Hierarchy Ranking Method for Multimodal Multiobjective Optimization With Local Pareto Fronts

Wenhua Li, Xingyi Yao, Tao Zhang, Rui Wang, *Senior Member, IEEE*, and Ling Wang

***Abstract*—Multimodal multiobjective problems (MMOPs) commonly arise in real-world situations where distant solutions in decision space share a very similar objective value. Traditional multimodal multiobjective evolutionary algorithms (MMEAs) prefer to pursue multiple Pareto solutions that have the same objective values. However, a more practical situation in engineering problems is that one solution is slightly worse than another in terms of objective values, while the solutions are far away in the decision space. In other words, such problems have global and local Pareto fronts (PFs). In this study, we proposed several benchmark problems with several local PFs. Then, we proposed an evolutionary algorithm with a hierarchy ranking method (HREA) to find both the global and the local PFs based on the decision maker's preference. Regarding HREA, we proposed a local convergence quality evaluation method to better maintain diversity in the decision space. Moreover, a hierarchy ranking method was introduced to update the convergence archive. The experimental results show that HREA is competitive compared with other state-of-the-art MMEAs for solving the chosen benchmark problems.**

***Index Terms*—Diversity-preserving mechanisms, evolutionary computation, local Pareto front (PF), multimodal multiobjective optimization.**

I. Introduction

MULTIOBJECTIVE optimization problems (MOPs) [1] arise frequently in the real world where multiple objective functions are optimized simultaneously. In general, an MOP can be expressed as follows:

$$\begin{aligned} &\text{Minimize} \quad F(\mathbf{x}) = \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})\} \\ &\text{s.t.} \qquad \mathbf{x} = (x_1, x_2, \ldots, x_n) \in \Omega \end{aligned} \tag{1}$$

where Ω denotes the search space, m is the number of objectives, and $\mathbf{x}$ is a decision vector that consists of n decision variables x_i. A solution, $\mathbf{x_a}$, is considered to Pareto dominate another solution, $\mathbf{x_b}$, *iff* $\forall i = 1, 2, \ldots, m, f_i(\mathbf{x_a}) \leq f_i(\mathbf{x_b})$ and $\exists j = 1, 2, \ldots, m, f_j(\mathbf{x_a}) < f_j(\mathbf{x_b})$. Furthermore, a Pareto optimal solution is a solution that is not Pareto dominated by any other solution. The set of Pareto optimal solutions is called a Pareto set (PS). The image of the PS is known as the Pareto front (PF).

To address MOPs, many multiobjective evolutionary algorithms (MOEAs) have been proposed and have been proven effective [2]. Most MOEAs try to find an evenly distributed and widespread PF of a MOP. Therefore, the decision makers (DMs) can choose a final solution from the PF based on their preferences [3]. Generally, convergence quality is the first considered indicator for most MOEAs. However, the lack of a diversity-maintenance mechanism leads to quick premature convergence, which weakens the searching ability of MOEAs.

Multimodal multiobjective problems (MMOPs) [4] have become a popular type of problem in which multiple Pareto optimal solution sets correspond to the same point on the PF. It is worth mentioning that in some literature, MMOP refers to multimodal optimization problem [5]. Many real-world problems have shown multimodal characteristics, e.g., diet design problems [6], space mission design problems [7], rocket engine design problems [8], and functional brain imaging problems [9]. Fig. 1 shows a simple example of an MMOP

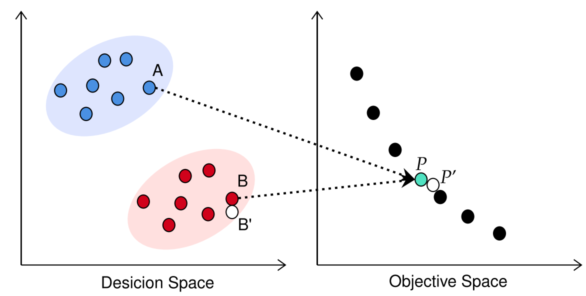


Fig. 1. Illustration of a two-objective MMOP.

Manuscript received 8 October 2021; revised 5 December 2021 and 28 January 2022; accepted 27 February 2022. Date of publication 2 March 2022; date of current version 31 January 2023. This work was supported in part by the National Science Fund for Outstanding Young Scholars under Grant 62122093; in part by the National Natural Science Foundation of China under Grant 72071205; in part by the Scientific Key Research Project of the National University of Defense Technology under Grant ZZKY-ZX-11-04; and in part by the Ji-Hua Laboratory Scientific Project under Grant X210101UZ210. *(Corresponding authors: Tao Zhang; Rui Wang.)*

Wenhua Li and Xingyi Yao are with the College of Systems Engineering, National University of Defense Technology, Changsha 410073, China (e-mail: liwenhua@nudt.edu.cn).

Tao Zhang and Rui Wang are with the College of Systems Engineering and the Hunan Key Laboratory of Multi-Energy System Intelligent Interconnection Technology, National University of Defense Technology, Changsha 410073, China (e-mail: zhangtao@nudt.edu.cn; ruiwangnudt@gmail.com).

Ling Wang is with the Department of Automation, Tsinghua University, Beijing 100084, China.

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TEVC.2022.3155757.

Digital Object Identifier 10.1109/TEVC.2022.3155757

with two objectives and two decision variables, from which we can see that point A and point B are far away in the decision space, while they share the same objective vector P. It is necessary to obtain all optimal solutions such that the DMs can better understand the problem. Moreover, it is easier to transfer to another solution if some constraints arise.

Traditional MOEAs show poor performance in solving MMOPs [10]. On the one hand, the lack of a diversity-maintenance mechanism leads to quick convergence without further exploring the whole decision space. On the other hand, different solutions that correspond to the same point in the objective space are difficult to maintain simultaneously. As shown in Fig. 1, we assume that point A has been found during the search process. Point B' corresponds to point P' in the objective space. Since the distance between P and P' is small in the objective space (in other words, they are crowded), P' is likely to be removed. Thus, it is difficult for traditional MOEAs to obtain a complete PF.

To address MMOPs, a number of multimodal MOEAs (MMEAs) have been proposed, and they have shown good performances on benchmark problems. Examples include the Omni-optimizer [11], [12], the double-niched evolutionary algorithm (DNEA) [13], decision space-based niching NSGA-II (DN-NSGA-II) [4], multiobjective particle swarm optimization (PSO) using ring topology and special crowding distance (MO_Ring_PSO_SCD) [14], the convergence-penalized density evolutionary algorithm (CPDEA) [15], and weighted indicator-based evolutionary algorithms (MMEA-WI) [16], to name a few. The above-mentioned algorithms are specially designed for solving MMOPs with several PSs and a single common PF. However, in real-world problems, a more common situation is that different PSs correspond to different PFs. In other words, two distant solutions in the decision space have similar but different objective values. Taking the path planning problems as an example, from the start to the destination, it is difficult to find two different paths that have the same length. Instead, these different paths may have very close path lengths.

In addition, in the design of a spacecraft launch mission [7], the two solutions x_i and their images $F(x_i)$ are expressed as follows:

$$\mathbf{x}_1 = (782, 1288, 1788), F(\mathbf{x}_1) = (0.462, 1001.7)$$
$$\mathbf{x}_2 = (1222, 1642, 2224), F(\mathbf{x}_2) = (0.463, 1005, 3) \quad (2)$$

where the first parameter determines the departure time from the Earth (in days after 01.01.2000), i.e., the launch time of the mission, and the two objective functions represent the portion of the vehicle's mass that does not reach the destination and the time of flight (in days), respectively. Apparently, $\mathbf{x}_1$ dominates $\mathbf{x}_2$ since $F(\mathbf{x}_1)$ is less than $F(\mathbf{x}_2)$ in both objectives. As a result, $\mathbf{x}_2$ is discarded during the evolution. However, the difference between these two solutions in the objective space is very small. Specifically, the mass fraction differs by 0.001, and the flight time differs by four days. In this situation, the launch times differ by 440 days, which is an important criterion for the DMs to accept this deterioration. Therefore, these two solutions represent two distinct launch opportunities.

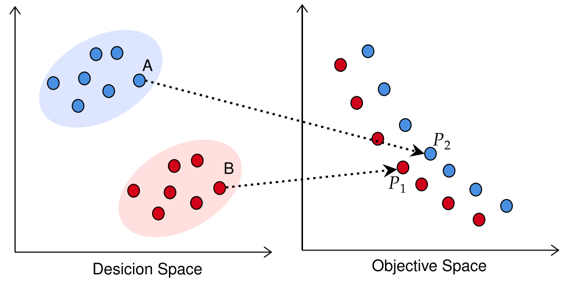


Fig. 2. Illustration of a two-objective MMOP with a local PF.

Fig. 2 illustrates an MMOP with a local PF (MMOPL), where the red points correspond to the global PF and the blue points refer to the local PF. Solutions A and B are distant in the decision space; however, they have very similar objective values. Through the above analysis, we accept it as a more general situation, which means that the traditional MMOP is a special case of MMOPL. In other words, if one algorithm can solve MMOPL well, then it is foreseeable that the algorithm can solve MMOP as well. Unfortunately, few studies attempt to solve MMOPLs. Among them, $P_{Q,\epsilon}$-MOEA [7] is designed specifically for space mission design problems, self-organizing multimodal multiobjective pigeon-inspired optimization algorithm (MMOPIO) is based on Pareto dominance [17], and NSGA-III-ADA [18] and eMOEA/D-ADA [18] are based on the decomposition method. However, since the above methods did not systematically discuss MMOPL, there is still room for performance improvement. Very recently, Lin *et al.* [19] proposed a novel MMEA with dual clustering in the decision and objective spaces (MMOEA/DC) to solve CEC 2019 MMOPs [20] with local PFs. DBSCAN is modified to maintain diversity in decision space. MMOEA/DC has proven effective in CEC 2019 MMOPs. However, since the clustering method cannot always correctly classify different PSs, the performance is not steady. Moreover, the above-mentioned algorithms cannot find the solutions that have deteriorated according to DM preferences. In other words, if DMs are agreeable to situations with local PFs that have a certain gap between the global PF, then there is no MMEA satisfying this requirement to date.

In this study, we first develop the imbalanced distance minimization benchmark problems (IDMPs) [15] test suite to construct the MMOPL benchmark problems, termed IDMP_e. To address the MMOPLs, we introduce a novel MMEA with a hierarchy ranking method, termed HREA. Compared to traditional MMEAs that focus on obtaining all equivalent global Pareto optimal set (global PSs), HREA can find global PSs and local Pareto optimal sets (local PSs) in a single algorithm run with good convergence and diversity. In the proposed algorithm, we introduce a convergence archive to improve convergence ability. For the main population, we propose the use of the local convergence indicator as the selection criterion to maintain the diversity of solutions in the decision space. As a result, solutions in the main population do not directly converge to the global and local PSs but continue to explore the whole decision space. For the convergence archive, we propose a hierarchy ranking method to select solutions that satisfy the DM's preference. Then, the number of solutions located on different PFs is balanced. In addition, we introduce a user-predefined parameter ϵ to control the quality of the obtained local PF. Specifically, when the DM sets $\epsilon = 0$, the algorithm only outputs the global PF and the equivalent global PSs. That is, HREA degenerates to normal MMEA. When the DM sets $\epsilon = 1$, the algorithm outputs all global PSs and local PSs without considering the quality of local PSs.

The remainder of this study is structured as follows. Section II gives the preliminary works of MMOPs, including the definition and related works. Section III illustrates the benchmark problems and the proposed evolutionary algorithm in detail. In addition, a set of experiments is conducted in Section IV to further discuss the performance of HREA. Finally, we conclude this work in Section V and briefly discuss directions for future work.

II. Preliminary Work

A. Definition of MMOP With Local Pareto Fronts

MMOPs have been widely studied and have several definitions according to different works. We adopt the definitions proposed in [20], which are explained as follows.

Definition 1: For an arbitrary solution, $\mathbf{x}$, in a solution set, $\mathbf{S}_L$, if there is no neighborhood solution $\mathbf{y}$ satisfying $\|\mathbf{x} - \mathbf{y}\| \leq \delta$ (δ is a small positive value), dominating any solution in the set $\mathbf{S}_L$, then $\mathbf{S}_L$ is called the local PS.

Definition 2: For an arbitrary solution in a solution set $\mathbf{S}_G$, if there is no solution dominating any solution in the set $\mathbf{S}_G$, then $\mathbf{S}_G$ is called the global PS.

Definition 3: The set of all the vectors in the objective space that corresponds to the local PS is defined as the local PF.

Definition 4: The set of all the vectors in the objective space that corresponds to the global PS is defined as the global PF.

According to the definitions, for a normal MMOP, there is only one single global PF that corresponds to a single PF or multiple global PSs, which is illustrated in Fig. 1. For an MMOPL, there are local PFs and a global PF, which is shown in Fig. 2.

B. Related Studies

Fig. 1 shows the general situation of MMOPs studied by most existing studies, where two different global PSs share the same global PF. Recently, a remarkable review study was conducted by Tanabe and Ishibuchi [10]. As mentioned in [10], during the past decades, many studies have been conducted to solve MMOPs. Among them, the Omni-optimizer [11], [12] is probably the most representative one, which is extended from NSGA-II [21]. To maintain the diversity of solutions in the decision space, the Omni-optimizer introduced several strategies, including the Latin hypercube sampling-based population, restricted mating selection and alternative crowding distance. The alternative crowding distance aims to evaluate solution diversity in both the objective and decision spaces. Sharing the same ideas, the niching-covariance matrix adaptation (CMA) approach [22] introduced an aggregate distance metric in the objective and decision spaces to divide solutions into multiple niches. The DNEA [13] and DN-NSGA-II [4] are two enhanced algorithms based on the Omni-optimizer, which have been proven effective on min-max fairness (MMF) [4] test problems. Recently, Liu *et al.* [15] proposed a new MMEA based on convergence-penalized density method (CPDEA) that can handle MMOPs that have imbalance difficulties in searching different PSs. Motivated by this idea, MMEA-WI was proposed [16] with a weighted indicator to obtain all PSs and perform better than other state-of-the-art MMEAs on the selecting problems.

Unlike the above MMEAs, the PSO variation operator is utilized in MO_PSO_MM [23] and MO_Ring_PSO_SCD [14] to generate new solutions and has achieved good performance on traditional MMOPs. The differential evolution (DE) operator is also utilized to promote diversity in the decision space [24], e.g., local binary pattern-based adaptive DE (LBPADE) [25], automatic niching DE with contour prediction approach (ANDE) [26], and dual-strategy DE with affinity propagation clustering (DSDE) [5]. Furthermore, decomposition-based MOEAs usually outperform Pareto dominance-based algorithms on MaOPs. It is reasonable that advanced MMEAs can be developed by using decomposition methods, e.g., MOEA/D-AD [27]. Specifically, MOEA/D-AD assigns one or more individuals to each subproblem to handle multimodality. Such strategies can also be found in [28].

Although there are many MMEAs, few works focus on MMOPLs. Most of the abovementioned MMEAs only focus on obtaining the global PF and global PSs. $P_{Q,\epsilon}$-MOEA [7] uses the ϵ-dominance relationship to find solutions that have acceptable quality. This approach adopted an unbounded archive to store solutions found up to this point. In this way, the local PSs, as well as the solutions located around the global PS, can be maintained. DNEA-L [29] is developed based on DNEA [13] by introducing a multifront archive update method. The experimental results show that DNEA-L can find local PSs on polygon-based problems. Several efforts by Tanabe and Ishibuchi resulted in proposing NSGA-III-ADA [18] and eMOEA/D-ADA [18]. Very recently, Lin *et al.* [19] proposed a novel MMEA with dual clustering in the decision and objective spaces to focus on MMOPLs. The experiments show that the proposed algorithms can find local PFs with acceptable quality. In addition, a clearing-based evolutionary algorithm with a layer-to-layer evolution strategy (CEA-LES) [30] is proposed to solve MMOPs with local PFs. In CEA-LES, a layer-to-layer evolutionary strategy is proposed to develop equivalent global PSs and local PSs, which shows good performance on benchmark problems.

C. Proposed MMOPL Test Suite

Since MMOPLs have not been systematically studied, there are few benchmark problems. In the 2019 IEEE Congress on Evolutionary Computation, Liang *et al.* [20] proposed several MMOPs. Among these MMOPs, there are several MMOPLs, e.g., MMF11, MMF12, MMF13, and MMF15. Fig. 3 presents the PSs and PFs of MMF11, MMF12 and MMF13. For these problems, there is a single global PS and a single local PS. Generally, the difficulty level of these problems is

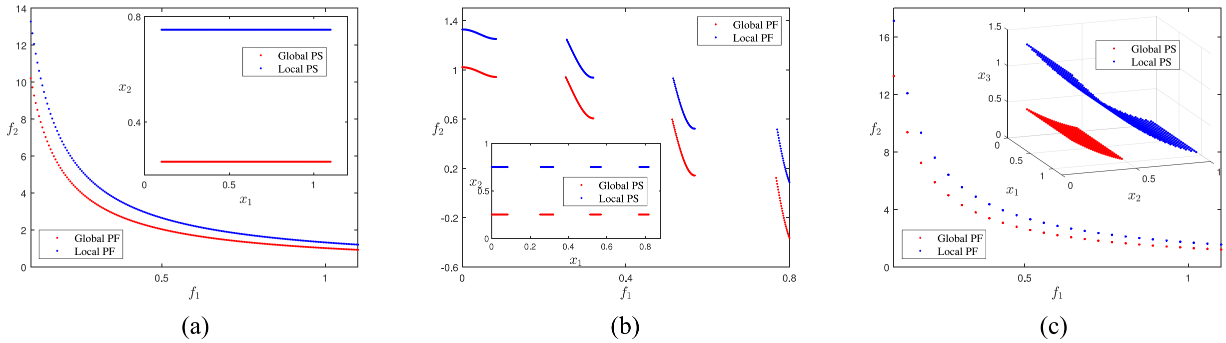


Fig. 3. PSs and PFs of (a) MMF11, (b) MMF12, and (c) MMF13.

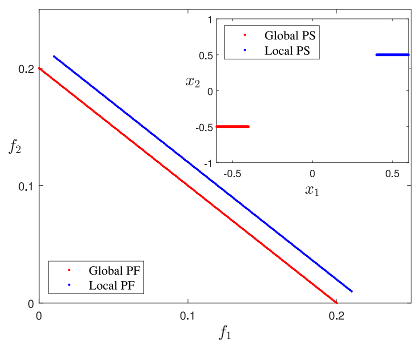


Fig. 4. PFs and PSs of IDMPM2T1_e, where $\epsilon = 0.01$ and $\alpha = 3$.

easy. In addition, these problems cannot be extended to high dimensions.

Liu *et al.* [15] proposed the multimodal multiobjective IDMPs. The main characteristic of IDMPs is that the computational resources for solving different PSs are different. That is, the number of fitness evaluations to obtain different PSs is different. For more details, the readers can refer to [15]. Notably, the IDMP test suite is scalable and can be easily extended to high dimensions. The general objective functions of the two-objective IDMPs can be expressed as follows:

$$\begin{aligned} f_1(\boldsymbol{x}) &= \min \ (|x_1 + 0.6| + g_1(\boldsymbol{x}), |x_1 - 0.4| + g_2(\boldsymbol{x})) \\ f_2(\boldsymbol{x}) &= \min \ (|x_1 + 0.4| + g_1(\boldsymbol{x}), |x_1 - 0.6| + g_2(\boldsymbol{x})) \\ &\text{s.t.} \ \ x_1 \in [-1, 1], x_2 \in [-1, 1] \end{aligned} \quad (3)$$

where $g_1(\boldsymbol{x}) \geq 0$ and $g_2(\boldsymbol{x}) \geq 0$ are difficulty functions corresponding to the first and second equivalent PSs, respectively. To reasonably construct the MMOPLs, the easiest way is to directly add a small positive noise to the objective function. In the problem of Type1, the modified function can be expressed as follows:

$$\begin{aligned} g_1(\boldsymbol{x}) &= |x_2 + 0.5| \\ g_2(\boldsymbol{x}) &= \alpha|x_2 - 0.5| + \epsilon \\ \alpha &\geq 1 \end{aligned} \quad (4)$$

where ϵ is a very small positive number, which is used to control the gap between the global and local PFs. After the modification, the local PS is $x_2 = 0.5, x_1 \in [0.4, 0.6]$, which is shown in Fig. 4. It is worth mentioning that if ϵ is added to $g_1(\boldsymbol{x})$, then the global PS is difficult to find, and the local PS is easy. As a reasonable approach, we would like to increase the difficulty in searching for local PSs to better illustrate algorithm performance.

We can introduce more multimodal single objective functions to IDMP to construct MMOPLs test problems. The readers can refer to the supplementary material for more details. The new constructed MMOPL test suite based on IDMP is termed IDMP_e (imbalanced distance minimization benchmark problems with epsilon-efficient solutions).

III. Proposed HREA Method

A. Motivation

Traditional MOEAs and MMEAs focus more on obtaining the global PSs but rarely consider finding local PSs. Thus, they are not suitable to solve MMOPLs. The main challenge of maintaining both global PSs and local PSs arises from the imbalance between convergence and diversity in the decision space. Specifically, when we perform an MMEA to deal with the MMOPL, the population is very likely to quickly converge to the global PSs because of Pareto dominance. Most MMEAs try to abandon the local PSs since local PSs may facilitate the algorithm being trapped in the local optimal regions.

To address this issue, the very recent algorithm MMOEA/DC [19] introduced a dual clustering method in the decision and objective spaces to maintain the global and local PSs. In each generation, the algorithm first implements a clustering method (NCM) in the decision space to recognize the separate PSs. Then, in each cluster, an environmental selection method is used to select better solutions. Second, another clustering method (HCM) is utilized in the objective space to maintain PS diversity. Through the dual clustering in the decision space and objective space, MMOEA/DC can properly maintain both global PSs and good local PSs with acceptable quality during the evolutionary process.

Inspired by MMOEA/DC, in this study, we proposed a novel MMEA based on a hierarchy ranking method, termed HREA, to obtain both global PSs and local PSs that satisfy user preferences. To summarize, the main contributions that distinguish this work from existing studies are presented as follows:

1) Based on IDMP test suites, we propose IDMP_e benchmark problems that contain several global and local PSs. In contrast to the existing MMOPLs, IDMP_e problems

Algorithm 1 General Framework of HREA

Input: Maximum generations *MaxGen*, population size N, acceptable local PF gap ϵ

Output: Convergence archive *Arc*

1: $Pop \leftarrow Initialization(N)$

2: $PopCD \leftarrow MPFEnvSel(Pop, N)$ /* Calculating the crowding distance of Population for mating selection. */

3: $[Arc, ArcCD] \leftarrow ArcUpdate(Pop, N, \epsilon)$ /* Calculating the crowding distance of Archive for mating selection. */

4: **while** $gen \leq MaxGen$ **do**

5:   **if** $gen >= 0.5 * MaxGen$ *and* $rand > P$ **then**

6:     $MPool \leftarrow TournamentSel(Arc, ArcCD, N)$

7:   **else**

8:     $MPool \leftarrow TournamentSel(Pop, PopCD, N)$

9:   **end if**

10:   $Off \leftarrow Variation(MPool)$

11:   $[Pop, PopCD] \leftarrow EnvSel(Pop, Off, N)$

12:   $[Arc, ArcCD] \leftarrow ArcUpdate(Arc, Off, N, \epsilon)$ /* Introducing parameter ϵ to control the quality of the obtained local PF. */

13: **end while**

have more than one different PS that corresponds to the same global and local PFs.

2) To better address MMOPLs, HREA is proposed. Specifically, for the main population in HREA, we propose the use of the local convergence quality to maintain all global PSs and local PSs. For the convergence archive, a hierarchy ranking method is applied to improve the convergence performance and control the quality of local PFs. In addition, a crowding balance method is performed after the selection operation. Then, HREA can well balance the diversity and convergence of the obtained solutions.
3) A user-predefined parameter ϵ is used in HREA to control the quality of the obtained local PSs. The parameter ϵ is specifically used during the updating process of the convergence archive. Specifically, when $\epsilon = 0$, HREA degenerates into a normal MMEA that focuses on global PSs only; when $\epsilon = 1$, HREA outputs all global and local PSs without considering solution quality.

B. General Framework of HREA

The algorithm framework of HREA is described in Algorithm 1. Similar to most MOEAs, HREA consists of the following parts: population initialization, mating selection, offspring generation, and environmental selection. Notably, we introduce the convergence archive to improve the convergence quality as well as to control the quality of the obtained local PFs. In each generation, we select parents from both the main population and the convergence archive to perform co-evolution; see lines 5-7. Such a co-evolving mechanism is not new in MOEAs while addressing MOPs; e.g., archives are used in the coevolutionary-constrained multiobjective optimization algorithm (CCMO) [31] and dual-population-based evolutionary algorithm for constrained multiobjective optimization (c-DPEA) [32] to balance constraints and exploration quality, and archives are used in CPDEA [15] and MMEA-WI [16] to balance diversity and convergence.

There are roughly two stages; see line 5. In the first 50% of evolutions, we only select parents from the main population. In the second stage, we randomly choose parents from the main population or archive according to parameter p to improve the convergence ability of the algorithm. To maintain the diversity of solutions in the decision space, we introduce an environmental selection strategy that utilizes the local convergence (line 11 in Algorithm 1), which is further illustrated in the following section. Then, a hierarchy ranking method is proposed to update the archive (line 12).

C. Environmental Selection

The main task for MMEAs is to maintain the diversity of the solutions in the decision space as well as the convergence in the objective space. Then, borrowing the idea from CPDEA [15], we propose using the local convergence quality to balance the conflict between diversity and convergence. Specifically, for an ordinary convergence quality, optimal solutions should not be dominated by any other solutions in the population. However, the local convergence quality is calculated only within neighboring solutions. As shown in Fig. 2, when calculating the local convergence quality of solution A, we first find its neighbors, set N_A. Then, if solution A is not dominated by any solutions in N_A, then it is considered a local Pareto optimal solution in the current stage. In this way, solutions in the local PSs are not directly compared to the solutions in the global PSs. Thus, the local PSs can be reserved. The local convergence quality of solution $\mathbf{x}_i$ can be calculated by the following:

$$c(i) = \frac{\sum_{j=1}^{n_i} B_{i,j}}{n_i} \tag{5}$$

where $B_{i,j}$ is 1 if $\mathbf{x}_i$ is dominated by $\mathbf{x}_j$ and 0 otherwise, and n_i is the number of the neighbors of solution $\mathbf{x}_i$.

One solution $\mathbf{x}_n$ is called a neighbor of the solution $\mathbf{x}_i$ if the distance of $\mathbf{x}_i$ and $\mathbf{x}_n$ in the decision space is smaller than V, which is calculated as follows:

$$V = \eta \left(\prod_{m=1}^{M} \left(x_m^{\max} - x_m^{\min} \right) \right)^{\frac{1}{M}} \tag{6}$$

where M is the number of the decision variables, and $x_m^{\max}$ and $x_m^{\min}$ are the maximum and minimum values of the mth decision variable in the current evolution process, respectively. The second part of V is used to evaluate the average distance of the decision space. η is an empirical parameter that is used to control the average length of neighbors, which is set to 0.2 according to previous research.

Fig. 5 illustrates the calculation method of the local convergence quality. As we can see, there is only one local PS, which is represented by the blue points. First, we need to calculate the value of V for the current evolution process, which is represented by the green circle. For p_5, its neighbors include p_1, p_2, and p_3. Therefore, $n_i = 3$. As we can see from Fig. 5, p_5 is dominated by p_2, which means that $B_{5,j} = [0, 1, 0]$. As a result, the local convergence quality of p_5 is $c(p_5) = (0 + 1 + 0)/3 = 0.33$. In the same way, we can

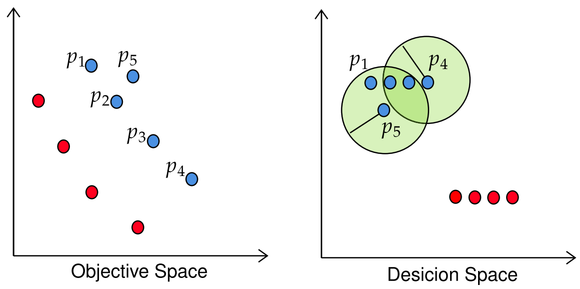


Fig. 5. Illustration of the local convergence quality, where the local PS and PF are represented by blue points, and the green circles indicate the value of V in this example.

Algorithm 2 Environmental Selection

Input: Population *Pop*, offspring *Off*, population size N

Output: Population *Pop*, crowding distance *PopCD*

1: $jointP \leftarrow Pop \cup Off$
2: $LocalC \leftarrow CalLocalConvergence(jointP)$
3: $PopCD \leftarrow CalCrowdDis(jointP)$
4: $jointP \leftarrow SortByRows(LocalC, PopCD)$ /* Use the local convergence quality and crowding distance to sort the joint population */
5: $Pop \leftarrow jointP(1 : N)$ /* The first N solutions form the new population */
6: $PopCD \leftarrow CalCrowdDis(Pop)$

calculate the local convergence quality of p_4, which is 0 in this case. As a result, p_4 is more likely to be selected during the evolution. Moreover, the whole calculation process occurs over a small region, which means that the global PS does not affect the selection of the local PS. As a result, all the local PSs can be reserved.

The process of the proposed environmental selection is illustrated in Algorithm 2. We first calculate the local convergence quality of all solutions as well as the solutions crowding distance (see lines 2 and 3). Then, these two criteria are utilized to sort the population (see Line 4). Finally, the first N solutions are selected as the new population. Specifically, the crowding distance of solution $\mathbf{x}_i$ can be expressed as follows [19]:

$$\text{CrowdDis}_i = \frac{N-1}{\sum_{j=1}^{N} 1/\|\mathbf{x}_j - \mathbf{x}_i\|} \tag{7}$$

where $\|\mathbf{x}_j - \mathbf{x}_i\|$ indicates calculating the Euclidean distance of solutions $\mathbf{x}_i$ and $\mathbf{x}_j$. Note that $\mathbf{x}_i$ and $\mathbf{x}_j$ have been previously normalized. As reported in [33], normalization may substantially affect MMEA. The detailed normalization method used in this study is the same as that in [19].

The utilization of the local convergence quality can well maintain the diversity of the solutions in the decision space. That is, during evolution, the population in the decision space converges to the local Pareto optimal and form several regions. Notably, if the minimum distance between two local Pareto optimal regions is less than the threshold value V, then these two regions blend into one region.

D. Archive Update

We propose an environmental selection strategy that utilizes the local convergence quality to maintain all the local optimal

Algorithm 3 Updating the Archive

Input: Archive *Arc*, offspring *Off*, population size N, acceptable local PF gap ϵ

Output: Updated archive *Arc*, archive crowding distance *ArcCD*

$JointArc \leftarrow Arc \cup Off$
$[FNo, MaxF] \leftarrow NDSort(JointArc)$ /* Obtain the Pareto rank of the JointArc and the max Pareto rank. */
$GlobalPF \leftarrow JointArc(FNo == 1)$ /* Obtain the global PF. */
$PickedPop \leftarrow JointArc(FNo == 1)$
$RemainPop \leftarrow JointArc(FNo \sim= 1)$ /* *RemainPop* contains solutions that have not been checked. */
while $isnotempty(RemainPop)$ *or* $MaxF == 1$ **do**
  $RemainPop \leftarrow DelPop(ReaminPop, PickedPop)$ /* Delete the extreme crowded solutions to avoid premature convergence. */
  $[FNo_1, MaxF] \leftarrow NDSort(RemainPop)$
  $NextPop \leftarrow RemainPop(FNo_1 == 1)$
  $[FNo, MaxF] \leftarrow NDSort([NextPop*(1-\epsilon), GlobalPF])$ /* Improving the quality of *NextPop* by ϵ to perform Nondominated Sort with the global PF, where *FNo* and *MaxF* are the Pareto rank and max rank of *NextPop*. */
  $PickedPop \leftarrow PickedPop \cup NextPop$
  $RemainPop \leftarrow RemainPop(FNo_1 \sim= 1)$
end while
$Arc \leftarrow BalanceEachPFNum(PickedPop, N)$
$ArcCD \leftarrow CalArcCrowdDis(Arc)$

regions. However, since we mainly focus on the diversity of solutions in the environmental selection, the convergence quality of the obtained solutions is unsatisfactory. In addition, all the local PFs are reversed, and even the objective values are severely worse than the global PF.

To address the above issues, we introduce a convergence archive to improve the convergence quality. Moreover, a user-defined parameter ϵ is introduced to control the quality of the local PFs. Specifically, we have the following definition.

Definition 5: Let us assume that $\mathbf{S}_G$ and $\mathbf{x}_L$ comprise the solution set of the global PF and a solution of the local PF. The objective vector of $\mathbf{x}_L$ is $F(\mathbf{x}_L)$. Then, we call $\mathbf{x}_L$ an ϵ-acceptable solution if vector $F(\mathbf{x}_L)*(1-\epsilon)$ is not dominated by any other solutions in $\mathbf{S}_G$.

Note that the ϵ-dominance relationship and ϵ-approximate PF proposed in [34] and [35] are slightly different from the ϵ-acceptable solution. Specifically, an ϵ-acceptable solution must be in an ϵ-approximate PF; however, solutions in an ϵ-approximate PF may not be an ϵ-acceptable solution. The main difference is that an ϵ-acceptable solution must be a solution in the local PF. For DMs, it is easy for them to embed their preference into the algorithm by a simple calculation. If the DM has no prior information about the problem, then we suggest setting $\epsilon = 0.3$ to balance the quality of the local PF (if it exists) and searching efficiency.

1) Hierarchy Ranking Method: The updating process of the convergence archive is illustrated in Algorithm 3. The main task is to maintain two populations, namely, *PickedPop* and *RemainPop*. First, we select nondominated solutions to form

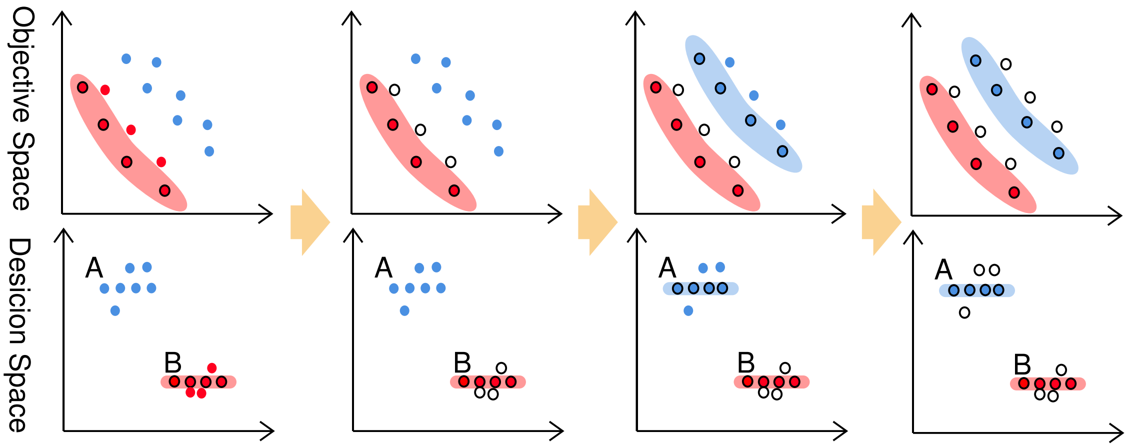


Fig. 6. Archive update process. Points with black edges and without edges represent the solutions that have been checked and not been checked, respectively. Points without face color refer to the solutions that have been deleted.

the global PF (lines 2 and 3). Then, we delete the solutions in *RemainPop* that are close to any solution in *PickedPop* (line 7). The reason for this is that if one solution is close to any solution in *PickedPop* in the decision space, then they are likely to be close in the objective space. In this situation, the hierarchy ranking method regards this solution as the next PF layer. While *RemainPop* is not empty, we first pick up the PF of *RemainPop*, called *NextPop*. Then, if there is a solution in *NextPop* that is not acceptable ($MaxF > 1$), then we terminate the loop. Otherwise, if all solutions in *NextPop* are acceptable, then we add *NextPop* into *PickedPop* and delete them in *RemainPop* (see lines 11 and 12 in Algorithm 3).

The selection process of solutions is illustrated in Fig. 6, which is called the hierarchy ranking method (HR). As we can see, HREA 1) first selects the global PF. Then, 2) the close solutions to the selected PF are deleted. Finally, 3) the next PF layer is selected. Then, steps 2 and 3 are repeated until there is no other PF.

2) Balancing the Solution Numbers of Each PF: To obtain a well-distributed PF, after the selection operation, we need to balance the numbers of solutions for each PF. There are roughly two stages. First, we need to allocate the number of solutions N_{PF}^{i}, where $i = 1, 2, \ldots, n$ is the index of the PF, which can be calculated by the span of the PFs or uniformly separated. Then, if the number of solutions in the ith PF is larger than N_{PF}^{i}, then a second selection strategy is introduced to maintain the solution number. The specific process is as follows: 1) first, the solution crowding distance is calculated by (7) and 2) second, the solution with the minimum solution crowding distance is deleted. The above steps are repeatedly executed until the number of solutions of the ith PF is N_{PF}^{i}. Then, the distribution and uniformity of the obtained PFs can be maintained.

IV. Experiment

A. Experimental Setting

1) Benchmark Problems: To benchmark the performance of HREA, the proposed test problems in IDMP_e are used. Moreover, some benchmarks in [20] are also used since they have local PFs.

Moreover, as the proposed HREA is also expected to solve traditional MMOPs, the MMF problems and IDMPs [36] are taken as test problems. The MMF problems are designed by mirroring the original Pareto optimal solution to form multiple equivalent subsets. The IDMPs are used to test the algorithm performance for problems of varying levels of difficulty in finding different PSs. The parameters for each benchmark problem are set with the suggested values according to their original paper. In addition, the true PF and PS reference points are provided by the original papers. For IDMP_e, these data can be found in the supplementary material and https://github.com/Wenhua-Li/HREA.git.

2) Performance Metrics: A number of performance metrics have been proposed to examine the performance of MOEAs, for example, hypervolume, GD, and IGD indicators [37], [38]. Most of them measure the performance of solutions in the objective space. Since MMEAs aim to obtain all PSs, the performance of solutions in the decision space should also be examined. Thus, the IGDX [39] can be adopted, which examines how well the obtained solution set approximates the true PSs. Although there are some combined performance metrics, e.g., the PSP [14] and inverted generational distance–multimodal (IGDM) [36], which evaluate the performance of solutions in both the objective space and decision space simultaneously, they are dependent on some user-defined parameters.

Overall, we select IGD and IGDX to evaluate the quality of the obtained solutions. IGDX measures the convergence and diversity of the obtained solutions in the decision space, while IGD measures the performance in the objective space. For a solution set, $\mathbf{X}$, the above two metrics are calculated as follows:

$$\mathrm{IGD}(\mathbf{X}) = \frac{1}{|\mathbf{X}^*|}\sum_{\mathbf{y}\in\mathbf{X}^*}\min_{\mathbf{x}\in\mathbf{X}}\{\mathrm{ED}(F(\mathbf{x}), F(\mathbf{y}))\} \tag{8}$$

$$\mathrm{IGDX}(\mathbf{X}) = \frac{1}{|\mathbf{X}^*|}\sum_{\mathbf{y}\in\mathbf{X}^*}\min_{\mathbf{x}\in\mathbf{X}}\{ED(\mathbf{x}, \mathbf{y})\} \tag{9}$$

TABLE I
IGDX RESULTS OF THE COMPARED ALGORITHMS ON MMOPS, WHERE THE BEST MEAN FOR EACH TEST INSTANCE IS HIGHLIGHTED

Problems	TriMOEA TAR	CPDEA	DN-NSGA-II	Omni_Opt	MO_Ring _PSO_SCD	MO_PSO _MM	MMOEA /DC	HREA
IDMPM2T1	5.79E-01	9.16E-04	2.68E-01	3.92E-01	4.89E-02	4.95E-03	**8.61E-04**	6.17E-03
IDMPM2T2	3.91E-01	**9.07E-04**	2.34E-01	1.69E-01	6.49E-03	6.05E-03	1.06E-03	9.81E-04
IDMPM2T3	1.46E-01	3.36E-03	1.65E-01	1.00E-01	3.44E-03	4.11E-03	**1.88E-03**	4.41E-03
IDMPM2T4	6.00E-01	**8.85E-04**	5.50E-01	5.78E-01	2.05E-01	3.85E-03	1.29E-01	5.13E-03
IDMPM3T1	7.23E-01	7.27E-03	2.92E-01	4.05E-01	1.21E-01	8.02E-02	8.35E-03	**6.82E-03**
IDMPM3T2	6.71E-01	7.21E-03	5.89E-01	6.33E-01	8.71E-02	8.10E-02	8.19E-03	**7.01E-03**
IDMPM3T3	3.15E-01	1.10E-02	4.24E-01	4.79E-01	1.92E-02	1.86E-02	**1.02E-02**	1.11E-02
IDMPM3T4	8.60E-01	**4.22E-02**	7.73E-01	8.50E-01	3.54E-01	1.96E-01	1.29E-02	9.75E-03
IDMPM4T1	1.19E+00	7.37E-01	7.77E-01	7.40E-01	8.21E-01	5.90E-01	4.38E-02	**3.16E-02**
IDMPM4T2	1.03E+00	6.19E-01	9.20E-01	9.62E-01	5.81E-01	6.23E-01	5.32E-01	**1.35E-02**
IDMPM4T3	8.18E-01	**3.90E-01**	7.23E-01	7.63E-01	9.59E-02	1.38E-01	4.63E-01	**2.21E-02**
IDMPM4T4	1.11E+00	7.14E-01	1.04E+00	1.03E+00	6.44E-01	5.61E-01	5.14E-01	**4.27E-02**
MMF1	6.27E-02	**3.32E-02**	9.06E-02	9.62E-02	4.91E-02	3.97E-02	4.49E-02	3.36E-02
MMF2	7.25E-02	1.54E-02	1.29E-01	1.21E-01	3.75E-02	2.71E-02	**1.20E-02**	2.35E-02
MMF3	7.30E-02	3.53E-02	1.07E-01	1.07E-01	4.71E-02	4.42E-02	**3.05E-02**	4.10E-02
MMF4	1.05E-01	1.90E-02	8.52E-02	8.01E-02	2.74E-02	2.33E-02	2.36E-02	1.83E-02
MMF5	9.73E-02	6.07E-02	1.74E-01	1.74E-01	8.46E-02	7.36E-02	7.73E-02	**5.84E-02**
MMF6	8.25E-02	5.49E-02	1.42E-01	1.41E-01	7.23E-02	6.38E-02	6.80E-02	**5.37E-02**
MMF7	3.45E-02	**1.89E-02**	5.27E-02	4.76E-02	2.59E-02	2.08E-02	2.73E-02	1.95E-02
MMF8	4.05E-01	4.13E-02	2.86E-01	3.07E-01	6.77E-02	5.35E-02	5.33E-02	**4.06E-02**
+/-/=	0/18/2	6/10/4	0/20/0	0/20/0	0/19/1	0/15/5	5/10/5	

where ED($\mathbf{x}, \mathbf{y}$) is the Euclidean distance between $\mathbf{x}$ and $\mathbf{y}$. $\mathbf{X}$ and $\mathbf{X}^*$ denote the obtained solution set and a set of a finite number of Pareto optimal solutions uniformly sampled from the true PS, respectively.

Notably, a small IGD value means that the solution set $\mathbf{X}$ has both reasonable convergence and diversity in the objective space, which indicates that the MMEA is an effective MOEA. An MMEA is recognized as a high-performance MMEA if the obtained $\mathbf{X}$ has a small IGDX value. Furthermore, a solution set with a satisfactory IGDX usually has an acceptable IGD, while a reasonable IGD does not naturally produce a reasonable IGDX [40].

3) Competitor Algorithms: To verify the effectiveness of HREA in solving MMOPs, TriMOEA-TA&R [36], CPDEA [15], DN-NSGA-II [4], the Omni-optimizer [11], [12], MO_Ring_PSO_SCD [14], MO_PSO_MM [23], and MMOEA/DC [19] are chosen as competitor algorithms. Specifically, MMOEA/DC is the latest MMEA that focuses on MMOPLs, and other selected algorithms are representative MMEAs. For all the algorithms, we set the population size $N = 100 * D$, and the maximum number of function evaluations N_E is set to $5000 * D$ [20], where D is the number of decision variables. Since the population size for the Omni-optimizer must be a multiple of four, we set N slightly larger than its value in the other algorithms to satisfy this constraint. The simulated binary crossover (SBX) and polynomial mutation (PM) operators are employed to generate offspring except for MO_Ring_PSO_SCD.

Note that the specific parameters in each algorithm are set according to the original papers. For MO_Ring_PSO_SCD, $C_1 = C_2 = 2.05$ and $W = 0.7298$. For TriMOEA-TA&R, p_{con}, σ_{niche}, and ϵ_{peak} are set to 0, 0.5, and 0.01, respectively. For HREA, p is set to 0.5. Furthermore, we introduce ϵ to control the quality of the local PFs. In this work, we set $\epsilon = 0.3$ for all experiments except those with a special explanation. All experiments are implemented based on the PlatEMO v1.6 [41] on a PC configured with an Intel i9-9900X @ 3.50 GHz and 64-GB RAM. For the convenience of subsequent researchers, the source code of HREA is open access.[1]

B. Results for MMOPs

This section shows the performance of HREA and the competitor algorithms on IDMP and MMF test suites. Specifically, the IDMP test problems are MMOPs where imbalance exists in finding different PSs, which are more representative in some ways. The results are reported in Table I, where the mean of the IGDX values over 21 independent runs is listed. Note that due to paper length limitations, the variance in IGDX is provided in the supplementary material. We use the Wilcoxon rank-sum test with $p < 0.05$ to compare HREA with each of the competitor algorithms. In the last column of each table, the symbols "+" and "−" indicate the number of test problems where the compared algorithm shows significantly better or worse performance, respectively, compared to HREA. In addition, the symbol "=" indicates the number of test problems where there is no significant difference between HREA and the compared algorithms.

Table I shows the IGDX comparison results from which we can observe that HREA shows better performance than other state-of-the-art algorithms on the chosen test problems. Specifically, HREA wins ten instances over 20 test problems. From the table, HREA, MMOEA/DC and CPDEA are shown to be the best three algorithms for solving the chosen problems. Notably, although the DN-NSGA-II and Omni-optimizer are designed for MMOPs, it seems that they show poor performance on the chosen benchmark problems since they do not win on any test instance. DN-NSGA-II and the Omni-optimizer are two impressive works in the early research of MMOPs. Since the IDMP test suites have imbalance

[1]The source code of HREA can be found at https://github.com/Wenhua-Li/HREA.git.

TABLE II
AVERAGE VALUES OF IGD RESULTS OF THE COMPARED ALGORITHMS ON MMOPs

Problems	**TriMOEA TAR**	**CPDEA**	**DN-NSGA-II**	**Omni_Opt**	**MO_Ring _PSO_SCD**	**MO_PSO _MM**	**MMOEA /DC**	**HREA**
IDMPM2T1	6.30E-04	7.15E-04	6.32E-04	6.64E-04	3.11E-03	2.51E-03	9.32E-04	**5.85E-04**
IDMPM2T2	6.44E-04	5.16E-04	5.67E-04	5.55E-04	1.51E-03	1.41E-03	6.02E-04	**5.43E-04**
IDMPM2T3	6.46E-04	6.57E-04	5.90E-04	5.70E-04	1.69E-03	1.61E-03	8.41E-04	**5.35E-04**
IDMPM2T4	6.40E-04	6.35E-04	1.56E-03	1.05E-03	3.13E-03	1.81E-03	6.66E-04	**5.02E-04**
IDMPM3T1	9.72E-03	**4.71E-03**	6.82E-03	6.82E-03	1.30E-02	1.20E-02	6.46E-03	4.90E-03
IDMPM3T2	9.80E-03	**4.35E-03**	7.55E-03	6.68E-03	1.07E-02	9.62E-03	5.61E-03	4.60E-03
IDMPM3T3	8.99E-03	**4.54E-03**	6.13E-03	6.18E-03	1.11E-02	9.64E-03	6.54E-03	4.59E-03
IDMPM3T4	9.84E-03	**4.47E-03**	1.69E-02	1.11E-02	2.03E-02	1.39E-02	5.86E-03	4.56E-03
IDMPM4T1	2.03E-02	**6.43E-03**	1.98E-02	1.63E-02	3.39E-02	3.35E-02	1.83E-02	8.18E-03
IDMPM4T2	2.07E-02	**5.68E-03**	2.48E-02	2.16E-02	3.11E-02	2.38E-02	1.67E-02	6.31E-03
IDMPM4T3	1.92E-02	**5.74E-03**	1.46E-02	1.21E-02	3.19E-02	3.09E-02	1.64E-02	6.32E-03
IDMPM4T4	2.10E-02	**5.74E-03**	9.41E-02	6.17E-02	6.55E-02	3.12E-02	1.84E-02	6.80E-03
MMF1	4.26E-03	**2.47E-03**	4.36E-03	3.68E-03	3.72E-03	2.67E-03	3.79E-03	3.44E-03
MMF2	3.19E-02	7.26E-03	2.91E-02	2.46E-02	1.98E-02	1.45E-02	1.02E-02	**5.82E-03**
MMF3	2.56E-02	7.32E-03	2.04E-02	2.44E-02	1.68E-02	1.36E-02	9.49E-03	**5.86E-03**
MMF4	3.80E-02	2.49E-03	3.22E-03	2.83E-03	3.58E-03	2.68E-03	3.39E-03	**2.45E-03**
MMF5	3.61E-03	**2.44E-03**	3.85E-03	3.22E-03	3.69E-03	2.73E-03	3.82E-03	3.56E-03
MMF6	3.52E-03	**2.40E-03**	3.76E-03	3.21E-03	3.49E-03	2.60E-03	3.81E-03	3.46E-03
MMF7	3.71E-03	**2.52E-03**	3.82E-03	3.25E-03	3.78E-03	2.62E-03	3.79E-03	3.54E-03
MMF8	5.31E-03	2.76E-03	4.03E-03	3.22E-03	4.78E-03	3.72E-03	3.83E-03	**2.43E-03**
+/-/=	0/18/2	11/7/2	0/17/3	3/16/1	0/18/2	4/15/1	0/15/5	

difficulties in searching different PSs, it is difficult for these primitive MMEAs to obtain all PSs. As a result, they perform poorly in terms of IGDX on these problems. For traditional MMOPs such as the MMF test suite, the performance of the selected algorithms has little difference. For IDMPs, HREA, MMOEA/DC, and CPDEA show significant superiority over other algorithms.

We also compare the IGD results for these algorithms in Table II, from which we can find that HREA and CPDEA perform better than other algorithms. The final distribution of solutions in the objective and decision spaces obtained by all algorithms are provided in the supplementary material. From the figures, MMOEA/DC, CPDEA, HREA, and MO_PSO_MM can obtain all different PSs. Specifically, HREA can well balance the convergence and diversity of solutions. To summarize, HREA is competitive when compared to other state of the arts on the selected benchmark problems.

In addition, since we adopt some CEC 2019 test problems as benchmarks, we also compare the performances of HREA and other algorithms on CEC 2019 test suites. The average IGDX and IGD results can be found in the supplementary material. From the results, we can see that for normal MMOPs (MMF1-8), HREA performs slightly worse than the winner algorithm (MMO-ClusteringPSO). For MMOPLs, HREA shows significantly better performance. Since this work focuses on MMOPLs, we can say that HREA is a competitive algorithm for MMOPLs and an eligible algorithm for normal MMOPs. Furthermore, we set $\epsilon = 0.3$ for all benchmark problems. However, since MMF1-8 are normal MMOPs without local PFs, it is reasonable for HREA to obtain a better result with a smaller ϵ.

C. Results for MMOPLs

This section shows the performance of HREA and the competitor algorithms on MMOPLs. Specifically, eight MMOPLs based on the IDMP test suite are proposed. In addition, it is reported that MMF10, MMF11, MMF12, MMF13, MMF15, and MMF15a also belong to MMOPLs. Note that all experiments are executed 21 times. The Wilcoxon rank-sum test with $p < 0.05$ is also presented in the last column in Table III. Due to paper length limitations, the variance in the result is not listed. For more information, refer to the supplementary material.

Table III lists the average IGDX results obtained by all the competitor algorithms. As we can see from the table, HREA wins 13 instances over 14 test problems. Specifically, MMOEA/DC is designed for MMOPLs, and other compared algorithms are only designed for the traditional MMOPs. As a result, HREA and MMOEA/DC perform significantly better than other algorithms on the chosen benchmarks. For some problems, there is no significant difference between HREA and MMOEA/DC. For IDMPM3T3_e, MMOEA/DC shows better performance. Notably, it seems that other MMEAs cannot well solve MMOPLs since their IGDX results are poor. In addition, Table IV lists the average IGD results for all algorithms. Similar to that of IGDX, HREA shows significantly better performance. To summarize, HREA shows excellent ability in solving the chosen MMOPLs. MMOEA/DC can obtain some of the local PSs. Other traditional MMEAs cannot well solve MMOPLs.

Fig. 7 shows the distribution of the obtained solutions in the objective and decision spaces. There is one global PF corresponding to two different PSs and two local PFs corresponding to five different PSs. As observed, HREA finds all seven different PSs with good convergence and distribution, while MMOEA/DC can only find some of them. Other algorithms can find only a single PS (IDMPM2T4_e has only one global PS).

To further analyze the effect of the main population and the convergence archive, Fig. 8 presents the distribution of solutions in the decision space during the evolution. From Fig. 8,

TABLE III
IGDX RESULTS OF THE COMPARED ALGORITHMS ON MMOPS WITH LOCAL PFS, WHERE THE BEST MEAN FOR EACH TEST INSTANCE IS HIGHLIGHTED

Problems	TriMOEA TAR	CPDEA	DN-NSGA-II	Omni_Opt	MO_Ring _PSO_SCD	MO_PSO _MM	MMOEA /DC	HREA
IDMPM2T1_e	6.73E-01	6.73E-01	6.73E-01	6.73E-01	6.74E-01	6.74E-01	8.93E-04	**6.38E-04**
IDMPM2T2_e	6.73E-01	6.73E-01	6.74E-01	6.74E-01	6.74E-01	6.74E-01	9.93E-04	**9.20E-04**
IDMPM2T3_e	4.96E-01	3.01E-01	5.73E-01	5.94E-01	9.96E-02	2.78E-01	1.44E-03	**1.40E-03**
IDMPM2T4_e	9.67E-01	1.01E+00	9.10E-01	8.87E-01	7.60E-01	7.62E-01	1.44E-01	**3.88E-03**
IDMPM3T1_e	7.90E-01	6.26E-01	7.82E-01	8.14E-01	6.40E-01	6.16E-01	8.10E-03	**6.99E-03**
IDMPM3T2_e	9.01E-01	4.96E-01	7.49E-01	8.49E-01	5.59E-01	5.04E-01	2.54E-01	**7.93E-03**
IDMPM3T3_e	6.83E-01	5.02E-01	7.88E-01	7.91E-01	5.05E-01	5.05E-01	2.55E-01	**9.22E-03**
IDMPM3T4_e	1.01E+00	8.50E-01	1.01E+00	1.06E+00	7.47E-01	7.74E-01	**2.34E-02**	5.05E-01
MMF10	2.01E-01	2.01E-01	1.46E-01	1.64E-01	1.69E-01	1.66E-01	9.93E-03	**7.41E-03**
MMF11	2.52E-01	2.49E-01	2.50E-01	2.50E-01	2.08E-01	2.37E-01	7.62E-03	**7.46E-03**
MMF12	2.48E-01	2.45E-01	2.47E-01	2.44E-01	1.89E-01	2.24E-01	3.16E-03	**2.76E-03**
MMF13	2.71E-01	2.52E-01	2.98E-01	2.93E-01	2.45E-01	2.54E-01	1.08E-01	**4.85E-02**
MMF15	2.72E-01	2.33E-01	2.27E-01	2.27E-01	1.61E-01	1.55E-01	6.53E-02	**5.24E-02**
MMF15a	2.82E-01	2.54E-01	2.43E-01	2.51E-01	1.79E-01	1.88E-01	5.78E-02	**5.95E-02**
+/-/=	0/14/0	0/14/0	0/14/0	0/14/0	0/14/0	0/14/0	1/8/5	

TABLE IV
IGD RESULTS OF THE COMPARED ALGORITHMS ON MMOPS WITH LOCAL PFS, WHERE THE BEST MEAN FOR EACH TEST INSTANCE IS HIGHLIGHTED

Problems	TriMOEA TAR	CPDEA	DN-NSGA-II	Omni_Opt	MO_Ring _PSO_SCD	MO_PSO _MM	MMOEA /DC	HREA
IDMPM2T1_e	7.38E-03	7.16E-03	7.22E-03	7.20E-03	7.17E-03	7.30E-03	1.39E-03	**1.02E-03**
IDMPM2T2_e	7.40E-03	7.23E-03	7.27E-03	7.25E-03	7.19E-03	7.47E-03	9.34E-04	**9.15E-04**
IDMPM2T3_e	1.81E-02	4.98E-03	5.56E-03	5.93E-03	8.47E-03	5.96E-03	**1.17E-03**	1.18E-03
IDMPM2T4_e	1.43E-02	1.59E-02	1.34E-02	1.21E-02	1.42E-02	1.46E-02	**1.48E-03**	1.52E-03
IDMPM3T1_e	2.80E-02	2.45E-02	2.57E-02	2.57E-02	2.49E-02	2.51E-02	7.68E-03	**6.89E-03**
IDMPM3T2_e	4.02E-02	3.67E-02	3.90E-02	4.20E-02	3.86E-02	3.72E-02	1.58E-02	**8.27E-03**
IDMPM3T3_e	3.99E-02	3.67E-02	3.89E-02	3.87E-02	3.75E-02	3.71E-02	1.56E-02	**8.98E-03**
IDMPM3T4_e	5.99E-02	5.87E-02	4.07E-02	4.38E-02	4.20E-02	3.97E-02	**8.79E-03**	2.53E-02
MMF10	2.30E-01	1.94E-01	1.78E-01	1.90E-01	2.05E-01	1.64E-01	**1.81E-02**	2.50E-02
MMF11	1.65E-01	9.32E-02	9.82E-02	9.56E-02	8.37E-02	8.84E-02	**2.08E-02**	2.79E-02
MMF12	8.55E-02	8.31E-02	8.32E-02	8.41E-02	6.76E-02	7.82E-02	**3.85E-03**	6.48E-03
MMF13	2.48E-01	1.40E-01	1.58E-01	1.50E-01	1.18E-01	1.17E-01	3.27E-02	**1.74E-02**
MMF15	2.10E-01	1.81E-01	2.36E-01	2.21E-01	1.86E-01	1.81E-01	1.27E-01	**1.23E-01**
MMF15a	2.40E-01	1.74E-01	2.24E-01	2.45E-01	1.99E-01	1.98E-01	1.34E-01	**1.30E-01**
+/-/=	0/13/1	0/13/2	0/14/0	0/14/0	0/14/0	0/14/0	5/3/6	

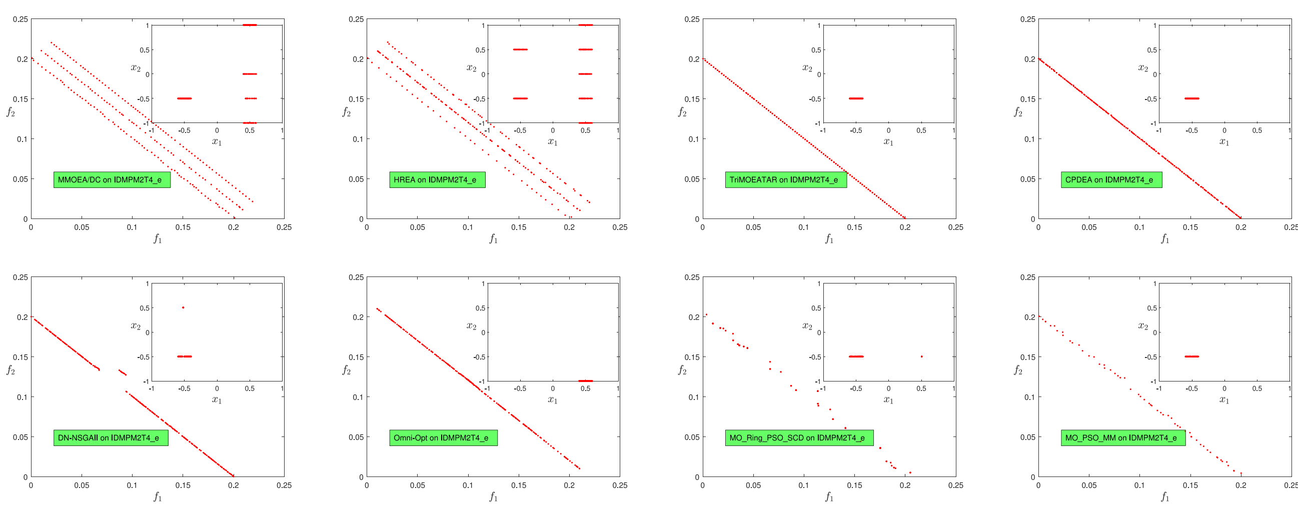


Fig. 7. Distribution of the obtained solutions in the decision and objective spaces on IDMPM2T4_e, where the PSs are shown in the subfigures in the upper right.

we can observe that in the main population, solutions do not directly move toward the local PSs, even in the final stage of the evolution, which occurs because we use the local convergence quality to continue exploring the decision space. As a comparison, solutions in the convergence archive are nondominated in each PS. In the early stage, there are few solutions. As the evolution proceeds, the number of solutions in each PS is well balanced.

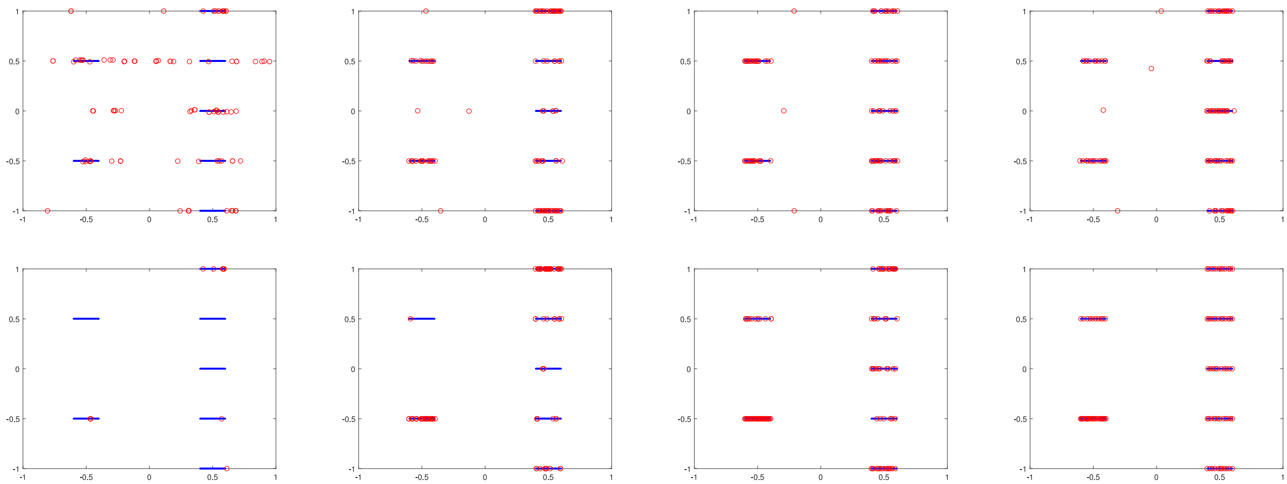


Fig. 8. Distribution of solutions in the decision spaces during the evolution process on IDMPM2T4_e. The first four subfigures are the 1000th, 3000th, 5000th, and 9000th evaluations of the main population. The last four subfigures are in the same states of the convergence archive.

D. Sensitivity Analysis of Parameters

1) Effect of Parameter ϵ: We introduce a user-defined parameter ϵ to control the convergence quality of the obtained solutions. Specifically, the quality of the local PFs can be expressed by Definition 5. To analyze the effect of ϵ, we choose MMF12 as the test problem. All the other parameters are set the same as those in Section IV-A.

Fig. 9 shows the obtained solutions by HREA with different ϵ values. From the figures, we can see that ϵ can control the quality of the local PFs. If we set $\epsilon = 0$, then HREA only focuses on the global PF and global PSs. In this situation, HREA behaviors resemble the traditional MMEAs. With the increase in ϵ, HREA can reserve local PFs with acceptable quality. Then, DMs can run HREA according to their preference and thus, obtain PSs with different qualities.

2) Effect of Parameter p: In Section III-B, we introduce a parameter p to randomly select parents from the convergence archive. Specifically, the larger p is, the more likely the HREA is to choose parents from the convergence archive. Reasonably, since the archive is introduced to improve the convergence ability of the algorithm, it is predictable that a larger p can achieve better performance. However, since the main population focuses on exploring the whole decision space and utilizing the local convergence to maintain all local PSs, p should not be set to an extremely large value. To analyze the effect of parameter p, HREA is examined on 14 MMOPL (IDMP_e test suite, MMF10, MMF11, MMF12, MMF13, MMF15, and MMF15a) problems with different $p \in [0, 1]$. Then, for each test problem, the average IGDX and IGD values over 21 runs are used to calculate the rank. The smaller the average rank is, the better the algorithm.

Due to paper length limitations, we only list the average rank of IGDX and IGD, which is shown in Table V. The detailed values can be found in the supplementary material. From the table, we can see that there is little difference in the average rank of IGDX and IGD as the value of parameter p changes, which means that the performance of HREA is not sensitive to p. As a tradeoff, we set $p = 0.5$ as the default value.

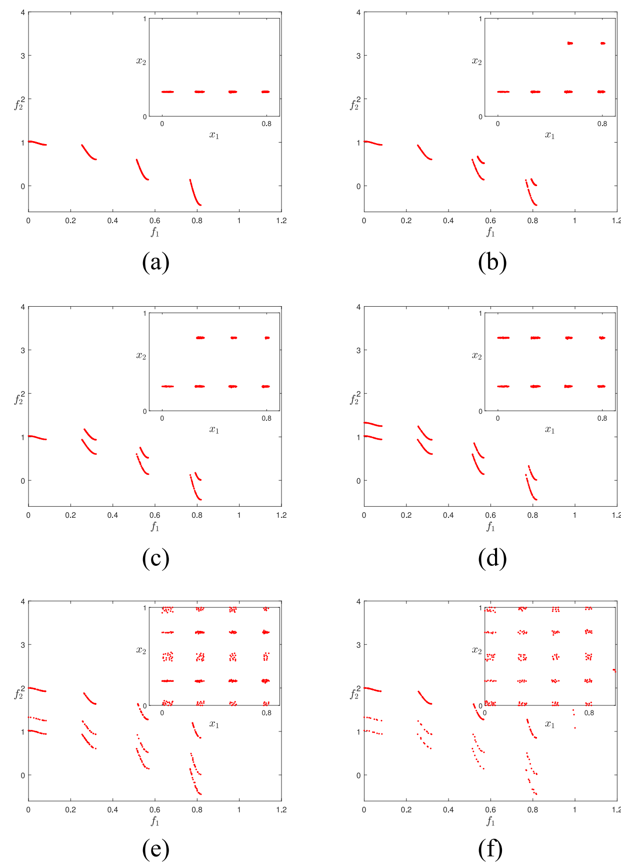


Fig. 9. Distribution of the obtained solutions in the decision and objective spaces on MMF12 by HREA with different ϵ values, where the PSs are shown in the subfigures in the upper right. (a) $\epsilon = 0$. (b) $\epsilon = 0.1$. (c) $\epsilon = 0.2$. (d) $\epsilon = 0.3$. (e) $\epsilon = 0.5$. (f) $\epsilon = 1$.

V. Conclusion

MMOPs are common in the real world. To date, most studies consider the case in which more than one distant solutions share the same objective values. However, for real-world problems, a more practical situation is that two distant solutions have similar objective values. That is, the global

TABLE V
AVERAGE RANK OF IGDX AND IGD OVER EIGHT TEST PROBLEMS OBTAINED BY HREA WITH DIFFERENT p VALUES

p	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
IGDX	8.125	7.75	5.875	6.375	5.75	4.625	6.125	6.75	6.25	5.875	2.5
IGD	5.875	6.25	5.875	8	6.625	6.2	5.875	4.875	6.125	5.375	6.525

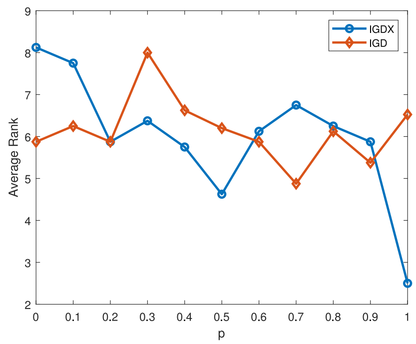


Fig. 10. Average rank of IGDX and IGD over 14 test problems obtained by HREA with different values of p.

Pareto optimal front and local optimal front are both of DMs interests. However, such problems have not been systematically investigated.

In this study, we first analyze the challenges of MMOPs with local PFs. A new benchmark, namely, IDMP_e, is proposed that has both global and local PFs. Specifically, the difficulty of obtaining different PSs is different. Accordingly, a novel evolutionary algorithm using the hierarchy ranking method (HREA) is proposed. Through the experiments, the effectiveness of HREA is demonstrated.

By solving the MMOPs with local PFs, it is found that the proposed strategies are also effective in solving traditional MMOPs. Thus, it is more practical to develop MMEAs by considering local PFs in the future. Moreover, the diversity-maintenance method can well improve the ability of the algorithm to escape from the local optima, which can also be utilized in traditional MOEAs to improve the searching ability. Currently, the existing MMEAs including this work only consider obtaining all PSs without considering the robustness of different PSs. For some problems, the landscapes of different regions have different steepness degrees. That is, the optimal solutions located in a gentle landscape can tolerant some mistakes, which is considered a more robust solution. However, no systematic works have discussed this situation.

REFERENCES

[1] K. Deb, "Multi-objective optimization," in *Search Methodologies*. Boston, MA, USA: Springer, 2014, pp. 403–449.

[2] K. Li, R. Wang, T. Zhang, and H. Ishibuchi, "Evolutionary many-objective optimization: A comparative study of the state-of-the-art," *IEEE Access*, vol. 6, pp. 26194–26214, 2018.

[3] W. Li, R. Wang, T. Zhang, M. Ming, and K. Li, "Reinvestigation of evolutionary many-objective optimization: Focus on the Pareto knee front," *Inf. Sci.*, vol. 522, pp. 193–213, Jun. 2020.

[4] J. J. Liang, C. Yue, and B.-Y. Qu, "Multimodal multi-objective optimization: A preliminary study," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, 2016, pp. 2454–2461.

[5] Z.-J. Wang *et al.*, "Dual-strategy differential evolution with affinity propagation clustering for multimodal optimization problems," *IEEE Trans. Evol. Comput.*, vol. 22, no. 6, pp. 894–908, Dec. 2018.

[6] G. Rudolph, B. Naujoks, and M. Preuss, "Capabilities of EMOA to detect and preserve equivalent Pareto subsets," in *Proc. Int. Conf. Evol. Multi-Criterion Optim.*, 2007, pp. 36–50.

[7] O. Schutze, M. Vasile, and C. A. Coello Coello, "Computing the set of epsilon-efficient solutions in multiobjective space mission design," *J. Aerosp. Comput. Inf. Commun.*, vol. 8, no. 3, pp. 53–70, 2011.

[8] F. Kudo, T. Yoshikawa, and T. Furuhashi, "A study on analysis of design variables in pareto solutions for conceptual design optimization problem of hybrid rocket engine," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, 2011, pp. 2558–2562.

[9] M. Sebag, N. Tarrisson, O. Teytaud, J. Lefevre, and S. Baillet, "A multi-objective multi-modal optimization approach for mining stable spatio-temporal patterns," in *Proc. IJCAI*, 2005, pp. 859–864.

[10] R. Tanabe and H. Ishibuchi, "A review of evolutionary multimodal multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 24, no. 1, pp. 193–200, Feb. 2020.

[11] K. Deb and S. Tiwari, "Omni-optimizer: A procedure for single and multi-objective optimization," in *Proc. Int. Conf. Evol. Multi-Criterion Optim.*, 2005, pp. 47–61.

[12] K. Deb and S. Tiwari, "Omni-optimizer: A generic evolutionary algorithm for single and multi-objective optimization," *Eur. J. Oper. Res.*, vol. 185, no. 3, pp. 1062–1087, 2008.

[13] Y. Liu, H. Ishibuchi, Y. Nojima, N. Masuyama, and K. Shang, "A double-niched evolutionary algorithm and its behavior on polygon-based problems," in *Proc. Int. Conf. Parallel Probl. Solving Nature*, 2018, pp. 262–273.

[14] C. Yue, B. Qu, and J. Liang, "A multiobjective particle swarm optimizer using ring topology for solving multimodal multiobjective problems," *IEEE Trans. Evol. Comput.*, vol. 22, no. 5, pp. 805–817, Oct. 2018.

[15] Y. Liu, H. Ishibuchi, G. G. Yen, Y. Nojima, and N. Masuyama, "Handling imbalance between convergence and diversity in the decision space in evolutionary multimodal multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 24, no. 3, pp. 551–565, Jun. 2020.

[16] W. Li, T. Zhang, R. Wang, and H. Ishibuchi, "Weighted indicator-based evolutionary algorithm for multimodal multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 25, no. 6, pp. 1064–1078, Dec. 2021.

[17] Y. Hu *et al.*, "A self-organizing multimodal multi-objective pigeon-inspired optimization algorithm," *Sci. China Inf. Sci.*, vol. 62, no. 7, pp. 1–17, 2019.

[18] R. Tanabe and H. Ishibuchi, "A framework to handle multimodal multiobjective optimization in decomposition-based evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 24, no. 4, pp. 720–734, Aug. 2020.

[19] Q. Lin, W. Lin, Z. Zhu, M. Gong, J. Li, and C. A. Coello Coello, "Multimodal multiobjective evolutionary optimization with dual clustering in decision and objective spaces," *IEEE Trans. Evol. Comput.*, vol. 25, no. 1, pp. 130–144, Feb. 2021.

[20] J. Liang, P. N. Suganthan, B. Qu, D. Gong, and C. Yue, "Problem definitions and evaluation criteria for the CEC 2019 special session on multimodal multiobjective optimization," Comput. Intell. Lab., Zhengzhou University, Zhengzhou, China, Rep., 201912, 2019.

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

[22] O. M. Shir, M. Preuss, B. Naujoks, and M. Emmerich, "Enhancing decision space diversity in evolutionary multiobjective algorithms," in *Proc. Int. Conf. Evol. Multi-Criterion Optim.*, 2009, pp. 95–109.

[23] J. Liang, Q. Guo, C. Yue, B. Qu, and K. Yu, "A self-organizing multi-objective particle swarm optimization algorithm for multimodal multi-objective problems," in *Proc. Int. Conf. Swarm Intell.*, 2018, pp. 550–560.

[24] J. Liang *et al.*, "Multimodal multiobjective optimization with differential evolution," *Swarm Evol. Comput.*, vol. 44, pp. 1028–1059, Feb. 2019.

[25] H. Zhao *et al.*, "Local binary pattern-based adaptive differential evolution for multimodal optimization problems," *IEEE Trans. Cybern.*, vol. 50, no. 7, pp. 3343–3357, Jul. 2020.
[26] Z.-J. Wang *et al.*, "Automatic niching differential evolution with contour prediction approach for multimodal optimization problems," *IEEE Trans. Evol. Comput.*, vol. 24, no. 1, pp. 114–128, Feb. 2020.
[27] R. Tanabe and H. Ishibuchi, "A decomposition-based evolutionary algorithm for multi-modal multi-objective optimization," in *Proc. Int. Conf. Parallel Probl. Solving Nature*, 2018, pp. 249–261.
[28] C. Hu and H. Ishibuchi, "Incorporation of a decision space diversity maintenance mechanism into MOEA/D for multi-modal multi-objective optimization," in *Proc. Genet. Evol. Comput. Conf. Companion (GECCO)*, 2018, pp. 1898–1901.
[29] Y. Liu, H. Ishibuchi, Y. Nojima, N. Masuyama, and Y. Han, "Searching for local pareto optimal solutions: A case study on polygon-based problems," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, 2019, pp. 896–903.
[30] W. Wang, G. Li, Y. Wang, F. Wu, W. Zhang, and L. Li, "Clearing-based multimodal multi-objective evolutionary optimization with layer-to-layer strategy," *Swarm Evol. Comput.*, vol. 68, Feb. 2022, Art. no. 100976.
[31] Y. Tian, T. Zhang, J. Xiao, X. Zhang, and Y. Jin, "A coevolutionary framework for constrained multiobjective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 25, no. 1, pp. 102–116, Feb. 2021.
[32] M. Ming, A. Trivedi, R. Wang, D. Srinivasan, and T. Zhang, "A dual-population-based evolutionary algorithm for constrained multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 25, no. 4, pp. 739–753, Aug. 2021.
[33] Y. Liu, H. Ishibuchi, G. G. Yen, Y. Nojima, N. Masuyama, and Y. Han, "On the normalization in evolutionary multi-modal multi-objective optimization," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, 2020, pp. 1–8.
[34] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, "Combining convergence and diversity in evolutionary multiobjective optimization," *Evol. Comput.*, vol. 10, no. 3, pp. 263–282, 2002.
[35] C. Horoba and F. Neumann, "Benefits and drawbacks for the use of epsilon-dominance in evolutionary multi-objective optimization," in *Proc. 10th Annu. Conf. Genet. Evol. Comput.*, 2008, pp. 641–648.
[36] Y. Liu, G. G. Yen, and D. Gong, "A multimodal multiobjective evolutionary algorithm using two-archive and recombination strategies," *IEEE Trans. Evol. Comput.*, vol. 23, no. 4, pp. 660–674, Aug. 2019.
[37] E. Zitzler, L. Thiele, M. Laumanns, C. Fonseca, and V. da Fonseca, "Performance assessment of multiobjective optimizers: An analysis and review," *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 117–132, Apr. 2003.
[38] C. A. Coello Coello and M. R. Sierra, "A study of the parallelization of a coevolutionary multi-objective evolutionary algorithm," in *Proc. Mexican Int. Conf. Artif. Intell.*, 2004, pp. 688–697.
[39] A. Zhou, Q. Zhang, and Y. Jin, "Approximating the set of pareto-optimal solutions in both the decision and objective spaces by an estimation of distribution algorithm," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 1167–1189, Oct. 2009.
[40] R. Tanabe and H. Ishibuchi, "A niching indicator-based multi-modal many-objective optimizer," *Swarm Evol. Comput.*, vol. 49, pp. 134–146, Sep. 2019.
[41] Y. Tian, R. Cheng, X. Zhang, and Y. Jin, "PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [educational forum]," *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 73–87, Nov. 2017.

Wenhua Li received the B.S. and M.S. degrees from the National University of Defense Technology, Changsha, China, in 2018 and 2020, respectively, where he is currently pursuing the Ph.D. degree in management science and technology.

His current research interests include multiobjective evolutionary algorithms, energy management in microgrids, and artificial intelligence.

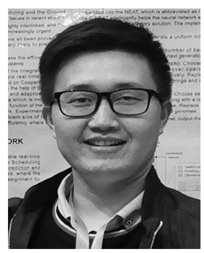
Xingyi Yao received the B.S. degree from the National University of Defense Technology, Changsha, China, in 2017, where she is currently pursuing the master's degree with the College of Systems Engineering.

Her main research areas are multicriteria decision making, optimal scheduling, data mining, and optimization methods.

Tao Zhang received the B.S., M.S., and Ph.D. degrees from the National University of Defense Technology (NUDT), Changsha, China, in 1998, 2001, and 2004, respectively.

He is a Full Professor with the College of Systems Engineering, NUDT. He is also the Director of the Hunan Key Laboratory of Multi-Energy System Intelligent Interconnection Technology, Changsha. His current research interests include optimal scheduling, data mining, and optimization methods on the energy Internet.

Prof. Zhang was the recipient of the Science and Technology Award of Provincial Level (first place in 2020 and 2021, and second place in 2015 and 2018).



Rui Wang (Senior Member, IEEE) received the bachelor's degree from the National University of Defense Technology, Changsha, China, in 2008, and the Doctoral degree from the University of Sheffield, Sheffield, U.K., in 2013.

He is currently an Associate professor with the National University of Defense Technology. His current research interests include evolutionary computation, multiobjective optimization, and the development of algorithms applicable in practice.

Dr. Wang received the Operational Research Society Ph.D. Prize in 2016, and the National Science Fund for Outstanding Young Scholars in 2021. He is also an Associate Editor of the *Swarm and Evolutionary Computation* and the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION.

Ling Wang received the B.Sc. degree in automation and the Ph.D. degree in control theory and control engineering from Tsinghua University, Beijing, China, in 1995 and 1999, respectively.

Since 1999, he has been with the Department of Automation, Tsinghua University, where he became a Full Professor in 2008. His current research interests include intelligent optimization and production scheduling.

Prof. Wang was the recipient of the National Natural Science Fund for Distinguished Young Scholars of China, the National Natural Science Award (second place) in 2014, the Science and Technology Award of Beijing City in 2008, and the Natural Science Award (first place in 2003 and second place in 2007) nominated by the Ministry of Education of China.