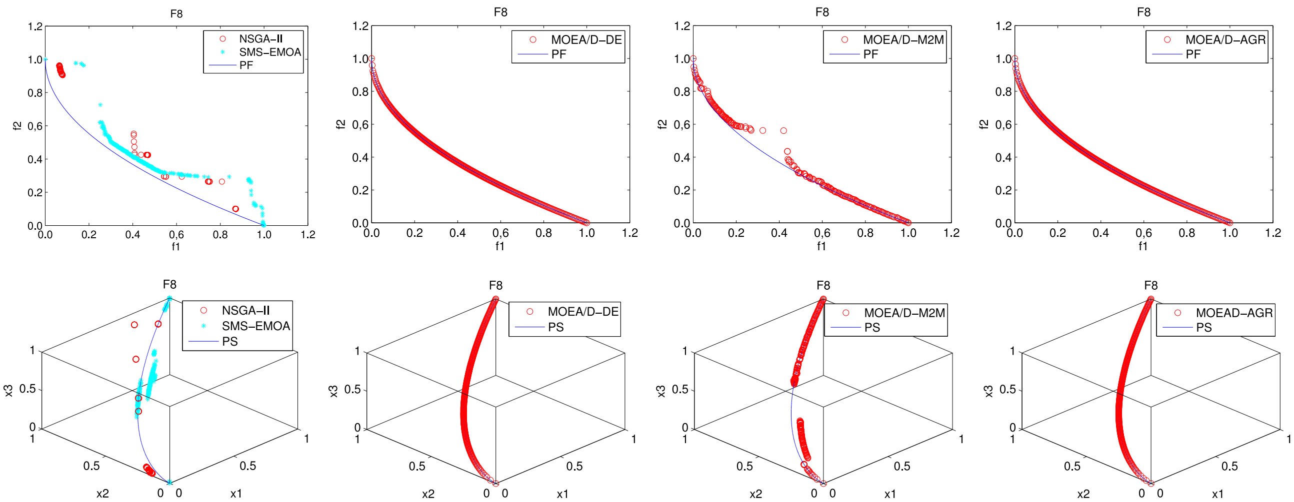


Fig. 8. Plots of the final solutions with the lowest IGD-metric values found by NSGA-II, SMS-EMOA, MOEA/D-DE, MOEA/D-M2M, and MOEA/D-AGR in 30 runs in the objective space (the top row) and the $x_1 - x_2 - x_3$ space (the bottom row) on F8.

TABLE IV
RESULT COMPARISONS ON MOP1–MOP7

Problem	IGD	NSGA-II	SMS-EMOA	MOEA/D-DE	ENS-MOEA/D	MOEA/D-M2M	MOEA/D-GR	MOEA/D-AGR	gMOEA/D-AGR
MOP1	Mean	3.55E-01	2.84E-01	3.13E-01	4.41E-01	2.65E-02	2.10E-02	2.14E-02	**2.07E-02**
	Std	5.82E-03	5.09E-02	7.65E-02	1.13E-01	**1.27E-03**	5.07E-03	4.39E-03	3.97E-03
	Rank	7−	5−	6−	8−	4−	2≈	3	1≈
MOP2	Mean	3.47E-01	3.54E-01	3.06E-01	3.11E-01	**9.87E-03**	6.37E-02	4.42E-02	5.52E-02
	Std	1.55E-02	**5.65E-17**	6.55E-02	6.39E-02	3.77E-03	7.30E-02	7.56E-02	6.91E-02
	Rank	7−	8−	5−	6−	1≈	4≈	2	3≈
MOP3	Mean	5.66E-01	4.08E-01	5.57E-01	5.67E-01	2.01E-02	4.97E-02	**1.63E-02**	4.87E-02
	Std	7.80E-02	**2.26E-16**	4.96E-02	5.62E-02	**1.03E-02**	8.21E-02	2.19E-02	7.99E-02
	Rank	6−	8−	5−	7−	2−	4≈	1	3≈
MOP4	Mean	2.51E-01	2.49E-01	2.83E-01	2.82E-01	**3.87E-03**	8.24E-02	1.64E-02	3.88E-02
	Std	1.34E-02	2.42E-02	2.34E-02	2.63E-02	**3.59E-04**	7.58E-02	1.36E-02	4.07E-02
	Rank	6−	5−	8−	7−	1+	4−	2	3≈
MOP5	Mean	1.91E-01	1.71E-01	3.14E-01	3.16E-01	2.33E-02	1.77E-02	**1.65E-02**	1.68E-02
	Std	1.13E-03	4.74E-02	7.62E-03	1.11E-02	**7.29E-04**	4.41E-03	2.99E-03	2.35E-03
	Rank	6−	5−	7−	8−	4−	3≈	1	2≈
MOP6	Mean	3.04E-01	3.04E-01	3.00E-01	3.00E-01	5.20E-02	4.46E-02	4.16E-02	**4.14E-02**
	Std	3.12E-06	**1.46E-06**	1.74E-02	2.25E-02	2.72E-03	3.91E-03	2.56E-03	2.40E-03
	Rank	7−	7−	5−	5−	4−	3−	2	1≈
MOP7	Mean	2.64E-01	3.75E-01	3.70E-01	3.69E-01	**8.18E-02**	1.07E-01	1.45E-01	1.40E-01
	Std	2.39E-05	**2.10E-06**	1.46E-02	2.28E-02	5.27E-03	3.11E-02	1.28E-02	2.19E-02
	Rank	5−	8−	7−	6−	1+	2+	4	3≈
Total		44	46	43	47	17	22	15	16
Final Rank		6	7	5	8	3	4	1	2

+ ,− and ≈ denote that the performance of the corresponding algorithm is significantly better than, worse than, and similar to MOEA/D-AGR respectively by Wilcoxon's rank sum test with $\alpha = 0.05$.

that NSGA-II and SMS-EMOA can only find several solutions close to the PF. MOEA/D-M2M also misses some part of the PF. The solutions obtained by both MOEA/D-DE and MOEA/D-AGR can approximate the PS and PF well. However, MOEA/D-AGR still performs remarkably better than MOEA/D-DE based on the Wilcoxon's rank sum tests in Table III.

F. Experimental Results on MOP1–MOP7

On MOP1–MOP7, some Pareto optimal solutions are easy to obtain and some others are extremely difficult to find. It has been argued in [19] that population diversity has to be prioritized for dealing with these problems. Experimental results in Table IV show that, in terms of the IGD values, MOEA/D-M2M, MOEA/D-GR, MOEA/D-AGR, and gMOEA/D-AGR perform much better than NSGA-II, SMS-EMOA, MOEA/D-DE, and ENS-MOEA/D. MOEA/D-M2M performs best on MOP2, MOP4, and MOP7 while MOEA/D-AGR performs best on MOP3 and MOP5, and gMOEA/D-AGR on MOP1 and MOP6. However, there is no significant difference among MOEA/D-AGR, gMOEA/D-AGR, and MOEA/D-M2M based on the Wilcoxon's rank sum test.

Fig. 9 plots the final solutions with the lowest IGD-metric values obtained by different algorithms in all the 30 runs on MOP1. Obviously, NSGA-II and MOEA/D-DE can only find a small part of the PF, SMS-EMOA, and ENS-MOEA/D also miss most of the PF. It can be observed that MOEA/D-AGR can approximate the PF quite well. MOEA/D-M2M cannot approximate the middle part of the PF well.

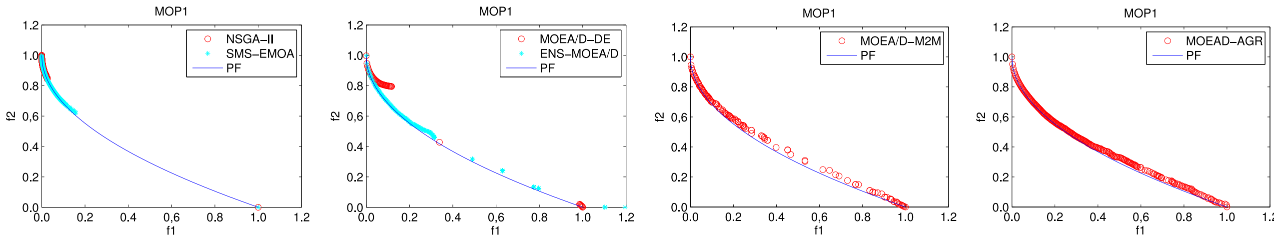
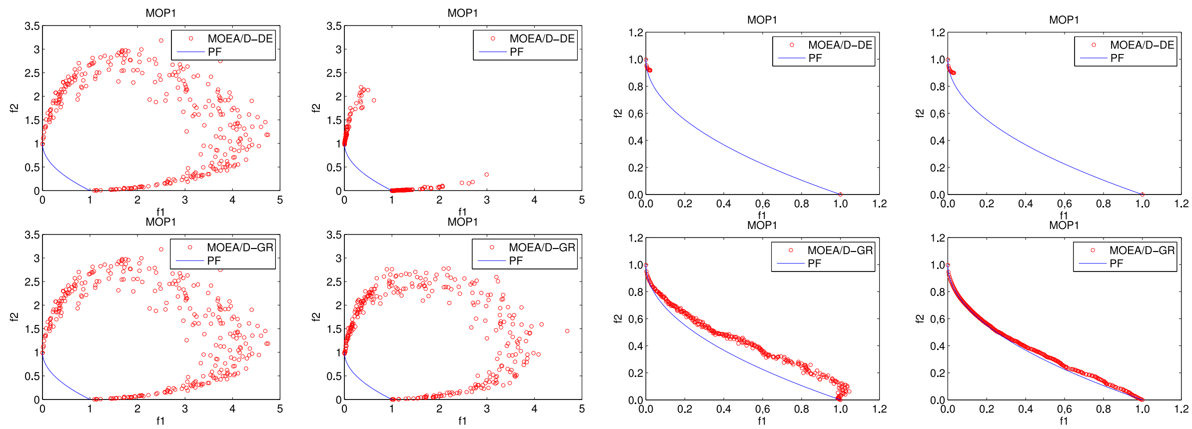


Fig. 9. Plots of the final solutions with the lowest IGD-metric values found by NSGA-II, SMS-EMOA, MOEA/D-DE, ENS-MOEA/D, MOEA/D-M2M, and MOEA/D-AGR in 30 runs in the objective space on MOP1.



(a) (b) (c) (d)

Fig. 10. Plot the solutions found by MOEA/D-DE (top) and MOEA/D-GR (bottom) in the objective space with different function evaluations on MOP1. (a) FEs = 0. (b) FEs = 900. (c) FEs = 60 000. (d) FEs = 300 000.

G. Benefits of GR and AGR Schemes

To study the benefits of the GR and AGR schemes, we have compared MOEA/D-DE with MOEA/D-GR and MOEA/D-AGR. The experimental results in terms of IGD-metric values are given in Table V.

It is evident from Table V that MOEA/D-AGR significantly outperforms other two algorithms. It yields the best results on 19 out of 23 test problems. MOEA/D-AGR is only slightly worse than MOEA/D-GR on ZDT4, DTLZ1, and MOP3. It is also shown in Table V that MOEA/D-DE performs the worst. MOEA/D-GR is remarkably better than MOEA/D-DE on MOP1–MOP7 and all the ZDT and DTLZ problems. Even though the performance of MOEA/D-GR is poorer than MOEA/D-DE in terms of the mean IGD-metric values on some of F1–F9 problems, the best results (i.e., minimal IGD-metric values) obtained by MOEA/D-GR in all the 30 runs are better than those obtained by MOEA/D-DE on almost all the problems.

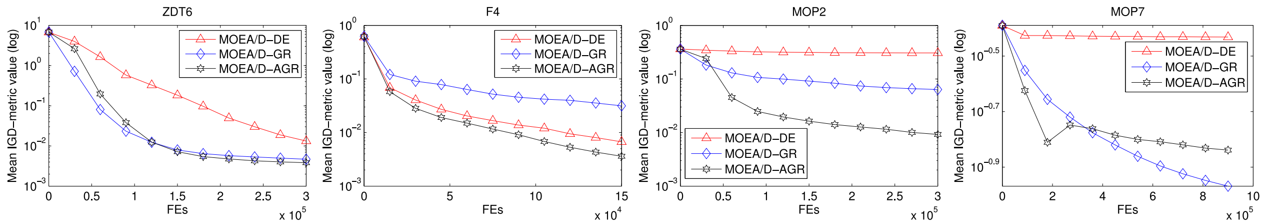
In order to demonstrate the advantages of the GR scheme, we have compared MOEA/D-DE and MOEA/D-GR with the same initial population on MOP1. The distribution of the solutions found by two algorithms with different function evaluations (i.e., 0, 900, 60 000, and 300 000) are shown in Fig. 10. We can observe that MOEA/D-DE is trapped at local PFs due to the loss of population diversity at its very early search stage. However, MOEA/D-GR can maintain the population diversity well and then get a better approximation of the PF. Since the only difference between two algorithms is their replacement operator, we can claim that the GR scheme is more effective on problems like MOP1.

TABLE V
MEDIAN AND MIN IGD-METRIC VALUES OF SOLUTIONS FOUND BY MOEA/D-DE, MOEA/D-GR, AND MOEA/D-AGR

IGD	MOEA/D-DE		MOEA/D-GR		MOEA/D-AGR	
Problem	median	min	median	min	median	min
ZDT1	0.0098	0.0053	0.0050	0.0043	**0.0049**	**0.0041**
ZDT2	0.0145	0.0060	0.0049	0.0042	**0.0048**	**0.0040**
ZDT3	0.0226	0.0135	**0.0122**	**0.0110**	**0.0122**	0.0114
ZDT4	0.2429	0.0660	**0.0090**	**0.0053**	0.0100	0.0054
ZDT6	0.0115	0.0054	0.0041	**0.0032**	**0.0036**	**0.0032**
DTLZ1	0.3012	0.0690	**0.2567**	0.0447	0.2612	**0.0429**
DTLZ2	0.0288	0.0282	0.0286	0.0282	**0.0284**	**0.0280**
F1	**0.0013**	**0.0013**	**0.0013**	**0.0013**	**0.0013**	**0.0013**
F2	0.0037	0.0032	0.0033	**0.0028**	**0.0031**	**0.0028**
F3	0.0036	0.0030	0.0045	0.0027	**0.0029**	**0.0025**
F4	0.0053	0.0036	0.0065	0.0029	**0.0033**	**0.0027**
F5	0.0097	0.0068	0.0107	0.0063	**0.0092**	**0.0061**
F6	0.0252	0.0240	0.0253	0.0240	**0.0242**	**0.0232**
F7	**0.0013**	**0.0013**	**0.0013**	**0.0013**	**0.0013**	**0.0013**
F8	**0.0014**	**0.0013**	0.0412	0.0019	**0.0014**	**0.0013**
F9	0.0040	0.0032	0.0037	0.0029	**0.0032**	**0.0027**
MOP1	0.3519	0.1593	**0.0199**	**0.0170**	**0.0199**	0.0171
MOP2	0.3543	0.1741	0.0161	0.0033	**0.0072**	**0.0028**
MOP3	0.5429	0.4774	**0.0079**	**0.0043**	0.0115	0.0048
MOP4	0.2939	0.2434	0.0822	0.0083	**0.0136**	**0.0019**
MOP5	0.3165	0.2936	0.0167	0.0135	**0.0157**	**0.0133**
MOP6	0.3041	0.2253	0.0438	0.0384	**0.0418**	**0.0357**
MOP7	0.3752	0.3225	**0.0947**	**0.0798**	0.1450	0.1140

MOEA/D-AGR can achieve good results on almost all the test problems except MOP7. Fig. 11 shows the evolution of the

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



(a) (b) (c) (d)

Fig. 11. Evolution of the mean IGD-metric values of MOEA/D-DE, MOEA/D-GR, and MOEA/D-AGR during the evolutionary process. (a) ZDT6. (b) F4. (c) MOP2. (d) MOP7.

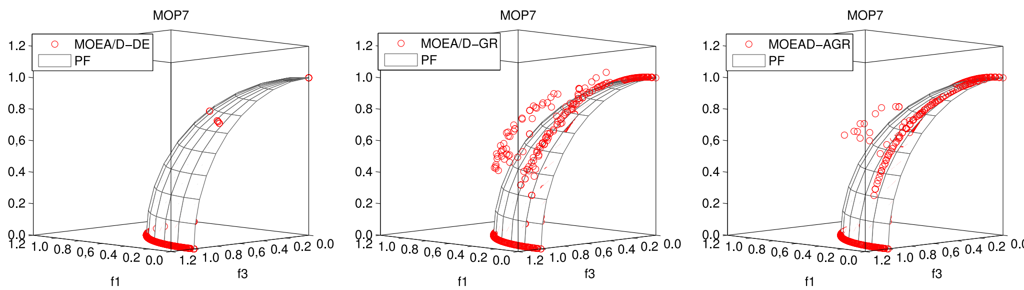


Fig. 12. Plot of the final solutions with the lowest IGD-metric values found by MOEA/D-DE, MOEA/D-GR, and MOEA/D-AGR in 30 runs in the objective space on MOP7.

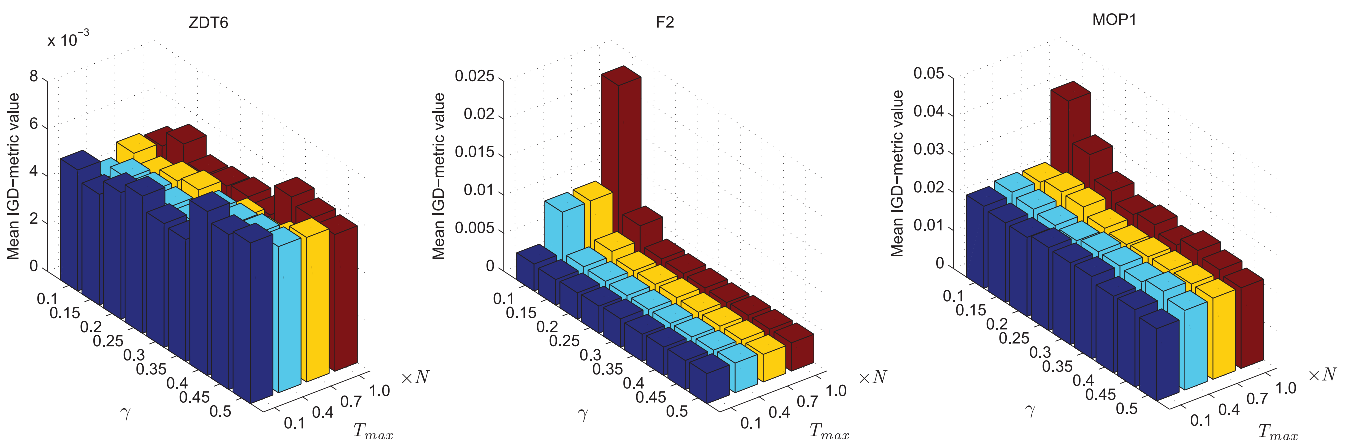


Fig. 13. Mean IGD-metric values found by MOEA/D-AGR with 36 different combinations of $T_{\max}$ and γ on ZDT6, F2, and MOP1.

mean IGD values of the population obtained by the three algorithms versus the number of function evaluations on ZDT6, F4, MOP2, and MOP7. It is clear that MOEA/D-AGR is more effective and efficient in reducing the IGD values. At its early stage, MOEA/D-AGR has relatively slow convergence speed since it maintains good population diversity and it converges fast at its late stage when the T_r values are large. However, the IGD-metric value increases rapidly in some generations in MOEA/D-AGR as shown in Fig. 11(d). This is because the solutions obtained by algorithms for MOP7 are still far away from the PF in these generations, and the decrease in population diversity leads to low IGD-metric values. Fig. 12 plots the distribution of the final solutions with the lowest IGD-metric values obtained by the three algorithms in all the 30 runs on MOP7. It confirms that no algorithm can obtain solutions close to the PF within the given number of function evaluations. In addition, MOEA/D-DE can only find several solutions close to the PF, MOEA/D-AGR also miss some parts of the PF and only MOEA/D-GR performs reasonably well.

H. Impacts of Parameter Settings

$T_{\max}$ and γ are two control parameters in AGR for adjusting T_r. $T_{\max}$ is a upper bound of T_r while γ determines how T_r increases during the search. To investigate the impacts of $T_{\max}$ and γ on the performance of MOEA/D-AGR, we have considered four values for $T_{\max}$: the integer parts of 0.1, 0.4, 0.7, and N, and nine values for γ : 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, and 0.5. All 36 different combinations of these values are examined on ZDT6, F2, and MOP1. The other parameter settings are the same as in Section VI-C. Thirty independent runs have been conducted for each combination on each test problem. Fig. 13 shows the mean IGD-metric values found by MOEA/D-AGR with these 36 different combinations of $T_{\max}$ and γ. These figures show that a large $T_{\max}$ and a small γ are good for ZDT6 while a small $T_{\max}$ and a large γ are good for F2 and MOP1. It is also evident that MOEA/D-AGR is less sensitive to the settings of $T_{\max}$ and γ except for some extreme combinations (e.g., too large a value of $T_{\max}$ and too small a value of γ). These figures also confirm that our settings of $T_{\max}$ and γ are reasonable by considering both the computational overhead and algorithm performance on all three sets of problems.

VII. Conclusion

Replacement is a key component in MOEA/D algorithms. Existing MOEA/D algorithms may mismatch solutions and subproblems in their replacement and then deteriorate the algorithm performance. To overcome this shortcoming, we have proposed a GR scheme which can assign a new solution to its most suitable subproblems. We have shown that the replacement neighborhood size can be used for adjusting the allocation of search effort on population diversity and convergence. An approach has been proposed to dynamically adjust the replacement neighborhood size. A steady-state MOEA/D variant and a generational one with this approach have been experimentally studied. These algorithms spend much effort on maintaining population diversity at their early search stage and on convergence at their late search stage. Our experimental results have shown that the proposed algorithms outperform some other algorithms on a number of test problems.

In the future, we will study how other components in MOEA/D affect its population diversity and convergence and then study how to improve them. Another related future research topic is how to qualitatively measure population diversity and convergence in MOEAs, which should be very important for applying machine learning methods in MOEAs.

References

[1] K. Miettinen, *Nonlinear Multiobjective Optimization*. Boston, MA, USA: Kluwer, 1999.

[2] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. Chichester, U.K.: Wiley, 2001.

[3] K. C. Tan, E. F. Khor, and T. H. Lee, *Multiobjective Evolutionary Algorithms and Applications: Algorithms and Applications*. London, U.K.: Springer, 2006.

[4] C. A. C. Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*. New York, NY, USA: Springer, 2007.

[5] P. A. Bosman and D. Thierens, "The balance between proximity and diversity in multiobjective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 174–188, Apr. 2003.

[6] C. W. Xie and L. X. Ding, "Selection strategies of evolutionary algorithms in multiobjective optimization," in *Proc. 5th Int. Conf. Nat. Comput. (ICNC)*, vol. 4. Tianjin, China, 2009, pp. 633–637.

[7] Y. S. Liau, K. C. Tan, J. Hu, X. Qiu, and S. B. Gee, "Machine learning enhanced multi-objective evolutionary algorithm based on decomposition," in *Proc. Intell. Data Eng. Autom. Learn. (IDEAL)*, Hefei, China, 2013, pp. 553–560.

[8] N. Srinivas and K. Deb, "Multiobjective optimization using nondominated sorting in genetic algorithms," *Evol. Comput.*, vol. 2, no. 3, pp. 221–248, 1994.

[9] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[10] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization," in *Proc. Evol. Methods Design Optim. Control Appl. Ind. Probl. (EUROGEN)*, Athens, Greece, 2001, pp. 95–100.

[11] J. D. Knowles and D. W. Corne, "Approximating the nondominated front using the Pareto archived evolution strategy," *Evol. Comput.*, vol. 8, no. 2, pp. 149–172, 2000.

[12] J. D. Knowles, D. W. Corne, and M. Fleischer, "Bounded archiving using the Lebesgue measure," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, vol. 4. Canberra, ACT, Australia, 2003, pp. 2490–2497.

[13] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. da Fonseca, "Performance assessment of multiobjective optimizers: An analysis and review," *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 117–132, Apr. 2003.

[14] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Parallel Problem Solving From Nature—PPSN VIII*. Berlin, Germany: Springer, 2004, pp. 832–842.

[15] N. Beume, B. Naujoks, and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *Eur. J. Oper. Res.*, vol. 181, no. 3, pp. 1653–1669, 2007.

[16] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *Evol. Comput.*, vol. 19, no. 1, pp. 45–76, 2011.

[17] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.

[18] H. Li and Q. Zhang, "Comparison between NSGA-II and MOEA/D on a set of multiobjective optimization problems with complicated Pareto sets," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, Apr. 2009.

[19] H.-L. Liu, F. Gu, and Q. Zhang, "Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems," *IEEE Trans. Evol. Comput.*, vol. 18, no. 3, pp. 450–455, Jun. 2014.

[20] H. Ishibuchi and T. Murata, "A multi-objective genetic local search algorithm and its application to flowshop scheduling," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 28, no. 3, pp. 392–403, Aug. 1998.

[21] T. Murata, H. Ishibuchi, and M. Gen, "Specification of genetic search directions in cellular multi-objective genetic algorithms," in *Evolutionary Multi-Criterion Optimization (EMO)* (LNCS 1993). Berlin, Germany: Springer, 2001, pp. 82–95.

[22] R. T. Marler and J. S. Arora, "Survey of multi-objective optimization methods for engineering," *Struct. Multidiscipl. Optim.*, vol. 26, no. 6, pp. 369–395, 2004.

[23] I. Giagkiozis, R. C. Purshouse, and P. J. Fleming, "Generalized decomposition," in *Evolutionary Multi-Criterion Optimization (EMO)* (LNCS 7811). Berlin, Germany: Springer, 2013, pp. 428–442.

[24] H. Jain and K. Deb, "An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach—Part II: Handling constraints and extending to an adaptive approach," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 602–622, Aug. 2014.

[25] Y. Qi *et al.*, "MOEA/D with adaptive weight adjustment," *Evol. Comput.*, vol. 22, no. 2, pp. 231–264, 2014.

[26] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach—Part I: Solving problems with box constraints," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 577–601, Aug. 2014.

[27] H. Ishibuchi, Y. Sakane, N. Tsukamoto, and Y. Nojima, "Effects of using two neighborhood structures on the performance of cellular evolutionary algorithms for many-objective optimization," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Trondheim, Norway, 2009, pp. 2508–2515.

[28] H. Ishibuchi, N. Akedo, and Y. Nojima, "Relation between neighborhood size and MOEA/D performance on many-objective problems," in *Evolutionary Multi-Criterion Optimization (EMO)* (LNCS 7811). Berlin, Germany: Springer, 2013, pp. 459–474.

[29] Z.-H. Zhan *et al.*, "Multiple populations for multiple objectives: A coevolutionary technique for solving multiobjective optimization problems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 43, no. 2, pp. 445–463, Apr. 2013.

[30] B.-B. Li and L. Wang, "A hybrid quantum-inspired genetic algorithm for multiobjective flow shop scheduling," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 3, pp. 576–591, Jun. 2007.

[31] H. Xia, J. Zhuang, and D. Yu, "Combining crowding estimation in objective and decision space with multiple selection and search strategies for multi-objective evolutionary optimization," *IEEE Trans. Cybern.*, vol. 44, no. 3, pp. 2168–2267, Mar. 2014.

[32] D. Liu, K. C. Tan, C. K. Goh, and W. K. Ho, "A multiobjective memetic algorithm based on particle swarm optimization," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 1, pp. 42–50, Feb. 2007.

[33] Y. Wang, Z. Cai, G. Guo, and Y. Zhou, "Multiobjective optimization and hybrid evolutionary algorithm to solve constrained optimization problems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 3, pp. 560–575, Jun. 2007.

[34] H. Ishibuchi, K. Narukawa, N. Tsukamoto, and Y. Nojima, "An empirical study on similarity-based mating for evolutionary multiobjective combinatorial optimization," *Eur. J. Oper. Res.*, vol. 188, no. 1, pp. 57–75, 2008.

[35] S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, "Decomposition-based multiobjective evolutionary algorithm with an ensemble of neighborhood sizes," *IEEE Trans. Evol. Comput.*, vol. 16, no. 3, pp. 442–446, Jun. 2012.

[36] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evol. Comput.*, vol. 8, no. 2, pp. 173–195, 2000.

[37] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, *Scalable Test Problems for Evolutionary Multiobjective Optimization*. London, U.K.: Springer, 2005.

[38] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 4–31, Feb. 2011.

[39] K. Deb and M. Goyal, "A combined genetic adaptive search (GeneAS) for engineering design," *Comput. Sci. Informat.*, vol. 26, no. 4, pp. 30–45, 1996.

[40] C. A. C. Coello and N. C. Cortés, "Solving multiobjective optimization problems using an artificial immune system," *Genet. Program. Evolv. Mach.*, vol. 6, no. 2, pp. 163–190, 2005.

[41] O. Schutze, X. Esquivel, A. Lara, and C. A. C. Coello, "Using the averaged Hausdorff distance as a performance measure in evolutionary multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 16, no. 4, pp. 504–522, Aug. 2012.

[42] Z.-H. Zhan, J. Zhang, Y. Li, and H.-H. Chung, "Adaptive particle swarm optimization," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 6, pp. 1362–1381, Dec. 2009.

Zhenkun Wang was born in Shandong, China, in 1990. He received the B.S. degree in electronic information engineering from Shandong Jianzhu University, Jinan, China, in 2011. He is currently pursuing the Ph.D. degree in circuits and systems with Xidian University, Xi'an, China.

His current research interests include multiobjective optimization, evolutionary computation, and other machine learning techniques.

Qingfu Zhang (M'01–SM'06) received the B.Sc. degree in mathematics from Shanxi University, Taiyuan, China, in 1984, the M.Sc. degree in applied mathematics and the Ph.D. degree in information engineering from Xidian University, Xi'an, China, in 1991 and 1994, respectively.

He is a Professor with the Department of Computer Science, City University of Hong Kong, Hong Kong, and the School of Computer Science and Electronic Engineering, University of Essex, Colchester, U.K., and a Changjiang Visiting Chair Professor with Xidian University. He is currently leading the Metaheuristic Optimization Research Group, City University of Hong Kong.

Dr. Zhang was a recipient of the 2010 IEEE Transactions on Evolutionary Computation Outstanding Paper Award. Multiobjective evolutionary algorithms based on decomposition, a multiobjective optimization algorithm developed in his group, won the Unconstrained Multiobjective Optimization Algorithm Competition at the Congress of Evolutionary Computation 2009. He is an Associate Editor of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION and the IEEE TRANSACTIONS ON CYBERNETICS.

Aimin Zhou (S'08–M'10) received the B.Sc. and M.Sc. degrees in computer science from Wuhan University, Wuhan, China, in 2001 and 2003, respectively, and the Ph.D. degree in computer science from the University of Essex, Colchester, U.K., in 2009.

He is currently an Associate Professor with the Department of Computer Science and Technology, East China Normal University, Shanghai, China. His current research interests include evolutionary computation, machine learning, image processing, and their applications. He has published over 30 peer-reviewed papers on evolutionary computation and image processing.

Maoguo Gong (M'07–SM'14) received the B.S. degree in electronic engineering (First Class Hons.) and the Ph.D. degree in electronic science and technology from Xidian University, Xi'an, China, in 2003 and 2009, respectively.

Since 2006, he has been a Teacher at Xidian University, where he was promoted as an Associate Professor and a Professor, both with exceptive admission, in 2008 and 2010, respectively. His current research interests include computational intelligence and applications.

Dr. Gong is the Vice-Chair of the IEEE CIS Task Force on Memetic Computing, an Executive Committee Member of the Chinese Association for Artificial Intelligence, a Senior Member of the Chinese Computer Federation, and an Editorial Board Member of five journals, including the *International Journal of Bio-Inspired Computation*.

Licheng Jiao (SM'89) received the B.S. degree from Shanghai Jiaotong University, Shanghai, China, in 1982, and the M.S. and Ph.D. degrees from Xi'an Jiaotong University, Xi'an, China, in 1984 and 1990, respectively.

Since 1992, he has been a Professor at the School of Electronic Engineering, Xidian University, Xi'an, where he is currently the Director of the Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education of China. His current research interests include image processing, natural computation, machine learning, and intelligent information processing.

Dr. Jiao is the President of the Computational Intelligence Chapter, the IEEE Xi'an Section, and the IET Xi'an Network, the Chairman of the Awards and Recognition Committee, the Vice Board Chairperson of the Chinese Association of Artificial Intelligence, a Councilor of the Chinese Institute of Electronics, a Committee Member of the Chinese Committee of Neural Networks, and an expert of the Academic Degrees Committee of the State Council.