REGULAR RESEARCH PAPER

Balancing performance between the decision space and the objective space in multimodal multiobjective optimization

Qite Yang[1] · Zhenkun Wang[2] · Jianping Luo[3] · Qiang He[4]

Abstract

Many multimodal multiobjective optimization algorithms aim to find as many Pareto optimal solutions as possible while the performance in the objective space is despised. More seriously, some algorithms even overemphasize the diversity of solution set in the decision space at the cost of convergence. How to improve convergence and diversity simultaneously is an important issue when solving multimodal multiobjective optimization problems. In this paper, we propose an evolutionary multiobjective optimization algorithm with a decomposition strategy in the decision space (EMO-DD). A decision subregion allocation and diversity archive preservation methods are proposed to promote the diversity of solutions in the decision space. Meanwhile, a bi-objective optimization problem is formulated for screening for solutions with great convergence and diversity. Combining a modified mating selection method, well-performed solutions both on the convergence and diversity are preserved and inherited. The performance of EMO-DD is compared with five state-of-the-art algorithms on fifteen test problems. The experimental results show that EMO-DD can solve multimodal multiobjective optimization problems, and can improve the performance of the solution set in both the decision and objective spaces.

1 Introduction

Real-world problems usually involve multiple conflicting objectives, which are known as multiobjective optimization problems (MOPs) [1–5]. A minimization MOP with m objec-

✉ Zhenkun Wang
  wangzk3@sustech.edu.cn

  Qite Yang
  yangqt@mail.sustech.edu.cn

  Jianping Luo
  ljp@szu.edu.cn

  Qiang He
  heqiang@bmie.neu.edu.cn

[1] School of System Design and Intelligent Manufacturing, Southern University of Science and Technology, Shenzhen, China

[2] School of System Design and Intelligent Manufacturing and Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, China

[3] College of Information Engineering, Shenzhen University, Shenzhen, China

[4] College of Medicine and Biological Information Engineering, Northeastern University, Shenyang, China

tives can be described as:

$$\text{minimize} \quad \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})),$$
$$\text{subject to} \quad \mathbf{x} \in \Omega. \tag{1}$$

where Ω is the decision space, $\mathbf{x} = (x_1, \dots, x_n) \in \Omega$ is a candidate solution; $\mathbf{f} : \Omega \rightarrow R^m$ constitutes m real-value objective functions; R^m denotes the objective space. Usually there is no unique optimal solution which simultaneously optimizes all the m objective functions. Instead, there is a set of trade-off solutions called Pareto optimal solutions. The identification of the Pareto optimal solution is based on the *Pareto dominance relationship*. Given two solutions $\mathbf{x}$ and $\mathbf{y}$, $\mathbf{x}$ is said to Pareto dominate $\mathbf{y}$ ($\mathbf{x} \succ \mathbf{y}$) if $f_i(\mathbf{x}) \leq f_i(\mathbf{y})$ for $\forall i \in \{1, \dots, m\}$ and $\exists j \in \{1, \dots, m\}$ satisfies $f_j(\mathbf{x}) < f_j(\mathbf{y})$. A solution is said to be the Pareto optimal solution if it cannot be dominated by any other solution. The set of all Pareto optimal solutions is called the Pareto set (PS), and the set of their corresponding objective vectors is called the Pareto front (PF).

The multimodal multiobjective optimization problem (MMOP) is a special kind of MOP whose PF corresponds to multiple PSs. That is, multiple Pareto optimal solutions are corresponding to the same objective vector [6–8]. These
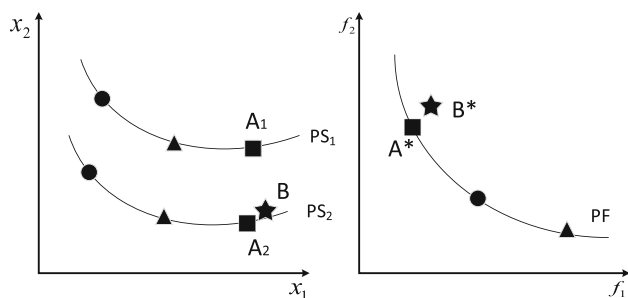
Fig. 1 Illustration of MMOP

Pareto optimal solutions are equivalent to each other. In fact, the MMOP is often encountered in real-world applications, such as engine design [9–11], game map generation [12] and distillation plant layout [13]. For example, a product engineer is mainly interested in a solution set through covering the whole PSs since it can help to know if a certain design can be realized by different parameters setting of the production process.

Figure 1 illustrates an example of the MMOP problem. Two PSs, PS_1 and PS_2 in the decision space correspond to the same PF in the objective space. A_1 and A_2 are two equivalent Pareto optimal solutions corresponding to A^* in the objective space. B is an approximate solution corresponding to B^* in the objective space. Suppose that there are two solutions A_1 and B in the current population, then B will be abandoned by general MOEAs since its objective vector B^* is too close to A^* and dominated by A^*. In the multimodal multiobjective optimization scenario, the preservation of B may be helpful to find A_2 or other Pareto optimal solutions on PS_2. Furthermore, it is also beneficial to improve the diversity of the whole solution set.

In recent decades, various promising multiobjective evolutionary algorithms (MOEAs) [14–19] have been proposed. These algorithms can search for a well-converged solution set with great diversity in the objective space but do not consider the diversity in the decision space at all. Nevertheless, M. Preuss gave two conjectures in [20], in which one is that: diversity maintenance is not only needed in the objective space but also the decision space. To obtain a solution set with better diversity in the decision space when solving MMOPs, many algorithms have been proposed based on heuristic algorithms in recent years, including Omni-optimizer [21] and DN-NSGA-II [22]. But these algorithms still have some limitations. For example, too much attention is paid to the diversity of the solution set in the decision space but the performance in the objective space is neglected. A solution set with insufficient convergence is meaningless, even if its diversity is very good. The abilities that an algorithm needs for solving MMOPs were illustrated in [23]. Firstly, it can find solutions with good convergence. Then it should have the ability to find diverse solutions in the decision and objective

spaces. How to make the solution set as diverse as possible and improve its convergence at the same time is a very significant issue.

This paper proposes an evolutionary optimization algorithm with decomposition in the decision space (EMO-DD) to solve MMOPs. In EMO-DD, a local optimization method is adopted within an evolutionary framework. This paper aims to explore the relationship of the solution set between its convergence and diversity in both the decision and objective spaces, and promote these two performances simultaneously.

Uniform reference vectors have an inherent property of reflecting great diversity. There are already a lot of MOEAs using predefined reference vectors to maintain the diversity of the solution set in the objective space. However, the advantage of reference vectors in diversity maintenance in the decision space is not utilized. Inspired by the great performance in the objective space, we attempt to utilize the property of uniform reference vectors in the decision space for solving MMOPs. In this paper, a well-distributed reference vector set is predefined in the decision space and decomposes it into several subregions. In each subregion, the solutions with excellent diversity in the decision space are retained.

Considering the maintenance of convergence and diversity in the decision space simultaneously, we attempt to quantify them into two different objectives and formulate them into a new bi-objective optimization problem. By optimizing this problem, well-converged solutions with good diversity in the decision space can be screened out. In each subregion, a special local optimization method is constructed, and the optimal solutions to the bi-objective problem are seemed as ones with good convergence and diversity in the decision space, and retained for the next generation evolution. It is well-known that the search ability of evolutionary algorithms can be improved by local optimization, which is one of the commonly used techniques in memetics. After maintaining the diversity of solution set in the decision space as much as possible, we select the solutions with better diversity in the objective space to achieve the balance between the two spaces.

The main contribution of this paper can be summarized as follows:

1. A decomposition method in the decision space is introduced to maintain the diversity of solution set in the decision space.
2. A bi-objective optimization problem is formulated to construct the relationship of solution set between its convergence and diversity.
3. A diversity archive and modified mating selection method are proposed to preserve and inherit solutions that perform better in both the decision and the objective space.

Systematic experiments are carried out on fifteen test problems. Compared with four other state-of-the-art multimodal multiobjective optimization algorithms, the experimental results show that the solution set obtained by EMO-DD has good diversity as well as great convergence.

The remainder of this paper is organized as below. Section 2 gives the related works on multimodal multiobjective optimization. Then the details of EMO-DD are described in Sect. 3. Section 4 presents the experimental study. Section 5 concludes this paper.

2 Related work

2.1 Prior works on multimodal multiobjective optimization

Most MOEAs paid attention to approximate the PF in the objective space rather than the diversity of PSs in the decision space. However, convergence and diversity maintenance in both the decision and objective spaces are very important for a good solution set. In recent years, a lot of works have been done on promoting the diversity of solutions in the decision space. Deb [21] proposed an algorithm called Omni-optimizer to solve MMOPs, in which the crowding distance in the decision space was used to maintain the diversity of solutions. Liang proposed an algorithm based on niching multiobjective optimization called DN-NSGA-II to locate the multiple PSs in the decision space. At the same time, two new multimodal multiobjective optimization problems called SS-UF1 and S-UF3 were proposed, which were the precursors of MMF [24]. Apart from these, many niching-based methods [25–28] have been proposed for solving MMOPs recently, such as fitness sharing [29] and crowding [30]. Lin [31] proposed an algorithm (MMOEA/DC) with dual clustering in the decision and objective spaces, in which both the global PSs and some good local PSs with acceptable quality were maintained. Liu et al. [32] proposed a general framework using two archives to guide the evolutionary search (TriMOEA-TA&R). The diversity archive simultaneously employed a clustering strategy to guarantee diversity in the objective space, and a niche-based clearing strategy to promote the same in the decision space. Xia [33] suggested a framework which included two crowding estimation methods, multiple selection methods for mating and search strategy for variation, to improve the MOEA's searching ability and the diversity of its solutions. In [34], a niching sharing method was proposed for maintaining diversity in both the decision and objective spaces. However, these niching-based algorithms have potential problems. Firstly, they need extra niche parameters provided by decision-makers, and the value of these parameters has a great impact on the performance of algorithms. In addition, some niching-based algorithms

even need extra fitness evaluations to maintain niche detection accuracy [35].

There are also many other MOEAs proposed for MMOPs in recent years. In [36], a new crossover method and two archives were adopted for maintaining the diversity of solution set in the objective and decision space. Liang et al. [37] proposed a new differential evolution optimization algorithm (MMODE) for MMOPs. A new decision-variable pre-selection mechanism and mutation-bound process were introduced for promoting the diversity of solutions. Tanabe and Ishibuchi [38] proposed a new decomposition-based MOEA, in which each decomposed subproblem can be assigned more than one solution to diversity.

In addition to the above studies, there are some researches on multimodal multiobjective optimization based on particle swarm optimization (PSO) [39–42]. PSO is a population-based optimization algorithm that simulating the swarm behaviour of insects and birds. These groups search for food in a cooperative way. Each member of the swarm changes its search mode by learning its own and other members' experiences. To solve MMOPs, Yue proposed a multiobjective particle swarm optimizer using ring topology and special crowding distance (MO_Ring_PSO_SCD), in which the ring topology helps to induce stable niches to locate much more Pareto optimal solutions, and the special crowding distance considers the crowding distance in both the decision and objectives spaces to maintain multiple PSs. Qu proposed a self-organized speciation based multiobjective particle swarm optimizer [43] to locate multiple Pareto optimal solutions for solving MMOPs, in which the self-organized mechanism is used for improving the efficiency of the species formulation as well as the performance of the algorithm. Liang et al. [44] proposed a new multiobjective PSO with a self-organizing mechanism called SMPSO-MM, in which a self-organizing map network is used to find the diversity structure of the population and build the neighbourhood in the decision space. PSO algorithm is indeed a suitable choice to optimize the MMOPs. But the particles in PSO share information through the current search to the best, and the whole search and update process is following the current optimal solution. So it is a single sharing mechanism to a large extent, which may fall into local optimum.

2.2 Decomposition of multiobjective optimization

The idea of decomposition has been used in many methods for solving MOPs [45–47], in which the most representative are MOEA/D [48] and NSGA-III [49]. By specifying a set of reference vectors in the objective space, MOEA/D decomposes the MOP into several single-objective optimization subproblems and solves them in a collaborative manner. At each generation, the population is composed of the best solutions to each subproblem. Many scalar aggregation functions

can be used as decomposition methods for converting MOPs into several scalar optimization problems. Weighted Sum Approach, Tchebycheff Approach (TCH) [50], and Boundary Intersection Approach are three representatives used in MOEAs with decomposition. In this paper, we use the TCH approach for decomposing a MOP if necessary. The TCH is formed as:

$$\text{minimize} \ \ g^{tch}(\mathbf{x}|\mathbf{w}, \mathbf{z}^*) = \max_{1 \le i \le m} \{\omega_i |f_i(\mathbf{x}) - z_i^*|\},$$

$$\text{subject to} \ \ \mathbf{x} \in \Omega. \tag{2}$$

where $\mathbf{z}^* = (z_1^*, \ldots, z_m^*)$ is the reference point. $\mathbf{w} = (\omega_1, \ldots, \omega_m)$ is a reference vector in the objective space, $\omega_i \ge 0$ for $\forall i \in \{1, \ldots, m\}$ and $\sum_{i=1}^m \omega_i = 1$.

The above approaches all have the ability to decompose the approximation of the PF into several scalar optimization problems. Based on a large number of predefined evenly distributed reference vectors, solutions are guided to converge to the scaler problem corresponding to its bounded reference vector. All optimized scalar problem solutions can compose into a well-distributed set. This decomposition method does have great performance both on convergence and diversity in the objective space, while the diversity of solution set in the decision space is rarely considered. This makes it possible that the obtained solutions are only a local set but cannot represent the whole PSs.

3 Proposed algorithm

In this section, we present the details of our proposed algorithm, EMO-DD. It is composed of four main components: the decision subregion allocation, the update mechanism of the diversity archive, the environmental selection, and the mating selection. The details are shown in the following subsections.

3.1 General framework

To maintain the diversity of solutions in the decision space, we borrow the idea from decomposition-based MOEAs to divide the decision space into several subregions and optimize the solutions in different subregions collaboratively. To make sure the solutions with good convergence and diversity can be preserved and inherited, we propose a new diversity archive to store them, and use a modified mating selection method to select parents from them.

The overall flowchart of EMO-DD is presented in Fig. 2. The EMO-DD starts with an initial parent population P and diversity archive DA. Then the offspring population is reproduced by P and DA while the stopping criterion is not met. After the offspring population reproduction, all solutions are
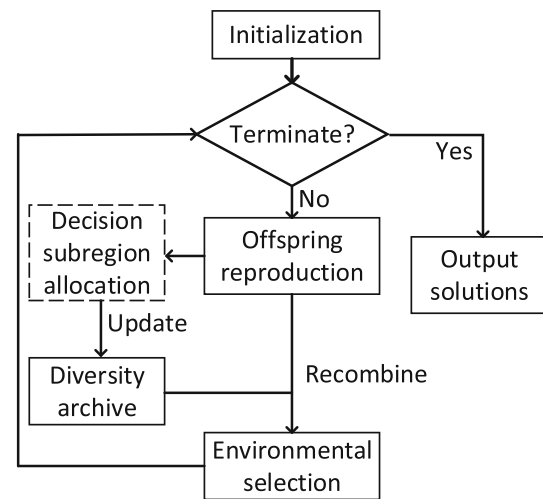

Fig. 2 Flowchart of EMO-DD

allocated into different decision subregions (Sect. 3.2), and both DA and P are updated based on it (Sects. 3.3 and 3.4). The population P will be reported in the last generation.

3.2 Decision subregion allocation

Before the decision subregion allocation, we need to normalize each solution $\mathbf{x}$ to make sure it has reached the requirement to be assigned subregion. Normalization can eliminate the effects of different ranges or scales. The normalized solution $\hat{\mathbf{x}} = (\hat{x}_1, \ldots, \hat{x}_n)$ is defined as:

$$\hat{x}_i = \frac{x_i - l_i + \varphi}{u_i - l_i + \varphi}, \ i \in \{, 2, \ldots, n\}. \tag{3}$$

where l_i and u_i are respectively the minimum and maximum values of the i-th variable among all solutions. To meet the case where $l_i = u_i$, a minimal positive number φ is used here. In this paper, we set it to 10^{-5}. After normalization, $\hat{\mathbf{x}}$ is in $(0, 1]^n$.

To allocate each solution into different decision subregions, we use N well-distributed reference vectors $WX = \{\mathbf{wx}^1, \ldots, \mathbf{wx}^N\}$ on the canonical simplex to divide the decision space into N subregions. The reference vectors can be generated by the same method used in decomposition-based MOEAs, i.e., the two-layer weight vector generation method [51]. Specifically, for each reference vector $\mathbf{wx}^i$, we assign it a subregion Δ^i in decision space as follows:

$$\Delta^i = \{\mathbf{x} \in \Omega | \langle \hat{\mathbf{x}}, \mathbf{wx}^i \rangle \le \langle \hat{\mathbf{x}}, \mathbf{wx}^j \rangle\}, \ j \in \{1, \ldots, N\} \tag{4}$$

where $\langle \hat{\mathbf{x}}, \mathbf{wx} \rangle$ is the acute angle between $\hat{\mathbf{x}}$ and $\mathbf{wx}$. By using this definition, each solution $\mathbf{x}$ of a population can easily be resided in its unique subregion Δ^k, whose index k is defined

as:

$$k = \underset{i \in \{1,\ldots,N\}}{\arg\min} \langle \hat{\mathbf{x}}, \mathbf{wx}^i \rangle \tag{5}$$

The pseudo-code of decision subregion allocation mechanism is given in Algorithm 1.

Algorithm 1 Decision Subregion Allocation (P, WX)

Input: population P, reference vector set WX
Output: solutions in the ith subregion Δ^i
1: **for each** $\mathbf{x} \in P$ **do**
2:    **for each** $\mathbf{wx} \in WX$ **do**
3:       Compute the acute angle $\langle \hat{\mathbf{x}}, \mathbf{wx} \rangle$;
4:    **end for**
5:    $k = \underset{i \in \{1,\ldots,N\}}{\arg\min} \langle \hat{\mathbf{x}}, \mathbf{wx}^i \rangle$;
6:    $\Delta^k = \Delta^k \bigcup \{\mathbf{x}\}$;
7: **end for**
8: **return** $\Delta^1, \ldots, \Delta^N$;

3.3 Update mechanism of the DA

To keep solutions with good diversity in the decision space when searching for the Pareto optimal solutions in MMOPs, we introduce a temporary diversity archive DA. The DA aims at preserving solutions as much convergent and diversified as possible in the decision space. Algorithm 2 is the pseudo-code of the update mechanism of the DA. After the reproduction mechanism, the offspring population Q is incorporated into the diversity archive DA. Then solutions in DA reside in the corresponding decision subregions (line 2). For each subregion Δ_i, the best solution is chosen into a temporary archive S and deleted from it if it is not empty (lines 3–8). In this process, the idea of MOEA/D is adopted to decompose the objective space and TCH function is used as the indicator of each candidate solution. Solution with minimum TCH value is chosen as the best one in each subregion. Note that any other scalarizing functions or fitness assignment can also be adopted as the indicator, i.e., the BPI approach used in [48] or fitness assignment used in [52].

After the above selection operation, if the size of S is less than N, then a new bi-objective optimization problem is formulated as follows:

minimize $\mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}), g_2(\mathbf{x}))$,

where

$$\begin{cases} g_1(\mathbf{x}) = 1/\left(\underset{\mathbf{y} \in \Delta^i, \mathbf{y} \neq \mathbf{x}}{\arg\min} ||\mathbf{x} - \mathbf{y}|| \right) \\ g_2(\mathbf{x}) = g^{tch}(\mathbf{x}|\lambda, \mathbf{z}^*) \end{cases}$$
$$\mathbf{x} \in \Delta^i, i = \{1, \ldots, N\}. \tag{6}$$

Algorithm 2 Update Mechanism of DA (DA, Q, WX, W)

Input: diversity archive DA, offspring population Q, reference vector set WX, reference vector set $W = \{\mathbf{w}^1, \ldots, \mathbf{w}^N\}$
Output: updated DA
1: $S = \emptyset$, $DA = DA \bigcup Q$;
2: Decision Subregion Allocation(DA, WX);
3: **for each** $i \in \{1, \ldots, N\}$ **do**
4:    **if** $\Delta^i \neq \emptyset$ **then**
5:       find best solution $\mathbf{x}$ in Δ^i;
6:       $S = S \bigcup \{\mathbf{x}\}$, $\Delta^i = \Delta^i \setminus \{\mathbf{x}\}$, $DA = DA \setminus \{\mathbf{x}\}$;
7:    **end if**
8: **end for**
9: **if** $|S| < N$ **then**
10:   **for each** $i \in \{1, \ldots, N\}$ **do**
11:     **for each** $\mathbf{x} \in \Delta^i$ **do**
12:       $g_1(\mathbf{x}) = 1/\left(\underset{\mathbf{y} \in \Delta^i, \mathbf{y} \neq \mathbf{x}}{\arg\min} ||\mathbf{x} - \mathbf{y}|| \right)$;
13:       $g_2(\mathbf{x}) = g^{tch}(\mathbf{x}|\mathbf{w}, \mathbf{z}^*)$;
14:     **end for**
15:   **end for**
16:   Use non-dominated sorting to divide DA into $\{F_1, F_2, \cdots\}$ based on g_1 and g_2 defined in Eq. (6);
17:   $i = 0$;
18:   **while** $|S| < N$ **do**
19:     $i = i + 1$, $S = S \bigcup F_i$;
20:   **end while**
21:   **while** $|S| > N$ **do**
22:     find worst solution $\mathbf{x}$ in F_i;
23:     $S = S \setminus \{\mathbf{x}\}$, $F_i = F_i \setminus \{\mathbf{x}\}$;
24:   **end while**
25: **end if**
26: $DA = S$;
27: **return** DA;

In Eq. (6), $g_1(\mathbf{x})$ quantifies the distribution density of solution $\mathbf{x}$ in the decision space, and $g_2(\mathbf{x})$ reflects its convergence degree. The optimal solutions to this bi-objective problem represent the most decentralized and best convergent ones. We do not mention the diversity of solution set in the objective space here does not mean it is unimportant. A solution set, which is widely distributed in the decision space and reflects the whole PSs, will not have a bad diversity in the objective space. In other words, the optimal solutions to Eq. (6) are also widely distributed in the objective space to a certain extent.

Same as the best solution selection indicator mentioned above, $g_2(\mathbf{x})$ can also be defined in more than one way. Based on Eq. (6), all solutions in each subregion are reformulated (lines 10–15). Then the non-dominated sorting is used to divide all solutions into several fronts (line 16). Starting from the first front, non-dominated solutions are chosen into S front by front until $|S| \geq N$ (lines 17–20). Then exceeded solutions in S are abandoned. In this paper, we choose solutions with the largest $g_2(\mathbf{x})$ as the worst ones and delete them from S (lines 21–24). Finally, DA is updated by assigning S to it.
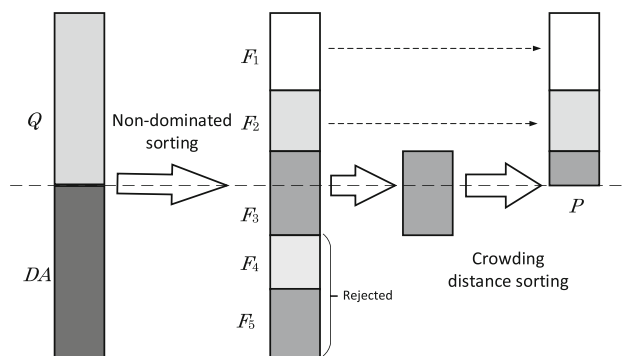
Fig. 3 Environmental selection

3.4 Environmental selection

After the update mechanism of diversity archive DA, we choose solutions with great diversity in the objective space as well as good convergence from the offspring population Q and DA. Figure 3 gives an illustration diagram about the environmental selection procedure. Firstly, a combined population of size $2N$ is formed based on DA and Q. Then we use the non-dominated sorting and crowding distance sorting procedure to select N best solutions into the next generation population P. Algorithm 3 gives the pseudo-code of the environmental selection procedure.

Algorithm 3 Environmental Selection (DA, Q, WX)

Input: diversity archive DA, offspring population Q, reference vector set WX
Output: updated population P
1: $P = \emptyset$, $Q = Q \bigcup DA$;
2: **if** $|Q| = N$ **then**
3:    $P = Q$;
4: **else**
5:    Use non-dominated sorting to divide Q into $\{F_1, F_2, \cdots\}$;
6:    $i = 0$;
7:    **while** $|P| < N$ **do**
8:      $i = i + 1$, $P = P \bigcup F_i$;
9:    **end while**
10:   $CrowDis$ = Crowding Distance Assignment (F_i);
11:   **while** $|P| > N$ **do**
12:     $\mathbf{x} = \underset{\mathbf{x} \in F_i}{\arg\min} \, CrowDis$;
13:     $P = P \setminus \{\mathbf{x}\}$, $F_i = F_i \setminus \{\mathbf{x}\}$;
14:   **end while**
15: **end if**
16: **return** P;

3.5 Mating selection

Mating selection plays an important role in the offspring reproduction process. It is usually implemented by selecting promising solutions from the current population to form a mating pool [53]. In this paper, we adopt the mating selec-

tion method derived from Li et al. [54] by selecting parents from both the diversity archive DA and the current population P. Algorithm 4 is the pseudo-code of mating selection procedure. Firstly, the proportions of non-dominated solutions in the DA (ρ_{da}) and P (ρ_p) are evaluated (lines 1 to 2). $\rho_{da} > \rho_p$ means the convergence of DA is better than P, then the first parent $\mathbf{p}_1$ is randomly selected from DA; Otherwise it is selected from P (lines 3 to 7). The selection of the second parent $\mathbf{p}_2$ depends on ρ_{da}. A random decimal between 0 and 1 is generated, and $\mathbf{p}_2$ comes from DA if ρ_{da} is larger than it; or $\mathbf{p}_2$ is selected from P (lines 8 to 12). By this mating selection, solutions with good diversity in DA can have a greater probability to be inherited without neglecting the convergence.

Algorithm 4 Mating Selection (DA, P)

Input: DA, population P
Output: Mating parents $\mathbf{p}_1$, $\mathbf{p}_2$
1: ρ_{da} = proportion of non-dominated solutions in DA;
2: ρ_p = proportion of non-dominated solutions in population P;
3: **if** $\rho_{da} > \rho_p$ **then**
4:   $\mathbf{p}_1$ = Tournament Selection from DA;
5: **else**
6:   $\mathbf{p}_1$ = Tournament Selection from P;
7: **end if**
8: **if** $rand < \rho_{da}$ **then**
9:   $\mathbf{p}_2$ = Tournament Selection from DA;
10: **else**
11:   $\mathbf{p}_2$ = Tournament Selection from P;
12: **end if**
13: **return** $\mathbf{p}_1$, $\mathbf{p}_2$;

---

4 Experimental study

In this section, contrast experiments are conducted to investigate the performance of the proposed algorithm. At first, test benchmark problems are given. Then the pros and cons of the three metrics used in the experiments are illustrated. Comparative experimental results and analyses are given in the last subsection.

4.1 Test problems

In this paper, we used the following fifteen multimodal multiobjective test problems to test the performance of our proposed algorithm including eight MMF problems (MMF1-8), four IDMP problems [55], two SYM-PART problems [56] and two-objective Omni-test problem [21]. Table 1 shows their different geometric characteristics, including the number of decision variables, objectives, and PSs. Among these test problems, MMF suit is simpler than others. It is worth noting that the PSs of MMF3 and MMF6 are overlapped

Table 1 Characteristics of fifteen multimodal multiobjective test problems

Test problem	Number of objectives	Number of decision variables	Number of Pareto sets
MMF1	2	2	2
MMF2	2	2	2
MMF3	2	2	2
MMF4	2	2	4
MMF5	2	2	4
MMF6	2	2	4
MMF7	2	2	2
MMF8	2	2	4
Onmi-test	2	3	27
SYM-PART-simple	2	2	9
SYM-PART-rotated	2	2	9
IDMP-T1	2	2	2
IDMP-T2	2	2	2
IDMP-T3	2	2	2
IDMP-T4	2	2	2

in every dimension. The difficulty of Omni-test stems from its large number of PSs. SYM-PART-rotated is a variant of SYM-PART-simple by simply rotating all the PSs. Rather than others, the complexity of finding equivalent PSs are different on IDMP test problems. So it is hard for those convergence-first MOEAs to find all PSs on IDMP.

4.2 Performance metrics

HV [57] and IGD [58] are two main metrics used in MOEAs since they can reflect the diversity and convergence of the obtained solution set in the objective space simultaneously. But they do not apply to multimodal multiobjective problems because they evaluate the performance of algorithms that focuses on the objective space while the decision space is ignored [59].

IGDX [60] is a variant of IGD, and it represents the average Euclidean distance between the obtained solutions and Pareto optimal solutions in the decision space. By measuring this distance, IGDX can evaluate both the convergence and the diversity in the decision space of the approximate solution set.

However, it is worth noting that the other one of the two conjectures [20] mentioned in Sect. 1 is: there are situations where the PS does not share the aspired nice properties of the received PF the user normally pay the attention to. This means a solution set well-performed in the decision space does not imply a well-performance in the objective space. According to the example given in [23], a solution set uniformly distributed on all may correspond to a few objective vectors. This example gave us a hint, that a solution set with very bad diversity in the objective space may still get a great

IGDX value. So IGDX is not an applicable metric used alone in multimodal multiobjective optimization.

Recently, Liu et al. [32] proposed a new metric for multimodal multiobjective optimization called IGDM. IGDM is a metric that integrating IGD and IDX, and considering performance not only in the objective space but also in the decision space. The smaller the IGDM value is, the more diverse and convergent the solution set performs in both the objective and decision spaces. This implies that IGDM is a comprehensive performance metric, which reflects the global performance in the two spaces by a numerical value. Per contra, IGDM cannot explain the specific quality of the solution set in any single space. In order to explore the specific performance of the algorithms in dealing with MMOPs, we use IGD, IGDX and IGDM metrics for assessing the performance, respectively.

Considering $F^* = \{\mathbf{f}_1^*, \ldots, \mathbf{f}_q^*\}$ is a set of well distributed reference points on the true PF, where q is the number of these points. Then the corresponded set of Pareto optimal solutions is $X^* = \{\mathbf{x}_{1,1}^*, \mathbf{x}_{1,2}^*, \ldots, \mathbf{x}_{1,a_1}^*, \mathbf{x}_{2,1}^*, \ldots, \mathbf{x}_{q,a_q}^*\}$, where $a_i, i = 1, 2, \ldots, q$ is the number of Pareto optimal solutions to each point in F^*. For an approximate solution set $P = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ obtained by an optimization algorithm, the IGD is defined as:

$$\text{IGD}(P, F^*) = \frac{1}{|F^*|}\left\{\sum_{\mathbf{f}^* \in F^*} \min_{\mathbf{x} \in P}(ed_f(\mathbf{f}(\mathbf{x}), \mathbf{f}^*))\right\}, \quad (7)$$

where $ed_f(\mathbf{f}(\mathbf{x}), \mathbf{f}^*)$ is the Euclidean distance between $\mathbf{f}(\mathbf{x})$ and $\mathbf{f}^*$ in the objective space.

The IGDX is defined as:

$$IGDX(P, X^*) = \frac{1}{|X^*|} \left\{ \sum_{\mathbf{x}^* \in X^*} \min_{\mathbf{x} \in P}(ed_x(\mathbf{x}, \mathbf{x}^*)) \right\}, \qquad (8)$$

where $ed_x(\mathbf{x}, \mathbf{x}^*)$ is the Euclidean distance between $\mathbf{x}$ and $\mathbf{x}^*$ in the decision space.

The IGDM is defined as:

$$IGDM(P, F^*, X^*) = \frac{1}{|X^*|} \left\{ \sum_{\mathbf{f}_i^* \in F^*} \sum_{j=1,...,a_i} d(\mathbf{f}_i^*, \mathbf{x}_{i,j}^*, P) \right\}, \qquad (9)$$

where

$$d(\mathbf{f}_i^*, \mathbf{x}_{i,j}^*, P) = \begin{cases} d_{max}, & \text{if } P_{i,j} = \emptyset, \\ \min\{d_{max}, ed(\mathbf{f}_i^*, P_{i,j})\}, & \text{if } P_{i,j} \neq \emptyset; \end{cases}$$
$$P_{i,j} = \{\mathbf{x}_k : j = \underset{l=1,...,a_i}{\text{argmin}} \, ed_x(\mathbf{x}_{i,l}^*, \mathbf{x}_k), \mathbf{x}_k \in P\}. \qquad (10)$$

The $ed(\mathbf{f}_i^*, P_{i,j})$ in Eq. (10) is the minimal Euclidean distance between $\mathbf{f}_i^*$ and $P_{i,j}$. d_{max} is a parameter defined by the user, and is set to 1 in this paper.

4.3 Comparison results with other algorithms

To demonstrate the effectiveness of EMO-DD, four state-of-the-art multimodal multiobjective optimization algorithms are chosen as comparative algorithms: Omni-optimizer [21], DN-NSGA-II, MO_Ring_PSO_SCD, and TriMOEA-TA&R [32]. In addition, MOEA/D-TCH is adopted for comparing their performance in the objective space. Simulated binary crossover [61] and polynomial mutation [62] operators are used (except in MO_Ring_PSO_SCD) with a distribution index of 20. The crossover probability is 1.0, and the mutation probability is 0.2. The generation of each run is set to 100, and the population size is set to 800. The crowding factor in DN-NSGA-II is set to half of the population size according to [22]. In TriMOEA-TA&R, p_{con}, σ_{niche} and ϵ_{peak} are 0, 0.5 and 0.01, respectively. For each test problem, 30 independent runs are implemented on PlatEMO [63] with Matlab code. The experimental results of IGD, IGDX and IGDM are shown in Table 2, 3 and 4, respectively, where the best mean and standard deviation are highlighted in bold and the second best are colored in italic.

Table 2 shows that all IGD results obtained by EMO-DD are highlighted in bold and italic. Other algorithms except MOEA/D-TCH have significantly worse IGD results than EMO-DD. This means that the solution sets obtained by EMO-DD are more convergent and diverse in the objective space than other compared multimodal multiobjective optimization algorithms. Comparing with MOEA/D-TCH, we

can see EMO-DD is not worse in convergence, even better on some problems. The evaluations of IGD are not very ideal for the remaining four algorithms, extremely on Omni-optimizer, DN-NSGA-II and MO_Ring_PSO_SCD.

Table 3 shows that EMO-DD and MO_Ring_PSO_SCD are the top two algorithms on the IGDX metric. Most of the IGDX values obtained by these two algorithms are highlighted in bold and italic, in which MO_Ring_PSO_SCD prefers MMF and EMO-DD prefers others. The other three multimodal multiobjective optimization algorithms perform similarly, though TriMOEA-TA&R gets the best IGDX values on two problems.

Table 4 shows that more than half of the IGDM results (11/15) obtained by EMO-DD are highlighted in bold, and the others are in italic (except on MMF3), which means that the solution sets searched by EMO-DD can have good performance in both the decision and objective spaces. The second best is TriMOEA-TA&R, on which most of the IGDM results are highlighted except IDMP. The third one, MO_Ring_PSO_SCD, also gets good IGDM values on several problems, especially on IDMP.

To further analyze the results mentioned above, the rank frequency distribution [64] is used for reorganizing the results of the three performance metrics. Tables 5, 6 and 7 are the rank frequencies for the IGD, IGDX and IGDM cases, respectively. The number in Rank i (i = 1,2, . . . ,m) under an algorithm indicates the times of the algorithm's performance ranked as i, where Rank 1 represents the best one, Rank 2 the second and so on. The rank score shown on the last row is calculated by formula $\sum_{i=1}^{5} (i \cdot (Rank\ i))$, which is an overall evaluation of the algorithms' performance. In general, a lower rank score represents a better performance the algorithm has.

We synthesize the rank information of the three tables. In Table 5, MO_Ring_PSO_SCD gets a much larger IGD rank score than others. Then in Table 6, the lowest IGDX rank score is marked under MO_Ring_PSO_SCD, and the gap between others is very obvious except EMO-DD. These mark scores indicate that the solution set obtained by MO_Ring_PSO_SCD has a great performance in the decision space but not the objective space. By contrast, MOEA/D-TCH gets a very low IGD rank score but the highest IGDX rank score. This indicates that MOEA/D-TCH is good at finding solutions with good performance in the objective space while ignoring the decision space. The IGD and IGDX rank scores obtained by the remaining four algorithms are not polarized like MOEA/D-TCH or MO_Ring_PSO_SCD. Two of the most noteworthy are TriMOEA-TA&R and EMO-DD. The former gets not bad rank scores on both IGD and IGDX cases but is not much better when compared with the latter. From the rank frequency distribution analysis, we can see EMO-DD has the

Table 2 Comparison results on IGD metric (mean and standard deviation)

Test problem	MOEA/D-TCH	Omni-optimizer	DN-NSGA-II	MO_Ring_PSO_SCD	TriMOEA-TA&R	EMO-DD
MMF1	**1.43E−06 (5.87E−07)**	6.80E−04 (1.63E−05)	8.05E−04 (3.74E−05)	9.81E−04 (3.03E−05)	1.78E−05 (1.14E−05)	3.23E−06 (1.21E−06)
MMF2	1.07E−04 (2.57E−04)	6.64E−04 (1.14E−04)	7.83E−04 (1.38E−04)	5.07E−03 (5.40E−04)	1.14E−02 (5.01E−02)	**1.02E−04 (1.12E−04)**
MMF3	5.58E−04 (7.74E−04)	6.21E−04 (1.61E−05)	9.20E−04 (7.43E−04)	3.81E−03 (3.16E−04)	2.28E−04 (2.28E−04)	**7.34E−05 (4.44E−05)**
MMF4	**3.45E−07 (2.39E−08)**	6.84E−04 (1.37E−05)	7.87E−04 (1.62E−05)	8.82E−04 (2.75E−05)	2.49E−03 (1.11E−02)	9.75E−06 (2.52E−06)
MMF5	**1.43E−06 (4.62E−07)**	6.68E−04 (1.25E−05)	7.83E−04 (1.95E−05)	9.44E−04 (1.58E−05)	2.67E−05 (6.28E−05)	2.84E−06 (9.23E−07)
MMF6	**1.44E−06 (3.57E−07)**	6.63E−04 (1.25E−05)	7.83E−04 (2.59E−05)	9.03E−04 (3.08E−05)	1.16E−05 (2.56E−06)	1.96E−06 (3.79E−07)
MMF7	1.67E−02 (3.54E−07)	7.56E−04 (2.15E−05)	9.60E−04 (3.71E−05)	9.00E−04 (3.38E−05)	2.96E−05 (3.53E−05)	**1.11E−06 (3.45E−07)**
MMF8	8.77E−02 (2.77E−04)	7.14E−04 (1.76E−05)	8.81E−04 (3.19E−05)	1.29E−03 (3.68E−05)	2.85E−04 (1.23E−03)	**1.02E−06 (7.91E−08)**
Omni-test	2.23E−04 (1.43E−04)	2.50E−03 (4.86E−05)	2.93E−03 (9.21E−05)	1.86E−02 (1.18E−03)	7.58E−04 (8.37E−04)	**2.04E−04 (8.85E−05)**
SYM-PART-simple	4.42E−04 (3.49E−05)	3.02E−03 (5.84E−05)	3.26E−03 (7.26E−05)	9.23E−03 (4.73E−04)	5.89E−03 (1.50E−03)	**1.06E−05 (3.77E−07)**
SYM-PART-rorated	4.41E−04 (3.80E−05)	3.25E−03 (7.08E−05)	3.81E−03 (1.08E−04)	1.13E−02 (7.44E−04)	1.02E−03 (4.74E−04)	**1.41E−05 (9.87E−07)**
IDMP-T1	**1.25E−08 (1.90E−09)**	1.46E−04 (1.84E−05)	1.45E−04 (1.52E−05)	9.20E−04 (4.05E−05)	6.23E−07 (6.95E−08)	3.53E−08 (5.59E−09)
IDMP-T2	**1.10E−08 (2.79E−10)**	1.35E−04 (5.63E−06)	1.30E−04 (1.00E−05)	2.92E−04 (9.70E−06)	6.45E−07 (5.33E−08)	3.08E−08 (1.68E−08)
IDMP-T3	**2.06E−08 (1.80E−08)**	1.21E−04 (3.37E−06)	1.19E−04 (3.34E−06)	3.43E−04 (1.59E−05)	1.05E−03 (2.50E−03)	1.70E−07 (6.05E−08)
IDMP-T4	**1.18E−08 (8.62E−10)**	1.36E−04 (4.62E−06)	1.34E−04 (7.11E−06)	3.89E−04 (1.71E−05)	6.36E−07 (6.45E−08)	5.77E−08 (3.51E−08)

Table 3 Comparision results on IGDX metric (mean and standard deviation)

Test problem	MOEA/D-TCH	Omni-optimizer	DN-NSGA-II	MO_Ring_PSO_SCD	TriMOEA-TA&R	EMO-DD
MMF1	1.05E−01 (4.17E−02)	1.98E−02 (2.81E−03)	2.23E−02 (3.26E−03)	**1.49E−02 (3.80E−04)**	4.60E−02 (3.90E−03)	1.83E−02 (3.70E−03)
MMF2	2.47E−01 (1.18E−01)	2.01E−02 (1.41E−02)	1.60E−02 (7.78E−03)	**9.08E−03 (9.49E−04)**	4.21E−02 (4.93E−02)	1.08E−02 (5.27E−03)
MMF3	1.71E−01 (5.80E−02)	1.59E−02 (1.08E−02)	1.34E−02 (5.18E−03)	**7.32E−03 (1.43E−03)**	4.52E−02 (7.58E−03)	4.31E−02 (7.52E−03)
MMF4	3.34E−01 (1.23E−01)	2.46E−02 (4.77E−03)	2.53E−02 (4.57E−03)	**8.57E−03 (2.57E−04)**	3.36E−02 (9.06E−02)	2.10E−02 (3.34E−03)
MMF5	2.58E−01 (8.18E−02)	6.66E−02 (6.83E−03)	6.67E−02 (6.39E−03)	**2.97E−02 (9.10E−04)**	7.02E−02 (1.41E−02)	5.31E−02 (8.26E−03)
MMF6	2.58E−01 (8.18E−02)	5.35E−02 (3.92E−03)	5.86E−02 (6.25E−03)	2.71E−02 (9.60E−04)	5.62E−02 (4.23E−03)	4.42E−02 (7.44E−03)
MMF7	1.26E−01 (5.99E−02)	9.34E−03 (1.13E−03)	1.04E−02 (1.96E−03)	**8.96E−03 (1.96E−04)**	3.03E−02 (2.53E−02)	2.38E−02 (3.64E−03)
MMF8	3.46E+00 (3.38E−02)	7.50E−02 (4.83E−02)	6.75E−02 (2.84E−02)	**2.20E−02 (1.24E−03)**	1.62E−01 (7.95E−02)	2.47E−01 (1.05E−01)
Omni-test	3.45E+00 (2.92E−01)	1.05E+00 (9.08E−02)	8.96E−01 (1.40E−01)	1.40E−01 (3.18E−02)	**7.15E−02 (1.14E−02)**	1.37E−01 (2.96E−02)
SYM-PART-simple	1.09E+01 (2.69E+00)	2.80E+00 (1.04E+00)	2.83E+00 (9.75E−01)	4.54E−02 (2.98E−03)	**1.23E−02 (7.80E−04)**	4.16E−02 (2.64E−03)
SYM-PART-rorated	1.02E+01 (1.65E−03)	1.30E+00 (7.84E−01)	9.06E−01 (6.75E−01)	**5.40E−02 (3.61E−03)**	2.58E−01 (3.78E−01)	5.73E−02 (9.29E−03)
IDMP-T1	6.73E−01 (2.29E−05)	6.37E−01 (7.35E−02)	6.63E−01 (8.65E−02)	4.64E−01 (1.23E−02)	6.06E−01 (2.07E−01)	**8.49E−04 (4.76E−04)**
IDMP-T2	6.73E−01 (6.61E−05)	6.43E−01 (7.20E−02)	5.93E−01 (9.09E−02)	4.53E−01 (1.64E−03)	5.06E−01 (2.97E−01)	**8.36E−04 (4.26E−04)**
IDMP-T3	6.59E−01 (1.17E−01)	5.06E−01 (1.77E−02)	5.74E−01 (1.98E−01)	4.86E−01 (1.10E−03)	**1.23E−01 (2.38E−01)**	2.16E−03 (8.15E−04)
IDMP-T4	6.73E−01 (2.07E−05)	6.48E−01 (5.93E−02)	6.63E−01 (4.42E−02)	4.55E−01 (3.78E−03)	5.06E−01 (2.98E−01)	**7.52E−02 (2.24E−01)**

Table 4 Comparison results on IGDM metric (mean and standard deviation)

Test problem	MOEA/D-TCH	Omni-optimizer	DN-NSGA-II	MO_Ring_PSO_SCD	TriMOEA-TA&R	EMO-DD
MMF1	2.50E−02 (3.85E−02)	2.54E−03 (3.55E−04)	3.07E−03 (9.20E−04)	2.32E−03 (5.68E−05)	6.08E−04 (2.79E−05)	**6.00E−04 (5.58E−05)**
MMF2	2.00E−01 (1.64E−01)	2.02E−02 (1.41E−02)	2.26E−02 (1.39E−02)	*9.80E−03 (1.03E−03)*	1.44E−02 (5.09E−02)	**1.29E−03 (7.46E−04)**
MMF3	1.72E−01 (1.30E−01)	1.59E−02 (1.08E−02)	1.43E−02 (7.07E−03)	*7.81E−03 (1.40E−03)*	**7.65E−03 (2.28E−03)**	8.72E−03 (2.59E−03)
MMF4	3.33E−01 (1.34E−01)	6.40E−02 (5.66E−03)	6.73E−02 (6.74E−03)	5.40E−02 (1.75E−04)	5.29E−02 (8.48E−02)	**3.43E−02 (1.31E−03)**
MMF5	1.30E−01 (1.01E−01)	1.51E−02 (1.49E−02)	1.45E−02 (1.82E−03)	4.41E−03 (1.59E−04)	*9.21E−04 (2.70E−04)*	**8.97E−04 (2.06E−04)**
MMF6	1.36E−01 (7.59E−02)	1.32E−02 (1.56E−03)	1.23E−02 (1.28E−03)	4.25E−03 (1.79E−04)	**7.71E−04 (9.91E−05)**	*1.61E−03 (1.50E−03)*
MMF7	1.31E−01 (8.66E−02)	2.09E−03 (2.70E−04)	2.52E−03 (3.18E−04)	2.18E−03 (4.88E−05)	5.62E−04 (3.64E−05)	**3.96E−04 (1.46E−05)**
MMF8	7.74E−01 (1.60E−03)	5.25E−02 (2.28E−02)	5.09E−02 (2.42E−02)	**1.24E−02 (4.37E−04)**	5.04E−02 (4.46E−02)	4.26E−02 (5.24E−02)
Omni-test	9.58E−01 (1.08E−02)	6.25E−01 (3.71E−02)	5.22E−01 (6.19E−02)	1.94E−01 (2.34E−02)	**4.00E−02 (1.31E−02)**	*5.95E−02 (1.91E−02)*
SYM-PART-simple	8.08E−01 (5.74E−02)	3.60E−01 (6.90E−02)	3.25E−01 (8.30E−02)	2.40E−02 (1.78E−03)	4.38E−04 (6.05E−05)	**7.71E−05 (9.87E−07)**
SYM-PART-rorated	8.89E−01 (2.64E−07)	1.80E−01 (8.39E−02)	1.26E−01 (6.87E−02)	2.93E−02 (2.58E−03)	4.51E−02 (4.85E−02)	**6.28E−04 (2.69E−04)**
IDMP-T1	5.00E−01 (2.37E−08)	4.54E−01 (1.16E−01)	3.47E−01 (1.86E−01)	1.24E−01 (5.78E−02)	*4.50E−01 (1.54E−01)*	**2.46E−04 (3.62E−04)**
IDMP-T2	5.00E−01 (3.48E−09)	4.42E−01 (1.52E−01)	4.13E−01 (1.43E−01)	3.36E−02 (1.16E−02)	3.77E−01 (2.19E−01)	**2.26E−04 (2.58E−04)**
IDMP-T3	4.90E−01 (5.43E−02)	1.29E−01 (9.34E−02)	5.66E−02 (5.28E−02)	4.59E−03 (4.58E−04)	1.54E−01 (1.83E−01)	**8.91E−04 (6.86E−04)**
IDMP-T4	5.00E−01 (1.08E−08)	4.81E−01 (4.89E−02)	4.82E−01 (4.94E−02)	*4.23E−02 (1.91E−02)*	3.75E−01 (2.22E−01)	**5.56E−02 (1.67E−01)**

Table 5 Rank frequency distribution analysis on IGD case

Algorithm	MOEA/D-TCH	Omni-optimizer	DN-NSGA-II	MO_Ring_PSO_SCD	TriMOEA-TA&R	EMO-DD
Rank 1	8	0	0	0	0	7
Rank 2	4	0	0	0	3	8
Rank 3	1	5	1	0	8	0
Rank 4	0	8	7	0	0	0
Rank 5	0	2	7	5	1	0
Rank 6	2	0	0	10	3	0
Rank score	31	57	66	85	53	23

Table 6 Rank frequency distribution analysis on IGDX case

Algorithm	MOEA/D-TCH	Omni-optimizer	DN-NSGA-II	MO_Ring_PSO_SCD	TriMOEA-TA&R	EMO-DD
Rank 1	0	0	0	9	2	4
Rank 2	0	1	1	3	1	9
Rank 3	0	5	3	3	4	0
Rank 4	0	5	7	0	2	1
Rank 5	1	3	4	0	6	1
Rank 6	14	1	0	0	0	0
Rank score	89	58	59	24	54	31

Table 7 Rank frequency distribution analysis on IGDM case

Algorithm	MOEA/D-TCH	Omni-optimizer	DN-NSGA-II	MO_Ring_PSO_SCD	TriMOEA-TA&R	EMO-DD
Rank 1	0	0	0	1	3	11
Rank 2	0	0	0	6	6	3
Rank 3	0	1	1	7	5	1
Rank 4	0	5	9	1	0	0
Rank 5	0	9	5	0	1	0
Rank 6	15	0	0	0	0	0
Rank score	90	68	64	38	35	20

ability to find a solution set with good performance in both the decision and objective spaces.

In summary, from the above analysis, we can conclude that MO_Ring_PSO_S CD improves its diversity in the decision space at the cost of greatly sacrificing the convergence. Such a big sacrifice may not be desirable since ensuring the Pareto optimality of solutions is the primary aim of solving MMOPs. TriMOEA-TA&R has considered improving the performance in both the decision and the objective space. But when comparing it with EMO-DD, we can find that it does not enough. EMO-DD can get a solution set with better performance in the two spaces than TriMOEA-TA&R simultaneously. This gives us some enlightenment, that the performance improvement in the objective and decision spaces may not be completely conflicting when solving MOPs. Table 7 also verifies our thought by numerical analysis. MO_Ring_PSO_SCD and TriMOEA-TA&R do have

good IGDM rank scores, but they are not better than EMO-DD.

To demonstrate the performance of these algorithms in both the decision and objective spaces further, the obtained solution sets and the corresponding objective vector fronts onto MMF6 (Fig. 4) and Omni-test (Fig. 5) are taken as examples for analysis, where the blue asterisks are well-distributed reference points extracted from PSs (the upper layer) and PF (the lower layer), and the red circles are obtained solutions (the upper layer) and the corresponding objective vectors (the lower layer), respectively.

In Fig. 4, the solutions obtained by MO_Ring_PSO_SCD almost cover the whole PSs. The solution sets obtained by the other four multimodal multiobjective optimization algorithms also can distribute to all parts of the PSs, although they are not as diverse as MO_Ring_PSO_SCD's. But when turn to the objective space, the other four algorithms all
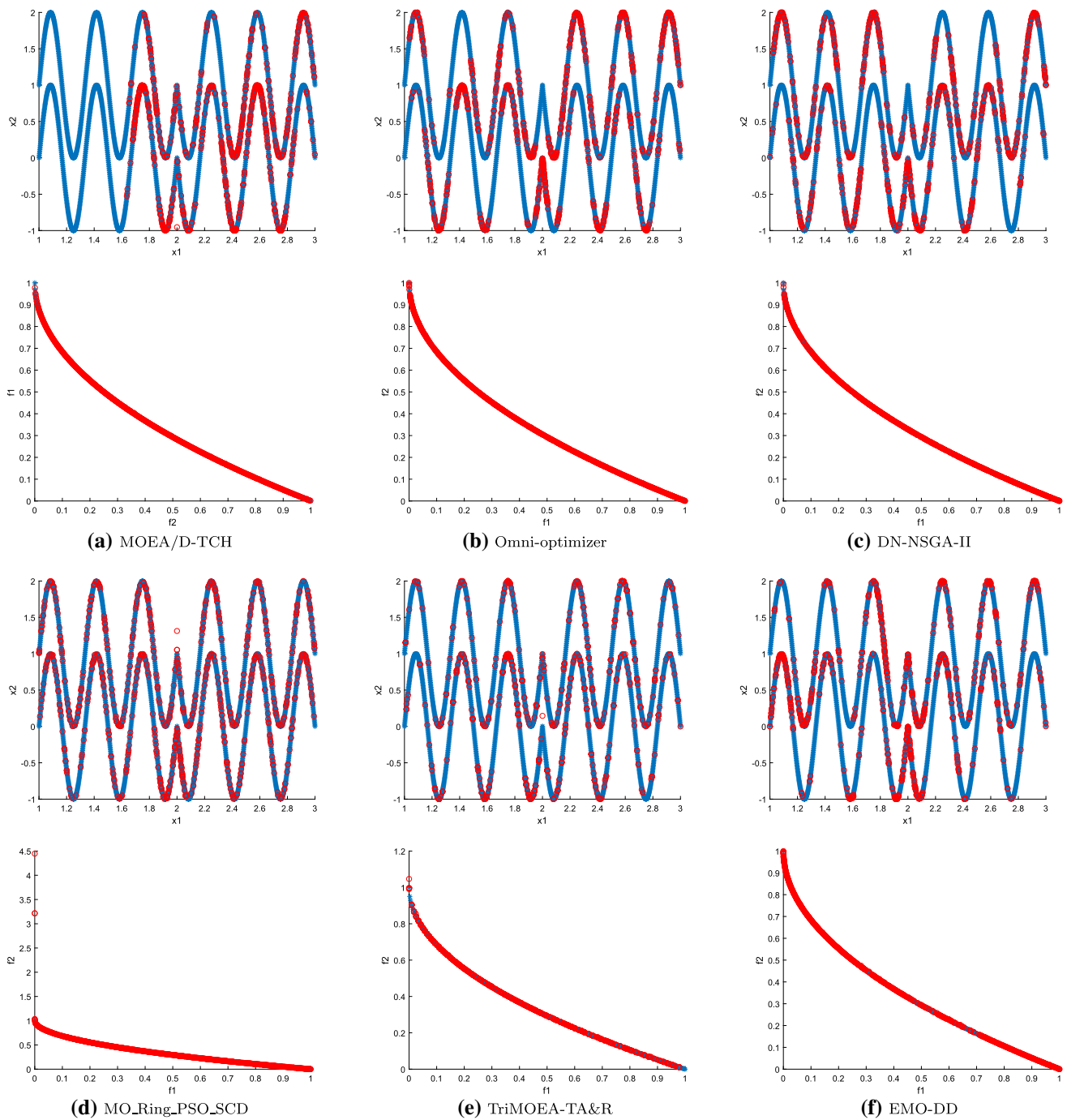
Fig. 4 Obtained solution set and corresponded objective vectors on MMF6

converge to the PF well, while some solutions searched by MO_Ring_PSO_SCD are even far away from it. It is worth noting that MOEA/D-TCH is even unable to find complete PS. That is the reason why it gets a very bad value on IGDX metric.

Omni-test is a tough problem that has 27 PSs. In Fig. 5, Omni-optimizer and DN-NSGA-II even cannot search out all the PSs. That means these two algorithms are not suitable

once the problem becomes complex. The solutions found by MOEA/D-TCH are not even on the PSs, which indicates all the solutions are not Pareto but local optimal solutions. However, the rest three algorithms can find the locations of all PSs, although the distribution of solutions on each PS part is different.

Figure 6 is presented for exploring the performance of the three algorithms, EMO-DD, MO_Ring_PSO_SCD
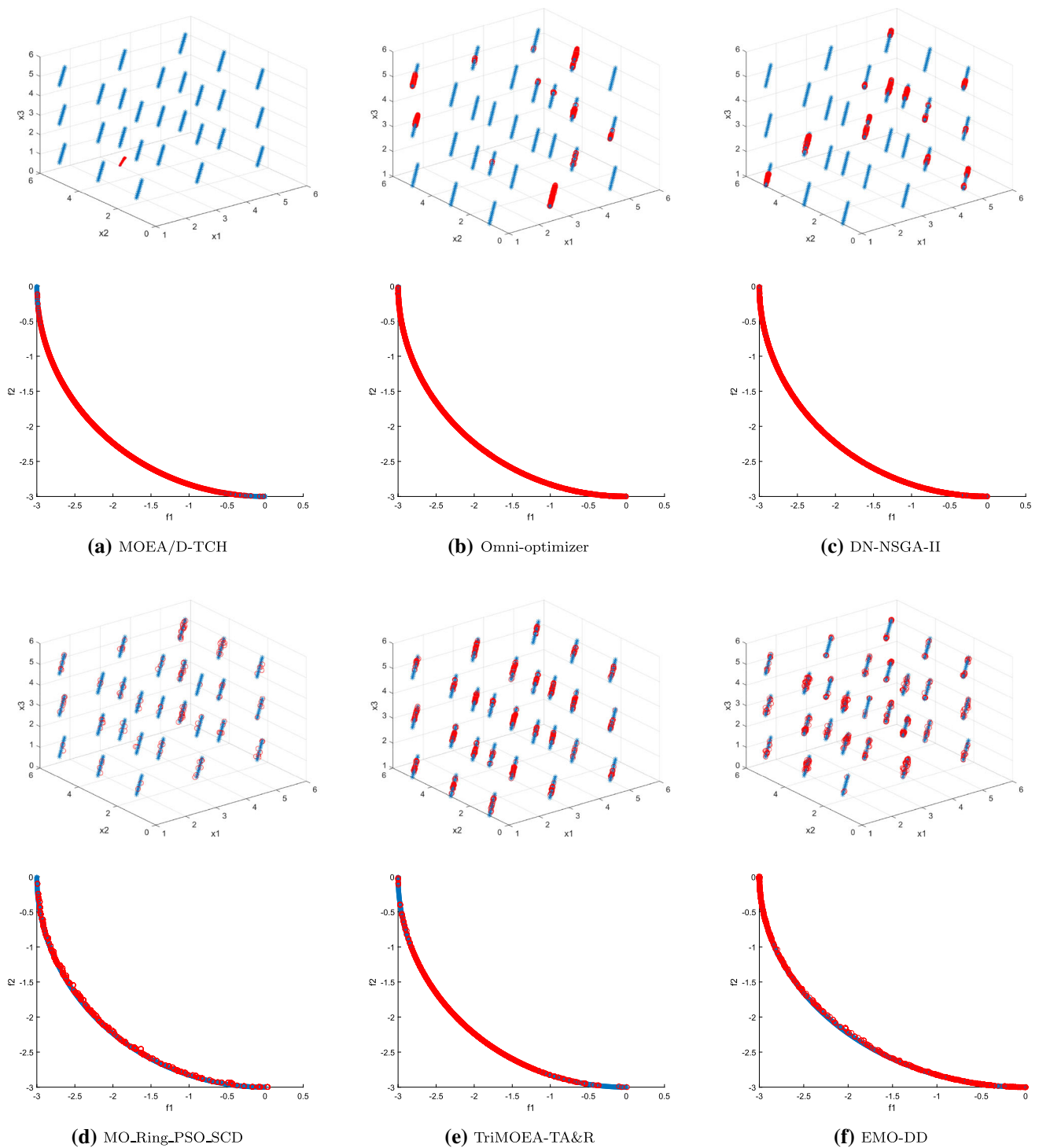
(a) MOEA/D-TCH   **(b)** Omni-optimizer   **(c)** DN-NSGA-II

(d) MO_Ring_PSO_SCD   **(e)** TriMOEA-TA&R   **(f)** EMO-DD

Fig. 5 Obtained solution set and corresponded objective vectors on two-objective Omni-test

and TriMOEA-TA&R, in dealing with multi-PS MMOPs. On the SYM-PART-rotated problem, Omni-optimizer and DN-NSGA-II cannot find the center part PS while the other three algorithms all fit the shape of the whole PSs. Not surprisingly, the diversity of solution set obtained by MO_Ring_PSO_SCD is the best. As for TriMOEA-TA&R, the proportion of solutions located on different PS part is various. Some PS parts are completely covered but some may only spot with several solutions. When turning to EMO-DD, it almost balances the coverage ratio of solutions on different PS part, although is not as good as MO_Ring_PSO_SCD.
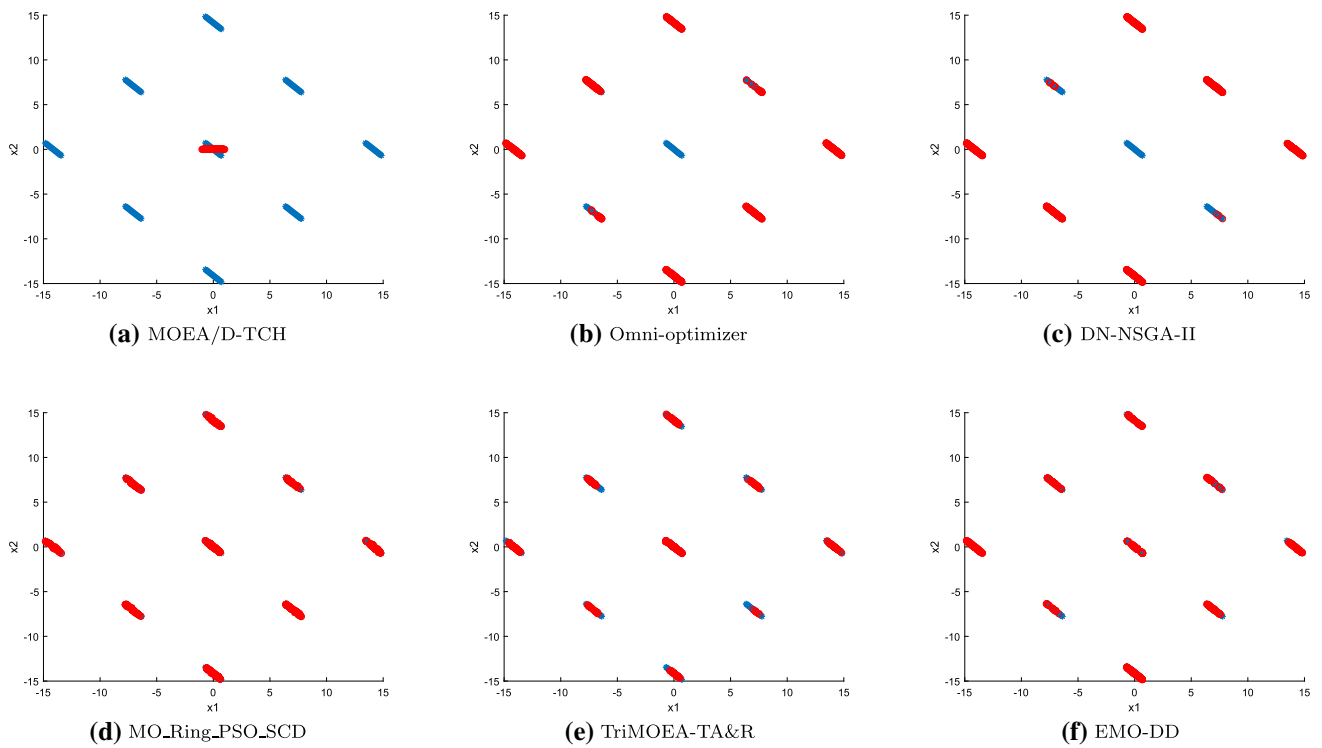
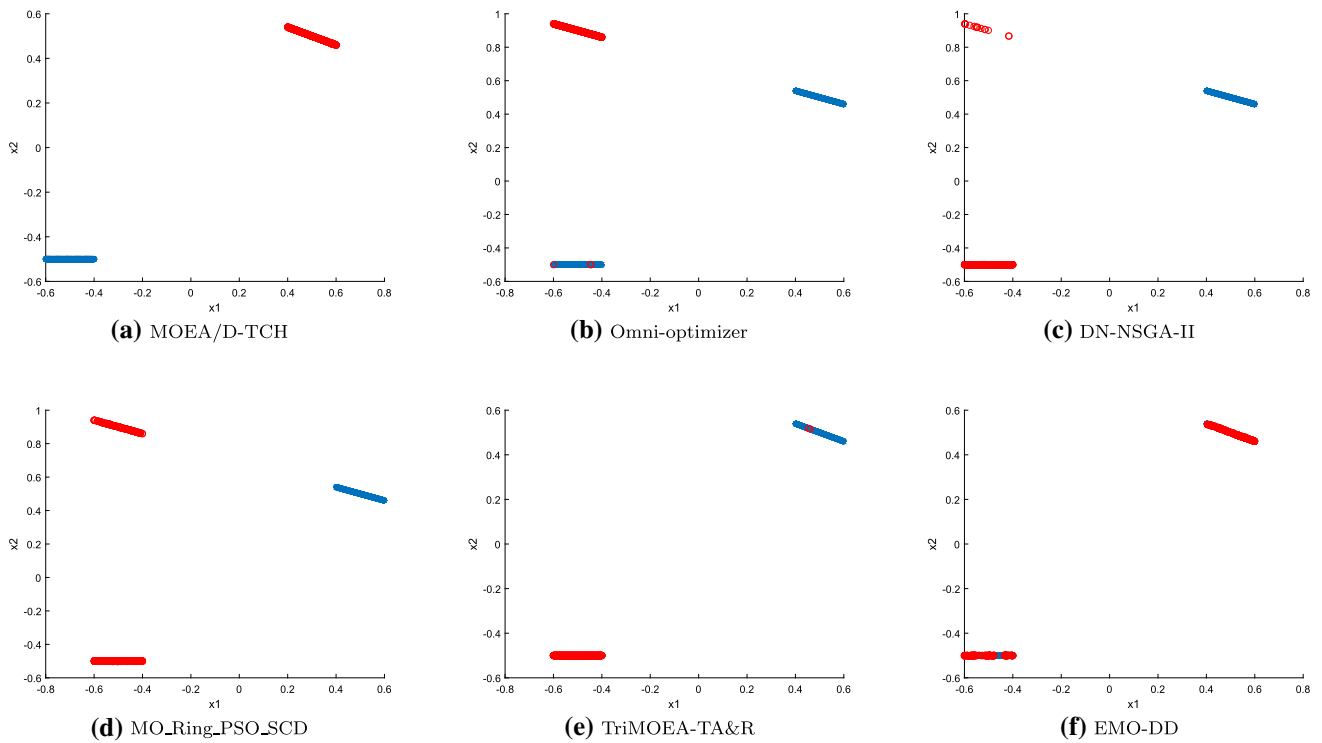**Fig. 6** Obtained solution set on SYM-PART-rotated

(a) MOEA/D-TCH (b) Omni-optimizer (c) DN-NSGA-II

(d) MO_Ring_PSO_SCD (e) TriMOEA-TA&R (f) EMO-DD

Fig. 7 Obtained solution set on IDMP-T3

(a) MOEA/D-TCH (b) Omni-optimizer (c) DN-NSGA-II

(d) MO_Ring_PSO_SCD (e) TriMOEA-TA&R (f) EMO-DD

IDMP represents a type of MMOPs with different search difficulty on each PS part. Figure 7 shows the solution set results on IDMP-T3, whose PSs consist of two parts and the upper right part (marked as EPS1) is more difficult to find than the bottom left one (marked as EPS2). It can be seen in Fig. 7 that all solutions found by MOEA/D-TCH are on EPS1. This is because MOEA/D-TCH blindly emphasizes the performance in the objective space, and discards all solutions on EPS2 after searched for EPS1. On the contrary, Omni-optimizer, DN-NSGA-II and MO_Ring_PSO_SCD are cheated by EPS1, though they can find EPS2. TriMOEA-TA&R can find the location of EPS1, but diversity is not satisfactory. Only EMO-DD finds both EPS1 and EPS2, and give them equal distribution. This shows that EMO-DD is better than others to jump local optimum and deal with this kind of deceptive problem.

5 Conclusion

In this paper, we have proposed an algorithm (EMO-DD) by using the decomposition method in the decision space to solve MMOPs. The whole population is decomposed and resided into several decision subregions and optimized collaboratively. In each subregion, solutions with good convergence and diversity in the decision space are screened out by optimizing a bi-objective problem. The two objectives of the problem reflect the convergence degree and distribution density of solutions, respectively. To ensure the well-performed solutions are preserved and inherited, a diversity archive and modified mating selection are proposed. The performance of EMO-DD has been analyzed and investigated through a series of comparative experiments with five state-of-the-art algorithms on fifteen problems. The experimental results show that EMO-DD has the ability to find diverse solutions with great convergence.

References

1. Deb K, Thiele L, Laumanns M, Zitzler E (2005) Scalable test problems for evolutionary multiobjective optimization. In: Evolutionary multiobjective optimization. Springer, London, pp 105–145
2. Mao-Guo G, Li-Cheng J, Dong-Dong Y, Wen-Ping M (2009) Evolutionary multi-objective optimization algorithms
3. Wang Z, Ong Y-S, Sun J, Gupta A, Zhang Q (2018) A generator for multiobjective test problems with difficult-to-approximate pareto front boundaries. IEEE Trans Evolut Comput 23(4):556–571
4. Wang Z, Ong Y-S, Ishibuchi H (2018) On scalable multiobjective test problems with hardly dominated boundaries. IEEE Trans Evolut Comput 23(2):217–231
5. Da B, Ong Y-S, Feng L, Qin AK, Gupta A, Zhu Z, Ting C-K, Tang K, Yao X (2007) Evolutionary multitasking for single-objective continuous optimization: Benchmark problems, performance metric, and baseline results. arXiv preprint arXiv:1706.03470

6. Yue C, Qu B, Yu K, Liang J, Li X (2019) A novel scalable test problem suite for multimodal multiobjective optimization. Swarm Evolut Comput 48:62–71
7. Li X, Epitropakis MG, Deb K, Engelbrecht A (2016) Seeking multiple solutions: an updated survey on niching methods and their applications. IEEE Trans Evolut Comput 21(4):518–538
8. Ishibuchi H, Peng Y, Shang K (2019) A scalable multimodal multiobjective test problem. In: International conference on evolutionary computation. IEEE, pp 310–317
9. Kudo F, Yoshikawa T, Furuhashi T (2011) A study on analysis of design variables in pareto solutions for conceptual design optimization problem of hybrid rocket engine. In: International conference on evolutionary computation. IEEE, pp 2558–2562
10. Hiroyasu T, Nakayama S, Miki M (2005) Comparison study of SPEA2+, SPEA2, and NSGA-II in diesel engine emissions and fuel economy problem. In: International conference on evolutionary computation, vol 1. IEEE, pp 236–242
11. Sebag M, Tarrisson N, Teytaud O, Lefevre J, Baillet S (2005) A multi-objective multi-modal optimization approach for mining stable spatio-temporal patterns. In: IJCAI, pp 859–864
12. Togelius J, Preuss M, Yannakakis GN (2010) Towards multiobjective procedural map generation. In: International workshop on procedural content generation in games, pp 1–8
13. Preuss M, Kausch C, Bouvy C, Henrich F (2010) Decision space diversity can be essential for solving multiobjective real-world problems. In: Multiple criteria decision making for sustainable energy and transportation systems. Springer, pp 367–377
14. Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evolut Comput 6(2):182–197
15. Li H, Wang L, Hei X, Li W, Jiang Q (2018) A decomposition-based chemical reaction optimization for multi-objective vehicle routing problem for simultaneous delivery and pickup with time windows. Memet Comput 10(1):103–120
16. Hammami M, Bechikh S, Hung C-C, Said LB (2019) A multi-objective hybrid filter-wrapper evolutionary approach for feature selection. Memet Comput 11(2):193–208
17. Zou J, Yang Q, Yang S, Zheng J (2020) Ra-dominance: a new dominance relationship for preference-based evolutionary multi-objective optimization. Appl Soft Comput 106192
18. Bui LT, Liu J, Bender A, Barlow M, Wesolkowski S, Abbass HA (2011) DMEA: a direction-based multiobjective evolutionary algorithm. Memet Comput 3(4):271–285
19. Wang S, Liu J, Jin Y (2020) Robust structural balance in signed networks using a multiobjective evolutionary algorithm. IEEE Comput Intell Mag 15(2):24–35
20. Preuss M, Naujoks B, Rudolph G (2006) Pareto set and EMOA behavior for simple multimodal multiobjective functions. In: International conference on parallel problem solving from nature. Springer, pp 513–522
21. Deb K, Tiwari S (2005) Omni-optimizer: a procedure for single and multi-objective optimization. In: International conference on evolutionary multi-criterion optimization. Springer, pp 47–61
22. Liang J, Yue C, Qu B-Y (2016) Multimodal multi-objective optimization: a preliminary study. In: International conference on evolutionary computation, pp 2454–2461
23. Tanabe R, Ishibuchi H (2019) A review of evolutionary multimodal multiobjective optimization. IEEE Trans Evolut Comput 24(1):193–200
24. Yue C, Qu B, Liang J (2017) A multiobjective particle swarm optimizer using ring topology for solving multimodal multiobjective problems. IEEE Trans Evolut Comput 22(5):805–817
25. Liang JJ, Qu B-Y, Mao X, Niu B, Wang D (2014) Differential evolution based on fitness Euclidean-distance ratio for multimodal optimization. Neurocomputing 137:252–260

26. Qu B-Y, Suganthan PN, Liang J-J (2012) Differential evolution with neighborhood mutation for multimodal optimization. IEEE Trans Evolut Comput 16(5):601–614

27. Liang JJ, Ma ST, Qu B-Y, Niu B (2012) Strategy adaptative memetic crowding differential evolution for multimodal optimization. In: International conference on evolutionary computation. IEEE, pp 1–7

28. Hao L, Gong M, Sun Y, Pan J (2006) Niching clonal selection algorithm for multimodal function optimization. In: International conference on natural computation. Springer, pp 820–827

29. Goldberg DE, Richardson J et al (1987) Genetic algorithms with sharing for multimodal function optimization. In: Genetic algorithms and their applications: proceedings of the 2nd international conference on genetic algorithms. Lawrence Erlbaum, Hillsdale, pp 41–49

30. Qing L, Gang W, Zaiyue Y, Qiuping W (2008) Crowding clustering genetic algorithm for multimodal function optimization. Appl Soft Comput 8(1):88–95

31. Lin Q, Lin W, Zhu Z, Gong M, Li J, Coello CAC (2020) Multimodal multi-objective evolutionary optimization with dual clustering in decision and objective spaces. IEEE Trans Evolut Comput

32. Liu Y, Yen GG, Gong D (2018) A multimodal multiobjective evolutionary algorithm using two-archive and recombination strategies. IEEE Trans Evolut Comput 23(4):660–674

33. Xia H, Zhuang J, Yu D (2013) Combining crowding estimation in objective and decision space with multiple selection and search strategies for multi-objective evolutionary optimization. IEEE Trans Cybern 44(3):378–393

34. Liu Y, Ishibuchi H, Nojima Y, Masuyama N, Shang K (2018) A double-niched evolutionary algorithm and its behavior on polygon-based problems. In: International conference on parallel problem solving from nature. Springer, pp 262–273

35. Wang Z-J, Zhan Z-H, Lin Y, Yu W-J, Wang H, Kwong S, Zhang J (2019) Automatic niching differential evolution with contour prediction approach for multimodal optimization problems. IEEE Trans Evolut Comput 24(1):114–128

36. Kim M, Hiroyasu T, Miki M, Watanabe S (2004) SPEA2+: improving the performance of the strength pareto evolutionary algorithm 2. In: International conference on parallel problem solving from nature. Springer, pp 742–751

37. Liang J, Xu W, Yue C, Yu K, Song H, Crisalle OD, Qu B (2019) Multimodal multiobjective optimization with differential evolution. Swarm and Evolut Comput 44:1028–1059

38. Tanabe R, Ishibuchi H (2018) A decomposition-based evolutionary algorithm for multi-modal multi-objective optimization. In: International conference on parallel problem solving from nature. Springer, pp 249–261

39. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: International conference on neural networks, vol 4. IEEE, pp 1942–1948

40. Nguyen HB, Xue B, Andreae P (2018) Pso with surrogate models for feature selection: static and dynamic clustering-based methods. Memet Comput 10(3):291–300

41. Reyes-Sierra M, Coello CC et al (2006) Multi-objective particle swarm optimizers: a survey of the state-of-the-art. Comput Intell Res 2(3):287–308

42. Kumar S, Kumar P, Sharma TK, Pant M (2013) Bi-level thresholding using PSO, artificial bee colony and MRLDE embedded with otsu method. Memet Comput 5(4):323–334

43. Qu B, Li C, Liang J, Yan L, Yu K, Zhu Y (2020) A self-organized speciation based multi-objective particle swarm optimizer for multimodal multi-objective problems. Appl Soft Comput 86:105886

44. Liang J, Guo Q, Yue C, Qu B, Yu K (2018) A self-organizing multi-objective particle swarm optimization algorithm for multimodal multi-objective problems. In: International conference on swarm intelligence. Springer, pp 550–560

45. Jin Y, Okabe T, Sendho B (2001) Adapting weighted aggregation for multiobjective evolution strategies. In: International conference on evolutionary multi-criterion optimization. Springer, pp 96–110

46. Jaszkiewicz A (2002) On the performance of multiple-objective genetic local search on the 0/1 knapsack problem-a comparative experiment. IEEE Trans Evolut Comput 6(4):402–412

47. Paquete L, Stützle T (2003) A two-phase local search for the biobjective traveling salesman problem. In: International conference on evolutionary multi-criterion optimization. Springer, pp 479–493

48. Zhang Q, Li H (2007) MOEA/D: a multiobjective evolutionary algorithm based on decomposition. IEEE Trans Evolut Comput 11(6):712–731

49. Deb K, Jain H (2013) An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: solving problems with box constraints. IEEE Trans Evolut Comput 18(4):577–601

50. Miettinen K (2012) Nonlinear multiobjective optimization, vol 12. Springer, Berlin

51. Li K, Deb K, Zhang Q, Kwong S (2014) An evolutionary many-objective optimization algorithm based on dominance and decomposition. IEEE Trans Evolut Comput 19(5):694–716

52. Zitzler E, Laumanns M, Thiele L (2001) SPEA2: improving the strength pareto evolutionary algorithm. TIK-report, vol 103

53. Yang S, Li M, Liu X, Zheng J (2013) A grid-based evolutionary algorithm for many-objective optimization. IEEE Trans Evolut Comput 17(5):721–736

54. Li K, Chen R, Fu G, Yao X (2018) Two-archive evolutionary algorithm for constrained multiobjective optimization. IEEE Trans Evolut Comput 23(2):303–315

55. Liu Y, Ishibuchi H, Yen GG, Nojima Y, Masuyama N (2019) Handling imbalance between convergence and diversity in the decision space in evolutionary multi-modal multi-objective optimization. IEEE Trans Evolut Comput

56. Rudolph G, Naujoks B, Preuss M (2007) Capabilities of EMOA to detect and preserve equivalent pareto subsets. In: International conference on evolutionary multi-criterion optimization. Springer, pp 36–50

57. Li M, Yang S, Liu X (2015) Pareto or non-pareto: bi-criterion evolution in multiobjective optimization. IEEE Trans Evolut Comput 20(5):645–665

58. Zitzler E, Thiele L, Laumanns M, Fonseca CM, Da Fonseca VG (2003) Performance assessment of multiobjective optimizers: an analysis and review. IEEE Trans Evolut Comput 7(2):117–132

59. Zhang W, Li G, Zhang W, Liang J, Yen GG (2019) A cluster based PSO with leader updating mechanism and ring-topology for multimodal multi-objective optimization. Swarm and Evolut Comput 50:100569

60. Zhou A, Zhang Q, Jin Y (2009) Approximating the set of pareto-optimal solutions in both the decision and objective spaces by an estimation of distribution algorithm. IEEE Trans Evolut Comput 13(5):1167–1189

61. Deb K, Agrawal RB et al (1995) Simulated binary crossover for continuous search space. Complex Syst 9(2):115–148

62. Deb K, Goyal M (1996) A combined genetic adaptive search (GeneAS) for engineering design. Comput Sci Inf 26:30–45

63. Tian Y, Cheng R, Zhang X, Jin Y (2017) PlatEMO: a MATLAB platform for evolutionary multi-objective optimization. IEEE Comput Intell Mag 12(4):73–87

64. Wimmer G, Šidlík P, Altmann G (1999) A new model of rank-frequency distribution. J Quant Linguist 6(2):188–193