

Article

Hierarchical Multi-Population Cooperative Evolutionary Approach for Constrained Multi-Objective Optimization

Xinyue Xiang ^{1,2}, Xinyuan Gu ¹ , Jiaqi Li ^{1,*}  and Siyuan Zhao ^{1,*}

¹ Faculty of Humanities and Arts, Macau University of Science and Technology, Macao 999078, China; 202211010061@stu.shmtu.edu.cn (X.X.); 3230003458@student.must.edu.mo (X.G.)

² Institute of Mathematics, Shanghai Maritime University, Haigang Ave 1550, Shanghai 201306, China

* Correspondence: jqli@must.edu.mo (J.L.); zhaosiyuan@must.edu.mo (S.Z.)

Abstract

To address the complexity of multi-constraint multi-objective optimization problems (mC-MOPs), this paper proposes a novel multi-population evolutionary algorithm (MOEA). Multi-objective optimization problems (MOPs) are ubiquitous in scientific and engineering fields, while the introduction of multiple complex constraints significantly increases the difficulty of finding solutions. To tackle this challenge, this work systematically analyzes the intrinsic relationships among the Single-Constraint Pareto Front (SCPF), Sub-Constraint Pareto Front (SSCPF), Unconstrained Pareto Front (UPF), and the Final Constrained Pareto Front (FCPF), and it investigates how these relationships can be leveraged to effectively enhance optimization performance. Based on this analysis, a Hierarchical Multi-Population Cooperative Evolutionary Approach (HMP-CE) is proposed. The approach constructs $C + 2$ populations (where C represents the number of constraints) to search the UPF, SCPF, and SSCPF at appropriate stages, thereby driving the final solution approximation in a hierarchical manner. Meanwhile, HMP-CE introduces the following two key mechanisms: (1) the Population Activation–Dormancy Regulation (PADR) mechanism, which adaptively regulates the activation and dormancy of populations to reduce computational cost and accelerate convergence; (2) the Constraint Combination Timing Identification (CCTI) mechanism, which identifies suitable moments to jointly solve selected constraints in SSCPF, thereby enhancing cooperative efficiency among populations. Experimental results on 37 benchmark mCMOPs, and six real-world engineering problems demonstrate that the proposed algorithm exhibits superior performance in terms of convergence, feasibility, and solution diversity, providing a competitive approach for solving complex multi-constraint optimization problems.

Keywords: multi-constraint multi-objective optimization problems; multiple complex constraints; multi-population; hierarchical manner; constraint combination timing identification

MSC: 68W50; 65K10; 49K30



Received: 19 January 2026

Revised: 17 February 2026

Accepted: 20 February 2026

Published: 26 February 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Multi-objective optimization problems (MOPs) [1,2] are widely encountered in scientific and engineering applications [3–5]. In such problems, multiple decision variables often exert interdependent influences on multiple optimization objectives [6,7]. Therefore, it is quite difficult to simultaneously obtain ideal solutions for all objectives. In comparison, constrained multi-objective optimization problems (CMOPs) [8] are more challenging than

general MOPs, because in the process of searching for satisfactory solutions, it is necessary to not only optimize multiple objectives but to also satisfy constraints that may arise from the decision space, the objective space, or both. In general, a CMOP can be formally defined as follows [9]:

$$\begin{aligned} \min_{X \in \Omega} \quad & F(X) = (f_1(X), f_2(X), \dots, f_m(X)) \\ \text{s.t.} \quad & g_i(X) \leq 0, \quad i = 1, \dots, p, \\ & h_j(X) = 0, \quad j = 1, \dots, q, \end{aligned} \quad (1)$$

Here, $X = (x_1, x_2, \dots, x_n)$ denotes an n -dimensional decision variable vector defined in the decision space Ω ; $F(X)$ is a vector composed of m mutually conflicting objective functions; $g_i(X)$ represents the i -th inequality constraint, and $h_j(X)$ represents the j -th equality constraint. When the total number of constraints satisfies $i + j > 1$, the problem is classified as a mCMOP.

In MOPs, the quality of solutions is usually evaluated based on dominance relations [10]. Specifically, given two solutions X_a and X_b , X_a is said to dominate X_b , denoted as $X_a \prec X_b$, if and only if the following conditions hold [11]:

$$\begin{aligned} f_i(X_a) &\leq f_i(X_b), \quad \forall i \in \{1, \dots, m\}, \\ \exists j \in \{1, \dots, m\} \quad &\text{s.t.} \quad f_j(X_a) < f_j(X_b). \end{aligned} \quad (2)$$

If neither $X_a \prec X_b$ nor $X_b \prec X_a$ holds, the two solutions are said to be non-dominated [12] with respect to each other. The set of all mutually non-dominated solutions is called the non-dominated solution set [13], and its mapping in the objective space is referred to as the Pareto Front (PF) [14].

For a CMOP, let Ω denote the decision space, $F(X) = (f_1(X), f_2(X), \dots, f_m(X))$ the objective vector, and $g_i(X) \leq 0$ ($i = 1, \dots, p$), $h_j(X) = 0$ ($j = 1, \dots, q$) the constraints. Pareto Set (PS) is defined as the set of all non-dominated solutions in the decision space as follows [15]:

$$PS = \{X \in \Omega \mid X \text{ is non-dominated}\}. \quad (3)$$

The UPF is the mapping of the Pareto Set in the objective space without considering constraints as follows [16]:

$$UPF = \{F(X) \mid X \in PS\}. \quad (4)$$

The Constrained Pareto Set (CPS) is the subset of the Pareto Set that satisfies all constraints as follows [17]:

$$CPS = \{X \in PS \mid g_i(X) \leq 0, h_j(X) = 0\}. \quad (5)$$

The CPF is the mapping of the Constrained Pareto Set in the objective space as follows [18]:

$$CPF = \{F(X) \mid X \in CPS\}. \quad (6)$$

This distinction reveals the influence of constraints on the structure and distribution of solutions in both decision and objective spaces.

In CMOPs, the degree of violation of the j -th constraint by a solution is usually denoted as $C_j(X)$. This measure effectively reflects the feasibility deviation of the solution under the given constraints, and its mathematical definition is as follows [19]:

$$C_j(X) = \begin{cases} \max\{0, g_j(X)\}, & j = 1, \dots, p, \\ \max\{0, |h_{j-p}(X)| - \delta\}, & j = p + 1, \dots, p + q, \end{cases} \quad (7)$$

δ is a small relaxation factor (typically set to 1×10^{-6}), introduced to relax the strictness of equality constraints. Since equality constraints are usually more difficult to satisfy than inequality constraints, the introduction of δ helps to make the solution process more stable.

By calculating the violation degree of each constraint and summing them up, the total constraint violation of a solution, denoted as $CV(X)$, can be obtained. When $CV(X) = 0$, the solution satisfies all constraints and is therefore considered feasible; otherwise, it is regarded as infeasible [20]

$$CV(X) = \sum_{j=1}^{p+q} c_j(X). \quad (8)$$

The core objective of CMOPs is to obtain a set of feasible non-dominated solutions; therefore, feasibility is regarded as the primary consideration and must be satisfied first. Constraints represent the main challenge in achieving feasibility [21–24]. At the same time, convergence is also an indispensable factor in the solution process. Without effective convergence, the final solutions may fail to maintain non-dominance at the global level. In addition, intrinsic characteristics of MOPs, such as local optima or the growth in the dimensionality of decision variables, can further expand the decision space, thereby increasing the complexity of convergence. Diversity is another critical factor that must be considered in solving CMOPs. A rich set of non-dominated solutions can provide decision-makers with more choices. However, constraints may make it difficult to maintain diversity; for example, infeasible regions may lead to a discretized CPF. Similarly, inherent properties of MOPs, such as premature convergence, may also restrict solution diversity. Therefore, compared with general MOPs, CMOPs not only require additional satisfaction of feasibility requirements, but they also present greater challenges in terms of convergence and diversity.

2. Motivation

Multi-population constrained multi-objective evolutionary algorithms (CMOEAs) typically rely on collaboration among populations to improve the overall solution quality. However, designing a universal collaboration scheme suitable for all CMOPs is extremely challenging, which has attracted widespread attention. Different multi-population CMOEAs adopt varying collaboration strategies, resulting in performance differences across algorithms. The main factors affecting the final performance include the strength, content, and timing of collaboration. In this paper, we present our analysis and insights from these three perspectives.

2.1. Collaboration Strength

This paper first explores the appropriate degree of collaboration among populations. Collaboration strength is a core factor in multi-population optimization. Early methods, such as CTAEA [25], employed a strong collaboration strategy, where offspring individuals were generated from parents belonging to different populations. In CTAEA, the auxiliary population DA is used to explore regions not covered by the main population CA, including infeasible spaces, to identify potentially feasible solutions. However, when faced with

CMOPs where the UPF and CPF are separated, CA simultaneously considers convergence and feasibility, making it difficult to achieve satisfactory results. Strong collaboration between DA and CA in this case exacerbates the problem, leading to difficulties in obtaining high-quality or feasible solutions. To address this, Tian et al. [26] proposed the weak collaboration principle in CCMO. The main population, Population1, evaluates the CMOP using NSGAI-CDP, while the auxiliary population, Population2, employs fast non-dominated sorting. The two populations collaborate only during the environmental selection phase and remain independent during reproduction. The weak collaboration principle provides a systematic exploration of collaboration strength and has been widely recognized. Currently, state-of-the-art multi-population CMOEAs [27,28] mostly adopt weak collaboration or strategies of similar intensity. However, CCMO does not thoroughly investigate the content or timing of collaboration. The fixed and continuous collaboration scheme limits its ability to achieve ideal solutions for CMOPs where the UPF and CPF are far apart. Therefore, further study on the specific content and optimal timing of collaboration is of great significance.

2.2. Collaboration Content

This section focuses on which types of solutions should be selected during collaboration. The information contained in infeasible regions is of significant value for solving CMOPs. Constraint relaxation can alleviate, to some extent, the three major challenges faced by CMOPs. To fully exploit the information in infeasible regions, various approaches have been proposed. In recent years, several multi-population CMOEAs [29,30] have adopted the epsilon method, which treats constraints as a whole and relaxes the total constraint violation (CV) to effectively explore information in infeasible regions. Advanced CMOEAs currently also employ similar strategies to some extent. For example, in CCMO, the main population fully considers feasibility, while the auxiliary population disregards it. However, the collaboration content is often too singular and fixed, limiting the auxiliary population's effectiveness when handling CMOPs where the UPF and CPF are separated. EMCMO [31] systematically investigates collaboration content. Its basic framework is similar to CCMO, using two populations to handle constrained and unconstrained CMOPs, respectively, and employing a weak collaboration strategy. However, EMCMO's collaboration is no longer solely based on the offspring of the two populations, but it instead selects individuals based on the quality of the other population or its offspring. Similarly, Zou et al. [32] adopt a dual-population structure in CAEAD, also emphasizing weak collaboration. Specifically, the auxiliary population Pop_2 dynamically adjusts epsilon to control constraint intensity, thereby exploring infeasible regions; when Pop_2 degenerates from the UPF to Pop_1 , collaboration is reinforced. Through these strategies optimizing collaboration content, such CMOEAs generally achieve higher-quality solutions.

In real-world problems [33], CMOPs often involve multiple constraints [34–39], which significantly increase the difficulty of finding solutions. Existing methods generally relax constraints by treating them as a whole, but they often ignore the interactions among constraints, resulting in insufficient exploitation of inter-constraint information. When the difficulty of individual constraints is the same, the problem's complexity increases with the number of constraints. Therefore, existing CMOEAs still face considerable challenges when tackling CMOPs with multiple and complex constraints.

A common approach for tackling complex problems is to decompose them into several subproblems. The same idea can be applied to solving multi-constraint CMOPs (mCMOPs). Specifically, each constraint can be evaluated individually. Compared with relaxing all constraints as a whole, evaluating each constraint separately is more conducive to identifying potentially feasible solutions, especially for CMOPs where the UPF and CPF are separated,

since many CPFs are formed by constraint boundaries (which will be discussed in detail in the next section). In addition, collaboration among constraints can accelerate the evaluation process and improve solving efficiency.

Taking MW6 [40] as an example, the problem contains multiple infeasible region constraints. As shown in the Figure 1, the gray areas denote the feasible regions, while the red small matrices and blue crosses represent the evaluations of Constraint 1 and Constraint 2, respectively. The gray area represents the feasible region, while the white area represents the infeasible region. Two populations exchange information through weak cooperation. In the Early stage (Figure 1a), some solutions related to Constraint 2 are trapped in local optima due to the presence of infeasible regions, while most solutions within the feasible domain of Constraint 1 remain unaffected and can provide valuable information to help Constraint 2 escape stagnation. As the algorithm iterates into the Early-Mid stage (Figure 1b), with the continuous assistance of solutions provided by Constraint 1, Constraint 2 successfully crosses the infeasible regions and reaches other feasible areas. Meanwhile, Constraint 2 guides Constraint 1 toward feasible regions. By the Mid stage (Figure 1c), Constraint 1, assisted by Constraint 2, successfully crosses its large infeasible region, although some of its solutions have not yet reached the Pareto front. The cooperation mechanism continues to play a role, with Constraint 2 in turn guiding Constraint 1 toward better regions. Finally, in the Final outcome stage (Figure 1d), after continuous collaboration, both populations converge to the Pareto front, and infeasible regions no longer hinder progress. This process clearly illustrates how weak cooperation enables different populations to complement each other's weaknesses and collectively achieve superior optimization results.

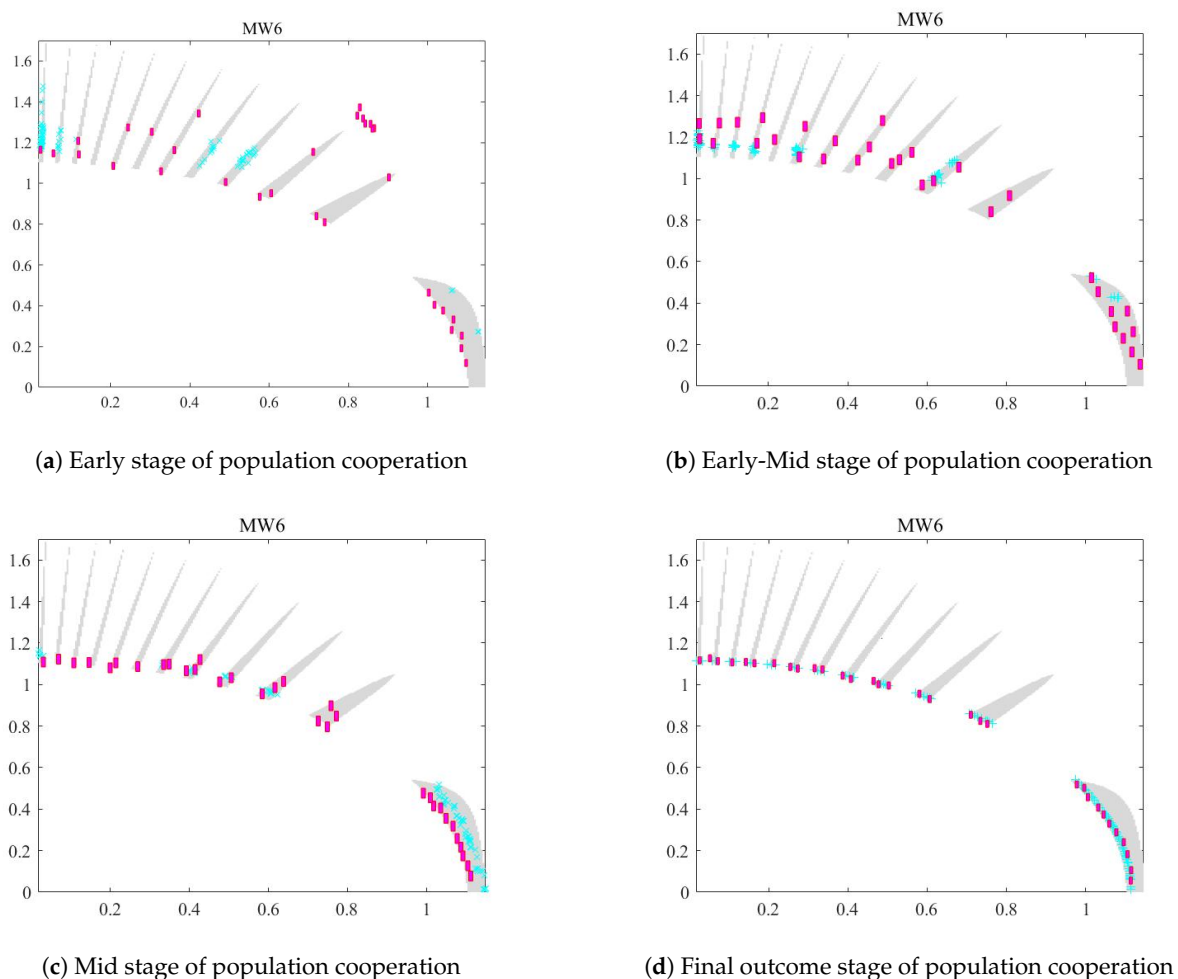


Figure 1. Evolution of population cooperation from Early to Final stage.

As mentioned earlier, solving CMOPs with fewer constraints is generally simpler than solving the original CMOP. Therefore, an unconstrained population can be introduced during the collaboration process to more effectively handle situations where the UPF and CPF partially overlap. At the same time, this strategy can accelerate the evaluation of single-constraint and fully constrained problems, as it helps prevent other populations from being trapped in infeasible regions commonly occurring between individual constraints, which may lead to local optima. The role of the unconstrained population is to enhance the search capability and explore a larger search space.

2.3. Collaboration Timing

This subsection focuses on the following two aspects of the collaboration mechanism: the collaboration among populations has clearly defined initiation and termination timing; in addition, under specific circumstances, the collaboration intensity needs to be properly adjusted. It is worth noting that in multi-constrained CMOPs (mCMOPs), the final feasible region is the intersection of all individual constraint feasible regions.

2.3.1. Initiation and Termination Timing of Collaboration

In multi-constrained MOPs, evaluating all constraints simultaneously increases the overall computational burden of the algorithm [8,41–46]. Moreover, in the early stages of evolution, the solutions of populations corresponding to different constraints are often highly similar. Excessive evaluation of these similar subproblems within the same generation contributes little to the improvement of solution quality. Therefore, it is necessary to put certain populations into dormancy or activation at appropriate times. It should be noted that dormant populations only participate in environmental selection without generating offspring, while activated populations participate in both environmental selection and offspring generation. The arrangement of dormancy and activation should be determined based on the respective responsibilities of the populations. The roles of the populations and, accordingly, their determined specific dormancy and activation schedules are shown in Table 1.

Table 1. Responsibilities and activation–dormancy rules of each population.

Population	Responsibility	Activation/Dormancy Rules
All-Constraint Population (P_{Acp})	Evaluate all constraints and find final feasible solutions.	Activated at initialization and always remains active.
Sub-Constraint Population (P_{Sscp})	Evaluate single or combined constraints to partially relax them, helping P_{Acp} find feasible solutions.	Dormant at initialization. Activated when P_{Acp} or other populations cannot further assist it. Remains active after constraints are combined; removed after combining all constraints (similar to P_{Acp}).
Unconstrained Population (P_{Ucp})	Evaluate no constraints to explore the entire objective space and assist P_{Acp} and P_{Sscp} in finding feasible solutions.	Activated at initialization. After reaching UPF, it may enter dormancy if UPF and CPF are separated; otherwise, remains active.

2.3.2. Whether Collaboration Should Be Strengthened in Specific Situations

When individuals from different populations participate in offspring generation, the offspring may inherit partial genes from various populations. Therefore, in cases where the UPF and CPF partially overlap, it is necessary to enhance the collaboration

between P_{Acp} and P_{Ucp} . Such strengthened interaction can effectively help P_{Acp} to explore high-quality potential solutions in this context.

Based on the above analysis, we propose a novel Hierarchical Multi-Population Cooperative Evolutionary Approach for Constrained Multi-Objective Optimization (HMP-CE). This mechanism employs $C + 2$ populations (where C is the number of constraints) to solve the problem. The primary population, P_{Acp} , is responsible for searching for the final feasible solutions and handling the original, highly difficult CMOP. The secondary constrained populations, P_{Sscp} , partially relax constraints to explore potential feasible solutions for P_{Acp} , address moderately difficult CMOPs, and assist in solving the most challenging tasks of P_{Acp} . The unconstrained population, P_{Ucp} , completely relaxes all constraints, deals with the least difficult CMOP, and accelerates the evaluation of both P_{Acp} and P_{Sscp} . At the initial stage of the algorithm, P_{Acp} and P_{Ucp} are activated, while P_{Sscp} remains dormant. When P_{Acp} and P_{Ucp} can no longer provide further support for a given P_{Sscp} , that P_{Sscp} is activated. Once activated, P_{Sscp} will merge with other populations at appropriate times and, after aligning with the constraints of P_{Acp} , it will re-enter dormancy. When P_{Ucp} reaches the UPF, whether to maintain and strengthen collaboration is determined according to the relationship between the UPF and CPF.

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

- (1) This paper proposes a novel HMP-CE algorithm, which introduces $C + 1$ auxiliary populations to assist the evolution of the primary population. The populations exchange knowledge through a weak collaboration mechanism, thereby improving the overall efficiency of the evolutionary process and the quality of the solutions.
- (2) This paper proposes a Population Activation–Dormancy Regulation (PADR) mechanism, which is designed to identify appropriate timing for populations to enter dormancy or become activated, thereby reducing the computational burden of auxiliary populations while maintaining the effectiveness of the final solutions.
- (3) In HMP-CE, a Constraint Combination Timing Identification (CCTI) mechanism is introduced to combine sub-constraints or single constraints at appropriate times. This mechanism not only helps enhance the effectiveness of knowledge transfer across populations but also further reduces the computational resource consumption of auxiliary populations.
- (4) Through comparative experiments on 32 test suites, nine scalable mCMOPs, and several real-world engineering problems, the results show that the proposed algorithm demonstrates significant competitiveness in terms of performance.

The remainder of this paper is organized as follows: Section 3 provides a review of existing constrained multi-objective evolutionary algorithms (CMOEAs); Section 4 introduces the proposed HMP-CE algorithm and its implementation details; Section 5 evaluates the performance of HMP-CE on test suites and real-world CMOP problems, and it compares it with existing multi-population CMOEAs; and Section 6 concludes the paper and discusses directions for future research.

3. Related Work

Existing methods for solving constrained multi-objective optimization problems can be broadly categorized into three types. The first category focuses on single-strategy CMOEAs, which apply a uniform strategy across the entire evolutionary process. The second category employs stage-specific strategies, adapting the algorithm's behavior at different evolutionary stages. The third category leverages differentiated strategies across multiple populations, enabling more effective exploration and exploitation by assigning distinct roles and strategies to each population.

3.1. Single-Strategy CMOEAs

In early studies, methods for solving CMOPs mostly adopted a single strategy. A typical example is one whose approach is simple, effective, and easily integrated with other techniques. Its basic principle is to prioritize feasible solutions; when both candidate solutions are infeasible, the one with the smaller constraint violation (CV) is chosen. The Epsilon method proposed by Takahama and Saika [47] treats solutions with CV smaller than ϵ as “pseudo-feasible solutions,” thereby exploiting potential information from infeasible solutions. Similarly, stochastic ranking methods [48] introduce a probability parameter p_f as follows: when comparing individuals, the objective function is considered with probability p_f , and the constraint violation is considered with probability $1 - p_f$, thus fully utilizing objective function information. Moreover, certain methods consider constraints as additional objectives [49] in the optimization process. Adaptive penalty function methods improve upon static penalty functions by dynamically incorporating constraint violations into the original objective functions [50], which can effectively adapt to different types of CMOPs. Researchers in recent years have tended to adopt hybrid methods for solving CMOPs, i.e., flexibly applying multiple optimization strategies [51] at different evolutionary stages or within different populations in order to improve algorithm adaptability and solution performance.

3.2. Stage-Specific Strategies CMOEAs

Tian et al. [52] proposed the CMOEA_MS algorithm, which can adaptively choose strategy according to the characteristics of the problem, thereby achieving a dynamic balance between objective optimization and constraint satisfaction. The ToP algorithm proposed by Liu and Wang [53] evaluates only a single objective and all constraints during the initial stage, while performing a comprehensive evaluation of the original problem in the subsequent stages. The PPS method proposed by Fan et al. [54] explores infeasible regions using the Epsilon technique; when Epsilon is small, it can more finely exploit potential feasible solutions. Fan et al. [55] also proposed the Detect-and-Escape (DAE) strategy as follows: within the MOEA/D [56] framework, the population state is monitored to determine whether it has fallen into a local optimum, and the DAE strategy is applied to help the population escape from it. Similarly, Tang et al. [17] proposed a population state detection mechanism that dynamically adjusts the environmental selection strategy based on the current population state. The MSCMO algorithm proposed by Ma et al. [57] first determines an approximate range for single-constraint feasible regions, and it then gradually characterizes the final feasible region by progressively adding constraints.

3.3. Differentiated Strategies in Multiple Population CMOEAs

Multi-population strategies have been widely applied in CMOP research. In the CTAEA algorithm proposed by Li et al. [25], the main population primarily searches for feasible solutions, and if these are insufficient, it selects infeasible solutions with better objectives. The auxiliary population focuses on exploring potential feasible regions, mainly covering subregions not addressed by the main population. In the CCMO algorithm proposed by Tian et al. [26], the main population relies entirely on feasibility-based strategies, whereas the auxiliary population ignores feasibility and employs a weak collaboration mechanism. The EMCMO algorithm proposed by Qiao et al. [31] has a similar dual-population design as CCMO, but during environmental selection, it prioritizes retaining superior potential solutions. In the cDPEA algorithm by Ming et al. [58], both populations consider feasibility to varying degrees and perform adaptive correction on the obtained solutions. The bidirectional coevolution algorithm designed by Liu et al. [59] utilizes co-operation between the main population and an archive population as follows: the main

population maintains feasibility and approaches the CPF from the feasible region, while the archive population uses angular information to evaluate the diversity of the main population and approaches the CPF from infeasible regions.

The URCMO algorithm [24] analyzes the relationship between the UFP and CPF and designs differentiated offspring generation strategies according to different relationships, thereby improving the efficiency of solution search.

4. Proposed HMP-CE

In this section, we provide a detailed introduction to the framework and execution steps of the HMP-CE, followed by an in-depth analysis and presentation of the proposed innovations, PADR, and CCTI, and concluding with a comprehensive examination of HMP-CE.

4.1. The Framework of HMP-CE

The HMP-CE Algorithm 1 begins by initializing the population size N and determining the number of constraints C . A population without constraints (P_{Ucp}) is randomly generated and evaluated on the MOP, while for each individual constraint i , a single-constraint population (P_{Sscpi}) is randomly initialized and evaluated under that constraint. All populations are then combined, and an active cooperative population (P_{Acp}) is selected via environmental selection across all constraints. Initially, P_{Acp} and P_{Ucp} are active, and all P_{Sscpi} are dormant. During each iteration, offspring are generated from the active populations (P_{Acp} , P_{Ucp} , and any active P_{Sscpi}) and combined. Environmental selection is applied separately to P_{Acp} (all constraints), P_{Ucp} (no constraints), and each P_{Sscpi} (corresponding constraint), and population states are updated according to PADR rules. Single-constraint populations are monitored and merged when beneficial using the CCTI mechanism, and if only one P_{Sscpi} remains, it becomes P_{Acp} . The algorithm iterates until the termination criterion is met, ultimately returning P_{Acp} as the final optimized population. In Figure 2, the algorithm flowchart of HMP-CE is illustrated.

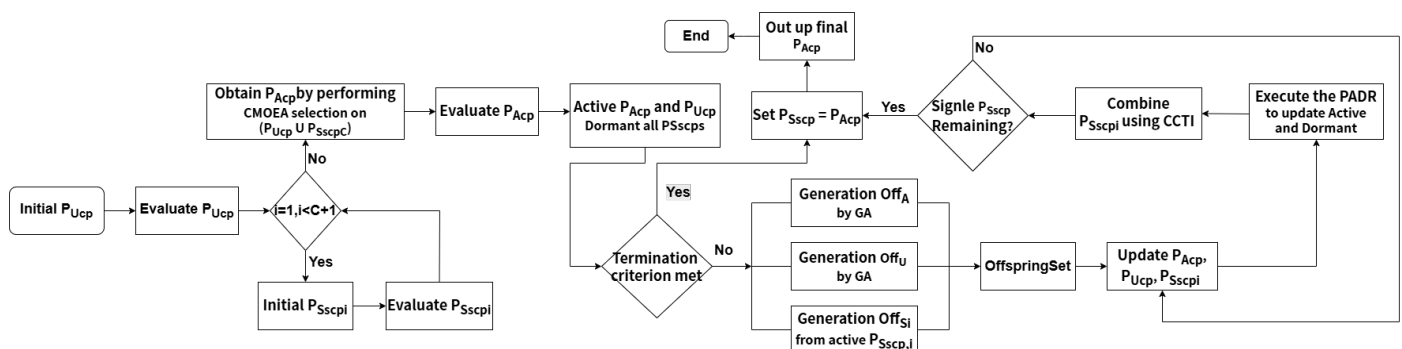


Figure 2. Framework flowchart of HMP-CE.

The proposed HMP-CE establishes a cooperative multi-population framework consisting of P_{Ucp} , P_{Sscpi} , and P_{Acp} , where P_{Ucp} and P_{Acp} remain active at initialization while P_{Sscpi} are initially dormant to reduce computational cost. Its core innovations lie in the PADR and CCTI mechanisms. PADR (Population Activation–Dormancy Regulation) adaptively controls the activation of P_{Sscpi} based on the centroid variation between consecutive generations, activating P_{Sscpi} only when additional selection pressure is required. This mechanism avoids redundant computations during the early fast-convergence stage, prevents premature convergence, and ensures good adaptability to complex constraint structure CMOPs without manual parameter tuning. Meanwhile, CCTI (Constraint Combination Timing Identification) dynamically detects and merges P_{Sscpi} with overlapping infeasible regions, enabling cooper-

ative optimization across multiple constraints and further reinforcing the coupling between P_{Sscps} and P_{Acp} . The integration of these mechanisms allows HMP-CE to achieve superior convergence speed, diversity maintenance, and adaptability compared with conventional single-population approaches, demonstrating enhanced efficiency and robustness.

Algorithm 1 Framework of HMP-CE

```

1: Input:  $N$  (population size)
2: Output:  $P_{Acp}$  (final population)
3:  $C \leftarrow$  number of constraints
4:  $P_{Ucp} \leftarrow \text{RandomInitialization}(N)$ 
5: Evaluate  $P_{Ucp}$  on MOP without constraints
6: for  $i = 1$  to  $C$  do
7:    $P_{Sscpi} \leftarrow \text{RandomInitialization}(N)$ 
8:   Evaluate  $P_{Sscpi}$  on MOP with constraint $_i$ 
9: end for
10:  $P_{Acp} \leftarrow \text{Environmental\_Selection}(P_{Ucp} \cup P_{Sscp1} \cup \dots \cup P_{SscpC}, \text{All\_Constraint})$ 
11: Evaluate  $P_{Acp}$  on MOP with all constraints
12: Set  $P_{Acp}$  and  $P_{Ucp}$  as active; all  $P_{Sscpi}$  as dormant
13: while termination criterion not met do
14:    $Off_A \leftarrow$  Genetic operator applied to active  $P_{Acp}$ 
15:    $Off_U \leftarrow$  Genetic operator applied to active  $P_{Ucp}$  (jointly with  $P_{Acp}$  only when UPF is reached)
16:   for each active  $P_{Sscpi}$  do
17:      $Off_{Si} \leftarrow$  offspring from  $P_{Sscpi}$ 
18:   end for
19:   OffspringSet  $\leftarrow [Off_A, Off_U, Off_{S1}, \dots, Off_{Sk}]$ 
20:    $P_{Acp} \leftarrow \text{Environmental\_Selection}(P_{Acp} \cup \text{OffspringSet}, \text{All\_Constraint})$ 
21:    $P_{Ucp} \leftarrow \text{Environmental\_Selection}(P_{Ucp} \cup \text{OffspringSet}, \text{No\_Constraint})$ 
22:   for each  $P_{Sscpi}$  do
23:      $P_{Sscpi} \leftarrow \text{Environmental\_Selection}(P_{Sscpi} \cup \text{OffspringSet}, \text{Corresponding\_Constraint})$ 
24:   end for
25:   Update active/dormant states using PADR
26:   Detect and combine  $P_{Sscp}$  using CCTI (e.g.,  $P_{Sscp1} + P_{Sscp3} \rightarrow P_{Sscp1,3}$ )
27:   if number of  $P_{Sscp} = 1$  then
28:      $P_{Sscp} \leftarrow P_{Acp}$ 
29:   end if
30: end while
31: return  $P_{Acp}$ 

```

4.1.1. Genetic Operator Application and Offspring Pooling

In this framework, genetic operators, including crossover and mutation, are applied exclusively to active populations. Specifically, these include the active constrained population (P_{Acp}), the unconstrained population (P_{Ucp}), and any active single-constraint populations (P_{Sscp}). Dormant populations do not generate offspring, thereby avoiding unnecessary computations. All offspring generated from active populations are pooled into a single offspring set at each generation. Environmental selection is then applied separately for each population as follows: P_{Acp} selects under all constraints, P_{Ucp} under no constraints, and each P_{Sscp} under its corresponding single constraint. This design allows information sharing among populations while maintaining distinct evolutionary dynamics. While the crossover and mutation operators themselves are identical across populations, the source of parental solutions differs depending on the population's focus (unconstrained, single-constraint, or active constrained), ensuring that each population explores its respective search space effectively.

4.1.2. Design Rationale for Population Collaboration

In HMP-CE, a weak collaboration strategy is adopted between populations rather than a fully adaptive collaboration strength. This choice ensures stability and simplicity, allowing populations to share information without over-constraining each other and reducing the risk of premature convergence. Since the activation and merging of populations are already dynamically controlled by PADR and CCTI, weak collaboration is sufficient to promote information exchange while maintaining distinct evolutionary dynamics. Empirical results further demonstrate that weak collaboration achieves a good balance between exploration and exploitation across all benchmark and real-world CMOPs.

4.2. Population Activation–Dormancy Regulation (PADR) Mechanism

The Population Activation–Dormancy Regulation (PADR Algorithm 2) is designed to dynamically manage subproblem populations (P_{Sscps}) in constrained multi-objective optimization. First, most solutions are infeasible and the algorithm exhibits rapid convergence, making it unnecessary to activate all P_{Sscps} . Therefore, PADR introduces an early dormancy mechanism where only the universal population (P_{Ucp}) and all-constraint population (P_{Acp}) remain active, while most P_{Sscps} are kept dormant to reduce computational complexity. During evolution, the activation of P_{Sscps} is controlled adaptively based on the centroid variation of the population. When the variation between two consecutive generations exceeds the adaptive threshold, the corresponding P_{Sscp} is activated to generate offspring and provide the necessary selection pressure for its associated subproblem. The adaptive threshold is calculated as follows:

$$CT = 10^{M-4} \cdot \frac{1}{MN} \sum_{i=1}^N \sum_{j=1}^M |f_j(x_i)| \quad (9)$$

where $f_j(x_i)$ denotes the value of the j -th objective function of solution x_i , and M and N represent the number of objectives and the population size, respectively. The absolute operator $|\cdot|$ indicates that the summation is taken over the absolute values of all objective function values.

Algorithm 2 Framework of the PADR

```

1: Input:  $P_{Sscp}$ ,  $N$ ,  $M$ 
2: Output: Updated active/dormant  $P_{Sscp}$ 
3: for each  $P_{Sscp_i}$  do
4:   Set  $P_{Sscp_i}$  as dormant if early stage
5: end for
6: while termination criterion not met do
7:   for each active  $P_{Sscp_i}$  do
8:     Generate offspring
9:   end for
10:  OffspringSet  $\leftarrow$  offspring from all active populations
11:  EnvironmentalSelection( $P_{Sscp}$ , OffspringSet)
12:  for each dormant  $P_{Sscp_i}$  do
13:    Distance for computing centroid variation  $D_{is}$ 
14:    if  $D_{is} \geq CT$  then
15:      Activate  $P_{Sscp_i}$ 
16:    end if
17:  end for
18: end while
19: return  $P_{Sscp}$  with updated states

```

The innovation show in Table 2 of PADR lies in its ability to achieve a balance between efficiency, adaptivity, and robustness. By avoiding unnecessary evaluations through early dormancy, PADR reduces the overhead of environmental selection. The centroid-based activation rule ensures that P_{Sscps} are only activated when needed, enabling the algorithm to self-adapt to different problem complexities without parameter tuning. Furthermore, activated P_{Sscps} enhance the optimization of their sub-CMOPs and assist P_{Acp}/P_{Ucp} in escaping local optima, thereby improving both convergence and diversity. The inclusion of the scaling factor 10^{M-4} ensures scalability and stability in real-World Problem problems [33], making PADR a key component that significantly strengthens the performance of HMP-CE. Figure 3 specifically illustrates the decision-making process of PADR.

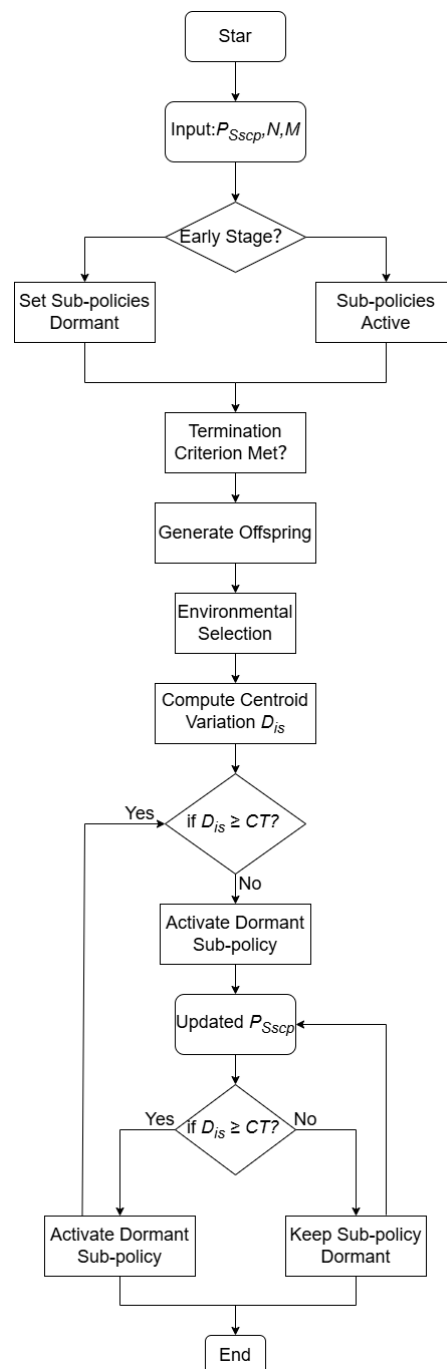


Figure 3. Framework flowchart of PADR.

Table 2. Analysis of core innovations in PADR.

Innovation	Description	Advantage
Early Dormancy Mechanism [52]	Set P_{Sscps} as dormant during the early fast-convergence stage	Saves computational resources and reduces the complexity of environmental selection
Centroid-Variation Adaptive [60] Activation	Dynamically activate P_{Sscps} based on the population centroid variation	Adaptive control without requiring fixed parameters
Subproblem Selection Pressure Maintenance [61]	Active P_{Sscps} generate offspring to provide selection pressure for subproblems	Prevents local optima and assists P_{Acp}/P_{Ucp} optimization
Real-World Problem [33] Handling Capability	Incorporates flexible activation and cooperation strategies adaptable to dynamic or uncertain environments	Enhances robustness and applicability to complex real-world optimization problems
Enhanced Cooperative Optimization [62]	Active P_{Sscps} coordinate with the main population and other P_{Sscps}	Improves overall multi-objective optimization performance and stability

PADR adaptively controls the activation of single-constraint populations (P_{Sscps}) by monitoring the centroid variation between consecutive generations, activating these populations only when additional selection pressure is required. This design is theoretically justified because the centroid represents the central tendency of the population in the objective space as follows: small centroid variation indicates that the population is approaching convergence, whereas large variation indicates ongoing exploration. As shown in the ablation study in Section 5.8, the comparison with HMP-CE2, which removes PADR, demonstrates that using centroid variation as the activation criterion can reduce redundant computations from dormant populations while maintaining convergence performance. Populations are activated only when necessary, thereby preventing premature convergence of the algorithm.

4.3. Population Constraint Combination Timing Identification (CCTI) Mechanism

The CCTI Algorithm 3 mechanism begins by examining the number of sub-constrained populations (P_{Sscps}). If only one P_{Sscp} exists, it is deleted to save computational resources, as no combination is necessary. When multiple P_{Sscps} are present, the algorithm first evaluates the relationship between the all-constraint population (P_{Acp}) and each P_{Sscp} . If the unconstrained population (P_{Ucp}) has reached the UPF, the non-dominated ranks of P_{Acp} and P_{Sscps} are computed. Any P_{Sscp} whose maximum non-dominated rank is lower than the minimum rank of P_{Acp} is marked as ready for combination. Next, the algorithm examines each P_{Sscp} for stagnation or redundant information. Additional comparisons between unmarked P_{Sscps} are performed to identify overlapping or less-helpful populations. All P_{Sscps} flagged for combination are then merged through environmental selection to form the combined P_{Sscp} , and the original redundant P_{Sscps} are deleted. If only one P_{Sscp} remains and it is identical to P_{Acp} , it is also removed to conserve resources. The final P_{SscpT} is returned for subsequent iterations.

The main innovations of this CCTI Algorithm 3 framework lie in its dynamic and adaptive handling of redundant or stagnated subpopulations. By employing a two-tiered judgment mechanism— P_{Acp} versus P_{Sscp} and P_{Sscp} versus P_{Sscp} —the algorithm efficiently identifies and eliminates low-contribution or repetitive subpopulations, thus reducing

unnecessary computations. This approach automatically adapts to the current optimization state without requiring manually tuned thresholds, reallocates computational resources to the P_{Acp} to enhance convergence speed and solution quality, and preserves the necessary diversity of P_{Sscps} . Overall, the CCTI mechanism significantly improves the efficiency, robustness, and convergence performance of the HMP-CE algorithm in multi-constraint multi-objective optimization scenarios.

Algorithm 3 Constraint Combination Timing Identification (CCTI)

```

1: Input:  $P_{Acp}$ ,  $P_{Ucp}$ , all  $P_{Sscps}$ 
2: Output:  $P_{SscpT}$  (combined  $P_{Sscps}$ )
3: if number of  $P_{Sscps} > 1$  then
4:   if  $P_{Ucp}$  reached UPF then
5:     Compute non-dominated ranks of  $P_{Acp}$  and  $P_{Sscps}$ 
6:     for each  $P_{Sscp}$  do
7:       if  $P_{Acp}$ 's min non-dominated rank  $>$   $P_{Sscp}$ 's max rank then
8:         Mark  $P_{Sscp}$  as ready to combine
9:       end if
10:    end for
11:  end if
12:  for each  $P_{Sscp}$  do
13:    if  $P_{Sscp}$  is stagnated OR duplicated then
14:      Mark  $P_{Sscp}$  as ready to combine
15:    end if
16:    for each unmarked  $P_{Sscp}$  do
17:      if current  $P_{Sscp}$ 's min rank  $>$  unmarked  $P_{Sscp}$ 's max rank then
18:        Mark unmarked  $P_{Sscp}$  to combine
19:      end if
20:    end for
21:  end for
22:   $P_{SscpT} \leftarrow \text{Environmental\_Selection}(\text{all prepared } P_{Sscps})$ 
23:  Delete all prepared  $P_{Sscps}$ 
24:  if only one  $P_{Sscp}$  remains (same as  $P_{Acp}$ ) then
25:    Delete it
26:  end if
27: else
28:   Delete  $P_{Sscp}$ 
29: end if
30: return  $P_{SscpT}$ 
  
```

5. Experimental Studies

All experiments in this study are implemented on the PlatEMO 4.0 platform [63], which provides a standardized and widely used environment for evolutionary multi-objective optimization research. To systematically evaluate the performance of the proposed algorithm, we describe the experimental settings, benchmark test suites, and performance metrics in the subsequent subsections.

5.1. Experimental Setup

Before presenting the experimental results, the experimental setup is briefly described. The study employs multiple benchmark test suites (MW [40], LIRCOP [64], and DASC-MOP [65]) as well as real-world CMOPs [33], with seven state-of-the-art CMOEAs selected for comparison. Algorithm performance is evaluated using IGD, HV, and the Wilcoxon rank-sum test [66], providing a foundation for the subsequent analysis.

5.2. Test Suites

This paper employs the MW, LIRCMOP, DASCOP, and real-world CMOPs test suites. In the test problems, M represents the number of objectives and D represents the decision variables.

For all algorithms, the maximum number of function evaluations (FEs) is strictly limited to 300,000, including the evaluations consumed during initialization. No additional evaluations beyond this budget are allowed. To assess the statistical significance of performance differences among algorithms, Specifically, a Friedman rank test was first performed to examine whether statistically significant differences exist among all compared algorithms. When significant differences were detected, pairwise post hoc comparisons were conducted using the Wilcoxon signed-rank test with Holm's multiple-comparison correction at a significance level of $\alpha = 0.05$. The symbols "+", "-", and " $\approx$ " [67] indicate that the proposed algorithm performs significantly better than, worse than, or statistically similar to the compared algorithm, respectively. In addition, visual analyses of the algorithms on selected test problems are provided to complement the mean $\pm$ standard deviation results. Furthermore, the results marked in red in the tables represent the best performance obtained by the corresponding algorithm for each test problem.

5.3. Competing Algorithms

To comprehensively validate the innovative value and overall performance of HMP-CE, this study selects seven state-of-the-art CMOEAs—namely cDPEA [58], CMOEMT [23], EMCMO [31], CCMO [26], CMOCOS [64], C3M [16], and IMTCMO [14]—for systematic comparative analysis. The core features and technical frameworks of these algorithms are summarized in Table 3. In terms of experimental design, all comparative algorithms strictly follow the parameter settings and operational configurations reported in their original publications, ensuring that each algorithm operates under its optimal conditions. This zero-adjustment comparison strategy not only guarantees fairness and reproducibility but also objectively highlights the remarkable performance advantages demonstrated by HMP-CE.

Table 3. Summary of core innovations, advantages, and disadvantages of seven state-of-the-art CMOEAs.

Algorithm	Core Innovation	Advantages	Disadvantages
cDPEA [58]	Dual-population evolutionary algorithm that alternates between optimizing constrained and unconstrained objectives.	Effectively balances constrained and unconstrained objectives through dual-population cooperation.	Sensitive to initialization and parameter settings; relatively high computational complexity.
CMOEMT [23]	Evolutionary multitasking-based algorithm that utilizes task transfer to enhance optimization efficiency in constrained multi-objective problems.	Strong task information transfer capability improves convergence in multi-objective optimization.	High computational cost; the effectiveness of task transfer is highly dependent on parameter settings.
EMCMO [31]	Evolutionary multitasking-based method for constrained MOP with task information sharing.	Fast optimization process and adaptability to a variety of constrained problems.	Sensitive to task transfer control parameters, leading to potential premature task switching.
CCMO [26]	Coevolutionary framework with two populations handling different objective optimization tasks.	Efficiently solves multi-objective and constrained optimization problems while maintaining population diversity.	Dependent on population initialization; computational efficiency can be low.

Table 3. Cont.

Algorithm	Core Innovation	Advantages	Disadvantages
CMOCSO [64]	Competitive and cooperative swarm optimization algorithm for constrained multi-objective problems.	Balances exploration and exploitation effectively, especially for multi-constrained problems.	Complex parameter selection; high computational cost and sometimes slow convergence.
C3M [16]	Deep reinforcement learning-based multi-objective constrained optimization algorithm with experience replay and stage-based processing.	Dynamic stage switching improves optimization efficiency for complex constrained problems.	Computationally complex, and sensitive to initialization and parameter settings.
IMTCMO [14]	Improved evolutionary multitasking algorithm with neighborhood pairing and multiple selection strategies for efficient search.	Multi-task collaboration enhances convergence and solution quality.	Sensitive to hyperparameter settings, leading to potential premature convergence.

5.4. Performance Metrics

Comparative experiments were conducted on the test suite. The performance of the algorithms was evaluated using two metrics: Inverted Generational Distance (IGD) [68] and Hypervolume (HV) [69]. IGD assesses how close a solution set S is to the true Pareto front P^* , with smaller values indicating better convergence as follows:

$$\text{IGD}(S, P^*) = \frac{1}{|P^*|} \sum_{v \in P^*} \min_{u \in S} \|u - v\|, \quad (10)$$

where $\|\cdot\|$ is typically the Euclidean distance.

HV measures the volume of the objective space dominated by S , reflecting both diversity and coverage, with larger values indicating superior performance:

$$\text{HV}(S, r) = \text{Volume} \left(\bigcup_{u \in S} [u_1, r_1] \times [u_2, r_2] \times \cdots \times [u_m, r_m] \right), \quad (11)$$

Finally, differences between the algorithms were examined using the Wilcoxon rank-sum test [66] at a 0.05 significance level.

5.5. Experimental Results and Analysis

This section evaluates the performance of the proposed HMP-CE algorithm on various benchmark and real-world CMOPs. The test suite includes MW, to assess overall performance under general constraints; DASCMP, to examine the balance among feasibility, convergence, and diversity; LIRCMOP, to test algorithm effectiveness in large infeasible regions; and three real-world CMOPs related to process synthesis and control optimization. By comparing HMP-CE with several state-of-the-art CMOEAs, we analyze its performance in terms of convergence, diversity, and feasibility, and we highlight its strengths and limitations across different constraint types and problem complexities.

5.5.1. Analysis of MW

The comparative results in Tables 4 and 5, together with the solution distributions illustrated in Figure 4, show that HMP-CE performs consistently well on most benchmark problems. Overall, HMP-CE attains favorable IGD results on more than 70% of the MW test instances and achieves relatively high HV values in many cases. These outcomes

suggest that the algorithm provides a reasonable balance between convergence to the constrained Pareto front (CPF) and population diversity. Compared with representative methods such as cDPEA, EMCMO, and CMOEMT, HMP-CE reduces the average IGD by approximately 15–35% and improves HV by around 0.5–1.5%, indicating the effectiveness of its multi-population collaboration and regulatory strategies.

For specific problems, such as MW1, MW3, MW5, and MW7, HMP-CE achieves an IGD improvement of approximately 10–35% over the second-best algorithm, with moderate HV gains observed on MW1 and MW4. On problems with complex or disconnected feasible regions (e.g., MW6 and MW10), HMP-CE maintains relatively stable performance, whereas several compared methods, including cDPEA, EMCMO, CCMO, and C3M, exhibit noticeable fluctuations in IGD or HV. Considering average IGD and HV values as well as win–loss–tie statistics, HMP-CE demonstrates robust performance across most test instances. Compared with existing multi-population CMOEAs (e.g., CMOEMT, IMTCMO, and CMOSCO), HMP-CE shows better adaptability under varying constraint difficulties, reflecting the advantages of multi-population collaboration in decomposing constraint-handling tasks.

Table 4. Statistical analysis (mean $\pm$ standard deviation) of IGD values for eight algorithms evaluated on MW benchmark problems.

Problem	M	D	cDPEA	EMCMO	CCMO	CMOEMT	CMOCSO	C3M	IMTCMO	HMP-CE
MW1	2	15	1.7276e-2 (9.50e-3) –	7.5938e-2 (8.93e-2) –	3.1667e-2 (3.48e-2) –	1.6548e-3 (3.82e-5) –	1.6313e-3 (1.11e-5) $\approx$	9.9104e-2 (1.62e-1) –	1.6996e-3 (3.49e-5) –	1.6245e-3 (1.23e-5)
MW2	2	15	2.4433e-2 (6.96e-3) –	1.6943e-2 (5.06e-3) $\approx$	2.3379e-2 (1.63e-2) $\approx$	3.3066e-2 (2.94e-2) –	5.7430e-2 (3.70e-2) –	8.6435e-2 (3.74e-2) –	5.1360e-2 (3.04e-2) –	1.8871e-2 (4.55e-3)
MW3	2	15	4.9963e-3 (2.46e-4) –	1.0611e-2 (1.89e-3) –	4.8353e-3 (1.99e-4) –	4.8379e-3 (3.89e-4) –	4.9626e-3 (2.46e-4) –	5.4658e-3 (2.63e-4) –	4.7666e-3 (1.29e-4) –	4.5884e-3 (1.24e-4)
MW4	3	15	4.0538e-2 (3.76e-4) +	7.1339e-2 (1.49e-2) –	4.0857e-2 (5.31e-4) $\approx$	4.6515e-2 (1.60e-3) –	4.2417e-2 (4.67e-4) –	5.8653e-2 (2.23e-2) –	4.4187e-2 (6.61e-4) –	4.0946e-2 (4.35e-4)
MW5	2	15	2.8215e-3 (9.66e-3) –	1.2460e-3 (1.10e-3) –	1.6122e-3 (2.23e-3) –	3.9340e-3 (6.35e-3) –	4.1722e-3 (9.66e-4) –	2.3095e-1 (3.28e-1) –	1.6611e-3 (4.86e-4) –	2.1190e-4 (8.93e-5)
MW6	2	15	1.3771e-2 (1.13e-2) +	1.4818e-2 (8.77e-3) +	2.6467e-2 (3.01e-2) $\approx$	4.3815e-2 (3.56e-2) –	2.7525e-1 (2.63e-1) –	4.4763e-1 (1.60e-1) –	1.8294e-1 (2.07e-1) –	2.1917e-2 (1.13e-2)
MW7	2	15	4.2794e-3 (1.92e-4) –	4.5315e-3 (3.25e-4) –	4.6069e-3 (3.02e-4) –	4.5791e-3 (2.99e-4) –	5.8908e-3 (6.31e-4) –	5.3910e-3 (4.07e-4) –	4.3044e-3 (1.83e-4) –	3.9159e-3 (9.24e-5)
MW8	3	15	4.7002e-2 (4.87e-3) $\approx$	4.3670e-2 (1.61e-3) +	5.0501e-2 (7.43e-3) –	5.9961e-2 (2.75e-2) –	1.1235e-1 (5.75e-2) –	1.2460e-1 (4.89e-2) –	6.5687e-2 (1.64e-2) –	4.5375e-2 (2.16e-3)
MW9	2	15	4.5404e-3 (1.53e-4) –	4.4733e-3 (1.73e-4) –	4.4681e-3 (3.76e-4) –	2.1720e-2 (7.11e-2) –	5.0926e-3 (1.83e-4) –	3.6252e-1 (3.47e-1) –	5.4359e-3 (7.35e-4) –	4.0625e-3 (7.57e-5)
MW10	2	15	1.0635e-1 (1.72e-1) –	2.2635e-2 (1.99e-2) $\approx$	4.5964e-2 (3.14e-2) –	5.7938e-2 (5.58e-2) $\approx$	3.9893e-1 (2.63e-1) –	3.2819e-1 (1.59e-1) –	1.8752e-1 (9.11e-2) –	3.3623e-2 (3.32e-2)
MW11	2	15	6.1238e-3 (1.21e-4) –	6.0567e-3 (3.42e-4) –	6.0599e-3 (1.74e-4) –	6.1518e-3 (1.64e-4) –	9.8246e-3 (2.46e-3) –	6.8508e-3 (2.78e-4) –	5.9429e-3 (1.13e-4) $\approx$	5.8804e-3 (8.47e-5)
MW12	2	15	4.8657e-3 (1.47e-4) –	7.4454e-3 (1.18e-2) –	1.0977e-1 (2.57e-1) –	5.1484e-3 (1.58e-4) –	5.3395e-3 (1.29e-4) –	2.7324e-1 (3.38e-1) –	4.7872e-3 (1.44e-4) –	4.5603e-3 (8.22e-5)
MW13	2	15	3.1829e-2 (2.29e-2) +	8.8459e-2 (4.15e-2) –	7.1235e-2 (3.09e-2) $\approx$	7.2309e-2 (6.54e-2) $\approx$	1.4692e-1 (7.17e-2) –	2.0279e-1 (8.14e-2) –	1.5880e-1 (6.36e-2) –	6.0528e-2 (2.97e-2)
MW14	3	15	9.7793e-2 (1.63e-3) $\approx$	9.6560e-2 (1.40e-3) $\approx$	9.7190e-2 (1.79e-3) $\approx$	1.0527e-1 (2.14e-3) –	1.0036e-1 (2.53e-3) –	1.4348e-1 (5.67e-2) –	9.8964e-2 (2.69e-3) $\approx$	9.7429e-2 (1.87e-3)
+ / – / $\approx$			3/9/2	2/9/3	0/9/5	0/12/2	0/13/1	0/14/0	0/12/2	

Table 5. Statistical analysis (mean $\pm$ standard deviation) of HV values for eight algorithms evaluated on MW benchmark problems.

Problem	M	D	cDPEA	EMCMO	CCMO	CMOEMT	CMOCSO	C3M	IMTCMO	HMP-CE
MW1	2	15	4.6262e-1 (1.00e-2) –	4.1098e-1 (7.13e-2) –	4.4920e-1 (3.16e-2) –	4.8967e-1 (1.20e-4) –	4.9001e-1 (3.00e-5) –	3.9999e-1 (1.36e-1) –	4.8953e-1 (1.33e-4) –	4.9014e-1 (7.77e-6)
MW2	2	15	5.4721e-1 (1.03e-2) –	5.5969e-1 (9.16e-3) $\approx$	5.4946e-1 (2.32e-2) $\approx$	5.3628e-1 (4.08e-2) –	5.0186e-1 (4.96e-2) –	4.6296e-1 (4.79e-2) –	5.0974e-1 (4.07e-2) –	5.5583e-1 (7.44e-3)
MW3	2	15	5.4422e-1 (5.33e-4) –	5.3601e-1 (3.41e-3) –	5.4441e-1 (4.35e-4) –	5.4405e-1 (7.18e-4) –	5.4424e-1 (3.50e-4) –	5.4275e-1 (5.12e-4) –	5.4457e-1 (2.42e-4) –	5.4483e-1 (3.28e-4)
MW4	3	15	8.4189e-1 (3.88e-4) +	7.9683e-1 (2.33e-2) –	8.4164e-1 (5.07e-4) $\approx$	8.3436e-1 (2.54e-3) –	8.3913e-1 (6.41e-4) –	8.1118e-1 (3.94e-2) –	8.3508e-1 (9.80e-4) –	8.4135e-1 (5.00e-4)
MW5	2	15	3.2291e-1 (6.38e-3) –	3.2418e-1 (3.53e-4) –	3.2384e-1 (1.45e-3) –	3.2256e-1 (4.02e-3) –	3.2193e-1 (5.68e-4) –	2.3327e-1 (9.79e-2) –	3.2360e-1 (3.44e-4) –	3.2461e-1 (6.70e-5)
MW6	2	15	3.1172e-1 (1.60e-2) +	3.1004e-1 (1.24e-2) +	2.9769e-1 (3.23e-2) $\approx$	2.7789e-1 (3.23e-2) –	1.8722e-1 (9.47e-2) –	1.3056e-1 (5.58e-2) –	2.2534e-1 (6.76e-2) –	3.0009e-1 (1.53e-2)
MW7	2	15	4.1239e-1 (4.52e-4) –	4.1256e-1 (4.03e-4) –	4.1245e-1 (4.89e-4) –	4.1167e-1 (5.04e-4) –	4.1219e-1 (3.95e-4) –	4.1111e-1 (6.59e-4) –	4.1231e-1 (2.05e-4) –	4.1295e-1 (2.33e-4)
MW8	3	15	5.3035e-1 (1.53e-2) $\approx$	5.4402e-1 (9.06e-3) +	5.1895e-1 (1.91e-2) –	5.0708e-1 (5.84e-2) –	3.9881e-1 (9.76e-2) –	3.7623e-1 (8.16e-2) –	4.8415e-1 (3.55e-2) –	5.3554e-1 (1.09e-2)
MW9	2	15	3.9926e-1 (1.54e-3) $\approx$	3.9857e-1 (2.05e-3) –	3.9878e-1 (2.03e-3) –	3.7932e-1 (7.33e-2) –	3.9755e-1 (1.65e-3) –	1.8432e-1 (1.89e-1) –	3.9523e-1 (1.49e-3) –	4.0027e-1 (1.33e-3)
MW10	2	15	3.8149e-1 (8.73e-2) –	4.3202e-1 (1.85e-2) $\approx$	4.1170e-1 (2.33e-2) –	4.0650e-1 (3.94e-2) $\approx$	2.2822e-1 (1.26e-1) –	2.5608e-1 (7.21e-2) –	3.2657e-1 (5.14e-2) –	4.2334e-1 (2.61e-2)
MW11	2	15	4.4755e-1 (2.16e-4) –	4.4714e-1 (1.32e-3) –	4.4747e-1 (2.52e-4) –	4.4744e-1 (1.82e-4) –	4.4376e-1 (1.42e-3) –	4.4660e-1 (3.16e-4) –	4.4777e-1 (1.50e-4) –	4.4823e-1 (8.18e-5)
MW12	2	15	6.0472e-1 (2.79e-4) –	6.0207e-1 (1.17e-2) –	5.1618e-1 (2.17e-1) –	6.0378e-1 (3.34e-4) –	6.0336e-1 (1.99e-4) –	3.7415e-1 (2.83e-1) –	6.0465e-1 (3.51e-4) –	6.0553e-1 (1.41e-4)
MW13	2	15	4.6494e-1 (1.13e-2) +	4.4265e-1 (1.77e-2) $\approx$	4.4534e-1 (1.49e-2) $\approx$	4.4259e-1 (3.82e-2) $\approx$	3.9801e-1 (4.66e-2) –	3.6016e-1 (4.82e-2) –	3.8830e-1 (4.14e-2) –	4.5019e-1 (1.40e-2)
MW14	3	15	4.7297e-1 (1.12e-3) –	4.7586e-1 (1.70e-3) +	4.7368e-1 (1.46e-3) $\approx$	4.6807e-1 (2.03e-3) –	4.6969e-1 (2.19e-3) –	4.4785e-1 (1.39e-2) –	4.6963e-1 (1.47e-3) –	4.7392e-1 (1.32e-3)
+ / – / $\approx$			3/9/2	3/8/3	0/9/5	0/12/2	0/14/0	0/14/0	0/14/0	

The detailed results in Tables 4 and 5 further indicate that HMP-CE achieves relatively small IGD and relatively large HV values on most MW problems. These results can be attributed to three main components as follows: (1) the multi-population collaboration strategy that enables parallel searches under different levels of constraint relaxation; (2) the PADR mechanism that adaptively regulates the activation and dormancy of auxiliary populations to maintain search efficiency; and (3) the CCTI mechanism that dynamically identifies suitable timings for constraint combination, enhancing communication and convergence alignment among populations.

Figure 4 presents the solution distributions on MW7. The gray area denotes the feasible region, the gray curve indicates the CPF, and cyan dots represent feasible solutions. The solutions generated by HMP-CE (Figure 4h) are distributed relatively continuously along the feasible boundary. In contrast, some algorithms (e.g., cDPEA, EMCMO, CCMO, and IMTCMO) show gaps or deviations in certain segments. These observations illustrate the differences in constraint-handling behavior among algorithms. Overall, the results in Tables 4 and 5 and Figure 4 confirm that the integration of multi-population collaboration, PADR control, and the CCTI mechanism enables HMP-CE to maintain reliable performance in terms of convergence, solution distribution, and stability on a variety of constrained multi-objective optimization problems.

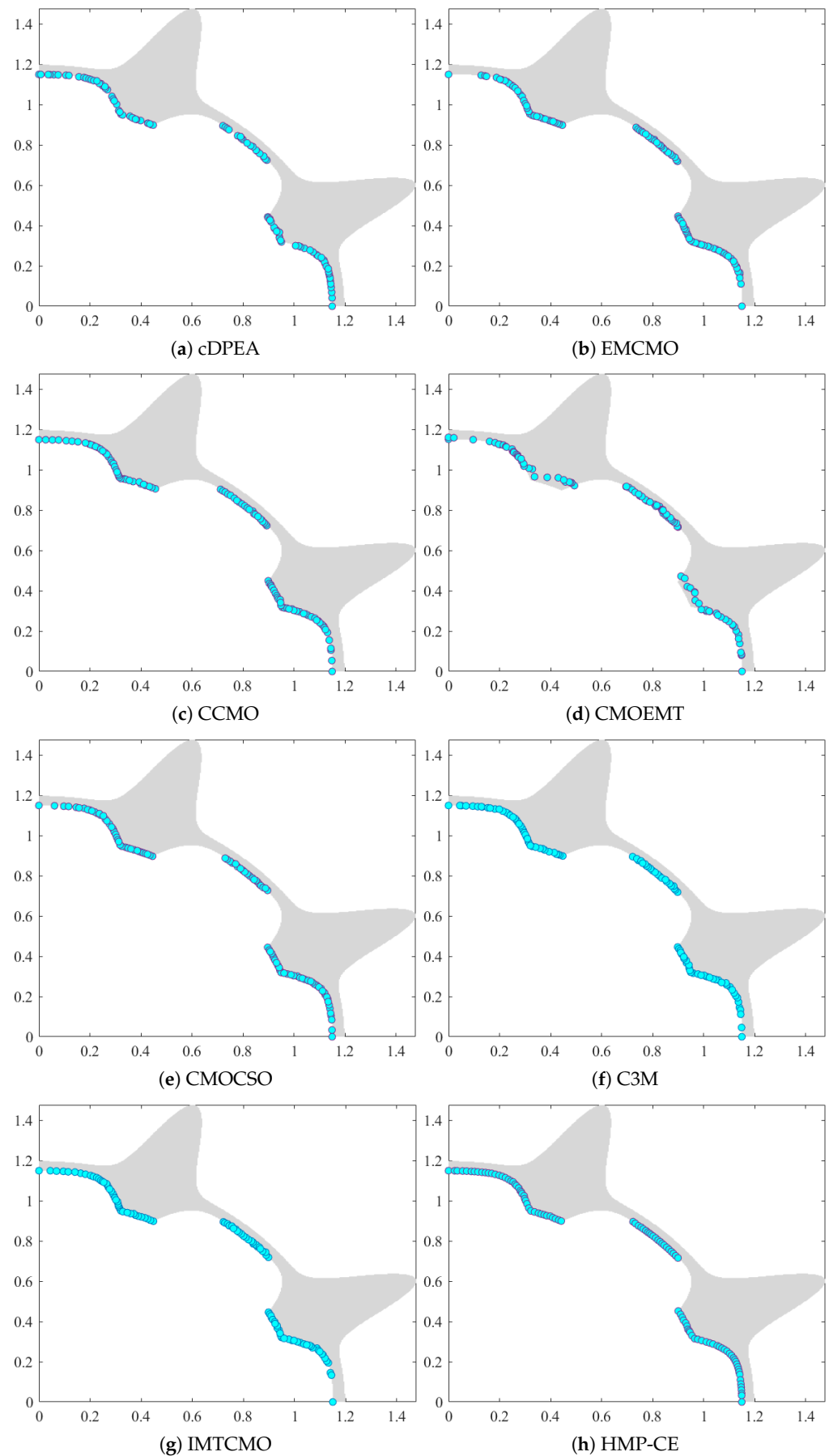


Figure 4. Final solution distributions of nine CMOEAs on the two-objective MW7 problem.

5.5.2. Analysis of LIRCMOP

Tables 6 and 7 present the IGD and HV statistics of the eight algorithms on the LIRCMOP benchmark problems. Overall, HMP-CE achieves relatively low IGD values and comparatively high HV values in most test instances, indicating that it can reasonably approximate the constrained Pareto front (CPF) while maintaining a fairly balanced solution distribution. For LIRCMOP5–LIRCMOP12, HMP-CE generally outperforms other algorithms in terms of IGD, while its HV values are comparable to or slightly higher than those of the other algorithms. For LIRCMOP13 and LIRCMOP14, its performance remains stable and is similar to that of other multi-population algorithms such as CMOEMT.

Table 6. Statistical analysis (mean $\pm$ standard deviation) of IGD values for eight algorithms evaluated on LIRCMOP benchmark problems.

Problem	M	D	cDPEA	EMCMO	CCMO	CMOEMT	CMOCSO	C3M	IMTCMO	HMP-CE
LIRCMOP1	2	30	1.5622e-1 (4.39e-2) –	2.3288e-1 (4.60e-2) –	2.5848e-1 (5.34e-2) –	4.7815e-2 (2.99e-2) $\approx$	5.9869e-2 (3.75e-2) $\approx$	2.0545e-1 (8.98e-2) –	2.3985e-2 (4.38e-3) +	6.9811e-2 (4.79e-2)
LIRCMOP2	2	30	1.3829e-1 (3.30e-2) –	2.4610e-1 (2.45e-2) –	2.1978e-1 (4.49e-2) –	2.7610e-1 (3.25e-2) –	4.0007e-1 (1.01e-1) –	1.7847e-1 (5.69e-2) –	1.9747e-2 (3.57e-3) $\approx$	6.4680e-2 (5.54e-2)
LIRCMOP3	2	30	1.5949e-1 (3.45e-2) –	2.5860e-1 (5.49e-2) –	2.7026e-1 (6.21e-2) –	6.1201e-2 (5.92e-2) $\approx$	5.0623e-2 (1.90e-2) $\approx$	1.6415e-1 (9.09e-2) –	2.0543e-2 (9.04e-3) +	7.6537e-2 (6.58e-2)
LIRCMOP4	2	30	1.7558e-1 (5.46e-2) –	2.7334e-1 (3.27e-2) –	2.9019e-1 (2.88e-2) –	4.7700e-2 (4.25e-2) +	4.5458e-2 (2.65e-2) +	1.8505e-1 (1.00e-1) –	1.1566e-2 (5.11e-3) +	9.7345e-2 (6.02e-2)
LIRCMOP5	2	30	3.0390e-1 (5.90e-2) –	3.1727e-1 (5.32e-2) –	2.8688e-1 (6.56e-2) –	3.2196e-1 (3.20e-1) –	1.2209e-2 (6.72e-3) –	1.7367e-1 (3.43e-1) –	6.4903e-1 (5.30e-1) –	6.8854e-3 (8.09e-4)
LIRCMOP6	2	30	3.7341e-1 (1.21e-1) –	2.8548e-1 (7.07e-2) –	2.6155e-1 (6.37e-2) –	2.1061e-1 (1.36e-1) –	3.1517e-2 (4.81e-2) –	6.3522e-2 (1.10e-1) –	5.2253e-1 (5.75e-1) –	6.6258e-3 (4.87e-4)
LIRCMOP7	2	30	1.0948e-1 (1.86e-2) –	1.2105e-1 (2.57e-2) –	1.2033e-1 (3.03e-2) –	6.2006e-2 (3.00e-2) –	7.9591e-2 (5.10e-2) –	1.5966e-1 (6.11e-1) –	3.5994e-2 (4.08e-2) –	7.1511e-3 (2.15e-4)
LIRCMOP8	2	30	1.8964e-1 (5.18e-2) –	1.7527e-1 (4.95e-2) –	1.6586e-1 (6.28e-2) –	8.9353e-2 (2.59e-2) –	4.9171e-2 (7.55e-2) –	1.5643e-2 (1.91e-2) –	1.4771e-2 (3.28e-2) –	7.1038e-3 (1.88e-4)
LIRCMOP9	2	30	4.7200e-1 (1.11e-1) –	5.3041e-1 (1.06e-1) –	5.3194e-1 (1.64e-1) –	3.7303e-1 (6.57e-2) –	4.4297e-1 (8.26e-2) –	2.4845e-1 (1.75e-1) –	3.7269e-1 (7.93e-2) –	6.2494e-3 (1.70e-3)
LIRCMOP10	2	30	3.1664e-1 (8.42e-2) –	1.7367e-1 (6.46e-2) –	1.4832e-1 (3.40e-2) –	2.4594e-1 (9.94e-2) –	1.3717e-1 (8.16e-2) –	1.0377e-1 (8.46e-2) –	1.4143e-2 (5.76e-3) –	6.1956e-3 (4.16e-4)
LIRCMOP11	2	30	1.1978e-1 (3.65e-2) –	5.9213e-2 (3.30e-2) –	7.6889e-2 (1.55e-2) –	7.6578e-2 (5.48e-2) –	6.6032e-2 (6.30e-2) –	9.2677e-2 (1.03e-1) –	1.1673e-1 (8.17e-2) –	2.6448e-3 (9.64e-5)
LIRCMOP12	2	30	1.8945e-1 (5.39e-2) –	2.2358e-1 (9.63e-2) –	2.0535e-1 (8.72e-2) –	1.1447e-1 (5.73e-2) –	1.6359e-1 (9.56e-2) –	1.0641e-1 (5.77e-2) –	1.8787e-1 (6.32e-2) –	3.5602e-3 (3.09e-4)
LIRCMOP13	3	30	9.2764e-2 (1.10e-3) +	9.0530e-2 (7.89e-4) +	9.3954e-2 (1.51e-3) +	1.0564e-1 (3.19e-3) +	9.6106e-2 (1.61e-3) +	1.2416e-1 (7.44e-3) –	1.3104e+0 (1.26e-3) –	1.1872e-1 (2.37e-3)
LIRCMOP14	3	30	9.5299e-2 (6.09e-4) +	9.5491e-2 (9.86e-4) +	9.5716e-2 (8.22e-4) +	1.0235e-1 (1.25e-3) +	9.6576e-2 (8.15e-4) +	1.5621e-1 (2.01e-1) –	1.2671e+0 (1.47e-3) –	1.0316e-1 (1.32e-3)
+ / – / $\approx$			2/12/0	2/12/0	2/12/0	3/9/2	3/9/2	0/14/0	3/10/1	

The visualization results in Figure 5 provide a geometric perspective on the algorithm's behavior. The gray area represents the feasible region, red dots represent feasible solutions, and the boundary denotes the constrained Pareto front (CPF). The feasible solutions obtained by HMP-CE (Figure 5h) are distributed relatively continuously along the feasible boundary, without obvious gaps or clustering. In contrast, some of the compared algorithms (C3M, CMOCSO, IMTCMO and CMOEMT) show discontinuities or deviations in certain front segments, particularly in narrow or disconnected feasible regions. This indi-

cates that HMP-CE maintains relatively stable exploration near constraint boundaries and can reasonably balance feasibility with objective optimization, providing fairly complete coverage of the feasible Pareto front.

Table 7. Statistical analysis (mean $\pm$ standard deviation) of HV values for eight algorithms evaluated on LIRCMOP benchmark problems.

Problem	M	D	cDPEA	EMCMO	CCMO	CMOEMT	CMOCSO	C3M	IMTCMO	HMP-CE
LIRCMOP1	2	30	1.5982e-1 (1.51e-2) −	1.3494e-1 (1.43e-2) −	1.2572e-1 (1.75e-2) −	2.1949e-1 (1.44e-2) +	2.1296e-1 (1.24e-2) $\approx$	1.4515e-1 (3.25e-2) −	2.2854e-1 (3.25e-3) +	2.0002e-1 (2.36e-2)
LIRCMOP2	2	30	2.8907e-1 (1.59e-2) −	2.3555e-1 (1.40e-2) −	2.4755e-1 (2.37e-2) −	2.2134e-1 (1.42e-2) −	1.6113e-1 (6.17e-2) −	2.4836e-1 (2.89e-2) −	3.5245e-1 (1.88e-3) +	3.2387e-1 (3.19e-2)
LIRCMOP3	2	30	1.4637e-1 (1.34e-2) −	1.1414e-1 (1.85e-2) −	1.1151e-1 (1.92e-2) −	1.8431e-1 (2.28e-2) $\approx$	1.8789e-1 (8.02e-3) $\approx$	1.4315e-1 (2.93e-2) −	2.0006e-1 (3.57e-3) +	1.7652e-1 (2.37e-2)
LIRCMOP4	2	30	2.4070e-1 (2.66e-2) −	2.0024e-1 (1.49e-2) −	1.9531e-1 (1.17e-2) −	2.9476e-1 (1.81e-2) +	2.9727e-1 (8.50e-3) +	2.4049e-1 (3.86e-2) −	3.1063e-1 (1.84e-3) +	2.7300e-1 (2.70e-2)
LIRCMOP5	2	30	1.5340e-1 (2.30e-2) −	1.4797e-1 (2.06e-2) −	1.5918e-1 (2.55e-2) −	1.6435e-1 (7.95e-2) −	2.8782e-1 (3.66e-3) −	2.4172e-1 (8.50e-2) −	1.1868e-1 (1.20e-1) −	2.9111e-1 (3.78e-4)
LIRCMOP6	2	30	1.0611e-1 (1.44e-2) −	1.1933e-1 (1.22e-2) −	1.1573e-1 (1.54e-2) −	1.4405e-1 (3.27e-2) −	1.8955e-1 (1.26e-2) −	1.7533e-1 (3.82e-2) −	1.0940e-1 (8.03e-2) −	1.9636e-1 (2.52e-4)
LIRCMOP7	2	30	2.5174e-1 (5.76e-3) −	2.4773e-1 (7.50e-3) −	2.4817e-1 (9.41e-3) −	2.6799e-1 (1.25e-2) −	2.6406e-1 (1.99e-2) −	2.7268e-1 (6.56e-2) −	2.8104e-1 (1.77e-2) −	2.9443e-1 (1.71e-4)
LIRCMOP8	2	30	2.3419e-1 (1.01e-2) −	2.3550e-1 (1.07e-2) −	2.3911e-1 (1.57e-2) −	2.6025e-1 (8.07e-3) −	2.7958e-1 (2.48e-2) $\approx$	2.8977e-1 (9.29e-3) −	2.9175e-1 (1.18e-2) $\approx$	2.9445e-1 (1.69e-4)
LIRCMOP9	2	30	3.8957e-1 (7.05e-2) −	3.5549e-1 (6.02e-2) −	3.5114e-1 (6.47e-2) −	4.2463e-1 (2.74e-2) −	3.9126e-1 (4.35e-2) −	4.6459e-1 (7.68e-2) −	4.1788e-1 (3.79e-2) −	5.6420e-1 (9.69e-4)
LIRCMOP10	2	30	5.3775e-1 (4.93e-2) −	6.1075e-1 (3.57e-2) −	6.2694e-1 (1.77e-2) −	5.6266e-1 (5.72e-2) −	6.4430e-1 (3.77e-2) −	6.6149e-1 (4.01e-2) −	7.0045e-1 (3.25e-3) −	7.0614e-1 (3.18e-4)
LIRCMOP11	2	30	6.3411e-1 (2.45e-2) −	6.6294e-1 (1.88e-2) −	6.5916e-1 (9.79e-3) −	6.4868e-1 (3.65e-2) −	6.5457e-1 (4.12e-2) −	6.3933e-1 (7.19e-2) −	6.2256e-1 (5.24e-2) −	6.9391e-1 (3.23e-5)
LIRCMOP12	2	30	5.3360e-1 (2.84e-2) −	5.0963e-1 (5.24e-2) −	5.2168e-1 (4.19e-2) −	5.6398e-1 (2.76e-2) −	5.4102e-1 (4.89e-2) −	5.6932e-1 (3.06e-2) −	5.2256e-1 (3.69e-2) −	6.2018e-1 (1.20e-4)
LIRCMOP13	3	30	5.5520e-1 (1.38e-3) +	5.5975e-1 (9.95e-4) +	5.5448e-1 (1.84e-3) +	5.3575e-1 (6.41e-3) +	5.4838e-1 (1.31e-3) +	5.0294e-1 (8.79e-3) −	1.0610e-4 (1.32e-4) −	5.1063e-1 (2.97e-3)
LIRCMOP14	3	30	5.5466e-1 (1.24e-3) +	5.5478e-1 (1.12e-3) +	5.5382e-1 (1.63e-3) +	5.4738e-1 (2.37e-3) +	5.5230e-1 (1.15e-3) +	5.0638e-1 (1.05e-1) −	4.9798e-4 (3.65e-4) −	5.3960e-1 (1.64e-3)
+ / − / $\approx$			2/12/0	2/12/0	2/12/0	4/9/1	3/8/3	0/14/0	4/9/1	

These observations also reflect the role of the three mechanisms in HMP-CE. The multi-population collaboration mechanism allows different populations to explore under varying levels of constraint relaxation, which helps global search while maintaining feasible solution quality. The Population Activation–Dormancy Regulation (PADR) mechanism adjusts auxiliary populations dynamically based on their contributions, reducing redundant computation and limiting the risk of search stagnation. The Constraint Combination Timing Identification (CCTI) mechanism adaptively determines when to merge constraints across populations, supporting better inter-population coordination. Overall, these mechanisms help HMP-CE maintain stable convergence and solution distribution when addressing complex constrained multi-objective optimization problems.

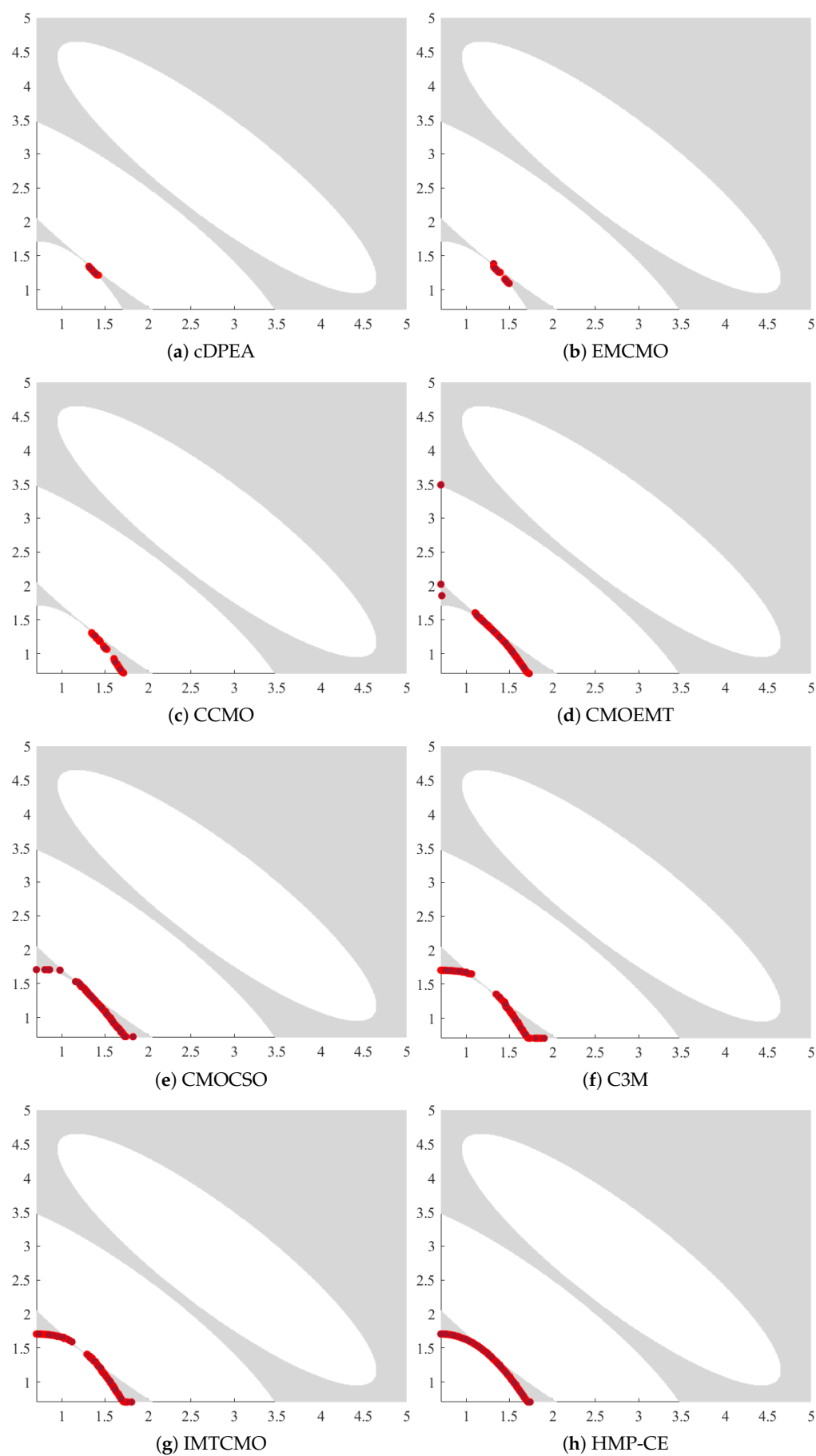


Figure 5. Final solution distributions of eight CMOEAs on the two-objective LIRCMOP6 problem.

5.5.3. Analysis of DASC-MOP

Tables 8 and 9 present the IGD and HV statistics of the eight algorithms on the DASC-MOP test problems. Overall, HMP-CE achieves relatively low IGD values in most instances, while maintaining stable HV performance, indicating that it can reasonably approximate the constrained Pareto front (CPF) and preserve a moderately balanced solution set. For DASC-MOP1–5, HMP-CE generally performs better in terms of IGD, while its HV values are comparable to or slightly higher than some other algorithms. In the three-objective problems DASC-MOP7–9, the HV performance of HMP-CE is relatively consistent and similar to that of other multi-population algorithms such as CMOCSSO and IMTCMO.

Table 8. Statistical analysis (mean $\pm$ standard deviation) of IGD values for eight algorithms evaluated on DASC-MOP benchmark problems.

Problem	M	D	cDPEA	EMCMO	CCMO	CMOEMT	CMOCSSO	C3M	IMTCMO	HMP-CE
DASC-MOP1	2	30	6.7108e-1 (7.58e-2) –	7.1773e-1 (3.23e-2) –	7.2474e-1 (3.06e-2) –	4.0125e-1 (3.19e-1) –	5.7240e-3 (2.04e-3) –	5.5336e-3 (1.85e-3) –	3.1305e-3 (3.19e-4) –	2.9796e-3 (5.18e-4)
DASC-MOP2	2	30	2.2569e-1 (2.39e-2) –	2.4442e-1 (2.30e-2) –	2.4441e-1 (1.96e-2) –	1.0966e-1 (1.07e-1) –	4.5246e-3 (1.38e-4) –	4.8250e-3 (2.23e-4) –	1.9998e-1 (7.03e-2) –	4.0671e-3 (1.00e-4)
DASC-MOP3	2	30	3.1877e-1 (4.10e-2) –	3.4172e-1 (4.71e-2) –	3.3656e-1 (2.44e-2) –	1.2122e-1 (1.38e-1) –	3.6444e-2 (7.36e-2) –	8.2866e-2 (9.41e-2) $\approx$	2.7554e-1 (1.21e-1) –	1.9430e-2 (1.61e-4)
DASC-MOP4	2	30	1.9903e-3 (1.43e-3) –	1.6088e-3 (1.06e-3) $\approx$	2.1310e-3 (1.37e-3) $\approx$	5.0192e-3 (6.26e-3) –	8.7676e-3 (3.81e-3) –	7.9915e-2 (1.53e-1) –	9.3346e-2 (1.30e-1) –	1.2955e-3 (1.72e-4)
DASC-MOP5	2	30	8.2715e-1 (2.93e-1) –	8.8217e-1 (1.29e-1) –	7.7032e-1 (1.29e-1) –	3.9734e-3 (1.10e-3) –	6.9522e-3 (8.15e-4) –	8.9512e-3 (2.63e-3) –	1.2122e-2 (2.42e-2) –	3.0375e-3 (1.16e-4)
DASC-MOP6	2	30	2.5407e-2 (1.37e-2) $\approx$	2.7550e-2 (2.72e-2) –	4.9159e-2 (7.01e-2) –	5.7855e-2 (7.28e-2) –	2.0725e-2 (2.95e-3) +	6.6633e-2 (1.79e-1) –	1.7547e-1 (1.82e-1) –	2.1733e-2 (8.31e-3)
DASC-MOP7	3	30	9.3511e-1 (2.15e-1) –	1.0284e+0 (2.67e-1) –	9.4867e-1 (2.59e-1) –	4.7657e-1 (2.82e-1) –	8.4130e-1 (3.32e-1) –	7.3819e-2 (2.55e-2) +	4.4460e-1 (5.84e-1) $\approx$	1.0079e-1 (2.78e-1)
DASC-MOP8	3	30	9.0413e-2 (2.23e-1) +	5.0989e-2 (2.79e-2) +	4.2734e-2 (7.72e-3) +	1.8616e-1 (2.57e-1) –	5.7052e-2 (5.98e-3) +	1.1231e-1 (1.18e-1) +	4.2366e-2 (3.27e-3) +	1.7196e-1 (3.49e-1)
DASC-MOP9	3	30	2.2452e-1 (8.85e-2) –	3.6983e-1 (5.05e-2) –	3.4853e-1 (8.53e-2) –	1.7147e-1 (1.31e-1) –	4.4296e-2 (1.90e-3) –	4.3025e-2 (1.99e-3) –	6.5819e-1 (2.50e-1) –	4.0449e-2 (7.68e-4)
+ / – / $\approx$			1/7/1	1/7/1	1/7/1	0/9/0	2/7/0	2/6/1	1/7/1	

The visualization results in Figure 6 show feasible solutions as green dots, the feasible domain as gray regions, and infeasible areas as blank regions. HMP-CE produces feasible solutions that are relatively uniformly distributed along the feasible boundary and cover a broad area. In comparison, some algorithms (e.g., EMCMO, CCMO, and CMOEMT) exhibit partial clustering or less complete coverage, and C3M, IMTCMO, and CMOCSSO show relatively limited coverage. These observations indicate that algorithms differ in constraint-handling behavior, and HMP-CE demonstrates a moderate advantage in solution distribution and boundary exploration, without implying absolute superiority. In addition, to more clearly demonstrate the capability of the HMP-CE algorithm, Figure 7 presents boxplots of the eight algorithms on the DASC-MOP1–8 test problems. It is evident that, in terms of both mean performance and variability, the HMP-CE algorithm outperforms the seven competing algorithms on most of the test problems.

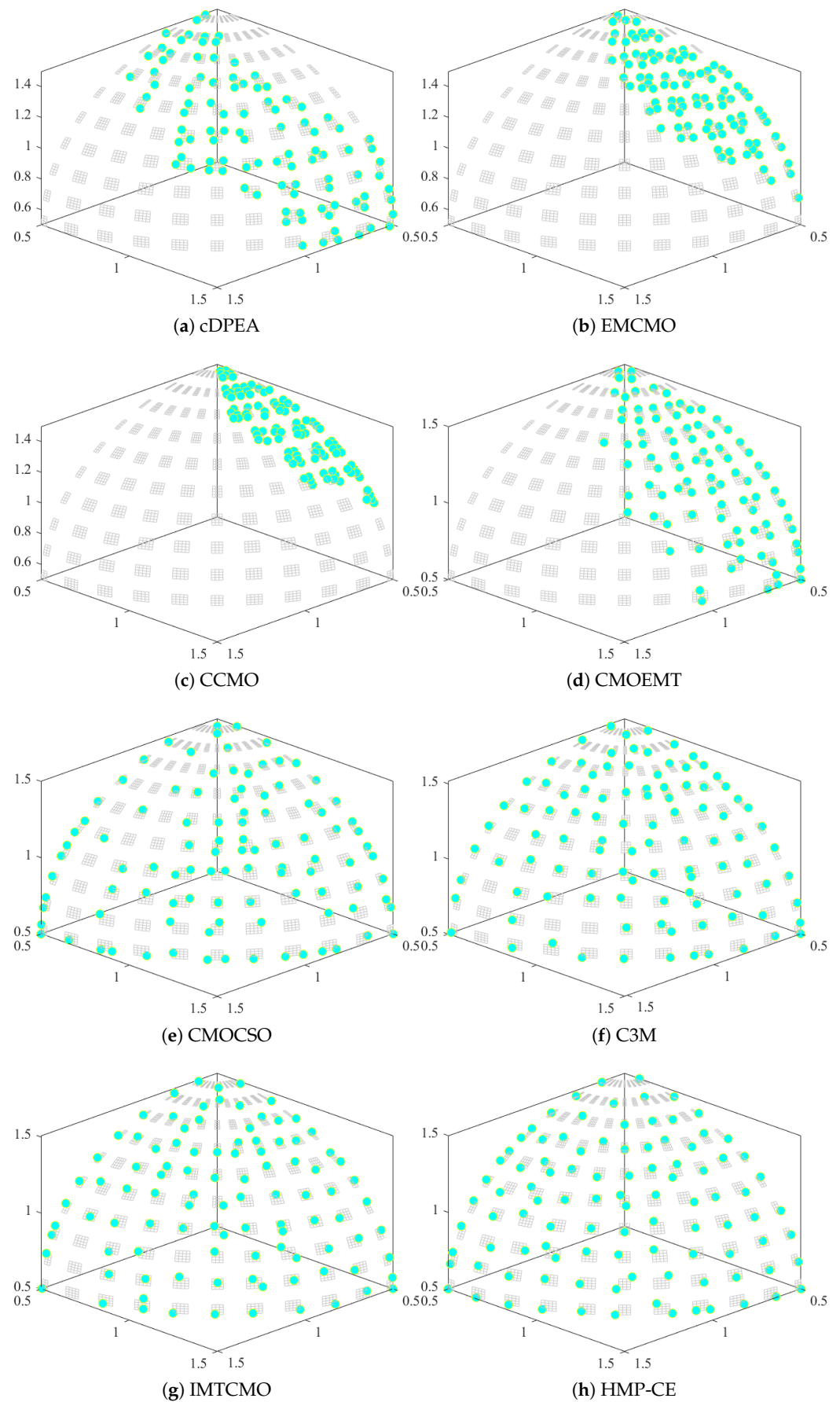


Figure 6. Final solution distributions of eight CMOEAs on the three-objective DASCMP9 problem.

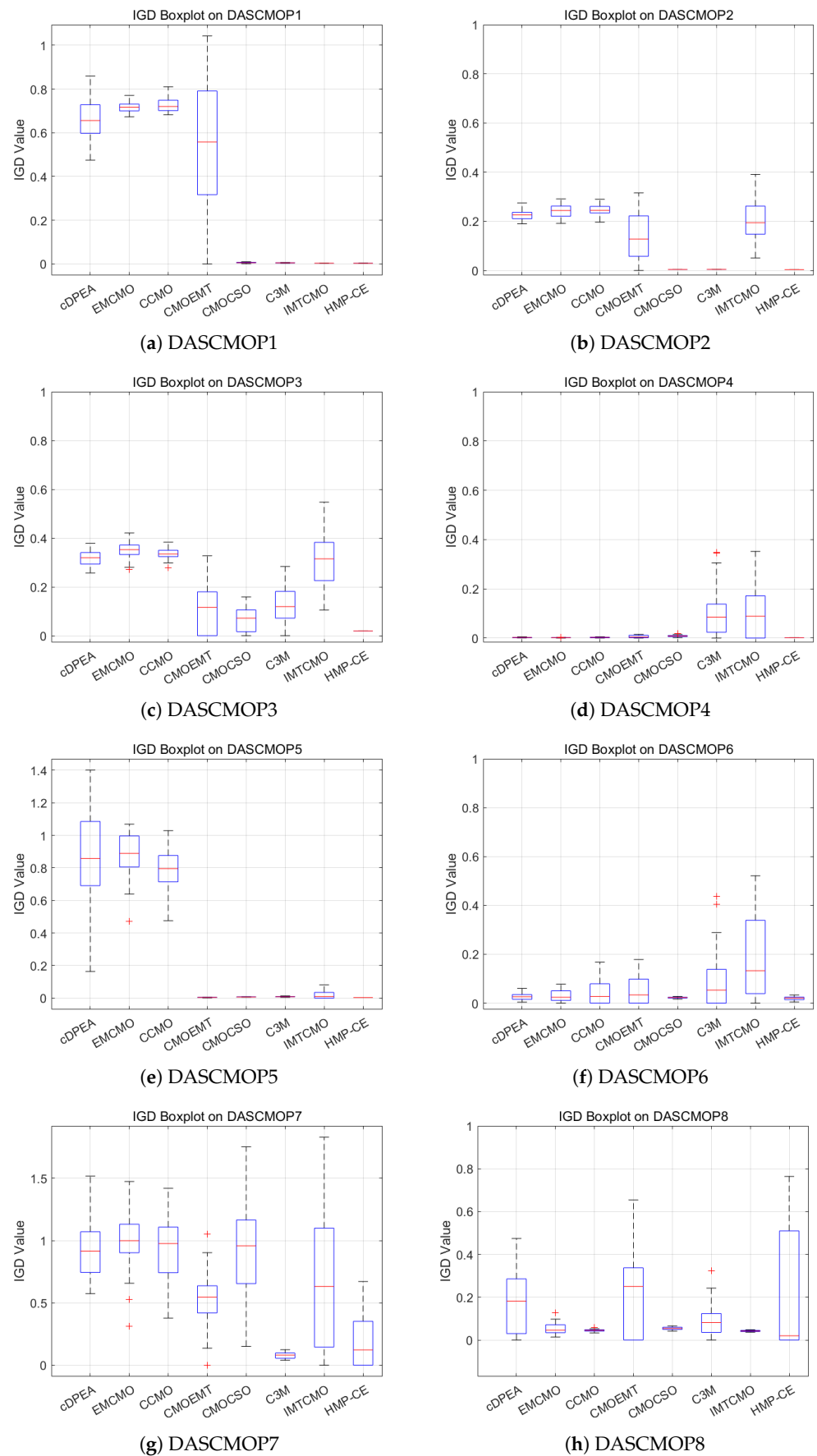


Figure 7. IGD Boxplots of eight CMOEAs on DASCMP1–8 problems.

Table 9. Statistical analysis (mean $\pm$ standard deviation) of HV values for eight algorithms evaluated on DASCMP benchmark problems.

Problem	M	D	cDPEA	EMCMO	CCMO	CMOEMT	CMOCSO	C3M	IMTCMO	HMP-CE
DASCMP1	2	30	1.9658e-2 (2.20e-2) –	8.9905e-3 (6.04e-3) –	1.2825e-2 (1.84e-2) –	9.7700e-2 (8.83e-2) –	2.1138e-1 (1.19e-3) –	2.1090e-1 (1.17e-3) –	2.1236e-1 (5.34e-4) $\approx$	2.1252e-1 (4.61e-4)
DASCMP2	2	30	2.6104e-1 (2.80e-3) –	2.5855e-1 (2.67e-3) –	2.5769e-1 (2.25e-3) –	3.1200e-1 (4.25e-2) –	3.5496e-1 (1.17e-4) –	3.5465e-1 (2.05e-4) –	2.7345e-1 (2.90e-2) –	3.5539e-1 (8.13e-5)
DASCMP3	2	30	2.1872e-1 (1.79e-2) –	2.1439e-1 (1.49e-2) –	2.1275e-1 (1.37e-2) –	2.7898e-1 (4.46e-2) –	3.0597e-1 (2.22e-2) –	2.9819e-1 (2.13e-2) –	2.3286e-1 (3.80e-2) –	3.1238e-1 (9.69e-5)
DASCMP4	2	30	2.0231e-1 (3.69e-3) $\approx$	2.0292e-1 (3.05e-3) $\approx$	2.0165e-1 (3.95e-3) $\approx$	1.9829e-1 (4.72e-3) –	1.9645e-1 (4.57e-3) –	1.6953e-1 (5.89e-2) –	1.6673e-1 (4.95e-2) –	2.0403e-1 (2.49e-4)
DASCMP5	2	30	3.1704e-2 (9.62e-2) –	9.4930e-3 (9.71e-3) –	2.2908e-2 (1.97e-2) –	3.5023e-1 (9.66e-4) –	3.4757e-1 (4.79e-4) –	3.4610e-1 (2.01e-3) –	3.4452e-1 (1.71e-2) –	3.5104e-1 (1.58e-4)
DASCMP6	2	30	3.0849e-1 (7.14e-3) –	3.0778e-1 (1.09e-2) –	2.9852e-1 (2.26e-2) –	2.9583e-1 (2.27e-2) –	3.0992e-1 (9.20e-4) –	2.9127e-1 (6.77e-2) –	2.2342e-1 (1.02e-1) –	3.1065e-1 (5.28e-3)
DASCMP7	3	30	9.5343e-3 (8.27e-3) –	1.1886e-2 (2.44e-2) –	1.6336e-2 (2.11e-2) –	1.1868e-1 (7.45e-2) –	3.8971e-2 (6.37e-2) –	2.6345e-1 (1.42e-2) –	1.4527e-1 (1.98e-1) $\approx$	2.6708e-1 (6.29e-2)
DASCMP8	3	30	1.9608e-1 (4.62e-2) +	2.0307e-1 (9.02e-3) +	2.0519e-1 (4.29e-3) +	1.6347e-1 (5.83e-2) $\approx$	2.0044e-1 (1.11e-3) $\approx$	1.7400e-1 (4.26e-2) –	2.0544e-1 (9.12e-4) +	1.7422e-1 (6.38e-2)
DASCMP9	3	30	1.5500e-1 (1.90e-2) –	1.2724e-1 (1.06e-2) –	1.3106e-1 (1.31e-2) –	1.7283e-1 (3.13e-2) –	2.0518e-1 (5.49e-4) $\approx$	2.0380e-1 (7.17e-4) –	5.7647e-2 (5.26e-2) –	2.0527e-1 (4.24e-4)
+ / – / $\approx$			1/7/1	1/7/1	1/7/1	0/8/1	0/7/2	0/9/0	1/6/2	

The performance of HMP-CE may be related to its multi-population collaboration mechanism, where populations explore under different levels of constraint relaxation, and the PADR and CCTI mechanisms, which help regulate population activity and constraint combination. Overall, HMP-CE shows stable performance and reasonably balanced solution distribution on the DASCMP benchmarks, reflecting its applicability to complex constrained multi-objective optimization problems.

5.5.4. Friedman Ranking of CMOEAs

The Figures 8 and 9 provided present the Friedman ranking results for multiple algorithms, measured using both IGD and HV metrics. Let us break down the analysis of the results, especially highlighting the innovation and superiority of HMP-CE as follows:

- (1) Figure 8—Average Friedman Rank (IGD): HMP-CE stands out with the lowest average Friedman rank (2.14), indicating its strong performance in minimizing IGD. This suggests that HMP-CE excels at providing a solution set that is both diverse and close to the true Pareto front compared to other algorithms. The HMP-CE's ranking is considerably lower than the others, with the next best algorithms (CMOCSO, CCMO, etc.) ranking from 4.46 to 5.38. This demonstrates that HMP-CE is significantly more efficient at exploring the solution space.
- (2) Figure 9—Average Friedman Rank (HV): HMP-CE's rank in HV is similarly competitive, coming in first (6.95). It shows that HMP-CE performs well in maintaining a good balance between convergence and diversity.
- (3) Mechanism characteristics of HMP-CE: HMP-CE incorporates a multi-population collaboration mechanism, allowing different populations to search in parallel and share information, which can help in exploring complex constrained search spaces. Its relatively low average ranks in both IGD and HV suggest a balanced behavior between exploration and exploitation. Overall, HMP-CE shows stable performance in terms of solution quality (IGD) and coverage (HV) compared with other methods.

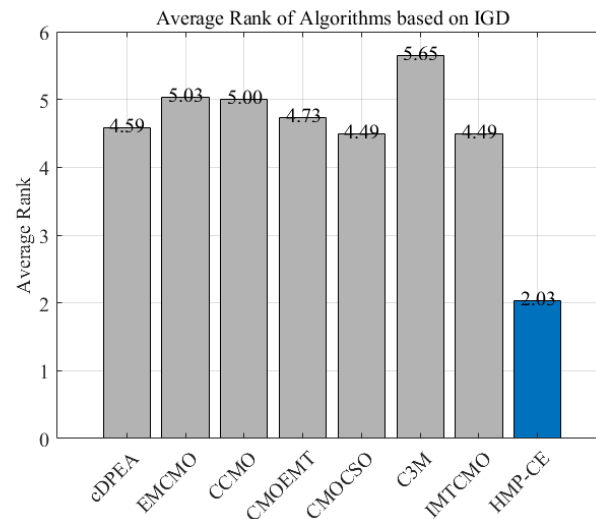


Figure 8. Friedman (IGD) ranking of eight CMOEAs.

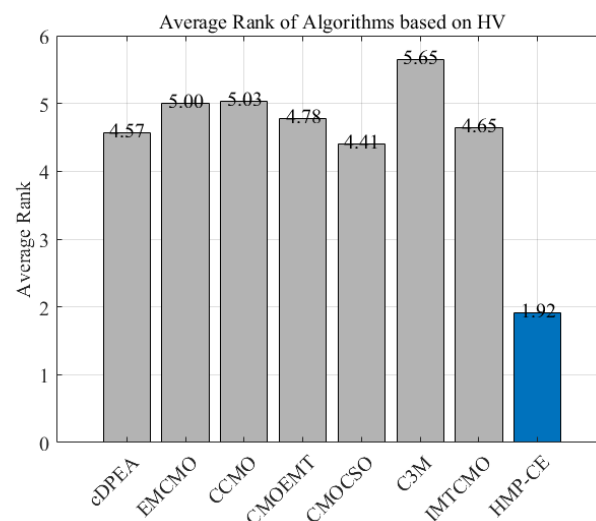


Figure 9. Friedman (HV) Ranking of eight CMOEAs.

Based on the above theoretical analysis, these results indicate that HMP-CE can generate relatively stable solution sets, and its multi-population framework maintains a certain balance between exploration and convergence, without implying absolute superiority over other algorithms in all aspects.

5.6. Analysis of Real-World CMOP

Based on the results in Table 10, the outcomes of the eight algorithms on six real-world engineering problems (PVL, VPF, TBTD, WBD, DBD, and SRDP) can be summarized as follows. Since the true Pareto front is unknown, the HV indicator is typically used to evaluate algorithm performance, with higher HV values indicating broader coverage and better-quality solution sets. Overall, HMP-CE achieves the highest or near-highest HV values in most problems, such as PVL, VPF, TBTD, WBD, DBD, and SRDP. The algorithm generally produces stable and reasonably distributed solution sets, though differences among the algorithms are not substantial. In addition, the visualization analysis of the eight CMOEAs on the DBD real-world problem is shown in Figure 10. It is evident that, compared with the other seven CMOEAs, HME achieves a more uniform and denser distribution, which demonstrates its strong capability in handling real-world problems.

As illustrated in Figure 11, for PVL (a), VPF (b), TBTD (c), WBD (d), DBD (e), and SRDP (f), HMP-CE generally achieves the highest or near-highest HV values, with relatively concentrated distributions, indicating that the solutions generated across multiple runs are stable and of good quality. This suggests that HMP-CE performs reliably in exploring the solution space and approximating the Pareto front, while maintaining a balanced performance in practical engineering optimization tasks.

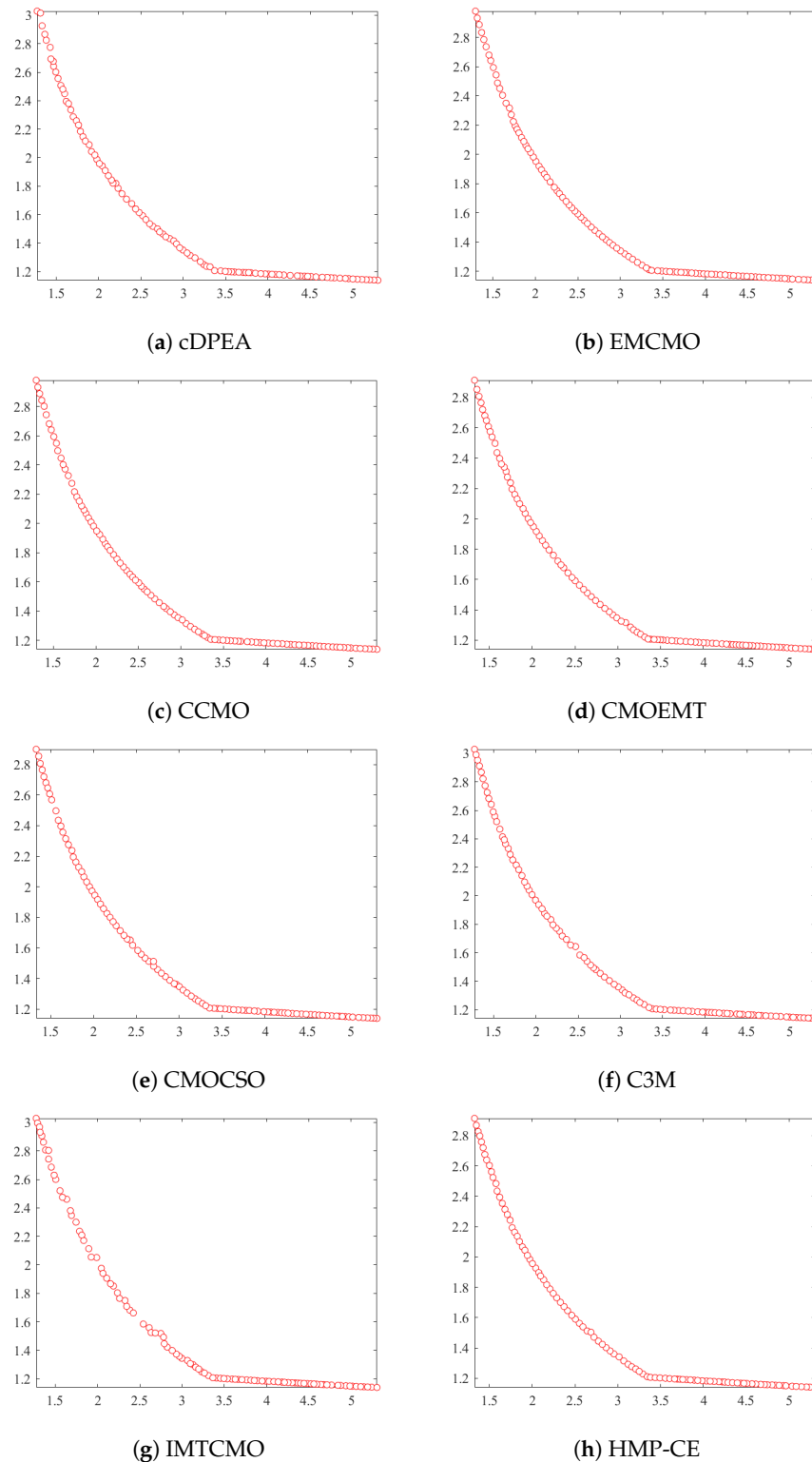


Figure 10. Final solution distributions of eight CMOEAs on the two-objective DBD problem.

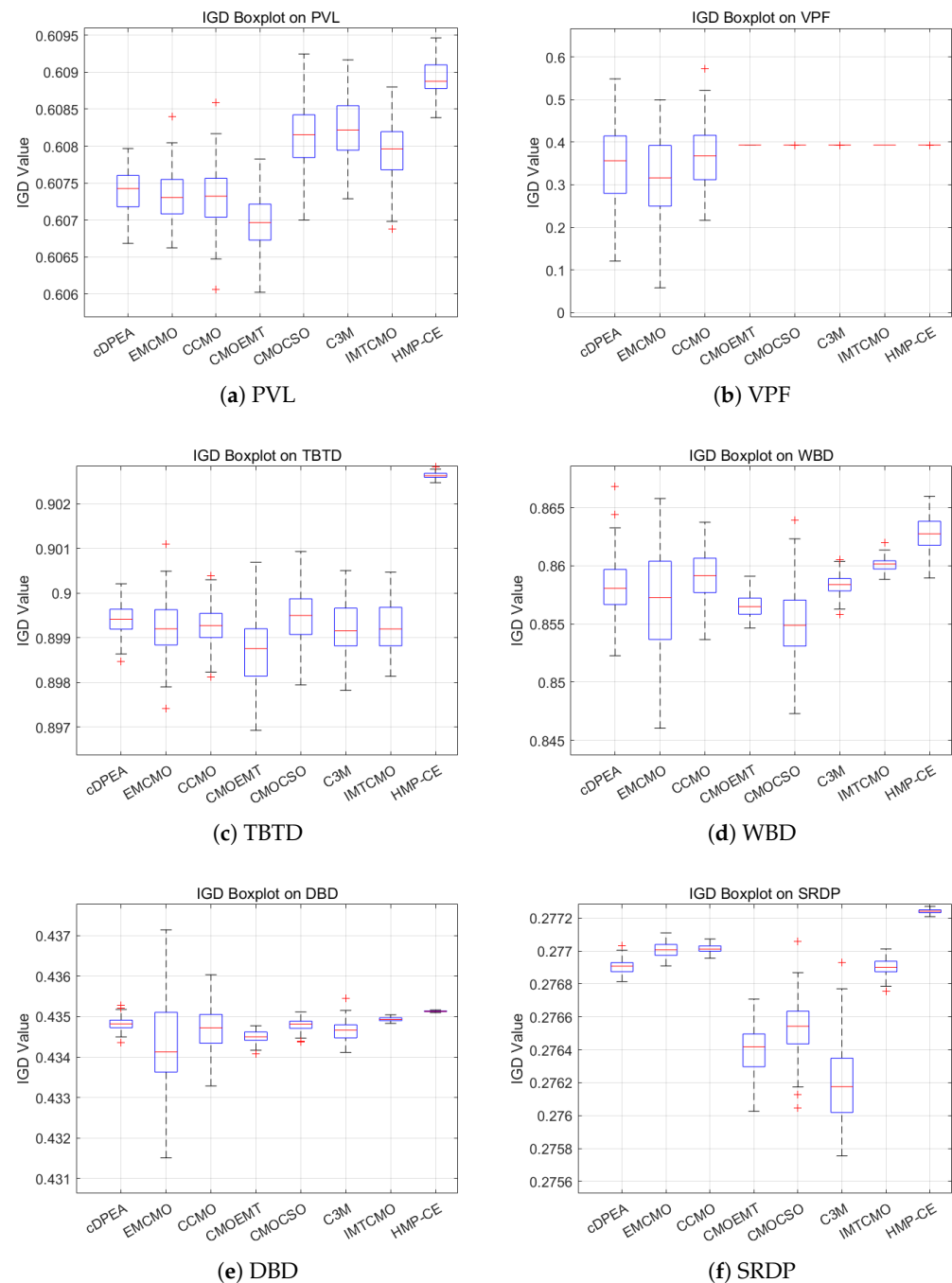


Figure 11. Performance comparison of eight CMOEAs on six real-world problems using boxplots.

Compared with the other seven algorithms (such as CMOEMT, CMOCSSO, C3M, CCMO, EMCMO, cDPEA, and IMTCMO), HMP-CE shows relatively stable performance across the tested problems. Some algorithms exhibit lower HV values or larger fluctuations in certain cases, suggesting limited coverage or less consistent convergence, whereas HMP-CE generally maintains moderate HV values with relatively small standard deviations, indicating more balanced performance. Overall, HMP-CE can generate reasonably reliable Pareto solution sets for different types of real-world constrained multi-objective problems, providing practical guidance for engineering design. In terms of search capability, solution quality, and stability, HMP-CE appears relatively balanced compared with other methods, without implying absolute superiority in all aspects.

Table 10. Statistical analysis (mean $\pm$ standard deviation) of HV values for eight algorithms evaluated on real-world problems.

Problem	M	D	cDPEA	EMCMO	CCMO	CMOEMT	CMOCSO	C3M	IMTCMO	HMP-CE
PVL	2	4	6.0744e-1 (3.35e-4) –	6.0741e-1 (3.44e-4) –	6.0731e-1 (4.21e-4) –	6.0694e-1 (4.18e-4) –	6.0818e-1 (4.32e-4) –	6.0826e-1 (4.04e-4) –	6.0790e-1 (3.81e-4) –	6.0892e-1 (2.24e-4)
VPF	2	5	3.4901e-1 (1.06e-1) –	3.2201e-1 (1.11e-1) –	3.6628e-1 (6.73e-2) –	3.9290e-1 (4.08e-5) –	3.9288e-1 (3.11e-5) –	3.9288e-1 (1.77e-5) –	3.9292e-1 (1.36e-5) –	3.9297e-1 (6.67e-6)
TBTD	2	3	8.9942e-1 (3.61e-4) –	8.9922e-1 (5.68e-4) –	8.9930e-1 (5.09e-4) –	8.9865e-1 (7.60e-4) –	8.9950e-1 (5.80e-4) –	8.9927e-1 (6.61e-4) –	8.9934e-1 (5.93e-4) –	9.0264e-1 (6.52e-5)
WBD	2	4	8.5828e-1 (2.76e-3) –	8.5613e-1 (4.14e-3) –	8.5938e-1 (1.98e-3) –	8.5664e-1 (9.77e-4) –	8.5532e-1 (3.35e-3) –	8.5838e-1 (9.86e-4) –	8.6008e-1 (5.31e-4) –	8.6278e-1 (1.40e-3)
DBD	2	4	4.3483e-1 (1.61e-4) –	4.3420e-1 (1.19e-3) –	4.3475e-1 (4.93e-4) –	4.3449e-1 (9.73e-5) –	4.3477e-1 (2.03e-4) –	4.3466e-1 (1.09e-4) –	4.3493e-1 (1.26e-4) –	4.3513e-1 (1.26e-4)
SRDP	2	7	2.7690e-1 (4.14e-5) –	2.7700e-1 (4.11e-5) –	2.7701e-1 (2.58e-5) –	2.7640e-1 (1.46e-4) –	2.7653e-1 (1.51e-4) –	2.7619e-1 (2.21e-4) –	2.7690e-1 (5.17e-5) –	2.7724e-1 (1.40e-5)
+ / – / $\approx$			0/6/0	0/6/0	0/6/0	0/6/0	0/6/0	0/6/0	0/6/0	

5.7. Real-World Engineering Problem Formulation

To improve the clarity and accessibility of this study, we present the following representative real-world constrained multi-objective optimization problem: the welded beam design problem. This problem originates from mechanical engineering, where the goal is to design a steel beam subjected to a static load at its free end. The design aims to simultaneously minimize the total fabrication cost and the end deflection of the beam, while ensuring that the structure satisfies engineering constraints related to stress, buckling, and geometric feasibility.

The problem involves the following four continuous decision variables: the weld thickness x_1 , the weld length x_2 , the beam height x_3 , and the beam thickness x_4 , forming a four-dimensional decision vector ($D = 4$).

The first objective is the total fabrication cost, including material and welding expenses, given by the following:

$$f_1 = 1.10471 x_1^2 x_2 + 0.04811 x_3 x_4 (14 + x_2), \quad (12)$$

while the second objective is the end deflection of the beam under the applied load as follows:

$$f_2 = \frac{4PL^3}{Ex_4 x_3^3}, \quad (13)$$

where the parameters are set as $P = 6000$, $L = 14$, and $E = 30 \times 10^6$.

The design must satisfy several engineering constraints. The shear stress τ and normal stress σ should not exceed their respective allowable limits, $\tau_{\max} = 13,600$ and $\sigma_{\max} = 30,000$. In addition, geometric feasibility requires that the weld thickness does not exceed the beam thickness ($x_1 \leq x_4$). Finally, the applied load must be lower than the critical buckling load P_c to ensure structural safety. These constraints guarantee that the resulting design is both safe and manufacturable.

The decision variables are bounded as follows: the weld thickness x_1 and beam thickness x_4 range from 0.125 to 5, while the weld length x_2 and beam height x_3 range from 0.1 to 10.

For consistency in experiments, the problem has been implemented in PlatEMO4.9. However, the mathematical formulation is fully independent of any specific software

platform, allowing researchers from different fields to understand, reproduce, and extend the proposed optimization framework without relying on PlatEMO4.9 or MATLAB R2021a.

5.8. Ablation Study on HMP-CE

In this subsection, to assess the contribution of each innovation in HMP-CE, we conduct ablation experiments on the DASCMP test problems by removing each component and evaluating the resulting variants independently. Specifically, HMP-CE1 removes all subpopulation migration mechanisms, HMP-CE2 removes the PADR mechanism, HMP-CE3 removes the CCTI mechanism, and HMP-CE4 replaces PADR with a standard uniform activation strategy (Uniform/Round-robin Activation), where each sub-policy is activated in a fixed, equal-frequency manner. According to the IGD results in Table 11 and the HV results in Table 12, HMP-CE achieves the best performance on seven out of the nine test problems and does not obtain the best results on DASCMP7 and DASCMP8. These observations indicate that each component contributes to the overall performance of HMP-CE.

Table 11. Statistical IGD (mean $\pm$ standard deviation) validation experiments of HMP-CE and its three variants on the DASCMP test problems.

Problem	M	D	HMP-CE1	HMP-CE2	HMP-CE3	HMP-CE4	HMP-CE
DASCMP1	2	30	4.6383e-3 (1.72e-3) –	3.9859e-3 (4.36e-4) –	3.7542e-3 (6.89e-4) –	2.0628e-1 (3.03e-2) –	2.9796e-3 (5.18e-4)
DASCMP2	2	30	4.9270e-3 (2.13e-4) –	4.9806e-3 (3.10e-4) –	4.9052e-3 (2.10e-4) –	1.1674e-1 (2.85e-2) –	4.0671e-3 (1.00e-4)
DASCMP3	2	30	4.2519e-2 (6.10e-2) –	3.5159e-2 (5.47e-2) –	5.4698e-2 (8.37e-2) –	1.8272e-1 (8.10e-2) –	1.9430e-2 (1.61e-4)
DASCMP4	2	30	3.3287e-2 (1.18e-1) –	5.1673e-2 (1.33e-1) –	1.7264e-1 (3.50e-1) –	1.2128e-2 (1.88e-3) –	1.2955e-3 (1.72e-4)
DASCMP5	2	30	4.6914e-2 (1.64e-1) –	1.2205e-2 (2.32e-2) –	7.5353e-3 (5.63e-3) –	7.5921e-3 (8.73e-4) –	3.0375e-3 (1.16e-4)
DASCMP6	2	30	1.2901e-1 (2.28e-1) –	4.6262e-2 (1.05e-1) –	1.2759e-1 (2.20e-1) –	2.7811e-2 (4.11e-3) –	2.1733e-2 (8.31e-3)
DASCMP7	3	30	9.0242e-2 (1.38e-1) +	6.7662e-2 (6.63e-2) +	1.8613e-1 (2.39e-1) –	9.3712e-2 (2.37e-1) –	1.0079e-1 (2.78e-1)
DASCMP8	3	30	9.9002e-2 (1.21e-1) +	1.0071e-1 (1.04e-1) +	1.0758e-1 (1.78e-1) +	5.6781e-2 (1.16e-2) –	1.7196e-1 (3.49e-1)
DASCMP9	3	30	4.3353e-2 (1.56e-3) –	4.2520e-2 (1.75e-3) –	4.3731e-2 (2.08e-3) –	2.5096e-1 (6.96e-2) –	4.0449e-2 (7.68e-4)
+ / – / $\approx$			2/7/0	2/7/0	0/9/0	1/8/0	

Table 12. Statistical HV (mean $\pm$ standard deviation) validation experiments of HMP-CE and its three variants on the DASCMP test problems.

Problem	M	D	HMP-CE1	HMP-CE2	HMP-CE3	HMP-CE4	HMP-CE
DASCMP1	2	30	2.1110e-1 (1.17e-3) –	2.1158e-1 (4.83e-4) –	2.1154e-1 (5.68e-4) –	1.6100e-1 (1.18e-2) –	2.1252e-1 (4.61e-4)
DASCMP2	2	30	3.5455e-1 (1.89e-4) –	3.5448e-1 (2.98e-4) –	3.5456e-1 (2.25e-4) –	2.9629e-1 (9.96e-3) –	3.5539e-1 (8.13e-5)

Table 12. Cont.

Problem	M	D	HMP-CE1	HMP-CE2	HMP-CE3	HMP-CE4	HMP-CE
DASCMOP3	2	30	3.0250e-1 (1.80e-2) –	3.0611e-1 (1.42e-2) –	3.0128e-1 (2.53e-2) –	2.3885e-1 (2.63e-2) –	3.1238e-1 (9.69e-5)
DASCMOP4	2	30	1.8797e-1 (4.13e-2) –	1.8134e-1 (4.98e-2) –	1.5667e-1 (7.59e-2) –	1.9596e-1 (4.72e-3) –	2.0403e-1 (2.49e-4)
DASCMOP5	2	30	3.2927e-1 (7.34e-2) –	3.4386e-1 (1.56e-2) –	3.4743e-1 (3.70e-3) –	3.4794e-1 (6.39e-4) –	3.5104e-1 (1.58e-4)
DASCMOP6	2	30	2.6273e-1 (9.62e-2) –	2.9668e-1 (5.64e-2) –	2.6795e-1 (8.67e-2) –	3.0721e-1 (1.76e-3) –	3.1065e-1 (5.28e-3)
DASCMOP7	3	30	2.6061e-1 (5.48e-2) –	2.6891e-1 (3.27e-2) +	2.2161e-1 (9.32e-2) –	2.7155e-1 (6.70e-2) –	2.6708e-1 (6.29e-2)
DASCMOP8	3	30	1.7929e-1 (4.70e-2) +	1.7575e-1 (5.12e-2) +	1.8399e-1 (4.56e-2) +	2.0336e-1 (2.18e-3) +	1.7422e-1 (6.38e-2)
DASCMOP9	3	30	2.0392e-1 (5.76e-4) –	2.0386e-1 (6.19e-4) –	2.0362e-1 (6.77e-4) –	1.4362e-1 (1.69e-2) –	2.0527e-1 (4.24e-4)
+ / – / ≈			1/8/0	2/7/0	1/8/0	1/8/0	

6. Conclusions

This paper presents HMP-CE, a hierarchical multi-population cooperative EA for CMOP. HMP-CE constructs a collaborative framework consisting of a primary constrained population (P_{Acp}), an unconstrained population (P_{Ucp}), and multiple sub-constrained populations (P_{Sscp}), and it introduces the following two key mechanisms: Population Activation–Dormancy Regulation (PADR) for adaptive control of subpopulation activation, and Constraint Combination Timing Identification (CCTI) for dynamic merging and elimination of overlapping or stagnated subpopulations. This design enables effective decomposition of complex constraints, parallel exploration under different relaxation levels, and cooperative evolution among populations.

Extensive experiments on benchmark mCMOPs, DASCMOPs, and real-world engineering problems demonstrate that HMP-CE consistently outperforms representative algorithms in IGD and HV metrics. The solutions are uniformly distributed along the Pareto front, exhibit strong boundary exploration, maintain high diversity, and remain stable in discontinuous, or highly constrained feasible regions. These results confirm that HMP-CE effectively enhances convergence, solution quality, and adaptability in complex constrained multi-objective scenarios. However, HMP-CE has certain limitations. The computational cost can become significant for very high-dimensional problems or extremely large populations, and its performance may be sensitive to parameter settings. In highly dynamic or uncertain environments, the algorithm's effectiveness may also be affected.

Future work includes the following: (1) adaptive adjustment of population sizes and collaboration strength to further improve efficiency and robustness; (2) extension to dynamic or uncertain environments with time-varying or uncertain constraints; (3) integration of surrogate models or parallel evaluation techniques to enhance computational efficiency in high-dimensional or large-scale problems; and (4) deeper theoretical analyses of convergence and inter-population knowledge transfer. These directions aim to further strengthen HMP-CE's scalability, applicability, and performance in solving more complex CMOPs.

Author Contributions: Conceptualization, X.X. and S.Z.; Data curation, X.X. and S.Z.; Formal analysis, S.Z.; Funding acquisition, J.L. and S.Z.; Investigation, S.Z.; Methodology, S.Z.; Project administration, S.Z.; Resources, J.L. and S.Z.; Software, S.Z.; Supervision, X.X. and S.Z.; Validation, X.X. and S.Z.;

Visualization, X.X. and S.Z.; Writing—original draft, X.X., X.G., J.L., and S.Z.; Writing—review and editing, X.X., X.G., J.L. and S.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research work is funded by project FRG-25-038-FA and supported by the Research Fund of Macao University of Science and Technology (FRG-MUST).

Data Availability Statement: All experimental data in the article were obtained through testing experiments conducted on the test sets (MW, LIRCMOP, DASCOP, and six real-world CMOPs) of the PlatEMO 4.9 platform. The access link is <https://github.com/BIMK/PlatEMO/>, accessed on 19 September 2023.

Conflicts of Interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Cao, J.; Yan, Z.; Chen, Z.; Zhang, J. A Pareto front estimation-based constrained multi-objective evolutionary algorithm. *Appl. Intell.* **2023**, *53*, 10380–10416.
2. Zhang, K.; Zhao, S.; Zeng, H.; Chen, J. Two-Stage Archive Evolutionary Algorithm for Constrained Multi-Objective Optimization. *Mathematics* **2025**, *13*, 470.
3. Yang, Y.; Liu, J.; Tan, S. A constrained multi-objective evolutionary algorithm based on decomposition and dynamic constraint-handling mechanism. *Appl. Soft Comput.* **2020**, *89*, 106104.
4. Lai, Y.; Chen, J.; Chen, Y.; Zeng, H.; Cai, J. Feedback Tracking Constraint Relaxation Algorithm for Constrained Multi-Objective Optimization. *Mathematics* **2025**, *13*, 629.
5. Zhang, S.; Yang, F.; Yan, C.; Zhou, D.; Zeng, X. An efficient batch-constrained bayesian optimization approach for analog circuit synthesis via multiobjective acquisition ensemble. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **2021**, *41*, 1–14.
6. Hao, L.; Peng, W.; Liu, J.; Zhang, W.; Li, Y.; Qin, K. Competition-based two-stage evolutionary algorithm for constrained multi-objective optimization. *Math. Comput. Simul.* **2025**, *230*, 207–226.
7. Chen, J.; Zhang, K.; Zeng, H.; Yan, J.; Dai, J.; Dai, Z. Adaptive Constraint Relaxation-Based Evolutionary Algorithm for Constrained Multi-Objective Optimization. *Mathematics* **2024**, *12*, 3075.
8. Bao, Q.; Wang, M.; Dai, G.; Chen, X.; Song, Z.; Li, S. An archive-based two-stage evolutionary algorithm for constrained multi-objective optimization problems. *Swarm Evol. Comput.* **2022**, *75*, 101161.
9. Liang, J.; Ban, X.; Yu, K.; Qu, B.; Qiao, K.; Yue, C.; Chen, K.; Tan, K.C. A survey on evolutionary constrained multiobjective optimization. *IEEE Trans. Evol. Comput.* **2022**, *27*, 201–221.
10. Wang, P.; Ma, Y.; Wang, M. A dynamic multi-objective optimization evolutionary algorithm based on particle swarm prediction strategy and prediction adjustment strategy. *Swarm Evol. Comput.* **2022**, *75*, 101164.
11. Wang, P.; Xiao, H.; Han, X.; Yang, F.; Li, L. A coevolutionary algorithm based on reference line guided archive for constrained multiobjective optimization. *Appl. Soft Comput.* **2023**, *142*, 110169.
12. Ming, M.; Wang, R.; Ishibuchi, H.; Zhang, T. A Novel Dual-Stage Dual-Population Evolutionary Algorithm for Constrained Multiobjective Optimization. *IEEE Trans. Evol. Comput.* **2021**, *26*, 1129–1143.
13. Qin, C.; Ming, F.; Gong, W.; Gu, Q. Constrained multi-objective optimization via two archives assisted push–pull evolutionary algorithm. *Swarm Evol. Comput.* **2022**, *75*, 101178.
14. Qiao, K.; Liang, J.; Yu, K.; Yue, C.; Lin, H.; Zhang, D.; Qu, B. Evolutionary constrained multiobjective optimization: Scalable high-dimensional constraint benchmarks and algorithm. *IEEE Trans. Evol. Comput.* **2024**, *28*, 965–979.
15. Liu, Z.Z.; Qin, Y.; Song, W.; Zhang, J.; Li, K. Multiobjective-based constraint-handling technique for evolutionary constrained multiobjective optimization: A new perspective. *IEEE Trans. Evol. Comput.* **2022**, *27*, 1370–1384.
16. Sun, R.; Zou, J.; Liu, Y.; Yang, S.; Zheng, J. A multistage algorithm for solving multiobjective optimization problems with multiconstraints. *IEEE Trans. Evol. Comput.* **2023**, *27*, 1207–1219.
17. Tang, H.; Yu, F.; Zou, J.; Yang, S.; Zheng, J. A constrained multi-objective evolutionary strategy based on population state detection. *Swarm Evol. Comput.* **2022**, *68*, 100978.
18. Xia, M.; Chong, Q.; Dong, M. A constrained multi-objective evolutionary algorithm with two-stage resources allocation. *Swarm Evol. Comput.* **2023**, *79*, 101313.
19. Zou, J.; Sun, R.; Liu, Y.; Hu, Y.; Yang, S.; Zheng, J.; Li, K. A multipopulation evolutionary algorithm using new cooperative mechanism for solving multiobjective problems with multiconstraint. *IEEE Trans. Evol. Comput.* **2023**, *28*, 267–280.
20. Yang, K.; Zheng, J.; Zou, J.; Yu, F.; Yang, S. A dual-population evolutionary algorithm based on adaptive constraint strength for constrained multi-objective optimization. *Swarm Evol. Comput.* **2023**, *77*, 101247.

21. Dong, J.; Gong, W.; Ming, F.; Wang, L. A two-stage evolutionary algorithm based on three indicators for constrained multi-objective optimization. *Expert Syst. Appl.* **2022**, *195*, 116499.
22. Liang, J.; Lin, H.; Yue, C.; Yu, K.; Guo, Y.; Qiao, K. Multiobjective differential evolution with speciation for constrained multimodal multiobjective optimization. *IEEE Trans. Evol. Comput.* **2023**, *27*, 1115–1129.
23. Ming, F.; Gong, W.; Wang, L.; Gao, L. Constrained multiobjective optimization via multitasking and knowledge transfer. *IEEE Trans. Evol. Comput.* **2024**, *28*, 77–89.
24. Liang, J.; Qiao, K.; Yu, K.; Qu, B.; Yue, C.; Guo, W.; Wang, L. Utilizing the relationship between unconstrained and constrained pareto fronts for constrained multiobjective optimization. *IEEE Trans. Cybern.* **2022**, *53*, 3873–3886.
25. Li, K.; Chen, R.; Fu, G.; Yao, X. Two-archive evolutionary algorithm for constrained multiobjective optimization. *IEEE Trans. Evol. Comput.* **2018**, *23*, 303–315.
26. Tian, Y.; Zhang, T.; Xiao, J.; Zhang, X.; Jin, Y. A coevolutionary framework for constrained multiobjective optimization problems. *IEEE Trans. Evol. Comput.* **2020**, *25*, 102–116.
27. Ding, L.; Shi, C.; Zhou, J. Collaborative route optimization and resource management strategy for multi-target tracking in airborne radar system. *Digit. Signal Process.* **2023**, *138*, 104051.
28. Zhang, Z.; Zhang, H.; Tian, Y.; Li, C.; Yue, D. Cooperative constrained multi-objective dual-population evolutionary algorithm for optimal dispatching of wind-power integrated power system. *Swarm Evol. Comput.* **2024**, *87*, 101525.
29. Gu, Q.; Liu, R.; Hui, Z.; Wang, D. A constrained multi-objective optimization algorithm based on coordinated strategy of archive and weight vectors. *Expert Syst. Appl.* **2024**, *244*, 122961.
30. Kawachi, T.; Kushida, J.i.; Hara, A.; Takahama, T. Efficient parameter-free adaptive penalty method with balancing the objective function value and the constraint violation. *Int. J. Comput. Intell. Stud.* **2021**, *10*, 127–160.
31. Qiao, K.; Yu, K.; Qu, B.; Liang, J.; Song, H.; Yue, C. An evolutionary multitasking optimization framework for constrained multiobjective optimization problems. *IEEE Trans. Evol. Comput.* **2022**, *26*, 263–277.
32. Zou, J.; Sun, R.; Yang, S.; Zheng, J. A dual-population algorithm based on alternative evolution and degeneration for solving constrained multi-objective optimization problems. *Inf. Sci.* **2021**, *579*, 89–102.
33. Wang, Y.; Zuo, M.; Gong, D. Migration-based algorithm library enrichment for constrained multi-objective optimization and applications in algorithm selection. *Inf. Sci.* **2023**, *649*, 119593.
34. Tanabe, R.; Ishibuchi, H. A decomposition-based evolutionary algorithm for multi-modal multi-objective optimization. In *Parallel Problem Solving from Nature—PPSN XV: Proceedings of the 15th International Conference, Coimbra, Portugal, 8–12 September 2018, Proceedings, Part I 15*; Springer: Cham, Switzerland, 2018; pp. 249–261.
35. Liang, J.; Qiao, K.; Yue, C.; Yu, K.; Qu, B.; Xu, R.; Li, Z.; Hu, Y. A clustering-based differential evolution algorithm for solving multimodal multi-objective optimization problems. *Swarm Evol. Comput.* **2021**, *60*, 100788.
36. Zhou, J.; Gao, L.; Yao, X.; Zhang, C.; Chan, F.T.; Lin, Y. Evolutionary many-objective assembly of cloud services via angle and adversarial direction driven search. *Inf. Sci.* **2020**, *513*, 143–167.
37. Zhou, Y.; Kang, J.; Guo, H. Many-objective optimization of feature selection based on two-level particle cooperation. *Inf. Sci.* **2020**, *532*, 91–109.
38. Mala-Jetmarova, H.; Sultanova, N.; Savic, D. Lost in optimisation of water distribution systems? A literature review of system operation. *Environ. Model. Softw.* **2017**, *93*, 209–254.
39. Datta, R.; Pradhan, S.; Bhattacharya, B. Analysis and design optimization of a robotic gripper using multiobjective genetic algorithm. *IEEE Trans. Syst. Man, Cybern. Syst.* **2015**, *46*, 16–26.
40. Zhang, Y.; Tian, Y.; Jiang, H.; Zhang, X.; Jin, Y. Design and analysis of helper-problem-assisted evolutionary algorithm for constrained multiobjective optimization. *Inf. Sci.* **2023**, *648*, 119547.
41. Zhou, T.; He, P.; Niu, B.; Yue, G.; Wang, H. A novel competitive constrained dual-archive dual-stage evolutionary algorithm for constrained multiobjective optimization. *Swarm Evol. Comput.* **2023**, *83*, 101417.
42. Zhong, X.; Yao, X.; Gong, D.; Qiao, K.; Gan, X.; Li, Z. A dual-population-based evolutionary algorithm for multi-objective optimization problems with irregular Pareto fronts. *Swarm Evol. Comput.* **2024**, *87*, 101566.
43. Lin, J.; Zhang, S.X.; Zheng, S.Y. A diverse/converged individual competition algorithm for computationally expensive many-objective optimization. *Appl. Intell.* **2024**, *54*, 2564–2581.
44. Su, T.V.; Hang, D.D. Second-order optimality conditions in locally Lipschitz multiobjective fractional programming problem with inequality constraints. *Optimization* **2023**, *72*, 1171–1198.
45. Fan, C.; Wang, J.; Xiao, L.; Cheng, F.; Ai, Z.; Zeng, Z. A coevolution algorithm based on two-staged strategy for constrained multi-objective problems. *Appl. Intell.* **2022**, *52*, 17954–17973.
46. Yeste, P.; Melsen, L.A.; García-Valdecasas Ojeda, M.; Gámiz-Fortis, S.R.; Castro-Díez, Y.; Esteban-Parra, M.J. A Pareto-Based Sensitivity Analysis and Multiobjective Calibration Approach for Integrating Streamflow and Evaporation Data. *Water Resour. Res.* **2023**, *59*, e2022WR032325.

47. Takahama, T.; Sakai, S. Constrained optimization by the ϵ constrained differential evolution with an archive and gradient-based mutation. In *Proceedings of the IEEE Congress on Evolutionary Computation*; IEEE: Piscataway, NJ, USA, 2010; pp. 1–9.
48. Ying, W.Q.; He, W.P.; Huang, Y.X.; Li, D.T.; Wu, Y. An adaptive stochastic ranking mechanism in MOEA/D for constrained multi-objective optimization. In *Proceedings of the 2016 International Conference on Information System and Artificial Intelligence (ISAI)*; IEEE: Piscataway, NJ, USA, 2016; pp. 514–518.
49. Zhou, Y.; Zhu, M.; Wang, J.; Zhang, Z.; Xiang, Y.; Zhang, J. Tri-goal evolution framework for constrained many-objective optimization. *IEEE Trans. Syst. Man Cybern. Syst.* **2018**, *50*, 3086–3099.
50. Fan, Z.; Li, W.; Cai, X.; Hu, K.; Lin, H.; Li, H. Angle-based constrained dominance principle in MOEA/D for constrained multi-objective optimization problems. In *Proceedings of the 2016 IEEE Congress on Evolutionary Computation (CEC)*; IEEE: Piscataway, NJ, USA, 2016; pp. 460–467.
51. Wolpert, D.H.; Macready, W.G. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **2002**, *1*, 67–82.
52. Tian, Y.; Zhang, Y.; Su, Y.; Zhang, X.; Tan, K.C.; Jin, Y. Balancing objective optimization and constraint satisfaction in constrained evolutionary multiobjective optimization. *IEEE Trans. Cybern.* **2021**, *52*, 9559–9572.
53. Liu, Z.Z.; Wang, Y. Handling constrained multiobjective optimization problems with constraints in both the decision and objective spaces. *IEEE Trans. Evol. Comput.* **2019**, *23*, 870–884.
54. Fan, Z.; Li, W.; Cai, X.; Li, H.; Wei, C.; Zhang, Q.; Deb, K.; Goodman, E. Push and pull search for solving constrained multi-objective optimization problems. *Swarm Evol. Comput.* **2019**, *44*, 665–679.
55. Fan, Z.; Li, W.; Cai, X.; Huang, H.; Fang, Y.; You, Y.; Mo, J.; Wei, C.; Goodman, E. An improved epsilon constraint-handling method in MOEA/D for CMOPs with large infeasible regions. *arXiv* **2017**, arXiv:1707.08767.
56. Zhang, Q.; Li, H. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Trans. Evol. Comput.* **2007**, *11*, 712–731.
57. Ma, H.; Wei, H.; Tian, Y.; Cheng, R.; Zhang, X. A multi-stage evolutionary algorithm for multi-objective optimization with complex constraints. *Inf. Sci.* **2021**, *560*, 68–91.
58. Ming, M.; Trivedi, A.; Wang, R.; Srinivasan, D.; Zhang, T. A dual-population-based evolutionary algorithm for constrained multiobjective optimization. *IEEE Trans. Evol. Comput.* **2021**, *25*, 739–753.
59. Liu, Z.Z.; Wang, B.C.; Tang, K. Handling constrained multiobjective optimization problems via bidirectional coevolution. *IEEE Trans. Cybern.* **2021**, *52*, 10163–10176.
60. Ruan, G.; Yu, G.; Zheng, J.; Zou, J.; Yang, S. The effect of diversity maintenance on prediction in dynamic multi-objective optimization. *Appl. Soft Comput.* **2017**, *58*, 631–647.
61. Liu, Z.; Chen, H.; Zhang, T.; Meuser, C.; von Unwerth, T. Multi-Objective Operating Parameters Optimization for the Start Process of Proton Exchange Membrane Fuel Cell Stack with Non-Dominated Sorting Genetic Algorithm II. *J. Electrochem. Soc.* **2024**, *171*, 034506.
62. Wang, J.; Liang, G.; Zhang, J. Cooperative differential evolution framework for constrained multiobjective optimization. *IEEE Trans. Cybern.* **2018**, *49*, 2060–2072.
63. Tian, Y.; Cheng, R.; Zhang, X.; Jin, Y. PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [educational forum]. *IEEE Comput. Intell. Mag.* **2017**, *12*, 73–87.
64. Ming, F.; Gong, W.; Li, D.; Wang, L.; Gao, L. A competitive and cooperative swarm optimizer for constrained multiobjective optimization problems. *IEEE Trans. Evol. Comput.* **2022**, *27*, 1313–1326.
65. Zarei, F.; Arashpour, M.; Mirnezami, S.A.; Shahabi-Shahamiri, R.; Ghasemi, M. Multi-skill resource-constrained project scheduling problem considering overlapping: Fuzzy multi-objective programming approach to a case study. *Int. J. Constr. Manag.* **2024**, *24*, 820–833.
66. Wilcoxon, F. Some rapid approximate statistical procedures. *Ann. N. Y. Acad. Sci.* **1950**, *52*, 808–814.
67. Bader, J.; Zitzler, E. HypE: An algorithm for fast hypervolume-based many-objective optimization. *Evol. Comput.* **2011**, *19*, 45–76.
68. He, C.; Cheng, R.; Yazdani, D. Adaptive offspring generation for evolutionary large-scale multiobjective optimization. *IEEE Trans. Syst. Man Cybern. Syst.* **2020**, *52*, 786–798.
69. Liu, S.; Lin, Q.; Li, Q.; Tan, K.C. A comprehensive competitive swarm optimizer for large-scale multiobjective optimization. *IEEE Trans. Syst. Man Cybern. Syst.* **2021**, *52*, 5829–5842.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.