

A Constrained Decomposition Approach with Grids for Evolutionary Multiobjective Optimization

Xinye Cai, *Member, IEEE*, Zhiwei Mei, Zhun Fan, *Senior Member, IEEE*, Qingfu Zhang, *Fellow, IEEE*

Abstract—Decomposition based multiobjective evolutionary algorithms (MOEAs) decompose a multiobjective optimization problem into a set of scalar objective subproblems and solve them in a collaborative way. Commonly used decomposition approaches originate from mathematical programming and the direct use of them may not suit MOEAs due to their population-based property. For instance, these decomposition approaches used in MOEAs may cause the loss of diversity and/or be very sensitive to the shapes of Pareto fronts (PFs). This paper proposes a constrained decomposition with grids (CDG) that can better address these two issues thus more suitable for MOEAs. In addition, different subproblems in CDG defined by the constrained decomposition constitute a grid system. The grids have an inherent property of reflecting the information of neighborhood structures among the solutions, which is a desirable property for restricted mating selection in MOEAs. Based on CDG, a constrained decomposition MOEA with grid (CDG-MOEA) is further proposed. Extensive experiments are conducted to compare CDG-MOEA with the domination-based, indicator-based and state-of-the-art decomposition-based MOEAs. The experimental results show that CDG-MOEA outperforms the compared algorithms in terms of both the convergence and diversity. More importantly, it is robust to the shapes of PFs and can still be very effective on MOPs with complex PFs (e.g., extremely convex, or with disparately scaled objectives).

Index Terms—Evolutionary multiobjective optimization, constrained decomposition, grids, robust to Pareto front

I. INTRODUCTION

Along with domination-based (e.g., [48], [12], [23], [24], [40], [7], [28]) and indicator-based (e.g., [47], [3], [20], [4], [2]) multiobjective evolutionary algorithms (MOEAs), decomposition-based MOEAs (e.g., [31], [42], [32], [13], [34], [18], [19]) have been recognized as a major type of approaches

Xinye Cai and Zhiwei Mei are with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu, 210016 P. R. China, and also with the Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing, 210023, China (e-mail: xinye@nuaa.edu.cn, zwmei@nuaa.edu.cn).

Zhun Fan is with Guangdong Provincial Key Laboratory of Digital Signal and Image Processing and the Department of Electronic Engineering, School of Engineering, Shantou University, Guangdong, P. R. China (e-mail: zfan@stu.edu.cn).

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

This work was supported in part by the National Natural Science Foundation of China (NSFC) under grant 61300159, 61732006, 61473241 and 61332002, by the Natural Science Foundation of Jiangsu Province of China under grant BK20130808, by China Postdoctoral Science Foundation under grant 2015M571751.

Copyright (c) 2012 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

to tackle multiobjective optimization problems (MOPs). As a representative of such approaches, the multiobjective evolutionary algorithm based on decomposition (MOEA/D) [42] has drawn a large amount of attention over the recent years. One critical difficulty for MOEA/D is on how to approximate a set of uniformly-distributed Pareto optimal solutions without knowing the shape of the *Pareto front* (PF) [30], [9] a priori. Commonly-used decomposition approaches in MOEA/D including Weighted Sum (WS), Tchebycheff (TCH) and Penalty-based Boundary Intersection (PBI) [30] may fail to achieve such a goal due to the following two reasons [29], [37].

First, WS, TCH and PBI tend to be very sensitive to the shapes of PFs [42]. An example of the Pareto optimal solutions obtained by TCH on MOPs with an extremely convex or concave PF is given in Fig. 1a. Although the Pareto optimal solutions obtained by TCH are well-distributed on the concave PF, the distribution of the solutions on the extremely convex PF is not satisfactory. In Fig. 1b, another example shows the Pareto optimal solutions obtained by TCH on MOPs with disparately scaled objectives. It can be clearly seen that these Pareto optimal solutions are very unevenly-distributed on PF, where almost half of the PF is not covered by any Pareto optimal solution.

It is worth noting that there has already been some research in the literature to address either one of the above scenarios. As far as we know, an inverted PBI has been proposed to tackle MOPs with extremely convex PFs in [33]. However, the use of inverted PBI to achieve well-distributed solution set still needs to assume the convexity of PFs. A combination of normal boundary intersection and the Tchebycheff approach has been proposed for MOPs with disparately scaled objectives in [43], where a satisfactory distribution of solutions can be achieved in bi-objective optimization problems but fails to extend to tri-objective optimization problems.

Second, in those commonly-used decomposition methods, the same solution is very likely to be assigned to many different subproblems, which may lead to the loss of diversity [37], [29]. The reason of such phenomenon can be explained as follows. Let x^i be the current solution for the i -th subproblem, then the *improvement region* of a solution x^i can be defined as the set $\{F(x) \mid x \text{ is better than } x^i \text{ for the } i\text{-th subproblem}\}$ [37], as shown in Fig. 2a-2c. Any new solution in the improvement region for x^i can improve and replace it for the i -th subproblem. As it can be observed in Fig. 2a-2c, the improvement regions for three commonly used decomposition approaches (WS, TCH and PBI), may be too large for causing the loss of diversity.

Over the recent years, some attempts have already been

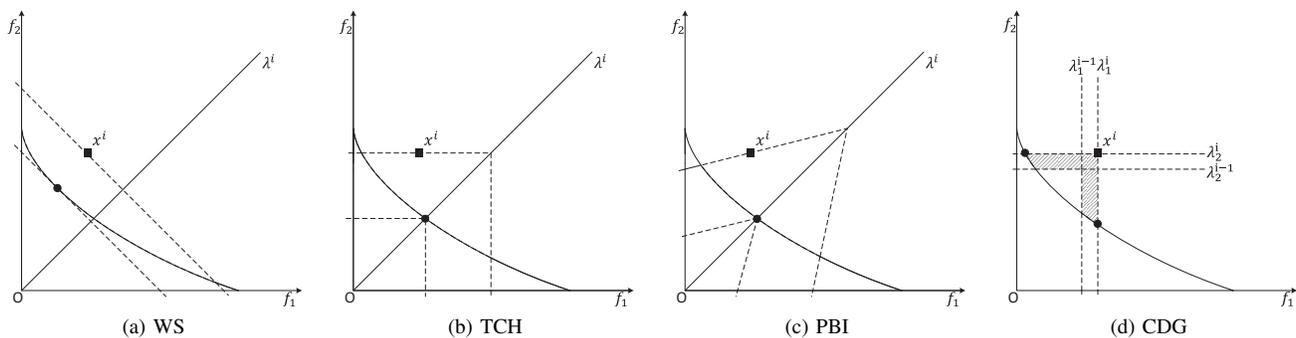


Fig. 2: An illustration of improvement regions for three commonly used decomposition approaches (Fig. 2a-2c). In each sub-figure, the *square* point is solution x^i of i -th subproblem with direction vector λ^i , the *solid circle* point is its optimal solution and the dash line represents its contour. Therefore, the entire shaded region is the improvement region for x^i .

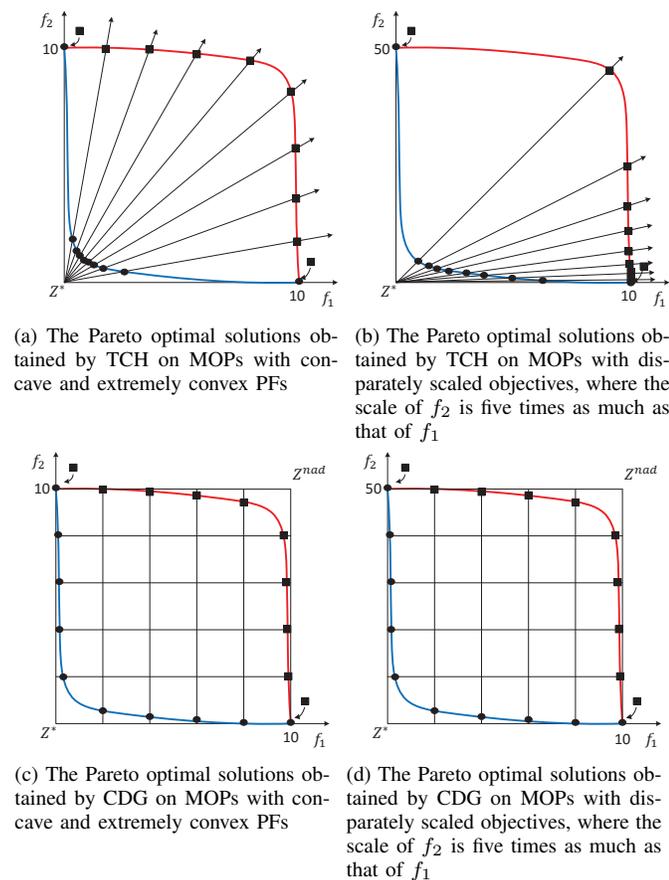


Fig. 1: An illustration of Pareto optimal solutions obtained by TCH or CDG on MOPs with different shapes of PFs

made to maintain better diversity in MOEA/D. Among them, Li et al. limit the number of subproblems allowed to be updated by a single offspring in [26]. The offspring is only allowed to update the most suitable subproblem in [38]. Decomposition approaches have been hybridized with the $R2$ indicator in [36]. An external archive is adopted to maintain the representative solutions and guide the search in the MOEA/D population in [5]. In [27], a global stable matching model (STM) is integrated into MOEA/D to find suitable matches between subproblems and solutions. In [1], an adaptive epsilon compar-

ison approach has been proposed to balance the convergence and diversity. An online geometrical metric has been proposed to enhance the diversity of MOEA/D in [14], [6]. Designing more suitable decomposition approaches for MOEAs is also a possible way for maintaining better diversity in MOEA/D. For instance, the use of different decomposition approaches for different search phases has been studied in [21]. In a very recent work [37], an angle θ is imposed on each subproblem as a constraint to improve the diversity in MOEA/D [37]. However, the appropriate setting of parameter θ for each subproblem is a very tedious task and different subproblems on different evolutionary stages may have different θ values.

In this paper, a constrained decomposition with grids (CDG) is proposed to address the above two aspects for decomposition-based MOEAs. In CDG, one objective function is selected to be optimized while all the other objective functions are converted into constraints by setting up both upper and lower bounds. In a sense, CDG can be considered as an extension of ϵ -constraint approach [17], [30]. If CDG is applied to all the objectives, the volumes of the improvement regions for a solution are appropriately reduced to the narrowed regions, where the same solution can be assigned to at most m subproblems (m is the number of objectives), as shown in Fig. 2d. This interesting characteristic gives CDG a natural ability for maintaining better diversity for MOEAs. In addition, as each objective is equally divided by constraints, unlike the commonly-used decomposition methods (WS, TCH, PBI), CDG is very robust to the shapes of PFs. These can be observed in Fig. 1c and Fig. 1d, where the Pareto optimal solutions obtained by CDG are well-distributed on MOPs with concave or convex PFs, and/or disparately scaled objectives.

Also, another interesting observation in Fig. 1c and Fig. 1d is that the contour lines of different constrained subproblems constitute a grid-coordinate-system. A grid has an inherent property of reflecting the information of neighborhood structures among the solutions [39]: each solution in the grid can be located by grid coordinates and the grid coordinates can help the solutions to locate its neighboring solutions, which is essential for the restricted mating selection in an MOEA. More details of constrained decomposition with grids are specified in Section III.

The rest of this paper is organized as follows. Section II

introduces some preliminaries on MOPs and three popular decomposition approaches are also introduced in this section. Section III details the proposed constrained decomposition with grids. Section IV describes the whole framework of CDG-MOEA. Section V introduces the benchmark test functions and the performance indicators used in the experimental studies. Experiments and discussions are presented in Section VI, where CDG-MOEA is compared with five decomposition-based, one domination-based and one indicator-based MOEAs on a set of well-known benchmark functions and a set of benchmark MOPs with complex PFs. In addition, the sensitivity analysis of parameters in CDG-MOEA is conducted in Section VII. Finally, Section IX concludes this paper.

II. PRELIMINARIES

A. Basic Definitions

A *multiobjective optimization problem* (MOP) can be defined as follows:

$$\begin{aligned} & \text{minimize} && F(x) = (f_1(x), \dots, f_m(x))^T, \\ & \text{subject to} && x \in \Omega. \end{aligned} \quad (1)$$

where Ω is the *decision space* and $F : \Omega \rightarrow R^m$ consists of m real-valued objective functions. $\{F(x)|x \in \Omega\}$ is the *attainable objective set*.

Let $u, v \in R^m$, u is said to *dominate* v , denoted by $u \prec v$, if and only if $u_j \leq v_j$ for every $j \in \{1, \dots, m\}$ and $u_k < v_k$ for at least one index $k \in \{1, \dots, m\}$ ¹. Given a set S in R^m , a solution $x \in S$ is called *nondominated* if no other solution in S dominates it. A solution $x^* \in \Omega$ is *Pareto-optimal* if $F(x^*)$ is nondominated in the attainable objective set. $F(x^*)$ is then called a *Pareto-optimal (objective) vector*. In other words, any improvement in one objective of a Pareto optimal solution is bound to deteriorate at least another objective. The set of all the Pareto-optimal solutions is called the *Pareto set (PS)* and the set of all the Pareto-optimal objective vectors is the *Pareto front (PF)* [30]. The *ideal* and *nadir objective vectors* are another two important concepts containing the information on the ranges of PFs as follows. The *ideal objective vector* z^* is a vector $z^* = \{z_1^*, \dots, z_m^*\}^T$, where $z_j^* = \min_{x \in \Omega} f_j(x), j \in \{1, \dots, m\}$. The *nadir objective vector* z^{nad} is a vector $z^{nad} = \{z_1^{nad}, \dots, z_m^{nad}\}^T$, where $z_j^{nad} = \max_{x \in PS} f_j(x), j \in \{1, \dots, m\}$.

B. Decomposition Approaches

Popular decomposition methods [30] used in MOEAs include Weighted Sum, Tchebycheff and Penalty-based Boundary Intersection, either one of which decomposes an MOP into a number of scalar optimization subproblems [42]. These widely used approaches can be defined as follows.

Let $\lambda^i = (\lambda_1, \dots, \lambda_m)^T$ be a direction vector for the i -th subproblem, where $\lambda_j \geq 0, j \in 1, \dots, m$ and $\sum_{j=1}^m \lambda_j = 1$.

- 1) **Weighted Sum (WS) Approach:** The i -th subproblem is defined as

¹In the case of maximization, the inequality signs should be reversed.

$$\begin{aligned} & \text{minimize} && g^{ws}(x|\lambda^i) = \sum_{j=1}^m \lambda_j f_j(x), \\ & \text{subject to} && x \in \Omega. \end{aligned} \quad (2)$$

Its search direction vector is defined as λ^i , as shown in Fig. 2a.

- 2) **Tchebycheff (TCH) Approach:** The i -th subproblem is defined as

$$\begin{aligned} & \text{minimize} && g^{te}(x|\lambda^i, z^*) = \max_{1 \leq j \leq m} \{|f_j(x) - z_j^*|/\lambda_j^i\}, \\ & \text{subject to} && x \in \Omega. \end{aligned} \quad (3)$$

where Ω is the feasible region, but $\lambda_j = 0$ is replaced by $\lambda_j = 10^{-6}$ because $\lambda_j = 0$ is not allowed as a denominator in (3). Its search direction vector is defined as λ^i , as shown in Fig. 2b.

- 3) **Penalty-based Boundary Intersection (PBI) Approach:** This approach is a variant of Normal-Boundary Intersection approach [8]. The i -th subproblem is defined as

$$\begin{aligned} & \text{minimize} && g^{pbi}(x|\lambda^i, z^*) = d_1^i + \beta d_2^i, \\ & && d_1^i = (F(x) - z^*)^T \lambda^i / \|\lambda^i\|, \\ & && d_2^i = \|F(x) - z^* - (d_1^i / \|\lambda^i\|) \lambda^i\|, \\ & \text{subject to} && x \in \Omega. \end{aligned} \quad (4)$$

where $\|\cdot\|$ denotes the L_2 -norm and β is the penalty parameter. Its search direction vector is defined as λ^i , as shown in Fig. 2c.

III. THE CONSTRAINED DECOMPOSITION WITH GRIDS

In this section, the setup of a grid system and the constrained decomposition based on the grid system are introduced as follows.

A. The Setup of a Grid System

The constrained decomposition is based on a grid-system and the setup of it is introduced in Algorithm 1, as follows. Each objective is divided into K equal intervals within the approximations of the ideal and nadir points, where K is a preset parameter. The width of each interval is

$$d_j = (z_j^{nad} - z_j^* + 2 \times \sigma) / K. \quad (5)$$

Fig. 3a shows a grid division for a bi-objective problem where $K = 4$.

The grid location of x along the j -th objective $g_j(x)$ can be calculated as

$$g_j(x) = \lceil (f_j(x) - z_j^* + \sigma) / d_j \rceil, \quad (6)$$

where $\lceil \cdot \rceil$ denotes the ceil function, $g_j(x)$ is the grid-coordinate of solution x and $f_j(x)$ is the value of j -th objective function. A small positive number σ is introduced to ensure that the value of g_j is more than 0 but not more than K . An example of the grid location along one objective is

demonstrated in Fig. 3b, where $K = 4$. In the example, the grid-coordinates of the solutions (denoted by “•”) along f_1 are assigned as follows: $a = 1, b = 1, c = 2, d = 2, e = 3, f = 3, g = 4$ and $h = 4$.

Algorithm 1: Grid-system Setup (GS)

Input : P : the current population;
 z^* : the approximation of the ideal point;
 z^{nad} : the approximation of the nadir point;
 m : the number of objectives;
 K : the number of the intervals on each objective.

Output: the grid locations of P .

```

1 for  $j = 1$  to  $m$  do
2    $d_j = (z_j^{nad} - z_j^* + 2 \times \sigma) / K$ ;
3 end
4 foreach  $x \in P$  do
5   for  $j = 1$  to  $m$  do
6      $g_j(x) = \lceil (f_j(x) - z_j^* + \sigma) / d_j \rceil$ ;
7   end
8    $G(x) = (g_1(x), \dots, g_m(x))$ ;
9 end

```

To use the grid locations for the restricted mating selection in MOEAs, the Grid Distance and Grid Neighbor are defined for convenience as follows.

Definition 1 (Grid Distance): Let $u, v \in R^m$ be two solutions, the grid distance $GD(u, v)$ between u and v is defined as

$$GD(u, v) = \max_{j=1, \dots, m} (|g_j(u) - g_j(v)|). \quad (7)$$

Definition 2 (Grid Neighbors): The grid neighbors of a solution x within distance T is defined as

$$GN(x, T) = \{x^* | GD(x, x^*) \leq T \quad x, x^* \in R^m\}. \quad (8)$$

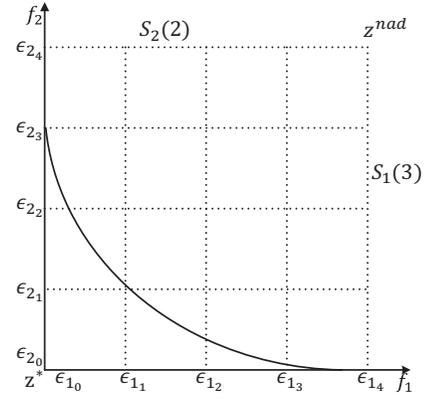
B. Constrained Decomposition with Grids (CDG)

A constrained decomposition can be defined by adopting the grid system specified in the last section. CDG can be considered as an extension of ϵ -constraint approach [30]. The constrained decomposition approach for the k -th subproblem of the l -th objective can be defined as follows.

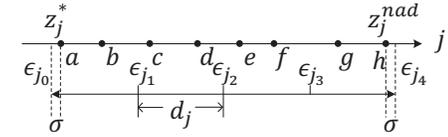
$$\begin{aligned}
 &\text{minimize} && f_l(x), \\
 &\text{subject to} && g_j(x) = k_j \quad \text{for all } j = 1, \dots, m, j \neq l, \\
 &&& k_j \in \{1, \dots, K\}, \\
 &&& x \in \Omega.
 \end{aligned} \quad (9)$$

where K is a division parameter which determines the number of grids.

With K intervals on each objective, the grids decompose an MOP into $m \times K^{m-1}$ subproblems. In general, the k -th subproblem of l -th objective contains a solution set $S_l(k)$ (k



(a) A grid system ($K = 4$) in a bi-objective optimization problem



(b) Grid location along one objective ($K = 4$)

Fig. 3: An illustration of the grid system produced by constrained decomposition

is a $(m - 1)$ -dimensional vector), which can be defined as:

$$\begin{aligned}
 S_l(k) = \{x | &g_1(x) = k_1, \dots, g_{l-1}(x) = k_{l-1}, \\
 &g_{l+1}(x) = k_{l+1}, \dots, g_m(x) = k_m\}, \quad (10)
 \end{aligned}$$

subject to $l \in \{1, \dots, m\} \quad k \in \{1, \dots, K\}^{m-1}$.

An example of such subproblems in a bi-objective optimization is given in Fig. 3a, where the feasible regions of two subproblems $S_1(3)$ and $S_2(2)$ are denoted in the shaded regions.

IV. CDG-MOEA

A. The Main Framework of CDG-MOEA

In this section, the main framework of constrained decomposition based multiobjective evolutionary algorithm with grids (CDG-MOEA) is presented in Algorithm 2, which includes 6 steps: initialization, reproduction, update of the ideal and nadir points, update of the grid system, rank-based selection and termination. In the following sections, each step is specified in details.

Algorithm 2:

Input:

- 1) an MOP;
- 2) a stopping criterion;
- 3) N : the population size of P ;
- 4) T : the maximum grid distance for neighborhood;
- 5) K : the number of the intervals in each objective.

Output: A solution set P ;

Step 1: Initialization:

Step 1.1 Generate an initial population $P = \{x^1, \dots, x^N\}$ randomly;

Step 1.2 Approximate the ideal and nadir points: $z^* = \text{UPDATE1}(P)$, $z^{nad} = \text{UPDATE2}(P)$;

Step 1.3 Initialize the grid system: $GS(P)$;

Step 1.4 Set $gen = 0$.

Step 2: Reproduction:

Step 2.1 Generate an empty set $Q = \emptyset$;

For each solution $x \in P$ **do**

Step 2.2 Obtain the neighboring solutions as the mating pool of x :

$$NS = \begin{cases} GN(x, T), & rand < \delta \text{ and} \\ \{x^1, \dots, x^N\}, & |GN(x, T)| > 2, \text{ otherwise.} \end{cases} \quad (11)$$

Step 2.3 Select two solutions x^k and x^l from NS randomly; and generate an offspring solution y from solution x , x^k and x^l by DE operators; then y is added to Q .

End for

Step 3: Update of the ideal and nadir points:

Step 3.1 $gen = gen + 1$;

Step 3.2 $P = P \cup Q$;

Step 3.3 Update the ideal point: $z^* = \text{UPDATE1}(P)$;

Step 3.4 If gen is a multiplication of 50, update the nadir point: $z^{nad} = \text{UPDATE2}(P)$.

Step 4: Update of the grid system:

Step 4.1 $\bar{P} = \{x | x \in P \wedge \exists j \in \{1, \dots, m\}, f_j(x) > z_j^{nad}\}$;

Step 4.2 $P = P \setminus \bar{P}$;

Step 4.3 Update the grid system: $GS(P)$.

Step 5: Rank-based selection:

Step 5.1 If $|P| < N$, randomly select $N - |P|$ solutions from \bar{P} and add them to P . Otherwise, $P = \text{RBS}(P)$.

Step 6: Termination:

Step 6.1 If the stopping criterion is satisfied, terminate the algorithm and output P . Otherwise, go to **Step 2**.

B. Initialization

In Step 1.1, a population P is initialized randomly. In Step 1.2, the ideal and nadir points are approximated based on P . The update of the ideal point is presented in Algorithm 3 and the update of the nadir point is presented in Algorithm 4.

Algorithm 3: Update the Ideal Point (UPDATE1)

Input : P : the current population.

Output: Updated ideal point z^* .

```

1 for  $j = 1$  to  $m$  do
2    $z_j^* = \min_{x \in P} \{f_j(x)\}$ ;
3 end

```

C. Reproduction

In the Step 2, N offspring solutions are generated from P . An empty set Q is generated for storing the offspring solutions. For each solution $x \in P$, its mating solutions are obtained by

$$NS = \begin{cases} GN(x, T), & rand < \delta \text{ and} \\ \{x^1, \dots, x^N\}, & |GN(x, T)| > 2, \text{ otherwise.} \end{cases} \quad (12)$$

Algorithm 4: Update the Nadir Point (UPDATE2)

Input : P : the combined population;

z^* : the current ideal point;

z^{nad} : the current nadir point.

Output: Updated nadir point z^{nad} .

/ To reduce computational cost, find a subset of P : SP */*

```

1  $SP = \emptyset$ ;
2 foreach  $x \in P$  do
3   for  $j = 1$  to  $m$  do
4     if  $f_j(x) < z_j^* + \frac{z_j^{nad}}{5}$  then
5        $SP = SP \cup x$ ;
6     end
7   end
8 end
/* find all the nondominated solutions in  $SP$ . */
9  $SP = \text{NONDOMINATED-SELECTION}(SP)$ ;
10 for  $j = 1$  to  $m$  do
11    $z_j^{nad} = \max_{x \in SP} \{f_j(x)\}$ ;
12 end

```

where δ is the probability that the mating solutions are selected from the grid neighbors.

In Step 2.3, two solutions x^k and x^l are selected from NS randomly. An offspring solution y is generated from solution x , x^k and x^l by DE operators [26]; and then y is added to Q .

D. Update of the Ideal and Nadir Points

In Step 3, the approximations of the ideal and nadir points are updated by using the combined population $P = P \cup Q$. In Algorithm 3, the ideal point z^* is approximated by the minimum value of each objective in P .

In Algorithm 4, the nadir point is approximated by the maximum value of each objective in the nondominated solutions of P . To further lower the computational cost of nondominated selection for approximating nadir point, only a subset of solutions SP in P that are close to the corner solutions is selected (line 1-8 of Algorithm 4), as follows:

$$SP = \{x | x \in P \wedge \exists j \in \{1, \dots, m\}, f_j(x) < z_j^* + \frac{z_j^{nad}}{5}\}. \quad (13)$$

After that, the nondominated solutions are selected from SP and the nadir point is updated with the maximum value of each objective in these nondominated solutions (line 9-12 of Algorithm 4).

E. Update of the Grid System

In Step 4.1-4.2, the solutions $\bar{P} = \{x | x \in P \wedge \exists j \in \{1, \dots, m\}, f_j(x) > z_j^{nad}\}$ (the ones located outside the nadir point approximation) are eliminated from P first. Then, in Step 4.3, the grid system is updated using P by calling Algorithm 1, which was already specified in Section III.B.

F. Rank-Based Selection

In Step 5, if the size of P is less than N , then $N - |P|$ solutions are selected randomly from \bar{P} to fill in P ; otherwise, the rank-based selection (RBS), presented in Algorithm 5, is called.

In Algorithm 5, N solutions are selected from Q ($|Q| > N$) by rank-based selection, which include decomposition-based ranking and lexicographic sorting. The details of them are explained as follows.

In the decomposition-based ranking, each solution set $S_l(k)$ for the k -th subproblem of the l -th objective is ranked based on constrained decomposition with grids defined in Eq. (9). After ranking all the m objectives, each solution x has m ranks, stored in a rank vector $R(x) = (r_1(x), \dots, r_m(x))$. It is worth noting that, with mK^{m-1} subproblems, at most mM times rankings are needed as some subproblems may contain no solutions, where $M = |Q|$.

In the lexicographic sorting, $R(x)$ is sorted in an ascending order and stored in $R'(x)$. Then, each solution $x \in Q$ is ranked in lexicographic order based on $R'(x)$ and the first N solutions are selected from Q and assigned to P .

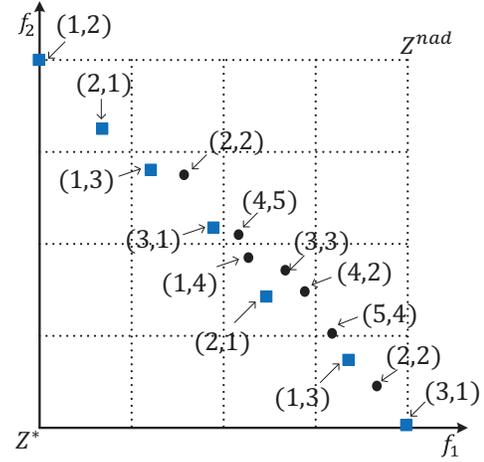


Fig. 4: The ranks of all the solutions in a grid system ($K = 4$) for a bi-objective optimization problem, where the final selected solutions are denoted in ■.

TABLE I: The procedures of decomposition-based ranking and lexicographic sorting for Fig. 4

Algorithm 5: Rank-based Selection (RBS)

Input : Q : the combined population.
Output: A population P .

/* Decomposition-based Ranking */

```

1 foreach  $x \in Q$  do
2   initialize  $R(x) = \{r_1(x), \dots, r_m(x)\} = \{0, \dots, 0\}$ ;
3 end
4 for  $l = 1$  to  $m$  do
5   forall subproblems  $S_l(k)$  do
6     if  $S_l(k) \neq \emptyset$  then
7        $[S', I] = \text{sort}_l(S_l(k))$ ; // sort  $S_l(k)$  by
            $f_l$  based on Eq. (9) and  $I$  stores
           the ranks
8       foreach  $x \in S_l(k)$  do  $r_l(x) = I(x)$ ;
9     end
10  end
11 end

```

/* Lexicographic Sorting */

```

12 foreach  $x \in Q$  do
13    $R'(x) = \text{sort}(R(x))$ ;
14 end
15  $Q = \text{LEXICOGRAPHIC-SORT}(Q)$ ;
16  $P = Q(1:N)$ ; // select first  $N$  solutions

```

$R(x)$	$R'(x)$	Lexicographic Sorting
1. (1,2)	1. (1,2)	1. (1,2) ✓
2. (2,1)	2. (1,2)	2. (1,2) ✓
3. (1,3)	3. (1,3)	8. (1,2) ✓
4. (2,2)	4. (2,2)	3. (1,3) ✓
5. (3,1)	5. (1,3)	5. (1,3) ✓
6. (4,5)	6. (4,5)	12. (1,3) ✓
7. (1,4)	7. (1,4)	14. (1,3) ✓
8. (2,1)	8. (1,2)	7. (1,4) ×
9. (3,3)	9. (3,3)	4. (2,2) ×
10. (4,2)	10. (2,4)	13. (2,2) ×
11. (5,4)	11. (4,5)	10. (2,4) ×
12. (1,3)	12. (1,3)	9. (3,3) ×
13. (2,2)	13. (2,2)	6. (4,5) ×
14. (3,1)	14. (1,3)	11. (4,5) ×

Fig. 4 and Table I show an example of rank-based selection in a grid system ($K = 4$) for a bi-objective optimization problem, where $|P| = 7$ solutions are selected out of $|Q| = 14$ solutions in Algorithm 5, as follows. All the solutions are ranked as $R(x)$ based on Eq. (9), as shown in Fig. 4 and the first column of Table I. The ranks are sorted and stored in $R'(x)$ (second column of Table I). After the lexicographic sorting, the first 7 solutions are selected (the last column of Table I).

G. Computational Complexity

In Algorithm 1, the setup of a grid system requires $O(mN)$ computations, where m is the number of objectives and N is the population size. In Algorithm 3, the update of the ideal point needs $O(mN)$ computations. In Algorithm 4, the computational complexity for the update of the nadir point is $O(mL^2)$, where L is the size of nondominated solutions in SP .

In Algorithm 5, the rank-based selection is divided into two parts: the decomposition-based ranking and the lexicographic sorting. Among them, the decomposition-based ranking needs to be done at most mM times, where $M \leq 2N$ is the size of solutions that are located inside the nadir point approximation. In the lexicographic sorting, the sorting in line 12-14 requires $O(mM \log m)$ computations and the second sorting in line 15 requires $O(mM \log M)$ computations. In summary, the computational cost of rank-based selection is $O(mM \log M)$, where $M \leq 2N$.

In the framework of CDG-MOEA (Algorithm 2), the computational cost of both Step 1 and Step 2 is $O(mN)$. The computational cost of Step 2-5 is $O(mM \log M)$. In summary, the

computational complexity of CDG-MOEA is $O(mM \log M)$.

V. EXPERIMENTAL SETTINGS

A. Test Instances

UF suite [45] is considered as the benchmark functions in our experimental studies. Among all the 10 benchmark problems, UF1-7 are bi-objective and UF8-10 are tri-objective optimization problems. The number of decision variables for all the test instances is set to 30.

GLT suite [15], [41] is another set of the benchmark problems used in our experimental studies to verify the robustness of CDG-MOEA on MOPs with complicated PFs (e.g., extremely convex or with disparately scaled objectives). The number of decision variables for all the test instances is set to 10.

B. Parameter Settings

To make a fair comparison, the following parameters of all the compared algorithms are set in a similar manner as in MOEA/D-DE [26]:

- $\delta = 0.9$;
- In DE: $CR = 1.0$, $F = 0.5$, $\eta = 20$ and $p_m = 1/n$;
- Function evaluations: 300,000 for each test instance;

The population size of all the compared algorithms is set to 300 for UF1-7 and GLT1-6; and 600 for UF8-10. For MOEA/D-DE (WS, TCH and PBI), the parameter of neighborhood size is set to 20, the same as the one in [26]. For CDG-MOEA, the grid division parameter K is set to 180 for bi-objective and 30 for tri-objective problems; and the grid neighborhood distance is set to 5 for bi-objective and 1 for tri-objective problems. The settings of the grid division parameter and population size are based on the sensitivity analysis in Section VII. Each of all the compared algorithms is run 30 times independently for all the benchmark problems.

C. Performance Metrics

- 1) **Inverted Generational Distance (IGD)** [46], [44]: It measures the average distance from a set of reference points P^* in the PF to the approximation set P . It can be formulated as follows.

$$IGD(P, P^*) = \frac{1}{|P^*|} \sum_{v \in P^*} dist(v, P) \quad (14)$$

where $dist(v, P)$ is the Euclidean distance between the solution v and its nearest point in P , and $|P^*|$ is the cardinality of P^* . If $|P^*|$ is large enough to represent the PF very well, $IGD(P, P^*)$ could measure both the diversity and convergence of P in a sense. In our experiments, 626 reference points are used on GLT1, 1000 reference points on GLT2-4, 2600 reference points on GLT5, and 1377 reference points on GLT6, for calculating IGD .

- 2) **Hypervolume Indicator (I_H)** [49]: Let $z^r = (z_1^r, \dots, z_m^r)^T$ be a reference point in the objective space that is dominated by all Pareto-optimal objective vectors. Let P be the obtained approximation to the PF in the objective space. Then, the I_H value of P (with regard

to z^r) is the volume of the region dominated by P and bounded by z^r , and it can be defined as

$$I_H(P) = volume\left(\bigcup_{f \in P} [f_1, z_1^r] \times \dots [f_m, z_m^r]\right). \quad (15)$$

Obviously, the higher the hypervolume, the better the approximation is. In our experiments, z^r is set to $(2, 2)^T$ for UF1-7, $(2, 2, 2)^T$ for UF8-10, $(1.2, 1.2)^T$ for GLT1 and GLT3, $(1.2, 12)^T$ for GLT2, $(1.2, 2.2)^T$ for GLT4 and $(1.2, 1.2, 1.2)^T$ for GLT5-6 when computing hypervolume for the nondominated sets obtained by all the algorithms.

VI. EXPERIMENTAL STUDIES AND DISCUSSIONS

In this section, the following experiments are conducted to test the performance of CDG-MOEA:

- comparisons on UF test suite, to verify the general performance of CDG-MOEA on balancing between convergence and diversity.
- comparisons on GLT test suite with complicated PFs (extremely convex and/or with disparately scaled objectives), to verify the robustness of CDG-MOEA on the shapes of PFs.

A. Comparisons on UF test suite

In this section, CDG-MOEA is compared with four classical decomposition-based MOEAs (MOEA/D-DE (WS, TCH or PBI) [26] and MSOPS-II [19]), one state-of-the-art decomposition-based MOEA (MOEA/D-ACD [37]), one domination-based MOEA (NSGA-II [12]), one indicator-based MOEA (IBEA [47]) and three grid-based MOEAs (Epsilon MOEA [11], Borg MOEA [16] and OMOPSO [35]) on UF test problems.

The performance of eleven compared algorithms in terms of IGD or I_H is presented in Table II, where the performance of the algorithm with the best mean IGD or I_H value is highlighted in boldface. It can be observed that, in terms of IGD , CDG-MOEA significantly outperforms other compared algorithms on all the test problems, except for UF3, UF4, UF8 and UF10. MOEA/D-ACD achieves the best performance on UF3, Borg has the best performance on UF4, MOEA/D-DE (PBI) has the best performance on UF8, and MSOPS-II has the best performance on UF10.

Similar performance can be observed on the comparisons of eight algorithms in terms of I_H , where CDG-MOEA is significantly better than other compared algorithms on six out of ten test problems.

The comparisons of CDG-MOEA with other ten algorithms on UF test suite verify that CDG in CDG-MOEA is able to achieve better diversity while maintaining good convergence in most test problems, compared with other commonly-used decomposition methods thus more suitable for MOEA framework.

TABLE II: Performance comparisons of CDG-MOEA with ten MOEAs on UF problems in terms of the mean and standard deviation values of IGD and I_H .

instance	CDG-MOEA	MOEA/D-DE			MOEA/D-ACD	MSOPS-II	NSGA-II	IBEA	ϵ -MOEA	Borg	OMOPSO	
		WS	TCH	PBI								
IGD												
UF1	mean	2.072E-03	6.113E-02 †	2.439E-03 †	2.394E-03 †	2.096E-03	7.023E-02 †	8.390E-02 †	1.084E-01 †	1.206E-01 †	8.217E-02 †	6.790E-02 †
	std	5.063E-05	1.399E-02	5.026E-04	7.929E-04	9.026E-05	5.731E-03	1.187E-02	1.715E-02	3.709E-02	2.426E-02	1.325E-02
UF2	mean	5.191E-03	5.442E-02 †	1.118E-02 †	3.532E-02 †	7.970E-03 †	5.707E-02 †	3.272E-02 †	4.877E-02 †	4.930E-02 †	1.871E-02 †	2.047E-02 †
	std	1.276E-03	1.102E-02	3.263E-03	3.582E-02	1.918E-03	2.137E-02	2.355E-03	2.148E-02	2.007E-02	4.799E-03	1.888E-03
UF3	mean	1.832E-02	1.252E-01 †	2.539E-02	4.670E-02 †	6.120E-03 ‡	3.141E-01 †	7.031E-02 †	3.136E-01 †	2.916E-01 †	9.785E-02 †	6.579E-02 †
	std	1.367E-02	3.208E-02	2.157E-02	3.615E-02	3.894E-03	1.784E-02	1.163E-02	1.145E-02	4.240E-02	4.687E-02	2.771E-02
UF4	mean	4.077E-02	3.431E-01 †	6.767E-02 †	6.177E-02 †	6.241E-02 †	5.393E-02 †	7.606E-02 †	4.389E-02 †	4.205E-02 †	3.230E-02 ‡	4.787E-02 †
	std	7.424E-04	4.382E-03	2.849E-03	3.588E-03	4.784E-03	2.545E-03	1.370E-02	2.477E-03	1.776E-03	7.427E-04	1.943E-03
UF5	mean	1.446E-01	3.026E-01 †	2.901E-01 †	3.674E-01 †	3.236E-01 †	3.429E-01 †	6.793E-01 †	2.851E-01 †	3.161E-01 †	1.842E-01 †	6.398E-01 †
	std	2.713E-02	3.446E-02	4.636E-02	1.476E-01	9.074E-02	1.004E-01	1.006E-01	9.974E-02	1.060E-01	3.768E-02	3.972E-01 †
UF6	mean	6.140E-02	2.731E-01 †	1.868E-01 †	3.792E-01 †	1.108E-01 †	2.960E-01 †	3.207E-01 †	2.993E-01 †	4.145E-01 †	2.851E-01 †	3.204E-01 †
	std	3.085E-02	1.451E-01	1.361E-01	2.329E-01	7.349E-02	2.346E-01	7.719E-02	1.567E-01	1.796E-01	1.018E-01	1.926E-01
UF7	mean	2.634E-03	3.514E-01 †	4.067E-03 †	6.088E-03 †	2.722E-03 †	3.858E-02 †	3.504E-01 †	2.279E-01 †	2.678E-01 †	7.735E-02 †	4.009E-02 †
	std	1.441E-04	4.038E-03	9.467E-04	4.618E-03	4.302E-04	6.747E-03	8.797E-03	1.549E-01	1.724E-01	1.276E-01	4.461E-02
UF8	mean	5.846E-02	5.359E-01 †	6.213E-02 †	2.401E-02 ‡	6.593E-02 †	1.902E-01 †	2.671E-01 †	4.420E-01 †	3.143E-01 †	3.706E-01 †	1.693E-01 †
	std	1.259E-02	5.586E-02	7.577E-03	7.142E-04	6.511E-03	4.524E-03	5.537E-02	2.371E-04	5.935E-02	1.186E-01	2.531E-02
UF9	mean	4.530E-02	3.653E-01 †	6.111E-02 †	9.091E-02 †	1.068E-01 †	2.344E-01 †	1.840E-01 †	2.000E-01 †	1.608E-01 †	2.200E-01 †	3.567E-01 †
	std	3.002E-02	3.942E-02	3.914E-02	5.620E-02	5.281E-02	3.507E-02	7.033E-02	6.355E-02	5.736E-02	6.541E-02	4.588E-02
UF10	mean	9.121E-01	4.066E-01 †	4.971E-01 †	5.632E-01 †	7.763E-01 †	2.438E-01 ‡	6.630E-01 †	5.189E-01 †	3.310E-01 †	4.080E-01 †	1.518E+00 †
	std	1.542E-01	6.183E-02	4.517E-02	1.081E-01	1.240E-01	1.205E-01	6.928E-02	6.460E-02	1.512E-01	1.087E-01	3.549E-01
I_H												
UF1	mean	3.663E+00	3.594E+00 †	3.650E+00 †	3.643E+00 †	3.656E+00 †	3.442E+00 †	3.386E+00 †	3.311E+00 †	3.317E+00 †	3.440E+00 †	3.493E+00 †
	std	1.000E-04	1.711E-02	7.779E-03	1.090E-02	2.100E-03	9.338E-02	1.784E-02	1.024E-01	1.089E-01	1.389E-01	9.189E-02
UF2	mean	3.650E+00	3.583E+00 †	3.607E+00 †	3.553E+00 †	3.639E+00 †	3.462E+00 †	3.609E+00 †	3.510E+00 †	3.500E+00 †	3.601E+00 †	3.622E+00 †
	std	1.500E-02	2.500E-02	4.059E-02	1.046E-01	1.512E-02	8.993E-02	7.740E-03	6.899E-02	6.053E-02	3.516E-02	1.397E-02
UF3	mean	3.630E+00	3.450E+00 †	3.549E+00	3.357E+00 †	3.654E+00	2.571E+00 †	3.519E+00 †	2.559E+00 †	2.590E+00 †	3.137E+00 †	3.559E+00 †
	std	3.900E-02	6.599E-02	1.111E-01	1.929E-01	7.162E-03	4.653E-02	4.087E-02	3.747E-02	1.052E-01	1.815E-01	4.195E-02
UF4	mean	3.231E+00	2.877E+00 †	3.135E+00 †	3.124E+00 †	3.152E+00 †	3.188E+00 †	3.079E+00 †	3.217E+00 †	3.213E+00 †	3.241E+00	3.201E+00 †
	std	4.870E-03	1.433E-02	1.674E-02	2.155E-02	1.978E-02	6.537E-03	1.368E-01	6.390E-03	3.752E-03	3.564E-03	5.123E-03
UF5	mean	3.165E+00	2.834E+00 †	2.581E+00 †	2.362E+00 †	2.566E+00 †	2.587E+00 †	1.590E+00 †	2.514E+00 †	2.397E+00 †	2.827E+00 †	1.935E+00 †
	std	1.000E-01	1.197E-01	1.649E-01	2.461E-01	3.017E-01	3.366E-01	2.378E-01	2.156E-01	2.895E-01	2.009E-01	8.799E-01
UF6	mean	3.231E+00	2.761E+00 †	2.906E+00 †	2.526E+00 †	3.067E+00 †	2.665E+00 †	2.623E+00 †	2.702E+00 †	2.390E+00 †	2.617E+00 †	2.534E+00 †
	std	7.290E-02	2.919E-01	2.469E-01	4.142E-01	2.546E-01	4.582E-01	1.522E-01	3.278E-01	3.686E-01	3.202E-01	5.677E-01
UF7	mean	3.494E+00	3.001E+00 †	3.480E+00 †	3.455E+00 †	3.484E+00 †	3.403E+00 †	2.545E+00 †	2.854E+00 †	2.784E+00	3.305E+00	3.403E+00
	std	2.580E-03	2.074E-03	8.107E-03	5.177E-02	8.424E-03	6.379E-02	9.270E-03	4.184E-01	4.475E-01	3.632E-01	1.626E-01
UF8	mean	7.323E+00	6.338E+00 †	7.273E+00 †	7.363E+00 ‡	7.297E+00 †	6.399E+00 †	6.373E+00 †	6.424E+00 †	6.476E+00	6.527E+00	6.657E+00
	std	3.373E-02	4.857E-01	2.209E-02	8.799E-03	1.372E-02	1.597E-02	2.263E-01	3.303E-04	1.456E-01	2.338E-01	2.619E-01
UF9	mean	7.653E+00	7.061E+00 †	7.419E+00 †	7.309E+00 †	7.312E+00 †	5.819E+00 †	7.085E+00 †	6.494E+00 †	6.761E+00 †	6.989E+00 †	5.797E+00 †
	std	1.417E-01	2.444E-01	1.513E-01	2.459E-01	2.389E-01	2.444E-01	2.235E-01	3.205E-01	2.346E-01	3.053E-01	3.945E-01
UF10	mean	2.185E+00	4.144E+00 †	3.347E+00 †	3.406E+00 †	2.425E+00 †	5.874E+00 †	2.510E+00 †	5.904E+00 ‡	5.216E+00 †	4.099E+00 †	6.961E-01 †
	std	6.270E-01	4.462E-01	2.021E-01	3.266E-01	4.125E-01	9.841E-01	3.098E-01	7.103E-01	1.223E+00	1.013E+00	7.549E-01

Wilcoxon's rank sum test at a 0.05 significance level is performed between CDG-MOEA and each of the other competing algorithms. † and ‡ denotes that the performance of the corresponding algorithm is significantly worse than or better than that of CDG-MOEA, respectively. The best mean is highlighted in boldface.

B. Comparisons on GLT test suite

To verify the robustness of CDG-MOEA on MOPs with different shapes of PFs, CDG-MOEA is compared with ten algorithms on GLT test suite in this section.

To visualize the performance of all the compared algorithms on MOPs with different characteristics, the final nondominated sets obtained by all the compared algorithms in the run with the median IGD value on GLT1, GLT2 and GLT6 are presented in Fig. 5-7. For GLT1 whose PF is two segments of disconnected straight lines, the performance of all the algo-

gorithms is very similar, except for MOEA/D-DE (WS), NSGA-II and IBEA. The nondominated set obtained by MOEA/D-DE (WS) is degenerated to two small regions on the corners of PF. This is because the weighted sum approach is unable to approximate the nonconvex parts of PF. For GLT2 whose PF has disparately scaled objectives and/or GLT6 whose PFs are extremely convex, only CDG-MOEA is able to approximate the widely and uniformly distributed nondominated solution set. These results verify that, unlike other algorithms, CDG-MOEA is able to remain effective on MOPs with complex PFs

TABLE III: Performance comparisons of CDG-MOEA with ten MOEAs on GLT problems in terms of the mean and standard deviation values of IGD and I_H .

instance	CDG-MOEA	MOEA/D-DE			MOEA/D-ACD	MSOPS-II	NSGA-II	IBEA	ϵ -MOEA	Borg	OMOPSO	
		WS	TCH	PBI								
IGD												
GLT1	mean	1.748E-03	1.771E-01 †	1.253E-03 ‡	1.430E-03 ‡	1.216E-03 ‡	1.780E-02 †	1.121E-01 †	1.012E-01 †	1.570E-01 †	9.084E-02 †	4.831E-02 †
	std	1.629E-04	9.065E-14	2.941E-06	3.824E-06	9.914E-06	6.084E-03	3.738E-02	2.479E-02	2.494E-02	5.252E-02	3.463E-02
GLT2	mean	1.158E-02	4.522E-02 †	1.205E-01 †	2.429E+00 †	1.354E-01 †	6.773E-02 †	1.087E-01 †	3.040E-01 †	4.712E-01 †	1.926E-01 †	8.558E-02 †
	std	1.492E-03	1.350E-03	1.947E-02	4.209E-04	1.980E-02	1.886E-02	1.359E-01	8.396E-02	3.242E-01	1.600E-01	2.885E-03
GLT3	mean	1.834E-03	2.374E-01 †	6.537E-03 †	4.535E-01 †	6.536E-03 †	4.743E-02 †	6.639E-02 †	1.506E-01 †	8.166E-02 †	1.047E-01 †	5.554E-02 †
	std	4.287E-05	1.383E-06	1.535E-04	1.932E-04	4.719E-04	1.598E-02	2.846E-02	6.907E-02	2.382E-02	2.081E-02	1.833E-03
GLT4	mean	2.923E-03	2.724E-01 †	3.383E-03 †	9.696E-02 †	3.246E-03 †	1.276E-02 †	1.444E-01 †	2.316E-01 †	3.251E-01 †	2.515E-01 †	1.229E-01 †
	std	6.339E-05	6.933E-02	8.270E-06	2.594E-02	6.589E-05	6.010E-03	4.927E-02	4.598E-03	1.371E-01	1.121E-01	6.068E-02
GLT5	mean	2.016E-02	2.904E-02 †	6.839E-02 †	1.246E-01 †	3.887E-02 †	1.306E-01 †	3.115E-02 †	6.964E-02 †	5.025E-02 †	3.112E-02 †	2.677E-02 †
	std	1.524E-03	5.505E-04	9.744E-04	4.637E-03	4.485E-03	7.923E-02	3.193E-03	7.197E-03	2.588E-02	7.382E-03	2.548E-03
GLT6	mean	1.882E-02	6.799E-02 †	4.115E-02 †	1.785E-01 †	2.398E-02 †	1.148E-01 †	3.207E-02 †	6.005E-02 †	3.927E-02 †	3.724E-02 †	3.059E-02 †
	std	5.105E-03	6.283E-02	8.534E-04	2.456E-02	2.688E-03	5.480E-02	4.172E-03	7.718E-03	7.995E-03	4.213E-03	4.620E-03
I_H												
GLT1	mean	8.128E-01	4.400E-01 †	8.141E-01 ‡	8.140E-01 ‡	8.133E-01 ‡	6.599E-01 †	5.663E-01 †	5.542E-01 †	4.777E-01 †	5.895E-01 †	6.834E-01 †
	std	3.679E-03	4.168E-15	1.142E-05	7.386E-06	7.031E-04	4.850E-02	5.928E-02	4.832E-02	5.546E-02	7.306E-02	4.206E-02
GLT2	mean	1.223E+01	1.218E+01 †	1.214E+01 †	8.635E+00 †	1.213E+01 †	1.201E+01 †	1.197E+01 †	1.162E+01 †	1.105E+01 †	1.199E+01 †	1.212E+01 †
	std	2.245E-04	1.510E-03	1.779E-02	1.061E-03	1.458E-02	7.229E-02	4.109E-01	2.146E-01	8.633E-01	3.071E-01	4.250E-03
GLT3	mean	1.389E+00	1.342E+00 †	1.388E+00 †	1.325E+00 †	1.388E+00 †	1.374E+00 †	1.376E+00 †	1.359E+00 †	1.373E+00 †	1.367E+00 †	1.376E+00 †
	std	6.441E-04	2.772E-07	1.242E-04	2.342E-05	2.815E-04	3.875E-03	5.631E-03	7.815E-03	4.673E-03	4.209E-03	4.490E-04
GLT4	mean	1.635E+00	1.370E+00 †	1.635E+00	1.551E+00 ‡	1.413E+00 †	1.490E+00 †	1.532E+00 †	1.475E+00 †	1.137E+00 †	1.297E+00 †	1.517E+00 †
	std	9.133E-05	2.235E-01	2.908E-05	2.803E-02	6.374E-02	6.773E-02	3.496E-02 †	5.266E-03	3.643E-01	3.153E-01	1.129E-01
GLT5	mean	1.690E+00	1.691E+00	1.686E+00 †	1.590E+00 †	1.692E+00	1.667E+00 †	1.688E+00 †	1.648E+00 †	1.685E+00	1.680E+00 †	1.688E+00 †
	std	2.481E-03	1.445E-04	2.140E-04	2.471E-03	2.442E-04	3.610E-02	2.065E-03	6.947E-03	1.080E-02	2.248E-03	1.372E-03
GLT6	mean	1.686E+00	1.672E+00 †	1.680E+00 †	1.579E+00 †	1.684E+00	1.671E+00 †	1.675E+00 †	1.621E+00 †	1.678E+00 †	1.664E+00 †	1.673E+00 †
	std	1.281E-03	3.448E-02	1.167E-03	6.630E-02	1.166E-03	2.059E-02	1.064E-03	1.140E-02	3.506E-03	1.585E-03	1.826E-03

Wilcoxon's rank sum test at a 0.05 significance level is performed between CDG-MOEA and the other competing algorithms. † and ‡ denotes that the performance of the corresponding algorithm is significantly worse than or better than that of CDG-MOEA, respectively. The best mean is highlighted in boldface.

(extremely convex PFs or with disparately scaled objectives).

Table III shows the performance of the eleven compared algorithms in terms of IGD and I_H on GLT1-6. It can be observed that CDG-MOEA performs significantly better than other compared algorithms on all test problems except for GLT1 in terms of IGD . MOEA/D-ACD achieves the best performance on GLT1, although the final nondominated solutions obtained by MOEA/D-ACD and CDG-MOEA are both uniformly distributed on the PF of GLT1, as shown in Fig. 5.

In terms of I_H , CDG-MOEA has the best performance on GLT2-3 and MOEA/D-DE (TCH) has the best performance on GLT1 with statistical significance. The performance of CDG-MOEA is very similar to that of MOEA/D-DE (TCH) on GLT4; and the performance of CDG-MOEA is very similar to that of MOEA/D-ACD on GLT5-6, although it can be observed in Fig. 7 that the solution set obtained by CDG-MOEA is more uniformly distributed than that obtained by MOEA/D-DE (TCH) or MOEA/D-ACD on GLT6. This can be explained by the fact that the solutions on the extremely convex regions of PFs have very little contribution on the value of I_H .

C. Convergence Plots

The performance of eleven compared algorithms (CDG-MOEA, MOEA/D-DE (WS, TCH, PBI), MOEA/D-ACD,

MSOPS-II, NSGA-II, IBEA, ϵ -MOEA, Borg and OMOPSO) during the evolutionary process, in terms of the average IGD values over 30 runs, is illustrated in Fig. 8 on UF and GLT problems. It can be observed that CDG-MOEA has the best performance in terms of both the convergence speed and the quality of the final nondominated sets on UF2, GLT2-3 and GLT5. On UF6, CDG-MOEA converges more slowly at the early stage, but it performs increasingly better and outperforms all the compared algorithms at the final stage. It is worth noting that the performance of MOEA/D-DE (PBI), in terms of IGD values, becomes even increasingly worse during the optimization process on GLT2-3. This phenomenon can be verified by the final nondominated sets obtain by MOEA/D-DE (PBI) on GLT2 (Fig. 6 (d)), where only half of the PF can be approximated by MOEA/D-DE (PBI) on GLT2.

VII. SENSITIVITY ANALYSIS

In this section, we investigate the sensitivity of the control parameters in CDG-MOEA, including the population size N , the division parameter K , the grid neighborhood distance T and the probability δ for the mating restriction, and the control parameters F and CR in the DE reproduction operator, on both bi- and tri-objective UF and GLT test problems.

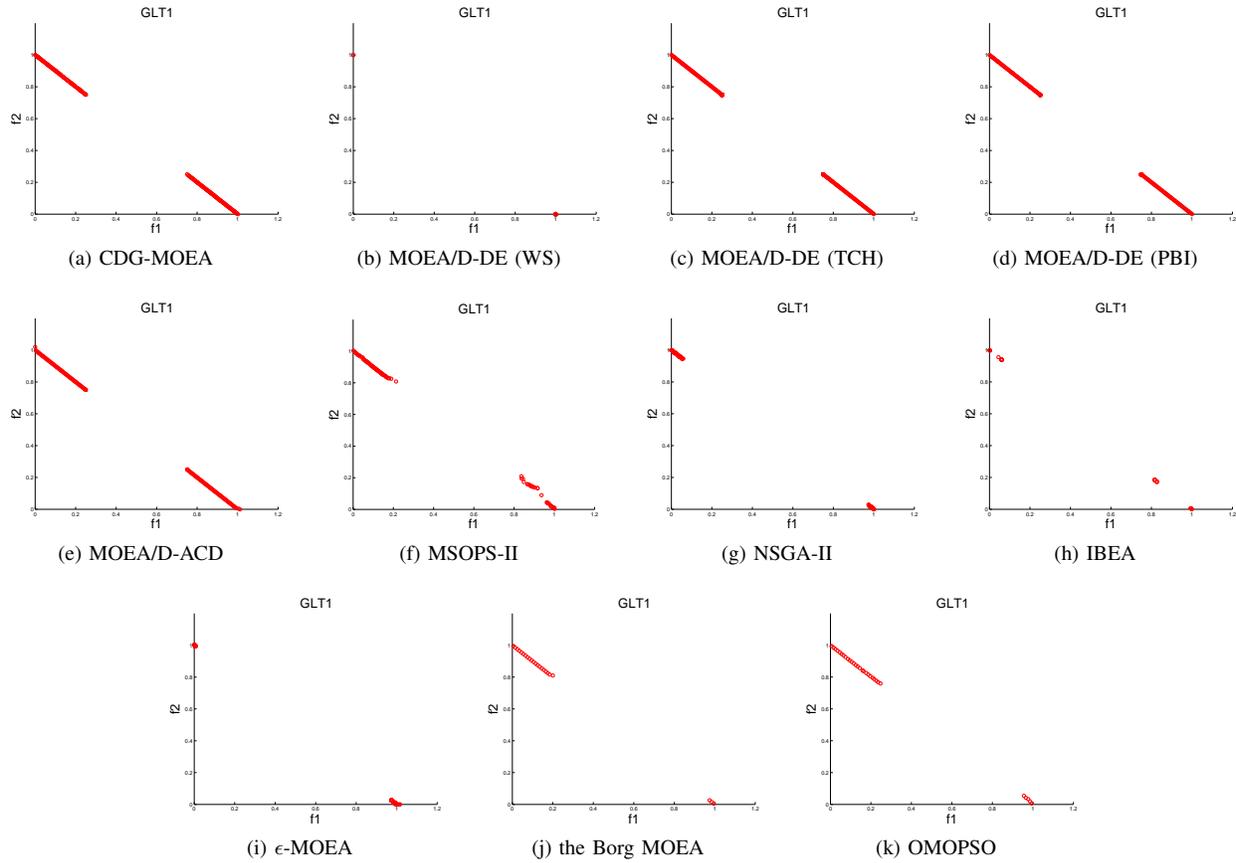


Fig. 5: The final nondominated solution set obtained by eleven algorithms in the run with the median IGD value on GLT1.

A. Sensitivity to Population Size N and Division Parameter K

For CDG, as described in Section III, there are a total number of mK^{m-1} subproblems and each solution can be at most the optimal solution for m subproblems, where m is the number of objectives. Although there is a complex interdependence between the population size N and the division parameter K , their relation for a MOP with a non-degenerate PF, can be analyzed as follows.

$$N = \frac{\alpha m K^{m-1}}{\beta} = \theta K^{m-1} \quad (16)$$

where α is the optimal average number of solutions for a subproblem; β is a coefficient depending on the shape of PF (e.g. the convexity or disconnection) and θ is the final coefficient after the simplification.

As the population size N and the division parameter K play an interdependent role on the performance of CDG-MOEA, we set combinations of different values of N (200-500 for bi-objective UF1 and GLT3; 300-1000 for tri-objective UF9 and GLT5) and K (100-260 for bi-objective UF1 and GLT3; 10-50 for tri-objective UF9 and GLT5) in the experiments. The other parameters are the same as those in Section V-B. Fig. 9 shows the mean IGD values of the populations obtained by CDG-MOEA with different population sizes N and division parameters K on UF1, UF9, GLT3 and GLT5.

Eq. 16 can be verified from Fig. 9 that N and K are

positively correlated for achieving the best performance for CDG-MOEA on both bi- and tri-objective problems. When $N = \theta K^{m-1}$, CDG-MOEA can achieve the best performance.

B. Sensitivity to Division Parameter K and Grid Neighborhood Distance T

As the division parameter K and the grid neighborhood distance T for the mating restriction also play an interdependent role on the performance of CDG-MOEA, we set combinations of different values of K (100-260 for bi-objective UF1 and GLT3; 10-50 for tri-objective UF9 and GLT5) and T (1-10 for bi-objective UF1 and GLT3; 1-5 for tri-objective UF9 and GLT5) in the experiments. The population size is set to 600 for UF1, 9 and 300 for GLT3, 5. The other parameters are the same as those in Section V-B. Fig. 10 shows the mean IGD values of the populations obtained by CDG-MOEA with different division parameters K and grid neighborhood distance T on UF1, UF9, GLT3 and GLT5.

Fig. 10 shows that CDG-MOEA with different K and T values may have the different performance. For UF problems, it can be observed that when the value of K increases, the value of T also needs to increase for achieving better performance. For GLT problems, CDG-MOEA is very sensitive to K and a suitable K value is needed for optimal performance, as discussed in the last section. However, given a fixed K value, CDG-MOEA is robust with regard to T on GLT problems.

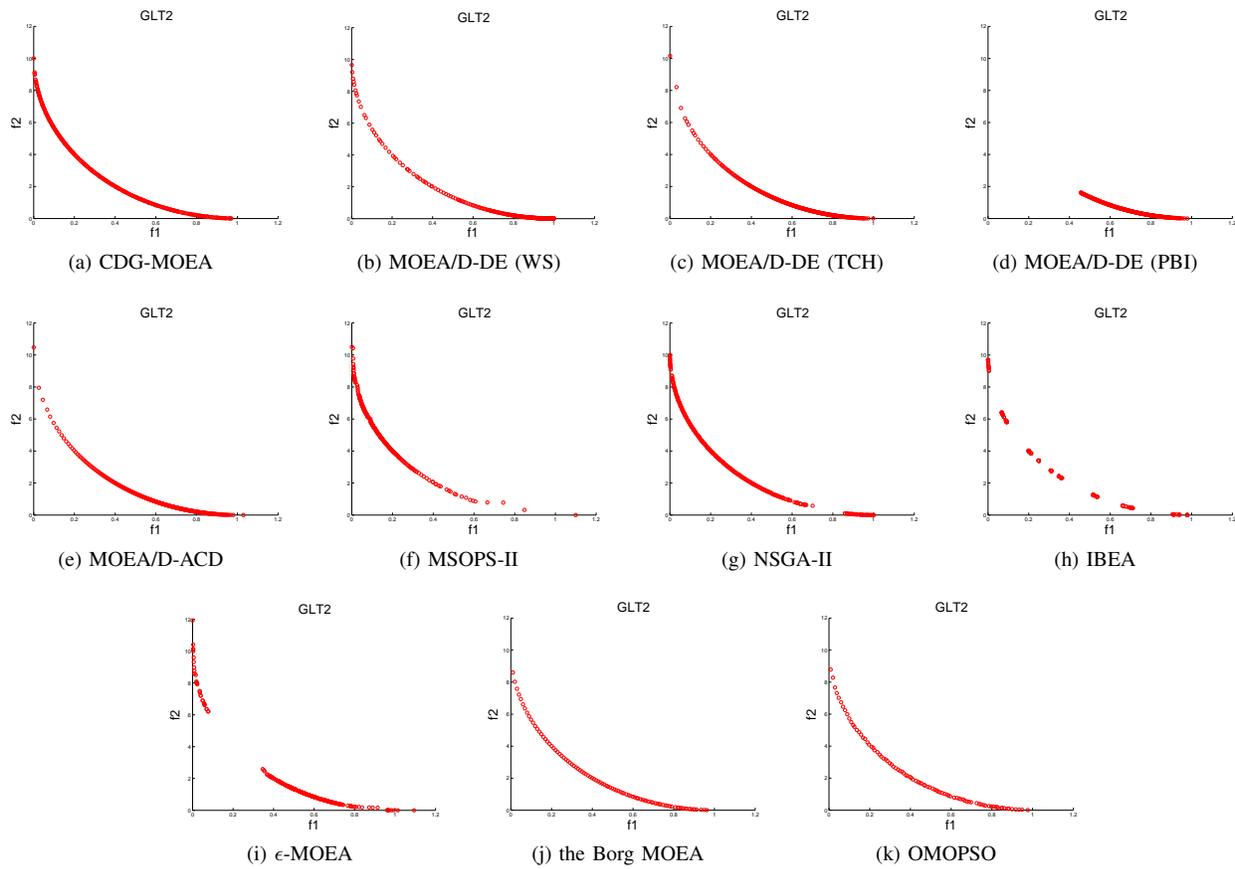


Fig. 6: The final nondominated solution set obtained by eleven algorithms in the run with the median IGD value on GLT2.

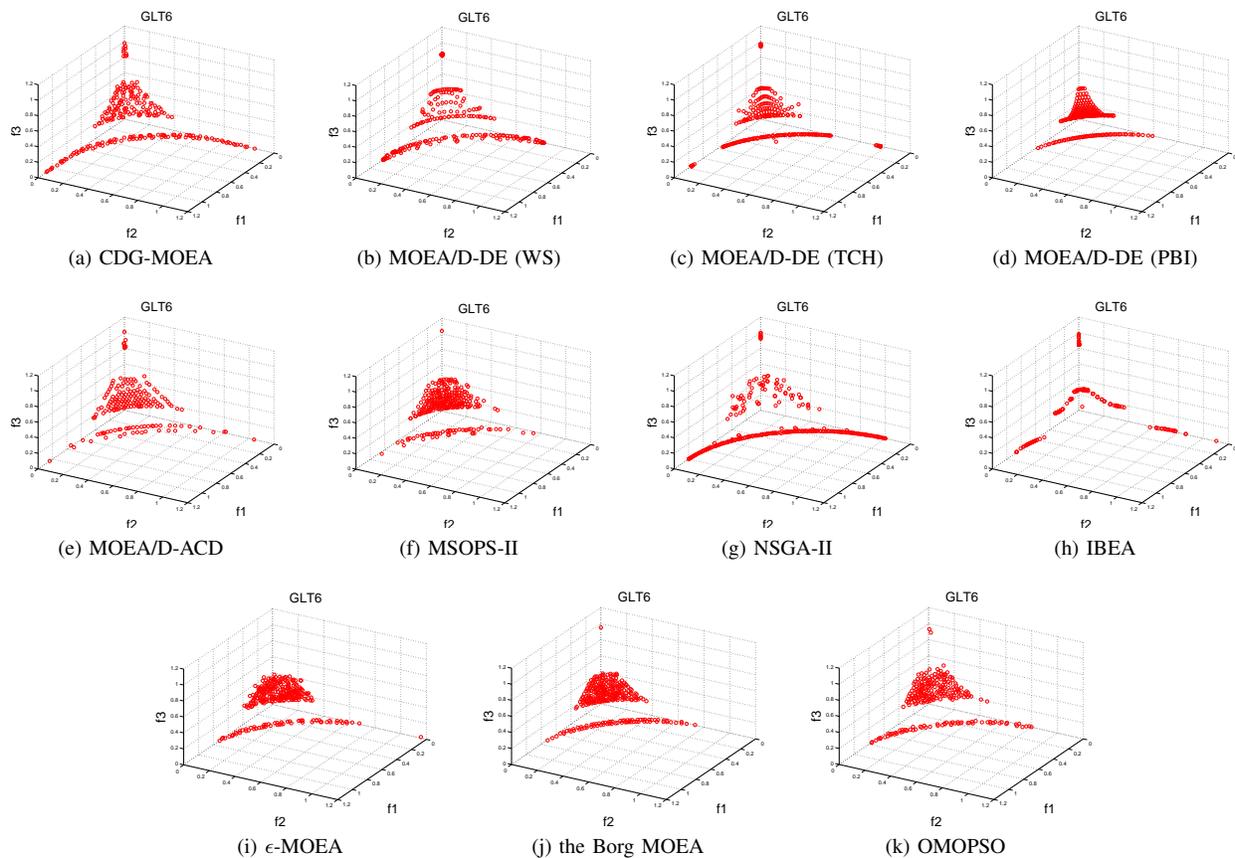


Fig. 7: The final nondominated solution set obtained by eleven algorithms in the run with the median IGD value on GLT6.

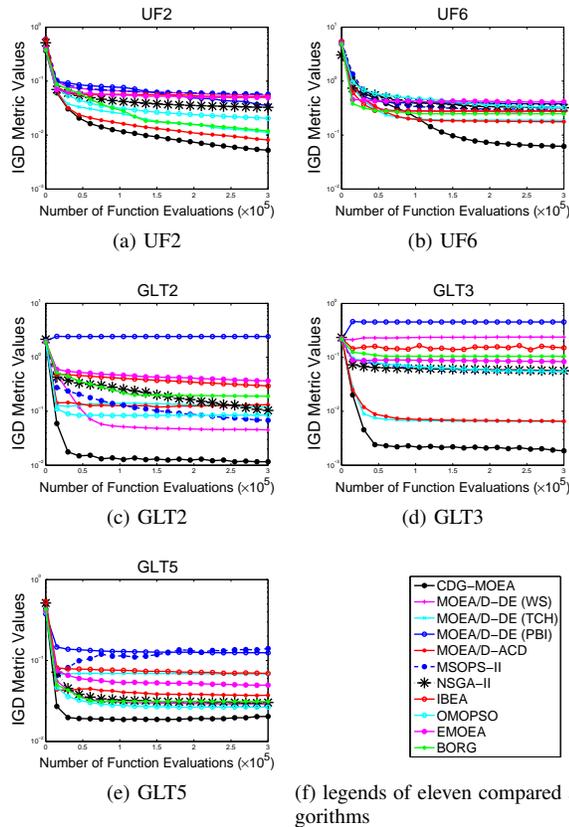


Fig. 8: The mean IGD values vs. the number of function evaluations obtained by eleven algorithms over 30 runs.

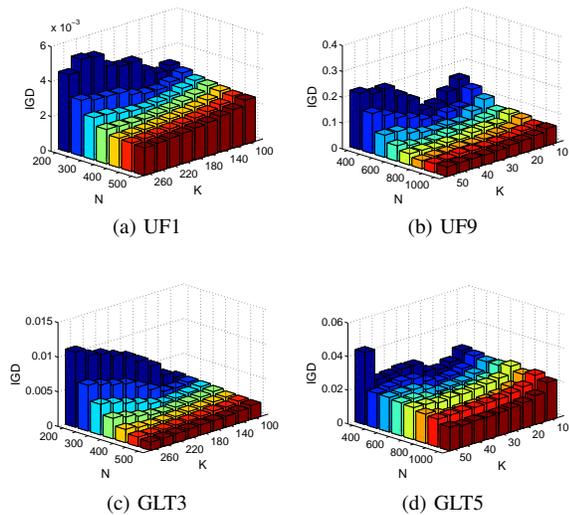


Fig. 9: The mean IGD values obtained by CDG-MOEA with different combinations of N and K on UF1, UF9, GLT3 and GLT5 over 30 runs.

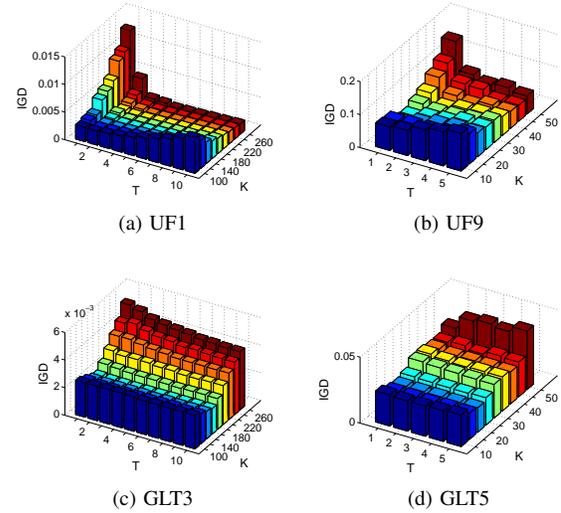


Fig. 10: The mean IGD values obtained by CDG-MOEA with different combinations of K and T on UF1, UF9, GLT3 and GLT5 over 30 runs.

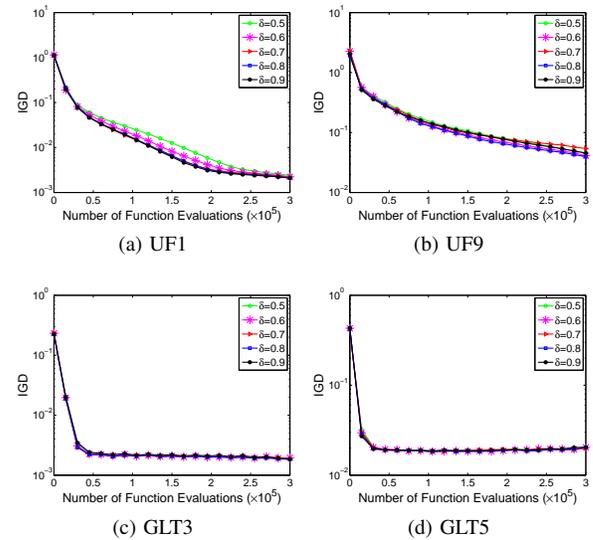


Fig. 11: The mean IGD values vs. the numbers of function evaluations obtained by CDG-MOEA with different δ over 30 runs on UF1, UF9, GLT3 and GLT5.

C. Sensitivity to Probability of Mating Restriction

To investigate the sensitivity of the probability for the mating restriction δ , CDG-MOEA with $\delta = 0.5, 0.6, 0.7, 0.8$ and 0.9 is separately tested. The remaining parameters are the same as those in Section V-B. Fig. 11 shows the evolution of the mean IGD values of the populations obtained by CDG-MOEA with different δ values on bi-objective UF1, GLT3 and tri-objective UF9, GLT5. It can be observed that CDG-MOEA with different δ achieves very similar performance, in terms of the final IGD values, which indicates that CDG-MOEA is not sensitive to δ . However, on UF1 and UF9, CDG-MOEA converges faster with larger δ values ($\delta = 0.8, 0.9$).

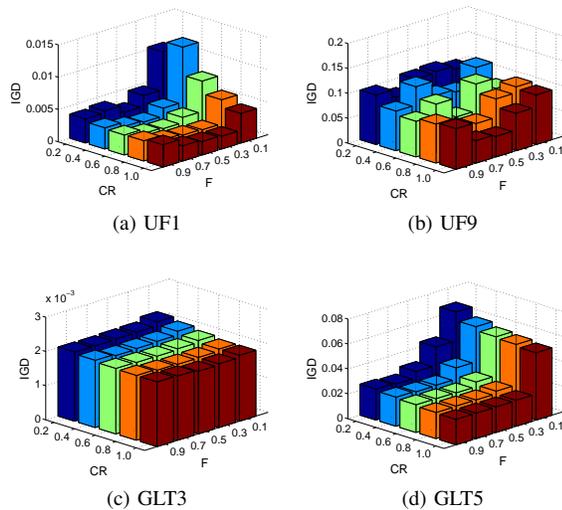


Fig. 12: The mean IGD values obtained by CDG-MOEA with different combinations of F and CR on UF1, UF9, GLT3 and GLT5 over 30 runs.

D. Sensitivity to F and CR

The sensitivity of the control parameters F and CR in the DE reproduction operator are tested in this section. We set a total of 25 combinations of five F values (0.1, 0.3, 0.5, 0.7 and 0.9) and five CR values (0.2, 0.4, 0.6, 0.8 and 1) on UF1, UF9, GLT3 and GLT5. All the other parameters remain the same as those in Section V-B. The mean IGD values of final populations obtained by CDG-MOEA with different combinations of F and CR on UF1, UF9 and GLT5 are shown in Fig. 12. It can be observed that when both F and CR have relatively large values (e.g., $F = 0.5, 0.7$ and $CR = 0.8, 1$), CDG-MOEA has better performance.

VIII. MORE DISCUSSIONS OF CDG-MOEA ON MANY-OBJECTIVE OPTIMIZATION

CDG-MOEA can be further extended to many-objective optimization problems if the following two issues can be well-addressed.

First, the setup of the grid-system in CDG is based on the approximation of both ideal and nadir point. In the current CDG-MOEA, Pareto-domination is used to approximate the nadir point. However, it is well-known that the selection pressure of Pareto-domination becomes weaker with the increasing number of objectives [22], [10], [25]. Therefore, to maintain the convergence for CDG-MOEA/D, other methods are needed for the better approximation of the nadir point.

Second, as discussed in Section VII-A, when $N = \theta K^{m-1}$, where θ is a coefficient, N is the population size, K is the grid division parameter and m is the number of objectives, CDG-MOEA can achieve the best performance. This indicates that the value of K (at least 2) would be far away from its optimal value (less than 1) when N is a small number with a large m value. In other words, a large K value and a very large N value are needed for CDG-MOEA on a many-objective optimization problem, to achieve good performance.

IX. CONCLUSION

This paper proposed a novel constrained decomposition approach with grids (CDG) to better fit the decomposition-based MOEA framework. The proposed CDG-MOEA is compared with seven classical or state-of-the-art MOEAs on two sets of test suites. The experimental results show that CDG-MOEA outperforms the compared algorithms in most test problems. More importantly, CDG-MOEA is very robust with the shapes of PFs and can remain effective on MOPs with complex PFs (e.g., extremely convex or with disparately scaled objectives). The sensitivity analysis of the parameters are also conducted in this paper. Further studies include the extension of CDG-MOEA for combinatorial multiobjective optimization problems and many-objective optimization problems.

REFERENCES

- [1] M. Asafuddoula, T. Ray, and R. A. Sarker, "A decomposition-based evolutionary algorithm for many objective optimization," *IEEE Transaction on Evolutionary Computation*, vol. 19, no. 3, pp. 445–460, 2015.
- [2] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *Evolutionary Computation*, vol. 19, no. 1, pp. 45–76, 2011.
- [3] M. Basseur and E. Zitzler, "Handling uncertainty in indicator-based multiobjective optimization," *International Journal of Computational Intelligence Research*, vol. 2, no. 3, pp. 255–272, 2006.
- [4] N. Beume, B. Naujoks, and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653–1669, 2007.
- [5] X. Cai, Y. Li, Z. Fan, and Q. Zhang, "An external archive guided multiobjective evolutionary algorithm based on decomposition for combinatorial optimization," *IEEE Transaction on Evolutionary Computation*, vol. 19, no. 4, pp. 508–523, 2015.
- [6] X. Cai, Z. Yang, Z. Fan, and Q. Zhang, "Decomposition-based-sorting and angle-based-selection for evolutionary multiobjective and many-objective optimization," *IEEE Transactions on Cybernetics*, 2016.
- [7] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga, "Handling multiple objectives with particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 256–279, 2004.
- [8] I. Das and J. E. Dennis, "Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems," *SIAM Journal on Optimization*, vol. 8, no. 3, pp. 631–657, 1998.
- [9] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*. Chichester, UK: John Wiley & Sons, 2001.
- [10] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints," *Evolutionary Computation, IEEE Transactions on*, vol. 18, no. 4, pp. 577–601, 2010.
- [11] K. Deb, M. Mohan, and S. Mishra, "A fast multi-objective evolutionary algorithm for finding well-spread pareto-optimal solutions," *KanGAL report*, vol. 2003002, pp. 1–18, 2003.
- [12] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [13] J. J. Durillo, A. J. Nebro, C. A. C. Coello, F. Luna, and E. Alba, "A comparative study of the effect of parameter scalability in multi-objective metaheuristics," in *Evolutionary Computation, 2008. CEC 2008. IEEE Congress on*. IEEE, 2008, pp. 1893–1900.
- [14] S. B. Gee, K. C. Tan, V. A. Shim, and N. R. Pal, "Online diversity assessment in evolutionary multiobjective optimization: A geometrical perspective," *IEEE Transaction on Evolutionary Computation*, vol. 19, no. 4, pp. 542–559, 2015.
- [15] F. Gu, H.-L. Liu, and K. C. Tan, "A multiobjective evolutionary algorithm using dynamic weight design method," *International Journal of Innovative Computing, Information and Control*, vol. 8, no. 5B, pp. 3677–3688, 2012.
- [16] D. Hadka and P. Reed, "Borg: An auto-adaptive many-objective evolutionary computing framework," *Evolutionary Computation*, vol. 21, no. 2, pp. 231–259, 2013.

- [17] Y. Y. Haimes, L. S. Lasdon, and D. A. Wismer, "On a bicriterion formulation of the problems of integrated system identification and system optimization," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 1, pp. 296–297, 1971.
- [18] E. J. Hughes, "Multiple single objective pareto sampling," in *Evolutionary Computation, 2003. CEC'03. The 2003 Congress on*, vol. 4. IEEE, 2003, pp. 2678–2684.
- [19] —, "MSOPS-II: A general-purpose many-objective optimiser," in *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*. IEEE, 2007, pp. 3944–3951.
- [20] C. Igel, N. Hansen, and S. Roth, "Covariance matrix adaptation for multi-objective optimization," *Evolutionary Computation*, vol. 15, no. 1, pp. 1–28, 2007.
- [21] H. Ishibuchi, Y. Sakane, N. Tsukamoto, and Y. Nojima, "Adaptation of scalarizing functions in MOEA/D: An adaptive scalarizing function-based multiobjective evolutionary algorithm," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2009, pp. 438–452.
- [22] H. Ishibuchi, N. Tsukamoto, and Y. Nojima, "Evolutionary many-objective optimization: A short review," in *2008 Congress on Evolutionary Computation (CEC'2008)*. Hong Kong: IEEE Service Center, June 2008, pp. 2424–2431.
- [23] J. D. Knowles and D. W. Corne, "Approximating the nondominated front using the Pareto archived evolution strategy," *Evolutionary Computation*, vol. 8, no. 2, pp. 149–172, 2000.
- [24] 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.
- [25] B. Li, J. Li, K. Tang, and X. Yao, "Many-objective evolutionary algorithms: A survey," *ACM Computing Surveys (CSUR)*, vol. 48, no. 1, p. 13, 2015.
- [26] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 284–302, 2009.
- [27] K. Li, Q. Zhang, S. Kwong, M. Li, and R. Wang, "Stable matching based selection in evolutionary multiobjective optimization," *IEEE Transaction on Evolutionary Computation*, vol. 18, pp. 909–923, 2014.
- [28] M. Li, S. Yang, and X. Liu, "Shift-based density estimation for pareto-based algorithms in many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 3, pp. 348–365, 2014.
- [29] H. Liu, F. Gu, and Q. Zhang, "Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems," *IEEE Transaction Evolutionary Computation*, vol. 18, no. 3, pp. 450–455, 2014.
- [30] K. Miettinen, *Nonlinear Multiobjective Optimization*. Boston: Kluwer Academic Publishers, 1999.
- [31] T. Murata, H. Ishibuchi, and M. Gen, "Specification of genetic search directions in cellular multi-objective genetic algorithms," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2001, pp. 82–95.
- [32] A. J. Nebro, J. J. Durillo, F. Luna, B. Dorronsoro, and E. Alba, "Mocell: A cellular genetic algorithm for multiobjective optimization," *International Journal of Intelligent Systems*, vol. 24, no. 7, pp. 726–746, 2009.
- [33] H. Sato, "Inverted PBI in MOEA/D and its impact on the search performance on multi and many-objective optimization," in *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*. ACM, 2014, pp. 645–652.
- [34] J. D. Schaffer and J. J. Grefenstette, "Multi-objective learning via genetic algorithms," in *Ijcai*, vol. 85, 1985, pp. 593–595.
- [35] M. R. Sierra and C. A. C. Coello, "Improving pso-based multi-objective optimization using crowding, mutation and ϵ -dominance," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2005, pp. 505–519.
- [36] H. Trautmann, T. Wagner, and D. Brockhoff, "R2-EMOA: Focused multiobjective search using R2-indicator-based selection," in *International Conference on Learning and Intelligent Optimization*. Springer, 2013, pp. 70–74.
- [37] L. Wang, Q. Zhang, A. Zhou, M. Gong, and L. Jiao, "Constrained subproblems in a decomposition-based multiobjective evolutionary algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 3, pp. 475–480, 2016.
- [38] Z. Wang, Q. Zhang, A. Zhou, M. Gong, and L. Jiao, "Adaptive replacement strategies for MOEA/D," *IEEE Transactions on Cybernetics*, vol. 46, no. 2, pp. 474–486, 2016.
- [39] S. Yang, M. Li, X. Liu, and J. Zheng, "A grid-based evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 5, pp. 721–736, 2013.
- [40] G. G. Yen and H. Lu, "Dynamic multiobjective evolutionary algorithm: adaptive cell-based rank and density estimation," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 3, pp. 253–274, 2003.
- [41] H. Zhang, A. Zhou, S. Song, Q. Zhang, X.-Z. Gao, and J. Zhang, "A self-organizing multiobjective evolutionary algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 5, pp. 792–806, 2016.
- [42] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.
- [43] Q. Zhang, H. Li, D. Maringer, and E. Tsang, "MOEA/D with NBI-style Tchebycheff approach for portfolio management," in *Evolutionary Computation (CEC), 2010 IEEE Congress on*. IEEE, 2010, pp. 1–8.
- [44] Q. Zhang, A. Zhou, and Y. Jin, "RM-MEDA: A regularity model-based multiobjective estimation of distribution algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 1, pp. 41–63, 2008.
- [45] Q. Zhang, A. Zhou, S. Zhao, P. N. 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, special session on performance assessment of multi-objective optimization algorithms, technical report*, vol. 264, 2008.
- [46] A. Zhou, Q. Zhang, Y. Jin, and E. Tsang, "A model-based evolutionary algorithm for bi-objective optimization," in *Evolutionary Computation, 2005. The 2005 IEEE Congress on*, 2005, pp. 2568–2575 Vol. 3.
- [47] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2004, pp. 832–842.
- [48] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm," 2001.
- [49] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.

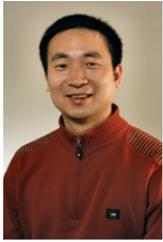


Xinye Cai received the BSc in information engineering from Huazhong University of Science and Technology, China in 2004, the MSc in electronic engineering from University of York, UK, in 2006 and the PhD in electrical engineering from Kansas State University, U.S, in 2009, respectively.

He is an Associate Professor at the Department of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics (NUAA), China. His current research interests include evolutionary computation, optimization, data mining and their applications. He is currently leading an Intelligent Optimization Research Group in NUAA, who has won the Evolutionary Many-Objective Optimization Competition at the Congress of Evolutionary Computation 2017.



Zhiwei Mei received his Master degree in Computer Science at the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China, in 2017. Currently, he is working as an engineer in MeiTuan. His research interests include evolutionary computation and multiobjective optimization.



Zhun Fan is the head of the Department of Electronic Engineering and Full Professor at Shantou University. Prior to joining Shantou University, he was an Assistant Professor, and then an Associate Professor at the Technical University of Denmark. He has also been working for the BEACON Center for Study of Evolution in Action at Michigan State University.

Dr. Zhun Fan received his Ph.D. (Electrical and Computer Engineering) in 2004 from the Michigan State University. He received the B.S. degree (Control Engineering) in 1995 and M.S degree (Control Engineering) in 2000 from Huazhong University of Science and Technology, China. His major research interests include intelligent control and robotic systems, robot vision and cognition, MEMS, computational intelligence, design automation and optimization of mechatronic systems etc.



Qingfu Zhang (M01-SM06) received the BSc in mathematics from Shanxi University, China in 1984, the MSc in applied mathematics and the PhD in information engineering from Xidian University, China, in 1991 and 1994, respectively.

He is a Professor at the Department of Computer Science, City University of Hong Kong, Hong Kong, a Professor on leave from the School of Computer Science and Electronic Engineering, University of Essex, UK, and a Changjiang Visiting Chair Professor in Xidian University, China. He holds two patents and is the author of many research publications. His main research interests include evolutionary computation, optimization, neural networks, data analysis, and their applications. He is currently leading the Metaheuristic Optimization Research (MOP) Group in City University of Hong Kong.

Dr. Zhang is an Associate Editor of the IEEE Transactions on Evolutionary Computation and the IEEE Transactions on Systems, Man, and Cybernetics-Part B. He is also an Editorial Board Member of three other international journals. MOEA/D, a multiobjective optimization algorithm developed in his group, won the Unconstrained Multiobjective Optimization Algorithm Competition at the Congress of Evolutionary Computation 2009, and was awarded the 2010 IEEE Transactions on Evolutionary Computation Outstanding Paper Award.