



Multimodal multi-objective optimization via determinantal point process-assisted evolutionary algorithm

Xinyu Cheng¹ · Wenyin Gong¹ · Fei Ming¹ · Xiaofang Zhu²

Received: 8 December 2022 / Accepted: 16 October 2023 / Published online: 8 November 2023
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2023

Abstract

Multimodal multi-objective optimization problems (MMOPs) are widely present in real life. Due to the need of balancing between convergence and diversity in multi-objective optimization, as well as the need of balancing between diversities in objective and decision spaces, exploring the Pareto optimal front and Pareto optimal solution set becomes rather difficult in solving MMOPs. Recently, some multimodal multi-objective optimization algorithms have emerged. However, most of them are convergence-first, which may result in poor diversity of the solution set in decision space. To remedy this defect, in this paper, a determinantal point process (DPP)-assisted evolutionary algorithm is proposed to effectively solve MMOPs. In the proposed method, i) the DPPs are used to select subsets to consider convergence and diversity in both objective and decision spaces; ii) a kernel matrix is designed to retain solutions with poor convergence but good diversity in the decision space to explore the equivalent Pareto optimal solution sets; and iii) we propose a framework that combines the population and archive to better solve MMOPs. The results show that the proposed algorithm achieves the best performance in 18 of the 28 benchmark problems compared to six state-of-the-art algorithms.

Keywords Multimodal multi-objective optimization · Evolutionary algorithm · Determinantal point processes · Subset selection · Decision space diversity.

1 Introduction

Multi-objective optimization problems (MOPs) exist widely in real life, which aim to optimize multiple objective functions simultaneously. Without loss of generality, we give the following formula to describe:

$$\begin{aligned} \text{Minimize} \quad & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})) \\ \text{subject to} \quad & \mathbf{x} = (x_1, x_2, \dots, x_n) \in \Omega \end{aligned} \quad (1)$$

where $\Omega \subseteq \mathbb{R}^n$ denotes the search space; m is the number of objective functions; $\mathbf{x} = (x_1, \dots, x_n)$ is a n -dimensional decision vector; and n is the number of decision variables. Due to the conflict between multiple objectives, there is no single solution that can make multiple objectives reach the optimal simultaneously [1]. Therefore, we need to find a set of Pareto optimal solutions (PS), which map to the objective space and form the Pareto optimal front (PF).

There are multiple Pareto optimal solution sets (PSs) corresponding to the same point on the PF, that is, the multimodal multi-objective optimization problems (MMOPs). Such practical problems abound, for example: flow-shop scheduling [2], path-planning [3], mining stable spatiotemporal patterns [4], multi-objective knapsack optimization [5], map-based [6] and the credit card fraud detection [7]. But since multiple solutions in the decision space correspond to a point on PF, some of which are inaccessible, we need to find as many equivalent solutions as possible. In other words, because some Pareto

✉ Wenyin Gong
wygong@cug.edu.cn

Xinyu Cheng
chengxinyu@cug.edu.cn

Fei Ming
feiming@cug.edu.cn

Xiaofang Zhu
304336480@qq.com

¹ School of Computer Science, China University of Geosciences, Wuhan 430074, Hubei, China

² School of Computer Science and Artificial Intelligence, Wuhan University of Technology, Wuhan 430071, Hubei, China

solution sets are not feasible in practice, it is necessary to provide alternative solutions for decision makers [8].

In the past two decades, a large number of multi-objective optimization evolutionary algorithms (MOEAs) have been proposed to solve MOPs with great success [9]. These MOEAs can be broadly divided into three categories: (1) Pareto dominance-based algorithms, *e.g.*, non-dominated sorting genetic algorithm II (NSGA-II) [10], strength Pareto evolutionary algorithm 2 (SPEA2) [11]; (2) indicator-based algorithms, *e.g.*, ϵ -dominance-based MOEA (GDE-MOEA) [12], Two_Arch2 [13]; and (3) decomposition-based algorithms, *e.g.*, MOEA based on decomposition (MOEA/D) [14], and reference vector guided evolutionary algorithm (RVEA) [15]. However, since they only consider the convergence and diversity of the objective space, their ability to solve MMOPs is limited.

In recent years, many multimodal multi-objective optimization evolutionary algorithms (MMOEAs) have been proposed, such as decision space-based niching NSGA-II (DNNSGAI) [16], MO_Ring_SCD_PSO [3], MMOEADC [17], DN-MMOES [18], MMEAWI [19] and HREA [20]. They are based on existing MOEAs, to a certain extent, with the addition of techniques to account for the diversity of the decision space [21], but most of them are convergence-first. That is, when selecting the solutions of next-generation population, the convergence of objective space is the first priority, and then, the diversity in decision space is considered. As a result, non-dominated solutions are preferred, so that these algorithms converge quickly to easily detectable PSs. In addition, most solutions are selected one by one. In this way, those solutions that are less convergent in the decision space, but more diverse, will be deleted. Therefore, it is difficult to fully explore the decision space and many PSs cannot be found. Under the same parameter settings to that of Sect. 4.1.3, some representative results are presented, which prove the deficiency and limitation of convergence-first methods mentioned above. We test two convergence-first MMOEAs, MO_Ring_SCD_PSO on MMF5 and MMMOP2, DNNSGAI on MMF6 and MMMOP1. Figure 1 shows the results. The different colored lines in Fig. 1a represent that there are 6 equivalent PSs of the same PF on MMMOP1. Similarly, there are 4 equivalent PSs of the same PF on MMF5 and MMF6 in Fig. 1b and d. In Fig. 1c, MMMOP2 has 5 equivalent PSs. It clearly shows that MO_Ring_SCD_PSO could not find all PSs, and the obtained solutions by DNNSGAI had poor diversity in the decision space.

To solve the above problems, this study puts forward an MMOEA based on determinantal point processes (DPPs), termed MMOEADPPs. The algorithm uses DPPs to select

the subset of the population, which aims to better balance convergence and diversity in both objective and decision spaces. Besides, we introduce an archive to maintain non-dominated solutions so as to better converge to the PF of the objective space and also improve the diversity of the decision space to search for more PSs. Experimental results show that our algorithm outperforms other six state-of-the-art algorithms on 28 benchmark problems. The main contributions of this work can be summarized as follows:

- *Introduce the DPPs:* We introduce the DPPs into MMOEA. Compared with the existing MMOEAs to select next generation one by one, it selects a subset at one time according to the probability model. Since the probability model comprehensively considers convergence and diversity in both spaces, based on it, the subset selected by DPPs not only considers convergence, but also maximizes the diversity of the two spaces. As a result, the current solutions with poor convergence but good diversity are preserved, which will facilitate the exploration of sparse regions of the decision space.
- *A kernel matrix:* A kernel matrix is designed to represent the convergence, diversity of the objective space and diversity of the decision space. For this reason, the trade-off solution subset that takes all three into account will be chosen through DPPs. The convergence of the objective space is represented based on the non-dominated fronts, and the Euclidean distance between individuals is used to measure the diversity of the objective space and decision space.
- *A framework of the population and archive:* For the purpose of better solving MMOPs, we propose a framework which integrates the population and archive. In the selection of solutions in the population, convergence and diversity in both objective and decision spaces are considered comprehensively by adopting DPPs. Accordingly, the algorithm can avoid falling into the local PS and better explore equivalent PSs in the decision space, while the archive can retain solution set that can represent PF and multiple PSs in non-dominated solutions. As a consequence, the convergence and diversity performance of the algorithm is guaranteed.

The remainder of this article is organized as follows. Section 2 briefly introduces the DPPs and the related work of MMOEAs. Then, the proposed algorithm MMOEADPPs is elaborated in Sect. 3. In Sect. 4, the results of the experiments and the corresponding analyses are presented. Finally, the conclusions and future research directions are given in Sect. 5.

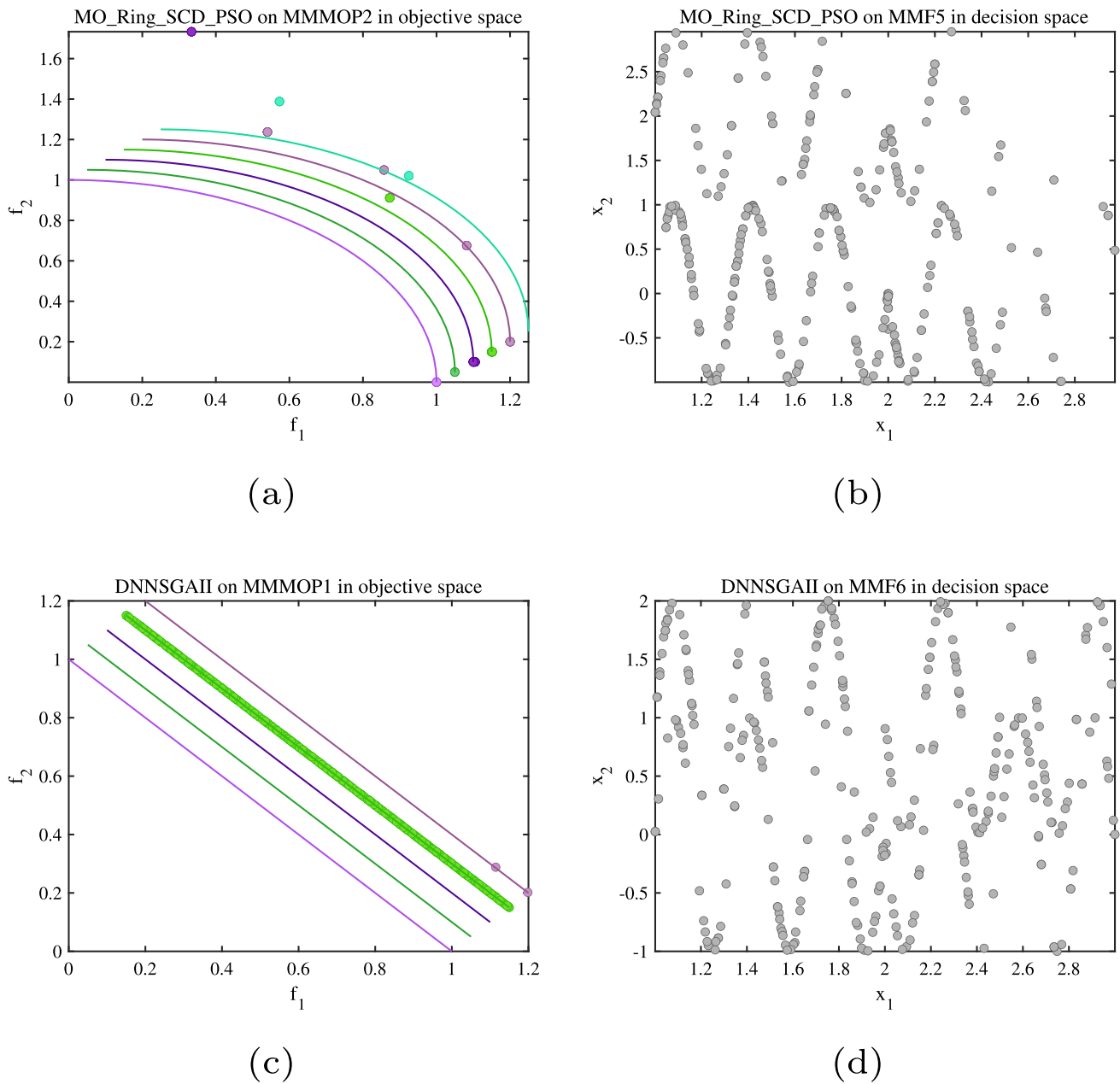


Fig. 1 Representative experimental results of convergence-first MOEAs. **a** MO_Ring_SCD_PSO on MMMOP2 in objective space. **b** MO_Ring_SCD_PSO on MMF5 in decision space. **c** DNNSGAI on MMMOP1 in objective space. **d** DNNSGAI on MMF6 in decision space. None of these algorithms could fully explore PF and PSs. Furthermore, there were large sparse areas in the decision space

choosing two similar terms is (inversely) correlated. In other words, the more similar items are, the less likely they are to be sampled simultaneously. DPPs apply to many practical problems, such as the web search problems [22], product recommendation [23] and graph sampling [24]. The point process measures the probability of selecting an instance Y of set \mathcal{Y} , which of the finite discrete points, *i.e.*, $\mathcal{Y} = \{1, \dots, N\}$, can be defined as follows:

2 Related work and motivations

2.1 DPPs

In machine learning (ML), the goal of a subset selection problem is to select a more diverse subset of items from a ground set. The balance of the consideration of diversity in ML problems often through the use of DPPs to keep, where DPP gives diversity on subsets such that the probability of

$$P_{\mathbf{L}}(Y = \mathcal{Y}) = \frac{\det(\mathbf{L}_Y)}{\det(\mathbf{L} + \mathbf{I})}, \quad (2)$$

where \mathbf{L} is an $N \times N$ semi-definite kernel matrix; it is equivalent to a table describing the relationship between any two individuals in population \mathcal{P} . The diversity and convergence relationship between any two individuals in the population can be queried by this relation table. \mathbf{I} is an identity matrix with size of N . $\det()$ is the determinant. When we set the subset size to k , we can get a variation of DPP, *i.e.*, kDPP. k represents the size of the subset that needs to be selected. Different values are taken in different test problems according to the population size N designed in Table 3. In other words, in our proposed algorithm, $k = N$. The formula is described as follows:

$$P_{\mathbf{L}}^k(Y = \mathcal{Y}) = \frac{\det(\mathbf{L}_Y)}{\sum_{\mathcal{Y}'=k} \det(\mathbf{L}_{\mathcal{Y}'})}, \quad (3)$$

However, its time complexity is as high as $O(N^k)$, so that it is very difficult to solve. In order to solve this problem, in [22], Kulesza and Taskar came up with a method: With the help of matrix eigenvalue decomposition, the above operations can be divided into two steps. First, the kernel matrix $\mathbf{L} = \sum_{n=1}^N \lambda_n \mathbf{v}_n \mathbf{v}_n^T$ is decomposed into the set of eigenvalues λ and the set of eigenvectors \mathcal{V} . Then, subset \mathcal{J} is extracted from \mathcal{V} according to the following formula, and its size is k ; then, the probability of this subset being selected is:

$$P_r(\mathcal{J} = \mathcal{J}) = \frac{\prod_{n \in \mathcal{J}} \lambda_n}{\sum_{\mathcal{J}=k} \prod_{m \in \mathcal{J}} \lambda_m}. \quad (4)$$

Thus, k individuals with large differences are selected as reference frames from the complete set \mathcal{P} . Then, based on the eigenvectors set \mathcal{J} , the offspring individual i is extracted with the following formula as probability, to evaluate the quality of each individual, until population size reach requirements.

$$P_r(i) = \frac{1}{|\mathcal{V}|} \sum_{\mathbf{v} \in \mathcal{V}} (\mathbf{v}_n^T \mathbf{e}_i)^2, \quad (5)$$

where only the i th element of \mathbf{e}_i is 1, and the rest is 0, *i.e.*, $\mathbf{e}_i = (0, \dots, 1, \dots, 0)$.

2.2 Existing MMOEAs

2.2.1 Convergence-first MMOEAs

Most algorithms pay more attention to convergence, but not enough diversity in the decision space. Peng *et al.* [8] proposed a diversity-enhanced subset selection framework to solve MMOPs. It first looks for uniformly distributed objective vectors in the objective space; then, the

corresponding equivalent solutions of these objective vectors are found in the decision space. Finally, the remaining solutions are selected from the already found equivalent solutions. However, it removes the dominated solutions before initially selecting the uniformly distributed objective vector. TriMOEATAR [25] suggested to use double archives and recombination strategies to select the next generation of the population, but the convergence archive focuses more on non-dominated solutions. Li *et al.* [26] also proposed a method using two-archive. The fitness scheme adopted to update the population is designed by selecting from non-dominated solutions. MO_Ring_PSO_SCD [3] used an index-based ring-topology particle swarm optimizer (PSO) and its special crowding distance are calculated according to non-dominated ranking. Tanabe and Ishibuchi [27] suggested a framework using decomposition-based evolutionary algorithms based on assignment, deletion and addition operations. The same subproblem may be allocated more than one individual, after that the one with the best convergence will be selected. CEA-LES [28] devised a layer-to-layer strategy to better locate equivalent PSs. However, clearing-based niching mechanism is applied after non-dominated ranking. In other words, it prefers the non-dominated solution. Li *et al.* [29] proposed a self-organizing quantum-inspired particle swarm-based algorithm. The algorithm maintains a non-dominated solution set by a maximum–minimum distance among solutions. MMOEADC [17] devised an MMOEA through dual clustering in decision and objective spaces. Firstly, the solutions in the decision space are divided into clusters to select individuals as population. Then it clusters the population in the objective space. After that, individuals in the most crowded clusters with the worst diversity are removed. However, local PSs give priority to the selection from the non-dominated solutions in the local cluster. Fan and Yan [30] suggested a method to embed zoning search in MMOEAs to improve their performance. The entire decision space is decomposed into multiple subspaces. In each subspace, an MMOEA will be selected to explore the PSs and PF. As a result, the chosen algorithm affects its performance. HREA [20] devised a hierarchy ranking approach to choose individuals desired by decision makers. However, the local convergence quality evaluation strategy is also based on non-dominated solutions. Li *et al.* [31] embedded grid search technique into the algorithm. Besides, an environment selection strategy is designed to remove inferior convergence individuals to better approach the non-dominated solutions.

2.2.2 Convergence-balanced MMOEAs

In the past two years, more and more studies have focused on how to balance convergence and diversity in both

objective and decision spaces. DN-MMOES [18] presented a two-stage double niched method. The niching strategy is first carried out in the decision space. Moreover, density operator is designed for the diversity of decision space. In addition, the mating pool selection strategy is not adopted. Therefore, it is possible to avoid selecting high-convergence individuals to produce offspring. MMEAWI [19] proposed a method for evaluating potential convergence quality that prioritizes diversity and through this weight indicator to guide the population evolution. CPDEA [21] suggested a convergence-penalized density mechanism to choose the next generation. In CPDEA, the selection criterion is based on the transformed distances which measured by the local convergence quality.

2.2.3 Motivation

The traditional selection operator lacks the evaluation of the overall quality of the solution set, so it is easy to lead to the abandonment of many solutions which have poor convergence but are in favor of exploring the decision space and objective space [32]. In addition, most of the MMOEAs proposed in recent years are convergence-first, which makes the diversity of the solution set not good enough, as shown in Fig. 1.

In MaOEADPPs [33], a new selection strategy in the multi-objective field is proposed. According to a ML technology, *i.e.*, DPPs, it is considered to select the overall solution set at one time, the algorithm works well in dealing with many-objective optimization problems. In DPPs, the probability that two elements are extracted as part of a subset is related not only to the probability that a single element is extracted, but also to the correlation of the two elements. That is, the higher probability a single element is selected, and at the same time, the lower the similarity between the elements, the more likely this set is to be selected. DPPs can better recall a more diverse subset. Based on this, we can get a more evenly distributed solution set. Inspired by it, we try to extend this idea to MMOEA. By designing the kernel matrix and using it to represent the convergence and diversity of the objective space and decision space, the three can be comprehensively considered and balanced according to the subset selected by DPPs. Thereby avoiding the above-mentioned defects of the traditional selection operator, and does not give too high priority to convergence like many MMOEAs. Furthermore, since different PSs of MMOPs are difficult to be fully detected, it is necessary to retain solutions that are not so convergent, but evenly distributed around equivalent PSs.

In view of the above-mentioned considerations, in Sect. 3, we will describe the proposed algorithm in detail,

including the design of the kernel matrix, the framework of population and archive and population update.

3 Proposed algorithm

3.1 The kernel matrix

A kernel matrix is designed to represent the convergence and diversity of the objective space and decision space, so that the solution subset selected by the DPPs can better balance the above properties. Moreover, it is mainly used to assist the selection of the population.

Given a population \mathcal{P} , \mathbf{x} is an individual in it, and n is the number of individuals. In [34], Liu *et al.* proposed that whether normalization leads to differences in algorithm performance. Therefore, the population should be normalized before computing the kernel matrix. Consequently, we use the following normalization method to normalize the individuals in both objective and decision spaces. Its formula is described as follows:

$$\bar{f}_i(\mathbf{x}) = \frac{f_i(\mathbf{x}) - z_i^*}{z_i^{nad} - z_i^*}, \quad (6)$$

where \mathbf{z}^* , \mathbf{z}^{nad} are the maximum and minimum values among these individuals, respectively. They are calculated by the following formulas:

$$z_i^* = \min(\{f_i(\mathbf{x}) | \mathbf{x} \in \mathcal{P}\}), \quad (7)$$

$$z_i^{nad} = \max(\{f_i(\mathbf{x}) | \mathbf{x} \in \mathcal{P}\}), \quad (8)$$

The kernel matrix \mathbf{L} is devised to assist the population in producing the next generation. For two individuals $\mathbf{x}, \mathbf{y} \in \mathcal{P}$, we employ the following formula to calculate its elements $L_{\mathbf{xy}}$:

$$L_{\mathbf{xy}} = \mathcal{D}_{\mathbf{xy}}^{\text{obj}} \mathcal{D}_{\mathbf{xy}}^{\text{dec}} / \sqrt{\mathcal{F}}, \quad (9)$$

where \mathcal{F} is used to evaluate the convergence of individuals. It is the non-dominated level of individuals in the population \mathcal{P} obtained after performing non-dominated sorting. In addition, $\mathcal{D}_{\mathbf{xy}}^{\text{obj}}$, $\mathcal{D}_{\mathbf{xy}}^{\text{dec}}$ are used to measure the diversity of individuals in the objective space and decision space. They are both calculated by the following formula:

$$\mathcal{D}_{(\mathbf{x}, \mathbf{y})} = \exp(-d(\mathbf{x}, \mathbf{y})), \quad (10)$$

where $d(\mathbf{x}, \mathbf{y})$ is the Euclidean distance between individuals \mathbf{x} and \mathbf{y} . In this way, $\mathcal{D}_{\mathbf{xy}}^{\text{obj}}$, $\mathcal{D}_{\mathbf{xy}}^{\text{dec}}$ are inversely proportional to the Euclidean distance between individuals. Therefore, the similarity between individuals is also taken into account in the selection of the kernel matrix \mathbf{L} . It is worth pointing out that the individuals \mathbf{x} and \mathbf{y} appearing in the above formula have been normalized.

3.2 The framework of MMOEADPPs

The overall flowchart of the proposed MMOEADPPs is shown in Fig. 2. In particular, the population \mathcal{P} and the archive \mathcal{A} are initialized. After that, a loop described below will continue until the termination condition is satisfied. First, mating pools are selected from the population and archive separately. Next, the offspring are produced and merged from the mating pools. Then, the next generation of the population and archive will be selected. The archive \mathcal{A} will be as the final output when the termination condition is met.

Algorithm 1 Framework of MMOEADPPs

Require: N (population size), G_{max} (termination condition)
Ensure: \mathcal{A} (final solution set)

- 1: $g \leftarrow 0$;
- 2: $\mathcal{P} \leftarrow$ Generate the initial population;
- 3: $\mathbf{z}_{obj}^*, \mathbf{z}_{dec}^* \leftarrow$ Initialize the ideal point of \mathcal{P} in both objective and decision spaces;
- 4: $\mathbf{z}_{obj}^{nad}, \mathbf{z}_{dec}^{nad} \leftarrow$ Initialize the nadir point of \mathcal{P} in both objective and decision spaces;
- 5: $\mathcal{A} \leftarrow$ Initialize the archive by HREA's archive update strategy;
- 6: **while** $g < G_{max}$ **do**
- 7: $\mathcal{M}_1 \leftarrow$ Randomly select $N/2$ solutions from \mathcal{P} ;
- 8: $\mathcal{O}_1 \leftarrow$ Generate offspring by GA operator based on \mathcal{M}_1 ;
- 9: $\mathcal{M}_2 \leftarrow$ Randomly select $N/2$ solutions from \mathcal{A} ;
- 10: $\mathcal{O}_2 \leftarrow$ Generate offspring by GA operator based on \mathcal{M}_2 ;
- 11: $\mathcal{O} \leftarrow \mathcal{O}_1 \cup \mathcal{O}_2$;
- 12: $\mathbf{z}_{obj}^*, \mathbf{z}_{dec}^* \leftarrow$ Update the ideal points of \mathcal{O} in both objective and decision spaces;
- 13: $\mathcal{P} \leftarrow$ UpdatePop($\mathcal{P}, \mathcal{O}, N, \mathbf{z}_{obj}^*, \mathbf{z}_{obj}^{nad}, \mathbf{z}_{dec}^*, \mathbf{z}_{dec}^{nad}$);
- 14: $\mathcal{A} \leftarrow$ Update the archive by HREA's archive update strategy;
- 15: $\mathbf{z}_{obj}^{nad}, \mathbf{z}_{dec}^{nad} \leftarrow$ Update the nadir points of \mathcal{P} in both objective and decision spaces;
- 16: $g \leftarrow g + 1$;
- 17: **end while**
- 18: **return** \mathcal{A}

Algorithm 1 shows the pseudocode of the overall framework of our proposed MMOEADPPs. First of all, setting current generation g to 0 and randomly initializing the population \mathcal{P} (lines 1-2). Next, according to Eqs. (7) and (8), the ideal point and the nadir point of the objective space and the decision space are initialized, respectively (lines 3-4). Then, using HREA's archive update strategy to initialize the archive \mathcal{A} based on \mathcal{P} (line 5), we set the maximum archive size to population size N in this study. The termination criterion is the maximum number of generations G_{max} set in advance. When the termination criterion has not been met (line 6), the following steps will be performed continuously.

- The mating pool \mathcal{M}_1 will consist of randomly selected $N/2$ solutions from \mathcal{P} (line 7) and the offspring \mathcal{O}_1 will be generated by the GA operator (line 8).
- Afterward, $N/2$ solutions selected at random from \mathcal{A} form the mating pool \mathcal{M}_2 (line 9) and the offspring \mathcal{O}_2 will be generated by the GA operator (line 10).
- Then, \mathcal{O}_1 and \mathcal{O}_2 are combined to offspring set \mathcal{O} (line 11).
- Next, the ideal points of the offspring in the objective space and the decision space are updated (line 12).
- Subsequently, the next generation of \mathcal{P} will be selected according to Algorithm 2 (line 13).

- And then, choosing the next generation of \mathcal{A} by HREA's archive update strategy (line 14).
- After that, the nadir points of the objective and decision spaces in \mathcal{P} will be updated and the generation index g increase by one (line 15-16).

At the end of evolution, \mathcal{A} will be taken as the final solution set.

3.3 Update of the population

The update of the population is based on DPPs. As given in Algorithm 2, the population update steps are as follows:

- First of all, the population \mathcal{P} is set to the union of itself and the offspring \mathcal{O} (line 1). The reason for this is that there may be solutions in the offspring that exist in

promising regions where PS may be present; at the same time, the information of solutions in the current population can also be fully utilized to explore more equivalent PSs.

Algorithm 2 UpdatePop(\mathcal{P} , \mathcal{O} , N , \mathbf{z}_{obj}^* , \mathbf{z}_{obj}^{nad} , \mathbf{z}_{dec}^* , \mathbf{z}_{dec}^{nad})

Require: \mathcal{P} (population), \mathcal{O} (offspring), N (population size), \mathbf{z}_{obj}^* (ideal point in objective space), \mathbf{z}_{obj}^{nad} (nadir point in objective space), \mathbf{z}_{dec}^* (ideal point in decision space), \mathbf{z}_{dec}^{nad} (nadir point in decision space)

Ensure: \mathcal{P}

- 1: $\mathcal{P} \leftarrow \mathcal{P} \cup \mathcal{O}$;
 - 2: Normalize \mathcal{P} ;
 - 3: $\mathcal{D}_{obj} \leftarrow$ Evaluate the diversity performance of solutions in \mathcal{P} in the objective space by Equation (10);
 - 4: $\mathcal{D}_{dec} \leftarrow$ Evaluate the diversity performance of solutions in \mathcal{P} in the decision space by Equation (10) ;
 - 5: $\mathcal{F} \leftarrow$ Obtain the Pareto rank of \mathcal{P} ;
 - 6: $\mathcal{L} \leftarrow$ Calculate the Kernel Matrix of \mathcal{P} based on Equation (9);
 - 7: $\mathcal{S} \leftarrow$ DPPs-Selection(\mathcal{L} , N);
 - 8: $\mathcal{P} \leftarrow \mathcal{P}(\mathcal{S})$;
 - 9: **return** \mathcal{P}
-

- Normalizing \mathcal{P} by Eq. (6) (line 2) to ensure that the solutions range within [0, 1].
- Then, the diversity of solutions in \mathcal{P} in the objective space and decision space is evaluated according to

Algorithm 3 DPPs-selection(\mathcal{L} , N)

Require: \mathcal{L} (Kernel Matrix), k (size of desired subset, i.e., the population size N)

Ensure: \mathcal{S} (selected subset)

- 1: ($\{\mathbf{v}_r\}_{r=1}^N, \{\lambda_r\}_{r=1}^N$) \leftarrow Eigen-decomposition (\mathbf{L});
 - 2: $\mathcal{V} \leftarrow$ Select k eigenvectors from $\{\mathbf{v}_r\}_{r=1}^N$ based on the largest k eigenvalues in $\{\lambda_r\}_{r=1}^N$;
 - 3: $\mathcal{S} \leftarrow \emptyset$;
 - 4: **while** $|\mathcal{V}| > 0$ **do**
 - 5: Select i with the largest value of $\sum_{\mathbf{v} \in \mathcal{V}} (\mathbf{v}_n^T e_i)^2$ from N ;
 - 6: $\mathcal{S} \leftarrow \mathcal{S} \cup i$;
 - 7: $\mathcal{V} \leftarrow \mathcal{V}_\perp$, the orthonormal basis for the subspace of \mathcal{V} orthogonal to e_i ;
 - 8: **end while**
 - 9: **return** \mathcal{S}
-

- Eq. (10) (line 3–4).
- Afterward, performing fast non-dominated sort in \mathcal{P} and the non-dominated fronts of the solutions are stored in \mathcal{F} (line 5).
- Next, calculating the kernel matrix according to Eq. (9), which is able to characterize the convergence and diversity of solutions in both decision and objective spaces (line 6).
- Subsequently, subsets are selected by the choose of DPPs-selection (line 7–8).
- Finally, the population \mathcal{P} will be output as the next generation (line 9).

A detailed description of the selection of determinantal point processes (DPPs-selection) is given in Algorithm 3. First, a set of eigenvectors $\{\mathbf{v}_r\}_{r=1}^N$ and eigenvalues $\{\lambda_r\}_{r=1}^N$ are obtained by eigendecomposition of the kernel matrix \mathcal{L} (line 1). Then, the eigenvalues in $\{\lambda_r\}_{r=1}^N$ are arranged in descending order to select the eigenvectors from $\{\mathbf{v}_r\}_{r=1}^N$ which associate with the k maximum eigenvalues (line 2). The k individuals with large differences are selected as the reference system. k is the desired subset size, which in our proposed algorithm is the population size N . After that, an empty set \mathcal{S} for output is created (line 3); Next, according

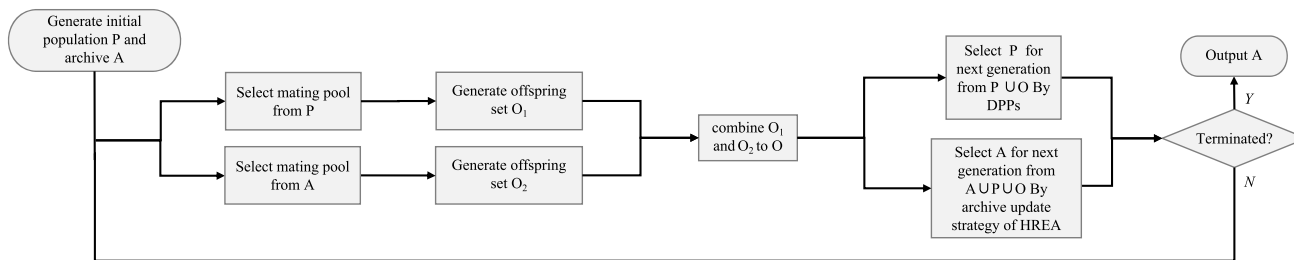


Fig. 2 Flowchart of the framework of MMOEADPPs

to the individual quality, extracting one offspring index i in each iteration until the population size reaches the requirement (line 4). The specific steps of iteration are as follows:

- First, picking the index of the element with the maximum value of $\sum_{v \in V} (v_n^T e_i)^2$ from N , where the dimension of e_i is N , only the i^{th} element of e_i is 1, and the rest is 0 (line 5).
- Then, adding the selected solution index to the index set S (line 6).
- Subsequently, finding out the orthonormal basis for the subspace of V which is orthogonal to e_i (line 7).

At last, the selected set of element indexes S is returned as output.

Although the update of the population comprehensively considers the convergence and diversity of individuals in the objective space and decision space, since the kernel matrix gives a relatively higher weight to the diversity, it focuses more on the consideration of diversity. Therefore, we introduce an archive that places extra emphasis on convergence of solutions while maintaining solution diversity. The archive update strategy of HREA preserves the non-dominated solutions to constitute the global PF. At the same time, the dominated solutions with good diversity and quality are retained to form a local PF. In addition, it balances the number of solutions on each PF to obtain a well-distributed PF [20]. For this reason, in our approach, we borrow its strategy of updating archive.

3.4 Computational complexity analysis

Each generation of the proposed MMOEADPPs needs to perform the following steps: (1) mating selection; (2) crossover and mutation; (3) update the population; and (4) update the archive. The mating pool uses a random selection method, so it consumes a runtime of $O(N)$, where N is the size of the population. GA operator is used to generate offspring from parents. Therefore, the computational complexity of crossover and mutation is $O(nN)$, where n is the scale of the decision variables. In addition, we adopt the

DPPs for the updating process of the population, which needs a runtime of $O(mN^2)$, where m is the number of objectives. Furthermore, the update of the archive takes $O(N^3)$. In consequence, the total computational complexity required by MMOEADPPs is $O(N^3)$.

4 Experimental results and analysis

In this section, we conduct experiments to verify the effectiveness of the proposed MMOEADPPs for solving MMOPs. First, the relevant settings of the experiments will be explained in detail in Sect. 4.1. Next, we demonstrate the feasibility and effectiveness of our proposed techniques, including the normalization of individuals, updating the population through DPPs, and the framework of the population and archive by separation experiments in Sect. 4.2. In Sect. 4.3, MMOEADPPs is compared with six state-of-the-art MMOEAs on two MMOP benchmarks. Subsequently, the parameter sensitivity of parameter e will be analyzed in Sect. 4.4. Finally, in Sect. 4.5, we use the proposed MMOEADPPs to deal with multimodal multi-objective location selection problem. All experiments for this work are carried out on the PlatEMO v3.4 [35] on a PC configured with an Intel i9-9900X @ 3.50GHz and 64G RAM.

Table 1 Parameters and the number of PS segments in the MMF test suite

Problem	m	n	Number of PS segments
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

Table 2 Parameters and the number of PS segments in the MMMOP test suite

Problems	Parameters						Number of PS segments
	m	n	K_A	K_B	c_i	d_i	
MMMOP1-A	2	3	1	1	–	–	5^{K_A}
MMMOP1-B	3	7	1	4	–	–	5^{K_A}
MMMOP2-A	2	3	1	1	–	–	6^{K_A}
MMMOP2-B	3	7	1	4	–	–	6^{K_A}
MMMOP3-A	2	2	0	1	–	3	$\prod_{i=m, \dots, m+K_A-1} c_i \times \prod_{i=1, \dots, m-1} d_i$
MMMOP3-B	3	7	0	5	–	2	$\prod_{i=m, \dots, m+K_A-1} c_i \times \prod_{i=1, \dots, m-1} d_i$
MMMOP3-C	2	6	1	4	3	3	$\prod_{i=m, \dots, m+K_A-1} c_i \times \prod_{i=1, \dots, m-1} d_i$
MMMOP3-D	3	7	1	4	2	2	$\prod_{i=m, \dots, m+K_A-1} c_i \times \prod_{i=1, \dots, m-1} d_i$
MMMOP4-A	2	2	0	1	–	4	$\prod_{i=m, \dots, m+K_A-1} c_i \times \prod_{i=1, \dots, m-1} d_i$
MMMOP4-B	3	7	0	5	–	3	$\prod_{i=m, \dots, m+K_A-1} c_i \times \prod_{i=1, \dots, m-1} d_i$
MMMOP4-C	2	6	1	4	2	4	$\prod_{i=m, \dots, m+K_A-1} c_i \times \prod_{i=1, \dots, m-1} d_i$
MMMOP4-D	3	7	1	4	2	3	$\prod_{i=m, \dots, m+K_A-1} c_i \times \prod_{i=1, \dots, m-1} d_i$
MMMOP5-A	2	2	0	1	–	3	$\prod_{i=m, \dots, m+K_A-1} c_i \times \prod_{i=1, \dots, m-1} d_i$
MMMOP5-B	3	7	0	5	–	1	$\prod_{i=m, \dots, m+K_A-1} c_i \times \prod_{i=1, \dots, m-1} d_i$
MMMOP5-C	2	6	1	4	2	2	$\prod_{i=m, \dots, m+K_A-1} c_i \times \prod_{i=1, \dots, m-1} d_i$
MMMOP5-D	3	7	1	4	2	1	$\prod_{i=m, \dots, m+K_A-1} c_i \times \prod_{i=1, \dots, m-1} d_i$
MMMOP6-A	2	2	0	1	2	–	$4^{K_A/2} \times \prod_{i=m+K_A, \dots, n} c_i$
MMMOP6-B	3	3	0	2	2	–	$4^{K_A/2} \times \prod_{i=m+K_A, \dots, n} c_i$
MMMOP6-C	2	4	2	1	2	–	$4^{K_A/2} \times \prod_{i=m+K_A, \dots, n} c_i$
MMMOP6-D	3	5	2	1	2	–	$4^{K_A/2} \times \prod_{i=m+K_A, \dots, n} c_i$

Table 3 Parameter settings on test problems, where N is the population size and G_{\max} is the maximal number of generations

	m	N	G_{\max}
MMF1-8	2	200	200
MMMOP	2	300	200
	3	300	300

4.1 Experimental settings

4.1.1 Benchmark for test

Two widely used benchmarks, MMF [3] and MMMOP [25], were selected as test problems to prove the validity of proposed MMOEADPPs. Table 1 gives a detailed introduction of the MMF test suite, including the number of objective functions m , the number of decision variables n and the number of equivalent PS segments corresponding

Table 4 Statistical results of IGDX obtained by MMOEADPPs and its variants on MMF benchmark problems

Problem	<i>M</i>	<i>D</i>	MMOEADualC	MMOEADNNSGAI	MMOEAPSO	MMOEAWI	MMOEADPPs
MMF1	2	2	3.1567e−2 (4.19e−4)−	3.1917e−2 (6.33e−4)	3.2332e−2 (1.09e−3)−	3.3271e−2 (1.10e−3)−	3.1315e−2 (5.74e−4)
MMF2	2	2	2.5878e−2 (1.62e−2)	3.3363e−2 (1.31e−2)	3.7006e−2 (1.94e−2)−	2.1064e−2 (1.15e−2)	1.0231e−2 (1.84e−3)
MMF3	2	2	1.9598e−2 (1.14e−2)	2.6584e−2 (1.62e−2)	2.5341e−2 (1.15e−2)−	1.5216e−2 (7.07e−3)	8.4017e−3 (1.08e−3)
MMF4	2	2	1.6470e−2 (3.72e−4) +	1.6789e−2 (4.11e−4)	1.6736e−2 (4.99e−4)	1.7961e−2 (4.86e−4)	1.7886e−2 (5.81e−4)
MMF5	2	2	5.5796e−2 (1.11e−3)	5.6097e−2 (1.04e−3)	5.6749e−2 (1.41e−3)−	5.8664e−2 (1.95e−3)−	5.4984e−2 (7.51e−4)
MMF6	2	2	5.2990e−2 (9.42e−4)	5.2956e−2 (8.86e−4)	5.3394e−2 (1.20e−3)	5.4777e−2 (1.59e−3)−	5.2746e−2 (1.14e−3)
MMF7	2	2	1.8672e−2 (1.10e−3)	1.8310e−2 (7.76e−4) +	1.8996e−2 (1.18e−3)	1.9120e−2 (1.05e−3)	2.0908e−2 (2.08e−3)
MMF8	2	2	5.8482e−2 (1.59e−2)	5.4429e−2 (1.85e−2)	5.8838e−2 (2.20e−2)−	4.7559e−2 (7.84e−3)−	3.7800e−2 (2.46e−3)
+ / − / ≈			2/5/1	2/5/1	2/6/0	1/6/1	

The best value of each row is bold

to the same PF. Similarly, the detailed introduction of the MMMOP test suite is given in Table 2. The lines with different colors for all figures in this article represent different PSs corresponding to the same PF.

4.1.2 Algorithms for comparison

Six state-of-the-art MMOEAs designed for MMOPs were selected as comparison algorithms to confirm the validity of our algorithm. They are listed below:

- Convergence-first algorithms: MO_Ring_PSO_SCD [3], DNNSGAI [16], TriMOEATAR [25], MMOEADC [17], HREA [20].
- Convergence-balanced algorithm: MMEAWI [19].

4.1.3 Parameter settings

The settings of specific parameters to all algorithms were consistent with those recommended in the original literature. For MO_Ring_PSO_SCD, $C_1 = C_2 = 2.05$ and $W = 0.7298$. As to TriMOEATAR, p_{con} , σ_{niche} , ϵ_{peak} were separately set to 0.5, 0.1 and 0.01. With regard to HREA, $p = 0.5$ and ϵ was set to 0.3. That is, the default parameters in PlatEMO were used.

All algorithms except MO_Ring_PSO_SCD used GA operator to generate offspring. The crossover and mutation operators were simulated binary crossover (SBX) [36] and polynomial mutation (PM) [37], respectively. Their parameter settings were as follows:

- The crossover probability p_c was set to 1; distribution index was $\eta_c = 20$;
- The mutation probability p_m was set to $1/n$, where n was the number of decision variables; distribution index was $\eta_c = 20$;

The main idea in designing the population size N and maximum number of generations G_{max} of experiments is to ensure that all algorithms have a chance to converge to PF and explore PSs as much as possible. With this in mind, we set them up as shown in Table 3. The parameter e of our MMOEADPPs is in the population update, which is temporarily set to 2. Its sensitivity studies are described in detail in Sect. 4.4.

4.1.4 Performance indicators

IGDX [38] and HV [39] are selected as performance indicators to compare the performance of different algorithms. IGDX can check how well the solution set found by the algorithm matches the real PSs, since the true PSs of the test problems we choose are known. It is calculated according to the following formula:

$$IGDX(\mathbf{X}) = \frac{1}{|\mathbf{X}^*|} \sum_{\mathbf{y} \in \mathbf{X}^*} \min_{\mathbf{x} \in \mathbf{X}} \{ED(\mathbf{x}, \mathbf{y})\}, \tag{11}$$

where \mathbf{X} is the obtained solution set and \mathbf{X}^* is the set of uniformly distributed reference points on the true PS. In this study, 10,000 uniformly distributed points were sampled on the real PS to calculate IGDX. $ED(\mathbf{x}, \mathbf{y})$ is the Euclidean distance between \mathbf{x} and \mathbf{y} . By measuring the

Table 5 Statistical results of IGDX obtained by MMOEADPPs and its variants on MMMOP benchmark problems

Problem	M	D	MMOEADualC	MMOEADNNSGAI	MMOEAPSO	MMOEAWI	MMOEADPPs
MMMOP1-A	2	3	3.3641e−2 (3.05e−2)	4.8295e−2 (4.83e−2)	5.0563e−2 (4.43e−2)	1.4891e−2 (1.72e−2)	4.5048e−3 (6.35e−5)
MMMOP1-B	3	7	4.1503e−1 (6.53e−2)	4.3001e−1 (6.67e−2)	4.1199e−1 (6.67e−2)	4.3360e−1 (5.99e−2)	3.3432e−1 (1.29e−2)
MMMOP2-A	2	3	3.4225e−3 (1.26e−3)	4.5729e−3 (4.38e−3)	4.3032e−3 (4.37e−3)	3.4405e−3 (1.65e−3)	3.2668e−3 (5.22e−4)
MMMOP2-B	3	7	2.6194e−1 (2.36e−2)	2.9630e−1 (2.22e−2)	2.9696e−1 (5.66e−2)	3.0368e−1 (2.09e−2)	2.6084e−1 (2.16e−2)
MMMOP3-A	2	2	7.0328e−3 (7.74e−4)	6.8216e−3 (5.02e−4)	6.8745e−3 (7.57e−4)	7.3708e−3 (9.17e−4)	8.4427e−3 (1.22e−3)
MMMOP3-B	3	7	2.9540e−1 (2.67e−2)	2.5433e−1 (3.37e−2)	2.4689e−1 (4.43e−2)	2.5064e−1 (5.31e−2)	2.2075e−1 (6.92e−2)
MMMOP3-C	2	6	9.0582e−2 (3.52e−2)	7.8251e−2 (3.77e−2)	7.3755e−2 (4.16e−2) +	1.1085e−1 (4.15e−2)	1.2772e−1 (4.21e−2)
MMMOP3-D	3	7	2.3466e−1 (3.25e−2)	2.2526e−1 (3.36e−2)	2.0239e−1 (4.64e−2)	2.0157e−1 (5.06e−2) ≈	2.0264e−1 (3.37e−2)
MMMOP4-A	2	2	8.8347e−4 (5.54e−6)	8.7981e−4 (6.20e−6)	8.7828e−4 (4.80e−6)	8.7228e−4 (4.57e−6)	8.6758e−4 (7.35e−6)
MMMOP4-B	3	7	2.4548e−2 (3.95e−4)	2.4111e−2 (1.18e−4)	2.4053e−2 (1.10e−4)	2.3949e−2 (1.77e−4) +	2.5450e−2 (6.84e−4)
MMMOP4-C	2	6	1.0971e−1 (1.25e−1)	1.2780e−1 (1.25e−1)	1.3530e−1 (1.25e−1)	9.4437e−2 (1.21e−1)	5.2994e−3 (4.13e−3)
MMMOP4-D	3	7	3.5480e−2 (8.27e−4)	3.4599e−2 (1.89e−4)	3.4502e−2 (1.77e−4) +	4.2107e−2 (4.16e−2)	3.5623e−2 (6.36e−4)
MMMOP5-A	2	2	8.8190e−4 (5.16e−6)	8.8194e−4 (5.80e−6)	8.7927e−4 (4.42e−6)	8.7026e−4 (2.78e−6)	8.6252e−4 (3.57e−6)
MMMOP5-B	3	7	2.4005e−2 (1.28e−4)	2.4155e−2 (9.24e−5)	2.4063e−2 (9.57e−5)	2.3911e−2 (8.38e−5) +	2.4091e−2 (1.16e−4)
MMMOP5-C	2	6	8.4951e−2 (1.19e−1)	1.5155e−1 (1.23e−1)	9.4587e−2 (1.21e−1)	1.0228e−1 (1.23e−1)	3.2553e−3 (1.87e−3)
MMMOP5-D	3	7	6.5025e−2 (7.87e−2)	9.5348e−2 (1.02e−1)	7.2563e−2 (8.63e−2)	5.7402e−2 (6.94e−2)	3.4699e−2 (1.91e−4)
MMMOP6-A	2	2	4.0826e−3 (2.37e−4)	3.9961e−3 (2.43e−4)	3.9433e−3 (1.69e−4) +	3.9710e−3 (8.45e−5)	4.1092e−3 (7.55e−5)
MMMOP6-B	3	3	4.4395e−2 (4.47e−4)	4.3487e−2 (2.49e−4)	4.3554e−2 (3.02e−4)	4.3743e−2 (3.20e−4)	4.3689e−2 (2.74e−4)
MMMOP6-C	2	4	3.2536e−1 (1.03e−1)	3.3035e−1 (8.94e−2)	3.3107e−1 (9.57e−2)	3.1020e−1 (8.48e−2)	3.2868e−2 (4.67e−3)
MMMOP6-D	3	5	3.4676e−1 (9.75e−2)	3.2251e−1 (9.71e−2)	3.2812e−1 (1.08e−1)	3.4300e−1 (8.97e−2)	1.4141e−1 (4.42e−2)
+ / − / ≈			4/9/7	6/10/4	5/8/7	4/7/9	

The best value of each row is bold

distance between the reference vector and the nearest point found, IGDX can well represent the diversity and convergence of the algorithm in the decision space. It is worth pointing out that a smaller value of IGDX indicates that an MMOEA is more efficient.

HV represents the volume or hypervolume enclosed between the obtained solution set and the reference points in the objective space. It measures the convergence and

spread of the solution set in the objective space. HV of the obtained solution set \mathbf{X} is calculated according to the following formula:

$$HV(\mathbf{X}) = VOL\left(\bigcup_{\mathbf{x} \in \mathbf{X}} [f_1(\mathbf{x}), \mathbf{r}_1^*] \times \cdots \times [f_m(\mathbf{x}), \mathbf{r}_m^*]\right), \tag{12}$$

where $VOL(\cdot)$ is the Lebesgue measure. \mathbf{r}^* is a set of reference points in the objective space. When calculating HV,

Table 6 Statistical results of HV obtained by MMOEADPPs and its variants on MMF benchmark problems

Problem	M	D	MMOEADualC	MMOEADNNSGAI	MMOEAPSO	MMOEAWI	MMOEADPPs
MMF1	2	2	9.0659e−1 (1.46e−4)	9.0656e−1 (2.37e−4)	9.0659e−1 (1.83e−4)	9.0660e−1 (2.68e−4)	9.0664e−1 (1.67e−4)
MMF2	2	2	8.5863e−1 (3.69e−3)	8.5796e−1 (2.74e−3)	8.5794e−1 (3.52e−3)	8.5843e−1 (3.24e−3)	8.5989e−1 (1.56e−4)
MMF3	2	2	8.1269e−1 (2.98e−3)	8.1226e−1 (1.83e−3)	8.1190e−1 (3.54e−3)	8.1331e−1 (3.81e−4)	8.1354e−1 (1.52e−4)
MMF4	2	2	7.2255e−1 (1.66e−4)	7.2239e−1 (9.83e−5)	7.2238e−1 (2.73e−4)	7.2235e−1 (1.11e−4)	7.2241e−1 (8.64e−5)
MMF5	2	2	9.6883e−1 (1.09e−4)	9.6881e−1 (1.46e−4)	9.6886e−1 (9.51e−5)	9.6886e−1 (4.31e−5)	9.6885e−1 (6.60e−5)
MMF6	2	2	9.5323e−1 (1.58e−4)	9.5317e−1 (1.79e−4)	9.5324e−1 (1.59e−4)	9.5323e−1 (1.41e−4)	9.5326e−1 (9.89e−5)
MMF7	2	2	8.8375e−1 (2.15e−4)	8.8382e−1 (1.08e−4)	8.8371e−1 (1.89e−4)	8.8376e−1 (2.55e−4)	8.8358e−1 (3.97e−4)
MMF8	2	2	9.7049e−1 (3.01e−4)	9.7048e−1 (3.32e−4)	9.7043e−1 (4.70e−4)	9.7026e−1 (5.25e−4)	9.7022e−1 (4.73e−4)
+ / − / ≈			3/0/5	2/2/4	0/2/6	1/3/4	

The best value of each row is bold

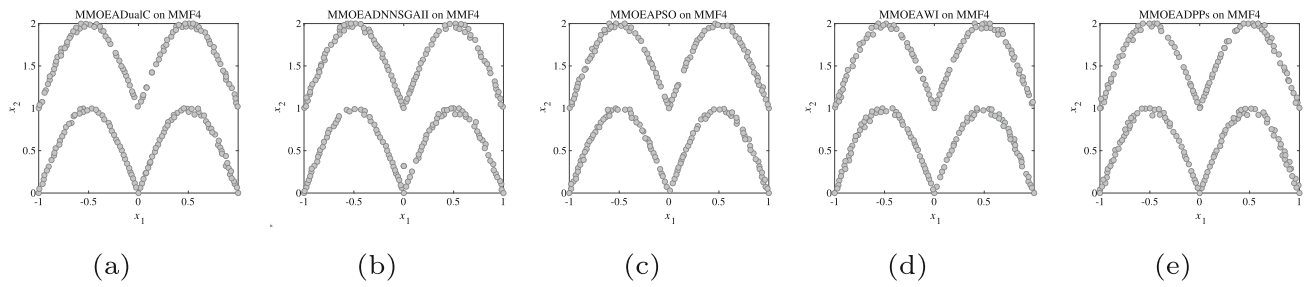


Fig. 3 Solutions obtained by MMOEADPPs and its variants in decision space with the median IGDX value among 30 runs on MMF4. **a** MMOEADualC, **b** MMOEADNNSGAI, **c** MMOEAPSO, **d** MMOEAWI, **e** MMOEADPPs

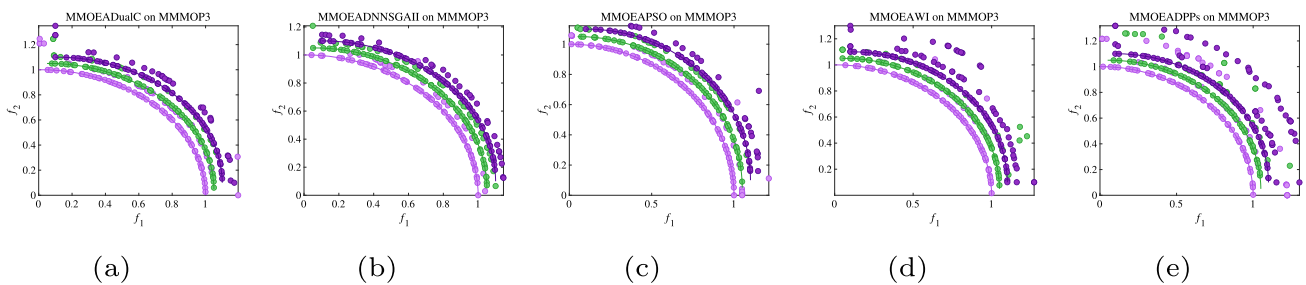


Fig. 4 Solutions obtained by MMOEADPPs and its variants in decision space with the median IGDX value among 30 runs on MMMOP3-A. **a** MMOEADualC, **b** MMOEADNNSGAI, **c** MMOEAPSO, **d** MMOEAWI, **e** MMOEADPPs

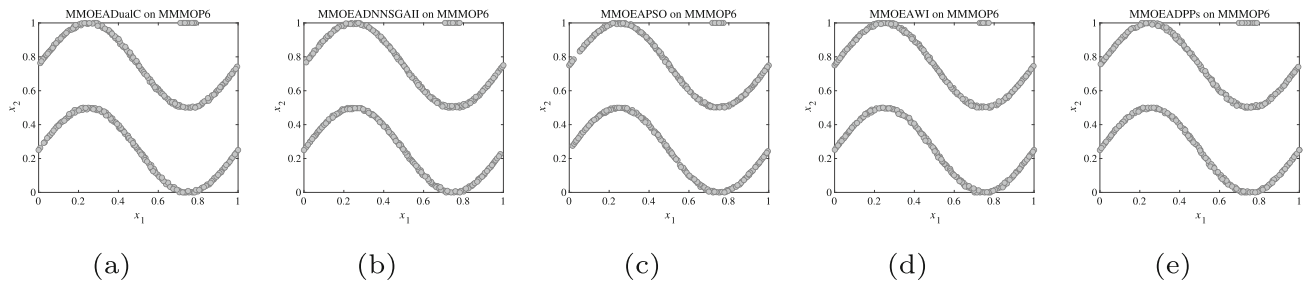


Fig. 5 Solutions obtained by MMOEADPPs and its variants in decision space with the median IGDX value among 30 runs on MMMOP6-A. **a** MMOEADualC, **b** MMOEADNNSGAI, **c** MMOEAPSO, **d** MMOEAWI, **e** MMOEADPPs

Table 7 Statistical results of IGDX obtained by MMOEADPPs and its variants on MMF benchmark problems

Problem	M	D	MMOEADPPsArchive	MMOEADPPsPop	MMOEADPPsNonnormalized	MMOEADPPsMPhybrid	MMOEADPPs
MMF1	2	2	3.3530e−2 (1.27e−3) −	4.5017e−2 (5.77e−4) −	3.1956e−2 (7.72e−4) −	3.1315e−2 (5.74e−4) ≈	3.1310e−2 (4.99e−4)
MMF2	2	2	4.9873e−2 (2.59e−2) −	1.7167e−2 (6.10e−4) −	1.1196e−2 (1.68e−3) −	1.0231e−2 (1.84e−3) ≈	9.3810e−3 (8.97e−4)
MMF3	2	2	2.7956e−2 (1.06e−2) −	1.5917e−2 (4.77e−4) −	9.2471e−3 (1.15e−3) −	8.4017e−3 (1.08e−3) ≈	8.6748e−3 (9.50e−4)
MMF4	2	2	1.7918e−2 (5.77e−4) ≈	3.4108e−2 (9.43e−4) −	1.7983e−2 (6.29e−4) ≈	1.7886e−2 (5.81e−4) ≈	1.7831e−2 (4.18e−4)
MMF5	2	2	5.9843e−2 (2.51e−3) −	7.3022e−2 (1.06e−3) −	5.5688e−2 (1.15e−3) −	5.4984e−2 (8.70e−4) ≈	5.4984e−2 (7.51e−4)
MMF6	2	2	5.5188e−2 (1.92e−3) −	6.2426e−2 (1.08e−3) −	5.2991e−2 (1.03e−3) ≈	5.2746e−2 (1.14e−3) ≈	5.2577e−2 (8.90e−4)
MMF7	2	2	1.9134e−2 (8.06e−4) +	3.6794e−2 (1.41e−3) −	2.1321e−2 (1.87e−3) ≈	2.0908e−2 (2.08e−3) ≈	2.0558e−2 (1.24e−3)
MMF8	2	2	8.2428e−2 (3.12e−2) −	7.3123e−2 (3.29e−3) −	3.7129e−2 (2.31e−3) ≈	3.7800e−2 (2.46e−3) ≈	3.7315e−2 (2.26e−3)
+ / − / ≈			1/6/1	0/8/0	0/4/4	0/0/8	

The best value of each row is bold

the objective values should be normalized first. In the normalized space, the reference point is (1.1, 1.1, . . . , 1.1). Note that a larger HV value generally represents a better result.

To ensure the reliability of the results, all the experimental outcomes of this study are obtained by running each algorithm independently for 30 times on each test problem. The mean and standard deviation values of IGDX are recorded. The experimental results are statistically analyzed using the Friedman test with Bonferroni correction at a significance level of 0.05 on the KEEL software [40]. Moreover, in the following table, “+,” “−” and “≈,” respectively, indicate that the performance of other algorithms is significantly better than, significantly worse than or similar to MMOEADPPs.

4.2 Studies on proposed components

In this section, we design different algorithm variants to demonstrate the effectiveness of our proposed strategies.

4.2.1 The effectiveness of embedded DPPs

In order to prove the effectiveness of incorporating DPPs into MMOEA, we replaced DPPs with the environment selection strategies of some advanced algorithms, including three convergence-first ones: MMOEADC [17], DNNSGAI [16], MO_Ring_PSO_SCD [3] and one convergence-balanced MMOEA: MMEAWI [19]. The average and standard deviation results of IGDX and HV obtained from 30 independent runs are shown in Tables 4, 5 and 6. It can be seen from the experimental results that MMOEADPPs performed better than other algorithm variants on the whole. However, on MMF, MMOEADPPs

Table 8 Statistical results of IGDX obtained by MMOEADPPs and its variants on MMMOP benchmark problems

Problem	M	D	MMOEADPPsArchive	MMOEADPPsPop	MMOEADPPsNonnormalized	MMOEADPPsMPhybrid	MMOEADPPs
MMMOP1-A	2	3	4.5002e−2 (5.41e−2) −	2.4453e−2 (1.08e−3) −	6.7842e−2 (7.49e−2) −	4.5048e−3 (6.35e−5) ≈	4.5031e−3 (7.58e−5)
MMMOP1-B	3	7	4.3866e−1 (6.75e−2) −	3.5493e−1 (8.82e−3) −	4.6669e−1 (7.16e−2) −	3.2732e−1 (1.19e−2) +	3.3432e−1 (1.29e−2)
MMMOP2-A	2	3	4.3969e−3 (5.56e−3) −	9.3045e−3 (1.71e−3) −	3.2668e−3 (5.22e−4) ≈	3.2446e−3 (3.87e−4) ≈	3.1848e−3 (3.67e−4)
MMMOP2-B	3	7	3.0178e−1 (5.84e−2) −	2.1869e−1 (9.98e−3) +	2.7037e−1 (2.07e−2) ≈	2.6321e−1 (1.88e−2) ≈	2.6084e−1 (2.16e−2)
MMMOP3-A	2	2	6.8833e−3 (5.81e−4) +	1.1632e−2 (1.04e−3) −	8.1288e−3 (9.62e−4) ≈	8.5515e−3 (1.14e−3) ≈	8.4427e−3 (1.22e−3)
MMMOP3-B	3	7	3.5582e−1 (3.84e−2) −	1.4324e−1 (4.82e−3) +	2.4398e−1 (4.58e−2) ≈	2.3672e−1 (3.96e−2) ≈	2.5064e−1 (5.31e−2)
MMMOP3-C	2	6	1.7485e−1 (2.79e−2) −	6.4174e−2 (3.01e−3) +	1.2510e−1 (3.82e−2) ≈	1.0980e−1 (3.44e−2) ≈	1.2772e−1 (4.21e−2)
MMMOP3-D	3	7	3.1559e−1 (3.79e−2) −	1.5238e−1 (5.12e−3) +	1.7892e−1 (2.55e−2) +	2.0053e−1 (3.35e−2) ≈	2.0264e−1 (3.37e−2)
MMMOP4-A	2	2	8.6088e−4 (2.65e−6) +	1.3756e−2 (6.03e−4) −	8.6217e−4 (5.10e−6) +	8.6556e−4 (4.07e−6) ≈	8.6758e−4 (7.35e−6)
MMMOP4-B	3	7	2.3870e−2 (1.02e−4) +	2.3881e−1 (1.54e−2) −	2.3923e−2 (2.23e−4) +	2.5133e−2 (4.84e−4) ≈	2.5450e−2 (6.84e−4)
MMMOP4-C	2	6	1.3450e−1 (1.26e−1) ≈	1.4217e−1 (2.41e−2) −	1.1921e−1 (1.17e−1) −	1.0456e−2 (7.47e−3) −	5.2994e−3 (4.13e−3)
MMMOP4-D	3	7	5.7103e−2 (6.96e−2) −	2.2590e−1 (1.52e−2) −	8.7092e−2 (9.70e−2) −	3.5753e−2 (8.82e−4) ≈	3.5623e−2 (6.36e−4)
MMMOP5-A	2	2	8.6252e−4 (3.57e−6) ≈	1.4405e−2 (6.98e−4) −	8.6421e−4 (6.11e−6) ≈	8.6349e−4 (3.90e−6) ≈	8.6154e−4 (3.75e−6)
MMMOP5-B	3	7	2.3880e−2 (9.29e−5) +	1.7875e−1 (1.71e−2) −	2.3950e−2 (1.78e−4) +	2.4102e−2 (9.23e−5) ≈	2.4091e−2 (1.16e−4)
MMMOP5-C	2	6	1.2638e−1 (1.26e−1) ≈	1.1640e−1 (2.47e−2) −	1.2463e−1 (1.20e−1) −	5.0358e−3 (3.55e−3) −	3.2553e−3 (1.87e−3)
MMMOP5-D	3	7	7.2299e−2 (8.64e−2) −	1.7044e−1 (1.80e−2) −	8.0235e−2 (9.26e−2) ≈	3.4673e−2 (1.59e−4) ≈	3.4699e−2 (1.91e−4)
MMMOP6-A	2	2	4.1732e−3 (1.93e−4) ≈	1.0468e−2 (2.58e−4) −	4.0719e−3 (1.38e−4) ≈	4.1135e−3 (1.17e−4) ≈	4.1092e−3 (7.55e−5)
MMMOP6-B	3	3	4.3689e−2 (2.74e−4) ≈	5.3500e−2 (5.28e−4) −	4.3678e−2 (3.50e−4) ≈	4.3841e−2 (3.41e−4) −	4.3610e−2 (2.95e−4)
MMMOP6-C	2	4	2.8720e−1 (8.05e−2) −	2.6019e−2 (7.28e−4) +	2.9018e−1 (1.55e−1) −	2.7507e−2 (2.25e−3) +	3.2868e−2 (4.67e−3)
MMMOP6-D	3	5	3.7299e−1 (8.31e−2) −	9.3445e−2 (8.90e−4) +	3.5086e−1 (9.79e−2) −	1.0785e−1 (3.51e−2) +	1.4141e−1 (4.42e−2)
+ / − / ≈			4/11/5	6/14/0	4/7/9	3/3/14	

The best value of each row is bold

was slightly worse on MMF4 and MMF7. For better observation, Fig. 3 shows the final solution distribution in decision space with the median IGDX values obtained by each algorithm on MMF4. As we can see, there is only a very small difference between MMOEADPPs and others. MMOEADPPs finally found the PS and had a relatively good diversity. On MMMOP, although MMOEADPPs performed well in most instances, its disadvantage was

clear on MMMOP3-A and MMMOP6-A. To investigate the causes of this phenomenon, Figs. 4 and 5, respectively, show the distribution of each algorithm in the objective space of MMMOP3-A and in the decision space of MMMOP6-A with the median IGDX values among 30 runs. It can be seen that on MMMOP3-A, MMOEADPPs failed to converge completely to PF, there were still many solutions floating in the objective space. For MMMOP6-A,

Table 9 Statistical results of HV obtained by MMOEADPPs and its variants on MMF benchmark problems

Problem	M	D	MMOEADPPsArchive	MMOEADPPsPop	MMOEADPPsNonnormalized	MMOEADPPsMPhybrid	MMOEADPPs
MMF1	2	2	9.0651e−1 (1.26e−4) –	9.0584e−1 (1.70e−4) –	9.0649e−1 (1.56e−4) –	9.0657e−1 (1.29e−4) ≈	9.0659e−1 (1.83e−4)
MMF2	2	2	8.5691e−1 (3.68e−3) –	8.5539e−1 (4.83e−4) –	8.5978e−1 (2.05e−4) –	8.5986e−1 (1.42e−4) ≈	8.5989e−1 (1.56e−4)
MMF3	2	2	8.1173e−1 (2.07e−3) –	8.0834e−1 (4.93e−4) –	8.1337e−1 (2.27e−4) –	8.1346e−1 (1.83e−4) –	8.1354e−1 (1.52e−4)
MMF4	2	2	7.2230e−1 (2.48e−4) ≈	7.2034e−1 (2.07e−4) –	7.2233e−1 (1.20e−4) –	7.2237e−1 (1.06e−4) ≈	7.2241e−1 (8.64e−5)
MMF5	2	2	9.6876e−1 (1.32e−4) –	9.6864e−1 (5.52e−5) –	9.6883e−1 (7.92e−5) ≈	9.6886e−1 (3.37e−5) ≈	9.6885e−1 (6.60e−5)
MMF6	2	2	9.5318e−1 (1.53e−4) –	9.5307e−1 (6.84e−5) –	9.5321e−1 (1.39e−4) ≈	9.5323e−1 (1.58e−4) ≈	9.5324e−1 (1.27e−4)
MMF7	2	2	8.8376e−1 (1.28e−4) +	8.8170e−1 (1.22e−3) –	8.8356e−1 (2.89e−4) ≈	8.8365e−1 (2.19e−4) ≈	8.8358e−1 (3.97e−4)
MMF8	2	2	9.7041e−1 (3.43e−4) ≈	9.6874e−1 (1.59e−3) –	9.7068e−1 (2.44e−4) +	9.7008e−1 (5.50e−4) ≈	9.7022e−1 (4.73e−4)
+ / − / ≈			1/5/2	0/8/0	1/4/3	0/1/7	

The best value of each row is bold

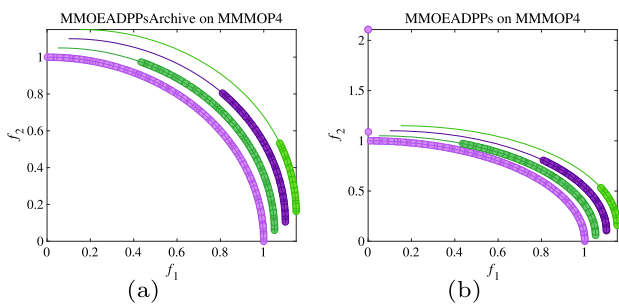


Fig. 6 Solutions obtained by MMOEADPPs and its variants MMOEADPPsArchive in objective space with the median IGDX value among 30 runs on MMMOP4-A. **a** MMOEADPPsArchive, **b** MMOEADPPs

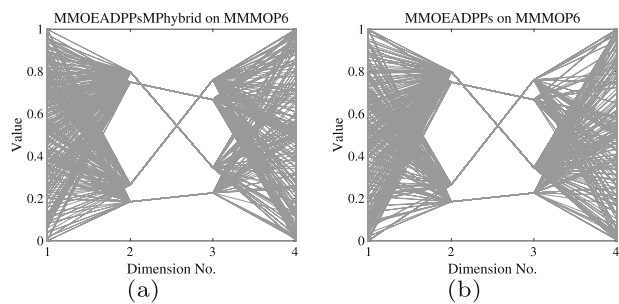


Fig. 7 Solutions obtained by MMOEADPPs and its variants MMOEADPPsPop in decision space with the median IGDX value among 30 runs on MMMOP6-C. **a** MMOEADPPsPop, **b** MMOEADPPs

each algorithm performed very well. However, because MMOEADPPs retained a few extreme points, its IGDX

index value was not good enough. But at the same time, the variant with the best IGDX indicator was not distributed continuously enough, indicating that the measurement of IGDX indicator was sometimes not accurate enough. In conclusion, the feasibility of embedding DPPs into MMOEA is confirmed.

4.2.2 The effectiveness of the proposed framework

In this section, we demonstrate the validity of the proposed framework by comparing MMOEADPPs with some of its variants:

- **MMOEADPPsArchive:** In this variant, population is not used in evolution to verify the need for population and archive framework.
- **MMOEADPPsPop:** Just population evolves to prove the framework works.
- **MMOEADPPsNonnormalized:** Individuals are not normalized before computing the kernel matrix.
- **MMOEADPPsMPhybrid:** Mating pool is selected from the union of population and archive.

The results of IGDX and HV are shown in Tables 7, 5 and 9, from which we can see the performance of MMOEADPPs was significantly better than that of MMOEADPPsArchive, MMOEADPPsPop and MMOEADPPsNonnormalized, but similar to that of MMOEADPPsMPhybrid, which may be due to the fact that the quality of the offspring produced was not greatly affected by the mixing of the mating pools. Besides,

Table 10 Statistical results of IGDX on MMF obtained by MMOEADPPs and other MMOEAs in comparison

Problem	<i>M</i>	<i>D</i>	DNNSGAI	HREA	MMEAWI	MMOADC	MO_Ring_PSO_SCD	TriMOEATAR	MMOEDPPs
MMF1	2	2	4.0630e−2 (1.09e−3) –	4.5555e−2 (3.04e−3) –	3.9315e−2 (2.38e−3) –	4.4168e−2 (1.50e−3) –	7.0439e−2 (1.53e−2) –	5.0633e−2 (5.95e−3) –	3.1315e−2 (5.74e−4)
MMF2	2	2	3.8920e−2 (2.76e−2) –	1.1003e−2 (2.57e−3) ≈	1.2651e−2 (5.00e−3) –	9.6972e−3 (4.61e−3) +	5.4905e−2 (1.99e−2) –	5.8015e−2 (3.54e−2) –	1.0231e−2 (1.84e−3)
MMF3	2	2	2.9610e−2 (2.08e−2) –	9.7549e−3 (1.70e−3) –	9.6593e−3 (1.75e−3) –	8.7299e−3 (3.29e−3) ≈	3.9927e−2 (1.29e−2) –	4.1935e−2 (1.91e−2) –	8.4017e−3 (1.08e−3)
MMF4	2	2	2.7888e−2 (2.43e−3) –	3.0238e−2 (3.08e−3) –	2.2946e−2 (9.17e−4) –	2.2331e−2 (2.58e−3) –	5.8339e−2 (1.87e−2) –	9.3070e−2 (1.48e−1) –	1.7886e−2 (5.81e−4)
MMF5	2	2	8.0978e−2 (4.99e−3) –	7.6525e−2 (4.61e−3) –	6.8993e−2 (2.24e−3) –	7.6504e−2 (2.73e−3) –	1.2223e−1 (2.00e−2) –	8.7805e−2 (8.89e−3) –	5.4984e−2 (7.51e−4)
MMF6	2	2	7.1807e−2 (4.00e−3) –	6.9031e−2 (4.51e−3) –	6.1703e−2 (1.81e−3) –	6.8639e−2 (2.35e−3) –	1.0263e−1 (1.29e−2) –	7.6052e−2 (7.19e−3) –	5.2746e−2 (1.14e−3)
MMF7	2	2	2.1696e−2 (1.29e−3) –	2.4422e−2 (1.83e−3) –	2.3600e−2 (1.46e−3) –	2.5461e−2 (3.90e−3) –	5.1773e−2 (1.14e−2) –	2.9677e−2 (2.03e−2) –	2.0908e−2 (2.08e−3)
MMF8	2	2	6.6935e−2 (1.68e−2) –	6.1039e−2 (5.19e−3) –	5.0133e−2 (3.90e−3) –	4.5980e−2 (5.23e−3) –	1.5853e−1 (5.86e−2) –	4.3409e−1 (7.98e−2) –	3.7800e−2 (2.46e−3)
+ / − / ≈			0/8/0	0/7/1	0/8/0	1/6/1	0/8/0	0/8/0	

The best value of each row is bold

without normalizing the individuals, the results were less than satisfactory. This possibly because it made the values in the kernel matrix had a large discrepancy, which was not conducive to the selection of subset by DPPs. On MMF, the superiority of MMOEADPPs was almost clear. However, on MMMOP, MMOEADPPsArchive was significantly better than MMOEADPPs on MMMOP4-A, MMMOP4-B and MMOEADPPsPop was also on MMMOP3, MMMOP6-C and MMMOP6-D. This phenomenon may be because, on some problems, it is more beneficial to explore and exploit equivalent PSs if all number of evaluations are given to individuals in the population that are emphasis on diversity or all of them are given to individuals in the archives that are biased toward convergence. In order to further find the reason, Figs. 6 and 7 show the distribution in the MMMOP4-A objective space and in the MMMOP6-C decision space with the median IGDX values, respectively. Figure 6 shows that MMOEADPPs did not sufficiently explore the equivalent PSs, and there were still large uncaptured areas. Meanwhile, compared to MMOEADPPsArchive, MMOEADPPs had several extreme points, making its IGDX value inferior to that of MMOEADPPsArchive. Therefore, the ability of MMOEADPPs to deal with extreme points needs to be

further improved. In Fig. 7, MMOEADPPs apparently did not find all the PS fragments, and there were some sparse places. But overall, our proposed population and archive framework is quite effective and achieves good results.

4.3 Comparisons with other methods

In this section, MMOEADPPs is compared with six state-of-the-art algorithms.

4.3.1 Comparison on MMF

The mean and standard deviation results of IGDX and HV for 30 independent runs are presented in Tables 10 and 11. On MMF, MMOEADPPs achieved the best results on all instances except MMF2. All results in decision space and objective space of MMOEADPPs based on the average IGDX value of 30 runs are given in Figs. 8 and 9. The solutions in objective and decision space with the median IGDX value among 30 runs on MMF2 are shown in Figs. 10 and 11. We can clearly observe that the diversity of MMOEADPPs was only slightly worse than that of MMOEADC, and the reason was that the diversity on the PS of decision space and PF of objective space were not

Table 11 Statistical results of HV on MMF obtained by MMOEADPPs and other MMOEAs in comparison

Problem	M	D	DNNSGAI	HREA	MMEAWI	MMOADC	MO_Ring_PSO_SCD	TriMOEATAR	MMOEDPPs
MMF1	2	2	9.0678e-1 (1.89e-4) +	9.0536e-1 (1.18e-3) -	9.0641e-1 (4.47e-4) -	9.0537e-1 (1.19e-3) -	9.0569e-1 (5.95e-4) -	9.0566e-1 (2.23e-3) -	9.0659e-1 (1.83e-4)
MMF2	2	2	8.5814e-1 (4.14e-3) -	8.5872e-1 (8.95e-4) -	8.5950e-1 (5.24e-4) -	8.5945e-1 (2.55e-4) -	8.4849e-1 (3.60e-3) -	8.5520e-1 (3.70e-3) -	8.5989e-1 (1.56e-4)
MMF3	2	2	8.1282e-1 (1.03e-3) -	8.1213e-1 (4.76e-4) -	8.1301e-1 (2.64e-4) -	8.1265e-1 (4.32e-4) -	8.0210e-1 (3.24e-3) -	8.0856e-1 (3.77e-3) -	8.1354e-1 (1.52e-4)
MMF4	2	2	7.2171e-1 (1.42e-3) -	7.2081e-1 (1.36e-3) -	7.2215e-1 (1.43e-4) -	7.2299e-1 (3.97e-4) +	7.2234e-1 (1.41e-4) \approx	7.0769e-1 (3.41e-2) -	7.2241e-1 (8.64e-5)
MMF5	2	2	9.6867e-1 (2.61e-4) -	9.6858e-1 (1.81e-4) -	9.6881e-1 (1.53e-4) \approx	9.6834e-1 (4.21e-4) -	9.6870e-1 (2.39e-4) -	9.6858e-1 (9.47e-4) \approx	9.6885e-1 (6.60e-5)
MMF6	2	2	9.5315e-1 (2.47e-4) \approx	9.5258e-1 (5.28e-4) -	9.5323e-1 (1.95e-4) \approx	9.5263e-1 (8.68e-4) -	9.5307e-1 (2.33e-4) -	9.5334e-1 (3.15e-4) +	9.5323e-1 (1.58e-4)
MMF7	2	2	8.8410e-1 (7.60e-5) +	8.8334e-1 (6.78e-4) \approx	8.8360e-1 (1.76e-4) \approx	8.8288e-1 (1.77e-3) \approx	8.8359e-1 (4.02e-4) \approx	8.8411e-1 (8.51e-4) +	8.8358e-1 (3.97e-4)
MMF8	2	2	9.7022e-1 (4.28e-4) \approx	9.6941e-1 (1.43e-3) -	9.7013e-1 (5.61e-4) \approx	9.7040e-1 (4.10e-4) \approx	9.7052e-1 (2.13e-4) +	9.6777e-1 (1.16e-2) -	9.7022e-1 (4.73e-4)
+ / - / \approx			2/4/2	0/7/1	0/4/4	1/5/2	1/5/2	2/5/1	

The best value of each row is bold

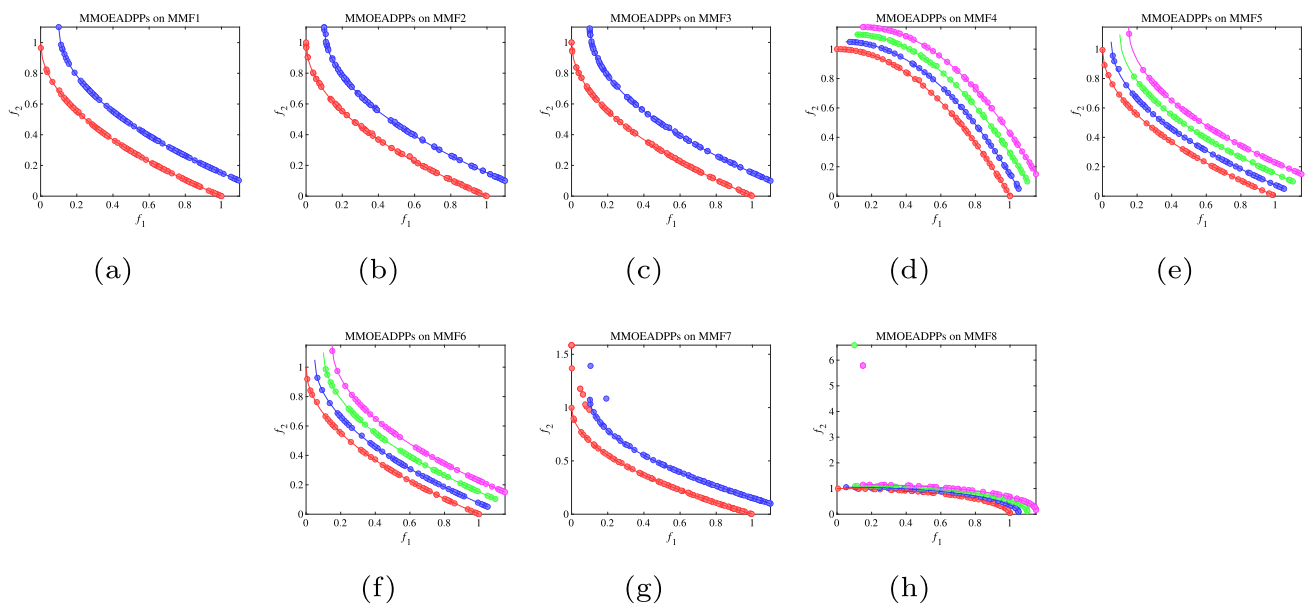


Fig. 8 Solutions in objective space with the median IGDX value among 30 runs on MMF instances obtained by MMOEADPPs

good enough. It can be seen that for the problem that the PS is distributed near the edge of decision space, rather than widely distributed in the whole decision space, the

performance of MMOEADPPs have need of further promotion.

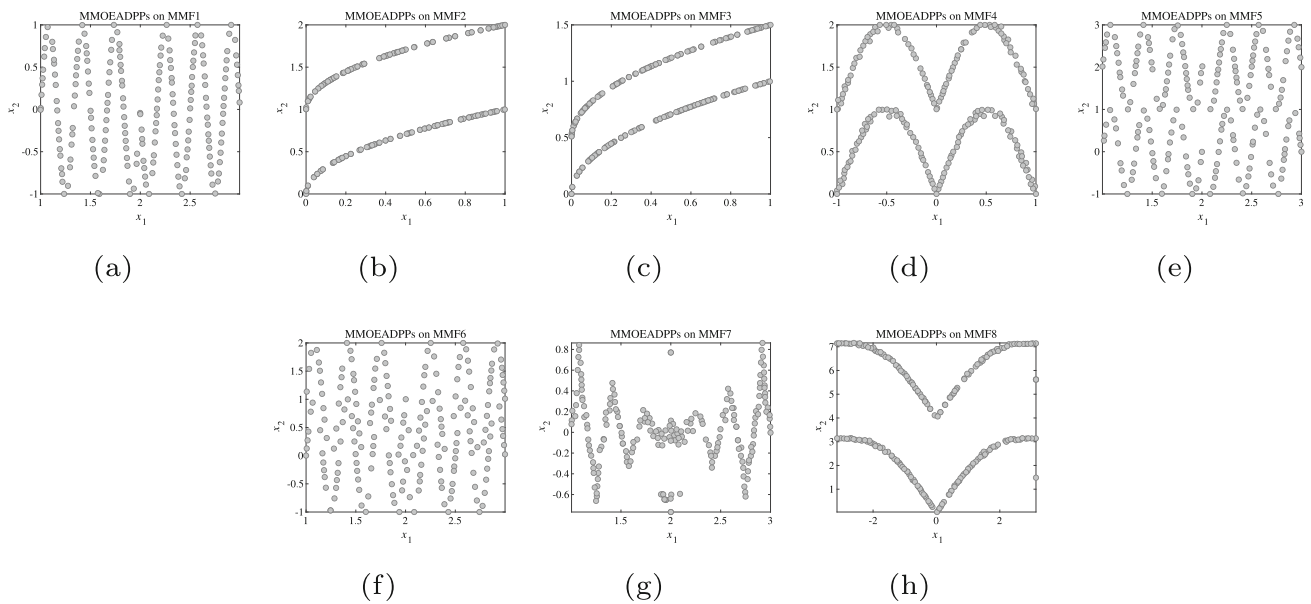


Fig. 9 Solutions with in decision space the median IGDX value among 30 runs on MMF instances obtained by MMOEADPPs

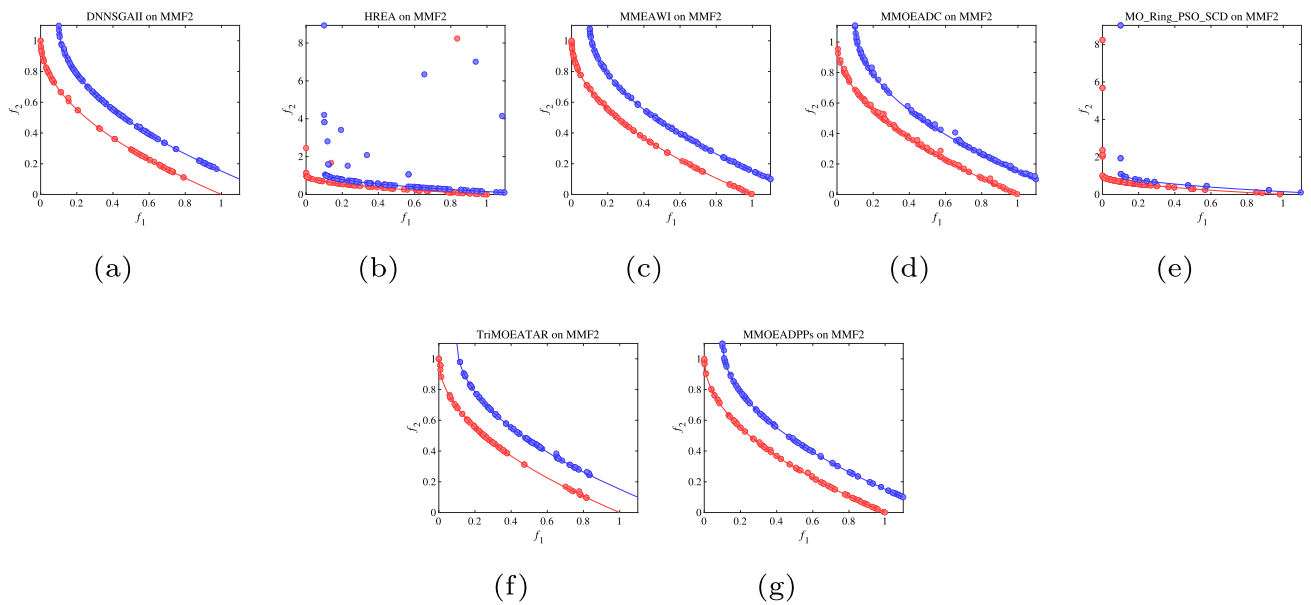


Fig. 10 Solutions in objective space with the median IGDX value among 30 runs on MMF2. **a** DNNSGAIL, **b** HREA, **c** MMEAWI, **d** MMOEADC, **e** MO_Ring_PSO_SCD, **f** TriMOEATAR, **g** MMOEADPPs

Figure 12 shows the PSs explored by all algorithms in the decision space on MMF5 based on the median IGDX. As mentioned earlier, each problem in the MMF test set has two equivalent PSs, which is also shown in Fig. 9. As shown in Fig. 12, all algorithms had found the two equivalent PSs, but only MMOEADPPs had the most comprehensive exploration of them, and its distribution in the decision space was also the most uniform. To further illustrate the comparison, the results of all algorithms on MMF8 in decision space are selected and shown in Fig. 13.

It can be seen that, except for MO_Ring_PSO_SCD and TriMOEATAR, which could not fully explore the PSs of the decision space, other algorithms could roughly cover the entire PSs. This was probably because MO_Ring_PSO_SCD only considered the diversity on the global PS of the decision space and ignored the local ones. Similarly, TriMOEATAR still paid more attention to the solutions with good convergence in the objective space, while MMOEADPPs achieved the best diversity of decision space compared to other algorithms. However, from

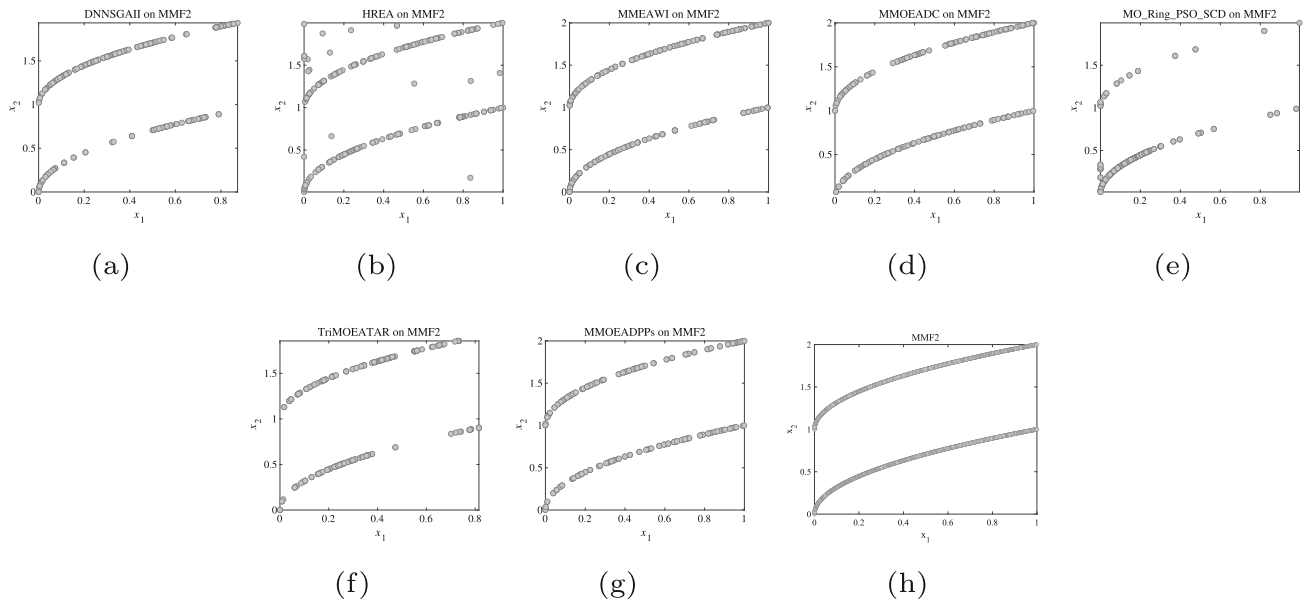


Fig. 11 Solutions in decision space with the median IGDX value among 30 runs on MMF2. **a** DNNSGAII, **b** HREA, **c** MMEAWI, **d** MMOEADC, **e** MO_Ring_PSO_SCD, **f** TriMOEATAR, **g** MMOEADPPs and **h** true PS

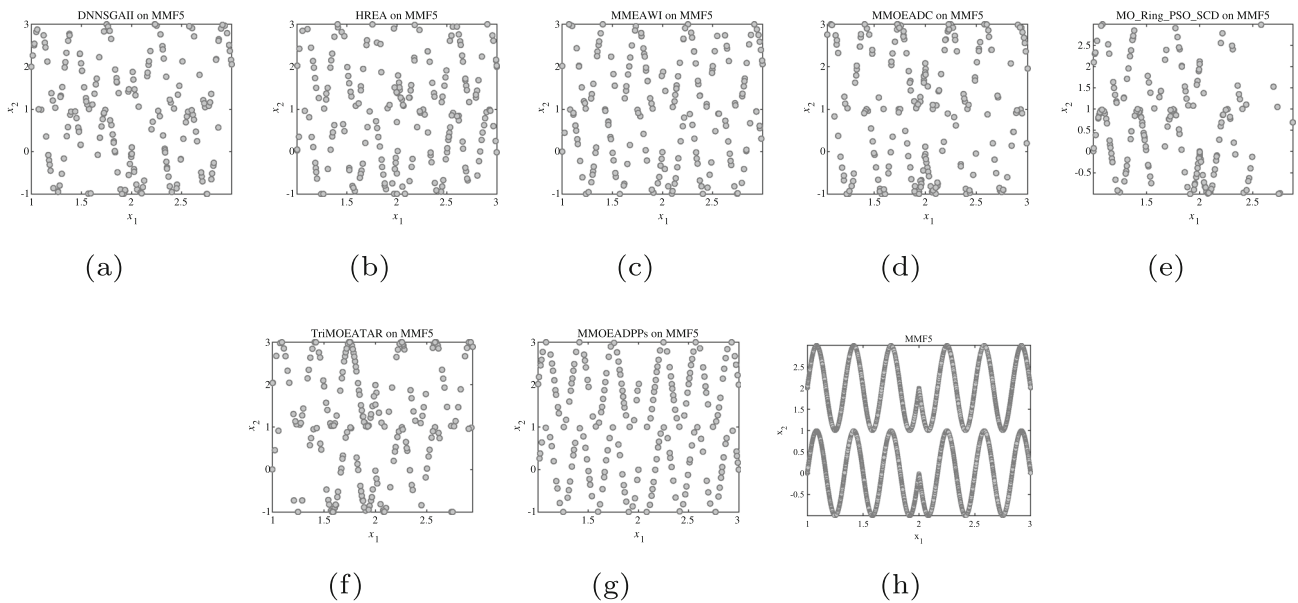


Fig. 12 Solutions in decision space with the median IGDX value among 30 runs on MMF5. **a** DNNSGAII, **b** HREA, **c** MMEAWI, **d** MMOEADC, **e** MO_Ring_PSO_SCD, **f** TriMOEATAR, **g** MMOEADPPs and **h** true PS

the distribution of MMOEADPPs, it can be seen that it retained several dominated solutions, which means that the ability of MMOEADPPs to control the dominated solutions in the final solution set needs to be improved. Figure 14 shows the convergence curves of MMOEADPPs and other algorithms on MMF5 and MMF8. It can be seen that MMOEADPPs not only converges faster, but also has the best final results.

4.3.2 Comparison on MMMOP

Table 12 shows the average and standard deviation results of IGDX values achieved by all algorithms on MMMOP, from which we can see, on the selected 20 test problems, MMOEADPPs achieved the best results than other algorithms on 11 test instances, followed by TriMOEATAR with nine for the best results. But we all know that

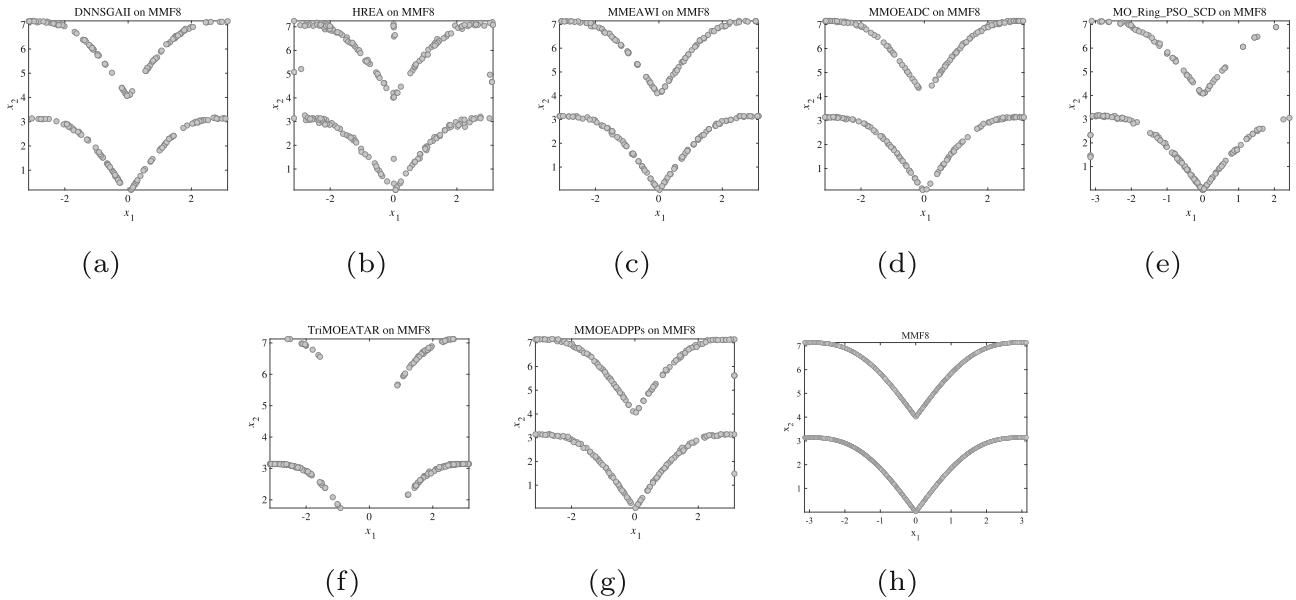
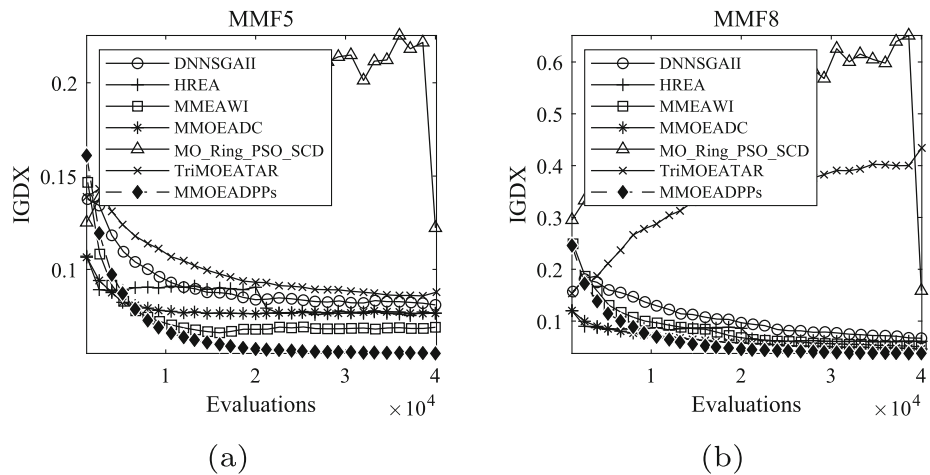


Fig. 13 Solutions in decision space with the median IGDX value among 30 runs on MMF8. **a** DNNSGAI, **b** HREA, **c** MMEAWI, **d** MMOEADC, **e** MO_Ring_PSO_SCD, **f** TriMOEATAR, **g** MMOEADPPs and **h** true PS

Fig. 14 Convergence curves of IGDX obtained by MMOEADPPs and other methods on MMF5 and MMF8 corresponding to results shown in Figs. 12 and 13



TriMOEATAR is tailored for MMMOP by the proposer of the MMMOP test set. Moreover, its decision variable analysis is out of reality, there are almost no independent convergence-related variables in real-world problems, and most of the variables are related [41]. At the same time, its recombination strategy makes the size of the final population expand by a factor of peak solutions based on the original population size N , which is unfair to the comparison of other algorithms [42]. But from the experimental results, our algorithm MMOEADPPs performed better than TriMOEATAR. It also reflects that MMOEADPPs has a strong ability to solve MMOPs. Figures 15 and 16,

respectively, show the distribution of MMOEADPPs in objective space and decision space on almost all test cases of MMMOP with the median IGDX values among 30 runs. In addition, Table 13 presents the multiple-problem statistical results of Friedman test.

However, on MMMOP3, the performance of MMOEADPPs was not good enough. To further analyze the reasons, in Figs. 17 and 18, we plot the results explored by all algorithms on MMMOP3-B. It can be seen that the final solution set of MMOEADPPs was relatively widely distributed in the decision space, so that it did not accurately explore the PSs, which also prevented it from

Table 12 Statistical results of IGDX on MMMOP obtained by MMOEADPPs and other MMOEAs in comparison

Problem	M	D	DNNSGAI	HREA	MMEAWI	MMOEADC	MO_Ring_PSO_SCD	TriMOEATAR	MMOEADPPs
MMMOP1-A	2	3	1.6884e-1 (6.35e-2)	1.8812e-2 (3.30e-3)	1.3171e-2 (1.41e-2)	1.9464e-2 (3.37e-2)	2.1543e-1 (5.11e-2) -	2.2024e-3 (4.93e-5) +	4.5048e-3 (6.35e-5)
MMMOP1-B	3	7	5.9537e-1 (6.65e-2)	4.4006e-1 (9.78e-3)	4.3416e-1 (9.49e-3)	4.5095e-1 (5.23e-3)	6.1854e-1 (3.43e-2) -	2.1748e-1 (1.92e-2) +	3.3432e-1 (1.29e-2)
MMMOP2-A	2	3	9.7013e-2 (9.97e-2)	5.2071e-3 (8.57e-4)	6.0368e-3 (5.72e-3)	3.5588e-3 (2.84e-4)	1.5385e-1 (7.41e-2) -	2.1916e-2 (3.58e-3) -	3.2668e-3 (5.22e-4)
MMMOP2-B	3	7	8.3858e-1 (6.29e-2)	4.2342e-1 (2.35e-2)	3.1748e-1 (3.02e-2)	5.1110e-1 (5.06e-2)	6.9086e-1 (6.26e-2) -	1.0223e-1 (8.39e-3) +	2.6084e-1 (2.16e-2)
MMMOP3-A	2	2	6.2526e-3 (3.24e-4)	9.0024e-3 (4.70e-4)	6.6941e-3 (3.45e-4)	7.6073e-3 (3.10e-4)	1.0866e-2 (1.18e-3) -	8.8065e-4 (3.19e-6) +	8.4427e-3 (1.22e-3)
MMMOP3-B	3	7	1.1070e-1 (3.29e-3)	4.4553e-1 (3.70e-2)	8.5079e-2 (2.60e-3)	2.3697e-1 (1.20e-2)	2.9119e-1 (2.20e-2) -	2.9595e-2 (4.40e-4) +	2.5064e-1 (5.31e-2)
MMMOP3-C	2	6	6.3655e-2 (5.86e-2)	3.4441e-1 (3.24e-2)	1.8560e-2 (9.48e-4)	5.6123e-2 (5.12e-3)	1.7701e-1 (2.57e-2) -	3.2510e-3 (8.88e-4) +	1.2772e-1 (4.21e-2)
MMMOP3-D	3	7	1.1670e-1 (4.79e-3)	4.2197e-1 (2.94e-2)	8.9204e-2 (2.58e-3)	2.4396e-1 (1.40e-2)	3.3314e-1 (2.53e-2) -	2.9784e-2 (5.04e-4) +	2.0264e-1 (3.37e-2)
MMMOP4-A	2	2	1.1228e-3 (5.01e-5)	1.9297e-3 (2.59e-4)	1.4032e-3 (6.85e-5)	2.7933e-3 (6.92e-4)	1.0581e-2 (2.76e-3) -	1.0534e-3 (7.00e-6) -	8.6758e-4 (7.35e-6)
MMMOP4-B	3	7	3.1754e-2 (6.92e-4)	4.8427e-1 (3.88e-2)	2.8000e-2 (4.54e-4)	2.8067e-1 (5.41e-2)	4.8640e-1 (3.47e-2) -	3.2862e-2 (4.92e-3) -	2.5450e-2 (6.84e-4)
MMMOP4-C	2	6	2.1750e-1 (7.86e-2)	3.2242e-1 (2.64e-2)	4.1028e-2 (2.94e-2)	2.6411e-2 (5.95e-2)	4.6993e-1 (5.04e-2) -	7.8922e-2 (1.01e-1) -	5.2994e-3 (4.13e-3)
MMMOP4-D	3	7	2.0451e-1 (7.44e-2)	4.6727e-1 (4.57e-2)	4.1176e-2 (6.95e-4)	3.4207e-1 (1.10e-1)	5.1336e-1 (4.45e-2) -	4.8170e-2 (6.03e-2) -	3.5623e-2 (6.36e-4)
MMMOP5-A	2	2	1.1316e-3 (2.38e-5)	1.8337e-3 (2.39e-4)	1.4149e-3 (5.89e-5)	1.5795e-3 (1.16e-4)	9.0532e-3 (3.14e-3) -	1.2674e-3 (8.52e-5) -	8.6252e-4 (3.57e-6)
MMMOP5-B	3	7	3.2262e-2 (7.78e-4)	5.0112e-1 (3.98e-2)	2.8162e-2 (4.80e-4)	1.1651e-1 (7.29e-2)	4.8831e-1 (4.37e-2) -	3.1391e-2 (1.17e-3) -	2.4091e-2 (1.16e-4)
MMMOP5-C	2	6	2.3454e-1 (5.30e-2)	3.2128e-1 (2.98e-2)	1.1370e-1 (7.98e-2)	3.9711e-2 (7.57e-2)	4.3858e-1 (5.90e-2) -	6.5485e-2 (1.07e-1) -	3.2553e-3 (1.87e-3)
MMMOP5-D	3	7	2.1874e-1 (6.16e-2)	4.8688e-1 (3.70e-2)	4.1299e-2 (6.88e-4)	1.3548e-1 (7.42e-2)	4.9143e-1 (4.17e-2) -	3.1077e-2 (1.51e-3) +	3.4699e-2 (1.91e-4)
MMMOP6-A	2	2	6.1105e-3 (7.59e-4)	5.9356e-3 (3.63e-4)	5.3949e-3 (1.99e-4)	6.4022e-3 (8.16e-4)	8.5966e-3 (1.21e-3) -	5.0120e-3 (3.32e-4) -	4.1092e-3 (7.55e-5)
MMMOP6-B	3	3	5.9472e-2 (1.76e-3)	5.3317e-2 (1.17e-3)	5.3561e-2 (1.12e-3)	6.9328e-2 (2.45e-3)	6.6450e-2 (5.98e-3) -	5.3115e-2 (1.17e-3) -	4.3689e-2 (2.74e-4)
MMMOP6-C	2	4	3.9208e-1 (4.56e-2)	3.2868e-2 (4.67e-3)	2.0835e-1 (3.19e-2)	3.5697e-2 (2.71e-2)	1.2387e-1 (2.27e-2) -	1.5177e-1 (9.62e-2) -	3.2834e-2 (2.38e-3)

Table 12 (continued)

Problem	M	D	DNNSGAI	HREA	MMEAWI	MMOADC	MO_Ring_PSO_SCD	TriMOEATAR	MMOEDPPs
MMMOP6-D	3	5	3.4388e−1 (4.41e−2)	2.2608e−1 (8.05e−3)	1.7638e−1 (2.34e−2)	1.4090e−1 (1.26e−2)	1.6413e−1 (1.56e−2) ≈	8.9695e−2 (3.56e−2) +	1.4141e−1 (4.42e−2)
+/-/≈			4/16/0	0/19/1	4/16/0	3/16/1	0/19/1	9/11/0	

The best value of each row is bold

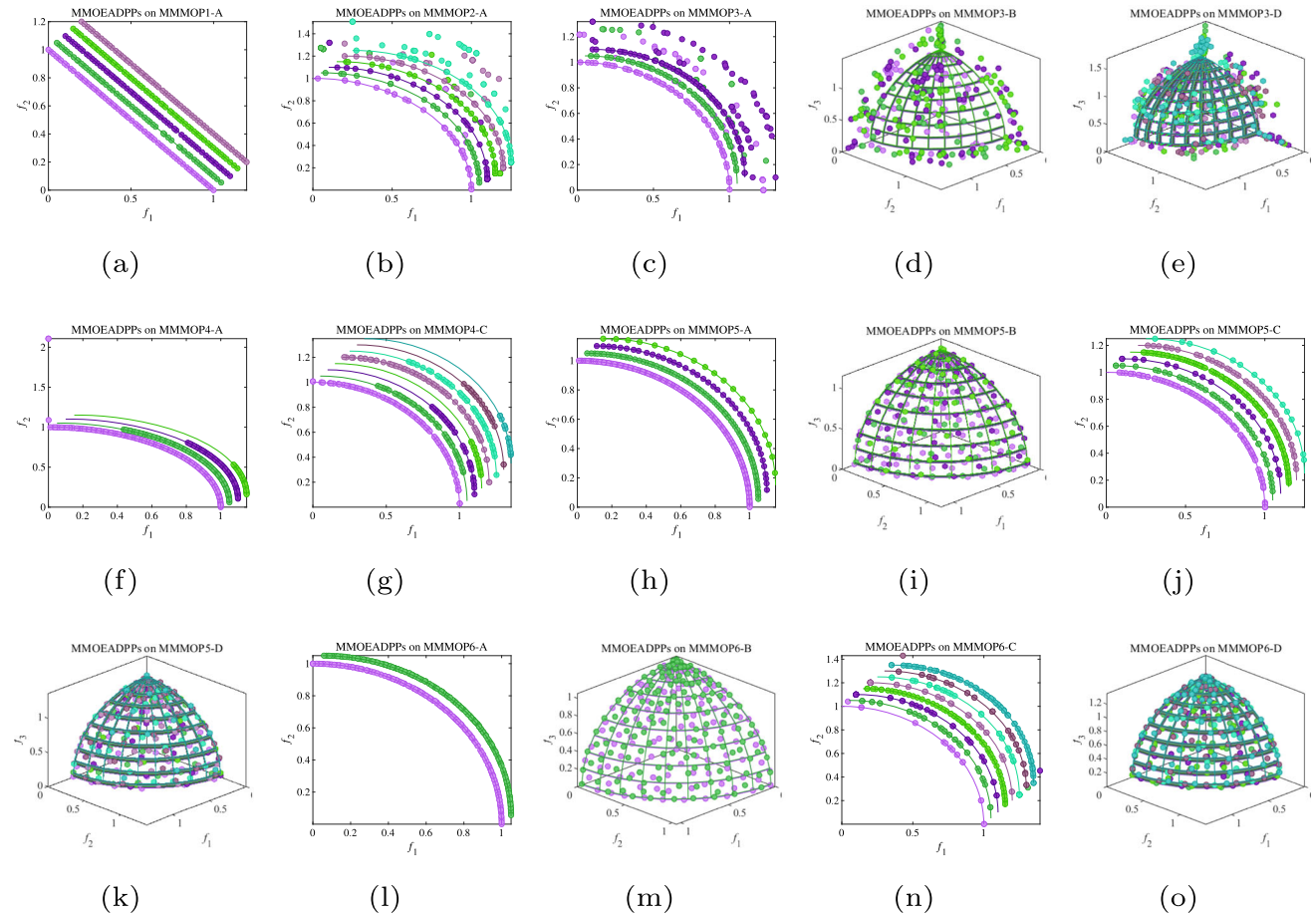


Fig. 15 Solutions in objective space with the median IGDX value among 30 runs on MMMOP instances obtained by MMOEDPPs

converging completely to the PF. Therefore, the convergence of MMOEDPPs on some problems needs to be strengthened. Likewise, MMMOP4-C and MMMOP5-A are selected. Figs. 19 and 20 gives the results on MMMOP4-C with the median IGDX values, the results on MMMOP5-A are shown in Figs. 21 and 22, to further study the comparisons. On MMMOP4-C, it can be seen that only MMOEDPPs detected all equivalent PSs, and the convergence to the PS in the decision space was also more consistent. MMOEDPPs also achieved the best convergence and diversity on MMMOP5-A; although the

fragment coverage with two PSs was not complete, it was overall very uniform. It is worth noting that when the solution set is very close to the PSs in the decision space, the coordinates of the distribution map will be more detailed, so that it looks a little messy due to excessive zooming, as in the diagram of MMOEDPPs in Fig. 22. The convergence curves obtained by MMOEDPPs and other methods on MMMOP4 and MMMOP6 are shown in Fig. 23. We can observe that compared with other algorithms, MMOEDPPs has better convergence performance.

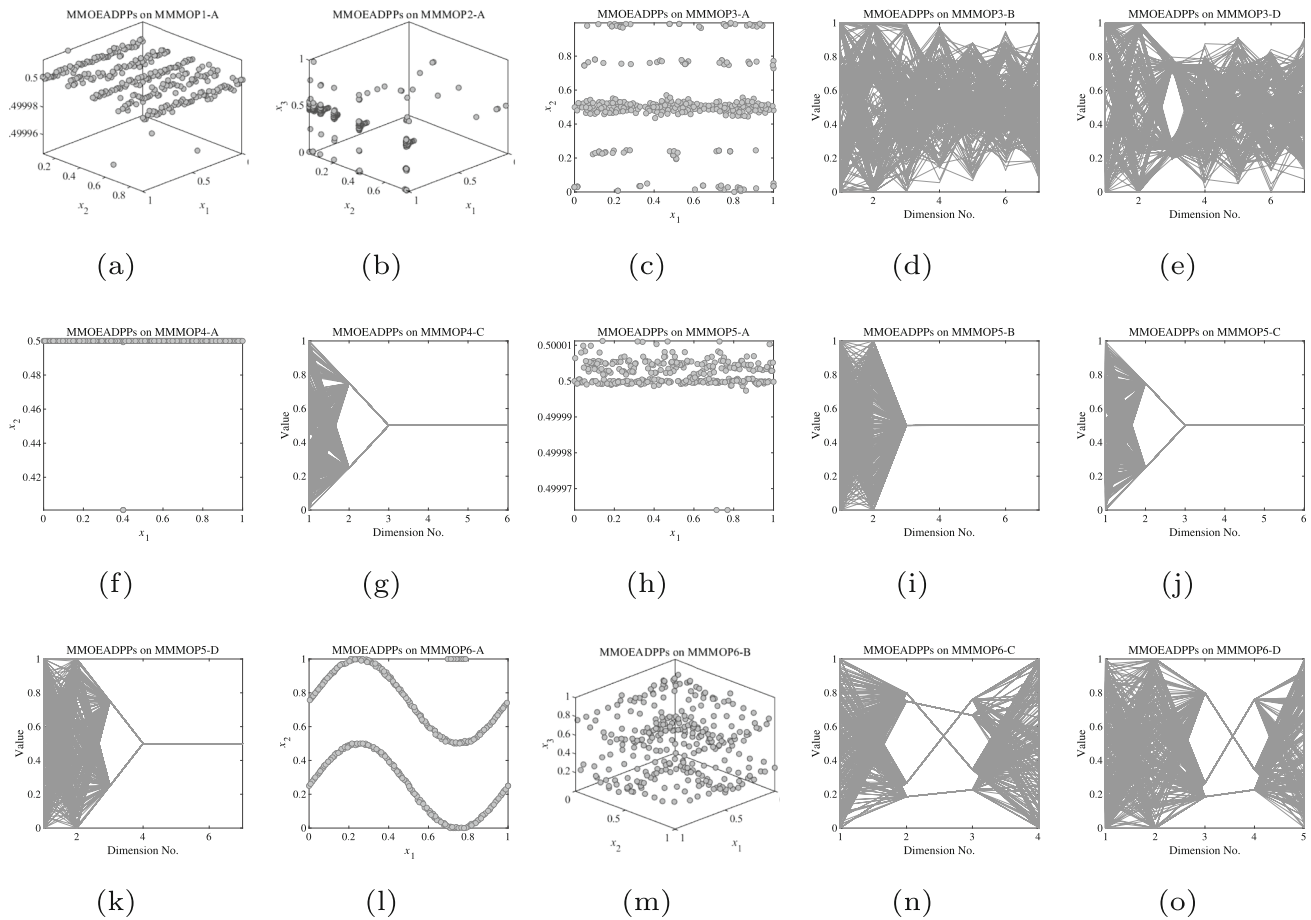


Fig. 16 Solutions in decision space with the median IGDX value among 30 runs on MMMOP instances obtained by MMOEADPPs

Table 13 Average Rankings of IGDX and HV by the Friedman test for MMOEADPPs and other MMOEAs in comparison

Algorithm	Ranking(IGDX)	Ranking(HV)
DNNSGAI	4.5	3.375
HREA	4.875	5.125
MMEAWI	3.0714	3.25
MMOEADC	3.8571	4.375
MO-Ring-PSO-SCD	6.4286	4.125
TriMOEATAR	3.4643	5.0625
MMOEADPPs	1.8036	2.6875

Bold values indicate the best results

4.4 Sensitivity analysis of parameter

In this section, we analyze the influence of different values of parameter e introduced in Eq. (9) in Sect. 3.1, which reflects the weight of the non-dominant layer \mathcal{F} in the kernel matrix \mathbf{L} . Therefore, a larger e means that convergence has a higher weight in the selection of subset via

DPPs, which leads to less focus on diversity. Based on this, in order to study the influence brought by different settings of parameter, we set several variants with different e values, *i.e.*, $\{1, 2, 4, 6, 8\}$. The mean values and standard deviations of IGDX and HV obtained from 30 independent runs on 28 test problems are given in Tables 15, 16 and 17. Furthermore, to further investigate the sensitivity of the parameters, we perform a multiple-problem analysis. Table 14 shows the average ranking of algorithm variants with different parameter settings by the Friedman test.

It can be seen from the tables that as e increases, such as 6 or 8, the performance of the algorithm is significantly reduced. This is due to the loss of diversity caused by too large e . When e is small, *e.g.*, 1, 2, there is relatively small difference on MMF. Reasonably, since we introduce an archive to further ensure the convergence of the algorithm while maintaining diversity, it can be expected that the kernel matrix will achieve better performance when considering the quality of the solution in a more emphasis on diversity way. Since the problems in the MMMOP test set have more PSs, a higher diversity weight, that is, $e = 1$,

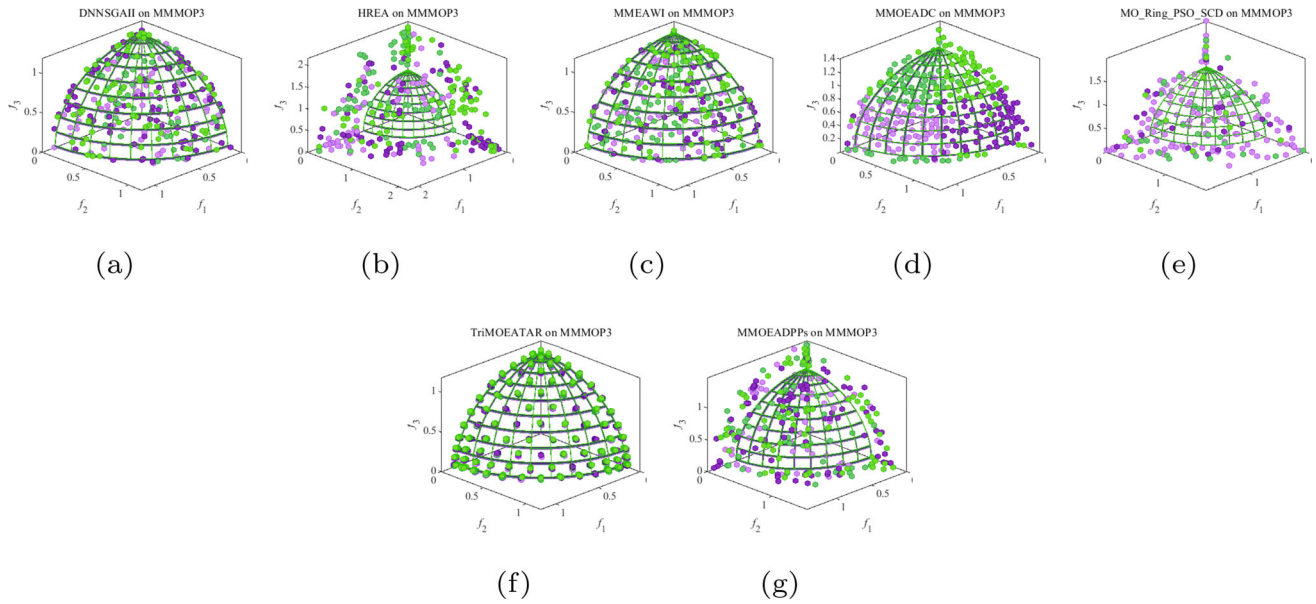


Fig. 17 Solutions in objective space with the median IGDX value among 30 runs on MMMOP3-B. **a** DNNSGAI, **b** HREA, **c** MMEAWI, **d** MMOEADC, **e** MO_Ring_PSO_SCD, **f** TriMOEATAR, **g** MMOEADPPs

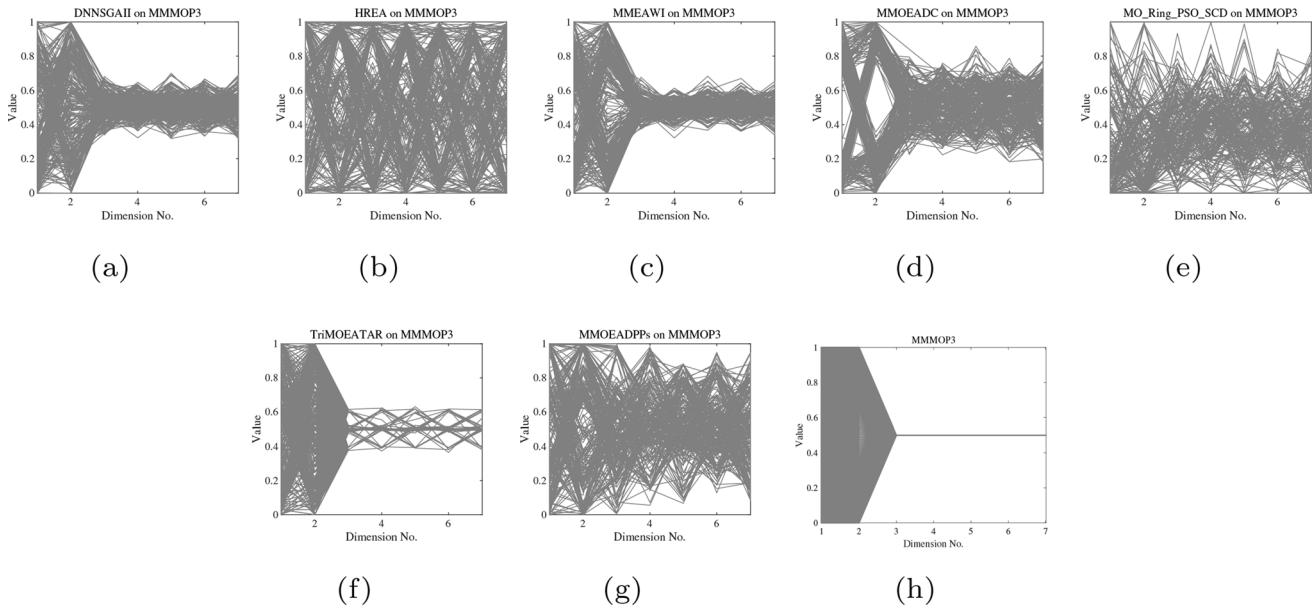


Fig. 18 Solutions in decision space with the median IGDX value among 30 runs on MMMOP3-B. **a** DNNSGAI, **b** HREA, **c** MMEAWI, **d** MMOEADC, **e** MO_Ring_PSO_SCD, **f** TriMOEATAR, **g** MMOEADPPs and **h** true PS

yields slightly better results than $e = 2$. However, from our statistical analysis of all experimental results, as shown in Table 14, the best average rankings on both IGDX and HV are consistently obtained when $e = 2$. As a result, $e = 2$ is considered to be set to better balance convergence and diversity.

4.5 Application

In this section, MMOEADPPs is compared with the other six algorithms on the multimodal multi-objective location selection problem [43], which is a simple real-world

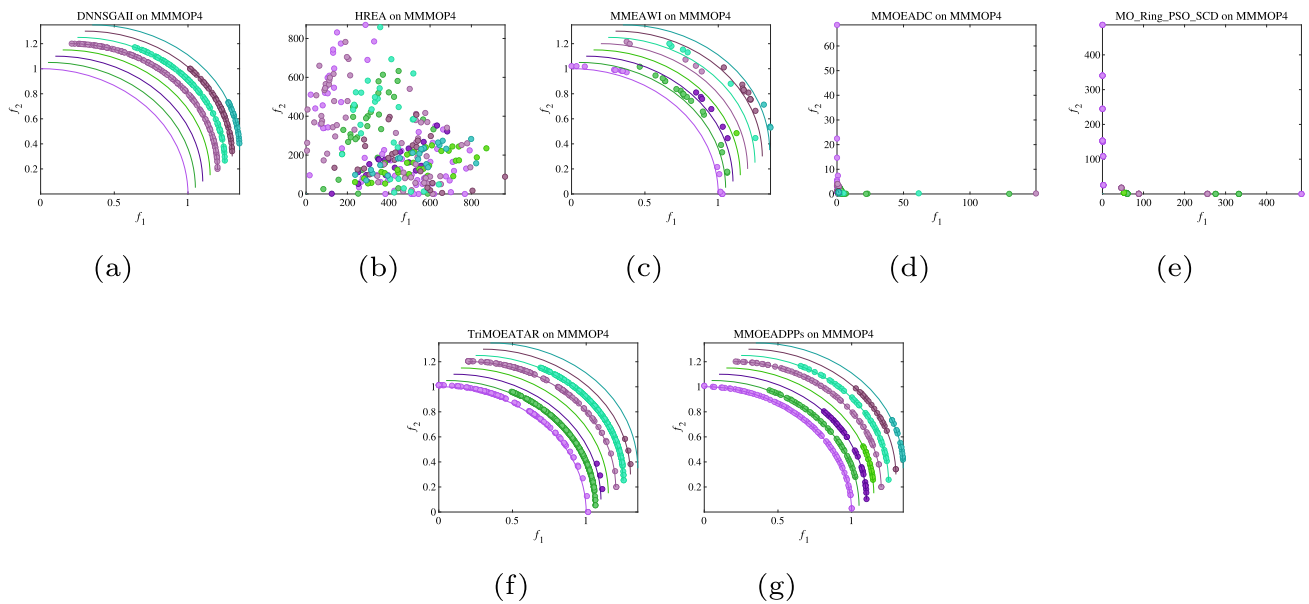


Fig. 19 Solutions in objective space with the median IGDX value among 30 runs on MMMOP4-C. **a** DNNSGAI, **b** HREA, **c** MMEAWI, **d** MMOEADC, **e** MO_Ring_PSO_SCD, **f** TriMOEATAR, **g** MMOEADPPs

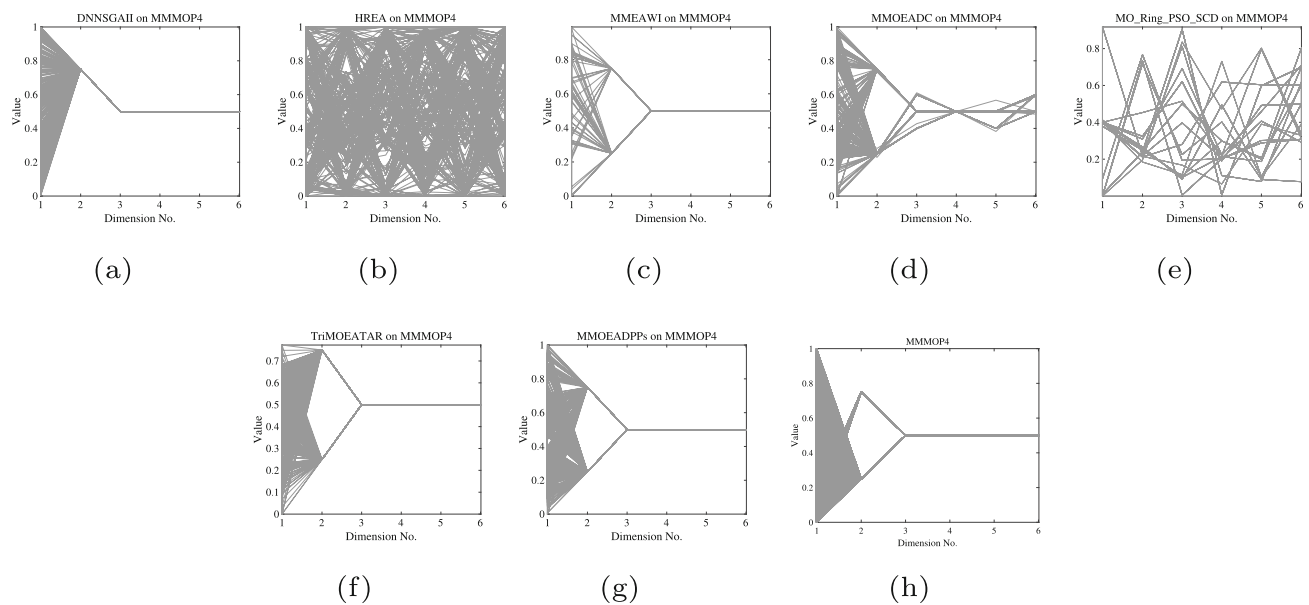


Fig. 20 Solutions in decision space with the median IGDX value among 30 runs on MMMOP4-C. **a** DNNSGAI, **b** HREA, **c** MMEAWI, **d** MMOEADC, **e** MO_Ring_PSO_SCD, **f** TriMOEATAR, **g** MMOEADPPs and **h** true PS

example, as shown in Fig. 24. In this map, tenants want to find locations that are close enough to hospitals, schools and convenience stores. Thus, this is a problem of distance minimization. Three non-dominated options with the same minimum distance to these objective locations are shown in Fig. 24, which are Option 1, Option 2 and Option 3, respectively.

There are more convenience stores next to Option 1, which may be more convenient for buying and living, but its location is relatively remote. Option 2 has more schools around it, so it may lead to higher rents. Option 3 locates in the suburb of the city, which makes it cheaper to rent. For some tenants, the convenience of life or low rents is more important, for others, the school district environment is necessary. If algorithms are unable to obtain multiple

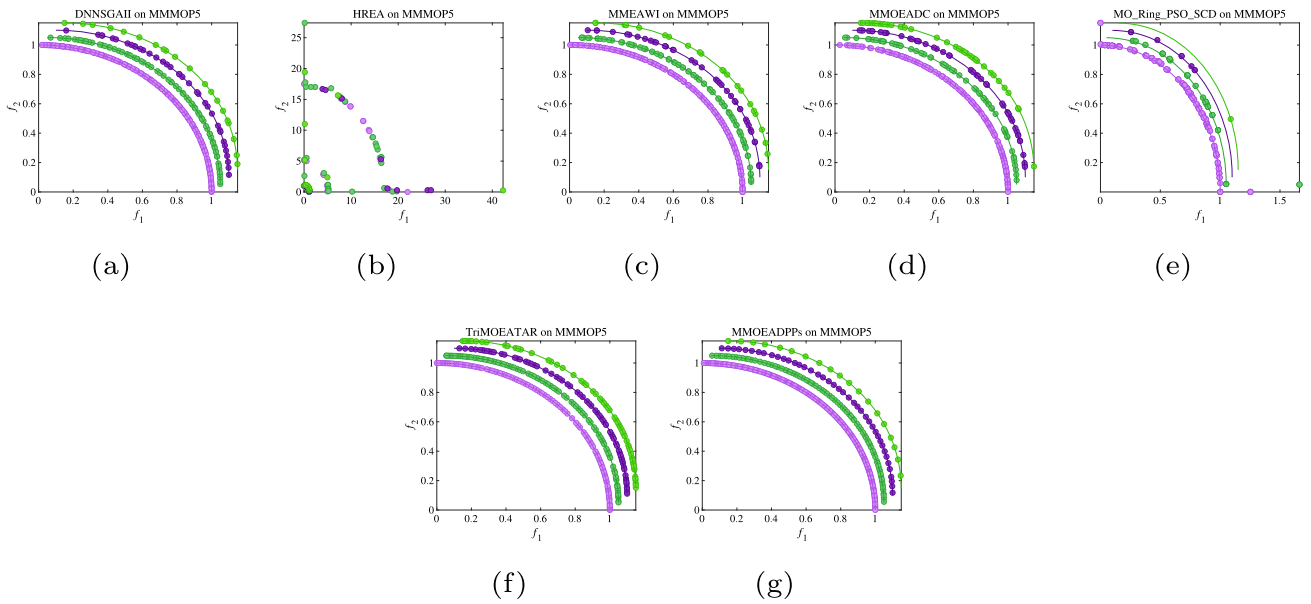


Fig. 21 Solutions in objective space with the median IGDX value among 30 runs on MMMOP5-A. **a** DNNSGAII, **b** HREA, **c** MMEAWI, **d** MMOEADC, **e** MO_Ring_PSO_SCD, **f** TriMOEATAR, **g** MMOEADPPs

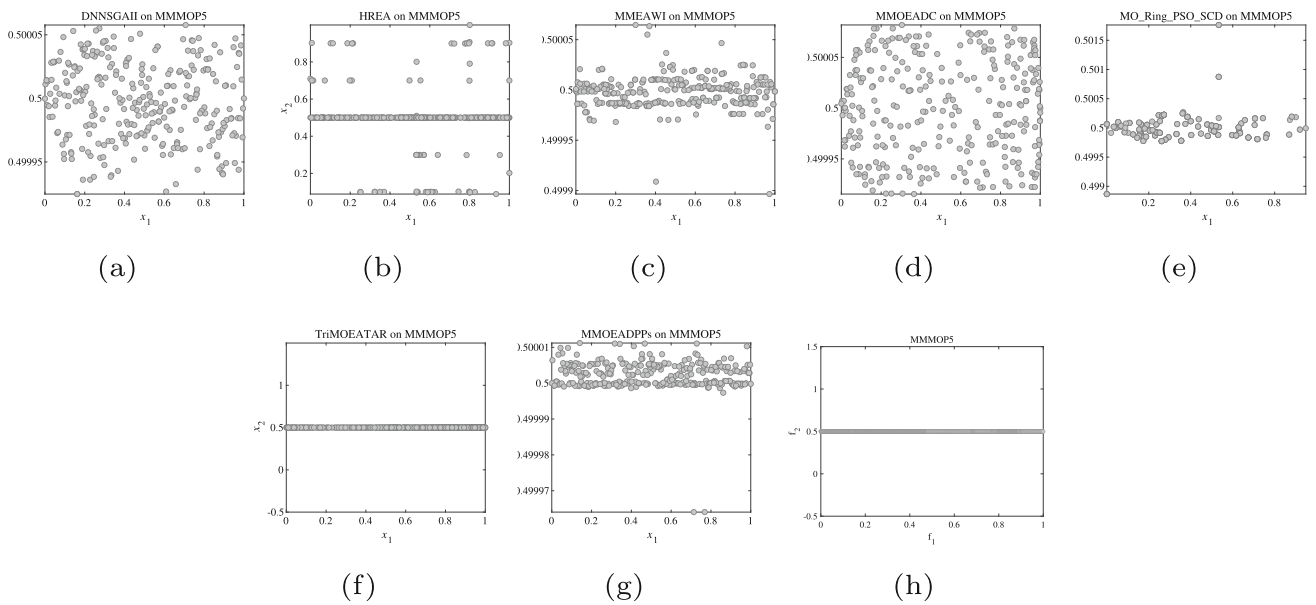


Fig. 22 Solutions in decision space with the median IGDX value among 30 runs on MMMOP5-A. **a** DNNSGAII, **b** HREA, **c** MMEAWI, **d** MMOEADC, **e** MO_Ring_PSO_SCD, **f** TriMOEATAR, **g** MMOEADPPs and **h** true PS

Pareto optimal solutions satisfying the conditions, they cannot meet the needs of different tenants. In this situation, it is valuable to retain more equivalent Pareto optimal solutions.

MMOEADPPs conducts comparative experiments on this problem with six other algorithms. The mean and standard deviation results of IGDX and HV for 30 independent runs are presented in Table 18. As given in

Table 18, MMOEADPPs achieves the best results in both IGDX and HV indicators, and MMEAWI also has a good result on HV indicator, probably because it proposes a weighted indicator to evaluate the potential convergence ability of individuals. Figure 25 shows the final solutions obtained by MMOEADPPs on this problem with the median IGDX value among 30 runs. Figure 25 shows that MMOEADPPs finds all the Pareto optimal regions and has

Fig. 23 Convergence curves of IGDX obtained by MMOEADPPs and other methods on MMMOP4-C and MMMOP6-A

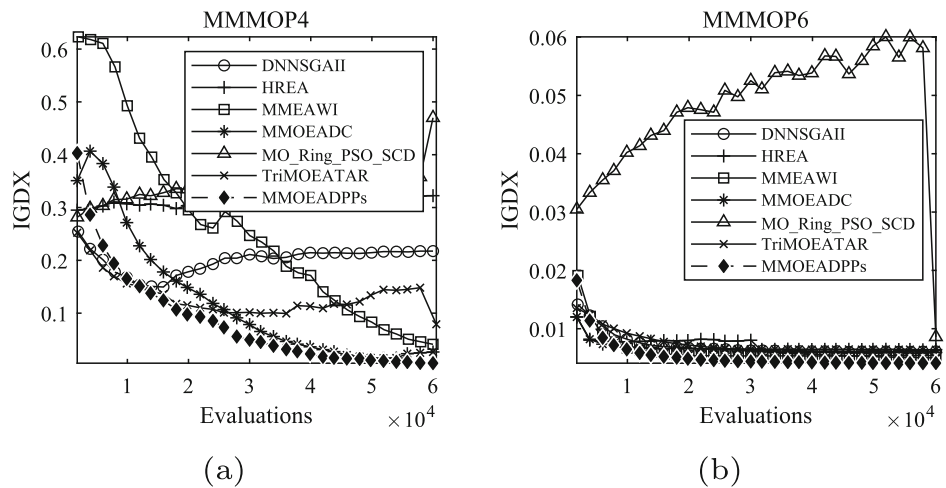


Table 14 Average rankings of IGDX and HV by the Friedman test for MMOEADPPs and its variants with different parameter settings

Algorithm	Ranking(IGDX)	Ranking(HV)
MMOEADPPs-e1	2.3571	2.625
MMOEADPPs-e4	2.875	3.25
MMOEADPPs-e6	3.5714	3.25
MMOEADPPs-e8	4.0357	3.25
MMOEADPPs	2.1607	2.549

5 Conclusions and future work

In this study, we apply DPPs to solve MMOPs for the first time. Different from the traditional method of selecting solutions one by one, MMOEADPPs select the subset at one time, which is used to balance the convergence and diversity of both objective space and decision space. By taking advantage of the DPPs’ self-selection property according to the probability and the correlation between individuals, the solution set obtained by our algorithm has better diversity. At the same time, the key of the design, that is, the kernel matrix, can comprehensively consider the convergence and diversity of the objective space and the diversity of the decision space, and can be well balanced among these three. In addition, the framework of population and archive is proposed. The population is used to better search the objective space and decision space, and the archive is used to store non-dominated solutions which can represent the PF and PSs. In the process of population evolution, we retain individuals with good convergence and more diversity, so as to improve the diversity of the whole. In the archive, individuals representing the global optimal PF as well as multiple different PSs are selected from the non-dominated solutions. In this way, we can avoid falling into local optimum, and at the same time, the search space is explored more to find more equivalent PSs. Experiments confirm the effectiveness of our proposed strategy and its superiority compared with six state-of-the-art algorithms.

However, during the experiments, we also found some limitations. For example, the convergence of MMOEADPPs on some problems was not strong enough under the existing population size and maximum number of evaluations, the existence of some extreme points made the algorithm sometimes not converge well to PF. Therefore,

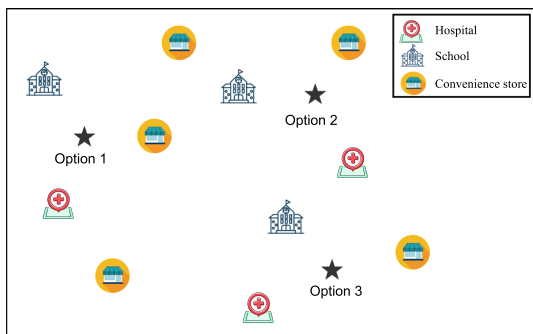


Fig. 24 Illustration of the multimodal multi-objective location selection problem

a remarkable diversity; however, there are also some extreme points that do not converge well, probably because the algorithm adopts a convergence-balanced strategy. In conclusion, MMOEADPPs can effectively locate multiple Pareto optimal regions, which is an effective approach to deal with multimodal multi-objective problems.

Table 15 Statistical results of IGD_X obtained by MMOEADPPs and its variants with different parameter settings for enlarging and reducing factors on MMF benchmark problems

Problem	<i>M</i>	<i>D</i>	MMOEADPPs-e1	MMOEADPPs-e4	MMOEADPPs-e6	MMOEADPPs-e8	MMOEADPPs
MMF1	2	2	3.1535e−2 (6.81e−4) ≈	3.1795e−2 (5.98e−4) –	3.1831e−2 (6.41e−4) –	3.2076e−2 (6.82e−4) –	3.1315e−2 (5.74e−4)
MMF2	2	2	1.0231e−2 (1.84e−3) ≈	1.0805e−2 (1.14e−3) –	1.2775e−2 (2.00e−3) –	1.3699e−2 (2.43e−3) –	1.0224e−2 (2.34e−3)
MMF3	2	2	8.4606e−3 (1.46e−3) ≈	9.9730e−3 (1.95e−3) –	1.0249e−2 (1.46e−3) –	1.0868e−2 (1.49e−3) –	8.4017e−3 (1.08e−3)
MMF4	2	2	1.7886e−2 (5.81e−4) ≈	1.7760e−2 (4.50e−4) ≈	1.8157e−2 (4.97e−4) –	1.7828e−2 (4.27e−4) ≈	1.7647e−2 (3.93e−4)
MMF5	2	2	5.5989e−2 (8.92e−4) –	5.5501e−2 (9.27e−4) –	5.5690e−2 (7.36e−4) –	5.5876e−2 (7.35e−4) –	5.4984e−2 (7.51e−4)
MMF6	2	2	5.2935e−2 (1.12e−3) ≈	5.3216e−2 (1.17e−3) ≈	5.3159e−2 (1.12e−3) ≈	5.3199e−2 (7.98e−4) ≈	5.2746e−2 (1.14e−3)
MMF7	2	2	2.0908e−2 (2.08e−3) ≈	2.0854e−2 (1.32e−3) ≈	2.0936e−2 (1.64e−3) ≈	2.0760e−2 (1.42e−3) ≈	2.0167e−2 (1.28e−3)
MMF8	2	2	4.2655e−2 (6.32e−3) –	3.7800e−2 (2.46e−3) ≈	3.8126e−2 (1.94e−3) ≈	3.8518e−2 (2.05e−3) ≈	3.7751e−2 (1.72e−3)
+ / − / ≈			0/2/6	0/4/4	0/5/3	0/4/4	

The best value of each row is bold

Table 16 Statistical results of HV obtained by MMOEADPPs and its variants with different parameter settings for enlarging and reducing factors on MMF benchmark problems. Best result in each row is highlighted

Problem	<i>M</i>	<i>D</i>	MMOEADPPs-e1	MMOEADPPs-e4	MMOEADPPs-e6	MMOEADPPs-e8	MMOEADPPs
MMF1	2	2	9.0659e−1 (1.83e−4) ≈	9.0648e−1 (2.35e−4) –	9.0656e−1 (1.09e−4) ≈	9.0654e−1 (1.70e−4) ≈	9.0660e−1 (8.96e−5)
MMF2	2	2	8.6004e−1 (1.78e−4) +	8.5962e−1 (1.86e−4) –	8.5941e−1 (3.91e−4) –	8.5934e−1 (3.31e−4) –	8.5989e−1 (1.56e−4)
MMF3	2	2	8.1362e−1 (1.57e−4) +	8.1322e−1 (2.36e−4) –	8.1317e−1 (2.66e−4) –	8.1304e−1 (2.50e−4) –	8.1354e−1 (1.52e−4)
MMF4	2	2	7.2239e−1 (1.28e−4) ≈	7.2237e−1 (1.53e−4) ≈	7.2239e−1 (8.77e−5) ≈	7.2237e−1 (1.06e−4) ≈	7.2241e−1 (8.64e−5)
MMF5	2	2	9.6884e−1 (8.88e−5) ≈	9.6885e−1 (6.60e−5) ≈	9.6883e−1 (8.53e−5) ≈	9.6881e−1 (9.82e−5) –	9.6885e−1 (4.52e−5)
MMF6	2	2	9.5318e−1 (2.35e−4) ≈	9.5324e−1 (9.59e−5) ≈	9.5323e−1 (1.58e−4) ≈	9.5324e−1 (6.46e−5) ≈	9.5326e−1 (7.66e−5)
MMF7	2	2	8.8362e−1 (2.34e−4) ≈	8.8354e−1 (2.29e−4) ≈	8.8358e−1 (3.97e−4) ≈	8.8358e−1 (2.58e−4) ≈	8.8364e−1 (2.45e−4)
MMF8	2	2	9.7022e−1 (4.73e−4) ≈	9.7008e−1 (3.85e−4) ≈	9.7030e−1 (4.55e−4) ≈	9.7031e−1 (4.33e−4) ≈	9.7034e−1 (2.96e−4)
+ / − / ≈			2/0/6	0/3/5	0/2/6	0/3/5	

Best value of each row is bold

we will further search for methods to effectively remove extreme points. In addition, how to better improve the diversity of the solution set obtained by MMOEADPPs is also worthy of further study. Since the kernel matrix of DPPs-selection is a key component of the proposed algorithm MMOEADPPs, we will try to improve it further in

future work and consider how to solve the phenomenon that the extreme points appearing in part of the test problems affect the final convergence. Finally, it is worthwhile to test MMOEADPPs on the imbalanced distance minimization benchmark problems (IDMPs) [21] and extend

Table 17 Statistical results of IGDX obtained by MMOEADPPs and its variants with different parameter settings for enlarging and reducing factors on MMMOP benchmark problems

Problem	M	D	MMOEADPPs-e1	MMOEADPPs-e4	MMOEADPPs-e6	MMOEADPPs-e8	MMOEADPPs
MMMOP1-A	2	3	4.4651e-3 (8.45e-5) +	7.2025e-3 (8.81e-3) -	1.2288e-2 (1.44e-2) -	1.3157e-2 (1.46e-2) -	4.5048e-3 (6.35e-5)
MMMOP1-B	3	7	3.7932e-1 (3.59e-2) -	3.3432e-1 (1.29e-2) ≈	3.3999e-1 (1.41e-2) ≈	3.4982e-1 (2.40e-2) -	3.2826e-1 (1.03e-2)
MMMOP2-A	2	3	3.2668e-3 (5.22e-4) ≈	3.2792e-3 (3.48e-4) ≈	3.0214e-3 (3.35e-4) ≈	3.3855e-3 (9.94e-4) ≈	3.0019e-3 (2.86e-4)
MMMOP2-B	3	7	2.6784e-1 (2.18e-2) ≈	2.6844e-1 (1.74e-2) ≈	2.6882e-1 (1.37e-2) ≈	2.7243e-1 (1.85e-2) -	2.6084e-1 (2.16e-2)
MMMOP3-A	2	2	7.2342e-3 (1.11e-3) +	9.2730e-3 (9.18e-4) -	9.0378e-3 (1.16e-3) ≈	9.0446e-3 (9.93e-4) -	8.4427e-3 (1.22e-3)
MMMOP3-B	3	7	2.5166e-1 (3.69e-2) ≈	2.5064e-1 (5.31e-2) ≈	2.5280e-1 (2.46e-2) ≈	2.5829e-1 (3.16e-2) ≈	2.4410e-1 (3.21e-2)
MMMOP3-C	2	6	8.1665e-2 (4.16e-2) +	1.8368e-1 (2.84e-2) -	1.7839e-1 (3.35e-2) -	1.7468e-1 (2.27e-2) -	1.2772e-1 (4.21e-2)
MMMOP3-D	3	7	2.0935e-1 (4.04e-2) ≈	2.2106e-1 (2.80e-2) -	2.4396e-1 (2.14e-2) -	2.6047e-1 (3.52e-2) -	2.0264e-1 (3.37e-2)
MMMOP4-A	2	2	8.7219e-4 (3.87e-6) -	8.6758e-4 (7.35e-6) ≈	8.6791e-4 (1.06e-5) ≈	8.6844e-4 (6.08e-6) ≈	8.6647e-4 (8.04e-6)
MMMOP4-B	3	7	2.3969e-2 (2.23e-4) +	2.5835e-2 (1.00e-3) ≈	2.5670e-2 (1.05e-3) ≈	2.5393e-2 (1.03e-3) ≈	2.5450e-2 (6.84e-4)
MMMOP4-C	2	6	2.2128e-3 (4.61e-4) +	2.8677e-2 (4.90e-2) -	7.3580e-2 (9.19e-2) -	7.8304e-2 (8.87e-2) -	5.2994e-3 (4.13e-3)
MMMOP4-D	3	7	3.4299e-2 (2.15e-4) +	3.7225e-2 (1.83e-3) -	3.6397e-2 (1.14e-3) -	3.6472e-2 (1.18e-3) -	3.5623e-2 (6.36e-4)
MMMOP5-A	2	2	8.6991e-4 (2.48e-6) -	8.6185e-4 (7.72e-6) +	8.6105e-4 (7.01e-6) ≈	8.6093e-4 (7.47e-6) ≈	8.6252e-4 (3.57e-6)
MMMOP5-B	3	7	2.3641e-2 (8.12e-5) +	2.4131e-2 (3.48e-4) ≈	2.4106e-2 (2.82e-4) ≈	2.4209e-2 (5.58e-4) ≈	2.4091e-2 (1.16e-4)
MMMOP5-C	2	6	1.9786e-3 (1.88e-4) +	1.7167e-2 (2.47e-2) -	6.9113e-2 (1.02e-1) -	7.9084e-2 (1.06e-1) -	3.2553e-3 (1.87e-3)
MMMOP5-D	3	7	3.4048e-2 (1.38e-4) +	3.4749e-2 (3.21e-4) ≈	3.4825e-2 (6.77e-4) ≈	3.9032e-2 (2.37e-2) ≈	3.4699e-2 (1.91e-4)
MMMOP6-A	2	2	4.0964e-3 (1.35e-4) ≈	4.1092e-3 (7.55e-5) ≈	4.1382e-3 (1.51e-4) ≈	4.1292e-3 (1.06e-4) ≈	4.0685e-3 (9.70e-5)
MMMOP6-B	3	3	4.3815e-2 (3.64e-4) ≈	4.3916e-2 (3.14e-4) -	4.3984e-2 (2.72e-4) -	4.3993e-2 (3.92e-4) -	4.3689e-2 (2.74e-4)
MMMOP6-C	2	4	1.9171e-1 (1.05e-1) -	3.8126e-2 (5.21e-3) -	4.2674e-2 (5.25e-3) -	4.9304e-2 (6.58e-3) -	3.2868e-2 (4.67e-3)
MMMOP6-D	3	5	3.1084e-1 (1.11e-1) -	9.4404e-2 (1.95e-3) +	9.6575e-2 (2.25e-3) +	9.7635e-2 (2.11e-3) +	1.4141e-1 (4.42e-2)
+ / - / ≈			9/5/6	2/9/9	1/8/11	1/11/8	

Best result in each row is bold

Table 18 Statistical results of IGDX and HV on location selection problem obtained by MMOEADPPs and other MOEAs in comparison

	DNNSGAI	HREA	MMEAWI	MMOEADC	MO_Ring_PSO_SCD	TriMOEATAR	MMOEADPPs
IGDX	1.9857e+0 (6.01e−1) −	1.0238e+0 (3.52e−3) −	1.1661e+0 (2.44e−1) −	1.0750e+0 (3.04e−1) −	1.0367e+0 (1.38e−2) −	2.1032e+0 (5.50e−1) −	7.8628e−1 (1.91e−1)
HV	9.2036e−1 (4.15e−3) −	8.6748e−1 (2.43e−2) −	9.2697e−1 (2.76e−3) ≈	9.2493e−1 (3.13e−3) −	9.1145e−1 (4.59e−3) −	9.1587e−1 (3.95e−2) −	9.2746e−1 (3.24e−3)

The best value of each row is highlighted

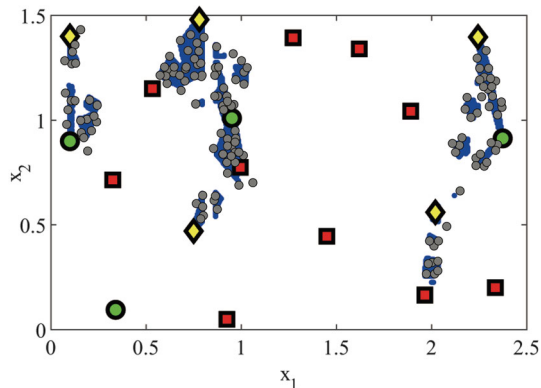


Fig. 25 Obtained solutions by MMOEADPPs on the multimodal multi-objective location selection problem. The blue regions are the Pareto optimal regions. The yellow diamond, green circle and red square, respectively, represent schools, hospitals and convenience stores

our algorithm for solving some real-world engineering problems in the future.

Acknowledgements This work was partly supported by the National Natural Science Foundation of China under Grant No. 62076225.

Data availability Comparison algorithms and benchmarks are available on Platemo [35] (<https://github.com/BIMK/PlatEMO>).

Declarations

Conflict of interest The authors declare that they have no conflicts of interest to this work. The article contains no violation of any existing copyright or other third party right.

References

- Zheng T, Liu J, Liu Y, Tan S (2022) Hybridizing multi-objective, clustering and particle swarm optimization for multimodal optimization. *Neural Comput Appl* 34:2247–2274
- Babaei Tirkolaee E, Goli A, Weber G-W (2020) Fuzzy mathematical programming and self-adaptive artificial fish swarm algorithm for just-in-time energy-aware flow shop scheduling problem with outsourcing option. *IEEE Trans Fuzzy Syst* 28(11):2772–2783
- Yue C, Qu B, Liang J (2018) A multiobjective particle swarm optimizer using ring topology for solving multimodal multi-objective problems. *IEEE Trans Evolut Computat* 22(5):805–817
- 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-05, proceedings of the nineteenth international joint conference on artificial intelligence*, Edinburgh, Scotland, UK, July 30–August 5, pp. 859–864
- Jaszkiewicz A (2002) On the performance of multiple-objective genetic local search on the 0/1 knapsack problem—a comparative experiment. *IEEE Trans Evolut Computat* 6(4):402–412
- Ishibuchi H, Yamane M, Akedo N, Nojima Y (2013) Many-objective and many-variable test problems for visual examination of multiobjective search. In: *Evolutionary Computation*
- Han S, Zhu K, Zhou M, Cai X (2021) Information-utilization-method-assisted multimodal multiobjective optimization and application to credit card fraud detection. *IEEE Trans Computat Soc Syst* 8(4):856–869
- Peng Y, Ishibuchi H (2021) A diversity-enhanced subset selection framework for multi-modal multi-objective optimization. *IEEE Trans Evolut Computat* 26(5):886–900
- Rizk-Allah AE, Rizk M, Hassanien Slowik A (2020) Multi-objective orthogonal opposition-based crow search algorithm for large-scale multi-objective optimization. *Neural Comput Appl* 32:13715–13746
- Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evolut Computat* 6(2):182–197
- Zitzler E, Laumanns M, Thiele L (2001) Spea 2: improving the strength pareto evolutionary algorithm. *TIK-Report* 103:201
- Menchaca-Mendez A, Coello CAC (2015) Gde-moea: A new moea based on the generational distance indicator and ϵ -dominance. In: *2015 IEEE congress on evolutionary computation (CEC)*, pp 947–955
- Wang H, Jiao L, Yao X (2015) Two_arch2: an improved two-archive algorithm for many-objective optimization. *IEEE Trans Evolut Computat* 19(4):524–541
- Zhang Q, Li H (2007) Moea/d: a multiobjective evolutionary algorithm based on decomposition. *IEEE Trans Evolut Computat* 11(6):712–731
- Cheng R, Jin Y, Olhofer M, Sendhoff B (2016) A reference vector guided evolutionary algorithm for many-objective optimization. *IEEE Trans Evolut Computat* 20(5):773–791
- Liang JJ, Yue CT, Qu BY (2016) Multimodal multi-objective optimization: A preliminary study. In: *2016 IEEE Congress on Evolutionary Computation (CEC)*, pp. 2454–2461
- Lin Q, Lin W, Zhu Z, Gong M, Li J, Coello CAC (2021) Multimodal multiobjective evolutionary optimization with dual clustering in decision and objective spaces. *IEEE Trans Evolut Computat* 25(1):130–144

18. Zhang K, Shen C, Yen GG, Xu Z, He J (2021) Two-stage double niched evolution strategy for multimodal multiobjective optimization. *IEEE Trans Evolut Computat* 25(4):754–768
19. Li W, Zhang T, Wang R, Ishibuchi H (2021) Weighted indicator-based evolutionary algorithm for multimodal multiobjective optimization. *IEEE Trans Evolut Computat* 25(6):1064–1078
20. Li W, Yao X, Zhang T, Wang R, Wang L (2022) Hierarchy ranking method for multimodal multi-objective optimization with local pareto fronts. *IEEE Trans Evolut Computat* 27(1):98–110
21. Liu Y, Ishibuchi H, Yen GG, Nojima Y, Masuyama N (2020) Handling imbalance between convergence and diversity in the decision space in evolutionary multimodal multiobjective optimization. *IEEE Trans Evolut Computat* 24(3):551–565
22. Kulesza A, Taskar B (2011) k-dpps: fixed-size determinantal point processes. In: proceedings of the 28th international conference on machine learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011
23. Gartrell M, Paquet U, Koenigstein N (2016) The bayesian low-rank determinantal point process mixture model. In: proceedings of the 10th ACM conference on recommender systems, pp. 349–356
24. Tremblay N, Amblard P-O, Barthelmé S (2017) Graph sampling with determinantal processes. In: 2017 25th European signal processing conference (EUSIPCO), pp. 1674–1678
25. Liu Y, Yen GG, Gong D (2019) A multimodal multiobjective evolutionary algorithm using two-archive and recombination strategies. *IEEE Trans Evolut Computat* 23(4):660–674
26. Li Z, Zou J, Yang S, Zheng J (2021) A two-archive algorithm with decomposition and fitness allocation for multi-modal multi-objective optimization. *Inf Sci* 574:413–430
27. Tanabe R, Ishibuchi H (2020) A framework to handle multimodal multiobjective optimization in decomposition-based evolutionary algorithms. *IEEE Trans Evolut Computat* 24(4):720–734
28. Wang W, Li G, Wang Y, Wu F, Zhang W, Li L (2022) Clearing-based multimodal multi-objective evolutionary optimization with layer-to-layer strategy. *Swarm Evolut Computat* 68:100976
29. Li G, Wang W, Zhang W, You W, Wu F, Tu H (2021) Handling multimodal multi-objective problems through self-organizing quantum-inspired particle swarm optimization. *Inf Sci* 577:510–540
30. Fan Q, Yan X (2021) Solving multimodal multiobjective problems through zoning search. *IEEE Trans Syst Man Cybern Syst* 51(8):4836–4847
31. Li G, Wang W, Zhang W, Wang Z, Tu H, You W (2021) Grid search based multi-population particle swarm optimization algorithm for multimodal multi-objective optimization. *Swarm Evolut Computat* 62:100843
32. Maity G, Roy SK, Verdegay JL (2020) Analyzing multimodal transportation problem and its application to artificial intelligence. *Neural Comput Appl* 32:2243–2256
33. Zhang P, Li J, Li T, Chen H (2021) A new many-objective evolutionary algorithm based on determinantal point processes. *IEEE Trans Evolut Computat* 25(2):334–345
34. Liu Y, Ishibuchi H, Yen GG, Nojima Y, Masuyama N, Han Y (2020) On the normalization in evolutionary multi-modal multi-objective optimization. In: 2020 IEEE congress on evolutionary computation (CEC), pp. 1–8
35. Tian Y, Cheng R, Zhang X, Jin Y (2017) PlatEMO: a MATLAB platform for evolutionary multi-objective optimization. *IEEE Computat Intell Magaz* 12:73–87
36. Agrawal R, Deb K, Agrawal R (2000) Simulated binary crossover for continuous search space. *Complex Syst* 9:115–148
37. Deb K, Tiwari S (2008) Omni-optimizer: a generic evolutionary algorithm for single and multi-objective optimization. *Europ J Operat Res* 185(3):1062–1087
38. 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 Computat* 13(5):1167–1189
39. Zitzler E, Thiele L (1999) Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE Trans Evolut Computat* 3(4):257–271
40. Alcalá-Fdez J, Sánchez L, García S, del Jesus MJ, Ventura S, Garrell JM, Otero J, Romero C, Bacardit J, Rivas VM, Fernández JC, Herrera F (2009) KEEL: a software tool to assess evolutionary algorithms for data mining problems. *Soft Comput* 13(3):307–318
41. Zhou Y, Xiang Y, He X (2021) Constrained multiobjective optimization: test problem construction and performance evaluations. *IEEE Trans Evolut Computat* 25(1):172–186
42. Ishibuchi H, Pang LM, Shang K (2022) Difficulties in fair performance comparison of multi-objective evolutionary algorithms [research frontier]. *IEEE Computat Intell Magaz* 17(1):86–101
43. Liang J, Lin H, Yue C, Yu K, Guo Y, Qiao K (2023) Multiobjective differential evolution with speciation for constrained multimodal multiobjective optimization. *IEEE Trans Evolut Computat* 27(4):1115–1129

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.