A Multi-Modal Multi-Objective Evolutionary Algorithm Using Two-Archive and Recombination Strategies

Yiping Liu, *Member, IEEE*, Gary G. Yen, *Fellow, IEEE*, and Dunwei Gong, *Member, IEEE*

***Abstract*—There have been few researches on solving multi-modal multi-objective optimization problems, whereas they are commonly seen in real-world applications but difficult for the existing evolutionary optimizers. In this study, we propose a novel multi-modal multi-objective evolutionary algorithm using two-archive and recombination strategies. In the proposed algorithm, the properties of decision variables and the relationships among them are analyzed at first to guide the evolutionary search. Then, a general framework using two archives, i.e., the convergence and the diversity archives, is adopted to cooperatively solve these problems. Moreover, the diversity archive simultaneously employs a clustering strategy to guarantee diversity in the objective space and a niche-based clearing strategy to promote the same in the decision space. At the end of evolution process, solutions in the convergence and the diversity archives are recombined to obtain a large number of multiple Pareto optimal solutions. In addition, a set of benchmark test functions and a performance metric are designed for multi-modal multi-objective optimization. The proposed algorithm is empirically compared with two state-of-the-art evolutionary algorithms on these test functions. The comparative results demonstrate that the overall performance of the proposed algorithm is significantly superior to the competing algorithms.**

***Index Terms*—Multi-modal multi-objective optimization, evolutionary optimization, convergence, diversity, niche, recombination.**

I. Introduction

MULTI-objective optimization problems (MOPs) are commonly seen in a variety of disciplines. They involve multiple objectives to be optimized simultaneously. Due to the conflicting nature of objectives, there is no single optimal solution to an MOP, rather a set of trade-off solutions, known as the Pareto optimal solution set.

Over the past two decades, a large number of Multi-Objective Evolutionary Algorithms (MOEAs) have been proposed to solve MOPs, e.g., elitist Non-dominated Sorting Genetic Algorithm II (NSGA-II) [1], advanced Strength Pareto Evolutionary Algorithm 2 (SPEA2) [2], and MOEA Based on Decomposition (MOEA/D) [3], to name a few. Compared to traditional mathematical programming techniques, MOEAs are particularly suited in searching for the Pareto optimal solution set in one single run. Generally, the goal of MOEAs is to find solutions not only close to the Pareto optimal front (i.e., convergence performance), but also uniformly and widely distributed (i.e., diversity performance in the objective space).

In real-world applications, many MOPs also exhibit multi-modal properties [4], [5], e.g., multi-objective knapsack optimization problem [6], architecture layout design problem [7], multi-objective scheduling problem [8] and map-based problem [9]. That means multiple different Pareto optimal solutions coexist in the decision space for the same point on the Pareto optimal front. We define these problems to be Multi-Modal Multi-Objective Optimization Problems (MMMOPs). To successfully solve an MMMOP, we need to find all the Pareto optimal solutions for each point on the Pareto optimal front. Intuitively, if an MOEA is applied to deal with an MMMOP, only one single Pareto optimal solution can usually be found for a point on the Pareto optimal front, since maintaining multiple Pareto optimal solutions in the decision space is not considered a design goal in the current MOEAs. Even if the MOEA is applied several times, not all Pareto optimal solutions to the MMMOP are guaranteed to be found.

On the other hand, there have been a great deal of researches on EAs to solve Multi-Modal Optimization Problems (MMOPs), aiming to find multiple optimal solutions for a single-objective optimization problem. Among various designs, niching methods are the most popular ones, e.g., fitness sharing [10] and crowding [11]. However, all these methods can only solve single-objective MMOPs.

Solving MMMOPs is definitely not an easy task by simply combining MOEAs with the above niching methods. There exist two major issues for EAs on tackling MMMOPs. The first issue is that EAs need to maintain a diverse population both in the objective and the decision spaces (i.e. diversity performances in the objective and the decision spaces, respectively), while converging the population towards the Pareto optimal front (i.e. convergence performance). Thus a selection strategy that can well balance these three performances is a must. Moreover, diversities in the objective and the decision spaces are quite different requirements. The former means to find a uniformly and widely distributed Pareto optimal solution set in the objective space, whereas the latter does not imply

Manuscript received March 9, 2018; revised May 13, 2018 and August 23, 2018; accepted October 28, 2018. This work was jointly supported by National Natural Science Foundation of China with grant No. 61803192, 61876075, 61773384, 61763026, 61673404, 61573361, and 61503220, National Basic Research Program of China (973 Program) with grant No. 2014CB046306-2, National Key R&D Program of China with grant No. 2018YFB1003802-01, and China Scholarship Council with grant No. 201606420005. (*Corresponding author: Gary G. Yen.*)

Y. Liu is with School of Information and Control Engineering, China University of Mining and Technology, Xuzhou 221116, China. (yiping0liu@gmail.com)

G. G. Yen is with the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74078, USA. (gyen@okstate.edu)

D. Gong is with School of Information and Control Engineering, China University of Mining and Technology, Xuzhou 221116, China. (dwgong@vip.163.com)

that in the decision space. When tackling an MMMOP that has bias in the decision space (see Benchmark MMMOP2 in Subsection IV.A for example), searching for uniformly distributed solutions in the decision space will result in poor diversity performance in the objective space. The second issue is that maintaining diversities both in the objective and the decision spaces may require an extremely large population. Assuming an MMMOP with each point on its Pareto front corresponding to i (e.g., 20) Pareto optimal solutions and j (e.g. 100) individuals (i.e., solutions) being to express a diverse population in the objective space, we will need a population with at least $i \times j$ (e.g., 2,000) individuals to solve the MMMOP. Such a large population will consume significant computing resources during the evolution process, especially if the objective functions are computationally expensive to evaluate.

In this study, we propose a novel Multi-Modal Multi-Objective Evolutionary Algorithm using Two-Archive and Recombination strategies (TriMOEA-TA&R) to solve MMMOPs. In the proposed algorithm, the independent convergence-related decision variables are detected by a decision variable analytical technique before the evolution. These decision variables have no interaction with the other variables and can be optimized separately. Moreover, optimizing them only contributes to convergence improvement in the objective space. We define the search space of these decision variables as the independent convergence-related decision subspace. There are two archives in the proposed algorithm, i.e., the convergence and the diversity archives, which have different tasks. The convergence archive focuses on locating diverse converged solutions in the independent convergence-related decision subspace. Meanwhile, the diversity archive aims at maintaining diversity in the objective space and the remainder decision subspace (i.e. removing the independent convergence-related decision subspace from the total search space). In each generation, a new population is formed by the offspring of solutions in these archives. Conversely, the archives are updated based on the new population. At the end of the evolution process, a final solution set with good overall performance is constructed by recombining the superior solutions in the archives.

The main contributions of this work can be summarized as follows:

First, a multi-modal multi-objective evolutionary algorithm using two-archive and recombination strategies, termed TriMOEA-TA&R is presented. It can achieve a Pareto optimal solution set with good convergence and diversities both in the objective and the decision spaces. There are two major strategies in TriMOEA-TA&R. The first is the two-archive strategy, where the division of labor makes the optimizing work easier. The other is the recombination strategy, which can generate a large number of Pareto optimal solutions based on the analysis of decision variables.

Next, a set of novel benchmark MMMOPs is proposed. These benchmark test functions are all scalable. Their building blocks are originated from the most popular benchmarks in multi-modal optimization and multi-objective optimization to preserve the designed problem characteristics. They also incorporate different new features in terms of the complexity of MMMOPs. These benchmarks can comprehensively and efficiently test an optimizer's ability in solving MMMOPs.

Last but not least, a novel performance metric, Inverted Generational Distance-Multi-modal (IGDM), is proposed to assess the performance of the solution set obtained by a multi-modal multi-objective optimizer. IGDM is an extension of Inverted Generational Distance (IGD) [12]. It can measure the convergence performance and the diversity performances both in the objective and the decision spaces.

The remainder of this paper is organized as below. In Section II, the related works on multi-objective optimization and multi-modal optimization are reviewed for the completeness of the presentation and the motivation of this work is also elaborated. The proposed algorithm is then described in detail in Section III. Section IV presents the experimental design, test functions, and performance indicator for investigating the performance of the proposed algorithm. The experimental results and relevant discussions are given in Section V. Section VI concludes the paper and provides pertinent observations and future research directions.

II. Preliminaries

A. Multi-Objective Evolutionary Optimization

Without loss of generality, an MOP can be mathematically expressed as follows [13]:

$$\begin{aligned} &\min / \max \boldsymbol{f}(\boldsymbol{x}) = \min / \max(f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), ..., f_M(\boldsymbol{x})) \\ &\text{s.t. } \boldsymbol{x} \in S \subset \boldsymbol{R}^n \end{aligned} \tag{1}$$

where $\boldsymbol{x} = (x_1, ..., x_n)$ represents an n-dimensional decision vector in the search space S. $f_m(\boldsymbol{x}), m = 1, ..., M$, is the m-th objective to be minimized/maximized, and M is the number of objectives. Note that we only consider MOPs (and MMMOPs) with no constraint or box constraints in this study.

In multi-objective optimization, the following concepts have been well defined and widely adopted [14].

Pareto Dominance: For any two different solutions, $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, to the MOP in Eq. (1), if there is no objective where $\boldsymbol{x}_1$ is worse than $\boldsymbol{x}_2$, and there exists at least one objective where $\boldsymbol{x}_1$ is better than $\boldsymbol{x}_2$, then $\boldsymbol{x}_1$ dominates $\boldsymbol{x}_2$, denoted as $\boldsymbol{x}_1 \prec \boldsymbol{x}_2$.

Local Pareto Optimal Solution: For a solution $\boldsymbol{x}'$, if there exists no solution $\boldsymbol{x}$ satisfying $||\boldsymbol{x} - \boldsymbol{x}'||_\infty \leq \epsilon$, where ϵ is a small positive number, dominates $\boldsymbol{x}'$, then $\boldsymbol{x}'$ is a local Pareto optimal solution. ($||\boldsymbol{v}||_\infty$ is the infinity norm of $\boldsymbol{v}$.)

Global Pareto Optimal Solution: If there exists no solution in the search space that dominates $\boldsymbol{x}^*$, then $\boldsymbol{x}^*$ is a global Pareto optimal solution.

Pareto Optimal Solution Set: All the global Pareto optimal solutions form a set called the Pareto optimal solution set (PS).

Pareto Optimal Front: The image of the Pareto optimal solution set in the objective space is known as the Pareto optimal front (PF).

Pareto-based MOEAs are the most popular algorithms to solve MOPs. These algorithms first adopt the Pareto dominance principle to select non-dominated solutions which are always preferred for good convergence. Then a density-based selection criterion is employed to promote a good diversity among the solutions. There are various density-based selection criteria in Pareto-based MOEAs, e.g., crowding distance in NSGA-II [1] and k-nearest method in SPEA2 [2]. Pareto-based MOEAs have been proven successful in solving a large number of MOPs.

The decomposition-based MOEAs are found to be a promising alternative to solve MOPs, MOEA/D [3] is well-regarded representative. In MOEA/D, a set of well distributed reference vectors are first defined. Individuals (solutions) in a population are guided to search towards the PF in the directions specified by the reference vectors.

Indicator-Based Evolutionary Algorithms (IBEA) [15] are theoretically well-supported options to the Pareto-based MOEAs. It adopts a performance indicator to account for both convergence and diversity of a solution. Among others, hypervolume (HV) [16] is a widely used indicator in IBEAs.

Recently, solving MOPs with a large number of objectives gained much attention. Some MOEAs have been proposed to balancing convergence and diversity in the high dimensional objective space, such as [17]–[19].

From the above, we can see that MOEAs are designed to obtain a well converged and distributed solution set in the objective space. Due to the lacking of diversity maintenance techniques in the decision space, they are incapable of solving MMMOPs. In this study, the proposed TriMOEA-TA&R can overcome this issue and provides diverse Pareto optimal solutions in the decision space to a given MMMOP.

B. Multi-Modal Evolutionary Optimization

Multi-modal evolutionary algorithms focus on locating multiple optimal solutions in a single run. Niching methods [20] are commonly used as the diversity-preserving mechanism in these EAs, where fitness sharing [10], crowding methods [11] are the classic choices. In fitness sharing methods, individuals in the same neighborhood will degrade each other's fitness, thereby discouraging the number of individuals occupying the same niche. In crowding methods, an offspring and its close parents compete with each other, and individuals with better fitness in the sparse areas are favored. There are a lot of niche-based evolutionary algorithms developed in the last two decades, e.g., clearing-based genetic algorithm [21], cluster-based differential evolution [22], locally informed particle swarm optimization [23], and transformation technique based on multi-objective optimization [24].

However, all the above methods can find multiple optimal solutions only for a single-objective optimization problem. In this study, we aim at designing a novel multi-modal evolutionary algorithm that is effective on multi-objective optimization problems.

C. Multi-Modal Multi-Objective Optimization

1) What is an MMMOP: To our best knowledge, multi-modal multi-objective optimization problems (MMMOPs) were first introduced in [4], and recently defined in [5]. In [5], MMMOPs were defined as the MOPs which have more than one Pareto optimal set, i.e., there are at least two different feasible regions in the decision space corresponding to the same region of the objective space. Actually, some MOPs (e.g., the one in Eq. (2) that will be discussed later) do not have multiple different feasible regions, but multiple different Pareto optimal solutions in the decision space corresponding to the same point in the objective space. These MOPs should also be classified as MMMOPs. Therefore, based on the definitions in Subsection II.A, we give a more general definition of an MMMOP in this study.

Definition of an MMMOP: For an MOP in Eq. (1), if there exists at least one local Pareto optimal solution or at least two different global Pareto optimal solutions for any point on the PF, then the MOP is considered an MMMOP.

Intuitively, it is much more complicated to solve an MMMOP by finding its local Pareto optimal solution(s). One issue is how to justify whether the local Pareto optimal solutions are really needed by the decision maker, since their objective values are not good. Another issue is how to achieve the desired local Pareto optimal solutions while not to get trapped into the needless local optimal regions. At the early stage of the research on MMMOPs, we only consider solving an MMMOP with multiple global Pareto optimal solutions for a point on its PF in this paper. In other words, our goal is to find the different global Pareto optimal solutions at the same location on a PF in this study. For brevity, we refer to a global Pareto optimal solution as a Pareto optimal solution in the rest of the paper.

MMMOPs are commonly seen in real-world applications. Let us discuss a simple example in multi-objective knapsack optimization problem [6]. Assume that Max is a salesperson who sells ice creams. There are six kinds of ice creams ($IC_i, i = 1, ..., 6$), and they have different monetary profits ($p_1 = 4, p_2 = 6, p_3 = 3, p_4 = 7, p_5 = 5, p_6 = 2$) and shelf lives ($q_1 = 6, q_2 = 4, q_3 = 7, q_4 = 3, q_5 = 5, q_6 = 2$). Since the refrigerator can only store three kinds of ice creams, Max has to select three different kinds of ice creams to maximize the total monetary profit and shelf life. This problem can be formulated as follows:

$$\begin{aligned} \max f_1(\boldsymbol{x}) &= \textstyle\sum_{i=1,...,6} p_i x_i \\ \max f_2(\boldsymbol{x}) &= \textstyle\sum_{i=1,...,6} q_i x_i \end{aligned} \tag{2}$$

where $f_1(\boldsymbol{x})$ is the total monetary profit, $f_2(\boldsymbol{x})$ is the total shelf life, $\boldsymbol{x} = (x_1, ..., x_6)$, $\sum_{i=1,...,6} x_i = 3$, and $x_i = 1$ or 0 means that IC_i is selected or not, respectively.

All the Pareto optimal solutions to the problem in Eq. (2) are listed as follows:

$$\begin{aligned} &\boldsymbol{x}_1^* = (1,0,1,0,1,0), \boldsymbol{x}_2^* = (1,1,1,0,0,0), \\ &\boldsymbol{x}_3^* = (1,0,1,1,0,0), \boldsymbol{x}_4^* = (0,1,1,0,1,0), \\ &\boldsymbol{x}_5^* = (1,1,0,0,1,0), \boldsymbol{x}_6^* = (0,0,1,1,1,0), \\ &\boldsymbol{x}_7^* = (0,1,1,1,0,0), \boldsymbol{x}_8^* = (1,0,0,1,1,0), \\ &\boldsymbol{x}_9^* = (1,1,0,1,0,0), \boldsymbol{x}_{10}^* = (0,1,0,1,1,0). \end{aligned} \tag{3}$$

The PF of the problem is shown in Fig. 1, where the Pareto optimal solutions are presented as black solid dots. We can see from Fig. 1 that $\boldsymbol{x}_3^*$ and $\boldsymbol{x}_4^*$, $\boldsymbol{x}_5^*$ and $\boldsymbol{x}_6^*$, and $\boldsymbol{x}_7^*$ and $\boldsymbol{x}_8^*$ have the same objective values, respectively, i.e, they are the same point in the objective space, respectively, whereas they are different solutions in the decision space. Therefore, the problem in Eq. (2) is an MMMOP.

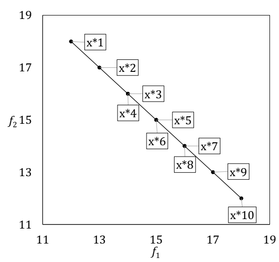


Fig. 1. Pareto optimal front of the optimization problem in Eq. (2).

2) Why solve an MMMOP: In multi-objective evolutionary optimization, the general task of an MOEA is to search for an evenly and widely distributed (in the objective space) Pareto optimal solution set. Then, the decision maker chooses one solution among them according to his/her preference. However, in real-world applications, due to some factors that are difficult to capture in the mathematical model, the solutions achieved by optimizing this model could be infeasible in practice. Therefore, providing multiple different Pareto optimal solutions (in the decision space) with the same quality (objective values) will give the decision maker more options for consideration.

See the example in the last subsection, Max gets solutions $\boldsymbol{x}_1^*, \boldsymbol{x}_2^*, \boldsymbol{x}_3^*, \boldsymbol{x}_5^*, \boldsymbol{x}_7^*, \boldsymbol{x}_9^*, \boldsymbol{x}_{10}^*$ by using an MOEA. He prefers solution $\boldsymbol{x}_5^*$ (IC_1, IC_2, and IC_5) since its objective values can perfectly satisfy his requirements. Unfortunately, his supplier suddenly informs him that IC_2 is in short supply because the cargo is missing. This accident is kind of uncertainty that is difficult to be captured in the optimization problem in Eq. (2). Then he cannot adopt solution $\boldsymbol{x}_5^*$ and is upset about that none of the remaining solutions can meet his requirements. He will be very happy if the optimizer can provide solution $\boldsymbol{x}_6^*$ (IC_3, IC_4, and IC_5), which has the same quality with solution $\boldsymbol{x}_5^*$ but does not need IC_2. Since in *a posteriori* situations, the decision maker's (e.g. Max's) preference is given after the optimization, it will be ideal if the optimizer can provide more solutions with other objective and decision values (e.g., $\boldsymbol{x}_4^*$ and $\boldsymbol{x}_8^*$).

Similar situations happen in other practical optimization problems. For instance, in architecture layout design optimization [7], there could be multiple optimal layouts (solutions) with the same building cost and energy efficiency (objectives). Since customers have different preferences about the artistic factors and these preferences are usually implicit [25], designers need to provide various optimal layouts for customer choices. For another example in a map-based problem [9], the goal is to find a location simultaneously closest to several places on a map, such as school, convenience store, and railway station. Since the aforementioned places are usually more than one on the map, there may exist several optimal locations that have the same objective values.

In addition, finding multiple Pareto optimal solutions may help to reveal hidden properties or relations of the problem under study, e.g., the distribution of Pareto optimal solutions in the decision space. This provides much richer information about the problem domain than regular MOEAs [20].

D. Motivation

Although solving MMMOP has a great significance in real-world applications, very few researches have been done up to this date. In the following, we elaborate our motivations in detail to address the issues in solving MMMOPs mentioned in Section I.

1) Diversity maintenance both in the objective and the decision spaces: The first issue is how to maintain diversities both in the objective and the decision spaces when searching for the Pareto optimal solutions.

A good diversity in the objective space means uniformity and wide spread. All the MOEAs mentioned in Subsection II.A are developed to achieve this goal. Similarly, the diversity metrics in the objective space are also designed to assess uniformity and/or spread. For example, Spacing (SP) [13] measures uniformity, Maximum Spread (MS) [26] measures spread, while IGD [12] and HV [16] measure both simultaneously.

There are also various diversity assessment methods in the decision space. The diversity metric in [27] is based on summing the distances from every point to the center-point. In [28], several diversity metrics were proposed based on ectropy. These metrics can be applied to guide the search in evolutionary algorithms to avoid premature convergence and to escape from local optima [29].

For (single-objective) multi-modal optimization, the goal is to find all the global and/or local optimal solutions. Maximum Peak Ratio (MPR) [30] is a very popular diversity metric. This metric measures not only the objective value of the solutions but also the number of optimal solutions. The more different optimal solutions found, the better value of MPR will be. That is, the diversity in the decision space for multi-modal optimization is measured by how many optimal solutions (or solutions very close to them) are found. The distribution of optimal solutions in the decision space is determined by the optimization problem. Thus they are not necessarily uniformly or widely distributed. Therefore, the diversity assessment methods such as those proposed in [27], [28] would be improper for multi-modal optimization.

From the above discussions, we can see that the diversities in the objective and the decision spaces bear different requirements. If a uniformly and widely distributed Pareto optimal solution set in the objective space is not uniformly or widely distributed in the decision space, or vice versa, we can say that the diversities in the objective and the decision spaces are inconsistent. In this situation, we should not maintain diversity only in the objective or the decision space.

There have been a few diversity maintenance methods for MMMOPs. DN-NSGA-II proposed in CEC 2016 [5] adopts a decision space based niching method in the mating selection. However, it does not simultaneously consider the diversities in the objective and the decision spaces. Omni-optimizer [4] calculates the crowding distances of each solution in both the objective and the decision spaces. For a solution, if either of its crowding distances is larger than the average value, the larger one will be assigned as its fitness value; otherwise, the smaller one will be. Very recently, a multi-objective particle swarm optimization algorithm using ring topology and special crowding distance (MO_Ring_PSO_SCD) [31] was proposed to handle MMMOPs. The special crowding distance is a modification of that in Omni-optimizer. The crowding distance in the objective space of a boundary solution in Omni-optimizer is infinity, which makes the average crowding distance of all solutions also infinity. Thus, the crowding distance in the objective space is unlikely to be larger than the average value, and the crowding distance in the decision space plays a more important role in selection. To address this issue, the crowding distance in the objective space of a boundary solution in MO_Ring_PSO_SCD is set to 1 and 0 for minimization and maximization problems, respectively. Both Omni-optimizer and MO_Ring_PSO_SCD try to promote uniformity and spread in the objective and the decision spaces. However, as we have discussed, uniformity and spread are not necessary for diversity maintenance in the decision space.

In this study, we first cluster solutions according to a set of well-distributed reference vectors to improve uniformity and spread in the objective space. Then, for each cluster, we employ a niching method to promote diversity in the decision space. In doing so, we expect to find different Pareto optimal solutions in the decision space for each cluster center (i.e., reference vector). We will describe this method in detail as the diversity archive method in Subsection III.D.

Another problem of diversity maintenance is the trade-off between convergence and diversity [32]. It will be difficult to design a selection criterion to well balance among convergence performance and diversity performances in the objective and the decision spaces. Focusing only on one performance may degenerate the others. In multi-objective optimization, two-archive methods have been developed to reduce the difficulty in selection [33]. For example, in many-objective optimization, since selecting good solutions in a high-dimensional objective space is a very challenging task, two archives were employed to focus on convergence and diversity, respectively [34].

Inspired by the works in [33], [34], we propose a novel two-archive method for solving MMMOPs in this study. In addition to the diversity archive, a convergence archive is developed focusing on convergence, which is described in Subsection III.C. These two archives complement with each other to make the optimizing work easier during the evolution process. In addition, they are also designed to obtain good solutions for the recombination strategy. Please see our explanations in the next subsection.

2) Reducing population size in particular situations: The other issue in solving MMMOPs is that maintaining diversities both in the objective and the decision spaces may require an extremely large population as well as a huge computational cost. In this study, we take advantage of independent convergence-related decision variables to develop a recombination strategy, which can reduce the population size during the evolution process.

Recently, the independent convergence-related decision variables have gained much attention in large-scale multi-objective evolutionary optimization [35], [36]. There could exist a number of independent convergence-related decision variables in MOPs (not necessarily large-scale) [37], [38]. In [35], [36], a decision variable analysis method is first adopted to classify the decision variables into convergence-related, diversity-related, or mixed type and to obtain the interactions among them. Then, the population is decomposed into several subpopulations to accelerate the convergence speed.

No matter whether a multi-objective problem is large-scale or multi-modal, we can apply the decision variable analysis method to obtain the property of solutions and the relationships among them. In this study, we exploit this information to reduce the population size when solving MMMOPs.

Let us first introduce what is an *independent convergence-related* decision variable. A decision variable is independent if it is not interacting with other decision variables [39]. Various interdependence detecting techniques for decision variables have been developed, e.g., perturbation [40] and model building, [41]. Please refer to the above literatures for detailed information.

We have the following definition if and only if all the decision variables $x_i \in \boldsymbol{x}, i = 1, ..., n$ are independent with each other [39]:

$$\arg \min_{(x_1,...,x_n)} \boldsymbol{f}(\boldsymbol{x}) = [\arg\min_{x_1} \boldsymbol{f}(\boldsymbol{x}), ..., \arg\min_{x_n} \boldsymbol{f}(\boldsymbol{x})] \quad (4)$$

The above definition suggests that each decision variable can be optimized separately, and optimal solutions can be finally obtained by combining the optimized values of the decision variables.

On the other hand, a decision variable x_i is convergence-related, if changing $x_i \in \boldsymbol{x}$ can only result in a new solution which equals to $\boldsymbol{x}$, dominates $\boldsymbol{x}$, or is dominated by $\boldsymbol{x}$ [38]. On the contrary, if changing $x_i \in \boldsymbol{x}$ can only cause a new solution that is incomparable or equivalent to $\boldsymbol{x}$, it is diversity-related. The decision variables which are both convergence- and diversity-related are called mixed variables. That is, when changing a mixed variable $x_i \in \boldsymbol{x}$, some new solutions are comparable with $\boldsymbol{x}$, while the others are incomparable. Please refer to [35], [36] for detailed information about how to classify these properties of decision variables.

From the above definitions, we can make the following inference. For an MOP in Eq. (1), assume that $\boldsymbol{x}_1^*$ and $\boldsymbol{x}_2^*$ are two Pareto optimal solutions and x_c is a independent convergence-related decision variable. If the decision variables except x_c in $\boldsymbol{x}_1^*$ and $\boldsymbol{x}_2^*$ have the same values, $\boldsymbol{x}_1^*$ and $\boldsymbol{x}_2^*$ must be at the same location on the PF, since different values of x_c will not lead to the change of location on the PF. Then, since their values of x_c are different, the MOP is an MMMOP according to the definition in Subsection II.C. For a Pareto optimal solution $\boldsymbol{x}_3^*$ at the another location on the PF, if the value of x_c in $\boldsymbol{x}_3^*$ is different from that in $\boldsymbol{x}_1^*(\boldsymbol{x}_2^*)$, a new solution $\boldsymbol{x}_4^*$ can be obtained by replacing the value of x_c in $\boldsymbol{x}_3^*$ with that in $\boldsymbol{x}_1^*(\boldsymbol{x}_2^*)$. Since x_c is independent with other decision variables, $\boldsymbol{x}_4^*$ must be a Pareto optimal solution according to the definition in Eq. (4). Meanwhile, $\boldsymbol{x}_4^*$ is at the same location on the PF as $\boldsymbol{x}_3^*$. This means that once we can get several Pareto optimal solutions at the same location like $\boldsymbol{x}_1^*$ and $\boldsymbol{x}_2^*$ and some Pareto optimal solutions at the other locations like $\boldsymbol{x}_3^*$, we can easily generate a large number of Pareto optimal solutions like $\boldsymbol{x}_4^*$ by the above recombination operation. Taking a simple example for an easy understanding, a given MMMOP is formulated as follows:

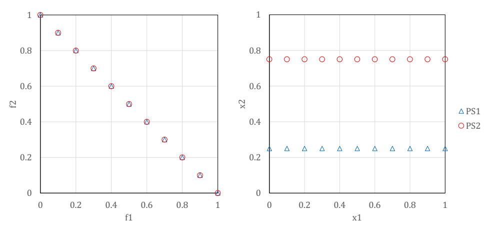


Fig. 2. The PF of problem in Eq. (5) in the objective space and PS_1 and PS_2 in the decision space.

$$\begin{aligned} &\min f_1(\boldsymbol{x}) = x_1 + 1 - |sin(2\pi x_2)| \\ &\min f_2(\boldsymbol{x}) = 1 - x_1 + 1 - |sin(2\pi x_2)| \\ &\text{s.t. } x_1 \in [0,1], x_2 \in [0,1] \end{aligned} \quad (5)$$

Clearly, x_1 is diversity-related, x_2 is convergence-related, and they are independent with each other. Assume we have obtained a PS, PS_1, that is well-distributed on the PF, where $PS_1 = \{$ $(0, 0.25)$, $(0.1, 0.25)$, $(0.2, 0.25)$, $(0.3, 0.25)$, $(0.4, 0.25)$, $(0.5, 0.25)$, $(0.6, 0.25)$, $(0.7, 0.25)$, $(0.8, 0.25)$, $(0.9, 0.25)$, $(1, 0.25)\}$, and we also find that $(0, 0.25)$ and $(0, 0.75)$ are two Pareto optimal solutions at the same location on the PF. By replacing the values of x_2 with 0.75 in PS_1, we then achieve a new PS, PS_2, where $PS_2 = \{$ $(0, 0.75)$, $(0.1, 0.75)$, $(0.2, 0.75)$, $(0.3, 0.75)$, $(0.4, 0.75)$, $(0.5, 0.75)$, $(0.6, 0.75)$, $(0.7, 0.75)$, $(0.8, 0.75)$, $(0.9, 0.75)$, $(1, 0.75)\}$. Fig. 2 shows PS_1 and PS_2 in the decision space and the corresponding PF in the objective space. Pareto optimal solutions (or solutions very close to them) which have different independent convergence-related decision variable values are defined as peak solutions, e.g., $(0, 0.25)$ and $(0, 0.75)$. They play an important role in the recombination strategy. We will describe how the recombination strategy works in Subsection III.E.

Although integrating the above recombination strategy with existing MOEAs is possible, existing MOEAs may not be able to find peak solutions efficiently. Therefore, in this paper, the aforementioned two-archive strategy is particularly designed to work with the recombination strategy. Besides accelerating convergence rate, the convergence archive is also designed to find peak solutions. Meanwhile, the diversity archive aims at maintaining diversities in the objective space and the remainder decision subspace.

For the example in Eq. (5), the convergence archive is expected to find two peak solutions, $(0, 0.25)$ and $(0, 0.75)$, while the diversity archive is expected to find PS_1 (it contains eleven solutions). In an ideal situation, their sizes can be set to 2 and 11, respectively, during the evolution process. Then, we can get the whole PS (e.g., $PS_1 \cup PS_2$) using the recombination strategy at the end of the evolution process. In this way, we may only need a population which has a similar size to the archives, but obtain a large number of Pareto optimal solutions at last. Similarly, for the example in Section I, if the 20 Pareto optimal solutions for each point on the PF are peak solutions, we can set the sizes of the convergence and the diversity archives to 20 and 100, respectively, and then obtain $20 \times 100 = 2,000$ Pareto optimal solutions by the recombination strategy. Therefore, if the independent convergence-related decision variables can be detected, using our proposed recombination strategy will dramatically reduce the population size during the evolution process. Some MMMOPs may not contain independent convergence-related decision variables. However, if an MMMOP does have such decision variables, it will gain great convergence and/or computational benefits by utilizing the recombination strategy. Please refer to Section III in the Supplementary Material (SM) for the investigation on the effect of the recombination strategy.

III. The Proposed Method

A. A General Framework

Algorithm 1 A General Framework

Require: P (Population), N (Population Size), A_C (Convergence Archive), N_C (Size of A_C), A_D (Diversity Archive), N_D (Size of A_D)

1: $[X_{IC}$,$X_{re}]$=*Decision_Variable_Analysis*$(\boldsymbol{x})$
2: R = *Reference_Vector_Generation*(R)
3: P = *Initialize*(P)
4: A_C =*Update_Convergence_Archive*(A_C, P, N_C, X_{IC})
5: A_D = *Update_Diversity_Archive*(A_D, P, R, N_D, X_{re})
6: **while** the stopping criterion is not met **do**
7: $\quad P'$ = *Mating_selection*(A_C, A_D, p_{con})
8: $\quad P$ = *Reproduction*(P')
9: $\quad A_C$ =*Update_Convergence_Archive*(A_C, P, N_C, X_{IC})
10: $\quad A_D$ = *Update_Diversity_Archive*(A_D, P, R, N_D, X_{re})
11: **end while**
12: **if** $X_{IC} \neq \emptyset$ **then**
13: $\quad FS$ = *Recombination*(A_C, A_D, X_{IC})
14: **else**
15: $\quad FS = A_D$
16: **end if**
17: **return** FS

Algorithm 1 presents the overall framework of the proposed TriMOEA-TA&R for solving MMMOPs. Note that for a concise description, the MMMOP to be solved is assumed to be minimized from here. In addition, the objectives and decision variables are assumed to have the same scale. If not, they can be normalized according to their lower and upper bounds.

In Algorithm 1, first, a decision variable analytical technique is adopted to detect the set of independent convergence-related decision variables, X_{IC} (line 1). Then, a set of reference vectors well-scattered in the objective space, R, is generated to guide the update of the diversity archive (line 2). This procedure is similar with that of the reference vector based methods, such as MOEA/D [3]. Next, an initial population, P, is created by randomly generating N individuals (line 3). In each generation, the mating selection is performed to choose parents from the convergence and the diversity archives (A_C and A_D) (line 7), where p_{con} is the probability of selecting parents from the convergence archive. A larger value of p_{con} indicates that the convergence archive plays a more important role in the evolution process. Please refer to Subsection IV.B

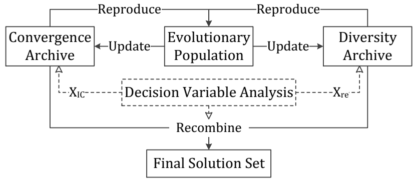


Fig. 3. The relationships among the evolutionary population, the convergence archive, the diversity archive and the final solution set in TriMOEA-TA&R.

and IV.C in the SM for the investigation on the effect of p_{con}. A tournament selection strategy is adopted in the mating selection, where solutions with lower rank are favored. The rank in the convergence and the diversity archives is calculated as that in line 18, Algorithm 3 and line 1, Algorithm 4, respectively. Then, offspring are created to form the new population (line 8). Once a new population is produced, both A_C and A_D will be updated based on it (lines 4, 5, 9, and 10). At the end of the evolution, if X_{IC} is not empty, the final solution set (FS) is generated by recombining solutions in A_C and A_D (lines 12 and 13). Otherwise, FS is equal to A_D (line 15). Fig. 3 shows the relationships among the evolutionary population, the convergence archive, the diversity archive and the final solution set for a better understanding.

In the following, we describe in detail the decision variable analytical technique, the approaches to updating the convergence and the diversity archives, and the recombination strategy to obtain the final solution set, which are four important components in TriMOEA-TA&R.

B. Decision Variable Analysis

Algorithm 2 *Decision_Variable_Analysis*($\boldsymbol{x}$)

1: *Convergence&Diversity_Analysis*($\boldsymbol{x}$)
2: *Interdependence_Analysis*($\boldsymbol{x}$)
3: $X_{IC} = \emptyset$
4: $X_{re} = \emptyset$
5: **for all** $x_i \in \boldsymbol{x}$ **do**
6:     **if** x_i is convergence-related **then**
7:         $X_{IC} = X_{IC} \cup \{x_i\}$
8:     **else**
9:         $X_{re} = X_{re} \cup \{x_i\}$
10:     **end if**
11: **end for**
12: **while** $\exists x_i \in X_{IC}, x_j \in X_{re}$, x_i and x_j are interactive **do**
13:     $X_{re} = X_{re} \cup \{x_i\}$
14:     $X_{IC} = X_{IC}/\{x_i\}$
15: **end while**
16: **return** X_{IC}, X_{re}

Algorithm 2 describes the decision variable analytical technique used. Function *Convergence&Diversity_Analysis*($\boldsymbol{x}$) classifies the decision variables into convergence-related, diversity-related, or mixed variables (line 1). Function *Interdependence_Analysis*($\boldsymbol{x}$) analyzes the interdependence relationship between every two decision variables (line 2). Both functions are similar with those of [35]. Please refer [35] for detailed discussions. After getting the properties of the decision variables and the relationships among them, the convergence-related decision variables will be selected into X_{IC}, and the others into X_{re} (lines 5-10). Then, if there exists any decision variable in X_{IC} that is interactive with any in X_{re}, it will be moved into X_{re} (lines 12-15). That is, any decision variable in X_{IC} is independent with every decision variable in X_{re}, and vice versa. Based on the definition in Eq. (4), we can deduct that X_{IC} and X_{re} can be optimized separately. This implies that for a Pareto optimal solution, we can replace its decision variables in X_{IC} (or X_{re}) with those of another Pareto optimal solution, and then obtain a new Pareto optimal solution. We take advantage of this in the recombination strategy in Subsection III.E.

C. Updating the Convergence Archive

Algorithm 3 *Update_Convergence_Archive*(A_C, P, N_C, X_{IC})

Require: σ_{niche} (niche radius in the variable space)
1: $Q = A_C \cup P$
2: **for all** $\boldsymbol{x} \in Q$ **do**
3:     $f_S(\boldsymbol{x}) = \frac{\sum_{m=1}^{M} f_m(\boldsymbol{x})}{M}$
4: **end for**
5: **for** every two solutions $\boldsymbol{x}_i, \boldsymbol{x}_j \in Q$ **do**
6:     $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$ = *Variable_Distance* $(\boldsymbol{x}_i, \boldsymbol{x}_j, X_{IC})$
7: **end for**
8: $Q' = \emptyset$
9: $A_C = \emptyset$
10: $nrank = 1$
11: **while** $|A_C| < N_C$ **do**
12:     **if** $Q = \emptyset$ **then**
13:         $Q = Q'$
14:         $Q' = \emptyset$
15:         $nrank = nrank + 1;$
16:     **end if**
17:     $\boldsymbol{x}_{\min} = \arg\min_{\boldsymbol{x} \in Q} f_S(\boldsymbol{x})$
18:     $rank(\boldsymbol{x}_{\min}) = nrank$
19:     $A_C = A_C \cup \{\boldsymbol{x}_{\min}\}$
20:     $Q = Q/\{\boldsymbol{x}_{\min}\}$
21:     **for all** $\boldsymbol{x}_i \in Q$ **do**
22:         **if** $d(\boldsymbol{x}_{\min}, \boldsymbol{x}_i) < \sigma_{niche}$ **then**
23:             $Q' = Q' \cup \{\boldsymbol{x}_i\}$
24:             $Q = Q/\{\boldsymbol{x}_i\}$
25:         **end if**
26:     **end for**
27: **end while**
28: **return** A_C

Algorithm 3 gives the approach to updating the convergence archive. This procedure is similar to our previous proposed one-by-one selection strategy [42]. The difference is that the procedure in this study aims at finding multiple Pareto optimal solutions in the independent convergence-related decision subspace. Our previous study in [42] has shown that the procedure can well balance the convergence and the diversity performances. Updating the convergence archive in this study can be divide into two main steps.

Main Step 1. Selecting only one solution with the best value of the convergence indicator (lines 17-20). That is, the solution, $\boldsymbol{x}_{\min}$, with the minimum convergence indicator ($f_S(\boldsymbol{x})$) is selected from the candidate solution set, Q.

Main Step 2. De-emphasizing solutions closer to the one selected in the first main step in the independent convergence-related decision subspace (lines 21-26). In this main step, the solutions in Q whose distances to $\boldsymbol{x}_{\min}$ are smaller the threshold, σ_{niche}, are de-emphasized and selected into Q'.

The first main step focuses on providing a strong selection pressure towards the PF using the convergence indicator, while the second main step aims at improving the diversity in the independent convergence-related decision subspace. By repeating the above two steps, the convergence archive is able to find different Pareto optimal solutions (or solutions very close to them) in the independent convergence-related decision subspace. That is, finding multiple Pareto optimal solutions at the same position on the PF in the convergence archive is very promising. These solutions are defined as peak solutions, which are used for the recombination strategy.

In lines 2-4, the convergence indicator of each candidate solution, $f_S(\boldsymbol{x})$, in Q is calculated by summing the objective values. Note that any other scalarizing functions can also be adopted as the convergence indicator. The selection pressure towards PF is boosted by the convergence indicator instead of the Pareto dominance even if there are a large number of objectives.

In lines 5-7, for every two solutions in Q, their distance in the independent convergence-related decision subspace is calculated. Any method to calculate the distance can be used. Here we just simply adopt the Tchebycheff distance in this study. In the second main step, if the distance between any solution and the selected one is smaller than σ_{niche}, it will be deleted from Q and selected into Q' (lines 22-25).

In lines 12-19, once Q is empty, it is reloaded with the de-emphasized solutions in Q', and the ranks of solutions to be selected is increased by 1. Consequently, a solution that is de-emphasized for more times has a larger rank. The ranks are used for selecting solutions with good convergence performance in the mating selection in Algorithm 1.

The guideline of setting σ_{niche} in the convergence archive (also the diversity one) is to effectively distinguish different Pareto optimal solutions. Generally, the higher dimension of the decision space and the smaller population size, the larger value of σ_{niche} will be. If σ_{niche} is too large, some Pareto optimal solutions very close to others may be de-emphasized. Conversely, if σ_{niche} is too small, a large number of solution close to the optimal ones may be preserved in the archive, which will decrease the ability in finding the true Pareto optimal solutions. We show an example of how σ_{niche} effects on finding peak solutions and the ranks of solutions in Subsection IV.A and IV.D in the SM.

Besides, we show the effect of the convergence archive on converging in Subsection IV.C in the SM.

D. Updating the Diversity Archive

The purpose of the diversity archive is to maintain diversity in the objective and the remainder decision subspace. Algorithm 4 presents the approach in updating the diversity archive, which has four main steps.

Algorithm 4 *Update_Diversity_Archive*(A_D, P, R, N_D, X_{re})

Require: σ_{niche} (niche size in the variable space)

1: $F = F_1 \cup F_2 \cup ...F_i$ =*Nondominated_rank* $(A_D \cup P)$
2: $A_D = F_1 \cup F_2 \cup ...F_{i-1}$
3: $Q = F_i$
4: $N_Q = N_D - |F_1 \cup F_2 \cup ...F_{i-1}|$
5: **for all** $\boldsymbol{x}_i \in Q, r_j \in R$ **do**
6: $\quad \theta(\boldsymbol{x}_i, \boldsymbol{r}_j)$ = *Angle* $(\boldsymbol{x}_i, \boldsymbol{r}_j)$
7: **end for**
8: **for** every two solutions $\boldsymbol{x}_i, \boldsymbol{x}_j \in Q$ **do**
9: $\quad d(\boldsymbol{x}_i, \boldsymbol{x}_j)$ = *Variable_Distance* $(\boldsymbol{x}_i, \boldsymbol{x}_j, X_{re})$
10: **end for**
11: $C_1 = C_2 = ...C_{|R|} = \emptyset$
12: $C'_1 = C'_2 = ...C'_{|R|} = \emptyset$
13: **for all** $\boldsymbol{x}_i \in Q$ **do**
14: $\quad jj = \arg \min_{j=1,2,...,|R|} \theta(\boldsymbol{x}_i, \boldsymbol{r}_j)$
15: $\quad$ **if** $\exists \boldsymbol{x}_{ii} \in C_{jj}, d(\boldsymbol{x}_i, \boldsymbol{x}_{ii}) < \sigma_{niche}$ **then**
16: $\quad\quad C'_{jj} = C'_{jj} \cup \{\boldsymbol{x}_i\}$
17: $\quad$ **else**
18: $\quad\quad C_{jj} = C_{jj} \cup \{\boldsymbol{x}_i\}$
19: $\quad$ **end if**
20: **end for**
21: **while** $|C_1 \cup C_2 \cup ...C_{|R|}| > N_Q$ **do**
22: $\quad c_{\max} = \max_{j=1,2,...,|R|} |C_j|$
23: $\quad \boldsymbol{x}_{\max} = \arg \max_{\boldsymbol{x}_i \in C_j, |C_j| = c_{\max}} \theta(\boldsymbol{x}_i, \boldsymbol{r}_j)$
24: $\quad C_j = C_j / \{\boldsymbol{x}_{\max}\}$
25: **end while**
26: **while** $|C_1 \cup C_2 \cup ...C_{|R|}| < N_Q$ **do**
27: $\quad c_{\min} = \min_{C'_j \neq \emptyset, j=1,2,...,|R|} |C_j|$
28: $\quad \boldsymbol{x}_{\min} = \arg \min_{\boldsymbol{x}_i \in C'_j, |C_j| = c_{\min}} \theta(\boldsymbol{x}_i, \boldsymbol{r}_j)$
29: $\quad C_j = C_j \cup \{\boldsymbol{x}_{\min}\}$
30: **end while**
31: $A_D = A_D \cup C_1 \cup C_2 \cup ...C_{|R|}$
32: **return** A_D

Main Step 1. De-emphasizing the dominated solutions (lines 1-4). The nondominated ranking [1] is performed on the set combined by A_D and P to form several nondominated fronts (line 1), where i in F_i is the minimal value such that $|F_1|+|F_2|+...|F_i| > N_D$. Note that the result of nondominated ranking is also used for the mating selection in Algorithm 1. The first $i-1$ nondominated fronts are combined to form A_D (line 2). Then, the candidate solutions set, Q, is set to F_i (line 3), and the number of solutions that remain to be selected, N_Q, is set to $N_D - |F_1 \cup F_2 \cup ...F_{i-1}|$ (line 4).

Main Step 2. Clustering the candidate solutions into a set of well-distributed reference vectors (i.e. clustering centers) in the objective space (lines 11-20). That is, the solutions in Q are clustered into $C_j, j = 1, ..., |R|$ according to their angels (calculated in lines 5-7) to the closest reference vector, $\boldsymbol{r}_j, j = 1, ..., |R|$. The purpose of this main step is to improve the diversity in the objective space.

Main Step 3. De-emphasizing similar solutions in the remainder decision subspace (lines 15-16). When clustering solutions to each reference vector in the second main step, after finding the closest $\boldsymbol{r}_{jj}$ for the solution, $\boldsymbol{x}_i$, if there exists any solution in C_{jj} whose distance to $\boldsymbol{x}_i$ (calculated in lines 8-10) is smaller than σ_{niche}, $\boldsymbol{x}_i$ will be de-emphasized and chosen into C'_{jj}. This niche-based clearing procedure is similar to the second main step in the convergence archive. The design goal of this main step is to improve diversity in the remainder decision subspace.

Main Step 4. Making the number of selected solutions equal to N_Q (lines 21-30). If the total number of solutions in $C_j, j = 1, ..., |R|$ is larger than N_Q, the solution farthest away from its cluster center ($\boldsymbol{r}_j$) among the largest C_j will be discarded (lines 21-25). Otherwise, if the total number of solutions in $C_j, j = 1, ..., |R|$ is smaller than N_Q, the solution closest to its cluster center ($\boldsymbol{r}_j$) in C'_j will be added into the smallest C_j (lines 26-30). This main step is designed to further promote the diversity performance in the objective space.

According to Algorithm 4, the diversity in the objective space is ensured by the clustering solutions in terms of the reference vectors. Meanwhile, the diversity in the decision space is promoted using the niche-based clearing procedure. Therefore, the diversity archive has the ability in maintaining diversities both in the objective and the decision spaces.

E. Obtaining Final Solution Set

Algorithm 5 *Recombination*(A_C, A_D, X_{IC})

Require: ϵ_{peak} (accuracy level)
1: $\boldsymbol{x}_{\min} = \arg\min_{\boldsymbol{x} \in A_C} f_S(\boldsymbol{x})$
2: $A_{peak} = \{\boldsymbol{x} : f_S(\boldsymbol{x}) - f_S(\boldsymbol{x}_{\min}) < \epsilon_{peak}, \boldsymbol{x}$ is in the first rank in $A_C\}$
3: $FS = \emptyset$
4: **for all** $\boldsymbol{x}_i \in A_{peak}, i = 1, ..., |A_{peak}|$ **do**
5:     **for all** $\boldsymbol{x}_j \in A_D, j = 1, ..., N_D$ **do**
6:         $\boldsymbol{x}_l = 0^n$
7:         **for** $k = 1$ to n **do**
8:             **if** $x_k \in X_{IC}$ **then**
9:                 $x_{k,l} = x_{k,i}$
10:             **else**
11:                 $x_{k,l} = x_{k,j}$
12:             **end if**
13:         **end for**
14:         $FS = FS \cup \{\boldsymbol{x}_l\}$
15:     **end for**
16: **end for**
17: **return** FS

At the end of the evolution, if X_{IC} is empty, the final solution set is equal to the diversity archive. Otherwise, a recombination strategy is performed to generate the final solution set, which is presented in Algorithm 5. We define the Pareto optimal solutions (or solutions very close to them) which have different independent convergence-related decision variable values as peak solutions in this study. In Algorithm 5, an accuracy level [43], ϵ_{peak}, is required to distinguish the peak solutions in A_C. First, $\boldsymbol{x}_{\min}$ is set to the solution with the smallest value of convergence indicator in A_C (line 1). Then, if the difference of the convergence indicator between a solution in the first rank in A_C and $\boldsymbol{x}_{\min}$ is smaller than the accuracy level, it will be regarded as a peak solution and placed into A_{peak} (line 2). Next, for each solution $\boldsymbol{x}_i$ in A_{peak} and each solution $\boldsymbol{x}_j$ in A_D, a new solution $\boldsymbol{x}_l$ is created and added into P (lines 4-16). When creating $\boldsymbol{x}_l$, if the k-th decision variable belongs to X_{IC}, $x_{k,l}$ will be set to $x_{k,i}$ (lines 8-9). Otherwise, it will be set to $x_{k,j}$ (line 11). As aforementioned in Subsection III.B, the decision variables in X_{IC} and X_{re} can be optimized separately, which suggests that if both $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are Pareto optimal solutions, $\boldsymbol{x}_l$ is certainly a Pareto optimal solution. Using this recombination strategy, a solution set with satisfactory convergence and diversities both in the objective and the decision spaces can be obtained. In addition, the size of the final solution set is $|A_{peak}| \times N_D$, whereas only $N_A + N_D$ solutions are required to be maintained during the evolution. This implies that if the given MMMOP has a lot of peak solutions, TriMOEA-TA&R will have a significant advantage in computational efficiency due to the small population size. Please also refer to the computational complexity analysis of TriMOEA-TA&R in Section VI in the SM.

IV. Experimental Design

This section describes the experimental design for examining the performance of the proposed TriMOEA-TA&R. The test problems and the performance metric used in the experiments are given at first. Then, the parameters are set for the comparative studies of the test problems and the competing algorithms.

A. Test Problems

Since multi-modal multi-objective optimization is a relatively new research area, to our best knowledge, either benchmark test functions or mathematical models from real world applications are very few. In [4], [5], [31], several multi-modal multi-objective test problems were proposed. However, they are non-scalable and some of them are relatively easy for a well-developed optimizer.

This subsection introduces the novel multi-modal multi-objective benchmark test functions, i.e., MMMOP1-6, proposed in this paper. These benchmarks are developed from the most popular multi-modal (CEC 2013 [43], CEC 2015 [44]) and multi-objective (CEC2009 [45], DLTZ [37]) benchmarks. All these test functions are scalable. For these test functions, the following notations are used. M is the number of objectives. n is the number of decision variables. $X_A = \{x_M, x_{M+1}, ..., x_{M+k_A-1}\}$, and $|X_A| = k_A$. $X_B = \{x_{M+k_A}, x_{M+k_A+1}, ..., x_n\}$, and $|X_A| = k_B$. $n = M - 1 + k_A + k_B$. Please refer to Tables I and II in the SM for the common properties and the detailed definitions of MMMOP1-6, respectively. Figs. 1-7 in the SM show the PFs and PSs of these problems.

The characteristics of these test functions are summarized as follows.

1) MMMOP1: MMMOP1 is a relatively simple optimization problem to test an optimizer's ability in finding multiple Pareto optimal solutions. It is developed from "Equal Maxima" and "DLTZ1." DTLZ1 has numerous local Pareto optimal solutions. However, in this study, we only consider solving an MMMOP by finding multiple global Pareto optimal solutions for a point on the PF. Therefore, Equal Maxima is combined with DLTZ1 to construct MMMOP1. It has 5^{k_A} Pareto optimal solutions which have different independent convergence-related decison variable values for each point on its PF.

2) MMMOP2: MMMOP2 is originated from "Vincent" and "DLTZ4." It has two distinct features. The first is that it has bias in the decision space. Therefore, it is difficult for an optimizer in maintaining diversities both in the objective and the decision spaces. The second is that the Pareto optimal solutions have vastly different spacing between them. This further increases the difficulty in preserving all of them. MMMOP2 has 6^{k_A} Pareto optimal solutions which have different independent convergence-related decison variable values for each point on its PF.

3) MMMOP3: The building blocks of MMMOP3 are from "Rastrigin" and "DLTZ2." Different from MMMOP1 and MMMOP2, there are multiple Pareto optimal solutions which have different diversity- and/or convergence-related decision variable values for a point on the PF of MMMOP3. Another feature of MMMOP3 is that the number of Pareto optimal solutions for a point on its PF can be controlled by the user, which is $\prod_{i=M,...,M+k_A-1} c_i \times \prod_{i=1,...,M-1} d_i$. $d_i > 0$ and $c_i > 1$ are the parameters defined by the user.

4) MMMOP4: MMMOP4 is similar with MMMOP3. However, the significant feature of MMMOP4 is that the points on the Pareto front correspond to various numbers of Pareto optimal solutions. In addition, MMMOP4 has numerous local optima by incorporating the building blocks of "DLTZ3."

5) MMMOP5: MMMOP5 is also similar with MMMOP3. The main difference between MMMOP5 and MMMOP3 is that the Pareto opitmal regions in MMMOP5 have different densities. In addition, like MMMOP4, MMMOP5 also has numerous local optima.

6) MMMOP6: MMMOP6 is developed from "Himmelblau" and "UF8." It has convergence-related and mixed decision variables, where some convergence-related decision variables are interactive with the mixed ones. In addition, for every two decision variables in X_A, there are four Pareto opitmal solutions for a point on the PF, where two solutions are much closer to each other than the other two. In MMMOP6, there exist $4^{k_A/2} \times \prod_{i=M+k_A,...,n} c_i$ Pareto optimal solutions for each point on the Pareto front, where $c_i > 0$ is a design parameter.

In addition, we also include MMF1-8 in [31] as test problems in the experiments.

B. Performance Metric

Like the test problems, building a proper metric to evaluate the performance of an optimizer is another issue in multi-modal multi-objective optimization. In [5], IGDX [46] is suggested to assess the performance of a multi-modal multi-objective optimizer. IGDX is a variant of IGD, where X symbolizes decision variables. IGDX uses a set of uniformly-distributed reference points on the PS in the decision space instead of that on the PF in the objective space in IGD. It also calculates the distance between the reference point set and the solution set in the decision space other than that in the objective space. However, as discussed in Subsection II.D, maintaining diversities in the objective and the decision spaces are not the same task. For the optimization problems such as MMMOP2 and MMMOP5, finding solutions well-distributed in the decision space does not imply that a well-distributed PF would be obtained. Moreover, a solution closer to the PS than others is not necessarily closer to the PF than others. Consequently, IGDX is not a very good metric for multi-modal multi-objective optimization. IGD is also not applicable since it cannot measure diversity in the decision space. Recently, the Pareto Sets Proximity (PSP) is proposed in [31]. It is composed of IGDX and Cover Rate (CR). CR can only measure the spread of a solution set in the decision space. Therefore, PSP still has the same issue as IGDX.

In this study, we propose a novel metric, Inverted Generational Distance-Multi-modal (IGDM), to assess the performance of a multi-modal multi-objective optimizer. IGDM functions like combining IGD with MPR. It can measure not only the convergence performance but also the diversity performances both in the objective and the decision spaces. A solution set that is more uniformly and widely distributed in the objective space would have a smaller IGDM value. In addition, for a given PF, the more diversified Pareto optimal solutions (or solutions very close to them) found in the decision space, the smaller IGDM value will be.

Let $F^* : \{\boldsymbol{f}_1^*, \boldsymbol{f}_2^*, ..., \boldsymbol{f}_q^*\}$ be the set of well distributed reference points (in the objective space) on the PF, where q is the number of reference points. Let A be $\{a_1, a_2, ..., a_q\}$, where $a_i, i = 1, 2, ..., q$ is the number of Pareto optimal solutions to each point, $\boldsymbol{f}_i^*$. Let $X^* : \{\boldsymbol{x}_{1,1}^*, \boldsymbol{x}_{1,2}^*, \boldsymbol{x}_{1,a_1}^*, \boldsymbol{x}_{2,1}^*, ..., \boldsymbol{x}_{q,a_q}^*\}$ be the set of Pareto optimal solutions (in the decision space) corresponding to F^* and A. Let $P : \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_p\}$ be the approximate solution set obtained by an optimizer, where p is the number of the obtained solutions. Metric IGDM is defined as follows:

$$
\begin{aligned}
&IGDM(P, F^*, X^*) = \frac{\sum_{\boldsymbol{f}_i^* \in F^*} \sum_{j=1,...,a_i} d(\boldsymbol{f}_i^*, \boldsymbol{x}_{i,j}^*, P)}{|X^*|} \\
&d(\boldsymbol{f}_i^*, \boldsymbol{x}_{i,j}^*, P) = \begin{cases} d_{\max}, \text{if } P_{i,j} = \emptyset \\ \min\{d_{\max}, ed(\boldsymbol{f}_i^*, P_{i,j})\}, \text{else} \end{cases} \\
&P_{i,j} = \{\boldsymbol{x}_k : j = \arg\min_{l=1,...,a_i} ed(\boldsymbol{x}_{i,l}^*, \boldsymbol{x}_k), \boldsymbol{x}_k \in P\}
\end{aligned} \tag{6}
$$

where $ed(\boldsymbol{f}_i^*, P_{i,j})$ is the minimal Euclidean distance between $\boldsymbol{f}_i^*$ and $P_{i,j}$, $ed(\boldsymbol{x}_{i,l}^*, \boldsymbol{x}_k)$ is the Euclidean distance between $\boldsymbol{x}_{i,l}^*$ and $\boldsymbol{x}_k$, and $d_{\max}$ is a parameter defined by the user.

The relations and distinctions between IGDM and IGD are as follows:

(1) In addition to a reference set of uniformly distributed points, F^*, along the PF in the objective space, IGDM has another reference set, X^*, which contains Pareto optimal solutions in the decision space corresponding to each points in F^*.

(2) In IGDM, $d(\boldsymbol{f}_i^*, \boldsymbol{x}_{i,j}^*, P)$, i.e., the distance between a point, $\boldsymbol{f}_i^*$, and the approximate solution set, P, is calculated

multiple times according to the number of the Pareto optimal solutions, i.e., a_i, while the distance is calculated only once in IGD.

(3) In IGDM, when calculating $d(\boldsymbol{f}_i^*, \boldsymbol{x}_{i,j}^*, P)$, the solutions in P is first clustered into several subset based on their distances to the reference solutions in X^* in the decision space. Only solutions in the corresponding subset, i.e., $P_{i,j}$, are adopted to calculated $d(\boldsymbol{f}_i^*, \boldsymbol{x}_{i,j}^*, P)$.

(4) IGDM has a new parameter, $d_{\max}$. $d_{\max}$ is a penalty value set by the user. When calculating $d(\boldsymbol{f}_i^*, \boldsymbol{x}_{i,j}^*, P)$, if the corresponding subset is empty, $d(\boldsymbol{f}_i^*, \boldsymbol{x}_{i,j}^*, P)$ is set to $d_{\max}$. In addition, if the corresponding subset is not empty, but $ed(\boldsymbol{f}_i^*, P_{i,j})$ is larger than $d_{\max}$, it is also set to $d_{\max}$.

(5) IGDM is an extension of IGD. Conversely, IGD is a special case of IGDM. When all $a_i = 1, i = 1, ..., q$ and $d_{\max} = \infty$, IGDM is equal to IGD. This relationship is similar with that between MMMOPs and MOPs, where MMMOPs are MOPs extended with multi-modal properties.

Please note the following points when using IGDM.

The first is that setting $d_{\max}$ to a proper value is non-trivial. If $d_{\max}$ is too large, the performance of an optimizer will be penalized too much for any Pareto optimal solution not found. On the other hand, if $d_{\max}$ is too small, it will be difficult to distinguish the differences among optimizers. According to our preliminary investigations, for the optimization problems whose PF is in the range of $[0, 1]$, $d_{\max}$ is recommended to be set as 1. For other optimization problems, their PFs and the approximate solution set can be normalized into the range of $[0, 1]$. Under this setting, the value of IGDM is in the range of $[0, 1]$. A small value of IGDM indicates a good overall performance of the approximate solution set. Particularly, IGDM $= 0$ suggests that the approximate solution set is uniformly distributed on the PF, and all the Pareto solutions are found. IGDM $= 1$ implies that the approximate solution set has a poor performance, and the distance between any solution in it and the PF is larger than 1. The user would have less interest in comparing two approximate solution sets whose IGDM $= 1$, since they are not even converged.

Secondly, IGDM is a composite metric to simultaneously measure the qualities of an approximate solution set in the following three aspects: convergence, diversity in the objective space, and diversity in the decision space. This implies that one approximate solution set that is good in one quality but poor in the others may have the same IGDM value as another approximate solution set that is poor in the particular quality but good in the others. This is similar to metric HV [16] that multiple approximate solution sets could result into the same, or similar, HV value given different convergence and diversity qualities in the objective space. Therefore, if the user expects to measure only one or two qualities of an approximate solution set, GD [12], SP [13], and other metrics are recommended. However, if the user wants to comprehensively quantify an approximate solution set, IGDM is a good choice.

At last, the value of IGDM depends on the reference points sampled on the PF and the PS (i.e., F^* and X^*), since different choices of reference points generally result in different IGDM values of an approximate solution set. It may be difficult to use IGDM for some real-world MMMOPs due to their unknown or complex true PFs and/or PSs. That is, IGDM has the same issue as IGD and IGDX. One possible way to address this issue is to adopt non-dominated solutions found so far as the reference points. In this study, all the PFs and PSs of the test problems are known. We set the size of F^* to $1,000$ and $1,653$ for 2- and 3-objective test problems in the experiments, respectively, while the size of X^* is dependent on the test problems.

TABLE I
THE PARAMETER SETTINGS OF THE TEST PROBLEMS AND THE CORRESPONDING PARAMETER SETTINGS OF THE COMPARED ALGORITHMS.

	M	n	k_A	k_B	All c_i	All d_i	N	N_{oth}
MMMOP1-A	2	3	1	1	\	\	100	500
MMMOP1-B	3	7	1	4	\	\	120	600
MMMOP2-A	2	3	1	1	\	\	100	600
MMMOP2-B	3	7	1	4	\	\	120	720
MMMOP3-A	2	2	0	1	\	3	300	300
MMMOP3-B	3	7	0	5	\	2	480	480
MMMOP3-C	2	6	1	4	3	3	300	900
MMMOP3-D	3	7	1	4	2	2	480	960
MMMOP4-A	2	2	0	1	\	4	300	300
MMMOP4-B	3	7	0	5	\	3	480	480
MMMOP4-C	2	6	1	4	2	4	300	600
MMMOP4-D	3	7	1	4	2	3	480	960
MMMOP5-A	2	2	0	1	\	3	400	400
MMMOP5-B	3	7	0	5	\	1	480	480
MMMOP5-C	2	6	1	4	2	2	300	600
MMMOP5-D	3	7	1	4	2	1	480	960
MMMOP6-A	2	2	0	1	2	\	200	200
MMMOP6-B	3	3	0	2	2	\	480	480
MMMOP6-C	2	4	2	1	2	\	200	400
MMMOP6-D	3	5	2	1	2	\	480	960

C. Competing Algorithms

In the experiments, two algorithms are chosen for comparison to assess the performance of the proposed TriMOEA-TA&R.

The first algorithm is MO_Ring_PSO_SCD proposed in [31]. It adopts a particle swarm algorithm with ring topology for searching. Moreover, a special crowding distance is developed to maintain diversities both in the objective and decision spaces.

The second algorithm is DN-NSGA-II presented in [5]. It has a similar environmental selection strategy with NSGA-II. Besides, it uses a decision space based niching method in the mating selection, which is particularly designed for solving MMMOPs.

D. Parameter Settings

This section gives the parameter settings of the test problems and the competing algorithms.

In the experiments, 28 test problems are adopted. Table I lists the parameter settings of MMMOP1-6 and the corresponding parameter settings of the competing algorithms. These test problems have various features and difficulty levels due to different parameter settings. Generally, if a test problem has a larger number of objectives (M), decision variables (n) or Pareto optimal solutions in the decision space (controlled by k_A, k_B, c_i, and d_i), it is more difficult to solve. For example, the PF and PS of MMMOP3-D are more difficult to achieve than those of MMMOP3-A. In both MMMOP1 and MMMOP2, there are multiple Pareto optimal solutions which have different independent convergence-related decision variable values for each point on the PF. However, MMMOP2 is more difficult for an optimizer to maintain diversity in the decision space than MMMOP1. In the A and B types of MMMOP3, MMMOP4 and MMMOP5, there are multiple Pareto optimal solutions which only have different diversity-related decision variable values for each point on the PF. Thus, the effect of our proposed recombination strategy may be unapparent on them. The C and D types of MMMOP3, MMMOP4 and MMMOP5 have more Pareto optimal solutions in the decision space than the A and B types. Besides, the PSs of MMMOP4 and MMMOP5 are more complex than that of MMMOP3 according to Subsection IV.A. Hence, they would be more difficult to solve than MMMOP3. In MMMOP6, there are multiple Pareto optimal solutions which have different convergence-related decision variable values for each point on the PF. However, all the convergence-related decision variables in the A and B types of MMMOP6 are interactive with the mixed decision variables. This will make our proposed recombination strategy inactive.

In the test problems where multiple Pareto optimal solutions with different independent convergence-related decision variable values exist for each point on the PF, the population size of TriMOEA-TA&R (N in Table II) is smaller than that of the other algorithms (N_{oth} in Table II), since a large population is not necessary during the evolution. For a fair comparison by IGDM, N_{oth} is set no less than the final solution set of TriMOEA-TA&R. The termination criterion of all compared algorithms is the predefined maximum number of generations. We set the maximum number of generations to 500 for MMMOP1-6 to guarantee that all algorithms can converge. Note that for all the competing algorithms, the population size and the maximum number of generations are set to 800 and 100, respectively, for solving MMF1-8 according to the original study [31].

The other common parameters adopted by all competing algorithms are set as follows. Simulated binary crossover and polynomial mutation are used as the crossover and mutation operators, with both distribution indexes being set to 20. The crossover and mutation probabilities are 1.0 and $1/n$, respectively, where n is the number of decision variables.

The systematic approache [3] is adopted to generate reference vectors in TriMOEA-TA&R, where the divisions in each dimension is set 99 and 14 for 2-objective and 3-objective test problems, respectively.

In TriMOEA-TA&R, if ϵ_{peak} is set too small, it will be difficult to find all the peak solutions. On the contrary, if ϵ_{peak} is set too large, a large number of fake peak solutions (i.e. solutions far away from the PF) could be found. Although the true peak solutions can also be found, the final solution set will be extremely huge. This actually does not have obvious effect on the performance of TriMOEA-TA&R. However, the size of the final solution set of TriMOEA-TA&R will be huge, and the other competing algorithms will struggle to maintain a population in a similar size. Based on our preliminary investigation, we set ϵ_{peak} 0.01. Under this setting, fake peak solutions are hardly found.

TABLE II
RESULTS OF IGDM

IGDM	TriMOEA-TA&R	MO_Ring_PSO_SCD	DN-NSGA-II
MMMOP1-A	**4.200E-03**	5.483E-01 +	5.122E-01 +
MMMOP1-B	**7.961E-02**	5.982E-01 +	5.645E-01 +
MMMOP2-A	**1.377E-01**	4.901E-01 +	5.733E-01 +
MMMOP2-B	**3.732E-01**	5.837E-01 +	6.386E-01 +
MMMOP3-A	**5.125E-03**	5.355E-03 =	5.276E-03 =
MMMOP3-B	**5.411E-02**	6.977E-02 +	7.332E-02 +
MMMOP3-C	**1.718E-02**	4.678E-01 +	3.308E-01 +
MMMOP3-D	**5.865E-02**	1.432E-01 +	9.866E-02 +
MMMOP4-A	**5.311E-03**	5.407E-03 =	5.453E-03 =
MMMOP4-B	**6.022E-02**	8.322E-02 +	8.272E-02 +
MMMOP4-C	**3.251E-02**	4.870E-01 +	4.809E-01 +
MMMOP4-D	**5.833E-02**	4.504E-01 +	4.045E-01 +
MMMOP5-A	**4.925E-03**	9.724E-03 +	9.075E-03 +
MMMOP5-B	**5.325E-02**	7.544E-02 +	8.770E-02 +
MMMOP5-C	**1.076E-01**	4.734E-01 +	4.528E-01 +
MMMOP5-D	**5.250E-02**	4.473E-01 +	3.871E-01 +
MMMOP6-A	**7.036E-03**	1.264E-02 +	1.190E-02 +
MMMOP6-B	**9.043E-02**	9.586E-02 =	1.047E-01 +
MMMOP6-C	**2.544E-01**	6.997E-01 +	7.112E-01 +
MMMOP6-D	**1.347E-01**	5.938E-01 +	6.328E-01 +
Sumup of MMMOP	+ \ - \ =	17 \ 0 \ 3	18 \ 0 \ 2
MMF1	**2.331E-03**	2.337E-03 =	2.791E-03 +
MMF2	1.241E-02	**1.144E-02** =	2.065E-02 =
MMF3	3.290E-02	**3.037E-02** =	4.112E-02 +
MMF4	**2.531E-01**	2.532E-01 =	2.567E-01 +
MMF5	**4.329E-03**	4.443E-03 =	1.606E-02 +
MMF6	**4.105E-03**	4.390E-03 +	1.553E-02 +
MMF7	**1.890E-03**	2.215E-03 +	2.285E-03 +
MMF8	3.261E-01	**3.078E-01** -	3.319E-01 =
Sumup of MMF	+ \ - \ =	2 \ 1 \ 5	6 \ 0 \ 2
Sumup of All	+ \ - \ =	19 \ 1 \ 8	24 \0 \ 4

In TriMOEA-TA&R, the sizes of the convergence and the diversity archives are equal to the population size. σ_{niche} is set to 0.05 for MMMOP2-A, 0.3 for MMMOP4-D and MMMOP5-D, and 0.1 for the other test problems. If X_{IC} is detected to be empty, p_{con} is set to 0.2, otherwise it is set to 0.5. Please refer to Section IV in the SM for the sensitivity analyses of σ_{niche} and p_{con}.

In MO_Ring_PSO_SCD, both C_1 and C_2 are set to 2.05 and W is set to 0.7298 according to the original study [31].

In DN-NSGA-II, the Crowding Factor (CF) is set to half of the population size as the authors recommend [5].

Each algorithm is run for 30 times on each test instance, and the mean values of IGDM are calculated. In addition, the Wilcoxon's rank sum test is employed to determine whether one algorithm has a statistically significant difference with the other on IGDM, and the null hypothesis is rejected at a significant level of 0.05.

V. RESULTS AND DISCUSSIONS

In this section, the performance of TriMOEA-TA&R is empirically evaluated by comparing it with MO_Ring_PSO_SCD [31] and DN-NSGA-II [5]. Tables II shows the mean values of IGDM obtained by different algorithms. '$+$' ('$-$') indicates that TriMOEA-TA&R shows significantly better (worse) performance in the comparison. '$=$' indicates that there is no significant difference between the compared results. We also present the achieved solution sets of some representative instances in the decision space by these algorithms in a given single run to visually investigate their performance in Figs. 8-13 in the SM. From Table II and Figs. 8-13 in the SM, we have the following observations.

For MMMOP1, TriMOEA-TA&R significantly outperforms the other algorithms according to the results of IGDM.

MO_Ring_PSO_SCD and DN-NSGA-II do not achieve satisfactory results. We observe that TriMOEA-TA&R works very well in most runs. However, it fails to identify every peak solution in the convergence archive in a few runs on MMMOP1-B due to the huge search space, which results in undesirable performance. From Fig. 8 (a) in the SM, we can see that TriMOEA-TA&R reaches all the parts of PS. The solutions obtained by MO_Ring_PSO_SCD and DN-NSGA-II in Fig.8 (b) and (c) in the SM almost concentrate on a single line, but they also find several solutions on other lines.

For MMMOP2, TriMOEA-TA&R is the best one among these competing optimizers. However, it can be seen from Fig. 9 (a) in the SM that the solutions on the bottom line are very difficult to achieve. As we have discussed in Subsection III.C, when σ_{niche} is large, a peak solution very close to another one is likely to be de-emphasized. Conversely, if σ_{niche} is small, the efficiency of searching peak solutions will be very low in the high-dimensional decision space. Therefore, missing peak solutions may reduce the TriMOEA-TA&R's ability in obtaining all the parts of PS. From Fig. 9 (b) in the SM, we can see that although most of the solutions obtained by MO_Ring_PSO_SCD concentrate on one line, they are denser on the line as x_1 increases, since MO_Ring_PSO_SCD tries to maintain diversities both in the objective and the decision spaces. This is why MO_Ring_PSO_SCD outperforms DN-NSGA-II on MMMOP2 according to the mean values of IGDM.

TriMOEA-TA&R achieves the best mean value of IGDM on MMMOP3. MO_Ring_PSO_SCD and DN-NSGA-II perform well on MMMOP3 except the C type, and there is no significant difference between TriMOEA-TA&R and them on MMMOP3-A. We can see from Fig. 10 in the SM, the solutions obtained by TriMOEA-TA&R seem more uniformly distributed than those obtained by MO_Ring_PSO_SCD and DN-NSGA-II. This may indicate that the diversity maintenance strategy in TriMOEA-TA&R performs better than the crowding distance methods in MO_Ring_PSO_SCD and DN-NSGA-II.

The PS of MMMOP4 has the same geometrical shape with that of MMMOP3 in the decision space, whereas the number of Pareto optimal solutions varies in terms of the position of the PF in the objective space. Due to this feature, most of the mean values of IGDM obtained by each competing algorithm on MMMOP4 are a bit larger than those on MMMOP3. TriMOEA-TA&R performs best on MMMOP4. However, there is no significant difference between the results obtained by the competing algorithms on MMMOP4-A. From Fig. 11 (a) in the SM, we can see that TriMOEA-TA&R can obtain a diverse Pareto optimal solution set both in the convergence- (x_3) and diversity-related (x_1 and x_2) decision spaces.

TriMOEA-TA&R performs significantly better than the others on MMMOP5. From Fig. 12, we can see that the solutions obtained by TriMOEA-TA&R are clearly denser when $x_1 > 0.67$ and $x_2 > 0.67$. MO_Ring_PSO_SCD and DN-NSGA-II behave similarly as that in MMMOP3. This indicates that they cannot make a distinction among the Pareto optimal regions with different densities. Therefore, the mean value obtained by MO_Ring_PSO_SCD and DN-NSGA-II on MMMOP5 are not good as those in MMMOP3.

For MMMOP6, TriMOEA-TA&R achieves the best performance among these algorithms. Note that MMMOP6-A and -B do not have any independent convergence-related decision variable, the recombination strategy in TriMOEA-TA&R is inactive. However, the mean values of IGDM obtained by TriMOEA-TA&R are the best among the competing algorithms. It can be seen from Fig. 13 in the SM that most of the solutions obtained by TriMOEA-TA&R, MO_Ring_PSO_SCD and DN-NSGA-II are distributed well in each Pareto optimal region.

All the MMF test problems have two decision variables which have interactions with each other. Therefore, the recombination strategy in TriMOEA-TA&R is inactive like that on MMMOP6-A and -B. Even so, thanks to the two-archive strategy, TriMOEA-TA&R achieves five best mean IGDM values on MMF test problems, and it significantly outperforms MO_Ring_PSO_SCD on MMF6 and MMF7 and DN-NSGA-II on MMF1 and MMF3-7. However, TriMOEA-TA&R shows significantly worse performance than MO_Ring_PSO_SCD on MMF8. The reason is that for some points on the PF of MMF8, the distances among Pareto optimal solutions in the decision space are extremely small, which results in the failure of TriMOEA-TA&R in distinguishing them. In addition, although there is no significant difference between the results obtained by TriMOEA-TA&R and MO_Ring_PSO_SCD on MMF2 and MMF3, TriMOEA-TA&R receives worse mean IGDM values. We found that TriMOEA-TA&R can maintain a good diversity in the objective space on MMF2 and MMF3; however, some small parts of the PSs are missing. The possible reason is that the diversity archive in TriMOEA-TA&R first tries to promote diversity in the objective space and then does that in the decision space. Relatively insufficient search efforts in the decision space may result in the missing parts. Further balancing diversities in the objective and the decision spaces is an interesting improvement for TriMOEA-TA&R in our future work.

According the above observations, we can draw the following conclusions: (1) TriMOEA-TA&R outperforms the other algorithms on most test problems. For the test problems which contain independent convergence-related decision variables, e.g., MMMOP test problems expect MMMOP6-A and -B, TriMOEA-TA&R can gain great convergence and/or computational benefits from the recombination strategy. Reader can refer to Section III in the SM for the further investigation on the effect of the recombination strategy. On the other hand, owing to the two-archive strategy, TriMOEA-TA&R also works very well for the test problems with no independent convergence-related decision variable, e.g., MMMOP6-A and -B and MMF1-7. (2) MO_Ring_PSO_SCD and DN-NSGA-II can achieve encouraging results on some relatively simple test problems, e.g., MMMOP3-A, MMMOP4-A, and MMFs. Although they both employ the crowding distance method to maintain diversity in the decision space, they may fail to locate some Pareto optimal solutions, e.g., A and B types of MMMOP1 and MMMOP2, C and D types of MMMOP3, MMMOP4, MMMOP5, and MMMOP6.

In the SM, we also show the PSP results obtained by the competing algorithms and discuss the difference between PSP and IGDM by an example in Section V. Readers can refer to them if interested.

VI. Conclusion

In this paper, we have proposed a novel multi-modal multi-objective evolutionary algorithm using two-archive and recombination strategies, termed TriMOEA-TA&R. In the proposed method, the independent convergence-related decision variables are detected by a decision variable analytical technique at first, which will help to find multiple Pareto optimal solutions. A general framework of two archives, i.e., the convergence and the diversity archives, is proposed to cooperatively solve MMMOPs. The division of labor in the two archives reduces the difficulties in the environmental selection procedure. In addition, the recombination strategy can reduce the size of population under particular conditions.

We have also proposed a set of benchmark test functions, i.e., MMMOP1-6, and a performance metric, i.e., IGDM, for multi-modal multi-objective optimization. This will encourage more interests in this new research area in the Evolutionary Computation community. In this study, they are also adopted to demonstrate the effectiveness of TriMOEA-TA&R by comparing it with MO_Ring_PSO_SCD and DN-NSGA-II. The experimental results of IGDM demonstrate that TriMOEA-TA&R is the best among the compared algorithms on most test problems.

In this study, since our purpose is to establish a general framework to solve MMMOPs, some specific strategies can be further developed and employed in TriMOEA-TA&R. For example, the niching methods in the decision space is a basic clearing strategy. For some problems in which the multiple Pareto optimal solutions have vastly different spacing between them (e.g., MMMOP2 and MMF8), this basic clearing strategy may not find all Pareto optimal solutions. An advanced strategy to adaptively tune the niche size or without niche parameters can be developed for TriMOEA-TA&R in the future work. Similarly, a strategy to adaptively generate reference vectors during the evolution process can be employed for TriMOEA-TA&R to solve the MMMOPs with complex PFs.

In addition, it can be seen from the experiments that the compared algorithms may struggle to achieve good performance as the dimension of the decision space increases. On the other hand, the increasing number of objectives can pose a great challenge in the "curse of dimensionality." Investigating and improving TriMOEA-TA&R on solving MMMOPs with a large number of both objectives and decision variables is certainly interesting for our future research.

*The source codes of TriMOEA-TA&R, MMMOP1-6, and IGDM is available on https://github.com/yiping0liu.

References

[1] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[2] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: improving the strength Pareto evolutionary algorithm," Eidgenössische Technische Hochschule Zürich (ETH), Institut für Technische Informatik und Kommunikationsnetze (TIK), Tech. Rep., 2001.

[3] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.

[4] K. Deb and S. Tiwari, "Omni-optimizer: A generic evolutionary algorithm for single and multi-objective optimization," *European Journal of Operational Research*, vol. 185, no. 3, pp. 1062–1087, 2008.

[5] J. Liang, C. Yue, and B. Qu, "Multimodal multi-objective optimization: A preliminary study," in *2016 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2016, pp. 2454–2461.

[6] A. Jaszkiewicz, "On the performance of multiple-objective genetic local search on the 0/1 knapsack problem-a comparative experiment," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 4, pp. 402–412, 2002.

[7] J. Michalek, R. Choudhary, and P. Papalambros, "Architectural layout design optimization," *Engineering Optimization*, vol. 34, no. 5, pp. 461–484, 2002.

[8] Y. Han, D. Gong, Y. Jin, and Q. Pan, "Evolutionary multi-objective blocking lot-streaming flow shop scheduling with machine breakdowns," *IEEE Transactions on Cybernetics*, 2017, Early Access.

[9] H. Ishibuchi, N. Akedo, and Y. Nojima, "A many-objective test problem for visually examining diversity maintenance behavior in a decision space," in *Proceedings of the 13th annual conference on Genetic and evolutionary computation*. ACM, 2011, pp. 649–656.

[10] D. E. Goldberg, J. Richardson *et al.*, "Genetic algorithms with sharing for multimodal function optimization," in *Genetic algorithms and their applications: Proceedings of the Second International Conference on Genetic Algorithms*. Hillsdale, NJ: Lawrence Erlbaum, 1987, pp. 41–49.

[11] R. Thomsen, "Multimodal optimization using crowding-based differential evolution," in *IEEE Congress on Evolutionary Computation*, vol. 2. IEEE, 2004, pp. 1382–1389.

[12] D. A. Van Veldhuizen and G. B. Lamont, "On measuring multiobjective evolutionary algorithm performance," in *IEEE Congress on Evolutionary Computation*, vol. 1. IEEE, 2000, pp. 204–211.

[13] K. Deb, *Multi-objective optimization using evolutionary algorithms*. John Wiley & Sons, 2001.

[14] ——, "Multi-objective genetic algorithms: Problem difficulties and construction of test problems," *Evolutionary Computation*, vol. 7, no. 3, pp. 205–230, 1999.

[15] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Parallel Problem Solving from Nature-PPSN VIII*. Birmingham, UK: Springer, 2004, pp. 832–842.

[16] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *Evolutionary Computation*, vol. 19, no. 1, pp. 45–76, 2011.

[17] Y. Liu, D. Gong, X. Sun, and Y. Zhang, "Many-objective evolutionary optimization based on reference points," *Applied Soft Computing*, vol. 50, no. 1, pp. 344–355, 2017.

[18] Z. He and G. G. Yen, "Many-objective evolutionary algorithms based on coordinated selection strategy," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 2, pp. 220–233, 2017.

[19] W. Hu, G. G. Yen, and G. Luo, "Many-objective particle swarm optimization using two-stage strategy and parallel cell coordinate system." *IEEE Trans. Cybernetics*, vol. 47, no. 6, pp. 1446–1459, 2017.

[20] X. Li, M. G. Epitropakis, K. Deb, and A. Engelbrecht, "Seeking multiple solutions: an updated survey on niching methods and their applications," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 4, pp. 518–538, 2017.

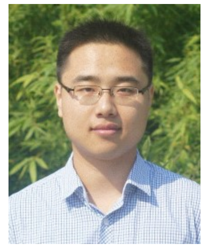
[21] A. Pétrowski, "A clearing procedure as a niching method for genetic algorithms," in *Proceedings of IEEE International Conference on Evolutionary Computation*. IEEE, 1996, pp. 798–803.

[22] W. Gao, G. G. Yen, and S. Liu, "A cluster-based differential evolution with self-adaptive strategy for multimodal optimization," *IEEE Transactions on Cybernetics*, vol. 44, no. 8, pp. 1314–1327, 2014.

[23] B. Qu, P. Suganthan, and S. Das, "A distance-based locally informed particle swarm model for multimodal optimization," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 3, pp. 387–402, 2013.

[24] Y. Wang, H. Li, G. G. Yen, and W. Song, "MOMMOP: Multiobjective optimization for locating multiple optimal solutions of multimodal optimization problems," *IEEE Transactions on Cybernetics*, vol. 45, no. 4, pp. 830–843, 2015.

[25] D. Gong, Y. Liu, X. Ji, and J. Sun, "Evolutionary algorithms with users preferences for solving hybrid interval multi-objective optimization problems," *Applied Intelligence*, vol. 43, no. 3, pp. 676–694, 2015.
[26] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evolutionary Computation*, vol. 8, no. 2, pp. 173–195, 2000.
[27] R. W. Morrison and K. A. De Jong, "Measurement of population diversity," in *International Conference on Artificial Evolution (Evolution Artificielle)*. Springer, 2001, pp. 31–41.
[28] B. Lacevic and E. Amaldi, "Ectropy of diversity measures for populations in euclidean space," *Information Sciences*, vol. 181, no. 11, pp. 2316–2339, 2011.
[29] R. K. Ursem, "Diversity-guided evolutionary algorithms," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2002, pp. 462–471.
[30] B. L. Miller and M. J. Shaw, "Genetic algorithms with dynamic niche sharing for multimodal function optimization," in *Evolutionary Computation, 1996., Proceedings of IEEE International Conference on*. IEEE, 1996, pp. 786–791.
[31] C. Yue, B. Qu, and J. Liang, "A multi-objective particle swarm optimizer using ring topology for solving multimodal multi-objective problems," *IEEE Transactions on Evolutionary Computation*, 2017, Early Access.
[32] P. A. Bosman and D. Thierens, "The balance between proximity and diversity in multiobjective evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 174–188, 2003.
[33] K. Praditwong and X. Yao, "A new multi-objective evolutionary optimisation algorithm: The two-archive algorithm," in *International Conference on Computational intelligence and Security*, vol. 1. IEEE, 2006, pp. 286–291.
[34] H. Wang, L. Jiao, and X. Yao, "Two_arch2: An improved two-archive algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 4, pp. 524–541, 2015.
[35] X. Ma, F. Liu, Y. Qi, X. Wang, L. Li, L. Jiao, M. Yin, and M. Gong, "A multiobjective evolutionary algorithm based on decision variable analyses for multiobjective optimization problems with large-scale variables," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 2, pp. 275–298, 2016.
[36] X. Zhang, Y. Tian, R. Cheng, and Y. Jin, "A decision variable clustering-based evolutionary algorithm for large-scale many-objective optimization," *IEEE Transactions on Evolutionary Computation*, 2016, to be published.
[37] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, *Scalable test problems for evolutionary multiobjective optimization*. Springer, 2005.
[38] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 5, pp. 477–506, 2006.
[39] T. Weise, R. Chiong, and K. Tang, "Evolutionary optimization: Pitfalls and booby traps," *Journal of Computer Science and Technology*, vol. 27, no. 5, pp. 907–936, 2012.
[40] M. Tezuka, M. Munetomo, and K. Akama, "Linkage identification by nonlinearity check for real-coded genetic algorithms," in *Genetic and Evolutionary Computation Conference*. Springer, 2004, pp. 222–233.
[41] Q. Zhang and H. Muhlenbein, "On the convergence of a class of estimation of distribution algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 2, pp. 127–136, 2004.
[42] Y. Liu, D. Gong, J. Sun, and Y. Jin, "A many-objective evolutionary algorithm using a one-by-one selection strategy," *IEEE Transactions on Cybernetics*, vol. 47, no. 9, pp. 2689–2702, 2017.
[43] X. Li, A. Engelbrecht, and M. G. Epitropakis, "Benchmark functions for CEC 2013 special session and competition on niching methods for multimodal function optimization," RMIT University, Evolutionary Computation and Machine Learning Group, Australia, Technical Report, 2013.
[44] B. Qu, J. Liang, Z. Wang, Q. Chen, and P. N. Suganthan, "Novel benchmark functions for continuous multimodal optimization with comparative results," *Swarm and Evolutionary Computation*, vol. 26, pp. 23–34, 2016.
[45] Q. Zhang, A. Zhou, S. Zhao, P. N. Suganthan, W. Liu, and S. Tiwari, "Multiobjective optimization test instances for the CEC 2009 special session and competition," University of Essex, UK, Technical Report, CES-487, 2008.
[46] 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 Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 1167–1189, 2009.



Yiping Liu (M'18) received the Ph.D. degree in control theory and control engineering from China University of Mining and Technology, China in 2017. He was a joint Ph.D. student financed by the China Scholarship Council during 2016-2017 in the School of Electrical and Computer Engineering, Oklahoma State University, USA. He is currently a research assistant professor in the Department of Computer Science and Intelligent Systems, Osaka Prefecture University, Japan. His research interest includes computational intelligence, evolutionary computation, multi-objective optimization, and machine learning.

Gary G. Yen (S'87-M'88-SM'97-F'09) received his Ph.D. degree in electrical and computer engineering from the University of Notre Dame in 1992. He is currently a Regents Professor in the School of Electrical and Computer Engineering, Oklahoma State University. His research interest includes intelligent control, computational intelligence, evolutionary multiobjective optimization, conditional health monitoring, signal processing and their industrial/defense applications.

Gary was an associate editor of the *IEEE Transactions on Neural Networks* and *IEEE Control Systems Magazine* during 1994-1999, and of the *IEEE Transactions on Control Systems Technology*, *IEEE Transactions on Systems, Man and Cybernetics* and IFAC Journal on *Automatica and Mechatronics* during 2000-2010. He is currently serving as an associate editor for the *IEEE Transactions on Evolutionary Computation*, *IEEE Transactions on Emerging Topics on Computational Intelligence* and *IEEE Transactions on Cybernetics*. Gary served as Vice President for the Technical Activities, IEEE Computational Intelligence Society in 2004-2005 and is the founding editor-in-chief of the *IEEE Computational Intelligence Magazine*, 2006-2009. He was the President of the IEEE Computational Intelligence Society in 2010-2011 and is elected as a Distinguished Lecturer for the terms 2012-2014 and again 2016-2018. He received Regents Distinguished Research Award from OSU in 2009, 2011 Andrew P Sage Best Transactions Paper award from IEEE Systems, Man and Cybernetics Society, 2013 Meritorious Service award from IEEE Computational Intelligence Society and 2014 Lockheed Martin Aeronautics Excellence Teaching award. Currently he serves as the chair of IEEE/CIS Fellow Committee. He is a Fellow of IEEE and IET.

Dunwei Gong (M16) received his Ph.D. degree in control theory and control engineering from China University of Mining and Technology, China in 1999. He is currently a Professor in School of Information and Control Engineering, China University of Mining and Technology, and the director of Center on Intelligent Optimization and Control Research, China University of Mining and Technology. His research interest includes evolutionary computation and search-based software engineering.