Multiobjective Differential Evolution with Speciation for Constrained Multimodal Multiobjective Optimization

Jing Liang, *Senior Member, IEEE*, Hongyu Lin, Caitong Yue*, *Member, IEEE*, Kunjie Yu, Boyang Qu, Ying Guo, Ke Chen, Kangjia Qiao

***Abstract*—This paper proposes a novel differential evolution algorithm for solving constrained multimodal multiobjective optimization problems (CMMOPs), which may have multiple feasible Pareto optimal solutions with the identical objective vector. In CMMOPs, due to the coexistence of multimodality and constraints, it is difficult for current algorithms to perform well in both objective and decision spaces. The proposed algorithm uses speciation mechanism to induce niches preserving more feasible Pareto optimal solutions and adopts an improved environment selection criterion to enhance diversity. The algorithm can not only obtain feasible solutions but also retain more well-distributed feasible Pareto optimal solutions. Moreover, a set of constrained multimodal multiobjective test functions is developed. All these test functions have multimodal characteristics and contain multiple constraints. Meanwhile, this paper proposes a new indicator, which comprehensively considers the feasibility, convergence, and diversity of a solution set. The experimental comparison between the proposed method and other state-of-the-art algorithms on the test functions and real-world location-selection problem validates its effectiveness for CMMOPs.**

***Index Terms*—Benchmark functions, constraints, multimodal, multiobjective, evolutionary algorithms, Speciation.**

I. Introduction

CONSTRAINED multiobjective optimization problems (CMOPs) are attracting more and more attention because there are usually various constrained conditions in the practical application. In general, a constrained multiobjective optimization problem can be expressed as follows [1]:

$$\min \mathbf{F}(\mathbf{x}) = \{f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_M(\mathbf{x})\}^{\mathrm{T}} \tag{1}$$

subject to:

$$\left\{ \begin{array}{ll} g_i(\mathbf{x}) \leq 0, & i = 1, \ldots, p \\ h_i(\mathbf{x}) = 0, & i = p+1, \ldots, q \end{array} \right. \tag{2}$$

where $\mathbf{x} = (x_1, \cdots, x_D)$ is a D-dimensional decision vector, $\mathbf{F}(\mathbf{x})$ is a M-dimensional objective vector, $g_i(\mathbf{x}) \leq 0$ $(i = 1, \ldots, p)$ is the i^{th} inequality constraint, and $h_i(\mathbf{x}) = 0$ $(i = p+1, \ldots, q)$ is the $(i-p)^{\mathrm{th}}$ equality constraint. Decision space consists of all the possible values of D-dimensional decision vector $\mathbf{x}$, where $\mathbf{x} \in \left\{x_i^{\mathrm{lower}} \leq x_i \leq x_i^{\mathrm{upper}}, i = 1, \ldots, D\right\}$, in which x_i^{lower} and x_i^{upper} represent the minimum value and maximum value of x_i respectively. Objective space consists of all the possible values of M-dimensional objective vector $\mathbf{F}$, where $\mathbf{F} \in \left\{f_j^{\mathrm{lower}} \leq f_j \leq f_j^{\mathrm{upper}}, j = 1, \ldots, M\right\}$. Similarly, f_j^{lower} and f_j^{upper} represent the minimum value and maximum value of f_j, respectively. The constraint violation (CV) value of decision vector $\mathbf{x}$ for the i^{th} constraint can be written as follows:

$$CV_i(\mathbf{x}) = \left\{ \begin{array}{l} \max\ (0, g_i(\mathbf{x})), i = 1, \cdots, p \\ \max\ (0, |h_i(\mathbf{x})| - \xi), i = p+1, \cdots, q \end{array} \right. \tag{3}$$

where ξ is a tolerance value for relaxing the equality constraints. If $\mathbf{x}$ satisfies equation (4), it will be identified as a feasible solution. Otherwise, $\mathbf{x}$ is deemed to be infeasible.

$$CV(\mathbf{x}) = \sum_{i=1}^{q} CV_i(\mathbf{x}) = 0 \tag{4}$$

In multiobjective optimization, a solution is incapable of minimizing all objectives simultaneously in equation (1) because of the contradiction among different objectives. A set of best trade-off solutions named Pareto optimal solution Set (PS) is usually provided. The corresponding objective values to the PS are termed the Pareto Front (PF). Similarly, in constrained multiobjective optimization, more appropriate and feasible Pareto optimal solutions with good distribution are required. The set of all feasible Pareto optimal solutions is termed the Constrained Pareto optimal solution Set (CPS) and the set of all Pareto optimal solutions considering no constraints is termed the Unconstrained Pareto optimal solution Set (UPS) [2]. Their corresponding objective values form the Constrained Pareto Front (CPF) and Unconstrained Pareto Front (UPF) [2].

In practical application, there are many CMOPs, such as vehicle routing [3], reconfigurable real-time systems development [4], and software engineering [5]. Therefore, many effective constrained multiobjective optimization evolutionary algorithms (CMOEAs) and multiple constraint handling techniques (CHTs) are designed for effectively solving these practical optimization problems. For example, in NSGA-II [6], the constrained dominance principle (CDP) was proposed, which gives higher priority to feasible solutions ensuring that the algorithm can find feasible regions. Then, Tian *et al.* [7] proposed a coevolutionary framework, called CCMO. The proposed framework can solve CMOPs more efficiently by sharing the information between different populations. Ma *et al.* [8] designed a constrained multiobjective framework on the basis of Shift Penalty (Ship), which gives rise to drive the population into feasible regions from different directions and help the population converge to the true PF. In addition, many difficult constrained multiobjective test functions have been designed to study the performance of CMOEAs.

The typical test functions, including CTP [9], C-DTLZ [10], and MW [11] have various properties in objective space, such as: discrete, narrow, irregular feasible regions, and degenerated PFs. It should be noted that these CMOEAs and CMOPs pay more attention to the objective space, while seldom focus on the decision space. However, in addition to CMOPs, there are multimodal multiobjective optimization problems (MMOPs) in multiobjective optimization problems, in which existing multiple Pareto optimal solutions with identical objective values [12]. Different from CMOPs, MMOPs requires algorithms to maintain the diversity of solutions in decision space.

Moreover, in multiobjective optimization, there are not only CMOPs and MMOPs, but also constrained multimodal multiobjective optimization problems (CMMOPs) [13]. In CMMOPs, there may exist at least two CPSs corresponding to the identical CPF. There is a chance that it is adequate to find one of the CPSs for some problems. Nonetheless, discovering more CPSs may be necessary for decision makers to consider appropriate solutions [14] [15].

A simple real-world example, which is the constrained multimodal multiobjective location-selection problem, is illustrated in Fig. 1. In this map, tenants want to find locations short enough from schools, hospitals, and convenience stores. Therefore, this is a distance minimization problem with three objectives to be optimized: (1) the distance from the selected location to the school; (2) the distance from the selected location to the hospital; (3) the distance from the selected location to the convenience store. Four non-dominated options with identical minimum distances to these objective locations are shown in Fig. 1, which are marked Option 1, Option 2, Option 3 and Option 4.

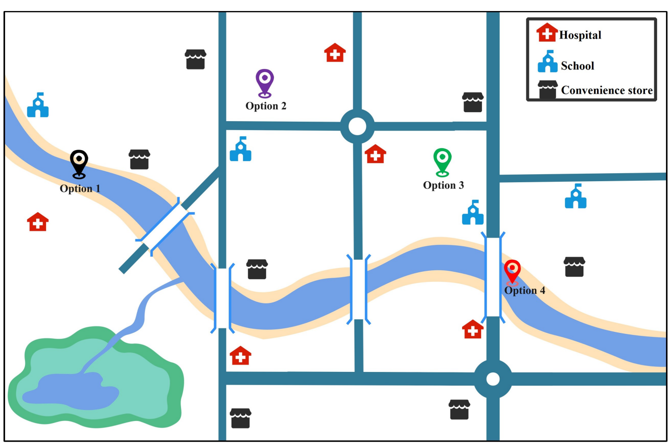


Fig. 1. Illustration of the constrained multimodal multiobjective location-selection problem. As can be seen, the 1^{st} dot and the 4^{th} dot are located in the river, which are clearly unacceptable. Meanwhile, the 2^{nd} dot and the 3^{rd} dot are both desirable options with different characteristics.

However, Option 1 and Option 4 are located in the river, which are obviously infeasible. Option 3 has a better view along the river and there are more schools around, which leads to higher rents. Option 2 locates in the suburb of the city, so it may lead to cheaper rents. The school district environment is necessary for some tenants, while others desire cheaper rents. If algorithms are unable to get multiple Pareto optimal solutions satisfying the constraints, they can not satisfy the requirements of different tenants. In this situation, it is valuable to reserve more feasible Pareto optimal solutions.

Solving CMMOPs is not an easy task since they have three characteristics: constrained, multiobjective, and multimodal. Therefore, this kind of problem puts forward higher requirements and challenges for current multiobjective optimization evolutionary algorithms (MOEAs) to balance diversity, feasibility, and convergence performance both in the decision and objective spaces.

The main task of solving CMMOPs is to search for multiple Pareto solution optimal sets in the feasible regions rather than the entire search space. The existing methods can either deal with CMOPs or MMOPs. However, they might not present an excellent performance when solving CMMOPs. Existing constrained multiobjective optimization methods can find only one CPS or several different parts of multiple CPSs while they can not simultaneously find multiple complete CPSs. The existing multimodal multiobjective optimization evolutionary algorithms (MMOEAs) aim at the whole decision space without a constraint handling technology, which may lead them to maintain many infeasible Pareto optimal solutions. However, a simple combination of existing MMOEAs and constraint handing technologies may not solve CMMOPs more effectively. Therefore, it is imperative to study specific constrained multimodal multiobjective optimization evolutionary algorithms.

Although there are many CMMOPs in practical applications, there are few researches on constrained multimodal multiobjective optimization and there are no test functions with the same characteristics. For these reasons, this paper designs a set of constrained multimodal multiobjective test functions, called CMMFs. In CMMFs, various constraints (for example, inequality constraints, nonlinear constraints, and linear constraints) are collected together to construct feasible regions and develop more properties of feasible regions (e.g, discrete, extremely small, and multimodal).

To solve CMMOPs, this paper proposes a constrained multimodal multiobjective differential evolution algorithm with speciation mechanism (CMMODE), which can reserve more Pareto optimal solutions under the premise of finding as many feasible solutions as possible. The algorithm proposes an improved environment selection method and uses the speciation mechanism to enable the niches to independently evolve. The combination of this mechanism and CDP can not only help to search for feasible regions but also help to find multiple CPSs with different convergence difficulties. Therefore, it makes CMMODE effective in maintaining multiple CPSs in decision space.

This paper includes three major contributions:

1) A systematical design principle of constrained multimodal mutiobjective optimization test functions is designed according to the properties of CMMOPs in practical applications. With this design principle, four kinds of constrained multimodal mutiobjective optimization test functions can be generated. Seventeen test functions are given as examples to test the performance of different algorithms, named CMMFs.

2) A novel multiobjective differential evolution algorithm with speciation mechanism and improved environment selection method is proposed for solving CMMOPs, called CMMODE. Speciation mechanism is effective to reverse more Pareto optimal solutions. The improved environment selection method helps the population to be evenly distributed on different CPSs.
3) A new performance indicator is proposed to comprehensively measure the feasibility, convergence, and diversity of a solution set. Then, the experimental comparisons between CMMODE and other competitive CMOEAs and multimodal multiobjective optimization evolutionary algorithms (MMOEAs) show that CMMODE performs better on CMMFs and the constrained multimodal multiobjective location selection problem. Moreover, the mechanism of CMMODE is analyzed in detail.

The reminder of this paper is structured as follows. In Section II, related studies of constrained multiobjective optimization and muitimodal multiobjective optimization are reviewed. The properties, designed method, and instances of CMMFs are introduced in detail in Section III. The proposed algorithm CMMODE is described and analyzed at full length in Section IV. Experimental comparisons, verification, and analysis of results are presented in Section V. At last, the conclusion and future study directions are provided in Section VI.

II. Related Works

In this section, relevant works in the field of constrained multiobjective optimization, multimodal multiobjective optimization, and constrained multimodal optimization are introduced. Firstly, previous works on constrained multiobjective optimization test functions are reviewed and existing constrained multiobjective optimization evolutionary algorithms are roughly divided into four categories for reviewing. Then, some multimodal multiobjective optimization test functions and multimodal multiobjective optimization algorithms are reviewed. Finally, some prior works on constrained multimodal optimization are also introduced.

A. Brief Introduction to Constrained Multiobjective Optimization

In practical application, there are many CMOPs, such as software engineering [5], bus scheduling problem [16], and water distribution network optimization [17] and so on. Therefore, it is necessary to propose more CMOEAs for solving CMOPs. Meanwhile, many constrained multiobjective optimization test functions are proposed to explore the performance of CMOEAs more deeply.

Deb *et al*. [9] designed seven two-objective constrained optimization problems, called CTPs. CTPs launch a challenge for the algorithms of finding true PF by dividing the unconstrained PF into discrete regions and narrowing the feasible region in the whole search space. However, CTPs still have some limitations. For example, the objective number is not scalable, and most of the CTPs only have one constraint condition, retaining large feasible regions. Based on the CTPs, Li *et al*. designed a set of CMOPs, called NCTPs [18]. These test functions add more constraints to narrow feasible regions and adopt distance functions to increase the convergence difficulties. Ma *et al*. proposed a novel set of CMOPs with more complex and difficult feasible regions, called MWs [11]. In the framework of MWs, global and local adjustment processes are introduced. The former is adopted to narrow the feasible regions, and the latter is used for controlling the complexity of the feasible region boundary and constructing CPFs with different shapes. However, these test functions focus on the constraints in objective space and pay little attention to the decision constraints. Liu *et al*. proposed a set of test functions named DOCs [19], considering the constraints in decision and objective space. Moreover, DOCs add various decision constraints for constructing feasible regions with complex characteristics. Zhou *et al*. designed a set of test functions with scalable objective number and decision variables, called CFs [20]. In the framework of CFs, convergence-hardness and diversity-hardness are introduced by constructing two kinds of constraints.

As the research on constrained multiobjective optimization deepens, more effective CMOEAs are emerging into this field. Existing CMOEAs can be mainly classified into four categories in the light of their internal mechanisms and CHTs: 1) methods of separating constraints and objectives; 2) methods based on penalty function; 3) multiobjective methods; and 4) methods of transforming CMOPs into other problems.

The methods of separating constraints and objectives mainly include constrained dominance principle (CDP) [6], ε constrained method [21], and stochastic ranking (SR) [22]. In CDP, any feasible solution dominates all the infeasible solutions. When comparing two infeasible solutions, the solution with a higher CV value is dominated by the other solution. Takahama *et al*. designed a ε constrained method [21] for relaxing constraints by using the parameter ε. In [23], [24], [25], this method is combined with the decomposition-based evolutionary algorithms for solving CMOPs. Fan *et al*. [26] designed a strategy for adaptively adjusting ε parameter level in the light of the infeasible proportion. Yao *et al*. [22] introduced a probability parameter pf to determine whether to compare the objective values or the CV value while comparing two individuals. In [27], [28], this method is combined with the decomposition-based evolutionary algorithms to effectively address CMOPs.

The methods based on penalty function mainly include constructing a penalty function related to the CV value and combining the penalty functions with the objective function. Tessema *et al*. introduced a new fitness function with normalized CV values to reverse potential infeasible individuals in current population [29]. Jiao *et al*. proposed a method to adaptively adjust the penalty function on the basis of evolutionary situation and balance objectives and constraints [30].

In the multiobjective methods, constraints are regarded as additional objectives to solve the CMOPs, which are converted to an unconstrained MOP and it can be dealt with MOEAs [31]. Ray *et al*. [32] proposed an algorithm driven by infeasible solutions, and it approaches the feasible region boundary from the infeasible region. Long *et al*. [33] regarded the feasibility, convergence, and diversity of obtained solutions as new objectives in multiobjective optimization problems.

The methods of transforming CMOPs into other problems mainly include that CMOPs are transformed into multi-population cooperative optimization problems or multi-stage optimization problems to seek more potential information of infeasible solutions. CCMO [7] is an advanced algorithm using the unconstrained evolved population to help the constrained population evolve. Qiao *et al*. [34] transformed the original CMOPs into two related tasks to address objectives and constraints respectively. In [35], a double-phased framework is designed for unearthing more information about potential infeasible solutions. In the former phase, the algorithm dose not adopt any CHT to search for the UPF in objective space. In the latter phase, it employs the improved ε constrained method to drive the population to approach the CPF.

These CMOEAs seldom take account of the distribution in decision space, leading to the difficulty in exploring discrete and distant feasible regions. Therefore, it is crucial to design specific strategies for the above situation.

B. Brief Introduction to Multimodal Multiobjective Optimization

In practical application, there are many multimodal multiobjective optimization problems, such as engine designing [36], game map generating [37], and so on. In the last few years, multimodal multiobjective optimization has attracted more attention. Many multimodal multiobjective optimization test functions and MMOEAs have been proposed.

Deb *et al*. proposed omni-test problems, in which the number of Pareto optimization solution sets are controllable [38]. Ishibuchi *et al*. proposed a kind of Polygon-based problems with multiple identical polygons [15], in which the objective is to minimize the distances from the endpoints of polygons. Yue *et al*. [39] designed a set of scalable MMOPs with more complex properties, called MMFs. In this test function suite, the numbers of objectives, decision variables, and PSs are extensible, and it also has local PSs. Liu *et al*. [40] proposed a novel test problem set, exciting multiple PSs with different convergence difficulties, called IDMP.

Meanwhile, there are also many studies on MMOEAs for solving MMOPs. In the early days, Deb *et al*. [38] proposed omni-optimizer for maintaining more equivalent PSs, in which the crowding distances in decision and objective space are simultaneously considered. Similarly, the two distances between solutions are accumulated to construct a niche function in [41]. Based on the omni-optimizer, Liang *et al*. [12] proposed a variant of NSGA-II, which selects adjacent individuals to compete in the tournament selection. Liu *et al*. employed the niche-based sharing function to maintain the diversity [42]. Yue *et al*. proposed a novel special crowding distance (SCD) and employed ring topology to reverse more Pareto optimal solutions [43], [44]. In addition, the same author proposed a new differential evolution algorithm, which considers the individuals on different Pareto rankings when calculating the crowding distance [45]. Besides, Liu *et al*. [46] designed a new MMOEA with double archives and recombination strategy, named TriMOEA-TA&R. It adopts diversity archive and convergence archive to collaboratively solve MMOPs. The former archive employs a niche-based clustering and clearing method to store more solutions, while the latter archive guides the population to multiple PSs. Then, the same author found an imbalance problem when searching for multiple PSs in MMOPs and proposed an efficient algorithm (CPDEA) for solving these problems [40]. It takes density estimation associated with the local convergence of solutions as the selection criterion. Meanwhile, Zhang *et al*. [47] designed a strategy to adaptively adjust the density of population and divided the evolutionary processing into two phases. In the former phase, the niche strategy is employed only in decision space; while in the latter phase, this strategy is also adopted in objective space.

These MMOEAs are not able to solve effectively CMMOPs, because they lack necessary constraint handling technologies. In Sections III and V, it is provided that simply embedding constraint handling technologies into MMOEAs is not enough to solve CMMOPs. To overcome this issue, a novel CMMODE is proposed in section IV.

C. Brief Introduction to Constrained Multimodal Optimization

Constrained multimodal optimization problems have attracted more attention in the early stage, and many studies on the constrained multimodal single-objective optimization have emerged. Deb *et al*. [48] proposed a set of constrained multimodal test functions called CMMPs, which contains multiple constraints and has multiple minimum solutions on constraint boundaries. Based on these CMMPs, Ahrari *et al*. [49] designed a constrained multimodal test function with multiple minimum solutions, and the number of minimum solutions is controllable. In order to visualize the search behaviors of MOEAs with constraint handling techniques, Y. Nojima *et al*. [50] designed several constraint Various distance minimization problems (DMPs).

In addition, there are some algorithms are proposed for solving constrained multimodal optimization problems. Kimura *et al*. [51] used Markov chain Monte Carlo (MCMC) method combined with genetic algorithms (GAs) to generate more feasible solutions and help solve constrained multimodal single-objective optimization problems. Zhou *et al*. [52] proposed an adaptive dual-objective particle swarm optimization algorithm to solve the constrained multimodal optimization problems. A new objective was added to deal with the constraints separately. The original single-objective optimization problem is transformed into a dual-objective optimization problem, thus improving the diversity of solutions and obtaining more feasible minimum solutions. In addition, this algorithm dynamically adjusts the inertia weight according to the change rate of the crowding distance in the evolution process to well balance the global and local exploration abilities. Then, the same author embedded the co-evolution mechanism and CDP into glowworm swarm optimization algorithm (GSO) to find feasible solutions and used the local search strategy to improve the diversity of the population [53]. However, there is still a lack of relevant research on constrained multimodal multiobjective optimization, so it is meaningful to carry out this study.

III. Design of Constrained Multimodal Multiobjective optimization test functions

In practical problems, there exists a number of constrained multimodal multiobjective optimization problems that have not been solved yet. However, there are few constrained multimodal multiobjective test functions so far. To carry on related scientific research, we construct a suite of constrained multimodal multiobjective optimization test functions called CMMFs according to the properties in practical problems. The principles, instances, and characteristics of CMMFs are introduced in this section.

A. Design Principle of CMMFs

The main characteristics of CMMFs include: 1) Constraints: The optimization problems contain multiple different constraints. 2) Multimodality: More than one Pareto optimal solution sets are required. 3) Multiobjective: Two or more objectives need to be optimized. Since the mapping relationship between the decision space and objective space is not intuitive, it is difficult to construct constraints and obtain multiple true CPSs. Therefore, to generate test functions with the above properties, we classify CMMFs into four categories. The proposed CMMFs have more than two equivalent CPSs in decision space. The general framework is given as follows:

$$\left\{\begin{array}{l} \min\ f_1(x)=g_1(\theta)\left(1+p\left(x_1,\cdots,x_n\right)^2+q\left(x_{n+1},\cdots,x_m\right)\right) \\ \min\ f_2(x)=g_2(\theta)\left(1+p\left(x_1,\cdots,x_n\right)^2+q\left(x_{n+1},\cdots,x_m\right)\right) \\ \cdots \\ \min\ f_n(x)=g_n(\theta)\left(1+p\left(x_1,\cdots,x_n\right)^2+q\left(x_{n+1},\cdots,x_m\right)\right) \end{array}\right. \tag{5}$$

where $g(\theta)$ is the first function to control the shape of PF that might affect the performance of algorithms [20]. θ is a function related to the decision vectors, and it controls the number of PSs. $p(x_1,\cdots,x_n)^2$ is the second function with the global minimum value 0. It determines the shape of PSs and controls the convergence difficulty of different PSs corresponding to the identical PF. Similarly, $q(x_{n+1},\cdots,x_m)$ is the third function with a global minimum value 0, which is related to decision vectors $x_{n+1},\cdots,x_m$ to increase the scalability and difficulty.

Here, a simple example is given to explain the construction of these test functions. The first step is to construct the unconstrained objective function and obtain the UPF. Suppose that, the equations of $g(\theta)$ and θ are given as follows:

$$\left\{\begin{array}{l} g_1(\theta)=\cos(\frac{\pi}{2}\theta) \\ g_2(\theta)=\sin(\frac{\pi}{2}\theta) \end{array}\right. \tag{6}$$

$$\theta=\left\{\begin{array}{l} \frac{2}{\pi}\arctan\left(\frac{|x_2|}{|x_1|}\right), x_1\in\left(-\frac{1}{2},1\right] \\ \frac{2}{\pi}\arctan\left(\frac{|x_2|}{\left|x_1+\frac{1}{2}\right|}\right), x_1\in\left[-1,-\frac{1}{2}\right] \end{array}\right. \tag{7}$$

The UPF can be obtained by equations (5-7), while $p(x)$ and $q(x)$ getting the minimum value 0. It is worth noting that the PF of (5) relies on $g(\theta))$ [20]. It is shown in Fig. 2(b) .

$$\left\{\begin{array}{l} f_1^2+f_2^2=1 \\ 0\le f_1,f_2\le 1 \end{array}\right. \tag{8}$$

The second step is to construct multiple UPSs by controlling $p(x)$ and $q(x)$. Supposing that $q(x)=(x_3-0.1)^2$, and the equation for $p(x)$ is as follows:

$$p\left(x_1,x_2\right)=\left\{\begin{array}{l} \frac{49}{100}-x_1^2-x_2^2, x_1\in\left(-\frac{1}{2},1\right] \\ \frac{1}{4}-\left(x_1+\frac{1}{2}\right)^2-x_2^2, x_1\in\left[-1,-\frac{1}{2}\right] \end{array}\right. \tag{9}$$

where $x_1,x_2\in[-1,1]$, and $p(x_1,x_2)^2$ can reach the minimum value (i.e.,0), while x_1,x_2 meeting the conditions : $x_1^2+x_2^2=\frac{49}{100}, x_1\le-\frac{1}{2}$ or $(x_1+\frac{1}{2})^2+x_2^2=\frac{1}{4}, x_1>-\frac{1}{2}$. Meanwhile, $q(x)$ gets the minimum value 0, when $x_3=0.1$. In this way, we can obtain two UPSs corresponding to the same UPF by equations (6-9) .

The third step is to construct constraints on the objective functions, thereby narrowing the feasible regions and generating CPF and multiple CPSs. Moreover, after adding constraints, the shapes and positions of CPSs and UPSs will be completely different, and the CPF may also be completely changed. In practice, the impact of decision constraints on the objective function is usually not intuitive, but practical applications are always full of these constraints. Therefore, we add the following easy decision constraints for explaining:

$$\begin{array}{l} c_1=x_1\cdot x_2<0 \\ c_2=x_1^2+x_2^2\le 1, x_1\ge 0 \\ c_3=x_1^2+x_2^2\ge 0.7, x_1\ge 0 \\ c_4=x_1+x_2\ge 0 \\ c_5=x_1+x_2+0.5\le 0 \\ c_6=\left(x_1+0.5\right)^2+x_2^2\le 0.04 \\ c_7=x_1-x_2+0.7\le 0 \end{array} \tag{10}$$

As depicted in Fig. 2(a), the feasible regions are divided into two parts. It is obvious that $p(x_1,x_2)^2$ can only reach 0.0441 instead of the minimum value 0 after adding the decision constraints. The value of θ will be narrowed down to $[0,\frac{1}{2}]$, which may lead to a substantial change of the CPF, as shown in Fig. 2(b). The equation of CPF is given as follows:

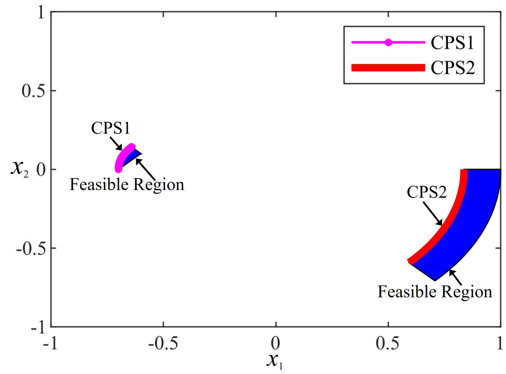
$$\left\{\begin{array}{l} f_1^2+f_2^2=1.0441^2 \\ 1.0441\frac{\sqrt{2}}{2}\le f_1\le 1.0441, 0\le f_2\le 1.0441\frac{\sqrt{2}}{2} \end{array}\right. \tag{11}$$

The above process forms a simple function with the following properties: 1) independent and narrow feasible regions in decision space. 2) entirely different shapes of CPF and CPSs affected by the constraints. 3) multiple CPSs corresponding to the identical CPF.

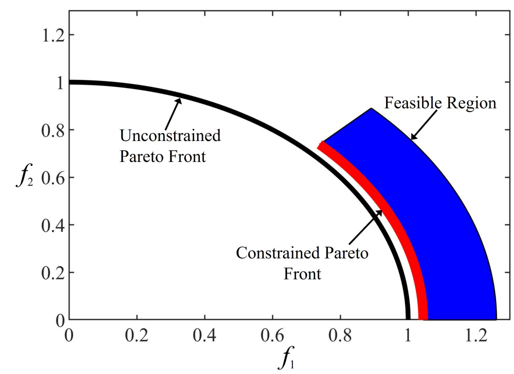
B. Instances and Characteristics of CMMFs

The above framework can construct constrained multimodal multiobjective test functions. In addition, the constrained single objective test functions proposed by Liang *et al.* [54] are used to construct the feasible regions in high-dimension space. Noting that in each instance, we can obtain the true PF after getting the minimum values of $p(x)$ and $q(x)$.

In this paper, 17 CMMFs are generated as instances to compare the proposed algorithm and other competitive algorithms. The detailed information of these test functions is presented in the supplementary materials. It should be noted that there may be differences in convergence difficulties among multiple



(a) Constrained Pareto solution sets in the decision space



(b) Constrained Pareto front in the objective space

Fig. 2. An example of CMMF. (a) Illustration of the discrete feasible regions and CPSs in decision space. The blue regions are feasible regions, and the two arcs are two CPSs. (b) Illustration of the feasible region and CPF in the objective space. The blue region is feasible region in objective space and the red arc segment is CPF. Obviously, the constraints completely separate the CPF from the UPF.

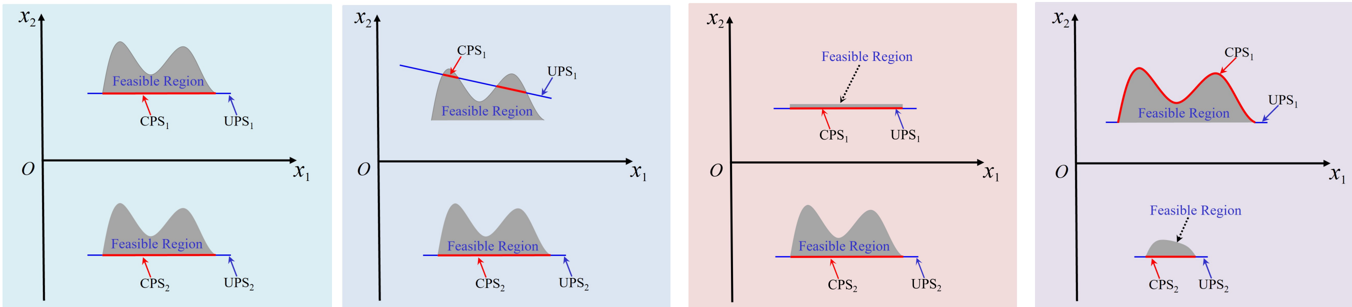
equivalent CPSs and differences in searching difficulty of feasible regions around multiple equivalent CPSs, which may prevent the algorithms from reserving more feasible Pareto optimal solutions. In addition, the properties of feasible regions around different CPSs may be different. For example, constraints may lead to a partial or complete separation of CPSs and UPSs. In this paper, CMMFs are divided into four types according to the properties of feasible regions and CPSs: ***Type I: same CPSs, same feasible regions***, ***Type II: different CPSs, same feasible regions***, ***Type III: same CPSs, different feasible regions***, ***Type IV: different CPSs, different feasible regions***. The four types of CMMFs are described in detail as follows:

1) *Type I (same CPSs, same feasible regions):* As shown in Fig. 3(a), UPS_1 and UPS_2 have the same convergence difficulties in decision space, and the parts of UPS_1 and UPS_2 that overlap with feasible regions are the real CPS_1 and CPS_2. Since the constraints around UPS_1 and UPS_2 are identical, it means that the searching difficulties of CPS_1 and CPS_2 are consistent.
2) *Type II (different CPSs, same feasible regions):* As shown in Fig. 3(b), UPS_1 and UPS_2 have different convergence difficulties and shapes in decision space. The differences between multiple CPSs may lead to different convergence difficulties [40]. However, the constraints around UPS_1 and UPS_2 are consistent, it may lead to different overlaps between UPS_1 and UPS_2 and feasible regions, thus forming CPS_1 and CPS_2 with different shapes and searching difficulties.
3) *Type III (same CPSs, different feasible regions):* As shown in Fig. 3(c), equivalent UPS_1 and UPS_2 have identical convergence difficulties and shapes in decision space. However, the constraints around UPS_1 and UPS_2 are different, the feasible region around UPS_1 is smaller than that around UPS_2. It may lead to a difference in the difficulties of searching the feasible regions around UPS_1 and UPS_2. In addition, it may form different overlaps between UPS_1 and UPS_2 and feasible regions, thus forming CPS_1 and CPS_2 with different shapes and searching difficulties.
4) *Type IV (different CPSs, different feasible regions):* As shown in Fig. 3(d), equivalent UPS_1 and UPS_2 have different shapes and convergence difficulties in decision space, and UPS_2 is much smaller than UPS_1. In addition, the constraints around UPS_1 and UPS_2 are also different, the feasible region around UPS_1 is much larger than that around UPS_2, thus forming CPS_1 and CPS_2 with different shapes and searching difficulties.

Due to the above-mentioned complex properties, finding and maintaining multiple CPSs in decision space for existing MOEAs are not easy. The reasons are explained as follows:

- There may be discrete, distant, and narrow feasible regions in decision space, which cause trouble for algorithms finding the complete CPF in objective space.
- The convergence difficulties among multiple CPSs may be different, which may lead algorithms to trap in some CPSs with smaller convergence difficulty while other CPSs are missing.
- Multiple CPSs may be located in different feasible regions, which may cause the algorithms to converge into the larger feasible region while other small feasible regions with Pareto optimal solutions are missing.
- Most CMOEAs seldom consider the distribution of solutions in decision space, which may make it difficult for them to retain more feasible Pareto optimal solutions.
- Most existing MMOEAs retain many infeasible solutions due to the lack of constrained handing techniques.

The main characteristics of CMMFs are summarized here. First, the proposed problems have multimodality of CPSs, irregularity of CPFs, and narrow feasible regions. Second, the proposed problems are scalable. The shapes and sizes of true CPFs and true CPSs as well as the number of true CPSs can be controlled. Third, the mapping from decision space to the objective space is more intuitive, to the benefit of exploring more properties. These problems can also help researchers to compare and explore the performances of algorithms more deeply and conveniently.



(a) Type I (b) Type II (c) Type III (d) Type IV

Fig. 3. Illustration of four types of CMMFs.

IV. Proposed Method

CMMFs require the algorithms to keep a great balance among diversity, convergence, and feasibility. In addition, simply combining CHTs with specific methods for preserving multiple Pareto optimal solutions fails to efficiently solve CMMOPs. Therefore, we design a novel differential evolution algorithm with specific strategy to address CMMOPs. The algorithm adopts a speciation mechanism and an improved selection criterion to find and maintain more feasible Pareto optimal solutions, named CMMODE. In this section, the procedure of CMMODE is provided and the reasons why it can effectively address CMMOPs are discussed in detail.

A. Procedure of CMMODE

In CMMODE, the initial population is first evaluated by fitness value calculations and all individuals are sorted in ascending order. Speciation mechanism is used to induce multiple niches. Then, an improved environment selection criterion is proposed to screen out feasible and well-distributed solutions for entering the next generation.

The general procedure of CMMODE is presented in Algorithm 1, in which N represents the entire population and $P_t(i)$ represents the i^{th} individual at the t^{th} generation. The population sorted by fitness value is expressed as Sorted_$\boldsymbol{P}_t$, and each niche is induced according to the sorting value. Among them, species seed (j) represents the individual with the least fitness value in each species, which steadily improves the individual quality in its own species.

The specific execution process is given as follows. Firstly, initial population P_0 is generated. The fitness values of all individuals are calculated and arranged in ascending order. The first individual in Sorted_$\boldsymbol{P}_t$ is chosen as the species seed and the nearest $K-1$ individuals around it are selected to form a single species. Then, the generated species are removed from Sorted_$\boldsymbol{P}_t$ and this process will not stop until no individuals exist in Sorted_$\boldsymbol{P}_t$, as shown in lines 6-12. When the ratio of current iteration count to the maximum iteration count is less than R, the algorithm enters the first stage. During this stage, the individuals in each species generate offspring through differential evolution (DE) operator [55] and select individuals to enter the next generation through improved environment selection method. It aims to limit the competition in the population, thus protecting potential individuals in the process of exploring multiple discrete feasible regions, as seen in lines 14-19. In the second stage, after the offspring in all species are generated, the parents and offspring of all species merge into a population. Then $\boldsymbol{P}_{t+1}$ are selected from $\boldsymbol{P}_t$ through improved environment selection, as shown in lines 20-26. The above steps will not stop until the termination condition is satisfied.

Algorithm 1 CMMODE

Input: a CMMOP and the population size N

1: //Initialize Population ($\boldsymbol{P}_0$)
2: Evaluate Population ($\boldsymbol{P}_0$)
3: **while** *Gen* $\leq$ *MaxGen* **do**
4: Calculate the fitness value for $\boldsymbol{P}_t$
5: Sorted_$\boldsymbol{P}_t$ = ascending_sort ($\boldsymbol{P}_t$)
6: //Generate species in $\boldsymbol{P}_t$
7: **for** j = 1: N/K
8: //Select $seeds$ in the species
9: $seed_j$ = The first individual in Sorted_$\boldsymbol{P}_t$
10: **species** (j) = The nearest k individuals to $seed_j$ in Sorted_$\boldsymbol{P}_t$
11: Sorted_$\boldsymbol{P}_t$ = Sorted_$\boldsymbol{P}_t$ $\setminus$ **species** (j)
12: **end for**
13: **If** CMMODE $is\ in\ the\ first\ phase$
14: **for** j = 1: N/K
15: **offspring** (j) = Offspring_Generating (**species** (j))
16: united_**species** (j) = **species** (j) $\bigcup$ **offspring** (j)
17: **species** (j) = Selection (united_**species** (j))
18: **end for**
19: $\boldsymbol{P}_{t+1}$ = **species**
20: // the second phase
21: **else** CMMODE $is\ in\ the\ second\ phase$
22: **for** j=1: N/K
23: **offspring** (j) = Offspring_Generating (**species** (j))
24: **end for**
25: $\boldsymbol{P}_t$ = **species** $\bigcup$ **offspring**
26: $\boldsymbol{P}_{t+1}$ = Selection ($\boldsymbol{P}_t$)
27: **end if**
28: $t \leftarrow t+1$
29: **end while**

Output: Final Population

The algorithm needs to simultaneously consider the convergence, diversity, and feasibility of the population so that a new environmental selection criterion is required. The improved environment selection criterion is implemented in two steps as follows. The first step is to sort the individuals in the population by constrained dominance principle [6]. The second step is to calculate the special crowding distance (SCD) values of all individuals. [43]. On account of the ascending order of the Pareto rankings in the first step, all individuals on the Pareto fronts with different Pareto rankings are successively selected to enter the next generation. This process will continue until the total number of individuals exceeds N while selecting the individuals on a certain Pareto front. This environment selection method is first to drive the population into feasible regions, and then select more dispersed individuals. It aims to reverse more feasible Pareto optimal solutions in discrete feasible regions.

B. Mechanism of CMMODE

This subsection analyzes the effective mechanism in CMMODE. By combining the speciation mechanism with the improved environment selection criterion, the CMMOPs can be effectively solved. The reasons are that many Pareto optimal solutions have identical objective values, and the constraints may cause difficulty in search of feasible regions, as well as the imbalanced convergence difficulty in search of multiple CPSs. Once a Pareto optimal solution is found, it may guide the population through evolution, leading to the loss of other feasible regions. Even if the algorithm finds other distant feasible Pareto optimal solutions, they may be eliminated through the original environment selection, because the solutions with the identical objective values already exist.

In CMMODE, individuals in each species are able to independently evolve with the help of speciation mechanism. The information interactions among all species are reflected in the transformation of the species seeds in each generation, thus improving the diversity of the population. More importantly, it is helpful to solve the imbalanced difficulty problem when searching for multiple different CPSs.

Because of the independent evolution in each niche, which may limit the competition among individuals in the population, the obtained Pareto optimal solution has no inhibition on the survive of other potential solutions. For example, as Fig. 4 shows, the positions of solutions A, B, and C are known. Solution A is a Pareto optimal solution, and Solution B and Solution C are potential solutions around multiple discrete feasible regions in the decision space respectively. Because of the speciation mechanism, Solution B may not directly compete with Solution A, so that it can survive to the next generation to help the population find other feasible regions and retain more feasible Pareto optimal solutions. Although Solutions B and C have identical objective values, there are more Pareto optimal solutions around Solution C in the decision space. Therefore, Solution C might be eliminated by other Pareto optimal solutions and thus is unable to survive to the next generation.

In addition, in the improved environment selection method, constrained dominance principle is taken as the first criterion

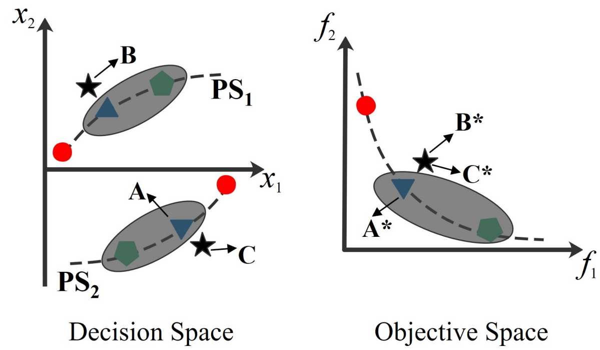


Fig. 4. Illustration of the constrained multimodal multiobjective problem.

and SCD value is taken as the second criterion, so that the population may be evenly distributed across discrete feasible regions. This explains why combining speciation mechanism with the improved environmental selection criterion in CMMODE contributes to solving CMMOPs more efficiently. In Section V-B, the experimental comparisons are conducted to verify the effectiveness of CMMODE.

V. Experiments

A. Performance Metrics and Experiment Settings

The experiments are carried on seventeen constrained multimodal multiobjective optimization test functions, denoted as CMMF1-CMMF17. To compare the performances of other competitive algorithms, this experiment employs three indicators.

1) *Inverted Generational Distance* (*IGD*) [56]: IGD represents the mean minimum Euclidean Distance (ED) from the reference solutions to the obtained solutions. $\mathbf{F}^*$ represents a set of uniformly distributed reference points on the true PF; $\mathbf{X}$ contains all the obtained solutions; and $\mathbf{F}$ contains all objective values of $\mathbf{X}$. The IGD value is calculated as

$$IGD(\mathbf{X}) = \frac{\sum_{f^* \in \mathbf{F}^*} \min\{ED(f^*, \mathbf{F})\}}{|\mathbf{F}^*|} \tag{12}$$

where $\min\ ED\{(f^*, \mathbf{F})\}$ denotes the minimum Euclidean distance between the obtained solutions in $\mathbf{F}$ and f^*. Smaller IGD values are desirable.

2) *Reversed Pareto Sets Proximity* (*RPSP*) [43]: The RPSP value calculation takes account of the cover rate (CR) [43] of the obtained optimal solutions on the true PSs and IGDX values [57], which is employed to evaluate the similarity degree between the obtained PSs and the true PSs. The RPSP value is obtained as follows:

$$IGDX(\mathbf{X}) = \frac{\sum_{x^* \in \mathbf{X}^*} \min\{ED(x^*, \mathbf{X})\}}{|\mathbf{X}^*|} \tag{13}$$

where $\mathbf{X}^*$ is consisted of uniformly distributed reference solutions on the true PSs.

$$CR(\mathbf{X}) = \left(\prod_{i=1}^{D} \eta_i\right)^{\frac{1}{2D}} \tag{14}$$

$$\eta_i = \left\{ \frac{\min\left(x_i^{*,\max}, x_i^{\max}\right) - \max\left(x_i^{*,\min}, x_i^{\min}\right)}{x_i^{*,\max} - x_i^{*,\min}} \right\}^2 \tag{15}$$

where x_i and x_i^* represent the i^{th} obtained solution and reference solution on the true PSs. $\eta_i = 0$ When $x_i^{\max} \leq x_i^{*,\min}$ or $x_i^{\min} \geq x_i^{*,\max}$; $\eta_i = 1$ when $x_i^{*,\max} = x_i^{*,\min}$.

$$rPSP(\mathbf{X}) = \frac{IGDX(\mathbf{X})}{CR(\mathbf{X})} \tag{16}$$

Generally, a smaller rPSP value means that the algorithm performs better in the decision space.

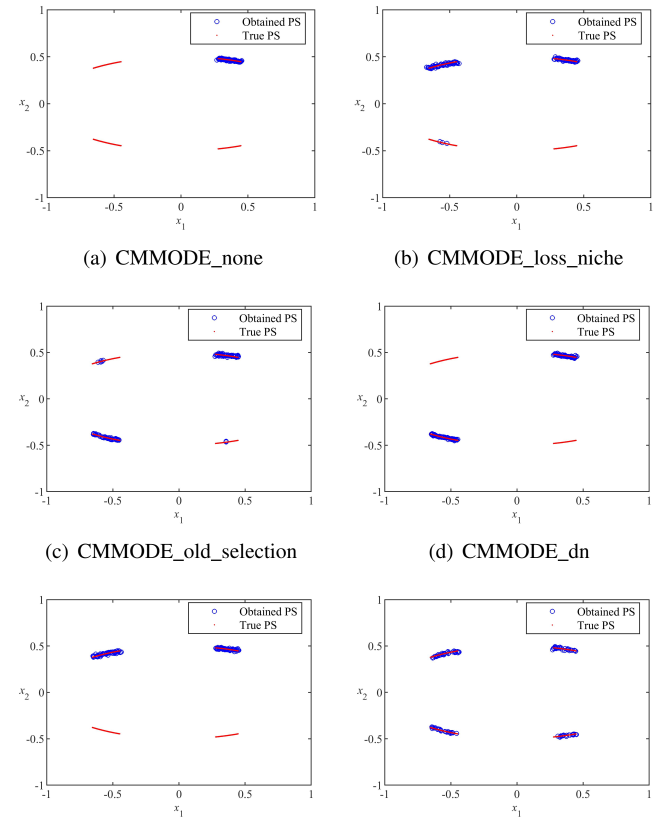
3) *Constrained Pareto Sets Proximity* (*CPSP*) : In order to comprehensively evaluate the convergence, diversity, and, feasibility of the solutions obtained by different algorithms, this paper proposes a new indicator. In particular, the mean CV value of the obtained solutions $\mathbf{X}$ and the mean RPSP value are calculated and normalized respectively. These two values are summed up to comprehensively reflect the performances of different algorithms, called CPSP. Smaller CPSP values are desirable.

To achieve the purpose of unbiased comparison in our experiments, the population size N is set to 100; the maximum evaluation times of all algorithms is set to 20000; and all experiments are conducted for 31 times. In CMMODE, the crossover probability (CR) is randomly selected from [0.3, 0.5, 1], and F is randomly selected from [0.6, 0.8, 1]. The value of K (indicates the size of each niche) is consistent with the original research setting [58]. In addition, the parameter R is set to 0.5. Meanwhile, CMMODE with different R values from [0, 0.1, 0.3, 0.5, 0.7, 0.9, 1] are experimentally compared. The RPSP, IGD, and IGDX values are shown in Table S-III-I, Table-S-III-II, and Table-S-III-III of the supplementary material. All parameters of other algorithms are set in accordance with their corresponding references [59], [60], [35], [61], [43], [12], [62].

B. Experimental validation of CMMODE

To verify the effectiveness of species mechanism and improved environment selection criterion, multiobjective DE algorithms with and without speciation mechanism or improved environment selection method are tested on seventeen CMMFs. In addition, other niche mechanisms are applied for comparative experiments. The detailed RPSP and IGD values of these algorithms are analyzed by running a Wilcoxon ranksum test, as shown in Table I and II respectively. Fig. 5 shows the distribution of solutions obtained by different variants of CMMODE on CMMF12. Fig. 6 shows the boxplots of the statistical results of RPSP values on four different types of test functions.

In Table I, Table II, Fig. 5, and Fig. 6, CMMODE_*none* indicates simple constrained multiobective DE algorithm without speciation mechanism and improved environment selection criterion. CMMODE_*loss_niche* is constrained multiobective DE algorithm with only improved environment selection method.



(a) CMMODE_none (b) CMMODE_loss_niche (c) CMMODE_old_selection (d) CMMODE_dn (e) CMMODE_crowding (f) CMMODE

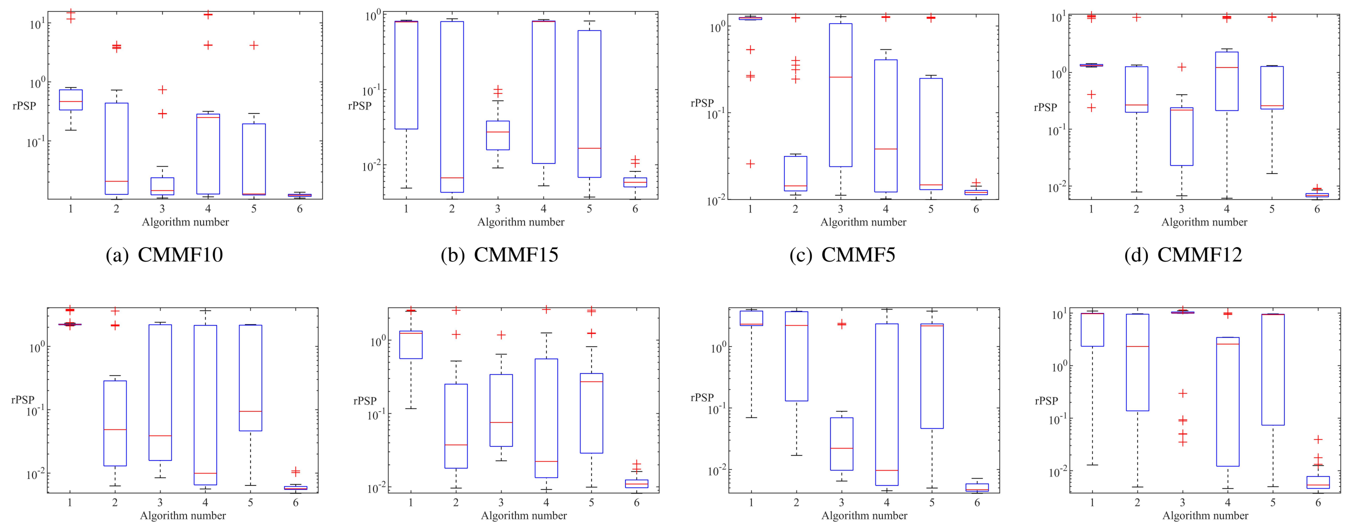
Fig. 5. Comparision of CPSs obtained by different multiobjective DE algotithms on CMMF12.

CMMODE_*old_selection* is constrained multiobective DE algorithm with only speciation mechanism. CMMODE is constrained multiobective DE with both speciation mechanism and improved environment selection. In addition, CMMODE_*dn* and CMMODE_*crowding* are constrained multiobective DE algorithms with improved environment selection, applying different niche strategies respectively [12],[63].

Fig. 5 shows the distribution of the population obtained by six algorithms on CMMF12, which are obtained from the results of the median RPSP values in 31 runs. It can be concluded from Fig. 5 that CMMODE performs best in decision space and finds all the CPSs, while other algorithms fails. The reason why CMMODE_*loss_niche* finds fewer equivalent CPSs is that solutions around different CPSs may compete, leading to the loss of equivalent Pareto optimal solutions. CMMODE_*old_selection* finds all the feasible regions, but the solutions fail to be distributed across the whole CPSs. Because it takes less account of the diversity of solutions in decision space, this algorithm is not conducive.

CMMODE_*dn* and CMMODE_*crowding* are incapable of fully utilizing the potential information of infeasible solutions, so they fail to find all the CPSs located in different feasible regions.

Meanwhile, from Table I and Fig. 6, the RPSP results of ranksum test show that CMMODE is significantly different from other five algorithms, and it also gives the best overall



(a) CMMF10 (b) CMMF15 (c) CMMF5 (d) CMMF12

(e) CMMF8 (f) CMMF11 (g) CMMF9 (h) CMMF14

Fig. 6. Box-plots of *rPSP* values of six algorithms on eight test functions, which represent four types of CMMFs respectively. The numbers on the horizontal axis of each graph represent the following algorithms: 1 = CMMODE_none, 2 = CMMODE_loss_niche, 3 = CMMODE_old_select, 4 = CMMODE_crowding, 5 = CMMODE_dn, 6 = CMMODE.

TABLE I
AVERAGE **RPSP** VALUES OVER 31 RUNS ON THE PROPOSED CMMFS, WHERE THE BEST RESULT FOR EACH CMMF IS SHOWN IN BOLD

Problem	CMMODE_none	CMMODE_loss_niche	CMMODE_old_selection	CMMODE_crowding	CMMODE_dn	CMMODE
CMMF1	2.77e-01 ± 2.69e-01(+)	**8.59e-02 ± 1.06e-01(-)**	3.72e-01 ± 1.65e-01(+)	5.59e-01 ± 3.02e-01(+)	1.15e-01 ± 1.08e-01(=)	3.86e-02 ± 6.47e-03
CMMF2	7.91e-01 ± 1.84e-01(+)	3.27e-02 ± 5.89e-03(+)	2.82e-01 ± 2.81e-01(+)	3.92e-02 ± 2.17e-02(+)	3.21e-02 ± 6.18e-03(=)	**8.59e-02 ± 1.06e-01**
CMMF3	4.58e-01 ± 7.47e-01(+)	3.20e-02 ± 5.09e-02(=)	5.87e-01 ± 1.07e+00(+)	1.69e-01 ± 2.89e-01(+)	4.28e-02 ± 1.32e-01(=)	**1.58e-02 ± 1.76e-03**
CMMF4	4.12e-01 ± 2.02e-01(+)	2.17e-02 ± 1.76e-03(+)	1.96e-01 ± 1.95e-01(+)	2.97e-02 ± 2.02e-02(+)	2.17e-02 ± 1.69e-03(+)	**2.03e-02 ± 1.46e-03**
CMMF5	1.05e+00 ± 3.78e-01(+)	1.42e-01 ± 3.16e-01(+)	4.52e-01 ± 5.06e-01(+)	2.95e-01 ± 4.16e-01(+)	2.35e-01 ± 4.10e-01(+)	**1.21e-02 ± 1.19e-03**
CMMF6	7.58e-01 ± 6.77e-01(+)	**3.06e-02 ± 2.86e-02(=)**	1.05e-01 ± 1.18e-01(+)	6.08e-02 ± 4.94e-02(+)	6.46e-02 ± 4.51e-02(+)	3.07e-02 ± 2.26e-02
CMMF7	2.62e+00 ± 7.81e-01(+)	2.19e-02 ± 1.21e-02(+)	2.68e+00 ± 6.49e-01(+)	3.67e-01 ± 8.93e-01(+)	1.79e-02 ± 1.45e-03(+)	**1.67e-02 ± 1.64e-03**
CMMF8	2.41e+00 ± 5.26e-01(+)	5.70e-01 ± 1.01e+00(+)	8.10e-01 ± 1.07e+00(+)	8.07e-01 ± 1.22e+00(+)	1.00e+00 ± 1.07e+00(+)	**6.04e-03 ± 1.29e-03**
CMMF9	2.74e+00 ± 9.17e-01(+)	1.97e+00 ± 1.47e+00(+)	2.49e-01 ± 6.87e-01(+)	1.21e+00 ± 1.51e+00(+)	1.49e+00 ± 1.43e+00(+)	**5.10e-03 ± 9.71e-04**
CMMF10	1.32e+00 ± 3.20e+00(+)	8.49e-01 ± 1.55e+00(+)	5.76e-02 ± 1.43e-01(+)	1.28e+00 ± 3.48e+00(+)	2.06e-01 ± 7.39e-01(+)	**1.20e-02 ± 6.52e-04**
CMMF11	1.27e+00 ± 7.52e-01(+)	2.21e-01 ± 4.93e-01(+)	2.26e-01 ± 2.67e-01(+)	3.57e-01 ± 5.97e-01(+)	4.09e-01 ± 6.44e-01(+)	**1.16e-02 ± 2.66e-03**
CMMF12	2.55e+00 ± 3.11e+00(+)	8.97e-01 ± 1.66e+00(+)	1.93e-01 ± 2.29e-01(+)	2.66e+00 ± 3.73e+00(+)	1.45e+00 ± 2.71e+00(+)	**7.06e-03 ± 8.47e-04**
CMMF13	1.07e+00 ± 1.54e+00(+)	1.80e-01 ± 2.86e-01(+)	9.80e-02 ± 1.28e-01(+)	1.16e+00 ± 1.86e+00(+)	4.17e-01 ± 9.60e-01(+)	**1.19e-02 ± 5.15e-03**
CMMF14	7.61e+00 ± 3.71e+00(+)	4.38e+00 ± 4.27e+00(+)	8.40e+00 ± 4.15e+00(+)	3.27e+00 ± 3.72e+00(+)	5.24e+00 ± 4.76e+00(+)	**7.75e-03 ± 6.67e-03**
CMMF15	4.57e-01 ± 3.92e-01(+)	2.94e-01 ± 3.91e-01(=)	3.22e-02 ± 2.20e-02(+)	4.55e-01 ± 4.00e-01(+)	2.20e-01 ± 3.51e-01(+)	**6.05e-03 ± 1.68e-03**
CMMF16	2.78e+00 ± 8.12e-01(+)	2.09e+00 ± 1.15e+00(+)	1.14e+00 ± 9.88e-01(+)	2.57e+00 ± 1.27e+00(+)	2.69e+00 ± 7.06e-01(+)	**1.39e-01 ± 8.46e-02**
CMMF17	5.32e-01 ± 4.71e-01(+)	3.18e-01 ± 2.77e-01(+)	2.14e-01 ± 1.56e-01(+)	6.77e-01 ± 8.01e-01(+)	3.12e-01 ± 1.80e-01(+)	**6.07e-02 ± 2.04e-02**
+/=/-	17/0/0	13/3/1	17/0/0	17/0/0	14/3/0	

performance in objective space, which indicates that speciation mechanism and improved environment selection method can effectively improve performance. The reasons are that the speciation mechanism can combine with CDP well, which enables the algorithm to find and maintain more feasible Pareto optimal solutions. Each niche separately evolves to avoid global competition and has a niche center named species seed. If species seeds can be found in different feasible regions, it is more likely to guide the population to find the Pareto optimal solutions located in multiple different feasible regions. In the improved environment selection method, the algorithm preferentially selects more distant solutions in the feasible regions to enhance diversity.

C. Comparison With Other Algorithms

To verify the effectiveness of CMMODE for solving CMMOPs, we compare CMMODE with three state-of-the-art CMOEAs and three excellent MMOEAs on 17 test functions. Among them, the three CMOEAs are C-TAEA [64], MOEA/D-DAE [61] and PPS [35], and they use different strategies to solve CMOPs. C-TAEA uses dual-archive strategy to help the population evolve. Convergence archive preserves

TABLE II
AVERAGE **IGD** VALUES OVER 31 RUNS ON THE PROPOSED CMMFS, WHERE THE BEST RESULT FOR EACH CMMF IS SHOWN IN BOLD

Problem	CMMODE_none	CMMODE_loss_niche	CMMODE_old_selection	CMMODE_crowding	CMMODE_dn	CMMODE
CMMF1	1.10e-02 ± 8.03e-03(+)	**5.48e-03 ± 1.44e-03(=)**	7.84e-03 ± 1.48e-02(+)	3.88e-02 ± 3.06e-02(+)	8.10e-03 ± 5.45e-03(+)	5.54e-03 ± 3.63e-03
CMMF2	7.35e-03 ± 1.99e-03(+)	5.02e-03 ± 2.49e-04(-)	6.58e-03 ± 1.98e-03(+)	5.08e-03 ± 2.47e-04(-)	**4.99e-03 ± 1.91e-04(-)**	5.24e-03 ± 3.01e-04
CMMF3	1.93e-02 ± 1.22e-02(+)	8.90e-03 ± 1.01e-02(+)	8.62e-03 ± 4.61e-03(+)	1.19e-02 ± 1.25e-02(+)	7.98e-0 3± 7.76e-03(+)	**4.99e-03 ± 2.00e-04**
CMMF4	1.03e-02 ± 1.19e-02(=)	5.24e-03 ± 2.32e-04(-)	6.52e-03 ± 2.03e-03(=)	5.57e-03 ± 3.51e-04(=)	**5.22e-03 ± 2.34e-04(-)**	5.45e-03 ± 3.72e-04
CMMF5	**2.42e-03 ± 2.12e-04(-)**	2.78e-03 ± 1.41e-04(=)	2.53e-03 ± 1.83e-04(-)	2.71e-03 ± 1.59e-04(=)	2.71e-03 ± 1.30e-04(=)	2.77e-03 ± 1.71e-04
CMMF6	1.82e-03 ± 1.60e-04(=)	1.80e-03 ± 5.68e-05(=)	**1.78e-03 ± 1.64e-04(-)**	1.82e-03 ± 7.72e-05(=)	1.82e-03 ± 6.84e-05(=)	1.83e-03 ± 7.72e-05
CMMF7	1.85e-02 ± 1.57e-02(+)	6.13e-03 ± 4.18e-04(+)	1.22e-02 ± 3.24e-03(+)	5.91e-03 ± 4.42e-04(=)	6.00e-03 ± 3.42e-04(=)	**5.80e-03 ± 3.92e-04**
CMMF8	1.89e-03 ± 1.84e-04(+)	1.89e-03 ± 9.47e-05(+)	1.99e-03 ± 4.03e-04(+)	1.92e-03 ± 2.34e-04(+)	1.84e-03 ± 6.96e-05(+)	**1.79e-03 ± 6.59e-05**
CMMF9	1.96e-03 ± 4.35e-04(=)	1.91e-03 ± 4.08e-04(+)	2.34e-03 ± 8.49e-04(+)	2.41e-03 ± 1.53e-03(+)	1.97e-03 ± 6.23e-04(+)	**1.75e-03 ± 6.49e-05**
CMMF10	3.59e-02 ± 1.10e-01(+)	8.89e-02 ± 1.77e-01(=)	**2.40e-03 ± 1.27e-04(-)**	6.10e-02 ± 1.50e-01(=)	1.90e-02 ± 7.95e-02(=)	2.86e-03 ± 1.65e-04
CMMF11	5.25e-03 ± 6.20e-03(+)	4.16e-03 ± 5.99e-04(=)	**3.98e-03 ± 3.37e-03(-)**	1.02e-02 ± 1.13e-02(+)	6.16e-03 ± 9.49e-03(=)	4.12e-03 ± 9.51e-04
CMMF12	1.06e-02 ± 2.02e-02(=)	7.01e-03 ± 1.57e-02(+)	**1.84e-03 ± 2.27e-04(-)**	1.72e-02 ± 2.43e-02(+)	7.02e-03 ± 1.59e-02(=)	1.88e-03 ± 8.80e-05
CMMF13	6.21e-02 ± 1.52e-01(+)	4.49e-03 ± 2.76e-03(+)	3.50e-03 ± 1.13e-03(=)	5.71e-02 ± 1.59e-01(+)	2.30e-02 ± 9.11e-02(+)	**3.24e-03 ± 2.99e-04**
CMMF14	3.83e-03 ± 2.24e-03(=)	3.38e-03 ± 1.73e-03(-)	6.36e-03 ± 5.51e-03(+)	1.16e-02 ± 1.43e-02(=)	**2.76e-03 ± 2.51e-04(-)**	4.01e-03 ± 2.63e-03
CMMF15	3.58e-03 ± 6.45e-04(=)	3.75e-03 ± 7.96e-04(=)	3.63e-03 ± 3.73e-04(=)	3.45e-03 ± 4.44e-04(=)	3.77e-03 ± 8.04e-04(=)	**3.45e-03 ± 3.37e-04**
CMMF16	3.17e-02 ± 2.20e-02(+)	1.48e-02 ± 1.71e-03(-)	2.17e-02 ± 6.79e-03(+)	6.54e-02 ± 7.63e-02(+)	**1.43e-02 ± 6.62e-04(-)**	1.53e-02 ± 9.92e-04
CMMF17	3.67e-02 ± 5.62e-02(+)	1.41e-02 ± 5.04e-03(+)	2.78e-02 ± 2.83e-02(+)	7.33e-02 ± 1.14e-01(+)	1.16e-02 ± 1.29e-03(=)	**1.14e-02 ± 1.65e-03**
+/=/-	10/6/1	7/6/4	9/3/5	9/7/1	5/8/4	

TABLE III
AVERAGE **RPSP** VALUES OVER 31 RUNS ON THE PROPOSED CMMFS, WHERE THE BEST RESULT FOR EACH CMMF IS SHOWN IN BOLD

Problem	CTAEA	MOEADDAE	PPS	C_MO_Ring_PSO_SCD	C_MMOEA/DC	C_DN-NSGA-II	CMMODE
CMMF1	5.60e-01 ± 2.90e-02(+)	6.91e-01 ± 1.03e-01(+)	9.67e-01 ± 5.11e-01(+)	6.05e-01 ± 5.05e-02(+)	5.20e-01 ± 3.66e-02(+)	6.44e-01 ± 1.58e-01(+)	**3.86e-02 ± 6.47e-03**
CMMF2	2.69e-01 ± 2.71e-01(+)	2.67e-01 ± 2.37e-01(+)	9.01e-02 ± 5.77e-02(+)	2.95e-02 ± 3.61e-03(=)	**2.82e-02 ± 4.53e-03(=)**	3.67e-02 ± 1.50e-02(=)	2.95e-02 ± 4.53e-03
CMMF3	7.95e-02 ± 2.59e-01(+)	3.70e-01 ± 4.86e-01(+)	7.48e-02 ± 2.10e-01(+)	1.97e-02 ± 4.05e-03(+)	4.29e-02 ± 1.02e-02(+)	4.14e-01 ± 6.39e-01(+)	**1.58e-02 ± 1.76e-03**
CMMF4	4.59e-02 ± 2.03e-02(+)	1.16e-01 ± 6.80e-02(+)	9.90e-02 ± 5.93e-02(+)	2.73e-02 ± 2.63e-03(+)	2.41e-02 ± 2.67e-03(+)	4.04e-01 ± 5.18e-02(+)	**2.03e-02 ± 1.46e-03**
CMMF5	5.85e-01 ± 5.13e-01(+)	5.21e-01 ± 4.63e-01(+)	2.40e-01 ± 2.29e-01(+)	1.56e-02 ± 1.27e-03(+)	2.24e-02 ± 3.09e-03(+)	7.34e-02 ± 1.03e-01(+)	**1.21e-02 ± 1.19e-03**
CMMF6	4.75e-02 ± 1.12e-01(+)	5.47e-02 ± 9.25e-02(=)	2.33e-02 ± 2.05e-02(-)	1.73e-02 ± 2.01e-03(-)	**1.59e-02 ± 1.43e-03(-)**	1.81e-02 ± 1.91e-03(-)	3.07e-02 ± 2.26e-02
CMMF7	2.09e+00 ± 9.83e-01(+)	9.52e-01 ± 1.24e+00(+)	4.05e-01 ± 8.67e-01(+)	1.84e-02 ± 1.75e-03(+)	1.99e-02 ± 1.73e-03(+)	2.14e-02 ± 3.31e-03(+)	**1.67e-02 ± 1.64e-03**
CMMF8	9.75e-02 ± 3.77e-01(+)	1.05e+00 ± 1.13e+00(+)	9.80e-01 ± 1.05e+00(+)	7.62e-03 ± 5.31e-04(+)	9.67e-03 ± 1.41e-03(+)	9.41e-02 ± 4.85e-01(+)	**6.04e-03 ± 1.29e-03**
CMMF9	7.47e-01 ± 1.25e+00(+)	8.15e-01 ± 1.32e+00(+)	1.52e+00 ± 1.29e+00(+)	6.90e-03 ± 7.94e-04(+)	4.25e-01 ± 8.47e-01(+)	1.68e-01 ± 9.02e-01(+)	**5.10e-03 ± 9.71e-04**
CMMF10	6.71e-01 ± 2.67e+00(+)	2.72e-01 ± 2.17e-01(+)	2.77e-01 ± 2.65e-01(+)	1.43e-02 ± 1.88e-03(+)	1.41e-02 ± 1.02e-03(+)	6.94e-01 ± 8.42e-01(+)	**1.20e-02 ± 6.52e-04**
CMMF11	1.10e+00 ± 8.61e-01(+)	5.67e-01 ± 6.72e-01(+)	2.96e-01 ± 3.42e-01(+)	1.51e-02 ± 2.84e-03(+)	2.22e-02 ± 4.85e-02(=)	1.66e-01 ± 1.98e-01(+)	**1.16e-02 ± 2.66e-03**
CMMF12	1.58e-01 ± 3.90e-01(+)	3.04e-01 ± 4.78e-01(+)	3.77e-01 ± 5.09e-01(+)	1.07e-01 ± 2.34e-01(+)	2.52e-01 ± 4.42e-01(+)	2.78e-01 ± 5.68e-01(+)	**7.06e-03 ± 8.47e-04**
CMMF13	1.31e-01 ± 2.68e-01(+)	6.31e-01 ± 4.84e-01(+)	3.48e-01 ± 8.64e-02(+)	7.17e-02 ± 4.97e-02(+)	2.01e-02 ± 5.74e-02(=)	3.14e-01 ± 2.79e-01(+)	**1.19e-02 ± 5.15e-03**
CMMF14	1.29e-01 ± 3.83e-01(+)	3.88e+00 ± 4.86e+00(+)	5.61e+00 ± 5.10e+00(+)	1.56e-02 ± 7.23e-03(+)	**4.92e-03 ± 4.59e-04(-)**	1.04e+01 ± 6.31e+00(+)	7.75e-03 ± 6.67e-03
CMMF15	6.54e-01 ± 1.69e-02(+)	9.09e-01 ± 5.33e-01(+)	7.86e-01 ± 2.38e-01(+)	6.71e-01 ± 1.08e-02(+)	6.50e-01 ± 3.83e-02(+)	6.89e-01 ± 2.28e-02(+)	**6.05e-03 ± 1.68e-03**
CMMF16	2.25e+00 ± 1.28e+00(+)	1.51e+00 ± 1.49e+00(+)	1.74e+00 ± 1.17e+00(+)	1.89e-01 ± 1.67e-01(=)	4.98e-01 ± 6.42e-01(+)	2.23e+00 ± 1.25e+00(+)	**1.39e-01 ± 8.46e-02**
CMMF17	5.01e-02 ± 2.14e-02(-)	2.13e-01 ± 1.30e-01(+)	3.34e-01 ± 1.75e-01(+)	**3.86e-02 ± 6.47e-03(-)**	3.92e-02 ± 3.02e-02(-)	1.03e-01 ± 1.69e-01(=)	6.07e-02 ± 2.04e-02
+/=/-	16/0/1	16/1/0	16/0/1	13/2/2	11/3/3	14/2/1	

the best individuals in the contemporary population to help the population converge to the feasible regions, while diversity archive preserves more widely distributed solutions. MOEA/D-DAE is a constrained multiobjective algorithm, which proposes a strategy to identify whether the population traps into sub-optimal small feasible regions.

In addition, three MMOEAs are MO_Ring_PSO_SCD [43], MMOEA/DC [62], and DN-NSGA-II [12], which are designed on account of Particle Swarm Optimization Algorithm (PSO) [65], DE operators [55], and Genetic Algorithm (GA) [66] respectively. It should be noted that, to ensure the effectiveness of indicators, three MMOEAs are all employed CDP method [6], which is the same as CMMODE.

The detailed RPSP, CPSP, IGD values of these algorithms are analyzed by running a Wilcoxon ranksum test, as shown in Table III, IV, and V. The results of ranksum test in Table IV show that CMMODE achieves the best 13 results in CMMF1-17.

However, the average CPSP values of CMMODE on CMMF2, CMMF6, CMMF11, CMMF17 are not the lowest. Because there exists larger feasible regions in CMMF2 and CMMF6, C_MO_Ring_PSO_SCD can reverse more feasible Pareto optimal solutions. In CMMF11 and CMMF17, their CPSs are discrete and difficult for searching. And it is evident that the performance of C_MO_Ring_PSO_SCD, C_MMOEA/DC and C_DN-NSGA-II on most test functions

TABLE IV
AVERAGE **CPSP** VALUES OVER 31 RUNS ON THE PROPOSED CMMFS, WHERE THE BEST RESULT FOR EACH CMMF IS SHOWN IN BOLD

Problem	CTAEA	MOEADDAE	PPS	C_MO_Ring_PSO_SCD	C_MMOEA/DC	C_DN-NSGA-II	CMMODE
CMMF1	6.94e-01 ± 4.06e-02(+)	8.46e-01 ± 1.29e-01(+)	1.00e+00 ± 2.46e-01(+)	7.38e-01 ± 5.00e-02(+)	1.66e+00 ± 5.56e+00(+)	8.45e-01 ± 2.33e-01(+)	**1.14e-01 ± 9.62e-02**
CMMF2	9.76e-01 ± 9.04e-01(+)	1.00e+00 ± 7.42e-01(+)	2.73e-01 ± 2.18e-01(+)	**1.04e-01 ± 1.61e-02(=)**	1.10e+00 ± 4.52e+00(=)	1.29e-01 ± 4.75e-02(=)	1.13e-01 ± 2.01e-02
CMMF3	4.93e-01 ± 1.07e+00(+)	1.00e+00 ± 1.49e+00(+)	2.75e-01 ± 7.80e-01(+)	3.17e-02 ± 3.52e-03(+)	1.09e+00 ± 4.53e+00(+)	3.01e-01 ± 5.18e-01(+)	**2.75e-02 ± 3.15e-03**
CMMF4	1.20e-01 ± 5.53e-02(+)	3.84e-01 ± 2.94e-01(+)	2.48e-01 ± 2.07e-01(+)	7.25e-02 ± 8.06e-03(+)	6.07e-02 ± 6.36e-03(=)	1.00e+00 ± 1.91e-01(+)	**5.64e-02 ± 3.03e-03**
CMMF5	1.00e+00 ± 8.10e-01(+)	6.27e-01 ± 6.38e-01(+)	5.24e-01 ± 6.51e-01(+)	2.96e-02 ± 1.82e-03(+)	1.04e+00 ± 2.34e+00(+)	8.19e-02 ± 1.48e-01(+)	**2.21e-02 ± 1.42e-03**
CMMF6	8.61e-01 ± 1.51e+00(+)	1.00e+00 ± 2.41e+00(=)	4.30e-01 ± 1.10e-01(-)	**3.84e-01 ± 4.43e-02(-)**	1.37e+00 ± 5.57e+00(+)	4.23e-01 ± 4.10e-02(-)	8.03e-01 ± 6.52e-01
CMMF7	1.00e+00 ± 3.95e-01(+)	4.72e-01 ± 5.61e-01(+)	6.73e-02 ± 2.05e-01(+)	8.24e-03 ± 7.25e-04(+)	1.01e+00 ± 4.40e-01(+)	9.47e-03 ± 1.29e-03(+)	**7.32e-03 ± 6.28e-04**
CMMF8	1.64e-01 ± 4.39e-01(+)	6.45e-01 ± 7.75e-01(+)	1.00e+00 ± 7.56e-01(+)	5.28e-03 ± 3.57e-04(+)	1.01e+00 ± 1.07e+00(+)	1.11e-01 ± 4.11e-01(+)	**3.98e-03 ± 4.02e-04**
CMMF9	5.70e-01 ± 7.31e-01(+)	5.54e-01 ± 6.39e-01(+)	1.00e+00 ± 7.07e-01(+)	4.03e-03 ± 6.80e-04(+)	1.28e+00 ± 8.98e-01(+)	4.76e-01 ± 1.28e+00(+)	**3.08e-03 ± 1.03e-03**
CMMF10	9.46e-01 ± 2.18e+00(+)	5.39e-01 ± 3.49e-01(+)	5.34e-01 ± 4.80e-01(+)	2.63e-02 ± 2.62e-03(+)	1.03e+00 ± 5.51e-01(+)	1.00e+00 ± 4.41e-01(+)	**2.22e-02 ± 1.90e-03**
CMMF11	1.00e+00 ± 7.73e-01(+)	4.01e-01 ± 5.74e-01(+)	2.74e-01 ± 2.78e-01(+)	**1.14e-02 ± 2.39e-03(-)**	1.04e+00 ± 2.38e+00(+)	1.66e-01 ± 3.49e-01(+)	1.40e-02 ± 3.55e-02
CMMF12	2.34e-01 ± 5.44e-01(+)	1.00e+00 ± 9.02e-01(+)	4.28e-01 ± 5.19e-01(+)	9.56e-02 ± 1.28e-01(+)	1.28e+00 ± 1.18e+00(+)	7.40e-01 ± 1.01e+00(+)	**1.93e-02 ± 5.10e-02**
CMMF13	1.90e-01 ± 2.77e-01(+)	1.00e+00 ± 8.60e-01(+)	7.13e-01 ± 5.29e-01(+)	4.41e-02 ± 1.10e-02(+)	1.10e+00 ± 1.43e+00(+)	1.49e-01 ± 2.26e-01(=)	**2.64e-02 ± 1.01e-02**
CMMF14	1.02e+00 ± 6.81e-01(+)	6.73e-01 ± 8.87e-01(+)	1.00e+00 ± 8.90e-01(+)	1.16e-03 ± 3.34e-04(=)	4.74e-02 ± 2.73e-02(+)	5.20e-01 ± 1.52e+00(+)	**1.15e-03 ± 5.09e-04**
CMMF15	1.14e+00 ± 6.78e-01(+)	4.07e-01 ± 9.19e-01(+)	1.00e+00 ± 1.64e+00(+)	3.42e-02 ± 1.18e-02(+)	2.60e-02 ± 5.38e-03(=)	1.87e-01 ± 6.58e-01(+)	**2.98e-02 ± 9.46e-03**
CMMF16	7.08e-01 ± 4.69e-01(+)	5.52e-01 ± 5.07e-01(+)	5.90e-01 ± 3.95e-01(+)	1.51e-01 ± 2.15e-01(+)	1.20e+00 ± 1.91e+00(+)	1.00e+00 ± 2.78e-01(+)	**3.87e-02 ± 2.61e-02**
CMMF17	1.26e-01 ± 4.54e-02(-)	8.10e-01 ± 6.55e-01(+)	1.00e+00 ± 5.17e-01(+)	**1.11e-01 ± 1.85e-02(-)**	1.17e+00 ± 1.73e+00(=)	2.02e-01 ± 1.15e-01(=)	1.93e-01 ± 4.53e-02
+/=/-	16/0/1	16/1/0	16/0/1	12/2/3	13/4/0	13/3/1	

TABLE V
AVERAGE **IGD** VALUES OVER 31 RUNS ON THE PROPOSED CMMFS, WHERE THE BEST RESULT FOR EACH CMMF IS SHOWN IN BOLD

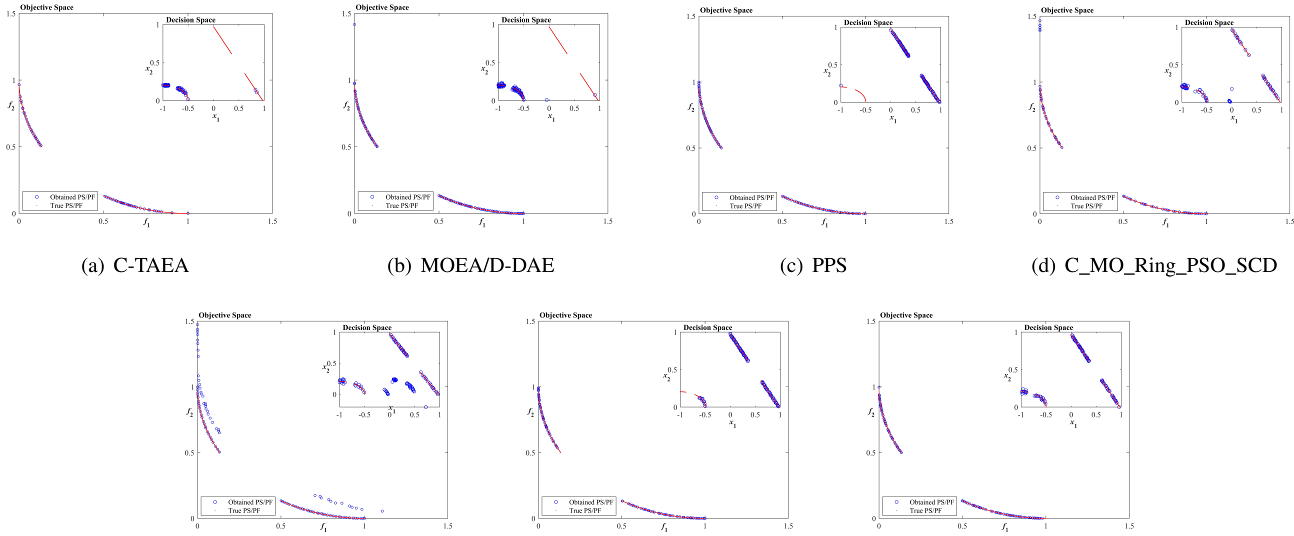
Problem	CTAEA	MOEADDAE	PPS	C_MO_Ring_PSO_SCD	C_MMOEA/DC	C_DN-NSGA-II	CMMODE
CMMF1	2.75e-01 ± 5.54e-04(+)	2.77e-01 ± 1.49e-03(+)	2.72e-01 ± 1.69e-04(+)	2.76e-01 ± 6.27e-04(+)	2.78e-01 ± 1.19e-03(+)	2.74e-01 ± 4.21e-04(+)	**5.54e-03 ± 3.63e-03**
CMMF2	**4.52e-03 ± 1.39e-04(-)**	5.08e-03 ± 2.61e-04(-)	6.94e-03 ± 1.00e-03(+)	7.33e-03 ± 7.66e-04(+)	5.49e-03 ± 1.07e-03(=)	7.10e-03 ± 6.68e-04(+)	5.24e-03 ± 3.01e-04
CMMF3	1.02e-02 ± 8.75e-03(+)	1.00e-02 ± 1.05e-02(=)	1.10e-02 ± 1.01e-02(+)	7.50e-03 ± 7.60e-04(+)	1.41e-02 ± 1.81e-03(+)	1.82e-02 ± 1.34e-02(+)	**4.99e-03 ± 2.00e-04**
CMMF4	7.03e-03 ± 6.22e-04(+)	**4.94e-03 ± 2.21e-04(-)**	6.32e-03 ± 7.12e-04(+)	7.30e-03 ± 1.06e-03(+)	5.88e-03 ± 1.25e-03(=)	6.30e-03 ± 4.10e-04(+)	5.45e-03 ± 3.72e-04
CMMF5	2.92e-03 ± 2.86e-04(+)	**2.34e-03 ± 8.73e-05(-)**	2.58e-03 ± 1.05e-04(-)	3.53e-03 ± 2.53e-04(+)	4.51e-03 ± 5.91e-04(+)	3.46e-03 ± 2.87e-04(+)	2.77e-03 ± 1.71e-04
CMMF6	**1.37e-03 ± 8.36e-06(-)**	1.59e-03 ± 4.30e-05(-)	1.68e-03 ± 3.68e-05(-)	2.46e-03 ± 2.83e-04(+)	1.62e-03 ± 2.54e-05(-)	2.84e-03 ± 3.41e-04(+)	1.83e-03 ± 7.72e-05
CMMF7	6.67e-03 ± 3.61e-04(+)	**5.06e-03 ± 3.10e-04(-)**	5.84e-03 ± 5.31e-04(=)	9.57e-03 ± 2.13e-03(+)	5.15e-03 ± 1.84e-04(-)	1.01e-02 ± 9.52e-04(+)	5.80e-03 ± 3.92e-04
CMMF8	**1.44e-03 ± 4.59e-05(-)**	1.68e-03 ± 5.20e-05(-)	1.88e-03 ± 4.99e-05(+)	2.43e-03 ± 1.65e-04(+)	3.05e-03 ± 5.26e-04(+)	2.60e-03 ± 1.59e-03(+)	1.79e-03 ± 6.59e-05
CMMF9	**1.59e-03 ± 3.59e-04(-)**	1.65e-03 ± 7.11e-05(-)	1.76e-03 ± 6.45e-05(=)	2.96e-03 ± 6.07e-04(+)	1.63e-03 ± 3.29e-05(-)	1.67e-02 ± 3.92e-02(+)	1.75e-03 ± 6.49e-05
CMMF10	3.16e-02 ± 1.10e-01(+)	3.15e-03 ± 4.11e-03(+)	2.48e-03 ± 9.65e-05(-)	3.46e-03 ± 5.72e-04(+)	2.46e-03 ± 1.10e-04(-)	4.05e-03 ± 1.19e-03(+)	**2.86e-03 ± 1.65e-04**
CMMF11	2.78e-02 ± 8.62e-04(+)	2.50e-02 ± 6.88e-04(+)	2.45e-02 ± 1.11e-04(+)	2.75e-02 ± 9.30e-04(+)	3.13e-02 ± 9.15e-03(+)	2.76e-02 ± 3.06e-03(+)	**4.12e-03 ± 9.51e-04**
CMMF12	3.12e-03 ± 9.55e-03(+)	2.43e-03 ± 3.01e-03(+)	4.20e-03 ± 9.28e-03(=)	2.41e-03 ± 2.79e-04(+)	2.87e-03 ± 4.57e-04(+)	7.95e-03 ± 1.58e-02(+)	**1.88e-03 ± 8.80e-05**
CMMF13	5.17e-03 ± 2.67e-03(+)	**2.98e-03 ± 2.58e-04(-)**	3.26e-03 ± 1.59e-04(=)	1.01e-02 ± 2.81e-03(+)	5.08e-03 ± 3.47e-04(+)	4.50e-03 ± 5.49e-04(+)	3.24e-03 ± 2.99e-04
CMMF14	3.07e-02 ± 2.37e-02(+)	5.14e-03 ± 1.45e-03(+)	2.57e-03 ± 1.18e-04(-)	3.99e-03 ± 4.27e-04(-)	**2.81e-03 ± 1.29e-04(-)**	2.06e-02 ± 7.23e-02(+)	4.01e-03 ± 2.63e-03
CMMF15	4.18e-02 ± 2.69e-02(+)	1.21e-02 ± 3.04e-03(+)	4.69e-03 ± 7.71e-04(+)	5.72e-03 ± 7.73e-04(+)	5.61e-03 ± 6.25e-04(+)	5.29e-03 ± 8.05e-04(+)	**3.45e-03 ± 3.37e-04**
CMMF16	5.68e-02 ± 5.93e-02(+)	1.10e-01 ± 7.77e-02(+)	3.34e-02 ± 2.84e-03(+)	3.05e-02 ± 3.13e-03(+)	3.27e-02 ± 4.17e-03(+)	5.89e-02 ± 5.39e-02(+)	**1.53e-02 ± 9.92e-04**
CMMF17	1.93e-02 ± 3.79e-03(+)	1.77e-02 ± 4.61e-03(+)	1.27e-02 ± 2.18e-03(+)	1.63e-02 ± 1.73e-03(+)	1.33e-02 ± 1.78e-03(+)	1.75e-02 ± 1.20e-02(+)	**1.14e-02 ± 1.65e-03**
+/=/-	13/0/4	8/1/8	9/4/4	16/0/1	10/2/5	17/0/0	

are better than C-TAEA, MOEA/D-DAE and PPS.

In addition, the reasons why C-TAEA, MOEA/D-DAE and PPS perform poorly in decision space on most test functions are that these algorithms ignore the diversity of population in decision space. It may trap the population into a larger feasible region and lose Pareto optimal solutions in other smaller feasible regions. C_DN-NSGA-II and C_MO_Ring_PSO_SCD possess higher diversity in decision space, thus losing little convergent selection pressure to drive the population approaching the CPF. Thus, the feasible Pareto optimal solutions with high convergence difficulty are still difficult to obtain. C_MMOEA/DC uses double clustering mechanisms in decision space and objective space respectively to obtain uniformly distributed Pareto optimal solutions.

The comparison between Table III and Table IV shows that the performance of C_MMOEA/DC deteriorates after the consideration of the feasibility of obtained solutions. This means that the algorithm retains some infeasible solutions finally. Due to the population lacking sufficient information interaction, it may trap into small feasible regions with no Pareto optimal solutions, affecting the convergence of the algorithm, as shown in Fig. 7(e). Among all the test functions, CMMF16 has the highest RPSP value, which indicates that it is the most difficult function among them and belongs to the Type IV problems. In Table V, CMMODE achieves eight smallest IGD values of all the test functions. In addition, C-TAEA achieves the best results in CMMF2, CMMF6, CMMF8 and CMMF9. In short, CMMODE has a narrow advantage over other competitive algorithms in objective space on CMMFs.

The two typical test functions CMMF11 and CMMF12 are selected to show the CPSs and CPF obtained by different algorithms. The CPSs and CPF obtained by different



(a) C-TAEA (b) MOEA/D-DAE (c) PPS (d) C_MO_Ring_PSO_SCD

(e) C_MMOEA/DC (f) C_DN-NSGA-II (g) CMMODE

Fig. 7. Comparision of CPSs and CPF obtained by six other advanced algorithms on CMMF11.

algorithms on CMMF11 are shown in Fig. 7. The CPSs obtained by different algorithms on CMMF12 are shown in the supplementary materials. From Fig. 7, it can be concluded that C-TAEA, MOEA/D-DAE and PPS obtain only one CPS. In addition, C_DN-NSGA-II finds multiple incomplete CPSs. C_MO_Ring_PSO_SCD and C_MMOEA/DC find multiple CPSs located in different feasible regions. However, the population is not well distributed in feasible regions and some solutions are trapped into small feasible regions without Pareto optimal solutions. In conclusion, CMMODE performs better than other algorithms in decision space, and it is more competitive compared with other CMOEAs in objective space on CMMFs.

D. Comparison on Location Selection Problem

In this section, CMMODE is compared with other six algorithms on a constrained multimodal multiobjective location-selection problem. This problem involves four schools, four hospitals and twelve convenience stores, as well as lake, river and bridges. The goal of this problem is to find reasonable locations, which are as close as possible to schools, hospitals and convenience stores and are located in the feasible regions. On the basis of the Euclidean distance from solution x to these locations , three minimization objectives and constraints are able to be expressed as follows:

$$\begin{cases} f_1(\mathbf{x}) = \min\ d(x, A_i), i = 1, \cdots, 5 \\ f_2(\mathbf{x}) = \min\ d(x, B_j), j = 1, \cdots, 4 \\ f_3(\mathbf{x}) = \min\ d(x, C_k), k = 1, \cdots, 10 \end{cases} \tag{17}$$

subject to:

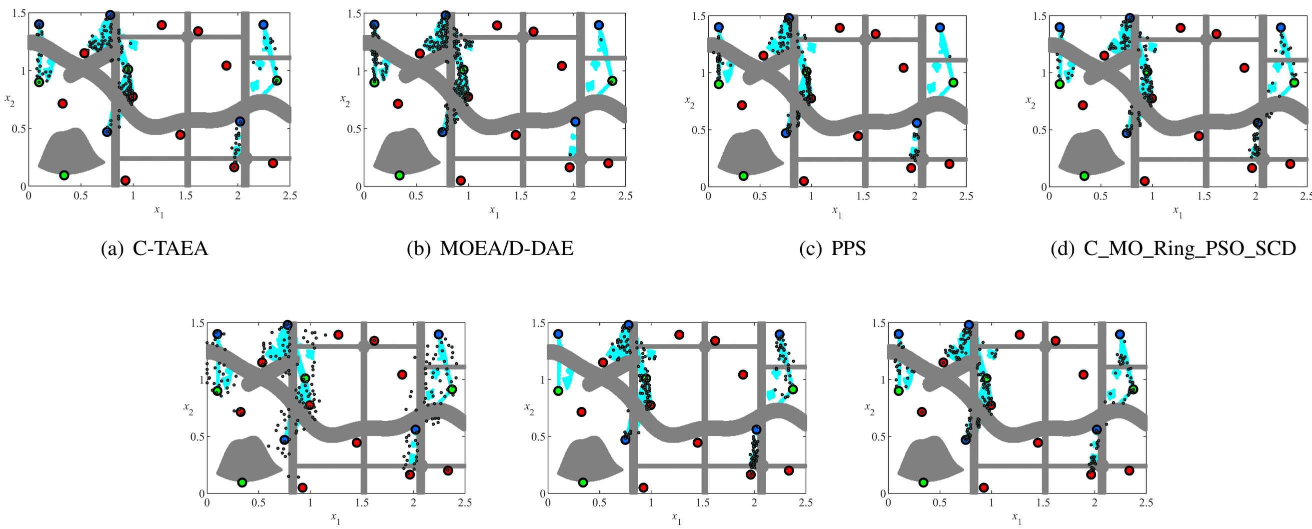
$$\mathbf{x} \notin Q_1 \cup Q_2 \cup Q_3 \tag{18}$$

where $A_i(i = 1, \cdots, 5)$, $B_j(j = 1, \cdots, 4)$ and $C_k(k = 1, \cdots, 10)$ denote the coordinates of schools, hospitals and convenience stores respectively. The infeasible regions consisting of lakes, rivers, and bridges are represented respectively as Q_1, Q_2 and Q_3. When the solution x satisfies all the constraints, the CV value is calculated to 0. Otherwise, the CV values are written as the reciprocal of the distances from x to the centers of all infeasible regions.

CMMODE conducts comparative experiments with other six algorithms on this problem. The experimental results in 31 runs are shown in Table VI and the final population obtained by different algorithms are shown in Fig. 8. It can be concluded from Fig. 8 and Table VI that C-TAEA finds more optimal regions and obtains smaller RPSP and IGD values than PPS and MOEA/D-DAE, because, C-TAEA employs a diversity archive to maintain more potential and infeasible solutions. Obviously, C_MMOEA/DC gets the widest population distribution because of its double-clustering mechanism. However, it does not converge well to multiple optimal regions. The solutions obtained by C_MO_Ring_PSO_SCD and CMMODE are distributed well in all the optimal regions, while C_DN-NSGA-II is unable to find all the optimal regions. In conclusion, CMMODE can effectively solve this problem.

VI. Conclusion

In this paper, a set of constrained multimodal multiobjective test functions is proposed, called CMMFs. In addition, we classify these test functions into four categories according to different properties. It is the first attempt to systematically construct multiobjective optimization test functions with constraints and multimodal properties. CMMFs pose challenges to the existing multiobjective evolutionary algorithms of obtaining multiple sets of feasible and well-distributed Pareto solution sets. Then, a multiobjective differential evolution algorithm with speciation mechanism and improved environment selection criterion is proposed to solve CMMFs, called CMMODE. In CMMODE, the combination of CDP and speciation mechanism is beneficial to find multiple discrete feasible regions and reverse more feasible equivalent Pareto optimal



(a) C-TAEA (b) MOEA/D-DAE (c) PPS (d) C_MO_Ring_PSO_SCD

(e) C_MMOEA/DC (f) C_DN-NSGA-II (g) CMMODE

Fig. 8. Obtained solutions by different algorithms on constrained multimodal multiobjective location-selection problem. The blue regions are the Pareto regions and the shaded regions indicate the infeasible regions. The blue, green, and red points represent schools, hospitals and convenience stores, respectively.

TABLE VI
AVERAGE RPSP AND IGD VALUES OVER 31 RUNS ON LOCATION-SELECTION PROBLEM, WHERE THE BEST RESULT IS SHOWN IN BOLD

	C-TAEA	MOEA/D-DAE	PPS	C_MO_Ring_PSO_SCD	C_MMOEA/DC	C_DN-NSGA-II	CMMODE
RPSP	3.19e-02 ± 4.90e-03(+)	5.84e-02 ± 4.67e-03(+)	4.50e-02 ± 1.14e-02(+)	**2.27e-02 ± 2.33e-04**(=)	2.78e-02 ± 3.12e-04(+)	6.56e-02 ± 4.23e-03(+)	**2.29e-02 ± 1.91e-03**
IGD	1.89e-02 ± 4.06e-03(+)	2.08e-02 ± 5.19e-03(=)	2.73e-02 ± 4.93e-03(+)	1.94e-02 ± 1.04e-03(+)	2.20e-02 ± 8.01e-04(+)	2.19e-02 ± 1.66e-03(+)	**1.88e-02 ± 1.31e-03**

solutions. Furthermore, the improved environment selection can enhance diversity and make the population uniformly distribute. The proposed algorithm also conducts comparative experiments with six competitive CMOEAs and MMOEAs on 17 CMMFs. The experimental comparison results show that CMMODE holds an advantage over six competitive algorithms in decision space and gains a narrow advantage over two CMOEAs in objective space. In addition, CMMODE performs well in the constrained multimodal multiobjective location-selection problem.

In the future work, more constrained multimodal multiobjective optimization problems with different properties will be proposed. Moreover, the mapping between decision and objective spaces of these problems will be clear enough to analyze and to compare the performances of different algorithms. Consequently, CMMODE will be improved and adopted in other practical applications.

References

[1] K. Deb, “Multi-objective optimisation using evolutionary algorithms: an introduction,” in *Multi-objective Evolutionary Optimisation for Product Design and Manufacturing*. Springer, 2011, pp. 3–34.

[2] C. He, R. Cheng, Y. Tian, X. Zhang, K. C. Tan, and Y. Jin, “Paired offspring generation for constrained large-scale multiobjective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 25, no. 3, pp. 448–462, 2020.

[3] N. Jozefowiez, F. Semet, and E.-G. Talbi, “Multi-objective vehicle routing problems,” *European journal of operational research*, vol. 189, no. 2, pp. 293–309, 2008.

[4] W. Lakhdhar, R. Mzid, M. Khalgui, Z. Li, G. Frey, and A. Al-Ahmari, “Multiobjective optimization approach for a portable development of reconfigurable real-time systems: From specification to implementation,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 3, pp. 623–637, 2018.

[5] Y. Xiang, X. Yang, Y. Zhou, and H. Huang, “Enhancing decomposition-based algorithms by estimation of distribution for constrained optimal software product selection,” *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 2, pp. 245–259, 2019.

[6] 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.

[7] Y. Tian, T. Zhang, J. Xiao, X. Zhang, and Y. Jin, “A coevolutionary framework for constrained multiobjective optimization problems,” *IEEE Transactions on Evolutionary Computation*, vol. 25, no. 1, pp. 102–116, 2020.

[8] Z. Ma and Y. Wang, “Shift-based penalty for evolutionary constrained multiobjective optimization and its application,” *IEEE Transactions on Cybernetics*, 2021, doi:10.1109/TCYB.2021.3069814. Early Access.

[9] K. Deb, A. Pratap, and T. Meyarivan, “Constrained test problems for multi-objective evolutionary optimization,” in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2001, pp. 284–298.

[10] K. Deb and H. Jain, “An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints,” *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2013.

[11] Z. Ma and Y. Wang, “Evolutionary constrained multiobjective optimization: Test suite construction and performance comparisons,” *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 6, pp. 972–986, 2019.

[12] J. J. Liang, C. T. Yue, and B. Y. Qu, “Multimodal multi-objective optimization: A preliminary study,” in *2016 IEEE Congress on Evolutionary Computation*, 2016, pp. 2454–2461.

[13] 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, 2016.

[14] M. Preuss, C. Kausch, C. Bouvy, and F. Henrich, “Decision space

diversity can be essential for solving multiobjective real-world problems," in *Multiple Criteria Decision Making for Sustainable Energy and Transportation Systems*. Springer, 2010, pp. 367–377.

[15] 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*, 2011, pp. 649–656.

[16] X. Zuo, C. Chen, W. Tan, and M. Zhou, "Vehicle scheduling of an urban bus line via an improved multiobjective genetic algorithm," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 1030–1041, 2014.

[17] H. Mala-Jetmarova, N. Sultanova, and D. Savic, "Lost in optimisation of water distribution systems? a literature review of system operation," *Environmental modelling & software*, vol. 93, pp. 209–254, 2017.

[18] J. Li, Y. Wang, S. Yang, and Z. Cai, "A comparative study of constraint-handling techniques in evolutionary constrained multiobjective optimization," in *2016 IEEE Congress on Evolutionary Computation*. IEEE, 2016, pp. 4175–4182.

[19] Z. Liu and Y. Wang, "Handling constrained multiobjective optimization problems with constraints in both the decision and objective spaces," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 5, pp. 870–884, 2019.

[20] Y. Zhou, Y. Xiang, and X. He, "Constrained multiobjective optimization: Test problem construction and performance evaluations," *IEEE Transactions on Evolutionary Computation*, vol. 25, no. 1, pp. 172–186, 2020.

[21] T. Takahama and S. Sakai, "Constrained optimization by the ε constrained differential evolution with gradient-based mutation and feasible elites," in *2006 IEEE International Conference on Evolutionary Computation*. IEEE, 2006, pp. 1–8.

[22] T. P. Runarsson and X. Yao, "Stochastic ranking for constrained evolutionary optimization," *IEEE Transactions on Evolutionary Computation*, vol. 4, no. 3, pp. 284–294, 2000.

[23] M. Asafuddoula, T. Ray, R. Sarker, and K. Alam, "An adaptive constraint handling approach embedded moea/d," in *2012 IEEE Congress on Evolutionary Computation*. IEEE, 2012, pp. 1–8.

[24] S. Zapotecas Martinez, A. Arias Montano, and C. A. Coello Coello, "Constrained multi-objective aerodynamic shape optimization via swarm intelligence," in *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*, 2014, pp. 81–88.

[25] Z. Yang, X. Cai, and Z. Fan, "Epsilon constrained method for constrained multiobjective optimization problems: some preliminary results," in *Proceedings of the companion publication of the 2014 Annual Conference on Genetic and Evolutionary Computation*, 2014, pp. 1181–1186.

[26] Z. Fan, H. Li, C. Wei, W. Li, H. Huang, X. Cai, and Z. Cai, "An improved epsilon constraint handling method embedded in MOEA/D for constrained multi-objective optimization problems," in *2016 IEEE Symposium Series on Computational Intelligence*. IEEE, 2016, pp. 1–8.

[27] M. A. Jan and R. A. Khanum, "A study of two penalty-parameterless constraint handling techniques in the framework of moea/d," *Applied Soft Computing*, vol. 13, no. 1, pp. 128–148, 2013.

[28] Z. Liu, Y. Wang, and B. Wang, "Indicator-based constrained multiobjective evolutionary algorithms," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2019.

[29] B. Tessema and G. Yen, "A self adaptive penalty function based algorithm for constrained optimization," in *2006 IEEE International Conference on Evolutionary Computation*, 2006, pp. 246–253.

[30] L. Jiao, J. Luo, R. Shang, and F. Liu, "A modified objective function method with feasible-guiding strategy to solve constrained multiobjective optimization problems," *Applied Soft Computing*, vol. 14, pp. 363–380, 2014.

[31] E. Mezura-Montes and C. A. C. Coello, "Constraint-handling in nature-inspired numerical optimization: past, present and future," *Swarm and Evolutionary Computation*, vol. 1, no. 4, pp. 173–194, 2011.

[32] T. Ray, H. K. Singh, A. Isaacs, and W. Smith, "Infeasibility driven evolutionary algorithm for constrained optimization," in *Constraint-handling in Evolutionary Optimization*. Springer, 2009, pp. 145–165.

[33] Q. Long, "A constraint handling technique for constrained multi-objective genetic algorithm," *Swarm and Evolutionary Computation*, vol. 15, pp. 66–79, 2014.

[34] K. Qiao, K. Yu, B. Qu, J. Liang, H. Song, and C. Yue, "An evolutionary multitasking optimization framework for constrained multiobjective optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 26, no. 2, pp. 263–277, 2022.

[35] Z. Fan, W. Li, X. Cai, H. Li, C. Wei, Q. Zhang, K. Deb, and E. Goodman, "Push and pull search for solving constrained multi-objective optimization problems," *Swarm and Evolutionary Computation*, vol. 44, pp. 665–679, 2019.

[36] F. Kudo, T. Yoshikawa, and T. Furuhashi, "A study on analysis of design variables in pareto solutions for conceptual design optimization problem of hybrid rocket engine," in *2011 IEEE Congress of Evolutionary Computation*. IEEE, 2011, pp. 2558–2562.

[37] J. Togelius, M. Preuss, and G. N. Yannakakis, "Towards multiobjective procedural map generation," in *Proceedings of the 2010 workshop on Procedural Content Generation in Games*, 2010, pp. 1–8.

[38] 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.

[39] C. Yue, B. Qu, K. Yu, J. Liang, and X. Li, "A novel scalable test problem suite for multimodal multiobjective optimization," *Swarm and Evolutionary Computation*, vol. 48, pp. 62–71, 2019.

[40] Y. Liu, H. Ishibuchi, G. G. Yen, Y. Nojima, and N. Masuyama, "Handling imbalance between convergence and diversity in the decision space in evolutionary multimodal multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 3, pp. 551–565, 2019.

[41] O. M. Shir, M. Preuss, B. Naujoks, and M. Emmerich, "Enhancing decision space diversity in evolutionary multiobjective algorithms," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2009, pp. 95–109.

[42] Y. Liu, H. Ishibuchi, Y. Nojima, N. Masuyama, and K. Shang, "A double-niched evolutionary algorithm and its behavior on polygon-based problems," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2018, pp. 262–273.

[43] C. Yue, B. Qu, and J. Liang, "A multiobjective particle swarm optimizer using ring topology for solving multimodal multiobjective problems," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 5, pp. 805–817, 2018.

[44] X. Li, "Niching without niching parameters: particle swarm optimization using a ring topology," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 1, pp. 150–169, 2009.

[45] C. Yue, P. N. Suganthan, J. Liang, B. Qu, K. Yu, Y. Zhu, and L. Yan, "Differential evolution using improved crowding distance for multimodal multiobjective optimization," *Swarm and Evolutionary Computation*, vol. 62, p. 100849, 2021.

[46] Y. Liu, G. G. Yen, and D. Gong, "A multimodal multiobjective evolutionary algorithm using two-archive and recombination strategies," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 4, pp. 660–674, 2018.

[47] K. Zhang, C. Shen, G. G. Yen, Z. Xu, and J. He, "Two-stage double niched evolution strategy for multimodal multi-objective optimization," *IEEE Transactions on Evolutionary Computation*, 2021.

[48] K. Deb and A. Saha, "Finding multiple solutions for multimodal optimization problems using a multi-objective evolutionary approach," in *Proceedings of the 12th annual conference on genetic and evolutionary computation*, 2010, pp. 447–454.

[49] A. Ahrari and K. Deb, "A novel class of test problems for performance evaluation of niching methods," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 6, pp. 909–919, 2017.

[50] Y. Nojima, T. Fukase, Y. Liu, N. Masuyama, and H. Ishibuchi, "Constrained multiobjective distance minimization problems," in *Proceedings of the Genetic and Evolutionary Computation Conference*, 2019, p. 586–594.

[51] S. Kimura and K. Matsumura, "Constrained multimodal function optimization using a simple evolutionary algorithm," in *2011 IEEE Congress of Evolutionary Computation (CEC)*. IEEE, 2011, pp. 447–454.

[52] Y. Zhou and S. Pei, "An effective adaptive multi-objective particle swarm for multimodal constrained function optimization." *J. Comput.*, vol. 5, no. 8, pp. 1144–1151, 2010.

[53] Y. Zhou, G. Zhou, and J. Zhang, "A hybrid glowworm swarm optimization algorithm to solve constrained multimodal functions optimization," *Optimization*, vol. 64, no. 4, pp. 1057–1080, 2015.

[54] J. Liang, T. P. Runarsson, E. Mezura-Montes, M. Clerc, P. N. Suganthan, C. C. Coello, and K. Deb, "Problem definitions and evaluation criteria for the CEC 2006 special session on constrained real-parameter optimization," *Journal of Applied Mechanics*, vol. 41, no. 8, pp. 8–31, 2006.

[55] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[56] C. A. C. Coello, G. B. Lamont, D. A. Van Veldhuizen *et al.*, *Evolutionary Algorithms for Solving Multi-objective problems*. Springer, 2007, vol. 5.

[57] 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.

[58] J. Li, M. E. Balazs, G. T. Parks, and P. J. Clarkson, "A Species Conserving Genetic Algorithm for Multimodal Function Optimization," *Evolutionary Computation*, vol. 10, no. 3, pp. 207–234, 2002.

[59] J. J. Liang and P. N. Suganthan, "Dynamic multi-swarm particle swarm optimizer with a novel constraint-handling mechanism," in *2006 IEEE International Conference on Evolutionary Computation*. IEEE, 2006, pp. 9–16.

[60] K. Li, R. Chen, G. Fu, and X. Yao, "Two-archive evolutionary algorithm for constrained multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 2, pp. 303–315, 2019.

[61] Q. Zhu, Q. Zhang, and Q. Lin, "A constrained multiobjective evolutionary algorithm with detect-and-escape strategy," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 5, pp. 938–947, 2020.

[62] Q. Lin, W. Lin, Z. Zhu, M. Gong, J. Li, and C. A. C. Coello, "Multimodal multiobjective evolutionary optimization with dual clustering in decision and objective spaces," *IEEE Transactions on Evolutionary Computation*, vol. 25, no. 1, pp. 130–144, 2020.

[63] K. A. De Jong, *An analysis of the behavior of a class of genetic adaptive systems.* University of Michigan, 1975.

[64] K. Li, R. Chen, G. Fu, and X. Yao, "Two-archive evolutionary algorithm for constrained multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 2, pp. 303–315, 2018.

[65] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-International Conference on Neural Networks*, vol. 4. IEEE, 1995, pp. 1942–1948.

[66] J. D. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," in *Proceedings of the First International Conference on Genetic Algorithms and Their Applications, 1985*. Lawrence Erlbaum Associates. Inc., Publishers, 1985.