AIMS Mathematics

http://www.aimspress.com/journal/Math

AIMS Mathematics, 6(6): 6220–6241.
DOI: 10.3934/math.2021365
Received: 06 March 2021
Accepted: 07 April 2021
Published: 09 April 2021

Research article

A constraint handling technique using compound distance for solving constrained multi-objective optimization problems

Jiawei Yuan*

Guangdong University of Technology, Guangdong, China

* **Correspondence:** Email: yuan.jiaweipaper@aliyun.com.

Abstract: Guiding the working population to evenly explore the valuable areas which are not dominated by feasible solutions is important in the process of dealing with constrained multi-objective optimization problems (CMOPs). To this end, according to the angular distance and ℓ_p-norm, this paper introduces a new compound distance to measure individual's search diameter in the objective space. After that, we propose a constraint handling technique using the compound distance and embed it in evolutionary algorithm for solving CMOPs. In the proposed algorithm, the individuals with large search diameters in the valuable areas are given priority to be preserved. This can prevent the working population from getting stuck in the local areas and then find the optimal solutions for CMOPs more effectively. A series of numerical experiments show that the proposed algorithm has better performance and robustness than several existing state-of-the-art constrained multi-objective evolutionary algorithms in dealing with different CMOPs.

Keywords: compound distance; constraint handling; multi-objective optimization; evolutionary algorithm
Mathematics Subject Classification: 90C29, 58E17, 11Y16

1. Introduction

This paper focuses on the constrained multi-objective optimization problems (CMOPs), which exist widely in the real world [1–7] and are generally formulated as follows [8–10]:

$$\begin{aligned} &\text{minimize} \quad \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_m(\mathbf{x}))^T \\ &\text{subject to} \quad \begin{cases} g_i(\mathbf{x}) \leqslant 0, & i = 1, \cdots, r, \\ h_j(\mathbf{x}) = 0, & j = r+1, \cdots, q, \\ \mathbf{x} = (x_1, \ldots, x_n)^T \in \Omega \subset R^n, \end{cases} \end{aligned} \tag{1.1}$$

where $f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_m(\mathbf{x})$ are m real-valued objective functions, $g_i(\mathbf{x}) \leqslant 0$ and $h_j(\mathbf{x}) = 0$ are inequality and equality constraints, respectively, and Ω is the search space. In order to facilitate the analysis, the constraint violation value of an individual $\mathbf{x}$ is usually calculated as:

$$c(\mathbf{x}) = \sum_{i=1}^{r} \max(g_i(\mathbf{x}), 0) + \sum_{j=r+1}^{q} \max(|h_j(\mathbf{x})| - \varepsilon, 0), \tag{1.2}$$

where ε (e.g., $\varepsilon = 10^{-4}$) is a sufficiently small positive number to relax equality constraints $h_j(\mathbf{x}) = 0, j = r+1, \cdots, q$. Obviously, if and only if $c(\mathbf{x}) = 0$, the individual $\mathbf{x}$ satisfies all the constraints and is called the feasible solution. When $c(\mathbf{x}) > 0$, $\mathbf{x}$ is recorded as an infeasible solution. For any two solution $\mathbf{x}^1, \mathbf{x}^2 \in \Omega$, if $\mathbf{F}(\mathbf{x}^1) \neq \mathbf{F}(\mathbf{x}^2)$ and $\forall i = 1, .., m$, $f_i(\mathbf{x}^1) \leqslant f_i(\mathbf{x}^2)$, then it is said that $\mathbf{x}^1$ dominates $\mathbf{x}^2$, denoted as $\mathbf{x}^1 \prec \mathbf{x}^2$. A solution that is not dominated by other solutions in Ω is called the unconstrained Pareto optimal solution, and the geometry formed by all unconstrained Pareto optimal solutions in the objective space is defined as the unconstrained Pareto front (UPF). Similarly, the constrained Pareto front (CPF) is formed by all the feasible solutions that are not dominated by other feasible ones in Ω. Therefore, the essence of dealing with CMOP is to find a set of feasible solutions which are uniformly distributed on its CPF.

In the research community, a variety of constrained multi-objective evolutionary algorithms (CMEAs) have been proposed to deal with CMOPs [11–16]. Therein, many different constraint handling techniques (CHTs) have been used to balance objectives and constraints statically or dynamically. In penalty-based CHTs [17–19], the constraint violation value of an individual is transformed into the punishment for its objectives by the dynamic and adaptive penalty functions. Feasibility-led CHTs, such as constraint dominance principle (CDP) [20–24] and epsilon constraint-handling method (EC) [25–27], prefer individuals with smaller constraint violation values. Considering that the working populations guided by these two types of CHTs are easy to get stuck in the local areas and then fail to find the CPFs [24, 28], several new CHTs tend to use the method of ignoring the feasibility and only optimizing the objectives to improve their global search ability. For example, the push and pull search (PPS) [29, 30] first pushes the working populations toward UPFs without considering constraints in its push stage, and then pulls them back to CPFs by an improved EC in the pull stage. In C-TAEA [31] and CCMO [32], one population that prefers feasible solutions is used to provide search pressure to CPFs, while another population that ignores feasibility to optimize the objectives is used to force the working population to jump out of the local regions. Many numerical experiments show that these new CHTs do improve the ability of the working population to cross the complex feasible regions in front of the CPFs. Nevertheless, as overly emphasize the importance of optimizing constraints and objectives, these CHTs may mislead the population to jump over the feasible regions containing constrained Pareto optimal solutions. This would lead to missing some or even all of the fragments of CPF. So, it is still a valuable research topic to design the effective CHTs to deal with CMOPs.

To deal with the CMOPs effectively, this paper proposes a compound distance-based CHT, which aims to guide the population to explore the valuable areas that are not dominated by feasible solutions and avoid the insufficient search for some areas. Specifically, according to the angle and ℓ_p-norm, we first introduce a compound distance into the objective space to measure individual's search diameter. Then, the individuals with large search diameters in the valuable areas are given priority to be preserved. This effectively guides the working population to distribute evenly in the valuable areas

and avoid missing the feasible regions containing constrained Pareto optimal solutions. In other words, the compound distance-based CHT is effective in guiding the working population to find CPFs. Accordingly, we propose a CMEA with the compound distance-based CHT, which is denoted as CD-CMEA, for solving CMOPs. Similar to the existing CMEAs, the proposed CD-CMEA also use a feasibility-oriented archive to record the best solutions for CMOPs. Besides, to balance the local search and global search in the evolutionary process, we use a dynamic mixing strategy to select individuals from the archive and the current population evenly scattered in the valuable areas to generate offspring. A series of numerical experiments on several benchmark problems are carried out to test the performance of the proposed CD-CMEA. And the experimental results show that CD-CMEA performs better than the state-of-the-art CEMAs in dealing with different CMOPs.

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

1. We propose a novel compound distance-based CHT that guides the population to explore the valuable areas evenly. It can effectively prevent the population from being stuck in local areas.
2. We introduce a dynamic mixing strategy to guide the generation of offspring in each generation. This helps to make full use of the information of the current population and the archive to balance the local search and global search in the evolution process.
3. We carry out a series of numerical experiments to verify the effectiveness of the proposed algorithm.

The rest of this paper is organized as follows. Section 2 describes the compound distance-based CHT in detail. In Section 3, we embed the compound distance-based CHT into evolutionary algorithm for dealing with CMOPs. Section 4 organizes a series of numerical experiments to compare the performances of the proposed algorithm and the existing CMEAs. Section 5 concludes this paper.

2. Compound distance-based constraint handling technique

In this section, we describe in detail the proposed compound distance-based CHT, which aims to guide the population to explore the valuable areas that are not dominated by feasible solutions and is quite different from the existing CHTs. Let the objective vector of any individual $\mathbf{x}^q$ in Ω be $\mathbf{F}^q = (f_1(\mathbf{x}^q), ..., f_m(\mathbf{x}^q))^T$, and $f_i^{\min}$ $(i = 1, ..., m)$ be the minimum value that has been found on the i-th objective. To facilitate the analysis, we shift the objective space to R_+^m according to the following transformations:

$$f_i(\mathbf{x}^q) = f_i(\mathbf{x}^q) - f_i^{\min}, \quad i = 1, ..., m, \tag{2.1}$$

and introduce two symbols as follows:

$$\begin{cases} f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})} = \max(f_i(\mathbf{x^1}), f_i(\mathbf{x^2})), \\ f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})} = \min(f_i(\mathbf{x^1}), f_i(\mathbf{x^2})). \end{cases} \tag{2.2}$$

Besides, before introducing the proposed compound distance-based CHT, we first give a lemma, which is helpful to prove the properties of the compound distance involved.

Lemma 1. *$\forall f_i(\mathbf{x^1}), f_i(\mathbf{x^2}), f_i(\mathbf{x^3}) > 0$,*

$$1 + \frac{f_{i,\min}^{(\mathbf{x^2},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^2},\mathbf{x^3})}} \geqslant \frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})}} + \frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^3})}}. \tag{2.3}$$

Proof. (i) If $f_i(\mathbf{x^1}) \geqslant \max(f_i(\mathbf{x^2}), f_i(\mathbf{x^3})) > 0$, then the right side of inequality (2.3) can be simplified as

$$\frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})}} + \frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^3})}} \leqslant \frac{f_i(\mathbf{x^2})}{f_{i,\max}^{(\mathbf{x^2},\mathbf{x^3})}} + \frac{f_i(\mathbf{x^3})}{f_{i,\max}^{(\mathbf{x^2},\mathbf{x^3})}} = 1 + \frac{f_{i,\min}^{(\mathbf{x^2},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^2},\mathbf{x^3})}}. \tag{2.4}$$

(ii) If $0 < f_i(\mathbf{x^1}) \leqslant \min(f_i(\mathbf{x^2}), f_i(\mathbf{x^3}))$, then we can get that

$$\frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})}} + \frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^3})}} \leqslant \frac{f_{i,\min}^{(\mathbf{x^2},\mathbf{x^3})}}{f_{i,\min}^{(\mathbf{x^2},\mathbf{x^3})}} + \frac{f_{i,\min}^{(\mathbf{x^2},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^2},\mathbf{x^3})}} = 1 + \frac{f_{i,\min}^{(\mathbf{x^2},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^2},\mathbf{x^3})}}. \tag{2.5}$$

(iii) If $0 < f_{i,\min}^{(\mathbf{x^2},\mathbf{x^3})} \leqslant f_i(\mathbf{x^1}) \leqslant f_{i,\max}^{(\mathbf{x^2},\mathbf{x^3})}$, then it follows $\frac{1}{f_{i,\min}^{(\mathbf{x^2},\mathbf{x^3})}} \geqslant \frac{1}{f_i(\mathbf{x^1})} \geqslant \frac{1}{f_{i,\max}^{(\mathbf{x^2},\mathbf{x^3})}} > 0$. The right side of inequality (2.3) is calculated as

$$\frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})}} + \frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^3})}} = \begin{cases} \dfrac{f_i(\mathbf{x}^2)}{f_i(\mathbf{x}^1)} + \dfrac{f_i(\mathbf{x}^1)}{f_i(\mathbf{x}^3)}, & f_i(\mathbf{x}^2) \leq f_i(\mathbf{x}^1) \leq f_i(\mathbf{x}^3), \\ \dfrac{f_i(\mathbf{x}^1)}{f_i(\mathbf{x}^2)} + \dfrac{f_i(\mathbf{x}^3)}{f_i(\mathbf{x}^1)}, & f_i(\mathbf{x}^3) \leq f_i(\mathbf{x}^1) \leq f_i(\mathbf{x}^2), \end{cases} \tag{2.6}$$

$$= \frac{f_{i,\min}^{(\mathbf{x}^2,\mathbf{x}^3)}}{f_i(\mathbf{x}^1)} + \frac{f_i(\mathbf{x}^1)}{f_{i,\max}^{(\mathbf{x}^2,\mathbf{x}^3)}}. \tag{2.7}$$

According to the *Sequence Inequality*,

$$\frac{f_{i,\min}^{(\mathbf{x^2},\mathbf{x^3})}}{f_i(\mathbf{x^1})} + \frac{f_i(\mathbf{x^1})}{f_{i,\max}^{(\mathbf{x^2},\mathbf{x^3})}} \leqslant \frac{f_i(\mathbf{x^1})}{f_i(\mathbf{x^1})} + \frac{f_{i,\min}^{(\mathbf{x^2},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^2},\mathbf{x^3})}} = 1 + \frac{f_{i,\min}^{(\mathbf{x^2},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^2},\mathbf{x^3})}}. \tag{2.8}$$

By substituting Eq (2.8) into Eq (9), we have

$$\frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})}} + \frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^3})}} \leqslant 1 + \frac{f_{i,\min}^{(\mathbf{x^2},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^2},\mathbf{x^3})}}. \tag{2.9}$$

Thus, we have from (i), (ii) and (iii) that $\forall f_i(\mathbf{x^1}), f_i(\mathbf{x^2}), f_i(\mathbf{x^3}) > 0$,

$$1 + \frac{f_{i,\min}^{(\mathbf{x^2},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^2},\mathbf{x^3})}} \geqslant \frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})}} + \frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^3})}}. \tag{2.10}$$

□

2.1. Compound distance

When dealing with unconstrained multi-objective optimization problems, the angle is widely used as a diversity measure and has shown its effectiveness of guiding the population to evenly search for different segments of PF [33–36]. However, it can not accurately reflect the relative distance between different individuals. For example, point $A = (1, 1)$ is close to point $B = (2, 2)$ and far from point

$C = (100, 100)$, but it is easy to get that the angular distance between A, B is equal to that between A, C. This is not conducive to guiding the population to search different regions evenly when dealing with CMOPs. In order to take the advantage of angle in guiding the population to search for the different segments of CPF and overcome its deficiency in measuring the relative distance between different individuals, we introduce a new compound distance in the objective space R_+^m of CMOPs, which takes into account both the angular distance and the relative position between considered individuals and defines the distance between any two points $\mathbf{F}^1, \mathbf{F}^2 \in R_+^m$ as:

$$d_p(\mathbf{F}^1, \mathbf{F}^2) = \langle \mathbf{F}^1, \mathbf{F}^2 \rangle + k \sqrt[p]{\sum_{i=1}^{m} (1 - \frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})}})^p}, \quad p \geqslant 1, \tag{2.11}$$

where $\langle *, * \rangle$ represents the angle between the two vectors, and k is a constant greater than 0. For the three points A, B, C mentioned above, we can easily calculate that

$$d_p(A, B) = k \sqrt[p]{\sum_{i=1}^{2} (1 - \frac{1}{2})^p} < d_p(A, C) = k \sqrt[p]{\sum_{i=1}^{2} (1 - \frac{1}{100})^p}, \quad p \geqslant 1. \tag{2.12}$$

This shows that the compound distance $d_p(*, *)$ can effectively evaluate the crowding degree of individuals. As a distance function in the objective space, $d_p(*, *)$ needs to satisfy four necessary properties, which are proved as below.

Property 1 (Non-negativity)**.** $d_p(\mathbf{F}^1, \mathbf{F}^2) \geqslant 0$*, for all* $\mathbf{F}^1, \mathbf{F}^2 \in R_+^m$.

Proof. For all $\mathbf{F}^1, \mathbf{F}^2 \in R_+^m$, it is the fact that

$$\begin{cases} \langle \mathbf{F}^1, \mathbf{F}^2 \rangle \in [0, \pi/2], \\ 1 - \dfrac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})}} \geqslant 0, \quad i = 1, ..., m. \end{cases} \tag{2.13}$$

Thus, according to the definition of $d_p(*, *)$ (see Eq (2.11)), we can easily get that $d_p(\mathbf{F}^1, \mathbf{F}^2) \geqslant 0$. □

Property 2 (Identity)**.** $d_p(\mathbf{F}^1, \mathbf{F}^2) = 0$*, if and only if* $\mathbf{F}^1 = \mathbf{F}^2$.

Proof. According to the definition of $d_p(*, *)$,

$$\begin{aligned} d_p(\mathbf{F}^1, \mathbf{F}^2) = 0 &\Leftrightarrow 1 - \frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})}}, \quad i = 1, ..., m, \\ &\Leftrightarrow f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})} = f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})}, \quad i = 1, ..., m, \\ &\Leftrightarrow f_i(\mathbf{x^1}) = f_i(\mathbf{x^2}), \quad i = 1, ..., m, \\ &\Leftrightarrow \mathbf{F}^1 = \mathbf{F}^2. \end{aligned} \tag{2.14}$$

□

Property 3 (Symmetry)**.** $\forall \mathbf{F}^1, \mathbf{F}^2 \in R_+^m$, $d_p(\mathbf{F}^1, \mathbf{F}^2) = d_p(\mathbf{F}^2, \mathbf{F}^1)$.

Proof. It is obvious. □

Property 4 (Triangle inequality)**.** $\forall \mathbf{F}^1, \mathbf{F}^2, \mathbf{F}^3 \in R_+^m$, $d_p(\mathbf{F}^1, \mathbf{F}^2) + d_p(\mathbf{F}^1, \mathbf{F}^3) \geqslant d_p(\mathbf{F}^2, \mathbf{F}^3)$.

Proof. For any $\mathbf{F}^1, \mathbf{F}^2, \mathbf{F}^3 \in R_+^m$, it is obvious that

$$\langle \mathbf{F}^1, \mathbf{F}^2 \rangle + \langle \mathbf{F}^1, \mathbf{F}^3 \rangle \geqslant \langle \mathbf{F}^2, \mathbf{F}^3 \rangle. \tag{2.15}$$

Besides, according to the *Minkowski Inequality*,

$$\sqrt[p]{\sum_{i=1}^{m}(1-\frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})}})^p} + \sqrt[p]{\sum_{i=1}^{m}(1-\frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^3})}})^p} \geqslant \sqrt[p]{\sum_{i=1}^{m}(2-\frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})}}-\frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^3})}})^p}. \tag{2.16}$$

By using Lemma 1, we can further transform Eq (2.16) as follows

$$\sqrt[p]{\sum_{i=1}^{m}(1-\frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})}})^p} + \sqrt[p]{\sum_{i=1}^{m}(1-\frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^3})}})^p} \geqslant \sqrt[p]{\sum_{i=1}^{m}(1-\frac{f_{i,\min}^{(\mathbf{x^2},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^2},\mathbf{x^3})}})^p}. \tag{2.17}$$

So, it can inferred from Eq (2.15) and Eq (2.17) that

$$\begin{aligned} d_p(\mathbf{F}^1, \mathbf{F}^2) + d_p(\mathbf{F}^1, \mathbf{F}^3) &= \langle \mathbf{F}^1, \mathbf{F}^2 \rangle + k\sqrt[p]{\sum_{i=1}^{m}(1-\frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})}})^p} + \langle \mathbf{F}^1, \mathbf{F}^3 \rangle + k\sqrt[p]{\sum_{i=1}^{m}(1-\frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^3})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^3})}})^p} \\ &\geqslant \langle \mathbf{F}^1, \mathbf{F}^2 \rangle + k\sqrt[p]{\sum_{i=1}^{m}(1-\frac{f_{i,\min}^{(\mathbf{x^1},\mathbf{x^2})}}{f_{i,\max}^{(\mathbf{x^1},\mathbf{x^2})}})^p} \\ &= d_p(\mathbf{F}^2, \mathbf{F}^3). \end{aligned} \tag{2.18}$$

□

2.2. Compound distance-based CHT

Using the compound distance $d_p(*,*)$, we define the search diameter of a point $\mathbf{F}^i$ in $\mathbb{F} \subset R_+^m$ as:

$$R_p(\mathbf{F}^i|\mathbb{F}) = \min_{\mathbf{F}^j \in \mathbb{F}\setminus\{\mathbf{F}^i\}} d_p(\mathbf{F}^i, \mathbf{F}^j). \tag{2.19}$$

Since Properties 1 to 4 have shown that $d_p(*,*)$ is a measure in the space R_+^m, the larger the value of $d_p(\mathbf{F}^i, \mathbf{F}^j)$ obviously means that the distance between two individuals $\mathbf{F}^i, \mathbf{F}^j$ is larger. So, a point $\mathbf{F}^i$ with a large search diameter $R_p(\mathbf{F}^i|\mathbb{F})$ means that it has a great contribution for $\mathbb{F}$ to searching different areas. Consider the point set $\mathbb{F} = \{A, B, C\} \subset R_+^2$ mentioned in Subsection 2.1 as an example. It is easy to get that $R_p(A|\mathbb{F}) = d_p(A, B)$, $R_p(B|\mathbb{F}) = d_p(B, A)$, $R_p(C|\mathbb{F}) = d_p(C, B)$. As $d_p(A, B) = d_p(B, A) = 2\sqrt[p]{2} < d_p(C, B) = 50\sqrt[p]{2}$, it follows that $R_p(A|\mathbb{F}) = R_p(B|\mathbb{F}) < R_p(C|\mathbb{F})$, which is consistent with the distribution of points A, B, C in space R_+^2. Therefore, in the process of reducing the size of the considered population, the proposed compound distance-based CHT preferentially removes the individuals with the smallest search diameters and retains the individuals with larger search diameters.

Suppose that the size of the considered population is N^*, which exceeds its limited size N. Then the pseudo codes of pruning it using the proposed compound distance-based CHT is shown in Algorithm 1. Specifically, it first calculates the search diameter of each solution by the use of Eq (2.19) and determines the number of deleted solutions according to $DN = N^* - N$. After that, the DN worst solutions with the smallest search diameters are removed from the considered population one by one (Lines 3 to 7 in Algorithm 1). Considering that the removal of the worst individual may affect the search diameters of some remaining individuals and then results in the inaccurate of determining the next worst solutions, we update the search diameters of all remaining individuals in Line 7. Finally, the N retained individuals in the considered population are output as the results.

Algorithm 1: Pruning steps by using the proposed compound distance-based CHT

Input: The considered population $\mathbf{X}$ of size N^* and the limited size N;
Output: The refined population $\mathbf{X}$ with size N;
1 Calculate the search diameter of each solution $\mathbf{x}$ in $\mathbf{X}$ by Eq (2.19);
2 Determine the number of solutions that need to be deleted: $DN = N^* - N$;
3 **for** $i = 1 : DN$ **do**
4 Determine the worst solution $\mathbf{x}' \in \mathbf{X}$ with the smallest search diameter: $\mathbf{x}' = \arg\min_{\mathbf{x}^j \in \mathbf{X}} R_p(\mathbf{F}^j|\mathbb{F})$;
5 Remove $\mathbf{x}'$ from $\mathbf{X}$: $\mathbf{X} = \mathbf{X} \setminus \{\mathbf{x}'\}$;
6 Update the search diameters of the remaining individuals in $\mathbf{X}$ by Eq (2.19);
7 **end**
8 **return** Population $\mathbf{X}$.

It is easy to get that in the process of determining the best individuals in the considered population, the proposed compound distance-based CHT do not take into account their performance in optimizing constraints and objectives. This is the main difference between the proposed compound distance-based CHT and the existing CHTs. In this way, the proposed compound distance-based CHT can avoid being lured to the local areas by deceptive constraints and guide the working population search for CPFs more effectively.

3. Proposed algorithm framework

In this section, we embed the compound distance-based CHT into evolutionary algorithm and propose a new CMEA with the compound distance-based CHT (denoted as CD-CMEA) to solve the CMOPs. The details of the proposed CD-CMEA are described below.

3.1. Update mechanism of the current population

Due to the emergence of new individuals, the size N^* of the working population $\mathbf{X}_t$ in the t-generation will exceed the preset size N. In order to effectively guide the $\mathbf{X}_t$ to search for CPFs, we use the compound distance-based CHT to select the N individuals of $\mathbf{X}_t$ who perform best in uniformly searching valuable areas to form the current population $\mathbf{X}_{t+1}$ of the next generation.

Specifically, we first identify the set of individuals located in the valuable areas as:

$$VX_t = \{\mathbf{x} \in \mathbf{X}_t | \nexists \mathbf{x}' \in \mathbf{X}_t, c(\mathbf{x}') = 0, \mathbf{x}' \prec \mathbf{x}\}. \tag{3.1}$$

If the size of VX_t is small than N, i.e., $|VX_t| < N$, we initialize the current population of the next generation $\mathbf{X}_{t+1}$ as VX_t and then select $(N - |VX_t|)$ individuals closest to the valuable areas from the remaining individuals to fill $\mathbf{X}_{t+1}$, where the distance from an individual to the valuable areas determined by VX_t is calculated as:

$$dis(\mathbf{x}, VX_t) = \min_{\mathbf{x}' \in VX_t} \min_{i \in \{1,\ldots,m\}} (f_i(\mathbf{x}) - f_i(\mathbf{x}')). \tag{3.2}$$

Otherwise, we use Algorithm 1 to select the best N individuals of VX_t who perform best in uniformly searching valuable areas to form $\mathbf{X}_{t+1}$. We simplify the above steps to the pseudo-code form in Algorithm 2.

Algorithm 2: Update mechanism of the current population

Input: The current working population $\mathbf{X}_t$ and preset size N;
Output: The population $\mathbf{X}_{t+1}$ for the next generation;
Identify the set VX_t of valuable individuals by Eq (3.5);
if $|VX_t| < N$ **then**
    Let $\mathbf{X}_{t+1} = VX_t$ and $\overline{\mathbf{X}}_t = \mathbf{X}_t \setminus VX_t$;
    Calculate the distance $dis(\mathbf{x}, VX_t)$ of each individual in $\overline{\mathbf{X}}_t$ by Eq (3.2);
    Select $(N - |VX_t|)$ individuals with the smallest values of $dis(\mathbf{x}, VX_t)$ from $\overline{\mathbf{X}}_t$ to fill $\mathbf{X}_{t+1}$;
else
    Use the Algorithm 1 to select N individuals of VX_t to form $\mathbf{X}_{t+1}$;
end
return $\mathbf{X}_{t+1}$.

3.2. Archive

In the proposed CD-CMEA, the best solutions for solving CMOPs are recorded in an archive, which prefers feasible solutions. The pseudo codes of forming the archive are shown in Algorithm 3. First, the feasible solutions of the working population are used to initialize the archive. Then, if the number of individuals in archive is smaller than the preset size N, the infeasible solutions with the smallest constraint violation values are selected to fill the archive; otherwise, we use environmental selection of PREA [37, 38], which has showed its excellent performance in dealing with different unconstrained MOPs and won the championship in IEEE WCCI 2018 Competition of Many-Objective Optimization, to prune the archive. The specific steps of environmental selection of PREA is as follows:

1. Shift the archive Λ to R_m^+ as:

$$f_i'(\mathbf{x}) = f_i(\mathbf{x}) - f_i^{Fmin} + 10^{-6}, \quad i = 1, ..., m, \tag{3.3}$$

where f_i^{Fmin} is the minimum value found by feasible solutions on the objective f_i.

2. Calculate the fitness value of each individual $\mathbf{x}^i \in \Lambda$ by the following equation:

$$Fitness(\mathbf{x}^i|\Lambda) = \min_{\mathbf{x}^j \in \Lambda \setminus \mathbf{x}^i} \begin{cases} \max\limits_{l \in \{1,...,m\}} \dfrac{f_l(\mathbf{x}^j)}{f_l(\mathbf{x}^i)}, & \text{if } \exists f_l(\mathbf{x}^j) > f_l(\mathbf{x}^i); \\ \min\limits_{l \in \{1,...,m\}} \dfrac{f_l(\mathbf{x}^j)}{f_l(\mathbf{x}^i)}, & \text{otherwise.} \end{cases} \tag{3.4}$$

3. Determine the nondominated solutions in Λ:

$$\Lambda' = \{\mathbf{x}|Fitness(\mathbf{x}|\Lambda) \geqslant 0, \mathbf{x} \in \Lambda\}. \tag{3.5}$$

If $|\Lambda'| \leqslant N$, terminate the calculation and let Λ retain only the N individuals with the best value of $Fitness(\mathbf{x}|\Lambda)$. Otherwise, let $\Lambda = \Lambda'$ and mark N solutions with the best fitness values:

$$\Lambda_B = \{\mathbf{x}_{q_1}, ..., \mathbf{x}_{q_N}\} \subset \Lambda. \tag{3.6}$$

These solutions are then used to form a promising region and to identify valuable candidates Λ_C as follows:

$$\begin{cases} \Lambda_C = \{\mathbf{x}|\mathbf{F}'(\mathbf{x}) \succ (f_1^{max*}, ..., f_m^{max*}), \mathbf{x} \in \Lambda\}, \\ \mathbf{F}'(\mathbf{x}) = (f_1'(\mathbf{x}), ..., f_m'(\mathbf{x})), f_i^{max*} = \max\limits_{\mathbf{x} \in \Lambda_B} f_i'(\mathbf{x}). \end{cases} \tag{3.7}$$

After that, the individuals with the minimum crowding distance in Λ are eliminated one by one until the size of Λ is reduced to N. In each iteration, the one with the smaller fitness value of the two nearest individuals is removed from Λ. Therein, a parallel distance is used to measure the distance between any two solutions in Λ and is calculated as follows:

$$d_{ij} = \{\sum_{l=1}^{m} (\widehat{f_l^i} - \widehat{f_l^j})^2 - [\sum_{l=1}^{m} (\widehat{f_l^i} - \widehat{f_l^j})]^2/m\}^{0.5}, \tag{3.8}$$

where $\widehat{f_l^i} = f_l'(\mathbf{x}_i)/f_l^{max*}$.

Algorithm 3: Process of forming archive

Input: The current working population $\mathbf{X}_t$ and preset size N;
Output: Archive Λ;

```
Initialize the archive Λ as all the feasible solutions in X_t;
if |Λ| < N then
    Select (N − |Λ|) individuals with the smallest constraint violation values from the
      remaining infeasible solutions to fill Λ;
else
    Reduce the size of Λ to N by the environmental selection of PREA [37];
end
return Archive Λ.
```

3.3. Offspring production

To balance the local search and global search in the evolutionary process, we introduce a dynamic mixing strategy to randomly select individuals from the current population and the archive to form the mating pool of size N. Therein, the number of individuals selected from the current population of the t-th generation is calculated as:

$$N_t = \lfloor \frac{N(T-t)}{T} \rfloor, \tag{3.9}$$

where T is the maximum number of evolutionary generation, and function $\lfloor * \rfloor$ returns the largest integer not exceed the argument. The individuals of mating pool selected from the archive is $(N - N_t)$. According to Eq (3.9), it is easy to get that with the deepening of the optimization process, that is, the increase of t value, the number N_t of individuals from the current population in mating pool gradually decreased from the initial N to 0. This helps the CMEA to make full use of the information of individuals evenly scattered in the valuable areas for global search in the early evolutionary stage, and to focus on local search using the archive in the late evolutionary stage. For each individual $\mathbf{x}$ in the mating pool, its two spouses have a 0.7 probability of being the individuals with the smallest angles with $\mathbf{x}$ in the objective space, and a 0.3 probability of being randomly selected from the mating pool. After that, we implement the differential evolution operator (DE) [39] and polynomial mutation (PM) [40] for these matched individuals to produce offspring. The specific steps for generating offspring are shown in Algorithm 4.

Algorithm 4: Offspring production

Input : The current working population $\mathbf{X}_t$, archive Λ, and the preset size N;
Output: Offspring $\mathbf{Y}$;

Let $\mathbf{Y} = \emptyset$;
Use Eq (3.9) to calculate the value of parameter N_t;
Randomly select N_t and $(N - N_t)$ individuals from $\mathbf{X}_t$ and Λ to form the mating pool MP;
for *each* $\mathbf{x}^i \in MP$ **do**
    if *rand* < 0.7 **then**
        The two individuals $\mathbf{x}', \mathbf{x}''$ with the smallest angle to $\mathbf{x}^i$ are regarded as the spouses of $\mathbf{x}^i$;
    else
        Randomly select two individuals $\mathbf{x}', \mathbf{x}''$ from MP and regard them as the spouses of $\mathbf{x}^i$;
    end
    Implement the genetic operators, including DE and PM, on $\mathbf{x}^i, \mathbf{x}', \mathbf{x}''$ to produce a new individual $\mathbf{y}^i$;
    Let $\mathbf{Y} = \mathbf{Y} \cup \{\mathbf{y}^i\}$;
end
return Offspring $\mathbf{Y}$.

3.4. Framework of the proposed CD-CMEA

The flow chart of the proposed CD-CMEA is given in Figure 1, and its pseudo-code is shown in Algorithm 5. Specifically, N individuals are randomly generated to initialize the current population $\mathbf{X}_0$ and the archive Λ. After that, the following steps are repeated many times until the stopping criteria are met, which include using Algorithm 4 to produce offspring $\mathbf{Y}$, updating the archive Λ by Algorithm 3, and using Algorithm 2 to select the best N solutions from $\mathbf{X}_t$, Λ and $\mathbf{Y}$ to form the next generation population $\mathbf{X}_{t+1}$. Finally, archive Λ is output as the result. For simplicity, in the proposed CD-CMEA, we set the parameters k and p of the compound distance (see Eq (2.11)) as $\frac{\pi}{2\sqrt[k]{m}}$ and 1, respectively. According to the framework of the proposed CD-CMEA, it can be easily found that the complexity of CD-CMEA mainly lies in two the processes: 1) calculating the search diameter of each individual by Eq (2.19); and 2) updating the archive set by the environmental selection of PREA. As the complexity of calculating the search diameter of an individual is $O(mN \log N)$, the complexity of calculating all individuals in the current population is $O(mN^2 \log N)$. Also, the number of objectives m is generally smaller than the population size N and the complexity of PREA is $O(N^3)$ [37]. Therefore, the complexity of the proposed CD-CMEA is $O(N^3)$.

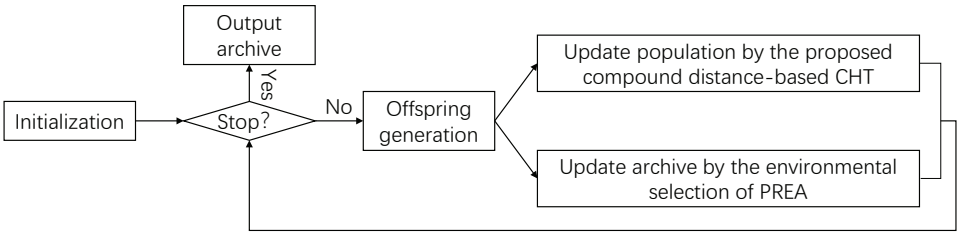

Figure 1. The flow chart of the proposed CD-CMEA.

Algorithm 5: Framework of CD-CMEA

Input: The preset size N and stopping criteria;
Output: The archive Λ;

Randomly generate the initial population $\mathbf{X}_0$ of size N and let $\Lambda = \mathbf{X}_0$, $t = 0$;
while *the stopping criteria are not met* **do**
  Use Algorithm 4 to produce offspring $\mathbf{Y}$;
  Merge $\mathbf{X}_t$, Λ and $\mathbf{Y}$ to form a big working population $\mathbf{Z}_t$;
  Select N solutions from $\mathbf{Z}_t$ to update Λ by Algorithm 3;
  Determine the current population $\mathbf{X}_{t+1}$ of size N for the next generation by Algorithm 2;
  Let $t = t + 1$;
end
return The archive Λ.

4. Numerical experiments and analysis

In this section, we have carried out a series of numerical experiments to investigate the performance of the proposed algorithm CD-CMEA, which is compared with that of other state-of-the-art CMEAs.

4.1. Experimental design

In our simulation experiments, several benchmark suites, including CTP1-8 [41], DASCMOP1-9 [42], MW1-14 [43], C-DTLZ-series [20] and DC1-3-DTLZ1 [31], are regarded as the test problems. To vividly show the performance difference between the proposed CD-CMEA and the existing CMEAs, six state-of-the-art algorithms, that is, CMOEA/D-DE-CDP [22], CMOEA/D-DE-SR [22], C-TAEA [31], PPS [29], ToP [44] and AnD [23], are selected for comparison. The core ideas of these comparison algorithms are given below.

1. *CMOEA/D-DE-CDP [22]* is a decomposition-based differential evolution algorithm. It uses the CDP to determine the best solutions in the current population. Specifically, in the process of evaluating the quality of candidate solutions, the solutions with smaller constraint violation values are considered to be better. And when the constraint violation values of the candidate solutions are all the same, the solutions with the best fitness values under the evaluation of Tchebycheff method are regarded as the optimal solutions.
2. *CMOEA/D-DE-SR [22]* is an improved version of CMOEA/D-DE-CDP and uses a stochastic ranking approach (SR) to balance the constraints and objectives. Specifically, when the candidate solutions are both infeasible and the randomly generated number is smaller than the preset small probability value p_f, it ignores the constraint and determines the optimal solutions according to the performances of these candidate individuals on optimizing objectives; Otherwise, it also uses the CDP to selects the best solutions.
3. *C-TAEA [31]* is a two-archive CMEA that simultaneously maintains two collaborative archives, i.e., the convergence oriented archive CA and diversity oriented archive DA. Therein, CA prefers feasible individuals and provides pressure to push the population to search for CPF, while DA evaluated by only optimizing the objectives focuses on exploring the areas that have not been exploited by CA.
4. *PPS [29]* is such a CMEA that divides the search process into two different stages: the push stage and the pull stage. In the push stage, constraints are ignored, and the population is evolved by optimizing only the objectives. In the pull stage, it uses an improved ϵ-CDP to pull the population to search for the CPF.
5. *ToP [44]* is a two-phase based CMEA for CMOPs. In the first phase, it transforms the original CMOP into a constrained single-objective optimization problem, which is then solved by a constrained single-objective evolutionary algorithm. In the second phase, it focuses on the promising area discovered in the first phase and uses the NSGA-II with CDP to find the CPF.
6. *AnD [23]* is a CMEA that uses angle-based selection strategy, shift-based density estimation strategy, and CDP. In the process of comparing individual quality, those with smaller violations of constraints will be selected for the next generation.

Considering that the released version of these compared algorithms may be their best form, we set their private parameters to be consistent with those in their published papers. For the sake of fairness, in terms of public parameters, the population size and the maximum function evaluation of all algorithms on each test problem are set to 100 and 2×10^5, respectively. And the number of independent runs of the algorithms on each test problem is set to 20.

Two widely used metrics, i.e., inverted generational distance (IGD) [45] and hypervolume (HV) [46], are adopted in this paper to evaluate the performance of the seven considered algorithms on each test problem. These two metrics are defined as follows:

IGD-metric: Let $\mathbf{P}$ be a set of uniformly distributed points on the CPF, and $\mathbf{S}$ is the final feasible results of an algorithm in the objective space, then the IGD of $\mathbf{S}$ to $\mathbf{P}$ is calculated as:

$$IGD(\mathbf{S}|\mathbf{P}) = \frac{\sum\limits_{\mathbf{p}\in\mathbf{P}} \min\limits_{\mathbf{s}\in\mathbf{S}} \|\mathbf{s}-\mathbf{p}\|_2}{|\mathbf{P}|}. \tag{4.1}$$

Obviously, the smaller the $IGD(\mathbf{S}|\mathbf{P})$ is, the better it is to approximate the CPF with $\mathbf{S}$. For each test instance in this paper, we use the method of Das and Dennis [47] to generate the uniformly distributed point set $\mathbf{P}$ with the size of 10,000.

HV-metric: Let $\mathbf{z}^* = (z_1, ..., z_m)^T = 1.1 \times (f_1^{\max}, ..., f_m^{\max})^T$ be a reference point determined by the maximum value of each objective, i.e., $f_i^{\max}, i = 1, ..., m$, in the CPF. Then the HV value of the feasible solution set $\mathbf{S}$ obtained by a CMEA is calculated as:

$$HV(\mathbf{S}|\mathbf{z}) = VOL(\bigcup_{\mathbf{s}\in\mathbf{S}}[s_1, z_1] \times ... \times [s_m, z_m]), \tag{4.2}$$

where $VOL(*)$ is the Lebesgue measure. It is easy to get that a large value of $HV(\mathbf{S}|\mathbf{z})$ means good quality of $\mathbf{S}$ in the objective space.

To hear the voice from statistics, we use the Wilcoxon Rank Sum Test with a significance level of 0.05 to analyze the performance differences between the proposed CD-CMEA and the other six existing CMEAs on each test problem. Therein, three symbols "+", "=" and "–" are introduced, which represent that the final results obtained by the compared algorithms are significantly better than, equal to, and worse than the proposed CD-CMEA, respectively.

4.2. Experimental results and analysis

The comparison results among the seven considering CMEAs on each test instance under IGD and HV metrics are listed in Tables 1 and 2. In these tables, we highlight the best algorithm for each test problem with a gray background and a bold way. Besides, in order to visually show the excellent performance of the proposed CD-CMEA, we take DASCMOP series as the examples to draw the feasible solutions obtained by these seven comparing algorithms in Figures 2 and 3. As can be seen from these Tables 1 and 2, compared with the other six CMEAs, the proposed CD-CMEA performs better in the test instances, and it has better robustness in dealing with different CMOPs. This is to be expected. Next, we will analyze the reasons for this in detail.

Table 1. Comparison results of the seven CMEAs on the test instances under IGD metric.

Instances	m	CD-CMEA	CMOEA/D-DE		C-TAEA	PPS	ToP	AnD
			CDP	SR				
CTP1	2	**0.0043**	0.0057(–)	0.0078(–)	0.0051(–)	0.0059(–)	0.0052(–)	0.0544(–)
CTP2	2	**0.0015**	0.0049(–)	0.0070(–)	0.0083(–)	0.0028(–)	0.0026(–)	0.0214(–)
CTP3	2	0.0056	0.0059(–)	0.0261(–)	0.0116(–)	0.0071(–)	**0.0043(=)**	0.0601(–)
CTP4	2	0.0614	0.0748(–)	0.1758(–)	0.0687(–)	0.0790(–)	**0.0475(+)**	0.1379(–)
CTP5	2	**0.0013**	0.0037(–)	0.0139(–)	0.0108(–)	0.0025(=)	0.0017(+)	0.0117(–)
CTP6	2	**0.0045**	0.9677(–)	0.9718(–)	0.0066(–)	0.0051(–)	0.3577(–)	0.0099(–)
CTP7	2	**0.0101**	0.0115(–)	0.0113(–)	0.0119(–)	0.0103(–)	0.0102(–)	0.0360(–)
CTP8	2	**0.0022**	1.8624(–)	1.2681(–)	0.0083(–)	0.0030(–)	0.8375(–)	0.0787(–)
DASCMOP1	2	**0.0032**	0.6681(–)	0.5085(–)	0.2684(–)	0.0052(=)	0.7762(–)	0.7180(–)
DASCMOP2	2	**0.0030**	0.2313(–)	0.2209(–)	0.1111(–)	0.0039(+)	0.7240(–)	0.2670(–)
DASCMOP3	2	**0.0194**	0.3364(–)	0.3449(=)	0.1587(+)	0.0195(+)	0.6844(–)	0.3546(–)
DASCMOP4	2	**0.0014**	0.2312(–)	0.2573(–)	0.0117(–)	0.0783(–)	NaN(–)	0.0387(–)
DASCMOP5	2	**0.0028**	0.0618(–)	0.0845(–)	0.0075(–)	0.0072(–)	NaN(–)	0.1507(–)
DASCMOP6	2	0.0268	0.1323(–)	0.1929(–)	**0.0258(–)**	0.1191(–)	NaN(–)	0.1201(–)
DASCMOP7	3	**0.0327**	0.1580(–)	0.1286(–)	0.0401(–)	0.1840(–)	NaN(–)	0.0392(–)
DASCMOP8	3	**0.0429**	0.1652(–)	0.1746(–)	0.0570(–)	0.1192(–)	NaN(–)	0.0511(–)
DASCMOP9	3	**0.0409**	0.3528(–)	0.3832(–)	0.2440(–)	0.0977(–)	0.5982(–)	0.2646(–)
MW1	2	**0.0021**	0.0039(–)	0.0037(–)	0.0022(–)	0.0029(–)	NaN(–)	0.0041(–)
MW2	2	**0.0054**	0.1131(–)	0.1551(–)	0.0131(–)	0.1567(–)	0.1048(–)	0.0241(–)
MW3	2	0.0054	**0.0043(+)**	0.1011(–)	0.0049(+)	0.0062(–)	0.6030(–)	0.0092(–)
MW4	3	0.0466	0.0609(–)	0.0616(–)	0.0467(–)	0.0574(–)	NaN(–)	**0.0447(=)**
MW5	2	**0.0016**	0.5217(–)	0.5961(–)	0.0088(–)	0.2308(–)	NaN(–)	0.1764(–)
MW6	2	**0.0029**	0.5138(–)	0.5580(–)	0.0078(–)	0.4263(–)	0.4271(–)	0.0407(–)
MW7	2	**0.0037**	0.0043(+)	0.0140(–)	0.0055(–)	0.0049(=)	0.0628(–)	0.0270(–)
MW8	3	**0.0495**	0.1887(–)	0.2072(–)	0.0544(–)	0.1646(–)	0.4963(–)	0.0519(–)
MW9	2	0.0089	0.3235(–)	0.4699(–)	0.0103(–)	0.6060(–)	0.0222(–)	**0.0078(=)**
MW10	2	**0.0041**	NaN(–)	NaN(–)	0.0149(–)	NaN(–)	NaN(–)	0.1618(–)
MW11	2	**0.0039**	0.0059(–)	0.0096(–)	0.0075(–)	0.0043(–)	0.2690(–)	0.3632(–)
MW12	2	0.0044	**0.0043(=)**	0.0168(–)	0.0067(–)	0.0064(–)	NaN(–)	0.0768(–)
MW13	2	**0.0044**	0.1973(–)	0.2892(–)	0.0135(–)	0.2190(–)	0.1783(–)	0.0465(–)
MW14	3	**0.0419**	0.1243(–)	0.1236(–)	0.0466(–)	0.0542(–)	0.1504(–)	0.0471(–)
C1-DTLZ1	3	0.0437	0.0609(–)	0.0551(–)	0.0467(–)	0.0517(–)	NaN(–)	**0.0436(=)**
C2-DTLZ2	3	**0.0450**	0.0703(–)	0.0739(–)	0.0562(–)	0.0555(–)	0.0580(–)	0.0462(=)
C3-DTLZ4	3	**0.0488**	0.0729(–)	0.1101(–)	0.0558(–)	0.0719(–)	0.0712(–)	0.0533(–)
DC1-DTLZ1	3	0.0321	0.2340(–)	0.1370(–)	**0.0248(+)**	0.0338(–)	0.0908(–)	0.0371(–)
DC2-DTLZ1	3	**0.0439**	0.0614(–)	0.0615(–)	0.0467(–)	0.0569(–)	NaN(–)	NaN(–)
DC3-DTLZ1	3	**0.0204**	0.8495(–)	1.4803(–)	0.0264(–)	0.1406(–)	3.7102(–)	0.0626(–)
Summary(+/=/–)		——	2/1/34	0/1/36	3/0/34	2/3/32	2/1/34	0/4/33

Table 2. Comparison results of the seven CMEAs on the test instances under HV metric.

Instances	m	CD-CMEA	CMOEA/D-DE		C-TAEA	PPS	ToP	AnD
			CDP	SR				
CTP1	2	**0.3793**	0.3790(–)	0.3779(–)	0.3789(–)	0.3782(–)	0.3785(–)	0.3631(–)
CTP2	2	0.4289	0.4284(–)	0.4237(–)	0.4233(–)	**0.4290(=)**	**0.4290(=)**	0.4142(–)
CTP3	2	0.4074	0.4082(=)	0.3892(–)	0.4017(–)	0.4058(–)	**0.4105(+)**	0.3762(–)
CTP4	2	0.3595	0.3495(–)	0.2536(–)	0.3548(–)	0.3462(–)	**0.3720(+)**	0.3058(–)
CTP5	2	0.4106	0.4117(+)	0.3919(–)	0.4041(–)	0.4096(+)	**0.4122(+)**	0.3746(–)
CTP6	2	**0.4657**	0.2081(–)	0.2046(–)	0.4631(–)	0.4648(–)	0.3688(–)	0.4594(–)
CTP7	2	0.5474	0.5474(=)	0.5469(–)	0.5446(–)	**0.5481(+)**	**0.5481(+)**	0.5152(–)
CTP8	2	**0.3703**	0.0367(–)	0.1368(–)	0.3632(–)	0.3700(=)	0.2184(–)	0.3431(–)
DASCMOP1	2	**0.2032**	0.0165(–)	0.0632(–)	0.1606(–)	0.1961(–)	0.0020(–)	0.0071(–)
DASCMOP2	2	**0.3510**	0.2547(–)	0.2563(–)	0.2954(–)	0.3506(+)	0.0456(–)	0.2419(–)
DASCMOP3	2	**0.3125**	0.2131(–)	0.2072(–)	0.2528(+)	0.3121(+)	0.0488(–)	0.2085(–)
DASCMOP4	2	**0.2039**	0.1539(–)	0.1493(–)	0.1968(–)	0.1844(–)	NaN(–)	0.1878(–)
DASCMOP5	2	**0.3512**	0.3211(=)	0.3147(–)	0.3480(–)	0.3491(=)	NaN(–)	0.2656(–)
DASCMOP6	2	0.3078	0.2653(–)	0.2427(–)	**0.3079(=)**	0.2731(–)	NaN(–)	0.2583(–)
DASCMOP7	3	**0.2879**	0.2402(–)	0.2493(–)	0.2871(+)	0.2238(–)	NaN(–)	0.2860(+)
DASCMOP8	3	**0.2076**	0.1733(–)	0.1643(–)	0.2033(–)	0.1773(–)	NaN(–)	0.2054(=)
DASCMOP9	3	**0.2078**	0.1324(–)	0.1191(–)	0.1459(–)	0.1905(–)	0.0765(–)	0.1437(–)
MW1	2	0.4875	0.4867(–)	0.4867(–)	**0.4889(=)**	0.4884(=)	NaN(–)	0.4871(–)
MW2	2	**0.5802**	0.4291(–)	0.3854(–)	0.5662(–)	0.3884(–)	0.4430(–)	0.5472(–)
MW3	2	0.5442	**0.5455(+)**	0.4883(–)	0.5450(+)	0.5436(–)	0.1787(–)	0.5386(–)
MW4	3	0.8359	0.8001(–)	0.8005(–)	**0.8382(+)**	0.8114(–)	0.0222(–)	0.8368(=)
MW5	2	**0.3235**	0.1575(–)	0.1352(–)	0.3189(–)	0.2483(–)	NaN(–)	0.2620(–)
MW6	2	**0.3269**	0.0841(–)	0.0771(–)	0.3171(–)	0.1261(–)	0.1326(–)	0.2923(–)
MW7	2	**0.4123**	0.4116(+)	0.4079(–)	0.4095(–)	0.4121(+)	0.3731(–)	0.4007(–)
MW8	3	**0.5328**	0.2570(–)	0.3010(–)	0.5208(–)	0.2966(–)	0.2250(–)	0.5251(–)
MW9	2	0.3908	0.1719(–)	0.1820(–)	**0.3915(–)**	0.1385(–)	0.3777(–)	0.3913(=)
MW10	2	**0.4533**	NaN(–)	NaN(–)	0.4398(–)	NaN(–)	NaN(–)	0.3452(–)
MW11	2	0.4471	0.4470(=)	0.4414(–)	0.4438(–)	**0.4476(=)**	0.3233(–)	0.2912(–)
MW12	2	**0.6047**	0.6043(=)	0.5866(–)	0.6008(–)	0.6019(–)	NaN(–)	0.5419(–)
MW13	2	**0.4749**	0.2753(–)	0.2375(–)	0.4602(–)	0.2565(–)	0.3033(–)	0.4236(–)
MW14	3	0.4584	0.3962(–)	0.3984(–)	0.4671(=)	0.4539(–)	0.3841(–)	**0.4683(+)**
C1-DTLZ1	3	**0.8385**	0.7999(–)	0.8082(–)	0.8376(–)	0.8195(–)	NaN(–)	0.8367(–)
C2-DTLZ2	3	0.5082	0.4978(–)	0.4785(–)	0.5070(–)	0.5017(–)	0.4730(–)	**0.5155(+)**
C3-DTLZ4	3	**0.7931**	0.7807(–)	0.7496(–)	0.7855(–)	0.7643(–)	0.7570(–)	0.7852(–)
DC1-DTLZ1	3	0.6202	0.5135(–)	0.5753(–)	**0.6492(+)**	0.6328(+)	0.5885(–)	0.6337(+)
DC2-DTLZ1	3	**0.8384**	0.8024(–)	0.8023(–)	0.8381(–)	0.8030(–)	NaN(–)	NaN(–)
DC3-DTLZ1	3	0.5155	0.2015(–)	0.1775(–)	**0.6584(+)**	0.5705(+)	0.0643(–)	0.6123(+)
Summary (+/=/–)		——	3/5/29	0/0/37	6/3/28	7/5/25	4/1/32	5/3/29

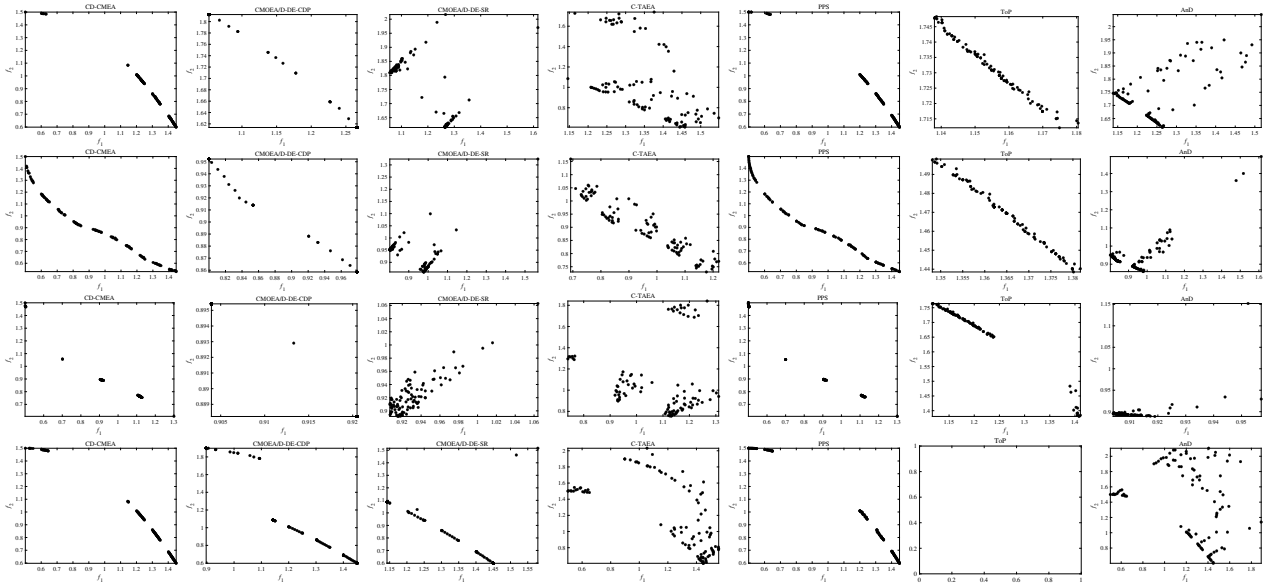

Figure 2. The feasible solutions obtained by CD-CMEA (1st column), CMOEA/D-DE-CDP (2nd column), CMOEA/D-DE-SR (3rd column), C-TAEA (4th column), PPS (5th column), ToP (6th column) and AnD (7th column) in dealing with DASCMOP1-4. Rows 1–4 refer to problems DASCMOP1–4, respectively.

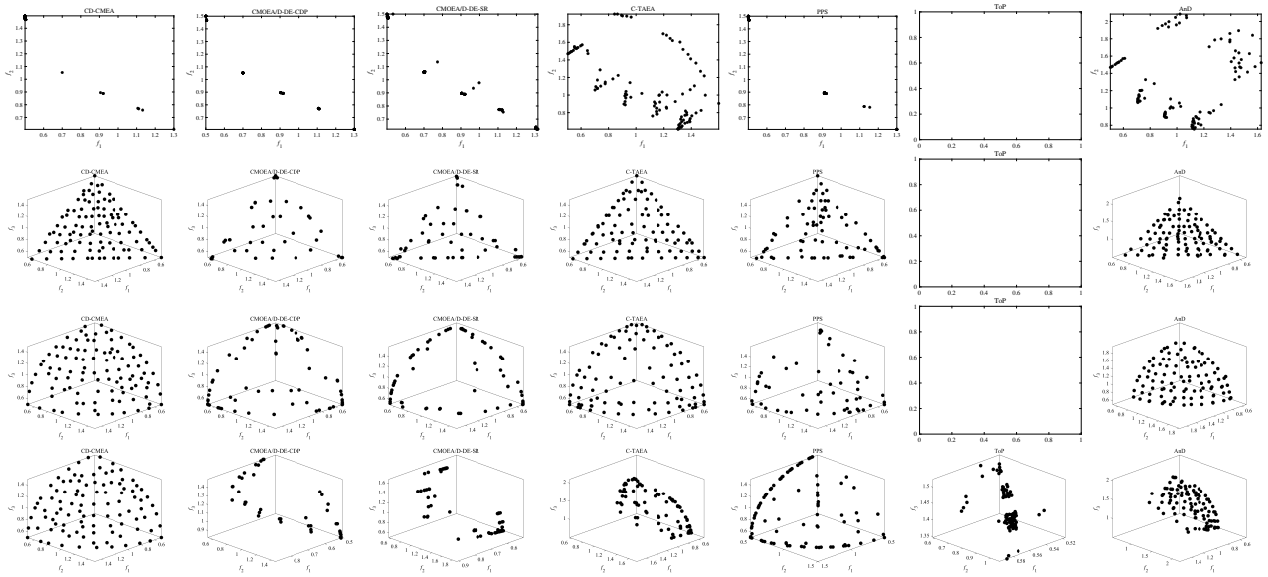

Figure 3. The feasible solutions obtained by CD-CMEA (1st column), CMOEA/D-DE-CDP (2nd column), CMOEA/D-DE-SR (3rd column), C-TAEA (4th column), PPS (5th column), ToP (6th column) and AnD (7th column) in dealing with DASCMOP5–9. Rows 1–5 refer to problems DASCMOP5–9, respectively.

When dealing with CTP1-5,7, C2-DTLZ2, C3-DTLZ4 and DC1-DTLZ1, we can see that there is little difference in the performance of the seven algorithms. The main reason for this is that the constraints of these instances are very simple and they actually do not pose any challenge for the seven algorithms to find the CPFs. So, the proposed CD-CMEA and the other six algorithms can easily find the CPFs of these test instances, and the slight difference in performance among them is caused by their diversity maintenance mechanism. Since these problems have irregular PFs, which are captured more accurately by the PREA adopted in CD-CMEA than by the reference point-based and angle-based diversity maintenance mechanism of other algorithms [37], it is not surprising that the proposed CD-CMEA outperforms CMOEA/D-DE-CDP, CMOEA/D-DE-SR, C-TAEA, PPS, and AnD on them. Besides, Tables 1 and 2 also show that ToP performs better than the proposed CD-CMEA in solving CTP1-5,7. That is not surprising. Because the update mechanism of ToP is NSGA-II, which has excellent performance in capturing irregular CPF in two-dimensional objective space. As NSGA-II performs poorly in selecting the individuals with good diversity in the space with the dimension number larger than 2, we can see that the performance of ToP on the three-objective instances C2-DTLZ2, C3-DTLZ4 and DC1-DTLZ1 is poor. When it comes to CTP6,8, whose CPFs below the infeasible regions in the objective space, the feasibility-led CMEAs, i.e., CMOEA/D-DE-CDP, CMOEA/D-DE-SR, ToP and AnD, are easily trap into the local areas and than fail to find the entire CPFs. As the individuals of the working population in the proposed CD-CMEA are forced to distribute evenly in the valuable areas according to their search diameters, they can easily cross the infeasible regions of the CTP6,8 and find the CPFs. So, it is natural that the performance of CD-CMEA is better than that of CMOEA/D-DE-CDP, CMOEA/D-DE-SR, ToP and AnD on CTP6,8. Due to the objective-led CHT that ignores the constraints to only optimize the objectives, C-TAEA and PPS can also effectively force the working population to cross the infeasible region in front of CPFs and find CPFs of CTP6,8. Nevertheless, because the diversity maintenance mechanism of CD-CMEA performs better than that of C-TAEA and PPS in capturing the irregular CPFs of CTP6,8, CD-CMEA is better than C-TAEA and PPS in dealing with CTP6,8.

For dealing with DASCMOP1–9, MW1–14, DC2-DTLZ1 and DC3-DTLZ1, it can see from Tables 1 and 2 that the proposed CD-CMEA is superior to CMOEA/D-DE-CDP, CMOEA/D-DE-SR, ToP and AnD. The reason for this is easy to get. As there are complex infeasible regions in front of the CPFs of DASCMOP1–9, MW1–14, DC2-DTLZ1 and DC3-DTLZ1 in the objective space, the feasibility-led CHT in CMOEA/D-DE-CDP, CMOEA/D-DE-SR, ToP and AnD fails to guide the working population to cross these infeasible regions, and this leads to the poor performances of CMOEA/D-DE-CDP, CMOEA/D-DE-SR, ToP and AnD on DASCMOP1–9, MW1–14, DC2-DTLZ1 and DC3-DTLZ1. For the three algorithms CD-CMEA, C-TAEA and PPS, they can effectively cross the complex infeasible regions of these instances and find the CPFs. However, because some instance of DASCMOP-series is the unbalanced problem, that is, the probability of populations appearing in different regions of the objective space is very different, the objective-led CHT in C-TAEA and PPS is easy to mislead the working population to the local feasible regions and miss some other feasible regions containing CPF fragments. So, C-TAEA and PPS fail to find the entire CPFs of some instance of DASCMOP1–9. As the individuals of the working population in the proposed CD-CMEA are forced to distribute evenly in the valuable areas according to their search diameters, they can easily find the entire CPFs of DASCMOP1–9. In addition, the CPFs of DASCMOP1–9, MW1–14, DC2-DTLZ1 and DC3-DTLZ1 are irregular, and CD-CMEA performs better than C-TAEA and PPS

in capturing them. Therefore, it is not surprising that CD-CMEA is superior to C-TAEA and PPS on DASCMOP1–9, MW1–14, DC2-DTLZ1 and DC3-DTLZ1.

In summary, compared with the six state-of-the-art CMEAs, i.e., CMOEA/D-DE-CDP, CMOEA/D-DE-SR, C-TAEA, PPS, ToP and AnD, the proposed CD-CMEA performs better and has better robustness in dealing with different CMOPs. In other words, the proposed CD-CMEA is a promising algorithm for CMOPs.

4.3. Further investigation

In order to investigate the effective of the proposed compound distance, we take the DC2-DTLZ1 as an example to compare the differences of archives obtained by the proposed algorithm using the compound distance and the Euclidean distance, respectively. The comparison results are shown in Figure 4. It can be seen from Figure 4 that Euclidean distance prefers the solutions far from the origin in the objective space and fails to guide the population to cross the infeasible regions in front of CPF. On the contrary, the proposed compound distance can effectively guide the population to search the different areas of the objective space evenly, whether they are close to or far from the origin. This makes the population guided by the proposed compound distance easily cross the infeasible regions in front of CPF and then find CPF of DC2-DTLZ1. Therefore, compared with Euclidean distance, the proposed compound distance is more beneficial to guide the population to explore the objective space evenly when solving the CMOPs.

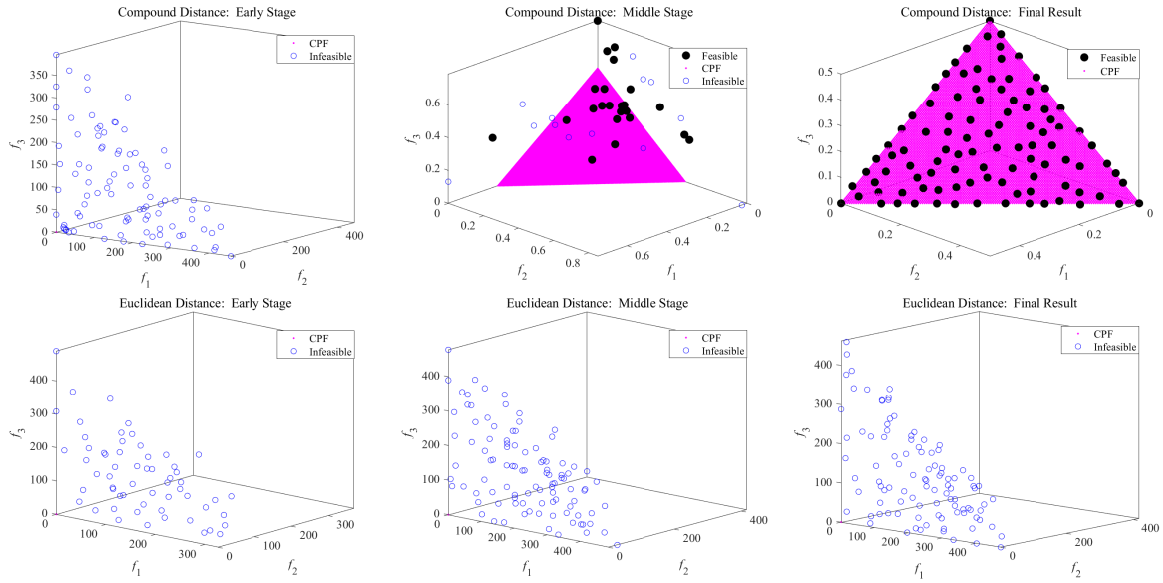

Figure 4. Comparison results of the archives obtained by the proposed compound distance and the Euclidean distance at different evolutionary stages in dealing with DC2-DTLZ1.

5. Conclusions and future work

In this paper, we introduce a compound distance into the objective space to measure individual's search diameter according to the angle and ℓ_p-norm. After that, a compound distance-based CHT, in which the individuals with larger search diameter in the valuable areas are given priority to be

preserved, is proposed to force the working population to fully explore the valuable areas that are not dominated by feasible solutions. Embedding the compound distance-based CHT in the evolutionary algorithm, we proposed a new method named CD-CMEA to deal with CMOPs. Numerical experimental results show that the proposed CD-CMEA performs better than the other six existing state-of-the-art CMEAs in dealing with different CMOPs. So, CD-CMEA is a promising method for solving CMOPs. Considering that the main characteristic of the proposed CD-CMEA is to force the working population to explore the valuable areas uniformly, we believe that the CHTs with similar properties should also perform well in dealing with CMOPs. Thus, in the future, we will further study the other forms of CHTs that can force the working population to explore the valuable areas evenly.

Conflict of interest

The authors declared that they have no conflicts of interest to this work.

References

1. M. Baioletti, A. Milani, V. Santucci, *MOEA/DEP: An algebraic decomposition-based evolutionary algorithm for the multiobjective permutation flowshop scheduling problem*, In: *European Conference on evolutionary Computation in Combinatorial Optimization*, Springer, Cham, (2018), 132–145.
2. M. Zangari, A. Mendiburu, R. Santana, A. Pozo, Multiobjective decomposition-based mallows models estimation of distribution algorithm. A case of study for permutation flowshop scheduling problem, *Inf. Sci.*, **397** (2017), 137–154.
3. O. Kabadurmus, M. F. Tasgetiren, H. Oztop, M. S. Erdogan, *Solving 0-1 bi-objective multi-dimensional knapsack problems using binary genetic algorithm*, Springer, Cham, (2021), 51–67.
4. K. Florios, G. Mavrotas, Generation of the exact pareto set in multi-objective traveling salesman and set covering problems, *Appl. Math. Comput.*, **237** (2014), 1–19.
5. H. K. Singh, T. Ray, R. Sarker, Optimum oil production planning using infeasibility driven evolutionary algorithm, *Evol. Comput.*, **21** (2013), 65–82.
6. Y. Yang, J. Liu, S. Tan, A constrained multi-objective evolutionary algorithm based on decomposition and dynamic constraint-handling mechanism, *Appl. Soft Comput.*, **89** (2020), 106104.
7. J. Yuan, H. L. Liu, C. Peng, Population decomposition-based greedy approach algorithm for the multi-objective knapsack problems, *Int. J. Pattern Recognit Artif. Intell.*, **31** (2017), 1759006.
8. H. Ma, H. Wei, Y. Tian, R. Cheng, X. Zhang, A multi-stage evolutionary algorithm for multi-objective optimization with complex constraints, *Inf. Sci.*, **560** (2021), 68–91.
9. P. Myszkowski, M. Laszczyk, Diversity based selection for many-objective evolutionary optimisation problems with constraints, *Inf. Sci.*, **546** (2021), 665–700.
10. M. Ming, A. Trivedi, R. Wang, D. Srinivasan, T. Zhang, A dual-population based evolutionary algorithm for constrained multi-objective optimization, *IEEE Trans. Evol. Comput.*, 2021. DOI: 10.1109/TEVC.2021.3066301.

11. K. Deb, An efficient constraint handling method for genetic algorithms, *Comput. Methods Appl. Mech. Eng.*, **186** (2000), 311–338.

12. Z. Fan, W. Li, X. Cai, H. Huang, Y. Fang, Y. You, et al., An improved epsilon constraint-handling method in MOEA/D for CMOPs with large infeasible regions, *Soft Comput.*, **23** (2019), 12491–12510.

13. Z. Liu, Y. Wang, Handling constrained multiobjective optimization problems with constraints in both the decision and objective spaces, *IEEE Trans. Evol. Comput.*, **23** (2019), 870–884.

14. J. P. Li, Y. Wang, S. Yang, Z. Cai, *A comparative study of constraint-handling techniques in evolutionary constrained multiobjective optimization*, In: *2016 IEEE Congress on Evolutionary Computation (CEC)*, IEEE, (2016), 4175–4182.

15. Y. Yang, J. Liu, S. Tan, H. Wang, A multi-objective differential evolutionary algorithm for constrained multi-objective optimization problems with low feasible ratio, *Appl. Soft Comput.*, **80** (2019), 42–56.

16. H. Afshari, W. Hare, S. Tesfamariam, Constrained multi-objective optimization algorithms: Review and comparison with application in reinforced concrete structures, *Appl. Soft Comput.*, **83** (2019), 105631.

17. M. A. Jan, N. Tairan, R. A. Khanum, *Threshold based dynamic and adaptive penalty functions for constrained multiobjective optimization*, In: *2013 1st International Conference on Artificial Intelligence, Modelling and Simulation*, IEEE, (2013), 49–54.

18. A. Panda, S. Pani, A symbiotic organisms search algorithm with adaptive penalty function to solve multi-objective constrained optimization problems, *Appl. Soft Comput.*, **46** (2016), 344–360.

19. L. Jiao, J. Luo, R. Shang, F. Liu, A modified objective function method with feasible-guiding strategy to solve constrained multi-objective optimization problems, *Appl. Soft Comput.*, **14** (2014), 363–380.

20. H. Jain, 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.*, **18** (2013), 602–622.

21. R. Cheng, Y. Jin, M. Olhofer, B. Sendhoff, A reference vector guided evolutionary algorithm for many-objective optimization, *IEEE Trans. Evol. Comput.*, **20** (2016), 773–791.

22. M. A. Jan, R. A. Khanum, A study of two penalty-parameterless constraint handling techniques in the framework of MOEA/D, *Appl. Soft Comput.*, **13** (2013), 128–148.

23. Z. Z. Liu, Y. Wang, P. Q. Huang, AnD: A many-objective evolutionary algorithm with angle-based selection and shift-based density estimation, *Inf. Sci.*, **509** (2020), 400–419.

24. Z. Fan, Y. Fang, W. Li, X. Cai, C. Wei, E. Goodman, MOEA/D with angle-based constrained dominance principle for constrained multi-objective optimization problems, *Appl. Soft Comput. J.*, **74** (2019), 621–633.

25. M. Asafuddoula, T. Ray, R. Sarker, A decomposition-based evolutionary algorithm for many objective optimization, *IEEE Trans. Evol. Comput.*, **19** (2014), 445–460.

26. S. Z. Martinez, C. A. C. Coello, *A multi-objective evolutionary algorithm based on decomposition for constrained multi-objective optimization*, In: *2014 IEEE Congress on evolutionary computation (CEC)*, IEEE, (2014), 429–436.
27. F. Qian, B. Xu, R. Qi, H. Tianfield, Self-adaptive differential evolution algorithm with α-constrained-domination principle for constrained multi-objective optimization, *Soft Comput.*, **16** (2012), 1353–1372.
28. C. Peng, H. L. Liu, E. D. Goodman, Handling multi-objective optimization problems with unbalanced constraints and their effects on evolutionary algorithm performance, *Swarm Evol. Comput.*, **55** (2020), 100676.
29. Z. Fan, W. Li, X. Cai, H. Li, C. Wei, Q. Zhang, et al., Push and pull search for solving constrained multi-objective optimization problems, *Swarm Evol. Comput.*, **44** (2019), 665–679.
30. Z. Fan, Z. Wang, W. Li, Y. Yuan, Y. You, Z. Yang, et al., Push and pull search embedded in an M2M framework for solving constrained multi-objective optimization problems, *Swarm Evol. Comput.*, **54** (2020), 100651.
31. K. Li, R. Chen, G. Fu, X. Yao, Two-archive evolutionary algorithm for constrained multiobjective optimization, *IEEE Trans. Evol. Comput.*, **23** (2018), 303–315.
32. Y. Tian, T. Zhang, J. Xiao, X. Zhang, Y. Jin, A coevolutionary framework for constrained multi-objective optimization problems, *IEEE Trans. Evol. Comput.*, **25** (2021), 102–116.
33. J. Yi, J. Bai, H. He, J. Peng, D. Tang, ar-MOEA: A novel preference-based dominance relation for evolutionary multiobjective optimization, *IEEE Trans. Evol. Comput.*, **23** (2018), 788–802.
34. Q. Zhang, H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, *IEEE Trans. Evol. Comput.*, **11** (2007), 712–731.
35. K. Li, K. Deb, Q. Zhang, S. Kwong, An evolutionary many-objective optimization algorithm based on dominance and decomposition, *IEEE Trans. Evol. Comput.*, **19** (2014), 694–716.
36. H. L. Liu, F. Gu, Q. Zhang, Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems, *IEEE Trans. Evol. Comput.*, **18** (2013), 450–455.
37. J. Yuan, H. L. Liu, F. Gu, Q. Zhang, Z. He, Investigating the properties of indicators and an evolutionary many-objective algorithm based on a promising region, *IEEE Trans. Evol. Comput.*, **25** (2021), 75–86.
38. J. Yuan, H. L. Liu, F. Gu, A cost value based evolutionary many-objective optimization algorithm with neighbor selection strategy, In: *2018 IEEE Congress on Evolutionary Computation (CEC)*, IEEE, (2018), 1–8.
39. X. Chen, Z. Hou, J. Liu, Multi-objective optimization with modified pareto differential evolution, In: *2008 International Conference on Intelligent Computation Technology and Automation (ICICTA)*, IEEE, (2008), 90–95.
40. K. Deb, M. Goyal, A combined genetic adaptive search (GeneAS) for engineering design, *Comput. Sci. Inf.*, **26** (1996), 30–45.
41. K. Deb, A. Pratap, T. Meyarivan, Constrained test problems for multi-objective evolutionary optimization, In: *International Conference on Evolutionary Multi-Criterion Optimization*, Springer, (2001), 284–298.

42. Z. Fan, W. Li, X. Cai, H. Li, C. Wei, Q. Zhang, et al., Difficulty adjustable and scalable constrained multi-objective test problem toolkit, *Evol. Comput.*, **28** (2020), 339–378.

43. Z. Ma, Y. Wang, Evolutionary constrained multiobjective optimization: Test suite construction and performance comparisons, *IEEE Trans. Evol. Comput.*, **23** (2019), 972–986.

44. Z. Z. Liu, Y. Wang, Handling constrained multiobjective optimization problems with constraints in both the decision and objective spaces, *IEEE Trans. Evol. Comput.*, **23** (2019), 870–884.

45. P. A. Bosman, D. Thierens, The balance between proximity and diversity in multiobjective evolutionary algorithms, *IEEE Trans. Evol. Comput.*, **7** (2003), 174–188.

46. E. Zitzler, L. Thiele, Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach, *IEEE Trans. Evol. Comput.*, **3** (1999), 257–271.

47. I. Das, J. E. Dennis, Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems, *SIAM J. Optim.*, **8** (1998), 631–657.