Evolutionary Constrained Multiobjective Optimization: Test Suite Construction and Performance Comparisons

Zhongwei Ma and Yong Wang, *Senior Member, IEEE*

***Abstract*—For solving constrained multiobjective optimization problems (CMOPs), many algorithms have been proposed in the evolutionary computation research community for the past two decades. Generally, the effectiveness of an algorithm for CMOPs is evaluated by artificial test problems. However, after a brief review of current artificial test problems, we have found that they are not well-designed and fail to reflect the characteristics of real-world applications (e.g., small feasibility ratio). Thus, in this paper, we first propose a new constraint construction method to facilitate the systematic design of test problems. Then, on the basis of this method, we design a new test suite consisting of 14 instances, which covers diverse characteristics extracted from real-world CMOPs and can be divided into four types. Considering that the comprehensive performance comparisons among the constraint-handling techniques (CHTs) remain scarce, we choose several representative CHTs and compare their performance on our test suite. The performance comparisons identify the strengths and weaknesses of different CHTs on different types of CMOPs and provide guidelines on how to select/design a CHT in a specific scenario.**

***Index Terms*—Constrained multiobjective optimization, constraint-handling techniques (CHTs), evolutionary algorithms (EAs), performance comparisons, test suite.**

I. Introduction

A. Constrained Multiobjective Optimization Problems

Constrained multiobjective optimization problems (CMOPs) have received increasing attention during the past two decades since they widely exist in many real-world applications, such as scheduling [1], vehicle body design [2], and systematic deployment optimization [3]. CMOPs are featured with one or more constraints and multiple conflicting objectives that need to be optimized simultaneously. CMOPs pose a great challenge to evolutionary algorithms (EAs) due to the fact that they require EAs to offer tradeoffs among the conflicting objectives subject to all constraints [4], [5].

Without loss of generality, a CMOP can be formulated as

$$\begin{aligned} &\min\ \vec{F}(\vec{x}) = (f_1(\vec{x}), f_2(\vec{x}), \ldots, f_m(\vec{x}))^T \\ &\text{s.t.} \begin{cases} g_j(\vec{x}) \le 0, & j = 1, \ldots, p \\ h_j(\vec{x}) = 0, & j = p+1, \ldots, p+q \\ x_k^{min} \le x_k \le x_k^{max}, & k = 1, \ldots, n \end{cases} \end{aligned} \tag{1}$$

where $\vec{x} = (x_1, \ldots, x_n) \in S$ is an n-dimensional decision vector, $x_k (k \in \{1, \ldots, n\})$ is the kth decision variable, $S \subset \mathbb{R}^n$ is the decision space, $\vec{F}$ is the objective vector consisting of m objectives, $f_i(\vec{x}) (i \in \{1, \ldots, m\})$ is the ith objective, $g_j(\vec{x}) \le 0$ is the jth inequality constraint, p is the number of inequality constraints, $h_j(\vec{x}) = 0$ is the $(j-p)$th equality constraint, q is the number of equality constraints, and x_k^{min} and x_k^{max} are the bound constraints of x_k.

The constraint violation of $\vec{x}$ on the jth constraint can be defined as follows:

$$CV_j(\vec{x}) = \begin{cases} \max\bigl(0, g_j(\vec{x})\bigr), & j = 1, \ldots, p \\ \max\bigl(0, \bigl|h_j(\vec{x})\bigr| - \delta\bigr), & j = p+1, \ldots, p+q \end{cases} \tag{2}$$

where δ is a very small tolerance value to relax equality constraints (e.g., 10^{-4}). Based on this definition, $\vec{x}$ is a feasible solution if its total constraint violation, i.e.,

$$CV(\vec{x}) = \sum_{j=1}^{p+q} CV_j(\vec{x}) \tag{3}$$

is equal to 0; otherwise, $\vec{x}$ is an infeasible solution. The feasible region is the set of all the feasible solutions, namely $\Omega = \{\vec{x} | CV(\vec{x}) = 0, \vec{x} \in S\}$.

B. Classification of CMOPs

In unconstrained multiobjective optimization [i.e., ignoring the constraints in (1)], given two solutions $\vec{a}$ and $\vec{b}$, $\vec{a}$ Pareto dominates $\vec{b}$ (denoted as $\vec{a} \prec \vec{b}$), if and only if $f_i(\vec{a}) \le f_i(\vec{b})$ for every $i \in \{1, \ldots, m\}$ and $f_j(\vec{a}) < f_j(\vec{b})$ for at least one index $j \in \{1, \ldots, m\}$. $\vec{x}^*$ is a Pareto optimal solution if there does not exist any solution that Pareto dominates it. The set of all the Pareto optimal solutions is the Pareto set. The Pareto front (PF) is the image of the Pareto set in the objective space.

Due to the presence of constraints, it is more difficult to obtain the PF of a CMOP, called constrained PF, than its

Manuscript received July 30, 2018; revised November 15, 2018 and January 16, 2019; accepted January 29, 2019. Date of publication February 1, 2019; date of current version November 27, 2019. This work was supported in part by the Innovation-Driven Plan in Central South University under Grant 2018CX010, in part by the National Natural Science Foundation of China under Grant 61673397, in part by the Hunan Provincial Natural Science Fund for Distinguished Young Scholars under Grant 2016JJ1018, and in part by the Beijing Advanced Innovation Center for Intelligent Robots and Systems under Grant 2018IRS06. *(Corresponding author: Yong Wang.)*

The authors are with the School of Information Science and Engineering, Central South University, Changsha 410083, China (e-mail: mzw_cemo@csu.edu.cn; ywang@csu.edu.cn).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the author. The total size of the file is 1.39 MB.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TEVC.2019.2896967

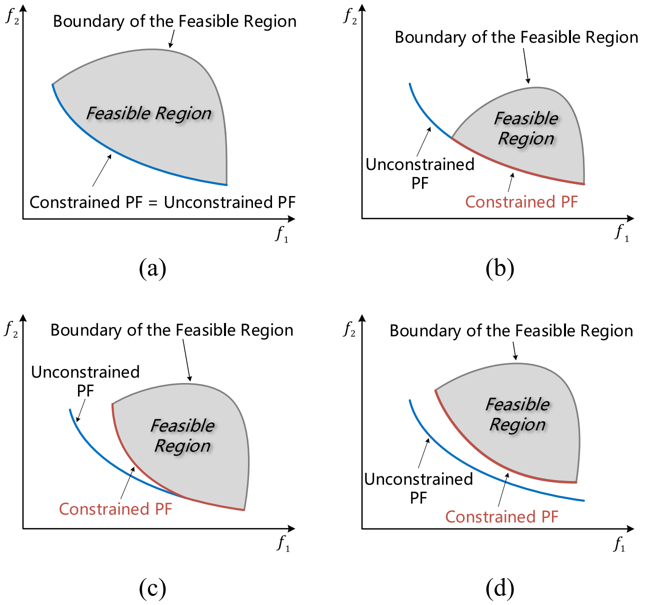


Fig. 1. Classification of CMOPs. (a) Type I. (b) Type II. (c) Type III. (d) Type IV. For the sake of clarity, in our following discussions, the feasible region and the boundary of the feasible region specifically refer to their counterparts in the objective space.

unconstrained counterpart [6]. Compared with unconstrained multiobjective optimization, some original Pareto optimal solutions may become infeasible in a CMOP and/or some solutions on the boundary of the feasible region of a CMOP may become the Pareto optimal solutions. In order to identify the relationship between unconstrained and constrained PFs, we classify CMOPs into the following four types.

1) *Type I:* As shown in Fig. 1(a), the whole unconstrained PF remains feasible, that is, the constrained PF is the same with the unconstrained PF. Some CMOPs with loose constraints often belong to this type, such as the optimization of biped robot gaits [7], [8].
2) *Type II:* As plotted in Fig. 1(b), the constrained PF is a part of the unconstrained PF, since constraints make a portion of the unconstrained PF infeasible. This type exists widely in real-world CMOPs and a similar engineering problem can be found in the topology optimization of multicell tubes [9].
3) *Type III:* The constrained PF consists of a part of the unconstrained PF and a part of the boundary of the feasible region. As shown in Fig. 1(c), constraints make a part of the unconstrained PF infeasible and some solutions on the boundary of the feasible region become the Pareto optimal solutions. A real-world CMOP belonging to this type is the reliability design problem in the field of automotive body optimization [10], [11].
4) *Type IV:* As plotted in Fig. 1(d), the unconstrained PF is entirely located outside the feasible region. Thus, the constrained PF is composed of a part of the boundary of the feasible region. A typical CMOP of this type is the robot gripper optimization problem [12].

When solving CMOPs, it is well known that we need to seek a balance between objectives and constraints in the search process [13]. However, there is little awareness on how to reasonably apply this principle when facing the different types of CMOPs mentioned above. In particular, from types I to IV, the search bias should be gradually switched from the objectives to constraints. For type I, to obtain the Pareto optimal solutions on the constrained PF, more focus should be placed on objectives. Instead, for type IV, more emphasis should be put on constraints since the Pareto optimal solutions are completely located on the boundary of the feasible region.

C. Present Artificial Test Problems/Suites and Their Drawbacks

In general, it is impractical to assess the performance of an algorithm by a specific real-world CMOP [14]. The reason is because this assessment process may require some domain knowledge and cannot effectively demonstrate an algorithm's generality. An alternative way is to use the artificial test problems that cover as many characteristics extracted from real-world CMOPs as possible. If an algorithm achieves desirable results on artificial test problems, we could believe that it has good potential to solve real-world CMOPs. Some attempts have been made on the design of artificial test problems for CMOPs. Examples include SRN [15], TNK [16], OSY [17], CTPs [18], CFs [19], NCTPs [20], and C-DTLZs [21].

SRN, TNK, and OSY are perhaps the earliest three test problems and have been frequently considered for performance comparisons of algorithms. SRN has a continuous constrained PF that is a part of the unconstrained PF (i.e., type II). TNK has a disconnected constrained PF that completely lies on the boundary of the feasible region (i.e., type IV). OSY is a type-IV CMOP and has six constraints. OSY's constrained PF has five parts and each part is the intersection of certain constraints.

Deb *et al.* [18] pointed out that the above three test problems have the following shortcomings: 1) they have few decision variables; 2) their objectives and constraints are not sufficiently nonlinear; and 3) their difficulties are not tunable in terms of constraints. Therefore, a general framework is suggested in [18] to design CMOPs with constraints that have tunable difficulties. Based on the proposed framework, seven instances are proposed (called CTPs). Most of them have disconnected or discrete constrained PFs. Note that, CTPs have large feasibility ratios.

Following the framework of CTPs, Zhang *et al.* [19] constructed ten test problems (called CFs). CFs have disconnected geometries of constrained PFs and belong to the type-II and type-III CMOPs. CFs also introduce complicated variable linkages. As a result, when solving CFs, an algorithm always struggles on convergence. Besides, most of CFs have large feasible regions.

Inspired by CTPs, Li *et al.* [20] proposed 18 new test problems (called NCTPs). Compared with the original CTPs, these test problems make the following improvements: 1) the Ronsenbrock function is employed as the distance function to increase the difficulty of convergence; 2) high-dimensional decision spaces are considered; and 3) an extra constraint is added to explicitly decrease the feasibility ratio.

Jain and Deb [21] designed a test suite of five CMOPs with a scalable number of objectives, which are the extended

TABLE I
Comparison of Current Test Problems and the Proposed Test Problems. "√" and "×" Denote the Presence and Absence of the Corresponding Characteristic, Respectively, and "⋆" Denotes That the Corresponding Characteristic Is Satisfied Partially

	SRN [15]	TNK [16]	OSY [17]	CTPs [18]	CFs [19]	NCTPs [20]	C-DTLZs [21]	MWs
Type	II	IV	IV	II, III, IV	II, III	I, III, IV	I, II, IV	ALL
Small Feasibility Ratio	√	√	√	×	×	⋆	⋆	√
Sufficient Nonlinearity of Constraints	×	√	×	√	√	√	×	√
More Than Two Constraints	×	×	√	×	×	×	√	√
Scalability of the Number of Objectives	×	×	×	×	×	×	√	√
High-dimensional Decision Vector	×	×	×	×	√	√	√	√
Proper Difficulty of Convergence	×	×	×	×	×	√	√	√
Diverse Geometries of Constrained PFs	×	×	×	×	×	×	×	√

versions of the DTLZ problems used in the field of unconstrained multiobjective optimization. Specifically, three kinds of constraints are introduced.

1) The first kind of constraints provides an infeasible barrier, and the unconstrained PF remains feasible (i.e., C1-DTLZs).
2) The second kind of constraints defines several isolated feasible regions along the unconstrained PF, and the constrained PF is a part of the unconstrained PF (i.e., C2-DTLZs).
3) The third kind of constraints makes the unconstrained PF infeasible and the constrained PF is made up of the boundary of the feasible region (i.e., C3-DTLZs).

According to our classification of CMOPs, C1-DTLZs, C2-DTLZs, and C3-DTLZs belong to types I, II, and IV, respectively. It is worth noting that the constraints of C-DTLZs are not sufficiently nonlinear, because the shape of the feasible region is regular and simple (e.g., spherical feasible region in C2-DTLZ2), and the boundary of the feasible region has very small curvature (e.g., linear boundary in C1-DTLZ1 and C3-DTLZ1). Besides, the feasibility ratios of some C-DTLZs (e.g., C1-DTLZ3, C3-DTLZ1, and C3-DTLZ4) are greater than 30% based on our calculation.

Based on the above discussions, present artificial test problems fail to comprehensively represent the characteristics of real-world CMOPs.

1) It is common that real-world CMOPs have very small feasible regions and complex nonlinear constraints [22], [23]. However, the feasibility ratios of current test problems are always large and/or their constraints are not sufficiently nonlinear. Note that if the feasibility ratio of a CMOP is very large, it is hard to identify the ability of a constraint-handling technique (CHT).
2) Most of current test problems have a single constraint, which is much less than the number of constraints in real-world CMOPs. For example, there are 9 and 11 constraints in the ship parametric design problem [24] and the speed-reducer design problem [25], respectively.
3) Real-world CMOPs may have an arbitrary number of objectives [26]. But current test problems (except C-DTLZs) are CMOPs with two objectives, which cannot be scalable in terms of the number of objectives.
4) Some test problems consider few decision variables. For example, the most commonly used test problems, i.e., SRN, TNK, OSY, and CTPs, have no more than six decision variables.

In addition, some other limitations of current test problems make them unsuitable for general use. First, some test problems are unable to provide proper difficulties of convergence. Second, some special geometries of constrained PFs are not suggested in current test problems, such as the jagged geometry in the aircraft landing scheduling problem [27]. Third, there is no test suite containing all the four types of CMOPs according to our taxonomy.

D. Motivation

Table I summarizes the drawbacks of current test problems. It is clear from Table I that it is necessary to further investigate the construction of artificial test problems in constrained multiobjective optimization. Motivated by the above consideration, we propose a new constraint construction method. This method enables us to design CMOPs with controllable sizes of the feasible regions and complex geometries of constrained PFs. Afterward, we suggest three kinds of distance functions that can provide appropriate difficulties of convergence for CMOPs. Based on the constraint construction method and the distance functions, we propose a test suite containing 14 instances. They cover all of the four types of CMOPs.

Besides, we choose several popular CHTs for performance comparisons on our test suite, since the comparative studies among CHTs have been scarcely reported. We compare six and four representative CHTs under the frameworks of NSGA-II [28] and MOEA/D [29], respectively, which are two well-known paradigms of multiobjective EAs [30]. The performance comparisons evaluate the performance of different CHTs on different types of CMOPs and help us to select/design a suitable CHT for a specific scenario.

The rest of this paper is organized as follows. Section II presents the constraint construction method. Section III gives the details of the distance functions and test suite. The performance comparisons among CHTs are conducted in Section IV. Finally, Section V concludes this paper.

II. Proposed Constraint Construction Method

The constraint construction method suggested in this paper provides a way to design CMOPs with desirable constraints. It consists of two main processes: 1) a global control process and 2) a local adjustment process. The global control process aims to control the size of the feasible region in the objective space, while the local adjustment process helps to adjust

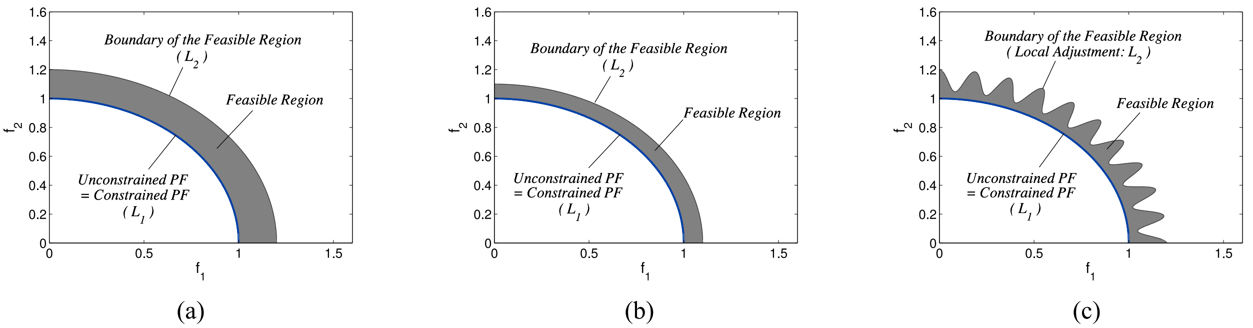


Fig. 2. CMOPs with constraints defined by (a) $L_2 : x^2 + y^2 - 1.2^2 = 0$, (b) $L_2 : x^2 + y^2 - 1.1^2 = 0$, and (c) local-adjusted L_2 in (a) with $A = -0.15$, $B = 20$, $C = 1$, and $D = 1$.

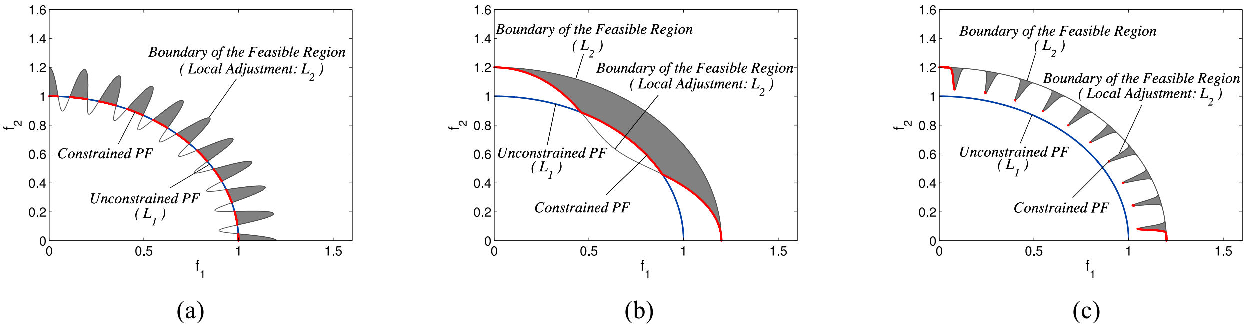


Fig. 3. CMOPs with different types generated by similar parameter settings of local adjustment. (a) A discontinuous type-II CMOP and the parameter settings are the same as in (10) except $A = -0.3$. (b) A continuous type-III CMOP and the parameter settings are the same with (a) except $B = 2$. (c) A discrete type-IV CMOP and the parameter settings are the same as in (10) except $D = 20$. The gray area(s) is/are the feasible region(s).

the complexity of the boundary of the feasible region and to generate different geometries of constrained PFs.

A. Global Control Process

This process first chooses a set of *similar functions*. For example, we use the following pair of functions:

$$\begin{cases} L_1 : x^2 + y^2 - 1^2 = 0 \\ L_2 : x^2 + y^2 - 1.2^2 = 0 \end{cases} \tag{4}$$

where L_1 and L_2 are the *similar functions* since they have similar mathematical forms and geometrical structures. Then, we transform L_1 and L_2 into two functions related to f_1 and f_2

$$L_1 \Rightarrow f_1^2 + f_2^2 - 1^2 = 0 \Rightarrow f_2 = \sqrt{1 - f_1^2} \tag{5}$$

$$L_2 \Rightarrow f_1^2 + f_2^2 - 1.2^2 = 0 \tag{6}$$

where (5) can be used to produce an unconstrained PF because it provides a conflicting relationship between f_1 and f_2. From (6), it is easy to obtain an inequality constraint

$$f_1^2 + f_2^2 - 1.2^2 \leq 0 \tag{7}$$

so, we can construct a CMOP as follows:

$$\begin{aligned} \min \quad & \begin{cases} f_1(\vec{x}) = x_1 \\ f_2(\vec{x}) = g\sqrt{1 - f_1^2} \end{cases} \\ \text{s.t.} \quad & c(\vec{x}) = f_1^2 + f_2^2 - 1.2^2 \leq 0 \end{aligned} \tag{8}$$

where the distance function $g \geq 1$. The unconstrained PF and the boundary of the feasible region in (8) are defined by (5) and (6), respectively. As shown in Fig. 2(a), the feasible region in the objective space is the gray area bounded by (5) and (6), which are derived from L_1 and L_2, respectively. If we replace L_2 in (4) with "$x^2 + y^2 - 1.1^2 = 0$", as shown in Fig. 2(b), the CMOP generated through the above process has a smaller size of the feasible region in the objective space than the feasible region produced by "$x^2 + y^2 - 1.2^2 = 0$". In other words, the similarity between *similar functions* determines the feasibility ratios of the constructed CMOPs. By choosing appropriate *similar functions*, we can explicitly control the size of the feasible region in the objective space, which enables us to design test problems with very small feasibility ratios as in real-world CMOPs.

B. Local Adjustment Process

When solving CMOPs by EAs, constraints can cause difficulties on both convergence and diversity. Moreover, constraints may increase the hardness of obtaining feasible solutions and change the feasibilities of the original Pareto optimal solutions. Therefore, constraints play a critical role in determining the difficulties of CMOPs.

Inspired by this fact, a local adjustment process is proposed to enhance the nonlinearity of constraints. In this paper, the local adjustment process is designed by adding a periodic function into the constraints constructed in the global control process. One possible formulation of the periodic function is

$$A \sin\left(B \cdot l(\vec{F})^C\right)^D \tag{9}$$

TABLE II
CHARACTERISTICS OF OUR TEST PROBLEMS, WHERE "SCALABILITY" IS THE SCALABILITY IN TERMS OF THE NUMBER OF OBJECTIVES ("NOF") OR THE NUMBER OF DECISION VARIABLES ("NOX"), "NOC" IS THE NUMBER OF CONSTRAINTS, "UPF" IS THE UNCONSTRAINED PF, AND "CPF" IS THE CONSTRAINED PF. IN ADDITION, A "MIXED" GEOMETRY DENOTES THAT IT INCLUDES BOTH CONVEX AND CONCAVE SEGMENTS

Problem	Type	Scalability		Constraint(s)		Geometry		Feasible Region		Decision Space	Distance Function
		NoF	NoX	NoC	Nonlinearity	UPF	CPF	Size	Connectivity		
MW1	II	No	Yes	1	Yes	Linear	Disconnected	$< 0.1‰$	Disconnected	$[0, 1]^n$	g_1
MW2	I	No	Yes	1	Yes	Linear	Linear	$< 0.1‰$	Disconnected	$[0, 1]^n$	g_2
MW3	III	No	Yes	2	Yes	Linear	Mixed	$< 0.1‰$	Connected	$[0, 1]^n$	g_3
MW4	I	Yes	Yes	1	Yes	Linear	Linear	$< 0.1‰$	Connected	$[0, 1]^n$	g_1
MW5	II	No	Yes	3	Yes	Concave	Discrete	$\approx 0.3‰$	Connected	$[0, 1]^n$	g_1
MW6	II	No	Yes	1	Yes	Concave	Disconnected	$< 0.1‰$	Disconnected	$[0, 1.1]^n$	g_2
MW7	III	No	Yes	2	Yes	Concave	Disconnected	$< 0.1‰$	Connected	$[0, 1]^n$	g_3
MW8	II	Yes	Yes	1	Yes	Concave	Disconnected	$< 0.1‰$	Disconnected	$[0, 1]^n$	g_2
MW9	IV	No	Yes	1	Yes	Convex	Concave	$< 0.1‰$	Connected	$[0, 1]^n$	g_1
MW10	III	No	Yes	3	Yes	Concave	Disconnected	$< 0.1‰$	Disconnected	$[0, 1]^n$	g_2
MW11	IV	No	Yes	4	Yes	Concave	Disconnected	$< 0.1‰$	Disconnected	$[0, \sqrt{2}]^n$	g_3
MW12	IV	No	Yes	2	Yes	Mixed	Mixed	$< 0.1‰$	Disconnected	$[0, 1]^n$	g_1
MW13	III	No	Yes	2	Yes	Disconnected	Disconnected	$\approx 6.5\%$	Disconnected	$[0, 1.5]^n$	g_2
MW14	I	Yes	Yes	1	Yes	Disconnected	Disconnected	$\approx 0.1‰$	Connected	$[0, 1.5]^n$	g_3

where $l(\vec{F})$ is a function related to the objective vector $\vec{F}$. (9) can produce a series of local shapes. Moreover, it has four parameters A, B, C, and D to control the local shapes. Specifically, A controls the magnitude of the local shapes, B determines the number of the local shapes, C affects the distribution of the local shapes, and D reflects the concave/convex degree of the local shapes. If we introduce (9) into the constraint in (8) with $A = -0.15$, $B = 20$, $C = 1$, and $D = 1$, the following CMOP is generated:

$$\begin{aligned} \min \ & \begin{cases} f_1(\vec{x}) = x_1 \\ f_2(\vec{x}) = g\sqrt{1 - f_1^2} \end{cases} \\ \text{s.t. } \ & c(\vec{x}) = f_1^2 + f_2^2 - (1.2 - 0.15\sin(20l))^2 \le 0 \\ & l = \arctan(f_2/f_1). \end{aligned} \tag{10}$$

As shown in Fig. 2(c), the constraint has a higher nonlinearity than that in (8). By carefully tuning the four parameters in (9), the local adjustment process enables us to design CMOPs with different types and diverse geometries of constrained PFs, such as the disconnected type-II CMOP in Fig. 3(a), the continuous type-III CMOP in Fig. 3(b), and the discrete type-IV CMOP in Fig. 3(c) [note that the last two CMOPs require two constraints, i.e., the constraints in (8) and (10) with minor modifications].

We would like to give the following remarks to the local adjustment process:

1) If the constraints constructed in the global control process are able to provide sufficient difficulties, the local adjustment process is not necessary. Under this condition, A can be set to zero.
2) We can also make use of the local adjustment to adjust the unconstrained PF, with the aim of improving its nonlinearity. It is because a complex unconstrained PF can increase the difficulties in finding desirable Pareto optimal solutions, especially for types I–III.

In summary, the proposed constraint construction method is able to produce CMOPs with various characteristics, such as small and controllable feasibility ratio, sufficient nonlinearity in terms of constraints, and diverse geometries of constrained PFs. In addition, if the global control process involves several *similar functions*, we can construct test problems that has multiple constraints, like real-world CMOPs.

III. PROPOSED TEST SUITE

Inspired by Deb [31] and Huband *et al.* [32], the construction form of CMOPs is shown as

$$\begin{aligned} \min \ & \begin{cases} f_1(\vec{x}) = g(\vec{x}_{\mathrm{II}}) \cdot s_1(\vec{x}_{\mathrm{I}}) \\ \qquad\vdots \\ f_m(\vec{x}) = g(\vec{x}_{\mathrm{II}}) \cdot s_m(\vec{x}_{\mathrm{I}}) \end{cases} \\ \text{s.t. } \ & c_j(\vec{x}) \ge 0, \ j = 1, \ldots, p \end{aligned} \tag{11}$$

where s_i ($i \in \{1, \ldots, m\}$) is the shape function of the ith objective, m is the number of objectives, c_j ($j \in \{1, \ldots, p\}$) is the jth constraint, p is the number of constraints, g is the distance function whose minimum value is 1, and $\vec{x}_{\mathrm{I}} = (x_1, \ldots, x_{m-1})$ and $\vec{x}_{\mathrm{II}} = (x_m, \ldots, x_n)$ are the subvectors of $\vec{x}$. In principle, the shape functions define the unconstrained PF, the distance function determines the difficulty of convergence, and the constraints control the difficulty of constraint handling.

In Section II, we have described the design process of constraints and constrained PFs. In this section, to accomplish the construction of test problems, we will present three kinds of distance functions that can provide appropriate difficulties of convergence. After that, we will introduce our newly designed test problems.

A. Distance Functions

Three kinds of distance functions with different characteristics are considered, namely biased distance function, multimodal distance function, and distance function with variable linkages. Note that these functions are scalable in terms of the number of decision variables.

A biased distance function is defined as follows:

$$\begin{aligned} g_1 &= 1 + \sum_{i=m}^{n}\left(1 - \exp\left(-10\left(z_i - 0.5 - \frac{i-1}{2n}\right)^2\right)\right) \\ z_i &= x_i^{n-m}, \ x_i \in \vec{x}_{\mathrm{II}} \end{aligned} \tag{12}$$

g_1 reaches its minimum value if $z_i = 0.5 + (i-1)/2n$ ($i \in \{m, \ldots, n\}$). Note that x_i^{n-m} tends to prevent z_i from achieving its optimal value (i.e., $0.5 + (i-1)/2n$). The reason is that x_i^{n-m} is biased toward zero for $x_i \in [0, 1]$ but the optimal value of z_i is greater than 0.5.

A multimodal distance function is formulated as follows:

$$g_2 = 1 + \sum_{i=m}^{n}\left(1.5 + \frac{0.1}{n}z_i^2 - 1.5\cos(2\pi z_i)\right)$$
$$z_i = 1 - \exp\left(-10\left(x_i - \frac{i-1}{n}\right)^2\right),\ x_i \in \vec{x}_{\mathrm{II}}. \tag{13}$$

g_2 is mapped to the global minimum if $z_i = 0$ $(i \in \{m, \dots, n\})$. $z_i = 1-\exp(-10(x_i-(i-1)/n)^2)$ ensures that when g_2 reaches the global minimum, the value of each x_i $(i \in \{m, \dots, n\})$ is different.

Besides, the distance function with variable linkages is given as follows:

$$g_3 = 1 + \sum_{i=m}^{n} 2\left(x_i + (x_{i-1} - 0.5)^2 - 1\right)^2. \tag{14}$$

g_3 gets its minimum value when $x_i = 1 - (x_{i-1} - 0.5)^2$ $(i \in \{m, \dots, n\})$.

B. Test Problems

Based on the constraint construction method and distance functions, we design a set of 14 test problems (called MW1–MW14) in this paper. They cover all the four types of CMOPs according to our taxonomy and have considerably small feasibility ratios, sufficiently nonlinear constraints, and other important characteristics. Table II summarizes the main characteristics of MWs.

MW1:

$$\min \begin{cases} f_1(\vec{x}) = x_1 \\ f_2(\vec{x}) = g_1(1 - 0.85f_1/g_1) \end{cases}$$
$$\text{s.t. } c(\vec{x}) = 1 - f_1 - f_2 + 0.5\sin(2\pi l)^8 \geq 0$$
$$l = \sqrt{2}f_2 - \sqrt{2}f_1. \tag{15}$$

This test problem is constructed by means of two *similar functions*: "$1 - 0.85x - y = 0$" and "$1 - x - y = 0$". The former is used to produce the unconstrained PF. In addition, by conducting the local adjustment process on the latter, the constraint is generated. MW1 belongs to type II and has a disconnected geometry of constrained PF. It uses the biased distance function [i.e., g_1 in (12)]. As shown in Fig. 4(a), the gray areas are the feasible regions and the red points are the images of the Pareto optimal solutions in the objective space.

MW2:

$$\min \begin{cases} f_1(\vec{x}) = x_1 \\ f_2(\vec{x}) = g_2(1 - f_1/g_2) \end{cases}$$
$$\text{s.t. } c(\vec{x}) = 1 - f_1 - f_2 + 0.5\sin(3\pi l)^8 \geq 0$$
$$l = \sqrt{2}f_2 - \sqrt{2}f_1. \tag{16}$$

In this test problem, the unconstrained PF and the constraint utilize the same *similar function*: "$1 - x - y = 0$". Moreover, by conducting the local adjustment process on this *similar function*, the constraint is designed. MW2's constrained PF is continuous and the same with the unconstrained PF. It adopts the multimodal distance function [i.e., g_2 in (13)]. As shown in Fig. 4(b), the gray areas are the feasible regions and the red points are the images of the Pareto optimal solutions in the objective space.

MW3:

$$\min \begin{cases} f_1(\vec{x}) = x_1 \\ f_2(\vec{x}) = g_3(1 - f_1/g_3) \end{cases}$$
$$\text{s.t. } c_1(\vec{x}) = 1.05 - f_1 - f_2 + 0.45\sin(0.75\pi l)^6 \geq 0$$
$$c_2(\vec{x}) = 0.85 - f_1 - f_2 + 0.3\sin(0.75\pi l)^2 \leq 0$$
$$l = \sqrt{2}f_2 - \sqrt{2}f_1. \tag{17}$$

This test problem takes three *similar functions* into account. The unconstrained PF is produced by "$1-x-y = 0$". In addition, through the local adjustment process on "$0.85-x-y = 0$" and "$1.05-x-y = 0$", two constraints c_1 and c_2 are designed, respectively. MW3 is a type-III CMOP, that is, its constrained PF contains a part of the unconstrained PF and a part of the boundary of the feasible region. The distance function with variable linkages is employed [i.e., g_3 in (14)]. As shown in Fig. 4(c), the grey areas are the feasible regions and the red points are the images of the Pareto optimal solutions in the objective space. Due to the fact that MW3 has some narrow parts in its feasible region, it is a hard task to find the Pareto optimal solutions on these parts.

MW4:

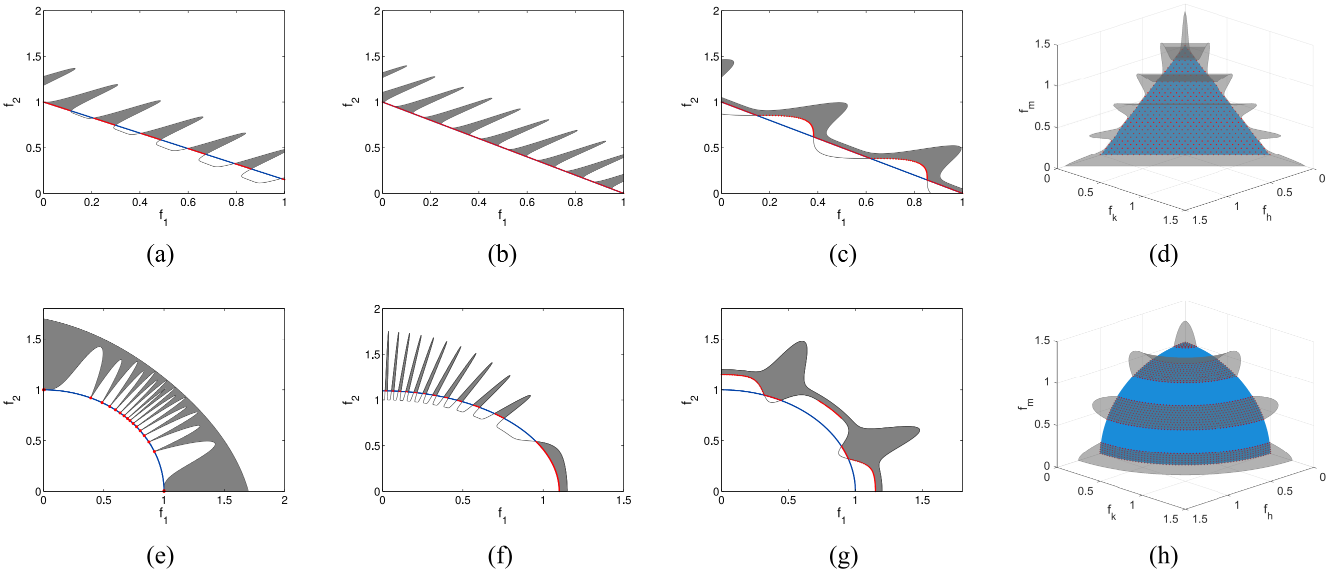
$$\min \begin{cases} f_1(\vec{x}) = g_1 \prod_{i=1}^{m-1}(1 - x_i) \\ f_{k=2:m-1}(\vec{x}) = g_1 x_{m-k+1} \prod_{i=1}^{m-k}(1 - x_i) \\ f_m(\vec{x}) = g_1 x_1 \end{cases}$$
$$\text{s.t. } c(\vec{x}) = \left(1 + 0.4\sin(2.5\pi l)^8\right) - f_1 - \cdots - f_m \geq 0$$
$$l = f_m - f_1 - \cdots - f_{m-1}. \tag{18}$$

Since our constraint construction method can be extended to a high-dimensional objective space, it enables us to design CMOPs with scalable number of objectives. In MW4, the unconstrained PF is defined by the hyperplane "$1 - x - y - z - \cdots = 0$". In addition, by applying the local adjustment process to this function, the constraint is obtained (like MW2, the unconstrained PF and the constraint utilize the same *similar function*). MW4 is a type-I CMOP and uses the biased distance function [i.e., g_1 in (12)]. Fig. 4(d) shows the relationship between the mth objective and any two other objectives f_k and f_h from $\{f_1, f_2, \dots, f_{m-1}\}$. The feasible region is the area bounded by the gray surface and the blue plane, and the red points are the samples of the images of the Pareto optimal solutions in the objective space.

MW5:

$$\min \begin{cases} f_1(\vec{x}) = g_1 x_1 \\ f_2(\vec{x}) = g_1\sqrt{1 - (f_1/g_1)^2} \end{cases}$$
$$\text{s.t. } c_1(\vec{x}) = (1.7 - 0.2\sin(2l_1))^2 - f_1^2 - f_2^2 \geq 0$$
$$c_2(\vec{x}) = \left(1 + 0.5\sin\left(6l_2^3\right)\right)^2 - f_1^2 - f_2^2 \leq 0$$
$$c_3(\vec{x}) = \left(1 - 0.45\sin\left(6l_2^3\right)\right)^2 - f_1^2 - f_2^2 \leq 0$$
$$l_1 = \arctan(f_2/f_1)$$
$$l_2 = 0.5\pi - 2|\arctan(f_2/f_1) - 0.25\pi|. \tag{19}$$

This test problem takes two *similar functions* into account: "$1-x^2-y^2 = 0$" and "$1.7^2-x^2-y^2 = 0$". The unconstrained



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 4. Visualization of MWs. (a) MW1. (b) MW2. (c) MW3. (d) MW4. (e) MW5. (f) MW6. (g) MW7. (h) MW8. The unconstrained PFs are shown in blue and the red points are the images of the Pareto optimal solutions in the objective space. In (a)–(c) and (e)–(g), the gray area(s) is/are the feasible region(s), in (d) the feasible region is the area bounded by the gray surface and the blue plane, and in (h) the feasible regions are the areas bounded by the gray surfaces and the blue surface.

PF is defined by the former. By applying the local adjustment process to the former, two constraints c_2 and c_3 are generated. In addition, by conducting the local adjustment process on the latter, the constraint c_1 is generated. MW5's constrained PF contains several discrete Pareto optimal solutions. Its distance function is g_1 in (12). MW5 is plotted in Fig. 4(e), where the gray area is the feasible region and the red points are the images of the Pareto optimal solutions in the objective space. The hardness of MW5 is that the images of the discrete Pareto optimal solutions are at the end of the tunnel-like feasible region.

MW6:

$$\min \begin{cases} f_1(\vec{x}) = g_2 x_1 \\ f_2(\vec{x}) = g_2\sqrt{1.1^2 - (f_1/g_2)^2} \end{cases}$$
$$\text{s.t.}\ \ c(\vec{x}) = 1 - (f_1/(1+0.15l))^2 - (f_2/(1+0.75l))^2 \geq 0$$
$$l = \cos\left(6\arctan(f_2/f_1)^4\right)^{10}. \tag{20}$$

This test problem is constructed by making use of two *similar functions*: "$1.1^2 - x^2 - y^2 = 0$" and "$1 - x^2 - y^2 = 0$". The former is used to produce the unconstrained PF. Besides, by conducting the local adjustment process on the latter, the constraint is produced. MW6 belongs to type II as the constraint makes a part of the unconstrained PF infeasible. Its distance function is g_2 in (13). MW6 is plotted in Fig. 4(f), where the gray areas are the feasible regions and the red points are the images of the Pareto optimal solutions in the objective space. MW6 is a highly disconnected CMOP and its feasible regions are distributed irregularly.

MW7:

$$\min \begin{cases} f_1(\vec{x}) = g_3 x_1 \\ f_2(\vec{x}) = g_3\sqrt{1 - (f_1/g_3)^2} \end{cases}$$
$$\text{s.t.}\ \ c_1(\vec{x}) = \left(1.2 + 0.4\sin(4l)^{16}\right)^2 - f_1^2 - f_2^2 \geq 0$$
$$c_2(\vec{x}) = \left(1.15 - 0.2\sin(4l)^8\right)^2 - f_1^2 - f_2^2 \leq 0$$
$$l = \arctan(f_2/f_1). \tag{21}$$

This test problem includes three *similar functions*. "$1 - x^2 - y^2 = 0$" is used to generate the unconstrained PF. In addition, by conducting the local adjustment process on "$1.2 - x^2 - y^2 = 0$" and "$1.15 - x^2 - y^2 = 0$", two constraints c_1 and c_2 are generated, respectively. MW7 is a type-III CMOP, that is, its constrained PF contains a part of the unconstrained PF and a part of the boundary of the feasible region. Its distance function is g_3 in (14). MW7 is plotted in Fig. 4(g), where the gray area is the feasible region and the red points are the images of the Pareto optimal solutions in the objective space. It is difficult to find the Pareto optimal solutions on the boundary of the feasible region because the feasible region near them is very narrow.

MW8:

$$\min \begin{cases} f_1(\vec{x}) = g_2 \prod_{i=1}^{m-1}\cos(0.5\pi x_i) \\ f_{k=2:m-1}(\vec{x}) = g_2\sin(0.5\pi x_{m-k+1})\prod_{i=1}^{m-k}\cos(0.5\pi x_i) \\ f_m(\vec{x}) = g_2\sin(0.5\pi x_1) \end{cases}$$
$$\text{s.t.}\ \ c(\vec{x}) = \left(1.25 - 0.5\sin(6l)^2\right)^2 - f_1^2 - \cdots - f_m^2 \geq 0$$
$$l = \arcsin\left(f_m/\sqrt{f_1^2 + \cdots + f_m^2}\right). \tag{22}$$

This test problem is scalable in terms of objectives and it is constructed by making use of two *similar functions*: "$1-x^2-y^2-z^2-\cdots = 0$" and "$1.25^2-x^2-y^2-z^2-\cdots = 0$". The unconstrained PF is defined by the former. In addition, through the local adjustment process on the latter, the constraint is defined. MW8 is a disconnected type-II CMOP and uses the multimodal distance function [i.e., g_2 in (13)]. Fig. 4(h) shows the relationship between the mth objective and any two other objectives f_k and f_h from $\{f_1, f_2, \ldots, f_{m-1}\}$.

The feasible regions are the areas bounded by the gray surfaces and the blue surface, and the red points are the samples of the images of the Pareto optimal solutions in the objective space.

MW9:

$$\begin{aligned} \min\ & \begin{cases} f_1(\vec{x}) = g_1 x_1 \\ f_2(\vec{x}) = g_1\left(1-(f_1/g_1)^{0.6}\right) \end{cases} \\ \text{s.t.}\ & c(\vec{x}) = \min\{T_1, T_2 \cdot T_3\} \le 0 \\ & T_1 = \left(1-0.64f_1^2-f_2\right)\left(1-0.36f_1^2-f_2\right) \\ & T_2 = 1.35^2-(f_1+0.35)^2-f_2 \\ & T_3 = 1.15^2-(f_1+0.15)^2-f_2. \end{aligned} \tag{23}$$

This test problem considers five *similar functions*. The unconstrained PF is defined by "$1-x^{0.6}-y=0$" and the constraints are produced by "$(1+a_i)^2-(x+a_i)^2-y=0$" and "$1-b_jx^2-y=0$", where $a_i \in \{0.35, 0.15\} (i=1,2)$ and $b_j \in \{0.64, 0.36\} (j=1,2)$. MW9 is a type-IV CMOP because its constrained PF is a part of the boundary of the feasible region. Besides, its distance function is g_1 in (12). As shown in Fig. 5(a), the gray area is the feasible region and the red points are the images of the Pareto optimal solutions in the objective space.

MW10:

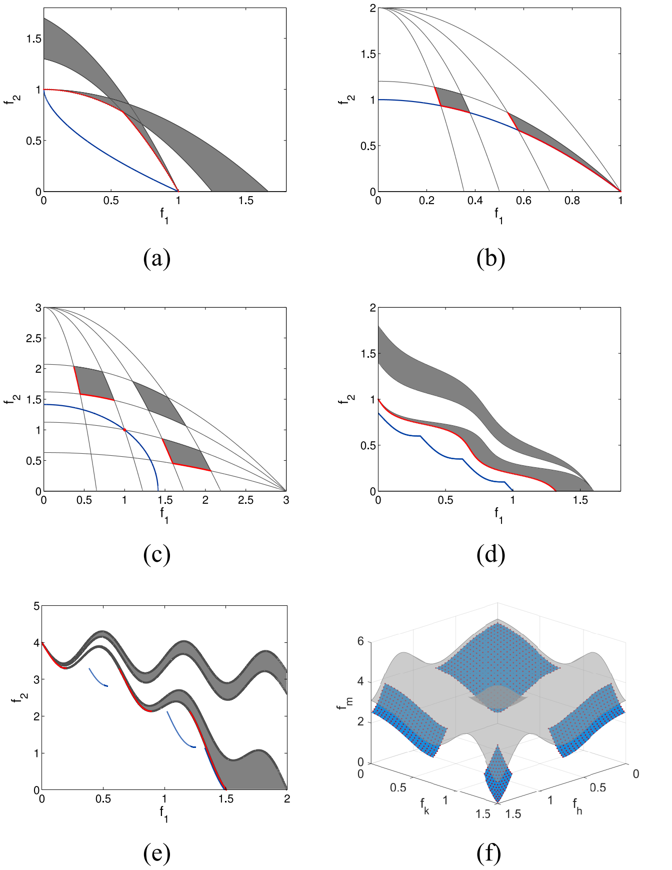
$$\begin{aligned} \min\ & \begin{cases} f_1(\vec{x}) = g_2 x_1^n \\ f_2(\vec{x}) = g_2\left(1-(f_1/g_2)^2\right) \end{cases} \\ \text{s.t.}\ & c_1(\vec{x}) = \left(2-4f_1^2-f_2\right)\left(2-8f_1^2-f_2\right) \ge 0 \\ & c_2(\vec{x}) = \left(2-2f_1^2-f_2\right)\left(2-16f_1^2-f_2\right) \le 0 \\ & c_3(\vec{x}) = \left(1-f_1^2-f_2\right)\left(1.2-1.2f_1^2-f_2\right) \le 0. \end{aligned} \tag{24}$$

This test problem is constructed based on six *similar functions*. The unconstrained PF is defined by "$1-x^2-y=0$" and the constraints are produced by "$2-a_ix^2-y=0$" and "$b_j-b_jx^2-y=0$", where $a_i \in \{2, 4, 8, 16\} (i=1,\ldots,4)$ and $b_j \in \{1.0, 1.2\} (j=1,2)$. MW10 is a disconnected type-III CMOP and adopts the multimodal distance function [i.e., g_2 in (13)]. As shown in Fig. 5(b), the gray areas are the feasible regions and the red points are the images of the Pareto optimal solutions in the objective space. MW10 has island-like feasible regions and the polynomial bias in f_1 prevents an algorithm from finding them.

MW11:

$$\begin{aligned} \min\ & \begin{cases} f_1(\vec{x}) = g_3 x_1 \\ f_2(\vec{x}) = g_3\sqrt{2-(f_1/g_3)^2} \end{cases} \\ \text{s.t.}\ & c_1(\vec{x}) = \left(3-f_1^2-f_2\right)\left(3-2f_1^2-f_2\right) \ge 0 \\ & c_2(\vec{x}) = \left(3-0.625f_1^2-f_2\right)\left(3-7f_1^2-f_2\right) \le 0 \\ & c_3(\vec{x}) = \left(1.62-0.18f_1^2-f_2\right)\left(1.125-0.125f_1^2-f_2\right) \ge 0 \\ & c_4(\vec{x}) = \left(2.07-0.23f_1^2-f_2\right)\left(0.63-0.07f_1^2-f_2\right) \le 0. \end{aligned} \tag{25}$$

This test problem includes nine *similar functions*. The unconstrained PF is produced by "$2-x^2-y^2=0$" and the constraints are generated on the basis of "$3-a_ix^2-y=0$" and "$c_j-b_jx^2-y=0$", where $a_i \in \{0.625, 1, 2, 7\} (i=1,\ldots,4)$, $b_j \in \{0.23, 0.18, 0.125, 0.07\} (j=1,\ldots,4)$, and $c_j \in \{2.07, 1.62, 1.125, 0.63\} (j=1,\ldots,4)$. MW11's constrained PF consists of an isolated point $(1, 1)$ and a part of the boundaries of the feasible regions. In addition, its distance function is g_3 in (14). As shown in Fig. 5(c), the gray areas are the feasible regions and the red points are the images of the Pareto optimal solutions in the objective space. It is an important task for an algorithm to find the isolated optimal solution because it Pareto dominates one patch of the feasible regions.



(a) (b) (c) (d) (e) (f)

Fig. 5. Visualization of MWs. (a) MW9. (b) MW10. (c) MW11. (d) MW12. (e) MW13. (f) MW14. The unconstrained PFs are shown in blue and the red points are the images of the Pareto optimal solutions in the objective space. In (a)–(e), the gray area(s) is/are the feasible region(s), and in (f), the feasible region is the area bounded by the gray surface and the surfaces that define the unconstrained PF (i.e., the four blue parts).

MW12:

$$\begin{aligned} \min\ & \begin{cases} f_1(\vec{x}) = g_1 x_1 \\ f_2(\vec{x}) = g_1\left(0.85-0.8f_1/g_1-0.08\left|\sin(3.2\pi f_1/g_1)\right|\right) \end{cases} \\ \text{s.t.}\ & c_1(\vec{x}) = T_1 \cdot T_4 \le 0,\ c_2(\vec{x}) = T_2 \cdot T_3 \ge 0 \\ & T_1 = 1-0.8f_1-f_2+0.08\sin(2\pi(f_2-f_1/1.5)) \\ & T_2 = 1-0.625f_1-f_2+0.08\sin(2\pi(f_2-f_1/1.6)) \\ & T_3 = 1.4-0.875f_1-f_2+0.08\sin(2\pi(f_2/1.4-f_1/1.6)) \\ & T_4 = 1.8-1.125f_1-f_2+0.08\sin(2\pi(f_2/1.8-f_1/1.6)). \end{aligned} \tag{26}$$

This test problem is constructed through five *similar functions*: "$0.85 - 0.8x - y = 0$" and "$a_i - b_i x - y = 0$", where $a_i \in \{1, 1, 1.4, 1.8\}(i = 1, \ldots, 4)$ and $b_i \in \{0.8, 0.625, 0.875, 1.125\}(i = 1, \ldots, 4)$. By conducting the local adjustment process on the first one, the unconstrained PF is generated. Furthermore, by conducting the local adjustment process on the others, the constraints are generated. MW12 is a type-IV CMOP and uses the biased distance function [i.e., g_1 in (12)]. As shown in Fig. 5(d), the gray areas are the feasible regions and the red points are the images of the Pareto optimal solutions in the objective space.

MW13:

$$\begin{aligned}
\min\ & \begin{cases} f_1(\vec{x}) = g_2 x_1 \\ f_2(\vec{x}) = g_2\big(5 - \exp(f_1/g_2) - 0.5\big|\sin(3\pi f_1/g_2)\big|\big) \end{cases} \\
\text{s.t.}\ & c_1(\vec{x}) = T_1 \cdot T_4 \le 0,\ c_2(\vec{x}) = T_2 \cdot T_3 \ge 0 \\
& T_1 = 5 - \exp(f_1) - 0.5\sin(3\pi f_1) - f_2 \\
& T_2 = 5 - \left(1 + f_1 + 0.5 f_1^2\right) - 0.5\sin(3\pi f_1) - f_2 \\
& T_3 = 5 - (1 + 0.7 f_1) - 0.5\sin(3\pi f_1) - f_2 \\
& T_4 = 5 - (1 + 0.4 f_1) - 0.5\sin(3\pi f_1) - f_2.
\end{aligned} \tag{27}$$

The Taylor's expansion, i.e., "$e^x = 1 + x/1! + x^2/2! + x^3/3! + \cdots$", is considered in this test problem to form a set of *similar functions*. They are defined by "$5 - \alpha_i(x) - y = 0$", where $\alpha_i(x) \in \{e^x, 1 + x + 0.5x^2, 1 + 0.7x, 1 + 0.4x\}(i = 1, \ldots, 4)$. By conducting the local adjustment process on "$5 - e^x - y = 0$", the unconstrained PF is defined. In addition, by conducting the local adjustment process on all the *similar functions*, the constraints are produced. MW13 is a type-III CMOP and has a disconnected geometry of constrained PF. It adopts the multimodal distance function [i.e., g_2 in (13)]. As shown in Fig. 5(e), the gray areas are the feasible regions and the red points are the images of the Pareto optimal solutions in the objective space. When approximating the Pareto optimal solutions of MW13, an algorithm will encounter infeasible barriers.

MW14:

$$\begin{aligned}
\min\ & \begin{cases} f_{k=1:m-1}(\vec{x}) = x_k \\ f_m(\vec{x}) = g_3/(m-1)\sum_{i=1}^{m-1}\big(6 - \exp(f_i) - 1.5\sin\big(1.1\pi f_i^2\big)\big) \end{cases} \\
\text{s.t.}\ & c(\vec{x}) = 1/(m-1)\sum_{i=1}^{m-1}(6.1 - \alpha(i)) - f_m \ge 0 \\
& \alpha(i) = 1 + f_i + 0.5 f_i^2 + 1.5\sin\left(1.1\pi f_i^2\right).
\end{aligned} \tag{28}$$

This test problem is scalable in terms of objectives and it is constructed based on two *similar functions*: "$6 - e^x - y = 0$" and "$6.1 - (1 + x + 0.5x^2) - y = 0$". By implementing the local adjustment process on the former, the unconstrained PF is produced. In addition, by applying the local adjustment process to the latter, the constraint is generated. MW14 is a type-I CMOP and has a disconnected geometry caused by Pareto dominance. The distance function with variable linkages is employed [i.e., g_3 in (14)]. Fig. 5(f) shows the relationship between the mth objective and any two other objectives f_k and f_h from $\{f_1, f_2, \ldots, f_{m-1}\}$. The feasible region is the area bounded by the gray surface and the surfaces that define the unconstrained PF (i.e., the four blue parts), and the red points are the samples of the images of the Pareto optimal solutions in the objective space.

Remark 1: In our test suite, MW4, MW8, and MW14 are scalable in terms of the number of objectives, and all MWs are scalable in terms of the number of decision variables since the distance functions adopted in MWs can be extended to any number of decision variables.

Remark 2: To the best of our knowledge, over 80 research papers have focused on solving current artificial test problems (e.g., CTPs, CFs, and C-DTLZs) during the past two decades. Some recent studies [33]–[35] have reported highly competitive results on these test problems. Compared with current artificial test problems, our test problems can provide new challenges to constrained multiobjective EAs (CMOEAs), which have been introduced in Tables I and II and Figs. 4 and 5, and will be further validated in Section IV.

IV. Experimental Study

In this section, several representative CHTs under the frameworks of NSGA-II and MOEA/D are compared on the proposed test problems. Furthermore, we systematically analyze the performance of different CHTs and briefly discuss the advantages of our test problems.

A. Performance Metrics

In our experiments, three performance metrics were used, namely inverted generational distance (IGD) [36], maximum spread (MS) [37], and generational distance (GD) [38]. Specifically, IGD measures the convergence and diversity of an algorithm, and MS and GD evaluate the degree of coverage and convergence of an algorithm, respectively. In this paper, our analysis was mainly based on IGD, whereas MS and GD were taken as the auxiliary metrics. Note that all the metrics only considered the feasible solutions in the final population.

Suppose that $\mathcal{P}$ is an obtained approximation in an implementation and $\mathcal{P}^*$ is a set of sampling points evenly distributed on the true PF. IGD is calculated as follows:

$$IGD = \frac{1}{|\mathcal{P}^*|}\sum_{i \in \mathcal{P}^*} d(i, \mathcal{P}) \tag{29}$$

where $|\mathcal{P}^*|$ is the cardinality of $\mathcal{P}^*$, and $d(i, \mathcal{P})$ is the minimum Euclidean distance between the ith member in $\mathcal{P}^*$ and all the members in $\mathcal{P}$. If $\mathcal{P}^*$ is large enough, it can well represent the whole PF. For our test problems (except MW5), over 1000 samples are provided in $\mathcal{P}^*$. It is apparent that the smaller the IGD value, the better the performance.

MS is defined as follows:

$$MS = \sqrt{\frac{1}{m}\sum_{k=1}^{m}\left[\frac{\min\left(\overline{\mathcal{P}_k^*}, \overline{\mathcal{P}_k}\right) - \max\left(\underline{\mathcal{P}_k^*}, \underline{\mathcal{P}_k}\right)}{\overline{\mathcal{P}_k^*} - \underline{\mathcal{P}_k^*}}\right]^2} \tag{30}$$

where $\overline{\mathcal{P}_k^*}$ and $\underline{\mathcal{P}_k^*}$ are the maximum and minimum values of the kth objective in $\mathcal{P}^*$, respectively, and $\overline{\mathcal{P}_k}$ and $\underline{\mathcal{P}_k}$ are the maximum and minimum values of the kth objective in $\mathcal{P}$, respectively. The larger the MS value, the better the coverage of $\mathcal{P}$. In addition, for CMOPs with multiple and separated feasible regions, this metric can reflect whether $\mathcal{P}$ covers these feasible regions widely.

GD is formulated as follows:

$$GD = \frac{1}{|\mathcal{P}|}\sum_{i\in\mathcal{P}} d\big(i, \mathcal{P}^*\big) \tag{31}$$

where $|\mathcal{P}|$ is the cardinality of $\mathcal{P}$, and $d(i, \mathcal{P}^*)$ is the minimum Euclidean distance between the ith member in $\mathcal{P}$ and all the members in $\mathcal{P}^*$. The smaller the GD value, the better the convergence performance.

Remark 3: In particular, if all the members in $\mathcal{P}$ are evenly distributed on the true PF, the values of IGD, MS, and GD will be zero, one, and zero, respectively.

B. Representative Constraint-Handling Techniques

Several popular CHTs were chosen for performance comparisons in our experimental studies. Under the framework of NSGA-II, six CHTs were selected.

Constrained dominance principle (CDP) is the simplest and the most commonly used CHT in constrained multiobjective optimization. CDP is first proposed to solve CMOPs in the constrained version of NSGA-II [28]. In CDP, feasible solutions are considered to be consistently better than infeasible ones.

Penalty functions have shown great potential for constrained single-objective optimization. This kind of methods always tunes the penalty factors to provide a proper degree of penalty. However, the penalty factors are generally problem-dependent. This limitation promotes the development of self-adaptive penalty functions, denoted as SP, whose penalty factors are adjusted adaptively based on the feedback information from the search. Woldesenbet *et al.* [39] designed a new SP for CMOPs and achieved competitive results.

In stochastic ranking (SR) [40], when an infeasible solution is compared with another solution, a user-defined parameter (i.e., P_f) is utilized to determine the comparison criterion, that is, comparing their constraint violations with the probability $(1 - P_f)$ or comparing their objective values with the probability P_f. Geng *et al.* [41] proposed an infeasible elitists and SR-based algorithm. Considering that the objective values may be incomparable based on Pareto dominance between two solutions, this algorithm assigns each solution a scalar value based on its ranking in nondominated sorting and its crowding distance.

In ϵ-constrained method [42] (called ϵ in this paper), a decreasing ϵ-level is defined to relax the constraint violations of infeasible solutions. When ϵ reduces to zero, the ϵ-constrained method is equivalent to CDP. Under this condition, infeasible solutions will be eliminated. Qian *et al.* [43] exploited this kind of CHT to solve CMOPs.

By taking constraints as one or more extra objectives, researchers proposed some multiobjective optimization-based CHTs (called MO in this paper). A successful attempt is the infeasible driven EA proposed by Ray *et al.* [44]. When not all solutions are feasible, this algorithm maintains a certain number of desirable infeasible solutions during the search, aiming to guide the population toward the boundary of the feasible region from both feasible and infeasible sides.

Researchers have developed some hybrid CHTs that combine several popular CHTs together [45]. An example is the adaptive tradeoff model (ATM) presented by Wang *et al.* [46]. ATM divides the search into three scenarios. If there is no feasible solution, the population is ranked by nondominance sorting with an extra objective defined by constraint violation, and the first half of nondominated solutions are selected in ascending order of constraint violation. This process is repeated until the population reaches its predefined size. If the population contains both feasible and infeasible solutions, ATM adopts a penalty function to select solutions for the next generation. When the population is entirely feasible, the solutions are compared based only on their objective values. Despite that ATM is originally designed for constrained single-objective optimization, its performance is still investigated by combining it with NSGA-II in this paper.

To make the comparisons fair, the above six CHTs (i.e., CDP [28], SP [39], SR [41], ϵ [43], MO [44], and ATM [46]) employed the original NSGA-II as the optimization algorithm, and the corresponding CMOEAs were denoted as CDP-NSGA-II, SP-NSGA-II, SR-NSGA-II, ϵ-NSGA-II, MO-NSGA-II, and ATM-NSGA-II, respectively. Note that, some special operators were not considered, such as the elitist preservations in [41] and [43].

Besides, some CHTs are developed under the framework of MOEA/D [29] or MOEA/D-DE [47]. For instance, Jan and Zhang [48] introduced a new SP into the Tchebycheff aggregation function. They also compared two CHTs, i.e., CDP and SR in [49]. Yang *et al.* [50] improved the ϵ-constrained method and investigated its performance in MOEA/D-DE. Similarly, these four CHTs (i.e., CDP [49], SP [48], SR [49], and ϵ [50]) adopted the original MOEA/D in our comparison, and their corresponding CMOEAs were denoted as CDP-MOEA/D, SP-MOEA/D, SR-MOEA/D, and ϵ-MOEA/D, respectively.

C. Parameter Settings

Our experiments were implemented under the following parameter settings.

1) Number of independent runs: 100.
2) Maximum generation number: $G = 600$.
3) Population size: $N_P = 100$.
4) Number of decision variables: $n = 15$.
5) Number of objectives for MW4, MW8, and MW14: $m = 3$.
6) Number of objectives for MW1–MW3, MW5–MW7, and MW9–MW13: $m = 2$.

Both NSGA-II and MOEA/D used simulated binary crossover (SBX) [28] and polynomial mutation (PM) [28] as the reproduction operators.

1) *SBX:* Crossover probability $p_c = 0.9$ and distribution index $\eta_c = 20$.
2) *PM:* Mutation probability $p_m = 1/n$ and distribution index $\eta_m = 20$.

In addition, some preliminary experiments show that the parameter settings of ϵ-NSGA-II provided in [43] seem not to be effective for our test problems. Therefore, we tuned them as follows.

1) $\theta = 0.1N_P$ and $T_c = 0.6G$.
2) $cp = (-5 - \log \epsilon_0)/\log(0.05)$, as suggested in [50].

D. Comparisons Under the Framework of NSGA-II

First, we compared the results obtained by six CHTs under the framework of NSGA-II. Tables S-R-I–S-R-III in the supplementary file show the results in terms of IGD, MS, and GD, respectively, including the average and standard deviation over 100 independent runs. In these tables, the best result of each test problem was highlighted in boldface.

As shown in Table S-R-I, the results in terms of IGD are quite divergent, that is, different CHTs show promising performance on different types of CMOPs. To be specific, ϵ-NSGA-II obtains better results than other compared algorithms on five test problems (i.e., MW1, MW2, MW6, MW8, and MW10), which are mainly type-II CMOPs; SP-NSGA-II performs the best on MW3, MW7, and MW13 that belong to type III; MO-NSGA-II significantly outperforms others on the type-IV test problems (i.e., MW9, MW11, and MW12); and ATM-NSGA-II wins on MW4 and MW14 that are type-I CMOPs. In addition, SR-NSGA-II approximates the constrained PF well on MW5, which is the only one with a discrete geometry.

As shown in Table S-R-II, the cases on MS are roughly similar to those on IGD. ϵ-NSGA-II obtains the best results on seven test problems (i.e., MW1–MW3, MW6, MW8, MW10, and MW14), for which ϵ-NSGA-II also exhibits superior performance in terms of IGD. SP-NSGA-II outperforms the five competitors on MW4, MW7, and MW13, and gets near-best result on MW3. MO-NSGA-II achieves the best on MW5, MW9, MW11, and MW12. Despite that ATM-NSGA-II is not the best algorithm on MW4 and MW14, it still has competitive results in terms of MS.

It can be observed from Table S-R-III that ϵ-NSGA-II performs the best in terms of GD on a majority of test problems (i.e., MW1–MW4, MW6, MW7, MW9, MW10, and MW14), which implies that it shows better performance of convergence than others in most cases. On MW8 and MW11, CDP-NSGA-II can well converge toward the constrained PFs. In addition, SP-NSGA-II, MO-NSGA-II, and ATM-NSGA-II beat others on MW13, MW12, and MW5, respectively. The above results also indicate that sometimes good values of GD do not correspond to good values of IGD.

From the above discussions, for each test problem, an algorithm that performs the best in terms of IGD generally has the best or near-best result of MS. This demonstrates that a good spread of population will facilitate an algorithm to approximate the whole constrained PF. It is mainly attributed to the fact that the feasible regions of our test problems are generally narrow and separated; thus, some parts of the constrained PF will be lost without exploring the feasible regions widely. In addition, under the framework of NSGA-II, different CHTs have their advantages on different types of CMOPs, which will be further discussed later.

TABLE III
RANKINGS OF CHTs UNDER THE FRAMEWORK OF NSGA-II BASED ON THE WILCOXON'S RANK-SUM TEST AT A 0.05 SIGNIFICANCE LEVEL

Type	Problem	1st $(b/e/w)$	2nd $(b/e/w)$
I	MW2	ϵ (5/0/0)	CDP, SP, ATM (1/3/1)
	MW4	ATM (2/3/0)	CDP, ϵ, MO (1/4/0)
	MW14	ATM (2/3/0)	CDP, SP, ϵ (1/4/0)
II	MW1	ϵ (5/0/0)	SP (2/2/1)
	MW5	SR (4/0/1)	MO (2/3/0)
	MW6	ϵ (5/0/0)	SP (1/3/1)
	MW8	ϵ (4/1/0)	CDP (1/4/0)
III	MW3	SP (5/0/0)	ϵ (4/0/1)
	MW7	SP (5/0/0)	SR (3/1/1)
	MW10	ϵ (5/0/0)	SR (2/2/1)
	MW13	SP (5/0/0)	CDP, ATM (3/1/1)
IV	MW9	MO (5/0/0)	SP (1/3/1)
	MW11	MO (5/0/0)	ϵ (1/3/1)
	MW12	MO (5/0/0)	SP (1/3/1)

TABLE IV
RANKINGS OF CHTs UNDER THE FRAMEWORK OF MOEA/D BASED ON THE WILCOXON'S RANK-SUM TEST AT A 0.05 SIGNIFICANCE LEVEL

Type	Problem	1st $(b/e/w)$	2nd $(b/e/w)$
I	MW2	ϵ (2/1/0)	SP (0/3/0)
	MW4	CDP, ϵ (2/1/0)	SP, SR (0/1/2)
	MW14	ϵ (3/0/0)	CDP, SP, SR (0/2/1)
II	MW1	ϵ (3/0/0)	SR (2/0/1)
	MW5	ϵ (3/0/0)	CDP, SP (1/1/1)
	MW6	ϵ (2/1/0)	SP, SR (0/2/1)
	MW8	ϵ (3/0/0)	CDP, SP, SR (0/2/1)
III	MW3	ϵ (3/0/0)	CDP (1/1/1)
	MW7	ϵ (3/0/0)	CDP, SP (1/1/1)
	MW10	ϵ (1/2/0)	SP (0/2/1)
	MW13	CDP, SP (2/1/0)	ϵ (1/0/2)
IV	MW9	CDP, SP (2/1/0)	ϵ (1/0/2)
	MW11	ϵ (3/0/0)	CDP, SP (1/1/0)
	MW12	SP (3/0/0)	CDP (2/0/1)

E. Comparisons Under the Framework of MOEA/D

This section compared the results derived from four CHTs combined with MOEA/D. Tables S-R-IV–S-R-VI in the supplementary file provide the results in terms of IGD, MS, and GD, respectively. They include the average and standard deviation over 100 independent runs. Similarly, the best result of each test problem was highlighted in boldface.

As shown in Table S-R-IV, ϵ-MOEA/D provides better results than other compared algorithms on ten test problems (i.e., MW1–MW3, MW5–MW8, MW10, MW11, and MW14). Thus, ϵ-MOEA/D can provide feasible solutions with good convergence and diversity on most of test problems, mainly including types I–III. In addition, CDP-MOEA/D achieves the best results on MW4 and MW9, and SP-MOEA/D shows encouraging performance on MW12 and MW13.

Again, similar phenomenon can be observed from Table S-R-V in terms of MS. ϵ-MOEA/D obtains the best results on nine test problems (i.e., MW1–MW3, MW5–MW8, MW11, and MW14), and very near-best result on MW10. In addition, CDP-MOEA/D performs the best on MW4, MW9, and MW10, and SP-MOEA/D outperforms others on MW12 and MW13.

From Table S-R-VI, it can be seen that ϵ-MOEA/D performs the best in terms of GD, which suggests that it has a better overall ability of convergence. It achieves better results than other compared algorithms on 11 test problems (i.e., MW1–MW8, MW11, MW13, and MW14). With regard to the remaining three test problems (i.e., MW9, MW10, and MW12), SP-MOEA/D surpasses others. It is interesting to note that although CDP-MOEA/D has the best results on MW4 and MW9 in terms of both IGD and MS, the corresponding results provided by CDP-MOEA/D are not good in terms of GD.

In summary, ϵ-MOEA/D has the best overall performance under the framework of MOEA/D, especially on types I–III. Similar to NSGA-II, on each test problem, the best IGD value is supported by the best or near-best MS value, which means that maintaining a good spread of population plays an important role in solving our test problems.

F. Performance Analysis

In order to further analyze the performance of CHTs, the Wilcoxon's rank-sum test at a 0.05 significance level was implemented to demonstrate whether the result obtained by an algorithm is significantly different from that resulting from another compared algorithm in terms of IGD on each test problem. In this paper, "$b/e/w$" indicate the number of CHTs that the current one is significantly better than, equivalent to, and significantly worse than, respectively. Then, CHTs are ranked in the lexicographical order: a CHT that has a higher "b" will have a better ranking; if two CHTs have the same "b", their values of "e" will be compared, and so forth; and if two CHTs have the same "$b/e/w$", they will share the same ranking. Clearly, a CHT with a better ranking indicates that it has better performance. The rankings of different CHTs under the frameworks of NSGA-II and MOEA/D are shown in Tables III and IV, respectively. Due to the space limitation, these tables only show the CHTs that rank the first (i.e., 1st) and the second (i.e., 2nd). First, based on the rankings in Table III, we provide the following analysis on the performance of different CHTs under the framework of NSGA-II.

1) On the type-II CMOPs, the ϵ-constrained method performs the best. This method relaxes the constraint violations of all infeasible solutions in the early search process; thus, some infeasible solutions satisfy the ϵ-level and are considered as feasible solutions (called pseudo-feasible solutions). Under this condition, these pseudo-feasible solutions will be evolved toward the unconstrained PF. Due to the fact that the constrained PF of each type-II CMOP is a part of the unconstrained PF, the population will promptly approach the constrained PF. As ϵ reduces to zero, the found objective-optimal solutions that are in the infeasible region will be eliminated and the real feasible solutions are maintained in the population. In a word, the ϵ-constrained method switches from "optimality priority" to "feasibility priority". This property makes the ϵ-constrained method suitable for this type of CMOPs. In addition, SR outperforms others on MW5 (i.e., the test problem with the discrete constrained PF). SR defines a certain probability that an infeasible solution is compared with others based on objectives. Therefore, some infeasible solutions with good objective values are likely to be preserved. This property encourages the infeasible solutions near the discrete Pareto optimal solutions. In contrast, the ϵ-constrained method will eliminate the promising infeasible solutions with good objective values after ϵ reduces to zero. As a result, it is not as competitive as SR on MW5.
2) ATM performs the best on the type-I test problems. Since their feasible regions contain some narrow parts, it is difficult to find feasible solutions in these narrow parts. For ATM, as the feasibility proportion of population rises, the degree of penalty on infeasible solutions decreases (i.e., [46, Property 3]). That is, the infeasible solutions with relatively small constraint violations are likely to survive if there exist enough feasible solutions in the population, which encourages the search around these narrow parts. The ϵ-constrained method obtains competitive results on these test problems. The reason is similar to that explained in type II. But this method cannot well exploit infeasible solutions, making it not as good as ATM.
3) On the type-III CMOPs, SP outperforms others. SP prefers those solutions with better objective values and lower constraint violations. Therefore, during the evolution, the population is capable of approaching the unconstrained PF and the boundary of the feasible region simultaneously. Moreover, in the later stage of evolution, contrary to ATM, SP increases the degree of penalty with the increase of the feasibility proportion of population [39]. This property ensures the feasibilities of solutions. Besides, the ϵ-constrained method achieves the best result on MW10. It is because over three-fourths of the constrained PF is the unconstrained PF. Therefore, this method has an advantage in solving MW10, as illustrated previously.
4) MO clearly surpasses others on the type-IV CMOPs. There are two main reasons:
   a) When not all solutions are feasible, this method always preserves a certain number of infeasible solutions.
   b) It considers the constraint violation as an extra objective when infeasible solutions are compared with each other, which supports the infeasible ones with lower constraint violations.

   As the constrained PF is a part of the boundary of the feasible region in each of these test problems, the population can approximate the boundary of the feasible region from both feasible and infeasible sides by utilizing such infeasible solutions.

Subsequently, based on the rankings in Table IV, we analyze the performance of the compared CHTs under the framework of MOEA/D in the following.

1) Obviously, the ϵ-constrained method shows the best performance on types I–III, for which the constrained PFs entirely/partly come from the unconstrained PFs. As analyzed previously, the ϵ-constrained method has natural advantages in approaching the unconstrained PF. Different from ϵ-NSGA-II, ϵ-MOEA/D achieves good performance on type III. This is mainly attributed to MOEA/D's ability of maintaining diversity via a set of

evenly distributed weights. As ϵ decreases, the infeasible solutions close to the unconstrained PF can be pulled to the boundary of the feasible region through the weights (note that these infeasible solutions are directly eliminated under the framework of NSGA-II). Actually, when the boundary of the feasible region is not very complicated, this mechanism is also effective, even on the type-IV test problems (e.g., MW11).

2) However, for the type-IV CMOPs, the superiority of the ϵ-constrained method degrades because the nonlinearity of the boundary of the feasible region increases. For these test problems, overall, CDP and SP perform better than the ϵ-constrained method. This phenomenon can be explained in the following. SP maintains two penalty factors and gives a stronger punishment to the infeasible solutions with higher constraint violations than those with lower constraint violations. This property enables the population to converge toward the boundary of the feasible region from both the feasible and infeasible sides. In addition, with the aid of uniformly distributed weights, CDP can concentrate on the search of the boundary of the feasible region.

G. Comparisons Cross Two Frameworks

The comparisons cross two frameworks (i.e., NSGA-II and MOEA/D) were conducted in this section. The four CHTs used in both NSGA-II and MOEA/D in the previous experiments were considered: CDP, SP, SR, and ϵ. As shown in Table S-R-VII in the supplementary file, for each CHT, the results of IGD obtained under the frameworks of NSGA-II and MOEA/D were compared on each test problem, and the better one was highlighted in boldface. Note that the results in Table S-R-VII were directly taken from Tables S-R-I and S-R-IV.

It can be seen from Table S-R-VII that NSGA-II performs better than MOEA/D on most test problems, which are mainly CMOPs with disconnected/discrete constrained PFs. While, MOEA/D is better than NSGA-II on MW4, and similar to NSGA-II on MW9, MW11, and MW12, which are mainly CMOPs with connected constrained PFs. Therefore, MOEA/D is sensitive to the geometry of the constrained PF. The reason is straightforward: the disconnected constrained PF contains some disconnected segments caused by the infeasible regions and/or Pareto dominance; thus, the weight vectors distributed outside these disconnected segments have no intersection with the constrained PF. This will result in several weight vectors corresponding to the same Pareto optimal solution. In this way, there exist many duplicates in the final population, leading to two consequences: 1) the computational resources are wasted and 2) the population cannot cover the constrained PF as well as that obtained by NSGA-II.

H. Comparisons Under the Framework of NSGA-III on MWs With Higher Numbers of Objectives

As pointed out in [51], the proportion of nondominated solutions in the population grows exponentially with the increase of the number of objectives. Therefore, multiobjective EAs (e.g., NSGA-II [28]) that employ Pareto dominance as a major selection criterion are not able to provide sufficient selection pressure to guide the population toward the true PF. Due to this fact, a well-known many-objective EA, i.e., NSGA-III [52], was considered as the optimization algorithm on MW4, MW8, and MW14 with higher numbers of objectives. Under the framework of NSGA-III, five representative CHTs (i.e., CDP [28], SP [39], SR [40], ϵ [42], and ATM [46]) were compared. The corresponding algorithms were denoted as CDP-NSGA-III (i.e., the constrained version of NSGA-III in [21]), SP-NSGA-III, SR-NSGA-III, ϵ-NSGA-III, and ATM-NSGA-III, respectively. NSGA-III works with a set of weight vectors that are predefined according to the number of objectives and the population size. For the CHT based on multiobjective optimization (i.e., MO [44]), the population is divided into a feasible subpopulation and an infeasible subpopulation, and they are sorted separately. Note that, the infeasible subpopulation has a varying size and an extra objective measured by the constraint violation of an infeasible solution. Therefore, MO is absent in the comparison, since NSGA-III is not able to process these two subpopulations simultaneously using one set of weight vectors. Besides, as a hybrid CHT, ATM treats the constraints as an extra objective if there is no feasible solution in the population. For the same reason, we used CDP as the replacement in this scenario.

For MW4, MW8, and MW14, we tested five numbers of objectives, i.e., $m = 3$, 5, 8, 10, and 15, and the number of decision variables was set as $n = m - 1 + k$, where $k = 13$. According to the suggestions in [52], the number of weight vectors (N_W) and the population size (N_P) for different numbers of objectives were summarized in Table S-R-VIII in the supplementary file. For other parameter settings, they were consistent with the suggestions in Section IV-C.

Table S-R-IX in the supplementary file provides the average and standard deviation of IGD values obtained by the five compared algorithms over 100 independent runs. It can be seen that ϵ-NSGA-III surpasses others on all the test problems except MW4 with $m = 3$ and 5 and MW14 with $m = 8$, where CDP-NSGA-III and ATM-NSGA-III are better. To be specific, CDP-NSGA-III achieves the best on MW4 with $m = 3$ and 5, and ATM-NSGA-III obtains the best result on MW14 with $m = 8$. In addition, the Wilcoxon's rank-sum test at a 0.05 significance level was conducted to demonstrate the statistical differences between the results obtained by the best algorithm and its competitors on each test problem. We can observe that ϵ-NSGA-III is the best algorithm and CDP-NSGA-III is also competitive.

The excellent performance of ϵ-NSGA-III is mainly attributed to the fact that the constrained PFs of MW4, MW8, and MW14 are derived from the unconstrained PFs. As analyzed in Section IV-F, the ϵ-constrained method has natural advantages in finding the unconstrained PF. Besides, the reason for the competitive performance of CDP-NSGA-III is obvious. With the aid of uniformly distributed weight vectors, CDP is able to provide sufficient selection pressure for solutions to approach the feasible region and make the population approximate the constrained PF from diverse directions. This is similar to the situation in CDP-MOEA/D.

I. Advantages of the Proposed Test Problems

Based on the above comparisons and analyses, the advantages of the proposed test problems can be summarized as follows.

1) *Proper Difficulties:* As revealed by the experimental results, our test problems can distinguish different algorithms through performance comparisons.
2) *Diverse Characteristics (as Shown in Table I):* These characteristics help us to systematically investigate an algorithm's performance and understand its strengths and weaknesses, which plays an important role in further enhancing its performance.
3) *Four Different Types:* As these four types are extracted from real-world CMOPs, they can sufficiently reflect the features of practical engineering problems. By testing an algorithm on different types, we can ascertain the types on which it shows superiority. This will provide important information for us to apply an algorithm to real-world CMOPs reasonably.

V. Conclusion

Artificial test problems can attract more researchers to design EAs for solving CMOPs. However, a careful investigation has demonstrated that current artificial test problems are not well designed, since they more or less ignore some important characteristics of real-world CMOPs: small feasibility ratio, sufficient nonlinearity of constraints, multiple constraints, scalable number of objectives, high-dimensional decision vector, and so on.

Recognizing their limitations, this paper presented a new constraint construction method, which can be considered as a guideline to design CMOPs. Then, a test suite of 14 instances was suggested based on our constraint construction method. We also equipped them with different kinds of distance functions. Moreover, to promote the understanding on the performance of different CHTs, several representative CHTs were compared under the frameworks of NSGA-II and MOEA/D on the proposed test problems.

Under the framework of NSGA-II, we found that: 1) the ϵ-constrained method [43] can well solve types I and II of CMOPs; 2) ATM [46] shows remarkable superiority on the type-I CMOPs with narrow feasible regions; 3) SP [39] can well approximate the unconstrained PF and the boundary of the feasible region simultaneously on type III; and 4) MO [44] significantly outperforms others on type IV. In addition, under the framework of MOEA/D, the ϵ-constrained method [50] has advantages on types I–III of CMOPs, and CDP [49] and SP [48] are more successful on the type-IV CMOPs. We also observed that the performance of some CHTs (e.g., CDP) is improved due to the fact that MOEA/D has the capability to maintain the diversity. However, as MOEA/D is sensitive to the geometry of the constrained PF, CHTs combined with MOEA/D performs not as well as combined with NSGA-II on the disconnected constrained PFs.

Based on the experimental comparisons, we would like to give the following suggestions when designing a CHT:

1) For the type-I CMOPs, good population diversity should be maintained to avoid the loss of some parts of the unconstrained PF.
2) For the type-II CMOPs, we can follow the principle that switches from "objective priority" to "constraint priority", like the ϵ-constrained method.
3) For CMOPs with type III, an effective way is to keep solutions with better objective values and lower constraint violations simultaneously, like self-adaptive penalty functions.
4) For the type-IV CMOPs, more focuses should be put on constraints and some mechanisms should also be incorporated to preserve good infeasible solutions.

The source code of MW is available from http://www.escience.cn/people/yongwang1/index.html.

References

[1] H. Farzin, M. Fotuhi-Firuzabad, and M. Moeini-Aghtaie, "A stochastic multi-objective framework for optimal scheduling of energy storage systems in microgrids," *IEEE Trans. Smart Grid*, vol. 8, no. 1, pp. 117–127, Jan. 2017.

[2] G. Sun, T. Pang, J. Fang, G. Li, and Q. Li, "Parameterization of criss-cross configurations for multiobjective crashworthiness optimization," *Int. J. Mech. Sci.*, vols. 124–125, pp. 145–157, May 2017.

[3] M. Gong, Z. Wang, Z. Zhu, and L. Jiao, "A similarity-based multiobjective evolutionary algorithm for deployment optimization of near space communication system," *IEEE Trans. Evol. Comput.*, vol. 21, no. 6, pp. 878–897, Dec. 2017.

[4] A. Zhou *et al.*, "Multiobjective evolutionary algorithms: A survey of the state of the art," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 23–49, 2011.

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

[6] Z. Fan *et al.*, "Difficulty adjustable and scalable constrained multi-objective test problem toolkit," *arXiv preprint arXiv:1612.07603*, 2016.

[7] C.-F. Juang and Y.-T. Yeh, "Multiobjective evolution of biped robot gaits using advanced continuous ant-colony optimized recurrent neural networks," *IEEE Trans. Cybern.*, vol. 48, no. 6, pp. 1910–1922, Jun. 2018.

[8] Y.-D. Hong and B. Lee, "Evolutionary optimization for optimal hopping of humanoid robots," *IEEE Trans. Ind. Electron.*, vol. 64, no. 2, pp. 1279–1283, Feb. 2017.

[9] J. Fang, G. Sun, N. Qiu, G. P. Steven, and Q. Li, "Topology optimization of multi-cell tubes under out-of-plane crushing using a modified artificial bee colony algorithm," *J. Mech. Design*, vol. 139, no. 7, 2017, Art. no. 071403.

[10] G. Sun, H. Zhang, J. Fang, G. Li, and Q. Li, "Multi-objective and multi-case reliability-based design optimization for tailor rolled blank (TRB) structures," *Struct. Multidiscipl. Optim.*, vol. 55, no. 5, pp. 1899–1916, 2017.

[11] G. Sun, H. Zhang, R. Wang, X. Lv, and Q. Li, "Multiobjective reliability-based optimization for crashworthy structures coupled with metal forming process," *Struct. Multidiscipl. Optim.*, vol. 56, no. 6, pp. 1571–1578, 2017.

[12] Z. Fan *et al.*, "An improved epsilon constraint-handling method in MOEA/D for CMOPs with large infeasible regions," in *Proc. IEEE Symp. Comput. Intell. (SSCI)*, 2016, pp. 1–8.

[13] B.-C. Wang, H.-X. Li, Q. Zhangi, and Y. Wang, "Decomposition-based multiobjective optimization for constrained evolutionary optimization," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published. doi: 10.1109/TSMC.2018.2876335.

[14] R. Tanabe and A. Oyama, "A note on constrained multi-objective optimization benchmark problems," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, 2017, pp. 1127–1134.

[15] N. Srinivas and K. Deb, "Multiobjective function optimization using nondominated sorting genetic algorithms," *Evol. Comput.*, vol. 2, no. 3, pp. 221–248, 1995.

[16] M. Tanaka, H. Watanabe, Y. Furukawa, and T. Tanino, "GA-based decision support system for multicriteria optimization," in *Proc. IEEE Int. Conf. Syst. Man Cybern.*, vol. 2, 1995, pp. 1556–1561.

[17] A. Osyczka and S. Kundu, "A new method to solve generalized multicriteria optimization problems using the simple genetic algorithm," *Struct. Optim.*, vol. 10, no. 2, pp. 94–99, 1995.

[18] K. Deb, A. Pratap, and T. Meyarivan, "Constrained test problems for multi-objective evolutionary optimization," in *Proc. 1st Int. Conf. Evol. Multi Criterion Optim. (EMO)*, 2001, pp. 284–298.

[19] Q. Zhang *et al.*, "Multiobjective optimization test instances for the CEC 2009 special session and competition," Univ. Essex, Colchester, U.K., and Nanyang Technol. Univ., Singapore, Rep. CES-487, 2008.

[20] J.-P. Li, Y. Wang, S. Yang, and Z. Cai, "A comparative study of constraint-handling techniques in evolutionary constrained multiobjective optimization," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, 2016, pp. 4175–4182.

[21] H. Jain and K. Deb, "An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: Handling constraints and extending to an adaptive approach," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 602–622, Aug. 2014.

[22] T. Ray and K. M. Liew, "A swarm metaphor for multiobjective design optimization," *Eng. Optim.*, vol. 34, no. 2, pp. 141–153, 2002.

[23] W. Gong, Z. Cai, and L. Zhu, "An efficient multiobjective differential evolution algorithm for engineering design," *Struct. Multidiscipl. Optim.*, vol. 38, no. 2, pp. 137–157, 2009.

[24] M. G. Parsons, "Formulation of multicriterion design optimization problems for solution with scalar numerical optimization methods," *J. Ship Res.*, vol. 48, no. 1, pp. 61–67, 2004.

[25] C. A. Coello Coello and G. T. Pulido, "Multiobjective structural optimization using a microgenetic algorithm," *Struct. Multidiscipl. Optim.*, vol. 30, no. 5, pp. 388–403, 2005.

[26] T. Ray, K. Tai, and K. C. Seow, "An evolutionary algorithm for multiobjective optimization," *Eng. Optim.*, vol. 33, no. 3, pp. 399–424, 2001.

[27] Y. Guo, X. Cao, and J. Zhang, "Multiobjective evolutionary algorithm with constraint handling for aircraft landing scheduling," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, 2008, pp. 3657–3662.

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

[29] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.

[30] A. Trivedi, D. Srinivasan, K. Sanyal, and A. Ghosh, "A survey of multiobjective evolutionary algorithms based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 21, no. 3, pp. 440–462, Jun. 2017.

[31] K. Deb, "Multi-objective genetic algorithms: Problem difficulties and construction of test problems," *Evol. Comput.*, vol. 7, no. 3, pp. 205–230, 1999.

[32] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Trans. Evol. Comput.*, vol. 10, no. 5, pp. 477–506, Oct. 2006.

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

[34] C. Peng, H.-L. Liu, and F. Gu, "An evolutionary algorithm with directed weights for constrained multi-objective optimization," *Appl. Soft Comput.*, vol. 60, pp. 613–622, Nov. 2017.

[35] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A reference vector guided evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 20, no. 5, pp. 773–791, Oct. 2016.

[36] P. A. N. Bosman and D. Thierens, "The balance between proximity and diversity in multiobjective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 174–188, Apr. 2003.

[37] C. K. Goh and K. C. Tan, "An investigation on noisy environments in evolutionary multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 11, no. 3, pp. 354–381, Jun. 2007.

[38] D. A. V. Veldhuizen and G. B. Lamont, "Evolutionary computation and convergence to a Pareto front," in *Proc. Late Breaking Papers Genet. Program. Conf.*, 1998, pp. 221–228.

[39] Y. G. Woldesenbet, G. G. Yen, and B. G. Tessema, "Constraint handling in multiobjective evolutionary optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 3, pp. 514–525, Jun. 2009.

[40] T. P. Runarsson and X. Yao, "Stochastic ranking for constrained evolutionary optimization," *IEEE Trans. Evol. Comput.*, vol. 4, no. 3, pp. 284–294, Sep. 2000.

[41] H. Geng, M. Zhang, L. Huang, and X. Wang, "Infeasible elitists and stochastic ranking selection in constrained evolutionary multiobjective optimization," in *Proc. Int. Conf. Simulat. Evol. Learn.*, 2006, pp. 336–344.

[42] T. Takahama and S. Sakai, "Constrained optimization by the ϵ constrained differential evolution with gradient-based mutation and feasible elites," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Vancouver, BC, Canada, 2006, pp. 1–8.

[43] F. Qian, B. Xu, R. Qi, and H. Tianfield, "Self-adaptive differential evolution algorithm with α-constrained-domination principle for constrained multi-objective optimization," *Soft Comput.*, vol. 16, no. 8, pp. 1353–1372, 2012.

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

[45] B. Y. Qu and P. N. Suganthan, "Constrained multi-objective optimization algorithm with an ensemble of constraint handling methods," *Eng. Optim.*, vol. 43, no. 4, pp. 403–416, 2011.

[46] Y. Wang, Z. Cai, Y. Zhou, and W. Zeng, "An adaptive tradeoff model for constrained evolutionary optimization," *IEEE Trans. Evol. Comput.*, vol. 12, no. 1, pp. 80–92, Feb. 2008.

[47] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, Apr. 2009.

[48] M. A. Jan and Q. Zhang, "MOEA/D for constrained multiobjective optimization: Some preliminary experimental results," in *Proc. Comput. Intell.*, 2010, pp. 1–6.

[49] M. A. Jan and R. A. Khanum, "A study of two penalty-parameterless constraint handling techniques in the framework of MOEA/D," *Appl. Soft Comput.*, vol. 13, no. 1, pp. 128–148, 2013.

[50] Z. Yang, X. Cai, and Z. Fan, "Epsilon constrained method for constrained multiobjective optimization problems: Some preliminary results," in *Proc. Companion Publication Annu. Conf. Genet. Evol. Comput.*, 2014, pp. 1181–1186.

[51] R. C. Purshouse and P. J. Fleming, "On the evolutionary optimization of many conflicting objectives," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 770–784, Dec. 2007.

[52] 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 Trans. Evol. Comput.*, vol. 18, no. 4, pp. 577–601, Aug. 2014.

Zhongwei Ma received the M.S. degree in computer science and technology from Xiangtan University, Xiangtan, China, in 2016. He is currently pursuing the Ph.D. degree in computer science and technology with Central South University, Changsha, China.

His current research interests include evolutionary computation, constrained optimization, dynamic optimization, and automotive lightweight design.

Yong Wang (M'08–SM'17) received the Ph.D. degree in control science and engineering from the Central South University, Changsha, China, in 2011.

He is a Professor with the School of Automation, Central South University. His current research interests include theory, algorithm design, and interdisciplinary applications of computational intelligence.

Dr. Wang was a recipient of the Highly Cited Researcher Award in Computer Science by Web of Science in 2017 and 2018. He is an Associate Editor of *Swarm and Evolutionary Computation*.