A Many-objective Evolutionary Algorithm Based on Rotated Grid

Juan Zou[a,e], Liuwei Fu[a,*], Jinhua Zheng[a,b,*], Shengxiang Yang[a,c], Guo Yu[a,d], Yaru Hu[a]

[a]*Key Laboratory of Intelligent Computing and Information Processing, Ministry of Education, Information Engineering College of Xiangtan University, Xiangtan, Hunan Province, China*
[b]*Ministry of Education and School of Computer Science and Technology Hengyang Normal University, HengYang, Hunan Province, China*
[c]*School of Computer Science and Informatics, De Montfort University, Leicester LE1 9BH, U.K.*
[d]*School of Computer Science , University of Surrey, Guidford, Surrey,Gu2 7XH, U.K.*
[e]*LED Lighting Research Technology Center of Guizhou TongRen, GuiZhou, China*

Abstract

Evolutionary optimization algorithms, a meta-heuristic approach, often encounter considerable challenges in many-objective optimization problems (MaOPs). The Pareto-based dominance loses its effectiveness in MaOPs, which are defined as having more than three objectives. Therefore, a more valid selection method is proposed to balance convergence and distribution. This paper proposes an algorithm using rotary grid technology to solve MaOPs (denoted by RGridEA). The algorithm uses the rotating grid to partition the objective space. Instead of using the Pareto non-dominated sorting strategy to layer the population a novel stratified method is used to enhance convergence effectively and make use of the grid to improve distribution and uniformity. Finally, with the other seven algorithm was tested on the test function DTLZ series analysis, confirming RGridEA is effective in resolving MaOPs.

Keywords: Many-objective optimization; evolutionary algorithms; clustering; genetic algorithms; Multi-objective optimization

---

*Corresponding author: Liuwei Fu, Jinhua Zheng
Email addresses: `fuliuwei1@gmail.com` (Liuwei Fu), `jhzheng@xtu.edu.cn` (Jinhua Zheng)

1. Introduction

Many optimization problems in the real world are usually involved in many objectives. Generally, a MaOPs can be formulated as:

$$\min_{x \in D} F(x) = (f_1(x), f_2(x), \ldots, f_M(x))^T \tag{1}$$

where $D \subseteq \mathbb{R}^n$ is the decision space, $x = (x_1, x_2, \ldots, x_n) \in D$ is the decision variable and $M \geq 2$ is the number of the objectives. For MaOPs , M is generally greater than three [1]. Examples of many-objective optimization problems include: time table problem [2] [8], radar optimization problem [3], water resource optimization problem [4], ground water monitoring problem [5], air traffic control problem [6], wing design problem [7], gearbox design problem [8], storm drainage system problem [9], vehicle design problem [10] and vehicle crash safety problem [11]. Most of those problems are NP-hard problems and many-objective optimization problems [2]. Due to high complexity and non-linearity, those problems are difficult to be solved by traditional optimization methods.

Multi-objective evolutionary algorithms (MOEAs) have characteristics of global random search and ability of dealing with highly-complex nonlinear problems. Currently, it has been proved that the multi-objective problems with 2 and 3 objectives can be solved well by traditional MOEAs, however they are less effective and less efficient to cope with MaOPs. The main reason why MOEAs can deal with multi-objective problems well is that most MOEAs use the Pareto dominance relationship as the primary method to distinguish the mutual relationship between the individuals, which defines a partially ordered relation to sort all of the individuals [11] [12] so as to prompt convergence. Meanwhile, MOEAs also utilize distribution information as the secondary method to evaluate the fitness of individuals. Thus, traditional evolutionary multi-objective (EMO) algorithms can ensure convergence and obtain good distribution as far as possible. However, with the increase of number of objectives, the non-dominated solutions increase exponentially [11] [13], thereby, the fitness based Pareto dom-

inance relationship will be difficult to distinguish the mutual relationship between the individuals, which could lead to greatly weaken the searching ability of the algorithms [14] [15] [16] [17].

In order to overcome these difficulties many evolutionary algorithms have been proposed to deal with MaOPs, and they can be divided into three classes.

- **Based on traditional Pareto dominance relationship.** Due to the Pareto relationship failing to prompt the convergence pressure in solving MaOPs, many efforts have been put into relaxing the Pareto dominance relationship. Drechsler et al. [18] put forward the winning relationship [40]-[43] method to determine the priority of the individuals in the non-dominated solution set. To some extent, the method has reduced the strength of the pareto dominance relationship, but it has no transitivity in the solution set. Ikeda et al. [19] put forward the α-dominance, which is designed to strengthen or weaken the Pareto dominance relationship by adjusting the α parameter, but it is difficult to find a suitable α in the optimization. Laumanns et al. [20] proposed the ϵ-dominance relationship. Although this relationship can enhance the selection pressure and maintain the distribution of the solution set to some extent, but it is difficult to determine appropriate parameters to various problems. In addition, David Hadka et al. [21] put forward a diagnostic evaluation framework which can assess the effectiveness, reliability, efficiency and controllability of MOEA. Salem et al. [22] put forward two kinds of diversity maintaining mechanisms and investigated their influence on algorithm convergence.

- **Based on Non-Pareto dominance relationship.** Non-Pareto methods mainly include indicators- or index-based methods and the methods based on decomposition [53]. Zitzler and Künzli [22] put forward the indicator-based evolutionary algorithms, namely, IBEA. Then several versions of improved IBEAs came out [23]. Literature [24] points out that the convergence of IBEA is better than that of the MOEAs based on the Pareto dominance relationship in solving the MaOPs with 3 to 6 ob-

3

jectives, but the computational cost is too much and selecting reference points is difficult. Zhang and Li [26] put forward the MOEA/D, and Hughes [27] proposed the MSOPS. Both of these divide a MaOP into many sub-problems and then optimize the sub-problems simultaneously. They can solve the MaOPs,but need prior knowledge well.

- **Dimension reduction.** In order to reduce complexity and redundant objectives for solving MaOPs, Deb et al. [28] [29] [30] applied the Principal Component Analysis (PCA) method to MOEAs and achieved good results. In addition, some scholars put forward the feature selection method [32] and subset covering method [31] [34] to reduce redundant objectives. These kind of methods have two shortcomings. One is the loss of some important information after the reduction, and the other is the setting of parameters, increasing the complexity of the problem.

Although these three classes of methods have provided new ideas for solving MaOPs, great improvements are still needed before EMO algorithms can be considered to be as effective for solving many-objective problems as they are for 2- and 3-objective problems. Existing algorithms, such as ϵ-MOEA, that have achieved good performance in solving MaOPs still have significant drawbacks like the difficulty in parameter setting. As highlighted by Purshouseet et al. [33] Research into evolutionary many-objective optimizations still in its infancy, and the need for efficient methodologies is pressing.

Thus, this paper proposes an algorithm using rotary grid technology to solve MaOPs (denoted by RGridEA). The algorithm uses the rotating grid to partition the objective space. It no longer uses the Pareto non-dominated sorting strategy to layer the population but proposes a novel stratified method so as to enhance the convergence effectively and use of the grid to improve distribution and uniformity.

4

2. Motivation

For EMO problems [1], with the increase of the number of objectives, the Pareto dominance relationship tends to be weakened in the optimization. The reason is that with the increasing of the number of objectives, the Pareto dominance relationship is invalid since most individuals are mutually non-dominated, thereby reducing selection pressure and search ability [14] [17]. Purshouse et al. [61] point out when the number of objectives increases to 4 or more, the performance of the EMO algorithms based on Pareto dominance relationship greatly decrease. Hughes [35] has shown that MOEAs based on Pareto dominance ranking are very effective in solving problems with few goals (2 or 3). However, their performance will be worse than that of the non-Pareto dominance based methods in dealing with the MaOPs [57] [58]. In addition, some recent research shows that when the number of objectives increases to 10 or more, the MOEAs based on Pareto dominance perform even worse than the random search based algorithms [36]-[38].

As shown in Figure 1(a), convergence and diversity can be controlled through adjusting the value of the angle which control the dominated region in the optimization. Sato et al. [62] put a similar idea into the frame of NSGA-II [44], which enhances performance in many-objective optimization. In Figure 1(a), individuals (A, B, C) are non-dominated in the Pareto dominance relationship (the region above the black solid line indicates the dominated area of a solution). After relaxing the dominance relationship, the region above the red dotted line indicates the dominated area of a solution so that individuals A and C are dominated by C in this situation.

In Figure 1(c), Laumanns et al. [38] put forward the ϵ-dominance relationship. Its main idea is to enlarge the dominated region by $1+ \epsilon$ times ($\epsilon > 0$). As shown in Figure 1(c), after the modification of the dominance relationship, the relationships between the four mutual non-dominated four points (A, B, C and D) in the Pareto dominance sense are changed, so that A and C are dominated by B, because A and C are included in the dominated region above the red

5
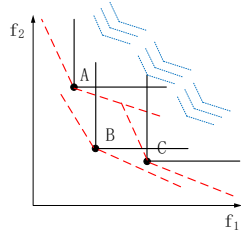
(a) Idea of angle-based dominance.
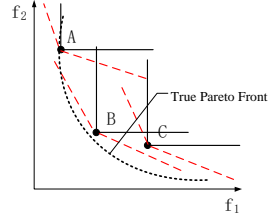
(b) The loss of boundary points in angle-based dominance.
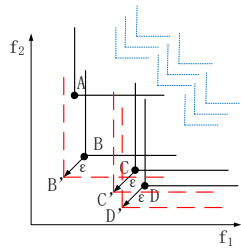
(c) Idea of ϵ-dominance.

(d) The loss of boundary points in ϵ-dominance.

(e) Idea of grid-based dominance.

(f) The loss of boundary points in grid-based dominance.

Figure 1: Illustration of different kinds of dominance relationships.

dashed line of B.

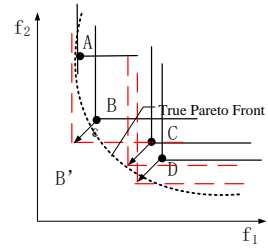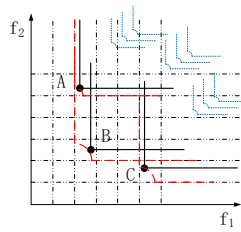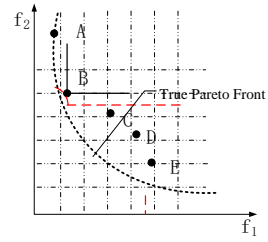As Figure 1(e) shows, S. Yang et al. [14] introduced the concept of grid-based dominance. By dividing the objective space into many small grids, then controlling the distribution of individuals in these small grids so as to enhance diversity. But the essence of the idea is still to relax the Pareto dominance relationship. In Figure 1(e), the points (A, B and C) are non-dominated in the Pareto dominance sense. After the amplification dominance relationship, B dominates A. In the same grid, the individual close to the left corner of the gird will be preferred in comparison with other individuals in the same grid. Therefore, the dominated region of a solution is changed much as shown in Figure 1(e).

These algorithms that relax the Pareto dominance relationship are able to solve MaOPs to some extent with some advantages. First, they have the characteristic of guiding the individuals to converge to the Pareto Objective Front(POF). As shown in Figure 1(a), 1(c), 1(e), the blue dashed lines show the direction of evolution or direction of convergence. Second, at the same time, these algorithms also keep the distribution [14].

On the other hand, these algorithms have some drawbacks. First, a common problem for the algorithms that relax the Pareto dominance relationship is that it is hard to control the degree of relaxation. For example, ϵ-MOEA has to adjust the parameter repeatedly to determine the best value for different problems. As from Table 1, it is specifically tests the influence of influence ε value in ϵ-MOEA algorithm , the experimental results has be great influenced by the value of ϵ. Second, relaxing the dominated relation will cause the missing of boundary individuals to a certain extent, as shown in Figure 1(b), 1(d), 1(f) that the boundary individual A is dominated by individual B in all situations. Furthermore it is known that the boundary individuals are very important in keeping the spread or diversity of the solutions in the evolutionary process. Third, some relaxation-based algorithms may destroy the partial order relation. For example, in Figure 1(c), the nondominated solutions C and D are in the same grid in the Pareto dominance sense. After the relaxation, $C^{'}$ can dominate

7

Table 1: The settings of the ϵ value in ϵ-MOEA.

Objective numberProblem	3	4	5	6	8	10
DTLZ1	0.033	0.06	0.06	0.06	0.0227	0.048
DTLZ2	0.052	0.1312	0.1385	0.1312	0.12	0.105
DTLZ3	0.059	0.1927	0.2	0.1927	0.3552	0.158
DTLZ4	0.0554	0.234	0.227	0.234	0.75	0.15
DTLZ6	0.0549	0.29	0.1567	0.29	1.15	0.225
DTLZ7	0.0565	0.308	0.85	0.308	1.45	0.46

D, and $D^{'}$ can dominate C. Thus, C and D are non-ϵ-dominated.

Therefore, this is a really crucial challenge to guide the the population evolving fast toward the optimal front while simultaneously maintaining the individuals's diversity during the evolutionary process. To handle these drawbacks and challenge, a many-objective evolutionary algorithm based on a rotation of grid (RGridEA) has put forward in this paper. On the one hand, the algorithm will adopt the idea of grids to maintain diversity. On the other hand, it will consider convergence and diversity separately and add the evolutionary direction to guide optimization. Although the RGridEA has a parameter that denote the number of grids, the parameter R can be adjusted dynamically by the size of objective space determined by the individuals, so the parameter R setting is relatively simple, and a constant value can be set for any problems with any dimensions, like R=10.

3. Rotation Grid based algorithm (RGridEA)

3.1. Rotation grid

Considering that the increase non-dominated individuals will result in the EMO algorithms hard-converging to the Pareto Front(PF) and that relaxing the Pareto dominance relations makes it difficult to determine the parameter and may cause the loss of boundary points. We use the grid partition method to keep the distribution. Then, the grid is redesigned and a rotating grid is proposed that redefines the stratification mechanism using the rotary grids.

8
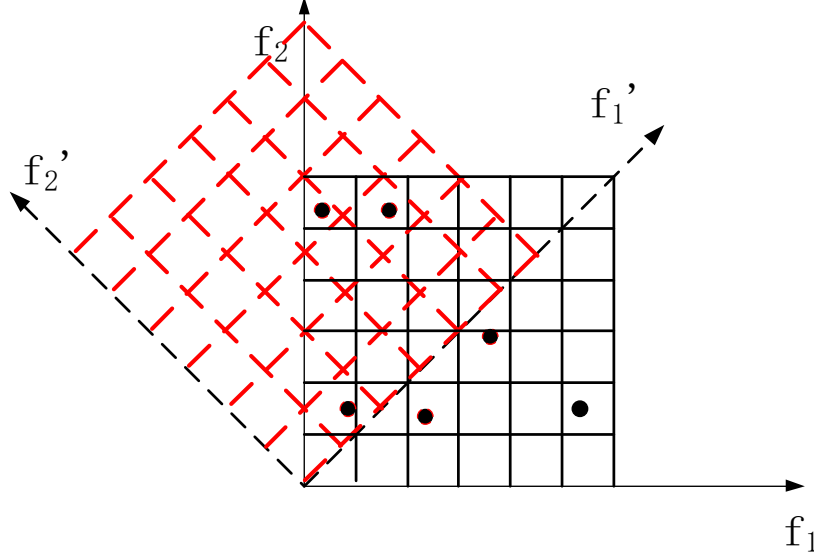
Figure 2: A two-dimensional example of rotating grid.

Definition 3.1 Rotating Grid: In n-dimensional objective space, the number of individuals is m. Then the largest objective value the M is obtained:

$$M = \max_{j=1}^{m}\{\max_{i=1}^{n}\{t_{ij}\}\} \tag{2}$$

where t_{ij} is the tth objective value of the jth individual. Then M is divided into R equal divisions and the length of each is M/R, namely, $a = M/R$.

We will divide each objective into R equal length with a, and construct a hyper-plane $(f_1, f_2, \cdots, f_{i-1}, f_{i+1}, \cdots, f_n)$ parallel to the coordinate system through $f_i=ka$, where $k = (1, 2, \cdots, R)$. After M turns, the objective space will be divided into $R \times R \times ... \times R$ hyper-grids.

The rotation grid is to rotate the original coordinate system and the grids together in 45° and make sure that one axis coincides with the vector $\vec{c}{=}(1,1,\cdots,1)$, which is called the rotating grids(RGrid).

Figure 2 shows a two-dimensional example of a rotating rigid. The black solid

9
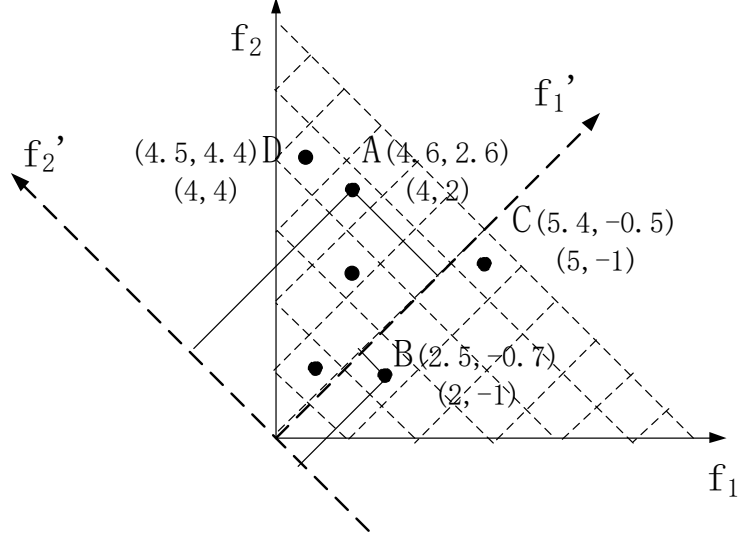
Figure 3: A two-dimensional example for the rotating grid coordinates.

lines represent the grids in the first quartile in a 2-D objective space, and the red dotted lines present the rotating grids.

Definition 3.2 rotating grid coordinates:

Algorithm 1 presents the framework of rotating grid coordinates. In the n-dimensional objective space by rotating method, we use the position of the rotating grid to represent the coordinate of the rotation grid. Given an individual $\vec{f}=(f_1, f_2, \cdots, f_n)^T$, its coordinate in the rotating coordinate system is $\vec{f'} = (f'_1, f'_2, \cdots, f'_n)^T$ in step 1. Then its rotation grid coordinate can be defined as follows:

$$\vec{\pi} = (\pi_1, \pi_2, \cdots, \pi_n) = (\lfloor f'_1/a \rfloor, \lfloor f'_2/a \rfloor, \cdots, \lfloor f'_n/a \rfloor)^T \tag{3}$$

In Figure 3, if $a = 1$, $f'_A = (4.6, 2.6)$, then $\pi_A = (\lfloor 4.6/1 \rfloor, \lfloor 2.6/1 \rfloor) = (4, 2)$. In the same way, $\pi_B = (2, -1), \pi_C = (5, -1), \pi_B = (4, 4)$.

Generally, in the n-dimensional objective space with m individuals, $MAX =$

10

$\underset{j=1}{\overset{m}{max}}\{\underset{i=1}{\overset{n}{max}}\{f_{ij}\}\}$ and $MIX = \underset{j=1}{\overset{m}{min}}\{\underset{i=1}{\overset{n}{min}}\{f_{ij}\}\}$ will be obtained, where f_{ij} is the jth objective value of the ith individual. Then it is to divide $\|MAX - MIN\|$ into R equal parts, and the length of each part is $\|MAX - MIN\|/R$, namely, $a = \|MAX - MIN\|/R$ in step 2.2. Thus, each coordinate can be divided into R equal parts, and a hyper-plane will be constructed by parallelling to the coordinate system $(f_1, f_2, \cdots, f_{i-1}, f_{i+1}, \cdots, f_n)$ using $f_i = ka$, where $k = (1, 2, \cdots, R)$. Therefore, the objective space is divided into $R \times R \times ... \times R$ hypercube grids. After that, the orthogonal matrix $P = (\vec{p_1}, \vec{p_2}, \cdots, \vec{p_n})$ is constructed, where $\vec{p_1}, \vec{p_2}, \cdots, \vec{p_n}$ are pairwise orthogonal, and $\vec{p_i}$ represents the ith rotated coordinate with 45°, providing that the first coordinate axis is rotated to coincide with the unit vector $(1, 1, \cdots, 1)$.

In objective space, after the rotation, the individual $\vec{f} = (\vec{f_1}, \vec{f_2}, \cdots, \vec{f_n})^T$ is transferred to $\vec{f'} = (\vec{f'_1}, \vec{f'_2}, \cdots, \vec{f'_n})^T$ in the rotating coordinate system. Its rotation grid coordinates are $\vec{\pi} = (\vec{\pi_1}, \vec{\pi_2}, \cdots, \vec{\pi_n}) = (\lfloor f'_1/a \rfloor, \lfloor f'_2/a \rfloor, \cdots, \lfloor f'_n/a \rfloor)^T$. Obviously, $\vec{f'} = P^{-1} \cdot \vec{f}$, because $\vec{p_1}, \vec{p_2}, \cdots, \vec{p_n}$ are pairwise orthogonal, and $\vec{f'} = P^{-1} \cdot \vec{f} = P^T \cdot \vec{f}$ in step 2.3.

Algorithm 1 how to calculate the rotation grid coordinate of an individual.

1: **Input:** The number of objectives: n;

2:     Population size: m;

3:     Population $S = (\vec{s_1}, \vec{s_2}, \cdots, \vec{s_m})$;

4:     Convergence direction vector $\vec{c} = (1, 1, \cdots, 1)^T$;

5:     Number of grids in each dimension: R.

6: **Output:** Rotation grid coordinates of population S: $\{\vec{\pi_1}, \vec{\pi_2}, \cdots, \vec{\pi_n}, \}$, where $\vec{\pi_i}$ is a n-dimensional vector.

7:     The coordinate of population $S = \{\vec{s_1}, \vec{s_2}, \cdots, \vec{s_n}\}$ is $f = \{\vec{f_1}, \vec{f_2}, \cdots, \vec{f_m}\}$ after translation to the first quadrant.

8: **Step 1)** calculate rotating coordinate matrix through orthogonal matrix.

9: **Step 1.1)** set a coordinate matrix: $\vec{a_1}, \vec{a_2}, \cdots, \vec{a_n}$ are all linearly independent, and then

11

10: $A = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ldots & \vdots \\ a_{n1} & a_{n2} & \ldots & a_{nn} \end{pmatrix}$ where one $\vec{a_i}$ must be the identity matrix.

11: **Step 1.2)** Transfer the matrix A into the orthogonal matrix $B = \{\vec{b_1}, \vec{b_2}, \cdots, \vec{b_n}\}$.

$$\begin{cases} \vec{b_1} = \vec{a_1} \\ \vec{b_2} = \vec{a_2} - \frac{[\vec{b_1}, \vec{a_2}]}{[\vec{b_1}, \vec{b_1}]} \vec{b_1} \\ \vec{b_n} = \vec{a_n} - \frac{[\vec{b_1}, \vec{a_n}]}{[\vec{b_1}, \vec{b_1}]} \vec{b_1} - \frac{[\vec{b_2}, \vec{a_n}]}{[\vec{b_2}, \vec{b_2}]} \vec{b_2} - \cdots - \frac{[\vec{b_{n-1}}, \vec{a_n}]}{[\vec{b_{n-1}}, \vec{b_{n-1}}]} \vec{b_{n-1}} \end{cases} \quad (4)$$

12: **Step 1.3)** Unitize matrix B and get rotation coordinate matrix P.

$$P = (\vec{p_1}, \vec{p_2}, \cdots, \vec{p_n}) = \frac{1}{\|\vec{b_1}\|} \vec{b_1}, \frac{1}{\|\vec{b_2}\|} \vec{b_2}, \cdots, \frac{1}{\|\vec{b_n}\|} \vec{b_n} \quad (5)$$

Step 2) calculate the rotation grid coordinates of population $S \{\vec{\pi_1}, \vec{\pi_1}, \cdots, \vec{\pi_1}\}$.

13: **Step 2.1)** translate the population S=$\{\vec{s_1}, \vec{s_2}, \cdots, \vec{s_m}\}$ to the first quadrant to obtain $f = \{\vec{f_1}, \vec{f_2}, \cdots, \vec{f_m}\}$.

14: **Step 2.2)** calculate the length of grid a.

15:     For$(i = 1; i <= m; i++)$

16:     {

17:         For$(j = 1; j <= n; j++)$

18:         {

19:             $MAX = \max_{j=1}^{m}\{\max_{i=1}^{n}\{t_{ij}\}\}$

20:             $MIN = \min_{j=1}^{m}\{\max_{i=1}^{n}\{t_{ij}\}\}$

21:         }

22:     }

23:     $a = (MAX - MIN)/R$

24: **Step 2.3)** calculate the $\vec{\pi}$.

25:     For$(i = 1; i <= m; i++)$

26:       {

27:         $\vec{f_i} = \vec{p^T} \cdot \vec{f_i}$

28:         $\vec{\pi} = \lfloor \vec{f_i}/a \rfloor$

29:        }

30: where the rotation grid coordinate of the jth individual in population A is $\vec{\pi_j}$.

---

235 *3.2. Rotating grid layer and the rotating grid cluster*

This chapter makes the stratified individual layered again according to the rotating grid layer.

Definition 3.3 rotating grid layer: In the n-dimension grid coordinates system, if the first dimensions of two points are the same, then the two points

240 are defined in the same rotating grid layer. As shown in Fig.3, points A and D are in the same rotating grid layer.

Definition 3.4 rotating grid cluster: In the n-dimension grid coordinates system, if the first dimensions coordinates of two points are not the same, but the other $n-1$ dimensional coordinates are the same, the two points are defined

245 at the same rotating grid cluster. As shown in Fig.2, points B and C are in the same rotating grid cluster.

Algorithm 2 shows how to determine whether two individuals are in the same rotating grid layer in step 1, the same rotating grid cluster in step 2 or the same rotation grid in step 3.

---

250 **Algorithm 2** how to determine two individuals whether are in the same rotating grid layer, the same rotating grid cluster or the same rotation grid.

1: **Input:** Rotating grid coordinates of two individuals $\vec{\pi_i}=\{\pi_{i1},\pi_{i2},...,\pi_{in}\}$ and $\vec{\pi_j}=\{\pi_{j1},\pi_{j2},...,\pi_{jn}\}$;

255 2: **Output:** Whether they are in the same rotation grid layer $\text{SL}(\pi_i,\pi_j)$, or in the same rotation grid cluster $\text{SC}(\pi_i,\pi_j)$, or in the same rotation grid $\text{SG}(\pi_i,\pi_j)$.

3: **Step 1)** judging whether two individuals are in the same rotating network layer.

260 4: Function $\text{SL}(\pi_i,\pi_j)$

5:    {

6:        If $\pi_{i1}==\pi_{j1}$ then

13

7:          return 1   //two individuals are in the same rotating network layer.

8:      Else return 0

9:    }

10: **Step 2)** judging whether two individuals are in the same rotation network cluster.

11: Function SC(π_i,π_j)

12: {

13:      If $\pi_{i1}\neq\pi_{j1}$ then

14:      {

15:        S=0;

16:        for (k=2;k<=n;k++)

17:        {

18:          If $\pi_{ik}\neq\pi_{jk}$ then

19:          {

20:            return 0; break;

21:          }

22:          Else s=1;

23:        }

24:        If s==1 then

25:          ruturn 1;   //two individuals are in the same rotation network cluster.

26:      }

27:      Else ruturn 0;

28: }

29: **Step 3)** judging whether two individuals are in the same rotating network.

30: Function SG(π_i,π_j)

31: {

32:    S=0;

33:    for (k=1;k<=n;k++)

34:    {

35:        If $\pi_{ik}\neq\pi_{jk}$ then

14

36:          {   return 0; break; }

37:          Else s=1;

38:      }

39:    If s==1 then return 1;   //two individuals are in the same rotating network.

40: }

3.3. The environmental selection in RGridEA

Algorithm 3 illustrates how to choose the individuals in the critical layer during environmental selection.

Algorithm 3 how to choose the individuals in the critical layer during environmental selection.

1: **Input:** Population $S = \overrightarrow{s_1}, \overrightarrow{s_2}, \cdots, \overrightarrow{s_m}$;

2: The transferred population $f = \overrightarrow{f_1}, \overrightarrow{f_2}, \cdots, \overrightarrow{f_m}$; the rotating grid coordinate of f is: $w = \overrightarrow{\pi_1}, \overrightarrow{\pi_2}, \cdots, \overrightarrow{\pi_m}$;

3: Individuals in the archive set $G = \{\overrightarrow{g_1}, \overrightarrow{g_2}, \cdots, \overrightarrow{g_r}\}$;

4: The transferred population of G: $fg = \{\overrightarrow{fg_1}, \overrightarrow{fg_2}, \cdots, \overrightarrow{fg_r}\}$;

5: The rotating grid coordinate of population G: $\{\overrightarrow{\pi g_1}, \overrightarrow{\pi g_2}, \cdots, \overrightarrow{\pi g_r}\}$;

6: The number of individuals which need to be chosen in the archive concentration: K.

7: **Output:** The chosen individuals:

8: **Step 1)** Select individuals in the critical layer by means of grid selection.

9: **Step 1.1)** Sort all individuals in the critical layer by means of the rotating grid layer sorting, $\overrightarrow{ssort_i}$ is the rotation grid coordinate of the individuals in the ith layer, $\overrightarrow{ssort_i}$ is the corresponding translation of coordinate.
$$Ssort = \{\overrightarrow{ssort_1}, \overrightarrow{ssort_2}, \cdots, \overrightarrow{ssort_m}\} \quad //\text{Sorted by rotating grid layer}$$

10: **Step 1.2)** select individuals in each layer

11:    For $(i = 1; i <= t; i++)$

12:    {

13:        if$(N + N_i <= K)$ //N_i represents the number of individuals in the ith layer.

15

14:    {

15:        If $(SG(\pi_k, \pi_j))$ //For any two individuals in the ith layer if they are in the same grid.

16:        {

17:    compare the corresponding f_k and f_j and select the individual with smaller

value and add it into SE and archive set; Then add other individuals in the same grid into the candidate set $BX = \{\overrightarrow{\pi b_1}, \overrightarrow{\pi b_2}, \cdots, \overrightarrow{\pi b_s}\}$, then

18:        $N = N + 1; r = r + 1;$

19:        }

20:        Else

21:        {

22:    add other individuals in this layer to the SE and archive set, then

23:        $N = N + 1; r = r + 1;$

24:        }

25:      }

26:        Else(the number of individuals in the ith layer is more than the required number of individuals)

27:        {

28:        While($j <= N_i$) //For each individual in ith layer.

29:        {

30:            for($p = 1; p <= r; p++$)

31:    //compute the number of individuals in the same rotation grid cluster as the selected individual in the archive set.

32:            {

33:                if $(SC(\pi g_p, \pi_j))$

34:                $W_j = W_j + 1;$

35:    //W_j represents the number of individuals in the same rotating grid cluster as the jth individual in the archive set.

36:            }

37:            $sort(W_j);$

38:            While($N <= K$)

39:              {

40: Select individuals according to the value of W_j in ascending order;

41:           $N = N + 1; r = r + 1;$

42:             }

43:        } End While

44:      } End Else

45:   } End For

46: **Step 2)** When the number of individuals selected from the rotation grid layers still cannot reach the required number of solutions in the archive set, then select individuals from the candidate set according to the rotary grid cluster.

47:   While$(j <= s)$

48:   //For each individual in the candidate set

49:   {

50:     for$(p = 1; p <= r; p + +)$

51:   //For each individual selected into the archive set

52:     {

53:       if $(SL(\pi g_p, \pi b_j))$

54:        $W_j = W_j + 1;$

55:   //W_j represents the number of individuals in the same rotary grid cluster with the ith individual in the archive set.

56:     }

57:     $sort(W_j);$

58:     While$(N <= K)$

59:     { Select individuals according to the value of W_j in ascending order into SE and archive set;

60:       $N = N + 1; r = r + 1;$

61:     } End While

62:   } End While
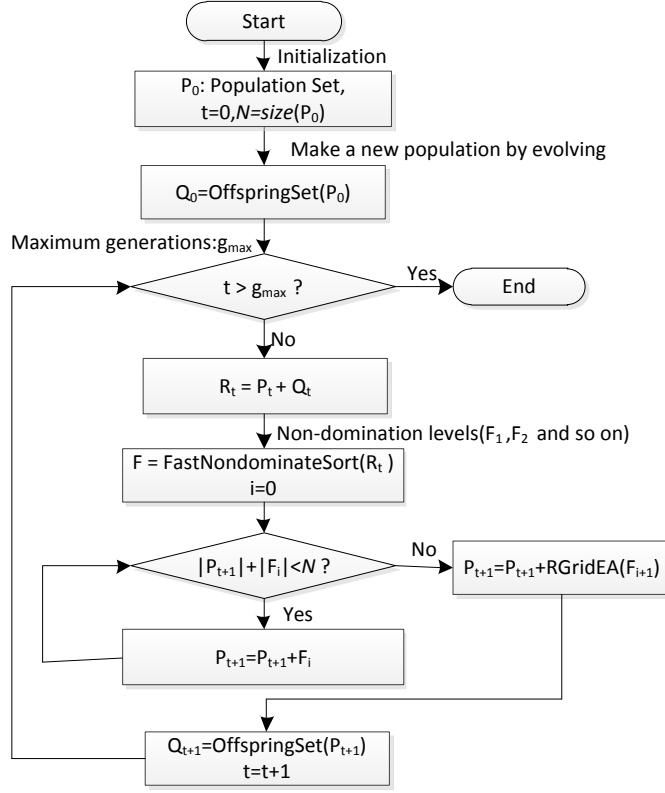
Figure 4: The flow chart of RGridEA .

385   **4. The framework of RGridEA and time complexity analysis**

4.1. The framework of RGridEA

The basic framework of RGridEA is the same with NSGA-II [44], but the rotation based grid selection is proposed for the individual selection in the critical layer. Specifically, the population must be randomly initialized. Then for each

390   generation, matching selection, crossover, and mutation must be done to produce a new generation. After the fast non-dominated sorting, for the individuals in the critical layer, the rotating grid coordinates of the individuals must be calculated according to Algorithm 1; then individuals are selected from the critical layer according to Algorithm 3. Figure 4 gives the overall flowchart of RGridEA.

18

As shown, RGridEA is applied to critical layer during the evolutionary process. The framework of the algorithm is shown in Algorithm 4.

Algorithm 4 the framework of RGridEA and time complexity analysis

---

1: **Input:** the number of objectives: n, population size: l, maximum generations: g_{max}.

2: **Output:** population $Q = \{i_1, i_2, \cdots, i_l\}$

3: Randomly initiate population $P_0 = i_1, i_2, \cdots, i_l$ and $t \to 0$.

4: $Q_0 = MakeNewPop(P_0)$ //Do the genetical operations including mating selection, crossover, mutation to produce the new generation, where the individual selection adopts the binary tournament selection.

5: While($t \leq g_{max}$)

6: {

7:   $R_t = P_t \bigcup Q_t;$

8:   $F = FastNondominateSort(R_t);$

9: //The challenge competition method [13] is used in the sort.

10:   $i = 1;$

11:   While ($\mid P_{t+2} \mid + \mid F_i \mid \leq l$)

12: //Put the individuals from the ith layer into the archive set.

13:   {

14:     $P_{t+1} = P_{t+1} \bigcup F_i;$

15:     $i = i + 1;$

16:   }

17:   $S = Pop(F_{i+1});$

18:   //output the individuals from the critical layer to the archive set S

19:   $M_{t+1} = RGridISEA(S < l - \mid P_{t+1} \mid, n, R)$

20:   //select individuals from the archive set according to Algorithm 1 and Algorithm 3.

21:   $P_{t+1} = P_{t+1} \bigcup M_{t+1};$

22:   $Q_{t+1} = MakeNewPop(P_{t+1});$

23:   //Do the operation of crossover, mutation and selection on the individ-

uals and generate a new population, where the selection adopts the binary tournament selection.

24:     $t = t + 1$

25: }

4.2. Time complexity analysis

The time complexity of RGridEA mainly comes from three aspects: mating selection, non-dominated sorting, and individual selection of RGridEA. Providing that the population size is m, and the number of objectives is n, then the analysis of the time complexity of these three aspects can be presented as follows.

Selection operation: this paper uses binary tournament selection. Two random shuffle costs is $O(m)$ since it is to select m individuals from m parent individuals; The time complexity of selecting one individual by comparing two parent individuals (comparing the dominance relationship and the convergence information) is $O(n)$. So the time complexity of the selection operation is $O(mn)$.

The non-dominated sorting: literature [13] point out costs of non-dominated sorting is $O(m^2n)$.

Individual selection of RGridEA: it includes two parts. The first one is to rotate the coordinate and calculate the rotation coordinate and rotating grid coordinate of individuals. The worst case of this part is that all individuals are in the critical layer. Thus, the number of individuals involved in coordinate conversion is m, and it will conduct m times by matrix multiplication between the $n \times n$ matrix and $n \times 1$ matrix, so the worst time complexity is $O(mn^2)$. Another part is the individual selection. The worst case is that the rotating grid layer divides the critical layer into R layers, and all individuals are in the critical layer. The average number of individuals in each layer is m/R. The time complexity of step 1.2 in Algorithm 3 is $O(R \times (m/R) \times m \times n) = O(m^2n)$; and the time complexity of step 2 in Algorithm 3 is $O(m^2n)$.

Thus,the worst time complexity of RGridEA is $O(m^2n + m^2n + m^2n + nm)$.

20

Due to n≪m in general, so the worst time complexity is $O(m^2 n)$.

5. Comparative experiments and analysis

In order to test the performance of RGridEA, proposed algorithm was compared with the other 7 evolutionary algorithms. The mentioned multi-objective evolutionary methods, genetic algorithm, have the ability of achieving a Pareto approximation set of multiobjective optimization problems in this paper. For completeness, we present a brief description of the 7 evolutionary algorithms.

1) Nondominated Sorting Genetic Algorithm II (NSGA-II). In NSGA-II [44]. a nondominated sorting approach is used to for each individual to create Pareto rank, and a crowding distance assign method is applied to implement density estimation. Currently, the NSGA-II is supposed to be the best known and most frequently applied EMOA.

2) SPEA2 [50].It is a general modification of the fitness assignment scheme that could make the SPEA2 better than its predecessor. The main differences of SPEA2 in comparison to SPEA are a fine-grained fitness assignment strategy, a density estimation technique, and an enhanced archive truncation method.

3) ϵ-MOEA. This approach was proposed in [45], it consists of a steady-state GA which maintains an archive of nondominated individual. The objective space is divided into a grid of boxes, whose size can be adjusted by the choice of ϵ. Note however, that this algorithm does not use the Pareto dominance relation when updating the archive. Instead, it uses the ϵ-dominance to update the archive at each generation. It has been found to be a very competitive MOEA. The parameter ϵ can control the degree of Pareto dominance relationship when comparing two individuals.

4) AR+CD$^{'}$. AR+CD$^{'}$ was proposed in [47]. AR+CD$^{'}$ adopts the CD$^{'}$ into AR to improve the convergence of the algorithm on MaOPs. Numerical studies have demonstrated the efficiency of the algorithm.

5) AR+DMO. DMO [63] employs a diversity management operator to control or promote the diversity requirement. If the diversity indicator is smaller

21

Table 2: The DTLZ test suite.

Problem	Defination	Decision space
DTLZ1	$f_l(\overrightarrow{x}) = \frac{1+d1(\overrightarrow{x_d})}{2} s1(\overrightarrow{x_p})$	$n = m - 1 + 5$
DTLZ2	$f_l(\overrightarrow{x}) = (1 + d2(\overrightarrow{x_d}))s2(\overrightarrow{x_p})$	
DTLZ3	$f_l(\overrightarrow{x}) = (1 + d1(\overrightarrow{x_d}))s2(\overrightarrow{x_p})$	
DTLZ4	$f_l(\overrightarrow{x}) = (1 + d2(\overrightarrow{x_d}))s2(t1(\overrightarrow{x_p}, \alpha))$	$n = m - 1 + 10$
DTLZ6	$d = d3(\overrightarrow{x_d})$ $f_l(\overrightarrow{x}) = (1 + d)s2(t2(\overrightarrow{x_p}, d, 1))$	
DTLZ7	$d = 2 + 9 \times d4(\overrightarrow{x_d})$ $h(\overrightarrow{x, d}) = m - \sum\limits_{t=1}^{m-1} [\frac{x_t}{d}(1 + sin(3\pi x_t))]$ $f_t(\overrightarrow{x}) = \begin{cases} x_t & l = 1, 2 \cdots, m - 1 \\ d \times h(\overrightarrow{x}, d) & l = m \end{cases}$	$n = m - 1 + 20$
Distance function	$d1(\overrightarrow{x_d}) = 100 \left(|\overrightarrow{x_d}| + \sum\limits_{x_l \in \overrightarrow{x_d}} [(x_t - 0.5)^2 - cos(20\pi(X_t - 0.5))] \right)$ $d2(\overrightarrow{x_d}) = \sum\limits_{x_l \in \overrightarrow{x_d}} (x_l - 0.5)^2$ $d3(\overrightarrow{x_d}) = \sum\limits_{x_t \in \overrightarrow{x_d}} x_l^{0.1}$ $d4(\overrightarrow{x_d}) = \frac{1}{|\overrightarrow{x_d}|} \sum\limits_{x_l \in x_d} x_l$	
Shape function	$s1_l(\overrightarrow{x_p}) = \begin{cases} \prod\limits_{\varnothing=1}^{m-1} x_\varnothing & l = 1 \\ (1 - x_{m-l+1}) \prod\limits_{\varnothing=1}^{m-1} x_\varnothing & l = 2, 3, \cdots m \end{cases}$ $s2_l(\overrightarrow{x_p}) = \begin{cases} \prod\limits_{\varnothing=1}^{m-1} cos(\frac{\pi}{2} x_\varnothing) & l = 1 \\ sin(\frac{\pi}{2} x_{m-l+1}) \prod\limits_{\varnothing=1}^{m-1} cos(\frac{\pi}{2} x_\varnothing) & l = 2, 3, \cdots, m \end{cases}$	
Conversion function	$t1(\overrightarrow{x_p}, \alpha) = x_l^\alpha$ $t2(\overrightarrow{x_p}, d, k) = \begin{cases} x_l & l = 1, 2, \cdots, k \\ \frac{0.5 + x_\varnothing d}{1 + d} & l = k + 1, k + 2, \cdots, |\overrightarrow{x_p}|, k \geq 1 \end{cases}$	
Decision vector	$\overrightarrow{x_p} = (x_1, x_2, \cdots, x_{m-1})^T$ $\overrightarrow{x_p} = (x_1, x_2, \cdots, x_{m-1})^T$ $\overrightarrow{x_d} = (x_m, x_{m+1}, \cdots, x_n)^T$	

than 1 according to test, the diversity promotion mechanism is activated, conversely deactivated.

6) HypE [48]. It is a new hypervolume-based evolutionary algorithm for many-objective optimization, which adopts Monte Carlo simulation to approximate the exact hypervolume values. In HypE, the nondominated solutions are compared according to their hypervolume-based fitness values. The experimental results showed that HypE outperforms in some problems to some MOEAs.

7) Preference ordering genetic algorithm (POGA) [49]. It uses the preference order-based approach as an optimality criterion in the ranking stage of MOEAs. POGA exerts the higher selection pressure over objective spaces of different dimensionality compared with the traditional Pareto dominance-based ranking scheme.

Additionally, another experiment is conducted to compare the time cost between NSGA-II and RGridEA. RGridEA and AR+DMO were implemented by C++. The source code of NSGA-II and ϵ-MOEA can be found in `www.iitk.ac.in/kangal/index.shtml`. The source code of HypE, AR+CD and POGA are presented in `http://www.tik.ee.ethz.ch/sop/pisa/`. AR+DMO and SPEA2 were adopted in jMetal 3.1. All experiments were conducted on the Inspur server (NF5280M3), and the computer with INTER XEON E3-1230 v2 CPU, 8G RAM. The OS was CentOS 6.4 x86_64. The compile system of C and C++ was GCC 4.4.7 and GCC 4.6.4. respectively, and the JAVA adopts the OpenJDK 1.7.0. In addition, we applied the Octave 3.6.3 to run the MATLAB codes.

5.1. The test suite and indicators

In order to compare the performance of the algorithms, the DTLZ test suit [51] is chosen as the test problems. Providing that the number of objectives is m, and the number of decision variables is n, the DTLZ test suite can be defined as Table 2 shows according to [60], where the decision vector is divided into two parts (distance vector $\overrightarrow{x_d}$, and position vector $\overrightarrow{x_p}$).

23

In order to compare the performance of the algorithms, we adopted three widely-used indicators to evaluate the final obtained solution sets such as the Generational Distance [53], Inverted Generational Distance [54].

Generational Distance(GD) was used to evaluate the convergence performance, and it is defined as follows:

$$GD = \frac{\sqrt{\sum\limits_{\vec{i} \in P} d_{\vec{i}}^2}}{n} \qquad (6)$$

where n is the number of individuals in the obtained solution set, and $d_{\vec{i}} = min_{\vec{j} \in PF^*} |\vec{i} - \vec{j}|$ shows the minimum Euclidean distance of individual $\vec{i}$ to the PF. Thus, the smaller the value of GD, the better the convergence.

Inverted Generational Distance(IGD)evaluates the comprehensive performance of an algorithm since it can also evaluate the convergence and the distribution of the obtained solution set, which is defined as follows:

$$IGD = \frac{\sum\limits_{\vec{j} \in PF^*} d'_{\vec{j}}}{n} \qquad (7)$$

IGD is a reverse mapping of GD. Specifically, $d'_{\vec{j}} = min_{\vec{i} \in P} |\vec{j} - \vec{i}|$ shows the minimum Euclidean distance of an individual to the PF, so the smaller the value of IGD, the better.

5.2. The settings of experimental parameters

We applied the real code in the experiments. The distribution parameter of crossover operator (Simulated binary crossover) was $\eta_c = 20$, and the crossover rate was $P_c = 1$. Also, the distribution parameter of mutation operator (Polynomial mutation) was $\eta_m = 20$ and the mutation rate $P_m = \frac{1}{n}$ where n is the number of decision variables. All experiments were conducted with 30 independent runs on the DTLZ test suite with 3, 4, 5, 6, 8 and 10 objectives. The population size was 100. The maximum generations for DTLZ1,DTLZ2,DTLZ4,DTLZ6 and DTLZ7 was 300, and of DTLZ3 was 1000 (because the DTLZ3 is designed hard to converge).

24

5.3. The experimental statistical results

In order to compare the performance of algorithms, the mean and standard deviation of the GD and IGD values was applied in this paper. Furthermore, multiple comparisons on analysis of variance (ANOVA) and the sample mean were used, for the reason that an evolutionary algorithm is a kind of stochastic algorithm which may produce the sampling error caused by limited samples.

Hypothesis H_0 and alternative hypothesis H_1 are given follows:

$$\begin{cases} H_0 & \mu_{ISEA} = \mu_{Other} \\ H_1 & \mu_{ISEA} \neq \mu_{Other} \end{cases} \tag{8}$$

According to the central limit theorem, the final obtained solutions obey the normal distribution after independent repeated trials, while this paper uses experiment which are independent repeated experiments. Therefore the Tamhane's T2[56] method was chosen to handle the statistical data. The variance analysis uses the significance with P value, and $P = 0.05$ in this paper. The bigger P value indicates that the original assumption is available with bigger probability.

The statistical results of the algorithm performance sample are given below in Table 3 and Table 4. The first and second lines of the table are the sample mean and standard deviation, and respectively show the best algorithm in the corresponding index with dark mark. At the same time,the P value given was calculated by the Levene method, if the value is less than it. It is worth noting that the P value is the result of Tamhane's T2 method.

5.4. IGD Values and their analysis

In this section, we compare the proposed RGridEA with all 7 algorithms. Table 3 presents the IGD values obtained by 8 algorithms on the DTLZ test suite.

Table 3: IGD test results

DTLZ1							
objective	RGridEA	AR+CD'	AR+DMO	ϵ-MOEA	HypE	NSGA2	POGA	SPEA2

		RGridEA	AR+CD'	AR+DMO	ε-MOEA	HypE	NSGA2	POGA	SPEA2
3	Mean	0.028561	0.136277	0.060633	0.019981	**0.159646**	0.045719	0.03552	0.021524
	Std	0.023996	0.034575	0.070813	**0.001236**	0.170258	0.061777	0.029733	0.001448
4	Mean	0.069613	0.117693	0.091457	**0.04669**	0.121595	0.084358	0.082788	0.573365
	Std	0.025114	0.009775	0.043402	**0.000831**	0.032629	0.031982	0.046602	0.983423
5	Mean	0.079775	17.67496	10.02853	**0.075874**	0.170082	5.980304	1.48549	37.51033
	Std	0.008694	9.329063	14.35102	**0.005329**	0.05802	6.057428	1.329665	11.87926
6	Mean	0.131885	94.1526	19.24484	**0.086961**	0.288835	22.59211	5.047828	92.74009
	Std	0.02347	20.40155	19.18083	**0.00253**	0.201418	30.66529	9.445533	28.99593
8	Mean	0.199832	145.2141	29.88714	**0.140228**	0.611608	33.19503	6.9883	168.9068
	Std	**0.036367**	13.27861	25.59748	0.057017	0.63875	34.70939	8.050367	33.05956
10	Mean	**0.247352**	148.0565	41.33891	0.265236	0.724545	46.77505	7.888394	231.4319
	Std	**0.087187**	14.73328	32.24075	0.095664	0.69791	53.81618	9.548837	22.53637
					DTLZ2				
		RGridEA	AR+CD'	AR+DMO	ε-MOEA	HypE	NSGA2	POGA	SPEA2
3	Mean	0.266874	0.31967	0.068084	0.063061	0.136618	0.068415	0.068729	**0.054097**
	Std	0.00776	0.056111	0.002689	0.00113	0.037203	0.002798	0.003411	**0.001101**
4	Mean	0.193631	0.417326	0.148501	0.134808	0.271268	0.14937	0.149676	**0.133904**
	Std	0.017642	0.04772	0.007576	**0.002792**	0.069403	0.007996	0.006698	0.006417
5	Mean	0.236568	0.509975	0.324514	**0.193873**	0.377519	0.341617	0.277904	0.342738
	Std	0.04441	0.043164	0.03401	**0.014452**	0.120056	0.035205	0.023277	0.043852
6	Mean	0.456282	0.346475	0.905958	**0.298814**	0.478796	1.057213	0.588598	1.289585
	Std	0.073984	0.02584	0.130757	**0.010768**	0.051488	0.171513	0.061792	0.203818
8	Mean	0.546411	1.541229	1.755973	**0.411955**	0.606467	1.793006	1.044553	2.329808
	Std	0.109702	0.116008	0.162437	**0.018439**	0.072319	0.192869	0.089803	0.043087
10	Mean	0.656709	1.872853	1.959931	**0.459939**	0.729899	1.975268	1.149311	2.452692
	Std	0.085033	0.092566	0.165439	**0.017792**	0.065009	0.157347	0.08072	0.037093
					DTLZ3				
		RGridEA	AR+CD'	AR+DMO	ε-MOEA	HypE	NSGA2	POGA	SPEA2
3	Mean	0.12843	0.648356	0.066753	0.062297	0.226699	0.066923	0.067037	**0.052759**
	Std	0.008129	0.029778	0.002991	0.002008	0.092496	0.003646	0.002994	**0.000947**
4	Mean	0.138521	0.783609	16.63191	**0.129448**	0.46331	14.46677	1.572408	10.95491
	Std	0.028352	0.042428	10.02766	**0.006541**	0.229766	10.56839	1.850401	5.910645
5	Mean	0.234526	92.45527	92.94022	**0.230989**	0.961029	83.69353	9.993877	201.7944
	Std	0.102498	35.51437	48.76644	**0.046033**	0.824167	60.00005	5.050345	46.55174
6	Mean	0.525172	439.573	154.0831	**0.354267**	1.616495	112.0992	10.16518	624.8617
	Std	**0.051103**	80.35887	83.65063	0.115132	1.997067	98.95917	4.958986	127.1297
8	Mean	0.744913	955.8956	277.8197	**0.281041**	2.874511	162.7794	7.963566	1244.775
	Std	**0.085053**	108.6427	142.3164	0.164069	3.205131	95.22757	4.619181	197.4352
10	Mean	**0.874502**	1167.792	340.5586	0.919162	3.800173	210.3699	4.95662	1601.777
	Std	**0.062536**	124.7224	149.6129	0.065131	4.960813	109.6	3.035206	113.7412
					DTLZ4				
		RGridEA	AR+CD'	AR+DMO	ε-MOEA	HypE	NSGA2	POGA	SPEA2
3	Mean	0.501215	0.501026	**0.065458**	0.299122	0.561451	0.065573	0.065505	0.181259
	Std	0.339054	0.359753	**0.002553**	0.28968	0.374712	0.00279	0.002664	0.261986
4	Mean	0.4235591	0.439872	0.143648	0.449114	0.545436	0.144268	**0.14022**	0.247719
	Std	0.27139	0.295403	0.007974	0.298494	0.309637	0.008533	**0.006991**	0.182765
5	Mean	**0.306022**	0.464845	0.948132	0.563729	0.6163	0.937738	0.381403	0.480924
	Std	0.224765	0.324729	0.134746	0.337503	0.256051	0.145107	**0.087956**	0.112237
6	Mean	**0.471886**	0.480859	1.859712	0.616535	0.698521	1.888705	0.927523	1.303985
	Std	0.048473	0.070907	0.104961	0.192454	0.10358	0.086146	0.148874	0.18296
8	Mean	**0.611476**	1.945896	2.260167	0.76917	0.909404	2.252227	1.080573	2.333698
	Std	**0.033233**	0.081767	0.069026	0.153821	0.069214	0.059757	0.238366	0.041921
10	Mean	**0.673568**	2.154221	2.3982	0.815702	1.04744	2.404948	0.857476	2.433122
	Std	**0.022879**	0.057134	0.0519	0.136822	0.075767	0.054379	0.253267	0.04483
					DTLZ5				
		RGridEA	AR+CD'	AR+DMO	ε-MOEA	HypE	NSGA2	POGA	SPEA2
3	Mean	0.017083	0.305364	0.005418	0.007006	0.026538	0.005493	0.005577	**0.00427**
	Std	**0.000003**	0.032535	0.000325	0.000321	0.010683	0.000349	0.000379	0.000339

		RGridEA	AR+CD'	AR+DMO	ε-MOEA	HypE	NSGA2	POGA	SPEA2
4	Mean	0.065259	0.176109	**0.045718**	0.047931	0.120164	0.049265	0.051853	0.12173
	Std	0.000253	0.078117	0.007958	0.004642	0.050161	0.009717	0.009498	0.024092
5	Mean	**0.05385**	0.084114	0.097881	0.094312	0.139484	0.099434	0.738656	0.348947
	Std	**0.00012**	0.05759	0.024854	0.012725	0.061614	0.029989	0.052058	0.09314
6	Mean	**0.065109**	0.154423	0.153414	0.247657	0.148626	0.169318	0.747462	1.0417
	Std	**0.000128**	0.260939	0.050725	0.001675	0.061322	0.070015	0.002388	0.221492
8	Mean	**0.088096**	1.239604	0.536217	0.27205	0.171581	0.681896	0.747986	2.013876
	Std	**0.000257**	0.656465	0.436929	0.003696	0.070986	0.467365	0.002348	0.552711
10	**0.673568**	0.20524	2.013417	1.316349	0.2781	0.158126	1.47854	1.406243	2.075518
	Std	**0.00065**	0.330493	0.495115	0.002077	0.050372	0.448959	0.877847	0.589816
DTLZ6									
		RGridEA	AR+CD'	AR+DMO	ε-MOEA	HypE	NSGA2	POGA	SPEA2
3	Mean	0.212404	0.443041	0.067521	0.074357	0.198109	0.066307	0.085423	**0.05691**
	Std	**0.000368**	0.037243	0.027026	0.026102	0.055989	0.021351	0.104091	0.023367
4	Mean	**0.150125**	1.817027	3.18071	0.462374	4.062924	3.247756	2.177313	1.936187
	Std	**0.000877**	0.409872	0.348075	0.034892	0.663313	0.370642	0.268199	0.157603
5	Mean	**1.139011**	4.90684	6.199361	1.664344	5.559299	6.595872	1.811516	9.074081
	Std	**0.053717**	0.235547	0.516051	0.173907	0.641651	0.517941	0.466277	0.214398
6	Mean	**1.514755**	6.115559	7.79062	2.455103	5.873688	7.920667	2.283867	9.86346
	Std	0.105779	0.255439	0.406192	7.059253	0.682043	0.424299	0.724811	**0.049215**
8	Mean	**1.097396**	7.819625	8.783086	2.021616	6.139422	8.839023	5.524955	9.984556
	Std	0.02754	0.307759	0.486317	5.99442	0.572013	0.382728	3.926093	**0.021291**
10	Mean	**1.705394**	8.726099	9.220922	2.962116	6.253106	9.247631	8.920017	10.02354
	Std	0.093294	0.239605	0.410501	8.534891	0.579737	0.393165	0.166094	**0.025091**
DTLZ7									
		RGridEA	AR+CD'	AR+DMO	ε-MOEA	HypE	NSGA2	POGA	SPEA2
3	Mean	**0.061654**	0.417182	0.092618	0.075752	0.209989	0.076936	0.076942	0.084658
	Std	**0.000001**	0.319255	0.085512	0.081425	0.112611	0.004712	0.003661	0.077641
4	Mean	0.233596	0.780904	0.234705	0.335609	0.485283	0.217204	0.215036	**0.194494**
	Std	**0.004616**	0.355797	0.055939	0.184456	0.177111	0.009399	0.009806	0.053076
5	Mean	0.474869	1.67758	0.456191	0.587288	0.871072	0.440482	**0.406589**	0.41971
	Std	**0.00893**	0.342107	0.024718	0.242301	0.204519	0.026684	0.018096	0.050711
6	Mean	**0.617735**	2.814414	0.763256	0.752538	0.943261	0.725357	0.666424	0.788131
	Std	**0.015139**	0.521839	0.041993	0.042622	0.130556	0.033909	0.029422	0.061987
8	Mean	**1.04787**	8.731296	2.415582	1.590351	1.052776	1.697227	2.356592	2.250059
	Std	**0.009583**	0.862194	0.659491	0.221246	0.07353	0.453913	0.51562	0.728682
10	Mean	1.922929	18.63377	8.4124	7.771367	**1.230667**	5.692742	4.331883	4.671672
	Std	0.342148	1.92115	2.322535	15.889629	**0.087323**	1.746011	0.852336	1.559201

It can be seen from Table 3 that IGD values of RGridEA are better than the other algorithms in most test instances, especially on DTLZ4, DTLZ5, DTLZ6 and DTLZ7 problems as well as on 8-dimensional DTLZ3. Therefor, RGridEA has the best performance than all others in solving many-objective problems. Furthermore, in bias problem DTLZ4, RGridEA performs much better than other algorithms. Among all 7 test instances, it wins 4 competitions. Finally, in degenerate problems,(e.g , DTLZ5, DTLZ7) and disconnected problems (e.g., DTLZ6), RGridEA wins 15 out of 18 instances. Relatively speaking, RGridEA does not show such outstanding performance on DTLZ1-3 problems. From Table 3 we can see that the RGridEA does not perform better than ε-MOEA

on DTLZ2, but the parameter setting for ϵ-MOEA is a big difficulty. RGridEA outperforms ϵ-MOEA in other test problems. For DTLZ3 problems, RGridEA shows an interesting search behavior, it remains competitive on 6-, 8- and 10-objective instances, but performs worst on 3-, 4- and 5-objective instances.

ϵ-MOEA is very competitive on DTLZ1, DTLZ2 and DTLZ3 instances. However, it does not show advantage over the other algorithms on the other problems. SPEA2 performs well on DTLZ2 problem instance. AR+CD$'$, NSGA2 and AR+DMO are not competitive on DTLZ instances, which is reflected in Table 3. HypE generally has the medium-high performance on most of problems among the compared algorithms. It is worth noting, for DTLZ1 and DTLZ7 problems, HypE performs best on 3- and 10-objective instances.

5.5. GD value and its analysis

Table 4 shows the GD values from algorithms on different test instances. It can be seen from Table 4 that the convergence of RGridEA is superior than the others in most test instances, especially on DTLZ1, DTLZ3,DTLZ4, DTLZ5, DTLZ6 and DTLZ7. However the convergence of RGridEA is not better than ϵ-MOEA from Table 4.

Table 4: GD test results

DTLZ1								
		RGridEA	AR+CD'	AR+DMO	ϵ-MOEA	HypE	NSGA2	POGA	SPEA2
3	Mean	0.030854	0.00059	0.067513	**0.000383**	0.190403	0.033901	0.041237	0.078273
	Std	0.344712	0.002906	0.232025	**0.000225**	0.285561	0.097641	0.127367	0.159874
4	Mean	0.001276	0.000333	0.425695	**0.000045**	0.185012	0.539174	0.359071	1.750147
	Std	0.002543	0.000782	0.568917	**0.000045**	0.325465	0.953105	0.72236	1.454311
5	Mean	**0.000065**	4.20213	10.82189	0.000107	0.242483	12.4375	8.503816	12.07769
	Std	**0.000090**	1.3166	1.298344	0.000075	0.448208	1.099128	2.28895	1.116022
6	Mean	0.000271	12.88878	15.21538	**0.000174**	0.220776	16.24124	12.34922	18.20003
	Std	0.000374	1.199853	0.810218	**0.000122**	0.534051	0.753615	2.569634	0.644595
8	Mean	**0.000291**	15.75474	16.27444	0.000372	0.259593	16.83331	14.52076	18.46974
	Std	**0.000096**	0.428023	0.493381	0.000705	0.562437	0.364554	2.081399	0.204585
10	Mean	0.049369	15.00267	15.31595	**0.00112**	0.20704	15.63227	13.25829	17.06005
	Std	0.156281	0.349063	0.367733	**0.002984**	0.408467	0.243538	3.667486	0.116552
DTLZ2								
		RGridEA	AR+CD'	AR+DMO	ϵ-MOEA	HypE	NSGA2	POGA	SPEA2
3	Mean	**0.000253**	0.000922	0.001199	0.000739	0.000433	0.001213	0.001126	0.001108
	Std	0.000259	0.000155	0.000223	**0.0000603**	0.000209	0.000278	0.000214	0.000277
4	Mean	**0.000858**	0.002649	0.005239	0.002097	**0.001093**	0.005087	0.004827	0.005079
	Std	**0.000145**	0.000334	0.002023	0.000152	0.001105	0.002042	0.001958	0.002032
5	Mean	**0.001280**	0.005171	0.048727	0.004249	0.003634	0.055324	0.033539	0.051932
	Std	0.000871	0.000852	0.010025	**0.00077**	0.003114	0.010409	0.007955	0.009203

		RGridEA	AR+CD'	AR+DMO	ϵ-MOEA	HypE	NSGA2	POGA	SPEA2
6	Mean	0.045080	0.018537	0.139435	**0.005306**	0.006671	0.155373	0.094484	0.19547
	Std	0.018512	0.005186	0.011659	**0.000564**	0.00371	0.012325	0.014103	0.00867
8	Mean	0.014615	0.165888	0.209846	**0.006583**	0.013679	0.214143	0.15554	0.229102
	Std	0.037376	0.011154	0.005981	**0.000936**	0.004434	0.005376	0.011185	0.003109
10	Mean	0.019431	0.192701	0.22182	**0.005523**	0.016008	0.223654	0.167559	0.233946
	Std	0.011771	0.007221	0.004246	**0.000658**	0.005195	0.003601	0.012938	0.002886
					DTLZ3				
		RGridEA	AR+CD'	AR+DMO	ϵ-MOEA	HypE	NSGA2	POGA	SPEA2
3	Mean	0.000559	**0.000131**	0.066048	0.000763	1.361116	0.039381	0.126899	0.070506
	Std	0.001439	**0.000148**	0.240774	0.00029	1.121367	0.182463	0.420816	0.251423
4	Mean	0.002108	**0.000368**	15.30629	0.003144	2.342108	16.31539	6.494867	20.98231
	Std	0.011045	**0.000268**	7.344312	0.000885	1.543308	8.907722	4.449479	5.48934
5	Mean	**0.004746**	30.1842	75.9899	0.00739	2.84328	81.47112	56.33622	81.4967
	Std	**0.004290**	6.904862	7.931629	0.00488	2.10862	7.259454	7.566403	7.438709
6	Mean	**0.003261**	85.97242	113.6477	0.01117	2.348787	117.4415	86.42885	154.2659
	Std	**0.002037**	6.684272	7.564625	0.008345	1.723368	8.546709	11.29582	7.292337
8	Mean	0.189075	141.5063	154.9687	**0.002462**	2.505086	160.1957	123.0878	196.226
	Std	0.438505	6.481064	7.565673	**0.003838**	1.889052	7.095127	15.29846	3.056798
10	Mean	0.003525	164.1845	174.352	**0.000129**	1.931424	179.6997	141.6373	203.5969
	Std	0.003110	5.1712	5.818044	**0.000488**	1.628887	5.254885	16.50758	2.468717
					DTLZ4				
		RGridEA	AR+CD'	AR+DMO	ϵ-MOEA	HypE	NSGA2	POGA	SPEA2
3	Mean	0.004183	0.001054	0.001116	0.00085	**0.000772**	0.001136	0.001057	0.000926
	Std	0.003453	0.002198	**0.000201**	0.000306	0.002149	0.000206	0.000289	0.000537
4	Mean	0.013804	0.001914	0.005406	0.002396	**0.001403**	0.006254	0.00429	0.007046
	Std	0.004698	0.001039	0.002917	**0.000729**	0.002135	0.003415	0.001999	0.004534
5	Mean	**0.002176**	0.005486	0.136959	0.005012	0.002372	0.138232	0.056784	0.081445
	Std	0.001906	0.002732	0.012362	**0.001834**	0.002798	0.013311	0.015594	0.031806
6	Mean	**0.002905**	0.042832	0.205886	0.008224	0.005802	0.207122	0.133861	0.193519
	Std	**0.001974**	0.012217	0.005864	0.003432	0.006051	0.004882	0.017106	0.01228
8	Mean	**0.003614**	0.202547	0.229406	0.015422	0.033844	0.229339	0.154709	0.228639
	Std	**0.001023**	0.006159	0.003439	0.010278	0.015024	0.003142	0.051504	0.002981
10	Mean	**0.009610**	0.215772	0.235464	0.015052	0.065244	0.23567	0.06412	0.231757
	Std	0.002975	0.003852	0.002811	0.010897	0.008801	**0.002736**	0.067834	0.003274
					DTLZ5				
		RGridEA	AR+CD'	AR+DMO	ϵ-MOEA	HypE	NSGA2	POGA	SPEA2
3	Mean	0.000763	**0.00004**	0.000179	0.000061	0.00012	0.00018	0.000186	0.000181
	Std	**0.000001**	0.00002	0.00008	0.000007	0.000057	0.000048	0.000052	0.000084
4	Mean	0.005841	**0.000917**	0.10273	0.051263	0.018704	0.113235	0.111263	0.133341
	Std	**0.000021**	0.001617	0.008053	0.00364	0.005793	0.007096	0.005873	0.004463
5	Mean	0.021565	**0.009267**	0.145011	0.052457	0.030852	0.149483	0.10895	0.153507
	Std	**0.000085**	0.007563	0.005088	0.003056	0.006616	0.005115	0.012744	0.004446
6	Mean	**0.018888**	0.052371	0.167373	0.058544	0.036697	0.17358	0.108655	0.199055
	Std	**0.000026**	0.022146	0.005799	0.003821	0.006763	0.005549	0.016567	0.007036
8	Mean	**0.018869**	0.152377	0.202504	0.05497	0.043513	0.210284	0.118365	0.235357
	Std	**0.000025**	0.043918	0.010433	0.003893	0.007991	0.009325	0.005622	0.003549
10	Mean	**0.022983**	0.202966	0.225852	0.0598	0.046156	0.230392	0.211908	0.239001
	Std	**0.000019**	0.01982	0.007958	0.005899	0.007116	0.006212	0.026824	0.002641
					DTLZ6				
		RGridEA	AR+CD'	AR+DMO	ϵ-MOEA	HypE	NSGA2	POGA	SPEA2
3	Mean	0.035041	0.00708	0.007266	**0.006738**	0.010529	0.007656	0.008519	0.009306
	Std	**0.000246**	0.002651	0.00245	0.001817	0.010278	0.002835	0.003879	0.005824
4	Mean	**0.047863**	0.250693	0.586745	0.116747	0.53661	0.596643	0.54047	0.392198
	Std	**0.000013**	0.038615	0.032237	0.012994	0.049948	0.033058	0.032756	0.018429
5	Mean	0.152053	0.55521	0.849487	**0.149331**	0.665707	0.876296	0.490861	0.936825
	Std	**0.000662**	0.021162	0.025509	0.006949	0.050654	0.023267	0.051736	0.012547
6	Mean	**0.208499**	0.667045	0.942097	0.255084	0.700554	0.95011	0.505441	0.985175
	Std	**0.000989**	0.023026	0.013748	0.019101	0.053282	0.012643	0.064176	0.003258

		RGridEA	AR+CD'	AR+DMO	ϵ-MOEA	HypE	NSGA2	POGA	SPEA2
8	Mean	**0.143077**	0.825261	0.980546	0.295246	0.726071	0.980224	0.789494	0.992084
	Std	**0.000201**	0.022617	0.007871	0.189665	0.047154	0.007953	0.197722	0.001455
10	Mean	**0.230326**	0.897908	0.989435	0.279693	0.739013	0.989548	0.930619	0.993127
	Std	0.001612	0.018887	0.004975	0.245624	0.045049	0.004777	0.014717	**0.001236**
DTLZ7									
		RGridEA	AR+CD'	AR+DMO	ϵ-MOEA	HypE	NSGA2	POGA	SPEA2
3	Mean	0.001531	0.001669	0.003615	**0.000695**	0.027896	0.00348	0.0036	0.003663
	Std	**0.000001**	0.000741	0.001124	0.0000443	0.046424	0.000908	0.000976	0.001306
4	Mean	0.046261	0.005303	0.01444	**0.002246**	0.311993	0.014399	0.01348	0.01162
	Std	0.000553	0.001192	0.003307	**0.000525**	0.244159	0.002898	0.00243	0.002915
5	Mean	0.011798	0.014157	0.063083	**0.003575**	0.907918	0.062591	0.032238	0.152918
	Std	**0.000010**	0.003925	0.024677	0.001718	0.194478 0	.023447	0.012821	0.042896
6	Mean	0.080284	0.050964	0.222384	**0.004428**	1.344934	0.215552	0.056611	0.520783
	Std	0.010190	0.020149	0.05441	**0.001672**	0.135003	0.055761	0.016864	0.108406
8	Mean	0.037093	0.642556	1.207679	**0.012123**	1.662746	1.124849	0.083018	2.170381
	Std	**0.000021**	0.105221	0.185632	0.00737	0.205173	0.194847	0.015639	0.243441
10	Mean	0.045678	1.831763	2.756529	**0.018053**	1.826922	2.493053	0.14593	3.630855
	Std	**0.000098**	0.23219	0.315025	0.014618	0.222988	0.257691	0.034992	0.28017

For DTLZ1, DTLZ2 and DTLZ3 problems, ϵ-MOEA performs better than RGridEA, but in 5- and 8-objective problems, RGridEA performs better. AR+CD$'$, AR+DMO, SPEA2 and NSGA2 consistently does not perform well in all higher dimensions of the problem. This is mainly due to its ineffectiveness of selection pressure in both mating selection and environmental selection in a high-dimensional space. POGA cannot obtain very satisfying results on DTLZ test suit. HypE has the medium-high performance on the most of the considered problems among the compared algorithm.

For bias DTLZ4 problem, the difference in the performances between RGridEA and ϵ-MOEA is clear from Table 4. RGridEA outperformed better than ϵ-MOEA on 6-, 8- and 10-objectives in terms of GD metric. HypE is very competitive on 3- and 4-objectives instances, which is reflected in Table 4. However, it does not show advantage over the other algorithms on problems having more than three objectives.

Similar observation is made for the DTLZ5, DTLZ6 and DTLZ7 problem. The proposed RGridEA works well on all the considered instances except for 4-objective DTLZ7 instances. Indeed, RGridEA is significantly outperformed by other seven algorithms on 17 out of 18 scaled problem instances, verifying the effectiveness of the Rotated Grid mechanism.
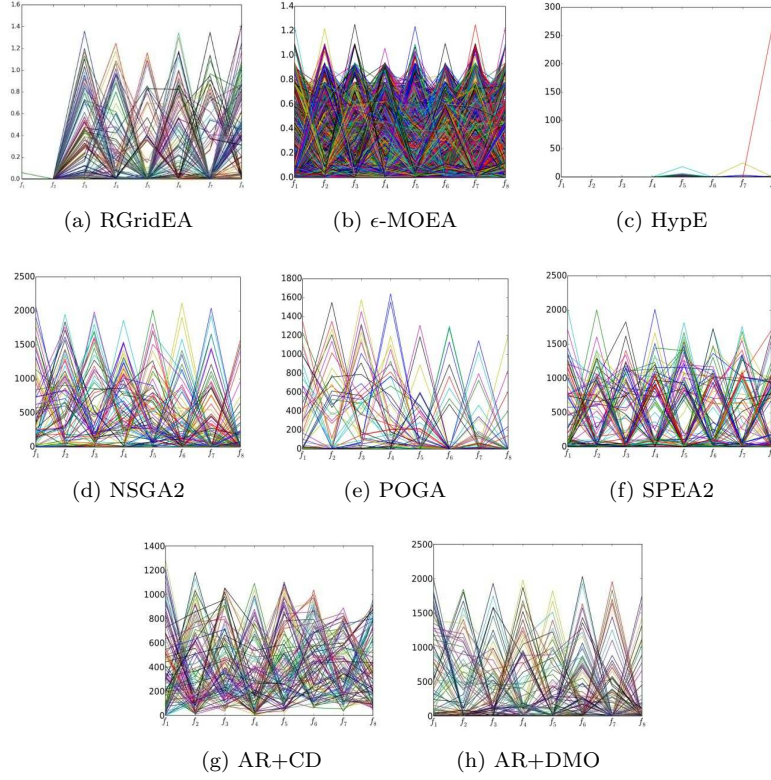
(a) RGridEA   (b) ϵ-MOEA   (c) HypE

(d) NSGA2   (e) POGA   (f) SPEA2

(g) AR+CD   (h) AR+DMO

Figure 5: Parallel coordinate plots on DTLZ3 with 8 objectives.

5.6. Experimental results of the parallel coordinates

⁶¹⁰ In order to give a more intuitive description of the performance, the parallel coordinate system is designed to show the convergence and distribution of the obtained solutions in high dimensional space. Each line in parallel coordinate represents a point or an individual in high-dimensional objective space. The x-coordinate shows the sequence of objectives, and the y-coordinate shows the ⁶¹⁵ value of each objective. If all objective values of the obtained solutions are between $[0, 1]$, then the convergence of the algorithm is good. If the lines can be evenly distributed in the space of $[0, 1]$, the distribution of the algorithm would be better.

31

(a) RGridEA (b) ϵ-MOEA (c) HypE

(d) NSGA2 (e) POGA (f) SPEA2

(g) AR+CD (h) AR+DMO
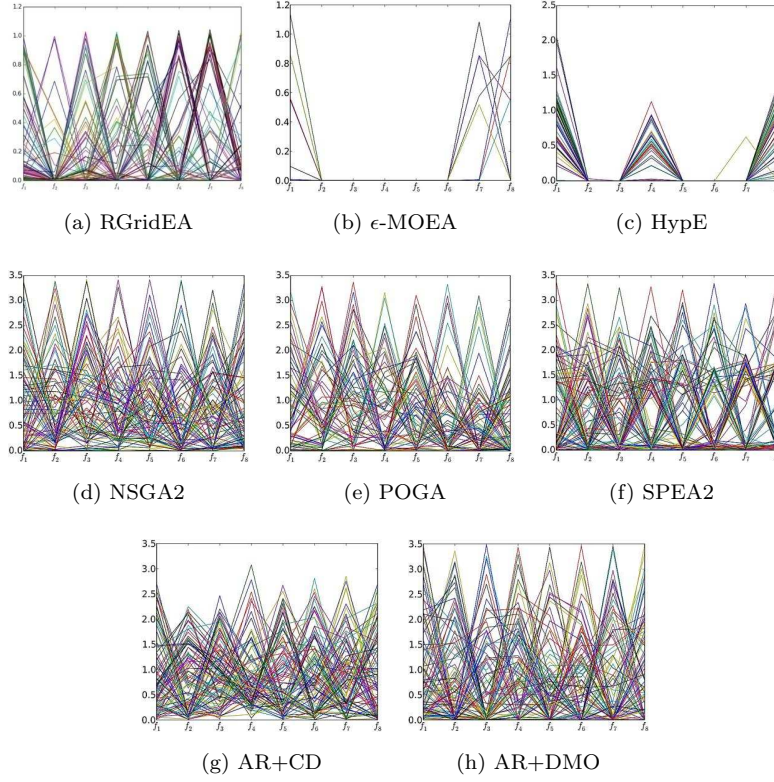
Figure 6: Parallel coordinate plots on DTLZ4 with 8 objectives.

Figure 5 shows the parallel coordinate plots of 8 algorithms in DTLZ3 with 8

620 objectives. DTLZ3 is a test instance designed to be difficult to converge. From the convergence perspective, only RGridEA and ϵ-MOEA can converge to the PF. The solutions of RGridEA are distributed poorly on the first objective, namely $f1$, but evenly on the other objectives. Overall, RGridEA and ϵ-MOEA both perform well.

625 Figure 6 shows the parallel coordinate plots of 8 algorithms in DTLZ4 with 8 objectives. Both RGridEA and ϵ-MOEA have better convergence than the other algorithms. The solutions of RGridEA have completely converged in the POF. The distribution of RGridEA is also good in comparison with other algorithms.

Thus, RGridEA has competitive convergence and distribution on DTLZ4.

6. Conclusion

This paper has proposed a novel many-objective evolutionary algorithm based on rotating grids, which are denoted by RGridEA. The algorithm has three advantages. First, it uses rotating grids to partition the objective space, which can enhance the distribution. Second, it rotates the coordinate and grids to separate the convergence information and distribution information, thereby, avoiding the interaction effect between the convergence and diversity in the original coordinate system. Third, it redefines the stratification mechanism which prevents the elimination of the boundary points in the optimization when relaxing the Pareto dominance relationship.

To demonstrate the strong competitiveness, we have made an extensive experimental comparison of RGridEA with seven algorithms. A number of well-know benchmark problems are chosen to challenge different abilities of the algorithms. In comparison with the other 7 algorithms, it can be concluded that the proposed RGridEA can generally maintain a good balance between convergence and diversity on most problems instances considered in this paper. In the future research, this advanced mechanism of rotated grid will be further extended in solving constrained and dynamic many-objective optimization problems.

7. Acknowledgements

References

[1] FARINA M, AMATO P. On the Optimal Solution Definition for Many-CriteriaOptimization Problems; proceedings of the Proceedings of the NAFIPS-FLINT InternationalConference, F, 2002 [C]. IEEE Service Center.

[2] R.Lakshmi,K.Vivekanandhan,R.Brintha, A New Biological Operator in GeneticAlgorithm for Class Scheduling Problem[J], International Journal of Computer Applications,2012, 60(12) 6-11.

[3] E. J. Hughes, "Radar Waveform Optimisation as a Many-Objective Application Benchmark," in Evolutionary Multi-Criterion Optimization, Matsushima, Japan, 2007, pp. 700-714.

[4] F. di Pierro, "Many-Objective Evolutionary Algorithms and Applications to Water Resources Engineering," Ph.D. Thesis, School of Engineering, Computer Science and Mathematics, University of Exeter, UK, 2006.

[5] P. M. Reed and J. B. Kollat, "Save now, pay later? Multi-period many-objective groundwater monitoring design given systematic model errors and uncertainty," Advances in Water Resources, vol. 35, pp. 55-68, January 2012.

[6] J. G. Herrero, A. Berlanga, and J. M. M. Lopez, "Effective Evolutionary Algorithms for Many-Specifications Attainment: Application to Air Traffic Control Tracking Filters," IEEE Transactions on Evolutionary Computation, vol. 13, pp. 151-168, February 2009.

[7] A. L. Jaimes, A. A. Montano, and C. A. C. Coello, "Preference Incorporation to Solve Many-Objective Airfoil Design Problems," in IEEE Congress on Evolutionary Computation, New Orleans, LA, 2011, pp. 1605-1612.

[8] A. Suflow, N. Drechsler, and R. Drechsler, "Robust Multi-Objective Optimization in High Dimensional Spaces," in Evolutionary Multi-Crierion Optimization, Matsushima, Japan, 2007.

34

[9] D. K. Saxena, J. A. Duro, A. Tiwari, K. Deb, and Q. Zhang, "Objective Reduction in Many-objective Optimization: Linear and Nonlinear Algorithms," Indian Institute of Technology KanGAL Report No.2010008, September 2010.

[10] 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 Transactions on Evolutionary Computation, 2013, in press.

[11] C. A. C. Coello, D. A. V. Veldhuizen, and G. B. Lamont, Evolutionary Algorithms for Solving Multi-Objective Problems. New York: Kluwer Academic, 2002.

[12] H. Ishibuchi, N. Tsukamoto, and Y. Nojima, "Evolutionary Many-Objective Optimization: A Short Review," in IEEE Congress on Evolutionary Computation, Hong Kong, 2008, pp. 2424-2431.

[13] 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, pp. 721-736, 2013.

[14] M. Li, S. Yang, and X. Liu, "A Test Problem for Visual Investigation of High-dimensional Multi-objective Search," in IEEE Congress on Evolutionary Computation, Beijing, China, 2014, in press.

[15] Zou J, Li Q, Yang S, et al. A prediction strategy based on center points and knee points for evolutionary dynamic multi-objective optimization[J]. Applied Soft Computing, 2017, 61:806-818.

[16] Gan R, Yu G, Zheng J, et al. The Effect of Diversity Maintenance on Prediction in Dynamic Multi-objective Optimization[J]. Applied Soft Computing, 2017.

[17] DRECHSLER N, DRECHSLER R, BECKER B. Multi-Objective Optimi-
sation Basedon Relation Favour; proceedings of the Evolutionary Multi-
Criterion Optimization, F, 2001 [C].Springer-Verlag.

[18] IKEDA K, KITA H, KOBAYASHI S. Failure of Pareto-based MOEAs:
DoesNondominated Really Mean Near to Optimal [C]; proceedings of the
IEEE Congress onEvolutionary Computation, F, 2001.

[19] LAUMANNS M, THIELE L, DEB K, et al. Combining Convergence and
Diversity inEvolutionary Multi-Objective Optimization [J]. Evolutionary
Computation, 2002, 10(3):263-82.

[20] David Hadka and Patrick Reed. Diagnostic Assessment of Search Controls
and FailureModes in Many-Objective Evolutionary Optimization, Evolu-
tionary Computation, Vol. 20, No.3, pp. 423–452, Fall 2012.

[21] Salem F. Adra and Peter J. Fleming. Diversity Management in
EvolutionaryMany-Objective Optimization, IEEE Transactions on Evolu-
tionary Computation, Vol. 15, No.2, pp. 183–195, April, 2011.

[22] E. Zitzler and S. Kunzli, "Indicator-based selection in multiobjec-
tivesearch,"in Proc. 8th Int. Conf. Parallel Problem Solving from Na-
ture.2004, pp. 832-842.

[23] ohannes Bader and EckartZitzler. HypE: An Algorithm for Fast
Hypervolume-BasedMany-Objective Optimization, Evolutionary Compu-
tation, Vol. 19, No. 1, pp. 45–76, Spring,2011.

[24] BRADSTREET L, WHILE R L, BARONE L. A Fast Incremental Hyper-
volumeAlgorithm [J]. IEEE Transactions on Evolutionary Computation,
2008, 12(6): 714-23.

[25] ZHANG Q, LI H. MOEA/D: A Multiobjective Evolutionary Algorithm
Based onDecomposition [J]. IEEE Transactions on Evolutionary Compu-
tation, 2007, 11(6): 712-31.

[26] E. J. Hughes, "Multiple single objective Pareto sampling,"in Proc.Congr. Evol. Comput, vol. 4, 2003, pp. 2678-2684.

[27] DEB K, SAXENA D K, On Finding Pareto-Optimal Solutions ThroughDimensionality Reduction for Certain Large-Dimensional Multi-Objective OptimizationProblems KanGAL Report No. 2005011 [R]: Indian Institute of Technology, 2005.

[28] SAXENA D K, DEB K. Non-linear Dimensionality Reduction Procedures forCertain Large-Dimensional Multi-objective Optimization Problems: Employing Correntropyand a Novel Maximum Variance Unfolding [M]. 2007: 772-87.

[29] SaxenaDhish ,Kumar Duro,JoaoA,TiwariAshutosh, Deb Kalyanmoy, Zhang qingfu.Objective reduction in many-objective optimization:linear and nonlinear algorithms, IEEEtransactions on Evolutionary Computation.Vol. 15, No. 2, pp. 1-22, April, 2010.

[30] D. Brockhoff and E. Zitzler. Improving hypervolume based multiobjective evolutionary algorithms by using objective reduction methods[C]. In Congress on Evolutionary Computation (CEC 2007), pages 2086-2093. IEEE Press, 2007.

[31] A.L.Jaines, C.A.C.Coello and D.Chakraborty. Objective reduction using a featureselection technique[C]. In proceedings of the Genetic and Evolutionary Computation Conference (GECCO 08). 673-680, ACM Press, Atlanta, USA, July 2008.

[32] R. C. Purshouse, C. Jalba, and P. J. Fleming, "Preference-driven coevolutionaryalgorithms show promise for many-objective optimisation,"in Proc. Evol. Multi-Criterion Optimization, 2011, pp. 136-150.

[33] FARINA M, AMATO P. On the Optimal Solution Definition for Many-Criteria Optimization Problems; proceedings of the Proceedings of the

NAFIPS-FLINT International Conference, F, 2002 [C]. IEEEServiceCenter.

[765] [34] Hughes E J, Evolutionary Many-objective optimization : Many once or one many[C]. Proc of 2005 IEEE Congress on Evolutionary Computation. Edinburgh : IEEE Service Center, 2005 :222-227

[35] D. W. Corne and J. D. Knowles, "Techniques for Highly MultiobjectiveOptimisation: Some Nondominated Points are Better than Others," in Genetic [770] and Evolutionary Computation Conference, New York, USA, 2007, pp. 773-780.

[36] J. D. Knowles and D. W. Corne, "Quantifying the Effects of Objective Space Dimension in Evolutionary Multiobjective Optimization," in Evolutionary Multi-Criterion Optimization, Matsushima, Japan, 2007, pp. 757-[775] 771.

[37] S. Mostaghim and H. Schmeck, "Distance Based Ranking in Many-Objective Particle Swarm Optimization," in Parallel Problem Solving from Nature, Dortmund, Germany, 2008, pp. 753-762.

[38] R. C. Purshouse, C. Jalba, and P. J. Fleming, "Preference-driven coevolu-[780] tionary algorithms show promise for many-objective optimisation,"in Proc. Evol. Multi-Criterion Optimization, 2011, pp. 136-150.

[39] Wagner T, Beume N, Naujoks B. Pareto-aggregation and indicator-based methods in many-objective optimization. In Proc 4th Int.Conf. Evol. Multi-Criterion Optimization, S. Obayashi, K. Deb, C. Poloni,T. Hiroyasu, and [785] T. Murata, Eds. New York: Springer-Verlag, Mar. 2007, vol. 4403, pp. 742 -756

[40] S. F. Adra and P. J. Fleming, "A Diversity Management Operator for Evolutionary Many-Objective Optimisation," in Evolutionary Multi-Criterion Optimization, Nantes, France, 2009, pp. 81-94.

[41] 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, pp. 348-365, 2014.

[42] E. J. Hughes, "Fitness Assignment Methods for Many-Objective Problems," in Multiobjective Problem Solving from Nature, J. Knowles, et al., Eds., ed: Springer, 2008, pp. 307-329.

[43] R. C. Purshouse, C. Jalba, and P. J. Fleming, "Preference-driven coevolutionary algorithms show promise for many-objective optimisation,"in Proc. Evol. Multi-Criterion Optimization, 2011, pp. 136-150.

[44] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A Fast and Elitist Multi-objective Genetic Algorithm: NSGA-II, IEEE Transactions on Evolutionary Computation, 6(2): 182-197, 2002

[45] Laumanns M, Thiele L, Deb K, Zitzler E. Combining Convergence and Diversity in Evolutionary Multi-objective Optimization. Evolutionary Computation, Vol.10, No.3, pp.263-282, Fall 2002.

[46] M. Li, J. Zheng, K. Li, Q. Yuan, and R. Shen, "Enhancing Diversity for Average Ranking Method in Evolutionary Many-Objective Optimization," in Parallel Problem Solving from Nature, Krakow, Poland, 2010, pp. 647-656.

[47] . Wagner, N. Beume, and B. Naujoks, "Pareto-, Aggregation-, and Indicator-Based Methods in Many-Objective Optimization," in Evolutionary Multi-Criterion Optimization, Matsushima, Japan, 2007, pp. 742-756.

[48] J. Bader and E. Zitzler, "HypE: An Algorithm for Fast Hypervolume-Based Many-Objective Optimization," Evolutionary Computation, vol. 19, pp. 45-76, 2011.

[49] F. di Pierro, S.-T. Khu, and D. A. Savic, "An Investigation on Preference Order Ranking Scheme for Multiobjective Evolutionary Optimiza-

tion," IEEE Transactions on Evolutionary Computation, vol. 11, pp. 17-45, 2007.

[50] Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the strength Pareto evolutionary algorithm. TIK-Report 103, 2001

[51] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable Test Problems for Evolutionary Multiobjective Optimization," in Evolutionary Multiobjective Optimization, A. Abraham, et al., Eds., ed: Springer, 2005, pp. 105-145.

[52] M. Li, S. Yang, and X. Liu, "Diversity Comparison of Pareto Front Approximations in Many-Objective Optimization," IEEE Transactions on Cybernetics, 2014, in press.

[53] P. A. N. Bosman and D. Thierens, "The Balance Between Proximity and Diversity in Multi-objective Evolutionary Algorithms," IEEE Transactions on Evolutionary Computation, vol. 7, pp. 174-188, April 2003.

[54] K. Deb and S. Jain, "Running Performance Metrics for Evolutionary Multi-Objective Optimization," in Genetic and Evolutionary Computation Conference, Orchid Country Club, Singapore, 2002, pp. 13-20.

[55] A. C. Tamhane, "Multiple Comparisons in Model I One-Way ANOVA with Unequal Variances," Communications in Statistics, vol. 6, pp. 15-32, 1977

[56] Hajela P, Lin C Y. Genetic search strategie in multicriterion optimal design[J].Structural and Multidisciplinary Optimization, 1992, 4(2):99-107

[57] Schaffer J D. Multiple objective optimization with vector evaluated genetic algorithms[C]. Proc of 1st Int Conf on Genetic Algorithms and Their Application. Hillsdale :L. Erlbaum Associates Inc, 1985 : 93-100.

[58] R. C. Purshouse, C. Jalba, and P. J. Fleming, "Preference-driven coevolutionary algorithms show promise for many-objective optimisation,"in Proc. Evol. Multi-Criterion Optimization, 2011, pp. 136-150.

[59] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable Test Problems for Evolutionary Multiobjective Optimization," in Evolutionary Multiobjective Optimization, A. Abraham, et al., Eds., ed: Springer, 2005, pp. 105-145.

[60] S. Huband, P. Hingston, L. Barone, and L. While, "A Review of Multiobjective Test Problems and a Scalable Test Problem Toolkit," IEEE Transactions on Evolutionary Computation, vol. 10, pp. 477-506, 2006.

[61] Z. Kang, L. Kang, X. Zou et al., A New Evolutionary Decision Theory for Many-Objective Optimization Problems. pp. 1-11.

[62] Hiroyuki Sato, Hern E. Aguirre, Kiyoshi Tanaka, Self-Controlling Dominance Area of Solutions in Evolutionary Many-Objective Optimization, 8th International Conference, SEAL 2010, pp 455-465

[63] Adra, S.F., Fleming, P.J.: A Diversity Management Operator for Evolutionary Many-Objective Optimisation. In: Ehrgott, M., Fonseca, C.M., Gandibleux, X., Hao, J.-K., Sevaux, M. (eds.) EMO 2009. LNCS, vol. 5467, pp. 81C94. Springer, Heidelberg (2009)

[64] Gan R, Yu G, Zheng J, et al. The Effect of Diversity Maintenance on Prediction in Dynamic Multi-objective Optimization[J]. Applied Soft Computing, 2017.