

Adaptive memory-prediction fusion mechanism based on change intensity detection for dynamic constrained multi-objective optimization

Huaiwen Liu^a, Baosheng Li^{id b,*}, Hongbiao Tian^b, Zhiyang Zhang^{id b}, Qianlong Dang^{id a,b}, Junhu Ruan^{a,*}

^a College of Economics and Management, Northwest A & F University, Yangling, 712100, China

^b College of Science, Northwest A & F University, Yangling, 712100, China

ARTICLE INFO

Keywords:

Dynamic constrained multi-objective optimization
Tri-population optimization
Prediction mechanism
Historical knowledge
Change intensity detection

ABSTRACT

Dynamic constrained multi-objective optimization problems (DCMOPs) pose significant challenges due to the time-varying nature of objectives and constraints, which complicate the application of traditional optimization methods. Existing studies have developed a range of strategies that utilize historical knowledge to track the Pareto optimal solutions of DCMOPs. However, current dynamic constrained multi-objective evolutionary algorithms (DCMOEAs) struggle to efficiently utilize historical knowledge for generating high-quality initial populations, and they often fail to maintain a balance between diversity and convergence while ensuring feasibility. To address these issues, an adaptive memory-prediction fusion mechanism driven by change intensity detection is proposed. Specifically, the mechanism adaptively exploits historical knowledge according to the detected intensity of environmental changes, thereby enabling more efficient knowledge utilization. Moreover, a dynamic tri-population optimization framework is constructed, which can adaptively adjust the sizes of two auxiliary populations for convergence and diversity. In addition, this framework is designed to focus computational resources on the task that is more important for the current optimization. Experimental results on 19 benchmark problems with 76 test instances and the dynamic raw ore allocation problem demonstrate that the proposed algorithm shows competitive overall performance compared with the other eight state-of-the-art DCMOEAs.

1. Introduction

Dynamic multi-objective optimization problems (DMOPs) and constrained multi-objective optimization problems (CMOPs) are prevalent in real-world applications and have garnered significant attention (Zhang et al., 2025a). In DMOPs, multiple objective functions vary over time (Nguyen & Yao, 2012). These problems arise in fields such as hydrothermal power scheduling (Deb et al., 2007) and assembly line scheduling (Zhou & Huang, 2023). CMOPs are widely encountered in vehicle routing (Wang et al., 2016), service facility location (Tan et al., 2018), and water supply system optimization (Mala-Jetmarova et al., 2017), where multiple objectives are optimized while satisfying a set of equality and inequality constraints (Liang et al., 2024). However, the objective functions and constraints in CMOPs may change over time in practical scenarios, while most DMOP formulations fail to adequately incorporate realistic constraints. Consequently, dynamic constrained multi-objective optimization problems (DCMOPs) have received increasing attention in recent years (Nguyen & Yao, 2009, 2012), involving multiple conflicting objectives and constraints changing over time.

Dynamic constrained multi-objective evolutionary algorithms (DCMOEAs) have been developed to effectively address DCMOPs. These algorithms extend traditional constrained multi-objective evolutionary algorithms (CMOEAs) by introducing mechanisms to detect environmental changes and adaptively modify the optimization process. Specifically, DCMOEAs are designed to approximate the constrained Pareto optimal set (CPS) in a dynamic environment, where the objectives and constraints change over time (Zhang et al., 2025b). In general, DCMOEAs consist of the environment detection mechanism, the change response strategy, and the static optimizer. The environment detection mechanism is used to detect environmental changes across different generations. Once a change is detected, the change response strategy generates an initial population that is more suitable for the new environment. Then, the static optimizer drives this initial population toward the constrained Pareto front (CPF) in the new environment. The difficulty of solving DCMOPs lies in the simultaneous presence of environmental dynamics and constraint-induced search barriers. From the dynamic optimization perspective, the algorithm should respond rapidly to environmental changes and generate a promising initial population for the

* Corresponding authors.

E-mail addresses: liuhuaiwen@nwfufu.edu.cn (H. Liu), baoshengli96@nwfufu.edu.cn (B. Li), 2023014517@nwfufu.edu.cn (H. Tian), jeanzhang@nwfufu.edu.cn (Z. Zhang), xidianqldang@163.com (Q. Dang), rjh@nwsuaf.edu.cn (J. Ruan).

<https://doi.org/10.1016/j.eswa.2026.134095>

Received 13 May 2026; Received in revised form 28 June 2026; Accepted 17 August 2026

Available online 23 August 2026

0957-4174/© 2026 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

new environment (Feng et al., 2022). An inappropriate initial population may delay the tracking process and cause the search to miss the newly shifted CPF. From the constrained optimization perspective, the algorithm should maintain a proper balance between convergence and diversity while ensuring feasibility (Liu et al., 2025). Since infeasible regions may separate feasible regions, the population needs to traverse infeasible areas effectively while preserving sufficient diversity to cover different parts of the CPF (Bao et al., 2025). Therefore, an effective DC-MOEA should not only design a reliable change response strategy, but also employ a capable static optimizer that integrates appropriate constraint handling techniques (CHTs) and diversity-maintenance mechanisms.

Existing studies of DCMOEA have mainly proceeded along two directions, including designing change response strategies to generate suitable initial populations for new environments (Chen et al., 2020) and developing static optimizers that remain effective under dynamic conditions (Bao et al., 2025). Current change response strategies can generally be classified into diversity-based (Deb et al., 2007), memory-based (Branke, 1999), and prediction-based approaches (Zhang et al., 2025b). However, each response strategy exhibits inherent limitations (Chen et al., 2024a). Specifically, diversity-based approaches usually enhance population spread without sufficiently utilizing historical knowledge, which may yield initial populations poorly matched to the new environment and thus impair convergence. Memory-based approaches tend to lose effectiveness when the algorithm encounters an unseen environment. Prediction-based approaches are often computationally more expensive. In addition, research on static optimizers for DCMOPs has primarily emphasized robust CHTs and the design of auxiliary tasks to assist population evolution (Zhang et al., 2025b). Representative CHTs include penalty-function methods (Ray et al., 2001), constraint dominance principle (CDP) (Deb et al., 2002), and constraint relaxation methods (Takahama & Sakai, 2010). When integrated into multi-objective evolutionary algorithms (MOEAs), these techniques enable effective treatment of CMOPs. Similarly, when further incorporated into dynamic multi-objective evolutionary algorithms (DMOEA), they provide a natural basis for handling DCMOPs. Notably, different CHTs impose different preferences over feasibility, convergence, and diversity. These preferences can substantially influence the performance of algorithms across tasks (Liu et al., 2025). Another common method is to introduce auxiliary populations or tasks to support the evolution of the main population (Tian et al., 2020). Because infeasible regions may severely hinder convergence, relaxing part or all constraints can help the population approach the unconstrained Pareto front (UPF) rapidly (Tian et al., 2020). Accordingly, cooperation between the main population and an unconstrained or partially constrained auxiliary population can improve the convergence (Qiao et al., 2023). However, when the CPF is far from the UPF, these auxiliary tasks become much less effective.

To overcome the above limitations, this paper proposes an adaptive memory-prediction fusion mechanism driven by change intensity detection, together with a tri-population optimization framework. The proposed detection mechanism not only determines whether a change has taken place but also quantifies its intensity. Based on this information, the algorithm adaptively combines two complementary sources of historical knowledge, including the change trend inferred from the two most recent environments and the information extracted from the most similar historical environment. In this way, the proposed method can remain effective whether or not past environments provide a close match to the current one. Furthermore, to fully exploit the detected change intensity and historical knowledge, a tri-population optimization framework is developed to jointly balance convergence and diversity while ensuring feasibility. This framework consists of a main population, a convergence auxiliary population, and a diversity auxiliary population. Computational resources allocated to the two auxiliary populations are dynamically adjusted according to the detected change intensity, thereby improving resource utilization. Specifically, the main contributions are summarized as follows:

1. **Environmental change intensity detection module:** This module is developed to identify environmental changes and quantify their intensity. By evaluating the differences in objective functions and constraint violations for the same individuals across consecutive environments, the proposed module characterizes the intensity of environmental change and provides a basis for selecting appropriate optimization strategies. This design improves the adaptability to different change patterns and problem characteristics.
2. **Adaptive memory-prediction fusion mechanism:** This mechanism is proposed to exploit two forms of historical knowledge, including the recent change trend and the most similar historical environment. Their relative importance is adjusted according to the detected change intensity. When the environmental change is mild, the mechanism places greater emphasis on prediction based on the most recent change trend. When the change is severe, it relies more on perturbing elite individuals from the most similar historical environment to enhance diversity. In addition, a confidence measure is introduced for the recent change trend, which improves the reliability of historical knowledge selection. Through the adaptive integration of these two information sources, the proposed mechanism can generate initial populations that better fit the current environment under different change scenarios.
3. **Dynamic tri-population optimization framework:** This framework is designed to promote convergence and diversity in a coordinated manner while ensuring feasibility. It contains a main task with full consideration of constraints, a convergence auxiliary task without constraints, and a diversity auxiliary task based on reference vectors. Through cooperation among the three populations, the proposed framework enhances the overall search performance in dynamic constrained environments. Moreover, the computational resources assigned to the two auxiliary populations are adjusted according to change intensity. When environmental changes are severe, more resources are allocated to the unconstrained auxiliary task to approach the UPF rapidly and provide promising search directions. When environmental changes are mild, more resources are assigned to the diversity auxiliary population to support further exploration of the CPF.

The remainder of this paper is organized as follows. Section 2 introduces the core concepts of DCMOPs. Section 3 reviews the related work and presents the motivation of this study. Section 4 details the proposed method and underlying principles. The experimental studies and results are presented in Section 5. Finally, Section 6 concludes this paper and outlines future work.

2. Preliminary study

Generally, a DCMOP can be represented as follows (Wei & Wang, 2012):

$$\min F(\vec{x}, t) = (f_1(\vec{x}, t), f_2(\vec{x}, t), \dots, f_M(\vec{x}, t))^T$$

$$\begin{cases} \vec{x} \in \Omega, t \in \Omega_T \\ h_i(\vec{x}, t) = 0, i = 1, \dots, p \\ g_j(\vec{x}, t) \leq 0, j = p + 1, \dots, p + q \end{cases}, \quad (1)$$

where $\vec{x} = (x_1, x_2, \dots, x_D)^T$ is a D -dimensional decision vector and $F(\vec{x}, t)$ is an M -dimensional objective vector. Ω and Ω_T represent the decision space and the time domain, respectively. Moreover, $h_i(\vec{x}, t)$ and $g_j(\vec{x}, t)$ denote the i th equality constraint and the j th inequality constraint, respectively. DCMOPs are primarily categorized into three types, including problems with dynamic objectives, problems with dynamic constraints, and problems with both objectives and constraints varying over time. The time variable is typically denoted by t , which is defined as follows:

$$t = \left\lfloor \frac{\tau}{\tau_t} \right\rfloor \frac{1}{n_t}, \quad (2)$$

where τ represents the generation counter, τ_i and n_i denote the frequency of environmental changes and the severity of change over time. τ_i is inversely proportional to the frequency of environmental changes. A smaller τ_i indicates more frequent environmental changes, while a smaller n_i signifies more drastic environmental changes over time t . The different combinations of n_i and τ_i correspond to different dynamic scenarios. Although the structure of basic test problem remains the same, the environmental trajectory, the severity of each change, and the available recovery time between two consecutive changes are different.

Definition 1 (Dynamic Feasible Solutions): Similar to CMOPs, solutions within the feasible regions are termed feasible solutions, while those outside the feasible regions are infeasible. At time t , the constraint violation (CV) of a solution can be calculated as follows:

$$CV(\vec{x}, t) = \sum_{k=1}^{p+q} CV_k(\vec{x}, t), \quad (3)$$

with

$$CV_k(\vec{x}, t) = \begin{cases} \max(0, |h_k(\vec{x}, t)| - \delta), & 1 \leq k \leq p, \\ \max(0, g_k(\vec{x}, t)), & p+1 \leq k \leq p+q, \end{cases} \quad (4)$$

where δ is a small positive value used to relax the constraint boundaries. When $CV(\vec{x}, t) = 0$, $\vec{x}$ is considered a feasible solution at time t .

Definition 2 (Dynamic Constrained Pareto Optimal Set): When a solution $\vec{x}^*$ is not dominated by any other solution, it is defined as a Pareto optimal solution. The set comprising all Pareto optimal solutions is called the Pareto optimal set (PS). Due to the presence of constraints, the set of all feasible Pareto optimal solutions is termed the CPS. The set formed by all constrained Pareto optimal solutions within time t is referred to as the dynamic constrained Pareto optimal set DCPS t :

$$DCPS^t = \{\vec{x}^* \in \mathbf{O}(t) \mid \nexists \vec{x} \in \mathbf{O}(t), \vec{x} \prec_t \vec{x}^*\}, \quad (5)$$

where $\mathbf{O}(t)$ is the dynamic feasible region, and $\vec{x} \prec_t \vec{x}^*$ indicates that $\vec{x}$ Pareto dominates $\vec{x}^*$ at time t .

Definition 3 (Dynamic Constrained Pareto Optimal Front): The dynamic constrained Pareto optimal front is the mapping of DCPS t onto the objective space, denoted as DCPF t :

$$DCPF^t = \{F(\vec{x}^*, t) \mid \vec{x}^* \in DCPS^t\}. \quad (6)$$

3. Related work

3.1. Dynamic response strategies

Given the frequent environmental changes, a central challenge for DCMOEAs is to identify the CPF for each environment within limited computational resources. Existing DCMOEAs are commonly categorized into three classes. The first class is diversity-based methods, which aim to preserve population diversity so that strong global search capability can be retained after changes (Goh & Tan, 2009). The second class is memory-based methods, and they are effective for periodically changing problems due to the reuse of historical solutions (Xu et al., 2018). As the third class, prediction-based methods track change trends and learn regularities to anticipate future environments (Cao et al., 2020). Owing to its strong performance in this context, prediction-based approaches have attracted increasing attention (Zhang et al., 2025b).

Specifically, diversity-based methods primarily seek to prevent premature convergence after an environmental change (Jiang et al., 2018). In the presence of feasibility preference, some optimal solutions in the new environment may cause the algorithm to discard solutions in other directions too aggressively. This behavior can degrade population diversity and eventually trap the search in local optimal regions (Wang et al., 2025). To mitigate this issue, Deb et al. (2007) proposed two variants of a dynamic NSGA-II-based response strategy. One replaces part of the population with randomly generated solutions when a change occurs, and the other perturbs a subset of existing solutions via mutation. Although both strategies can inject diversity, their effectiveness is often

limited. Woldesenbet and Yen (2009) proposed an alternative approach based on variable relocation, which helps solutions that have already converged in the current environment adapt to the new environment.

Memory-based methods store information from historical environments and retrieve it when similar environments appear, thereby enabling the population to reach feasible regions more rapidly. This strategy can substantially improve search efficiency when changes exhibit periodicity (Branke, 1999). Azzouz et al. (2017) combined a memory mechanism with local search and random strategies. This approach performs well on DCMOPs with dynamic objective functions. Additionally, Liang et al. (2019) proposed a hybrid memory-based strategy that adopts different response strategies depending on the presence of similar environments.

Prediction-based techniques aim to discover and exploit change regularities across historical environments in DCMOPs, and then use these regularities to estimate the locations of optimal solutions in the forthcoming environment (Zhou et al., 2014). Instead of directly reusing historical solutions, these methods track change trends through learned knowledge. Due to their strong tracking capability and broad applicability, prediction-based methods have become a major focus in DCMOEAs research. Yu et al. (2024) proposed a learning-assisted framework that collects historical knowledge from multiple perspectives and significantly enhances the performance of static optimizers on DCMOPs. Chen et al. (2024a) developed an optimization approach based on subspace knowledge transfer. Reference vectors are constructed to partition the objective space into feasible and infeasible subspaces. In the feasible subspace, Kalman filtering is employed to predict new solutions, while solution generation in the infeasible subspace relies on knowledge transferred from the nearest promising subspace. Zhang et al. (2025b) introduced a registration method from computer vision into dynamic multi-objective optimization. By matching corresponding solutions across environmental changes, this approach captures variations of the CPF in a more refined manner.

3.2. Constrained multi-objective optimizers

Due to the presence of constraints, the objective space is divided into feasible and infeasible regions. These infeasible regions often truncate feasible regions or split them into disconnected subregions, making the search prone to local optima (Zhu et al., 2020). A primary challenge in designing CMOEAs is to enable the population to traverse infeasible regions and explore the CPF as comprehensively as possible. Existing studies mainly focus on the design of CHTs and optimization strategies (Liang et al., 2024). Representative CHTs include the penalty function methods, the CDP methods, and the constraint relaxation methods (Ming et al., 2022). Common optimization strategies include multi-population methods, multi-stage methods, and hybrid methods (Liu et al., 2025).

Penalty-function methods construct a penalty term associated with the degree of constraint violation and incorporate it into the objective function to form a new fitness function, thereby transforming the constrained problem into an unconstrained one (Xia et al., 2021). Existing penalty function methods can be classified into static, dynamic, and adaptive variants (Liang et al., 2023; Woldesenbet et al., 2009). These methods are frequently adopted in decomposition-based CMOEAs, while the selection of appropriate penalty coefficients remains a major challenge. The CDP has become one of the most widely used CHTs. When comparing two solutions, the feasible solution is preferred over any infeasible solution. Among infeasible solutions, the one with smaller constraint violation is preferred. When two solutions are feasible, Pareto dominance determines the preference (Jain & Deb, 2014). CDP can handle constraints effectively, but its strong preference for feasibility may reduce population diversity. To mitigate this limitation, constraint relaxation methods have been introduced (Liang et al., 2021; Zapotecas-Martínez & Ponsich, 2020). A relaxation parameter is used to loosen constraint boundaries, and a solution is treated as feasible when the constraint violation is below the specified threshold.

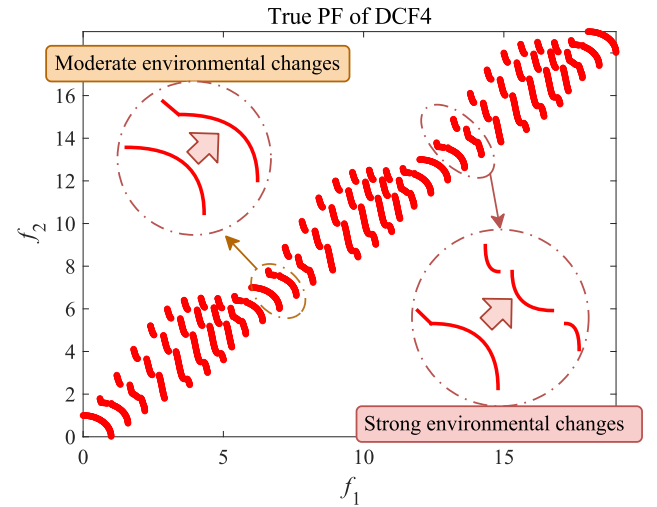
To further improve performance under complex constraints, substantial effort has been devoted to designing more effective optimization strategies (Dang et al., 2025). Multi-population methods introduce auxiliary tasks and construct corresponding auxiliary populations to assist the main population. Cooperation among populations can better balance the diversity and convergence while ensuring feasibility. CCMO (Tian et al., 2020) is a representative dual-population framework that exploits collaboration between two populations to utilize auxiliary search effectively. Li et al. (2019) proposed a constrained two-archive evolutionary algorithm that maintains a convergence archive and a diversity archive simultaneously. This dual-archive design provides an effective way to balance convergence, diversity, and constraint satisfaction in constrained multi-objective optimization. Liu et al. (2025) developed a tri-population CMOEA that incorporates constrained Pareto dominance, which helps preserve infeasible solutions with good objective functions. When combined with a diversity archive, this design supports broad exploration of the objective space. Gao et al. (2026) proposed a multiknowledge adaptive transfer framework that incorporates the synchronization knowledge in decision space and evolutionary direction knowledge. Similarly, multi-stage methods handle CMOPs by adopting different optimization strategies at different stages of the evolutionary process. The push-pull search (PPS) framework (Fan et al., 2019) is a representative example. In the push stage, constraints are ignored to drive rapid convergence, and feasibility is gradually emphasized in the pull stage to guide the population toward the CPF. To further enhance diversity maintenance, Ma et al. (2021) proposed another multi-stage approach. Constraints are ignored in the first stage, and additional constraints are progressively introduced in later stages according to an assessed priority order.

Hybrid approaches that integrate multi-population and multi-stage designs have received increasing attention in recent years. To exploit computational resources more effectively, CMEGL (Qiao et al., 2023) divides a tri-population framework into two stages. In the first stage, the main task and two auxiliary tasks evolve jointly. After the UPF is considered to be sufficiently approximated, optimization of the global auxiliary task is terminated and its computational budget is reassigned to the main task and a local auxiliary task. Ming et al. (2022) proposed a two-stage CMOEA with three populations. The three populations evolve independently in the first stage, and knowledge transfer is performed in the second stage to enhance the effectiveness of shared information.

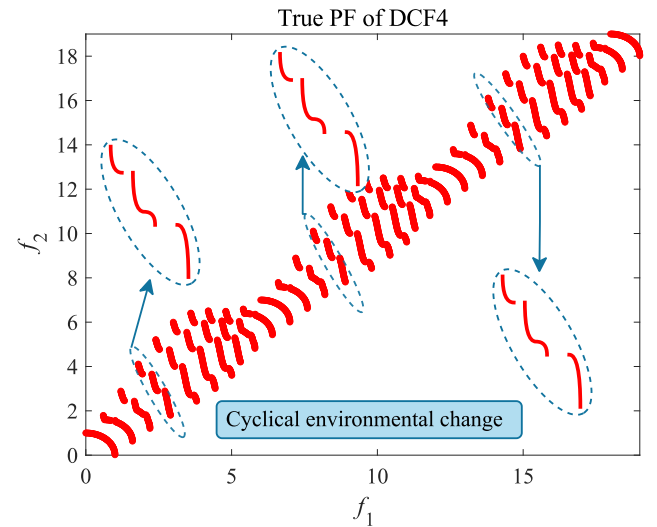
3.3. Motivation

DCMOPs involve time-varying objectives and constraints, requiring algorithms to continuously track DCPFs. Existing studies mainly enhance population adaptability through diversity introduction, historical solution reuse, and prediction techniques. However, the effectiveness of these strategies depends heavily on the characteristics of environmental changes. For example, prediction-based methods are generally more suitable when environmental changes exhibit exploitable regularity, memory-based methods can be beneficial when similar or cyclic environments reappear (Peng et al., 2015), and diversity-based strategies are often used to mitigate population diversity loss caused by abrupt or irregular changes (Liang et al., 2019). As shown in Fig. 1, even within the same problem, environmental change patterns may vary over time, highlighting the need for adaptive response strategies that consider change intensity.

In addition to dynamic response, the performance of DCMOEAs relies on the capability of their static optimizers. Complex constraints in CMOPs often partition the feasible region into multiple disconnected subregions, which can slow convergence and reduce diversity. Multi-population frameworks, typically combining a main population, a convergence-focused population, and a diversity-focused population, have shown promise for solving this issue (Liu et al., 2025). However, existing methods often allocate resources uniformly, ignoring that the value of auxiliary populations varies across optimization stages. In the



(a) The intensity of environmental changes varies in DCMOPs.



(b) Environmental changes in DCMOPs exhibit a cyclical pattern.

Fig. 1. The DCPFs identified in each environment are arranged in the objective space to provide a more intuitive representation of the patterns of environmental change.

early stages of optimization, convergence is more critical, whereas in the later stages, maintaining diversity is more important (Qiao et al., 2023). Similarly, after environmental changes, the algorithm should adaptively emphasize convergence or exploration depending on the intensity of change.

These observations motivate the development of a DCMOE that leverages historical knowledge to generate environment-adaptive initial populations and dynamically adjusts auxiliary population sizes based on detected change intensity. Therefore, this study proposes an adaptive memory-prediction fusion mechanism coupled with a dynamic tri-population optimization framework. This design aims to improve both the efficiency of DCPF exploration and the utilization of computational resources under varying environmental dynamics.

4. Proposed algorithm

4.1. Framework of AMPF

The overall framework of the proposed AMPF is illustrated in Fig. 2. To respond effectively to environmental changes, AMPF integrates an

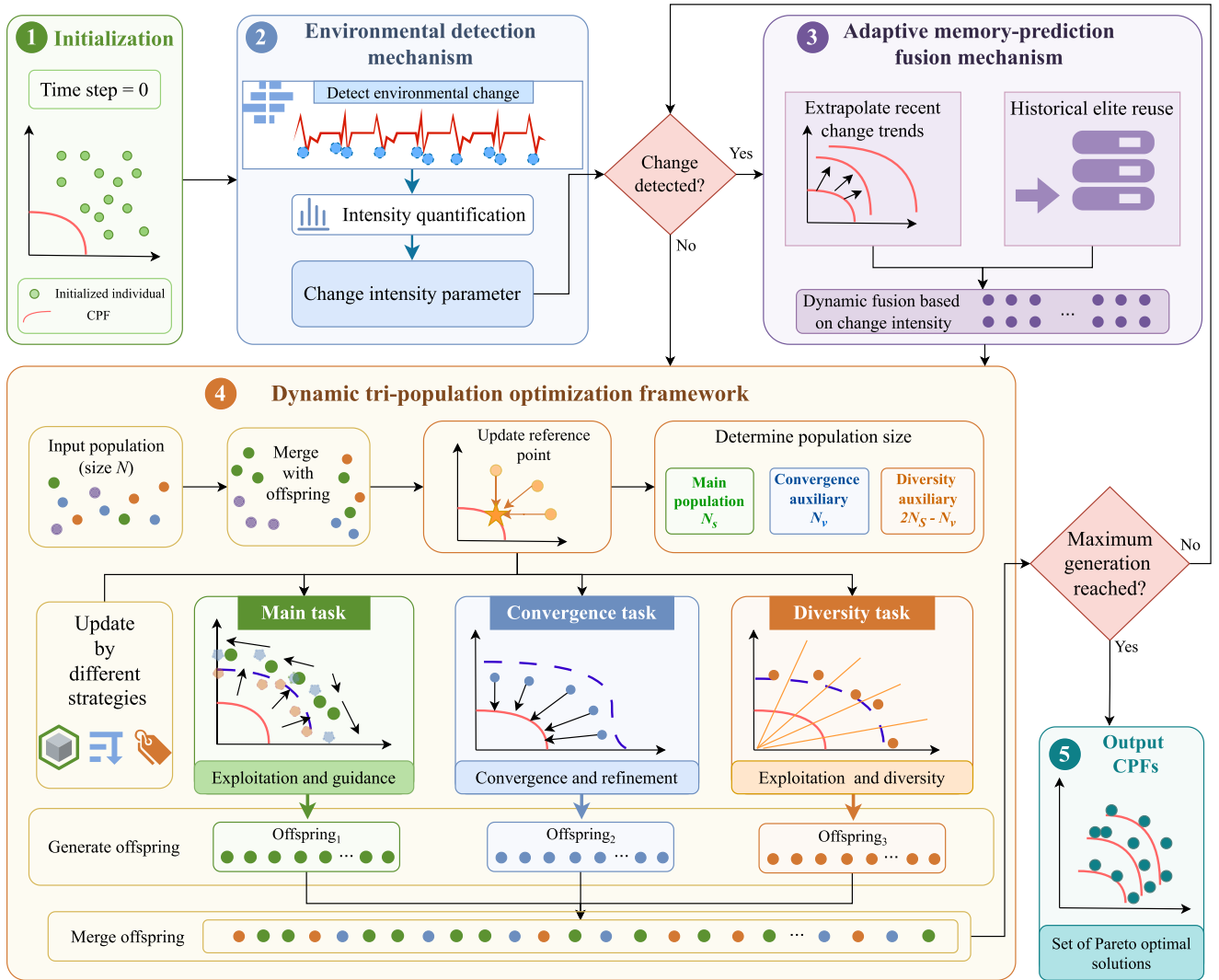


Fig. 2. The overall framework of the proposed AMPF.

adaptive memory-prediction fusion mechanism for dynamic response with a dynamic tri-population optimization framework for static search. When an environmental change is detected, the change intensity detection module first evaluates the variation between two consecutive environments and computes the change intensity parameter C_i . Then, this parameter is used to guide both the construction of the initial population in the new environment and the allocation of search resources among auxiliary tasks. As shown in Algorithm 1, AMPF initializes the time step t , the subpopulation size N_s , the historical population archive $AllPop$, the predicted-objective history $PredObjHist$, and the initial population (lines 1–6). During the evolutionary process, if no change is detected, the population is directly evolved by the dynamic tri-population optimization framework (line 18). Once a change occurs, AMPF calculates C_i and updates the time step (lines 9–11). When $t < 3$, the number of historical environments is insufficient for reliable learning, and a new initial population is randomly generated to preserve diversity (line 13). When $t \geq 3$, the adaptive memory-prediction fusion mechanism is activated to construct a more informative initial population for the new environment (line 15). The adaptive memory-prediction fusion mechanism exploits two complementary sources of historical knowledge. These two sources are adaptively fused according to the detected change intensity and the reliability of recent-trend matching. In particular, AMPF relies more on trend prediction when the recent dynamics are reliable and the environmental change is mild. In contrast, when the environment changes dras-

tically or the learned trend becomes less trustworthy, AMPF increases the contribution of historical elite solutions and applies stronger perturbations to them, thereby enhancing diversity and improving adaptation to uncertain environmental changes.

After constructing the initial population, AMPF employs the dynamic tri-population optimization framework to evolve the population. This framework contains one main population and two auxiliary populations. The main population optimizes the DCMOP using constrained Pareto dominance, while the convergence auxiliary population ignores constraints and adopts Pareto dominance to drive the search toward the UPF. The diversity auxiliary population maintains population distribution through an association based on reference vectors, which helps preserve coverage across different regions of the objective space. The three populations exchange information through a shared pool to improve complementary search behaviors. Moreover, the sizes of the two auxiliary populations are dynamically adjusted according to C_i . The convergence auxiliary population is assigned N_v individuals, whereas the diversity auxiliary population is assigned $2N_s - N_v$ individuals. When the environmental change is severe, more resources are allocated to the convergence auxiliary task to accelerate adaptation to the new CPF. When the change is mild, more resources are assigned to the diversity auxiliary task to support finer and more uniform CPF exploration. This adaptive allocation strategy improves the utilization of computational resources under different environmental change patterns.

Algorithm 1 General framework of AMPF.

Input: $F(\vec{x}, t)$ (DCMOP), N (population size)
Output: CPSs (set of constrained Pareto optimal sets)

```

1: Set time step  $t = 0$ ;
2:  $N_s = \lfloor N/3 \rfloor$ ;
3:  $N_v = N_s$ ;
4:  $AllPop \leftarrow \emptyset$ ;
5:  $PredObjHist \leftarrow \emptyset$ ;
6:  $Population \leftarrow$  Generate an initial population of size  $N$ ;
7: while not terminated do
8:   if change detected then
9:      $C_i \leftarrow$  Calculate the change intensity by Eqs. (10) and (11);
10:     $N_v = \text{floor}(C_i \cdot 2N_s)$ ;
11:     $t = t + 1$ ;
12:    if  $t < 3$  then
13:       $Population \leftarrow$  Generate an initial population of size  $N$ ;
14:    else
15:       $Population, PredObjHist \leftarrow$  Apply the adaptive memory-prediction fusion mechanism;
16:    end if
17:  end if
18:  Execute the dynamic tri-population optimization framework;
19:   $AllPop \leftarrow AllPop \cup Population$ ;
20: end while

```

AMPF does not assume environmental changes are strictly periodic. The memory-prediction mechanism is used adaptively. When the number of historical environments is insufficient, AMPF does not activate prediction and instead generates a new initial population to preserve diversity. When sufficient historical information is available, AMPF adaptively fuses recent-trend prediction and historical elite solutions according to the detected change intensity and the reliability of recent-trend matching. When environmental changes are severe or the learned trend is unreliable, AMPF increases the contribution of perturbed historical elite solutions rather than relying only on direct trend prediction. Meanwhile, the dynamic tri-population framework continues the search after initialization, allowing the algorithm to correct inaccurate prediction through cooperation among populations. These designs improve robustness across diverse dynamic scenarios.

4.2. Change intensity detection module

When an environmental change occurs, AMPF estimates the change intensity by re-evaluating the same sampled individuals in two consecutive environments. The relative changes in objective vectors and constraint violations are calculated as follows:

$$dObj_i = \frac{\|\mathbf{f}_i^{new} - \mathbf{f}_i^{old}\|_2}{\|\mathbf{f}_i^{old}\|_2 + \varepsilon}, \quad (7)$$

$$dCV_i = \frac{|CV_i^{new} - CV_i^{old}|}{|CV_i^{old}| + \varepsilon}, \quad (8)$$

where $\mathbf{f}_i^{old}$ and $\mathbf{f}_i^{new}$ denote the objective vectors of the i th sampled individual evaluated in the old and new environments, respectively. Similarly, CV_i^{old} and CV_i^{new} denote the corresponding constraint violations in the old and new environments, respectively. ε is a small positive constant used to avoid division by zero.

The general form of the change intensity indicator is designed to take both objective and constraint variations into account. Specifically, it is defined as follows:

$$\hat{C}_i = \text{median}(\{dObj_i\}_{i=1}^n) + w_{CV} \cdot \text{median}(\{dCV_i\}_{i=1}^n), \quad (9)$$

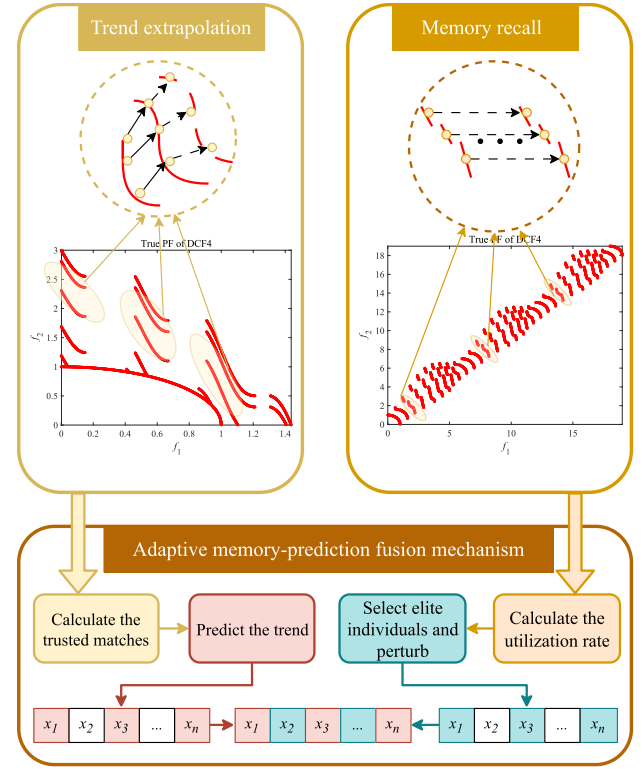


Fig. 3. Workflow of the adaptive memory-prediction fusion mechanism.

where n is the number of sampled individuals used for change detection and w_{CV} is the weight of constraint violation change. To examine the influence of w_{CV} , five variants are designed and tested in Section 5.3. The results show that $w_{CV} = 0.00$ provides the best overall performance. Therefore, the final version of the proposed algorithm uses objective space variation to estimate the environmental change intensity:

$$\hat{C}_i = \text{median}(\{dObj_i\}_{i=1}^n). \quad (10)$$

After normalization, the change intensity is scaled to the range $[0, 1]$:

$$C_i = \text{Norm}(\hat{C}_i), \quad (11)$$

where $\text{Norm}(\cdot)$ denotes the normalization operation. A larger value of C_i indicates a stronger environmental change, corresponding to a more drastic shift between two consecutive environments. In contrast, a smaller value indicates a milder environmental change.

Constraint violation is not included in the calculation of C_i in AMPF. The rationale is that the detected change intensity is mainly used to estimate the degree of population displacement in the objective space and to guide the dynamic response mechanism. The influence of constraints is handled in the subsequent constrained optimization process, especially through the main population and the auxiliary populations. This design avoids the instability caused by directly mixing objective space variation and constraint violation variation with different scales, while still allowing AMPF to respond to constraint changes during the evolutionary search.

4.3. Adaptive memory-prediction fusion mechanism

Fig. 3 shows the workflow of the proposed adaptive memory-prediction fusion mechanism. This mechanism aims to construct a high-quality initial population for the new environment by jointly exploiting recent change trends and information from perturbed historical elite solutions. Specifically, it learns the movement trend from the two most recent environments and extrapolates candidate solutions according to the

Algorithm 2 Adaptive memory-prediction fusion mechanism.

Input: *AllPop* (archived populations), *PredObjHist* (predicted-objective history), C_i (change intensity), N (population size)
Output: *Population*, *PredObjHist*

- 1: *Archive1* $\leftarrow$ Last N individuals of *AllPop*;
- 2: *Archive2* $\leftarrow$ Previous N individuals of *AllPop*;
- 3: $ind_x \leftarrow$ Match each individual with its nearest neighbor;
- 4: $valid \leftarrow$ Calculate the trusted matches by Eq. (12);
- 5: *Predec*, *Preobj* $\leftarrow$ First-order prediction model;
- 6: *Predec* $\leftarrow$ Retain the credible portion and use mean extrapolation for the remainder;
- 7: *PredObjHist* $\leftarrow$ *PredObjHist* $\cup$ *Preobj*;
- 8: $N_e \leftarrow$ Calculate the number of historical elites to be utilized based on C_i and $valid$ according to Eqs. (14) and (15);
- 9: *elites* $\leftarrow$ Select N_e optimal solutions from the most similar environment;
- 10: *elites* $\leftarrow$ Add perturbations to the original individuals;
- 11: *Population* $\leftarrow$ *Predec* $\cup$ *elites*;

detected change intensity. Meanwhile, it searches the historical archive to identify the environment most similar to the current one, from which elite solutions are selected and perturbed before being injected into the initial population. By adaptively coordinating trend-based prediction and perturbation-enhanced elite reuse, the proposed mechanism improves the adaptability and diversity of the initial population under different environmental change patterns.

The implementation of this mechanism is summarized in Algorithm 2. First, individuals from the two most recent environments are matched according to their nearest-neighbor relationships, and the reliability of the recent trend is estimated by the proportion of trusted matches. This reliability indicator is defined as follows:

$$valid = \frac{1}{N} \sum_{i=1}^N [d_{\min}(i) \leq MED + k_{MAD} \cdot MAD], \quad (12)$$

where $[\cdot]$ denotes the indicator function. $d_{\min}(i)$ represents the nearest-neighbor matching distance of the i th individual, and k_{MAD} is the threshold parameter to control the strictness of the validity criterion. MED and MAD are the median and median absolute deviation computed over the set $\{d_{\min}(i)\}_{i=1}^N$, respectively. A larger value of $valid$ indicates that more individuals can be reliably matched across consecutive environments, suggesting that the recent movement trend is more trustworthy. Based on the matched individuals, the first-order prediction model (Zhang et al., 2025b) is used to extrapolate the candidate solutions (line 5). When the environmental change is mild and the recent trend is reliable, the mechanism places more emphasis on trend-based prediction. In contrast, when the change is drastic or the matching reliability is low, the prediction step becomes more conservative. For unreliable components, mean-based extrapolation is adopted according to the change intensity, which helps avoid overly aggressive predictions.

The mechanism determines the number of historical elites to be selected and perturbed. Unlike a fixed-weight fusion strategy, the contribution of historical elites is dynamically determined by three factors: the detected change intensity C_i , the recent-trend reliability $valid$, and the confidence of the most similar historical environment. The confidence of the most similar historical environment is defined as follows:

$$confsim = \max \left(0, \frac{s_{med} - s_{\min}}{s_{med} + \varepsilon} \right), \quad (13)$$

where $s_{\min}$ denotes the minimum matching-distance score of the most similar historical environment, and s_{med} represents the median matching-distance score among all candidate historical environments. A larger $confsim$ indicates that the selected historical environment is more clearly distinguished from the other candidates.

The historical elite transfer rate is then calculated as follows:

$$\phi_e = confsim \cdot \min(1, C_i + (1 - valid)), \quad (14)$$

where ϕ_e denotes the transfer rate of historical elites. The number of historical elites to be transferred is calculated as follows:

$$N_e = \text{round}(\phi_e N), \quad (15)$$

where N is the population size and N_e is the number of historical elites selected from the most similar historical environment. When the environmental change is intense or the recent trend is unreliable, $C_i + (1 - valid)$ becomes larger, allowing more perturbed historical elites to participate in population initialization. When the selected historical environment has low confidence, $confsim$ becomes small and fewer historical elites are transferred.

The most similar historical environment is identified by comparing the predicted CPFs stored in *PredObjHist*. Then, the corresponding elite solutions are selected from this environment and perturbed under the guidance of the change intensity before being merged with the predicted population (lines 9–11). When the environmental change is mild and the recent matching reliability is high, N_e becomes small, and the mechanism mainly relies on recent trend prediction. In contrast, when the environmental change is intense or the recent trend is unreliable, more historical elites are selected if a similar environment can be identified with high confidence. The selected historical elites are further perturbed before being injected into the initial population. The perturbation strength is related to the detected change intensity. This design allows AMPF to reuse valuable historical search experience while introducing additional diversity, thereby reducing the risk of transferring outdated or overly concentrated solutions to the new environment. Finally, the perturbed historical elites and the predicted solutions are combined to form the complete initial population for subsequent optimization.

4.4. Dynamic tri-population optimization framework

The dynamic tri-population optimization framework is designed to balance feasibility, convergence, and diversity during the static search process. It consists of one main population and two auxiliary populations. The main population solves the original DCMOP and is responsible for maintaining feasible and high-quality solutions. The convergence auxiliary population ignores constraints and focuses on approaching the UPF, thereby providing objective-competitive solutions that can accelerate convergence. The diversity auxiliary population, in contrast, emphasizes the distribution of solutions by constructing reference vectors and associating each reference vector with its nearest solution. This reference-vector-based association encourages the survival of well-distributed individuals and helps preserve global search capability. To ensure the full sharing of knowledge, AMPF adopts a strong cooperation strategy. Candidate solutions from the three populations and their offspring are first merged into a shared pool, and each subpopulation is reselected from this pool according to its own selection criterion.

The implementation of the static optimizer is summarized in Algorithm 3. At each generation, the current *Population* is first merged with the *Offspring* generated in the previous generation, and the ideal point $Z_{\min}$ is updated accordingly (lines 2–3). Then, the populations are updated under different selection criteria to generate three different subpopulations. The main population *MainPop* is updated with respect to the original constrained task, where constrained Pareto dominance is adopted to prioritize feasible solutions while retaining promising infeasible ones. The convergence auxiliary population *ConPop* is updated by an unconstrained auxiliary task, where Pareto dominance is used to promote rapid movement toward the UPF. The diversity auxiliary population *DivPop* is updated by the diversity enhancement task, which maintains solution coverage through reference-vector-based association (lines 4–6). After population updating, the three populations generate N_s , N_v , and $2N_s - N_v$ offspring individuals, respectively. Then,

Table 1
Mean and standard deviation of MIGD obtained by AMPF and eight compared algorithms on two DCMOP test suites under various environmental settings.

Problem	n_f	τ_f	SKTEA	DCNSGAI-A	DCNSGAI-B	DCNSGA-III	MP-DR	DC-MOEA	HATC	TDCEA	AMPF									
DCF1	5	20	2.8339e-2	(8.70e-4)	-5.6939e-3	-2(2.68e-3)	-4.9588e-2	(4.12e-3)	-8.0685e-3	(5.54e-3)	-3.1922e-2	(1.09e-3)	-5.1246e-2	(3.26e-3)	-3.4523e-2	(4.34e-3)	-2.4911e-2	(9.47e-4)	-2.3203e-2	(4.04e-4)
	5	40	1.9502e-2	(5.91e-4)	-3.8599e-3	-2(2.83e-3)	-3.9253e-3	-2(2.50e-3)	-5.1028e-3	(3.61e-3)	-2.0794e-3	(1.03e-3)	-3.6594e-2	(2.97e-3)	-2.0965e-2	(2.95e-3)	-1.9111e-2	(2.55e-4)	-1.6620e-2	(4.03e-4)
	10	20	2.8086e-2	(1.38e-3)	-5.7212e-3	-2(3.84e-3)	-5.1606e-2	(7.82e-3)	-7.2006e-2	(5.30e-3)	-3.2686e-2	(1.60e-3)	-6.0361e-2	(7.20e-3)	-3.1663e-2	(2.94e-3)	-2.3933e-2	(2.58e-4)	-2.2854e-2	(4.12e-4)
	10	40	1.9867e-2	(6.66e-4)	-3.7950e-3	-2(2.61e-3)	-4.1620e-3	-2(6.57e-3)	-5.6097e-3	(2.93e-3)	-2.1577e-3	(1.06e-3)	-4.0531e-2	(2.98e-3)	-2.2760e-2	(8.67e-3)	-1.8738e-2	(6.93e-4)	-1.6434e-2	(3.42e-4)
	5	20	6.9494e-2	(9.29e-3)	-8.1086e-2	-2(4.33e-3)	-1.2956e-1	(9.65e-3)	-1.6170e-1	(1.09e-2)	-1.7309e-2	(4.87e-3)	-6.9157e-2	(1.40e-2)	-4.2393e-2	(8.74e-3)	-1.4624e-2	(1.93e-3)	-1.1005e-2	(5.70e-4)
DCF2	5	40	7.9775e-3	(3.11e-3)	-1.7696e-2	-2(2.86e-3)	-3.1196e-2	(5.28e-3)	-7.9671e-2	(5.71e-3)	-5.3882e-3	(1.14e-3)	-1.6762e-2	(3.08e-3)	-1.2442e-2	(2.87e-3)	-5.0738e-3	(1.65e-4)	-4.4281e-3	(8.31e-5)
	10	20	3.9666e-2	(7.88e-3)	-7.4698e-2	-2(5.07e-3)	-4.8894e-2	(6.81e-3)	-1.1097e-1	(7.40e-3)	-3.0868e-2	(5.16e-3)	-5.9758e-2	(1.31e-2)	-4.2019e-2	(7.18e-3)	-9.3980e-3	(4.81e-4)	-7.4788e-3	(2.92e-4)
	10	40	6.3581e-3	(3.15e-3)	-1.7726e-2	-2(3.23e-3)	-1.8438e-2	(3.07e-3)	-4.7487e-2	(7.02e-3)	-6.5948e-3	(2.14e-3)	-1.4130e-2	(2.05e-3)	-9.6306e-3	(2.31e-3)	-4.6424e-3	(4.62e-5)	-4.1182e-3	(5.43e-5)
	5	20	1.7529e-2	(1.42e-3)	-1.5934e-1	(7.97e-3)	-1.0379e-1	(6.87e-3)	-3.0459e-1	(1.60e-2)	-1.2436e-2	(9.15e-4)	-1.5237e-1	(4.07e-2)	-1.4003e-2	(1.38e-3)	-1.6492e-2	(9.43e-4)	-1.1293e-2	(3.97e-4)
	5	40	7.5242e-3	(2.13e-4)	-5.2025e-2	-2(2.72e-3)	-2.3973e-2	(1.45e-3)	-1.0502e-1	(4.88e-3)	-7.0446e-3	(6.06e-4)	-3.0978e-2	(3.12e-3)	-7.7036e-3	(8.72e-4)	-7.4472e-3	(1.02e-4)	-5.9917e-3	(1.08e-4)
DCF3	10	20	1.1540e-2	(5.25e-4)	-1.6009e-1	(6.78e-3)	-3.5395e-2	(2.16e-3)	-1.6392e-1	(1.50e-2)	-1.0001e-2	(8.92e-4)	-1.9594e-1	(6.09e-2)	-1.1131e-2	(1.16e-3)	-9.9814e-3	(2.49e-4)	-7.4754e-3	(2.14e-4)
	10	40	6.2237e-3	(1.12e-4)	-5.1455e-2	-2(4.27e-3)	-1.4593e-2	(1.09e-3)	-3.5660e-2	(2.64e-3)	-6.5904e-3	(4.32e-4)	-1.5522e-2	(1.19e-3)	-7.4050e-3	(7.88e-4)	-6.5414e-3	(1.00e-4)	-5.1839e-3	(7.43e-5)
	5	20	4.0237e-2	(1.41e-2)	-2.9296e-1	(3.73e-2)	-7.6850e-2	(1.14e-2)	-1.6761e-1	(3.37e-2)	-8.8227e-2	(2.56e-2)	-9.3728e-2	(2.24e-2)	-2.2285e-1	(7.18e-2)	-2.899e-2	(2.21e-3)	-2.0867e-2	(5.66e-4)
	5	40	1.7385e-2	(1.60e-3)	-2.5216e-1	(2.66e-2)	-6.6483e-2	(1.579e-2)	-4.0324e-2	(1.48e-2)	-7.3217e-2	(1.25e-2)	-1.6662e-1	(6.73e-2)	-1.5887e-2	(2.156e-3)	-1.5887e-2	(2.156e-3)	-1.3885e-2	(2.66e-4)
	10	20	5.7223e-2	(5.15e-2)	-2.8747e-1	(2.18e-2)	-7.6152e-2	(1.28e-2)	-1.7431e-1	(8.25e-2)	-6.0270e-2	(1.59e-2)	-9.1150e-2	(1.31e-2)	-2.6848e-1	(1.04e-1)	-1.9950e-2	(2.164e-3)	-1.9573e-2	(4.96e-4)
DCF4	10	40	1.6420e-2	(2.10e-3)	-2.3560e-1	(2.94e-2)	-6.4222e-2	(1.47e-2)	-1.3515e-1	(4.33e-2)	-3.5038e-2	(9.23e-3)	-7.7820e-2	(1.43e-2)	-1.4807e-1	(6.79e-2)	-1.4505e-2	(2.788e-4)	-1.3631e-2	(3.19e-4)
	5	20	4.812e-2	(2.81e-3)	-1.2771e-1	(6.36e-3)	-1.1452e-1	(7.49e-3)	-7.7036e-1	(6.71e-2)	-1.3343e-2	(1.16e-3)	-1.9145e-1	(5.60e-2)	-1.5999e-2	(1.55e-3)	-6.6816e-2	(8.75e-3)	-1.6524e-2	(6.50e-4)
	5	40	7.6063e-2	(1.72e-4)	-4.1263e-2	-2(6.15e-3)	-3.7424e-2	(4.56e-3)	-3.0242e-1	(2.64e-2)	-6.7820e-3	(7.29e-4)	-4.5646e-2	(9.67e-3)	-8.0276e-3	(9.96e-4)	-8.6957e-3	(3.95e-4)	-6.3919e-3	(1.20e-4)
	10	20	1.4442e-2	(5.36e-4)	-1.2656e-1	(8.05e-3)	-4.9387e-2	(3.50e-3)	-7.1074e-1	(6.93e-2)	-1.0776e-2	(1.46e-3)	-1.1783e-1	(4.86e-2)	-1.2219e-2	(1.08e-3)	-1.1274e-2	(4.05e-4)	-9.8082e-3	(2.91e-4)
	10	40	6.3344e-3	(9.44e-5)	-4.3219e-2	-2(3.67e-3)	-1.2331e-1	(1.1027e-1)	-1.1027e-1	(7.28e-4)	-6.2395e-3	(7.28e-4)	-2.2953e-2	(5.22e-3)	-7.5851e-3	(1.17e-3)	-5.8549e-3	(7.04e-5)	-5.6750e-3	(8.43e-5)
DCF6	5	20	3.6460e-2	(6.15e-3)	-2.2454e-1	(1.71e-2)	-2.2115e-1	(2.83e-2)	-4.4262e-1	(5.85e-2)	-1.4678e-2	(5.96e-3)	-3.1370e-1	(1.59e-1)	-3.3072e-2	(1.31e-2)	-5.4528e-2	(8.10e-3)	-1.2181e-2	(7.05e-4)
	5	40	1.2427e-2	(5.56e-4)	-1.3571e-1	(1.89e-2)	-1.1303e-1	(2.47e-2)	-7.4229e-1	(2.45e-1)	-6.4857e-3	(3.48e-3)	-1.4410e-1	(1.83e-2)	-2.2074e-2	(8.54e-3)	-9.9397e-3	(6.12e-4)	-5.4527e-3	(1.43e-4)
	10	20	1.3579e-2	(9.16e-4)	-2.2229e-1	(2.13e-2)	-4.4274e-2	(4.77e-3)	-3.6136e-1	(8.89e-1)	-1.8520e-2	(7.73e-3)	-2.5595e-1	(1.55e-1)	-4.0085e-2	(2.16e-2)	-1.2240e-2	(4.17e-4)	-7.4140e-3	(2.06e-4)
	10	40	5.8452e-3	(1.83e-4)	-1.2487e-1	(1.95e-2)	-2.0422e-2	(7.86e-3)	-2.9888e-1	(9.78e-2)	-1.7054e-2	(8.50e-3)	-7.0143e-1	(2.21e-2)	-3.2291e-3	(1.45e-2)	-5.9054e-3	(1.25e-4)	-4.1925e-3	(9.74e-5)
	5	20	1.9444e-2	(1.89e-3)	-1.6201e-1	(1.91e-2)	-1.1804e-1	(9.46e-3)	-6.2919e-1	(9.56e-2)	-1.5411e-2	(2.12e-3)	-4.1855e-1	(6.62e-2)	-2.0874e-2	(2.71e-3)	-1.3655e-2	(8.45e-4)	-1.3716e-2	(1.26e-4)
DCF7	5	40	1.0367e-2	(3.81e-4)	-4.1349e-2	-2(4.31e-3)	-2.8806e-2	(2.44e-3)	-1.2398e-1	(1.62e-2)	-1.0377e-2	(1.07e-3)	-4.8744e-2	(7.84e-3)	-1.5024e-2	(1.75e-3)	-9.9294e-3	(3.227e-4)	-9.6784e-3	(3.57e-4)
	10	20	1.1815e-2	(7.67e-4)	-1.5897e-1	(1.41e-2)	-3.8367e-2	(3.63e-3)	-5.5937e-2	(7.66e-3)	-1.6492e-2	(6.48e-3)	-4.2792e-1	(7.47e-2)	-1.9268e-2	(2.66e-3)	-9.7006e-3	(2.40e-4)	-9.3864e-3	(4.76e-4)
	10	40	4.2051e-3	(2.53e-4)	-3.6809e-2	-2(3.93e-3)	-1.4743e-2	(1.18e-3)	-4.3225e-2	(7.43e-3)	-9.7724e-3	(1.14e-3)	-4.2539e-2	(6.36e-3)	-1.3735e-2	(1.73e-3)	-8.4472e-3	(1.79e-4)	-8.1066e-3	(2.60e-4)
	5	20	1.0839e-2	(4.04e-4)	-9.7168e-2	-2(1.43e-2)	-6.9674e-2	(4.10e-2)	-1.6499e-1	(3.37e-2)	-1.1223e-2	(1.47e-3)	-8.3985e-2	(2.12e-2)	-1.1768e-2	(1.18e-3)	-9.2363e-3	(1.15e-4)	-9.8335e-3	(1.18e-4)
	5	40	7.7748e-3	(6.77e-5)	-2.8232e-2	-2(5.36e-3)	-1.7518e-2	(1.32e-2)	-4.6154e-2	(1.23e-2)	-7.3560e-3	(3.25e-4)	-1.6616e-2	(2.39e-3)	-7.8919e-3	(6.15e-4)	-7.4535e-3	(3.46e-5)	-7.3023e-3	(7.98e-5)
DCF9	10	20	1.0420e-2	(1.87e-4)	-9.5933e-2	-2(1.64e-2)	-2.5290e-2	(9.93e-3)	-5.5937e-2	(7.66e-3)	-1.6492e-2	(6.48e-3)	-4.4166e-2	(1.95e-2)	-1.6444e-2	(5.73e-3)	-8.8895e-3	(1.18e-4)	-9.3575e-3	(1.06e-4)
	10	40	7.7638e-3	(9.67e-5)	-2.6848e-2	-2(4.97e-3)	-1.2494e-2	(2.84e-3)	-2.6621e-2	(5.18e-3)	-1.8477e-3	(1.16e-3)	-1.5842e-2	(4.42e-3)	-8.2413e-3	(9.27e-4)	-7.4399e-3	(5.55e-5)	-7.0600e-3	(6.06e-5)
	5	20	6.4048e-2	(3.43e-2)	-2.3902e-1	(1.95e-2)	-2.2426e-1	(9.97e-3)	-3.5101e-1	(1.97e-2)	-1.8537e-2	(3.18e-3)	-2.0478e-1	(1.70e-2)	-2.6875e-2	(1.64e-2)	-1.2642e-2	(6.38e-4)	-1.1018e-2	(5.20e-4)
	5	40	6.3365e-2	(3.05e-2)	-2.3071e-1	(1.97e-2)	-1.3307e-1	(7.86e-3)	-1.8568e-1	(1.48e-2)	-1.0905e-2	(2.195e-3)	-6.9808e-2	(1.08e-2)	-1.3061e-2	(4.98e-3)	-6.4842e-3	(5.91e-5)	-6.1680e-3	(9.23e-5)
	10	20	6.9846e-2	(3.52e-2)	-2.4424e-1	(1.88e-2)	-8.1183e-2	(3.15e-2)	-2.0802e-1	(1.95e-2)	-1.9068e-2	(3.95e-3)	-1.9351e-1	(1.93e-2)	-2.1291e-2	(1.04e-2)	-9.1610e-3	(2.10e-4)	-9.0350e-3	(2.20e-4)
DCF10	10	40	6.1666e-2	(3.40e-2)	-9.0811e-2	-2(1.33e-2)	-1.9952e-2	(4.68e-3)	-7.7873e-2	(1.85e-2)	-1.1938e-2	(3.12e-3)	-2.7319e-2	(1.22e-2)	-1.4379e-2	(5.54e-3)	-6.8598e-3	(1.26e-4)	-6.4002e-3	(1.95e-4)
	5	20	9.6785e-2	(3.64e-2)	-4.9463e-1	(3.62e-2)	-5.3129e-1	(2.29e-2)	-1.2217e-1	(7.76e-2)	-3.3778e-2	(3.16e-3)	-6.2872e-1	(5.15e-2)	-3.7893e-2	(3.22e-2)	-5.3481e-2	(5.61e-3)	-2.9273e-2	(1.88e-3)

Table 1
Continued of Table 1.

Problem	τ_1	SKTEA	DCNSGAIL-A	DCNSGAIL-B	DCNSGA-III	MP-DFR	DC-MOEA	HATC	TDCEA	AMPF		
DCP1	5	40	2.8742e-2 (2.29e-2)	1.3106e-1 (9.89e-3)	2.4259e-1 (2.49e-2)	5.3552e-1 (2.86e-2)	9.8303e-3 (1.24e-3)	1.9516e-1 (3.09e-2)	1.2531e-2 (2 (1.38e-3)	1.1942e-2 (6.25e-4)	9.0261e-3 (3.53e-4)	
	10	20	5.5385e-2 (1.76e-2)	4.9209e-1 (2.99e-2)	2.9481e-1 (3.27e-2)	7.5946e-1 (4.44e-2)	2.1207e-2 (3.11e-3)	6.4658e-1 (7.63e-2)	2.5517e-2 (3.32e-3)	2.7808e-2 (1.58e-3)	1.6125e-2 (9.25e-4)	
	5	20	2.3169e-2 (9.96e-3)	1.2590e-1 (1.18e-2)	1.3273e-1 (1.19e-2)	3.3558e-1 (3.51e-2)	8.689e-3 (5.54e-4)	1.3938e-1 (2.23e-2)	1.1632e-2 (2 (1.95e-3)	9.4313e-3 (2.68e-4)	6.9567e-3 (2.53e-4)	
	5	20	7.2066e-1 (1.46e-1)	3.4487e-1 (2.49e-2)	2.1570e-1 (1.23e-2)	4.4108e-1 (2.94e-2)	1.8177e-2 (2.22e-3)	5.8784e-1 (4.69e-2)	2.3109e-2 (3.62e-3)	1.6518e-2 (9.80e-4)	1.1761e-2 (6.01e-4)	
	5	40	8.7920e-2 (2.14e-2)	9.8406e-1 (7.82e-3)	3.9312e-2 (2.46e-3)	2.3069e-1 (1.53e-2)	2.2882e-3 (8.72e-4)	2.0347e-1 (1.92e-2)	6.6238e-3 (3.16e-3)	4.6652e-3 (3.16e-4)	4.6082e-3 (1.33e-4)	
	10	20	6.5105e-2 (2.33e-2)	3.5207e-1 (2.47e-2)	6.3087e-2 (9.09e-3)	2.7939e-1 (2.98e-2)	1.2883e-2 (2.27e-3)	6.6668e-1 (4.74e-2)	1.4604e-2 (2.72e-3)	8.8251e-3 (3.16e-4)	8.0073e-3 (4.14e-4)	
	10	40	8.9173e-3 (2.44e-3)	9.5504e-2 (7.99e-3)	1.7299e-2 (2.44e-3)	9.2926e-2 (1.12e-2)	4.8809e-3 (9.26e-4)	2.2755e-1 (1.91e-2)	6.4167e-3 (3.14e-3)	3.7432e-3 (8.22e-5)	3.4181e-3 (9.69e-5)	
	5	20	8.3448e-2 (5.89e-3)	3.9457e-1 (1.53e-2)	3.8817e-2 (1.18e-2)	6.6813e-1 (2.73e-2)	3.7061e-3 (6.59e-3)	3.9637e-1 (4.01e-2)	4.5330e-2 (9.57e-3)	4.7320e-2 (6.27e-3)	2.8555e-2 (2.40e-3)	
	5	40	3.9526e-2 (5.83e-3)	1.7061e-1 (1.81e-2)	2.5645e-1 (1.35e-2)	3.8461e-1 (2.26e-2)	2.3034e-2 (6.40e-3)	2.1527e-1 (1.90e-2)	2.7339e-2 (6.64e-3)	1.0292e-2 (2.181e-3)	8.0926e-3 (1.27e-3)	
	10	20	9.5274e-2 (6.70e-3)	3.8231e-1 (1.69e-2)	2.8590e-1 (1.27e-2)	4.8460e-1 (2.88e-2)	3.1855e-2 (2.42e-3)	4.1627e-1 (8.42e-2)	2.34461e-2 (7.33e-3)	2.4056e-2 (3.59e-3)	1.3369e-2 (2 (1.99e-3)	
DCP2	10	40	6.9134e-2 (9.36e-3)	1.7338e-1 (1.48e-2)	1.8308e-1 (1.76e-2)	3.0555e-1 (2.51e-2)	2.4503e-2 (5.23e-3)	2.0950e-1 (2.67e-2)	2.6510e-2 (8.48e-3)	1.0853e-2 (2.77e-3)	7.6704e-3 (2.07e-3)	
	5	20	1.4708e-1 (0 (5.75e-2)	1.5616e-1 (0 (2.64e-2)	1.6368e-1 (0 (2.95e-2)	0 (4.56e-2)	8.2622e-1 (5.86e-2)	1.5183e-1 (0 (8.18e-2)	1.3601e-1 (0 (1.81e-1)	1.2342e-1 (0 (3.27e-2)	6.6423e-1 (1.432e-2)	
	5	40	1.2578e-1 (0 (5.64e-2)	1.4551e-1 (0 (3.85e-2)	1.5867e-1 (0 (4.35e-2)	1.6623e-1 (0 (1.77e-2)	3.2697e-1 (6.27e-2)	1.3833e-1 (0 (9.43e-2)	8.2381e-1 (2.60e-1)	7.9696e-1 (1.363e-2)	1.6984e-1 (2.86e-2)	
	10	20	1.5176e-1 (0 (5.48e-2)	1.5494e-1 (0 (3.46e-2)	1.6229e-1 (0 (3.71e-2)	1.6696e-1 (0 (1.81e-2)	7.6267e-1 (3.68e-2)	1.6056e-1 (0 (4.94e-2)	1.0567e-1 (0 (1.53e-1)	9.5848e-1 (4.63e-2)	1.61643e-1 (1.465e-2)	
	10	40	1.3320e-1 (0 (6.66e-2)	1.4640e-1 (0 (3.83e-2)	1.6010e-1 (0 (4.16e-2)	1.6085e-1 (0 (3.23e-2)	3.8262e-1 (6.63e-2)	1.4856e-1 (0 (6.25e-2)	6.4469e-1 (9.04e-2)	5.6252e-1 (3.54e-2)	2.3414e-1 (3.51e-2)	
	5	20	3.8346e-3 (5.07e-5)	3.3289e-1 (1.76e-2)	2.2202e-2 (6.30e-3)	8.8959e-3 (4.42e-3)	8.5251e-3 (3.73e-3)	1.1034e-2 (5.36e-3)	1.0373e-2 (4.40e-3)	2.9303e-3 (3.103e-4)	2.7585e-3 (1.20e-5)	
	5	40	3.4967e-3 (1.48e-5)	1.1565e-1 (1.42e-2)	7.8708e-3 (4.42e-3)	3.7256e-2 (2.131e-2)	8.2501e-3 (3.73e-3)	1.1034e-2 (5.36e-3)	1.0373e-2 (4.40e-3)	2.9303e-3 (3.103e-4)	2.7585e-3 (1.20e-5)	
	10	20	3.7699e-3 (7.11e-5)	3.1454e-1 (1.61e-2)	2.7358e-2 (4.85e-3)	1.1754e-1 (1.11e-2)	9.5049e-3 (3.91e-3)	2.2116e-2 (7.61e-3)	1.2707e-2 (4.75e-3)	2.8099e-3 (1.21e-5)	2.9278e-3 (1.140e-5)	
	10	40	3.3976e-3 (1.38e-5)	1.0886e-1 (9.68e-3)	9.8453e-3 (4.27e-3)	6.3373e-2 (8.40e-3)	3.6466e-3 (5.61e-4)	1.0997e-2 (3.72e-3)	4.7503e-3 (1.15e-3)	2.7908e-3 (8.47e-6)	2.6390e-3 (1.02e-5)	
	DCP5	5	20	1.8848e-1 (5.36e-3)	6.0001e-1 (1.94e-2)	2.0733e-1 (1.73e-2)	2.3745e-1 (1.67e-2)	1.6029e-1 (4.67e-2)	2.0261e-1 (1.63e-2)	2.0076e-1 (2.33e-2)	5.5962e-2 (2 (1.79e-3)	7.5182e-2 (2 (1.87e-2)
5		40	1.6307e-1 (7.11e-3)	3.1851e-1 (1.27e-2)	1.8720e-1 (6.82e-3)	2.0780e-1 (5.92e-3)	1.4376e-1 (4.43e-2)	2.8903e-1 (1.59e-2)	2.1951e-1 (3.47e-2)	3.8938e-2 (2.15e-3)	4.6997e-2 (2 (1.80e-3)	
10		20	1.9059e-1 (8.59e-3)	5.8544e-1 (1.68e-2)	2.1065e-1 (8.84e-3)	2.4892e-1 (1.42e-2)	1.4772e-1 (4.42e-2)	2.0650e-1 (1.41e-2)	1.9810e-1 (3.01e-2)	3.8504e-2 (2.12e-3)	6.6494e-2 (2 (1.28e-3)	
10		40	1.6342e-1 (1.04e-2)	3.1066e-1 (1.23e-2)	2.0019e-1 (3.16e-2)	2.1789e-1 (1.30e-2)	1.3076e-1 (4.86e-2)	1.9235e-1 (1.75e-2)	2.1994e-1 (1.78e-2)	3.2568e-2 (2 (1.44e-3)	4.4772e-2 (2 (1.30e-2)	
5		20	2.0373e-2 (8.71e-4)	1.5229e-1 (1.75e-2)	1.0466e-1 (4.55e-2)	2.3690e-1 (4.69e-2)	2.8314e-2 (4.51e-3)	8.2693e-2 (1.29e-2)	4.9850e-2 (2 (1.53e-2)	1.4162e-2 (2 (8.85e-4)	1.3516e-2 (2 (3.93e-4)	
5		40	1.2049e-2 (3.40e-4)	4.1947e-2 (5.91e-2)	6.9432e-2 (3.17e-2)	2.2087e-1 (5.78e-2)	1.2683e-1 (2.170e-3)	6.9588e-2 (9.68e-3)	2.7019e-2 (2.00e-2)	1.0201e-2 (2 (8.77e-4)	9.1797e-3 (2 (1.11e-4)	
10		20	2.0308e-2 (1.94e-3)	1.4555e-1 (1.32e-2)	1.1710e-1 (3.24e-2)	2.3251e-1 (7.16e-2)	2.8918e-2 (5.54e-3)	1.0675e-1 (1.70e-2)	7.4480e-2 (4.20e-2)	1.3069e-2 (2 (2.10e-4)	1.2818e-2 (2 (6.19e-4)	
10		40	1.1468e-2 (5.38e-4)	4.2798e-2 (6.85e-3)	1.2014e-1 (3.10e-2)	2.2119e-1 (7.04e-2)	1.2659e-2 (2.36e-3)	7.2504e-2 (1.23e-2)	2.1620e-2 (1.49e-2)	9.7939e-3 (3 (1.80e-4)	8.8788e-3 (3 (1.80e-4)	
DCP7		5	20	2.6453e-2 (6.33e-3)	1.8482e-1 (1.83e-2)	3.3610e-1 (2.78e-2)	3.6045e-1 (2.48e-2)	1.0158e-2 (6.79e-4)	3.4190e-1 (4.32e-2)	1.7897e-2 (2 (8.83e-3)	5.6290e-2 (7 (3.7e-3)	9.7188e-3 (4 (7.2e-4)
		5	40	7.9116e-3 (6.77e-4)	1.0496e-1 (1.00e-2)	1.7943e-1 (1.58e-2)	2.2786e-1 (1.77e-2)	5.0066e-3 (3.20e-4)	2.2036e-1 (1.69e-2)	9.7738e-3 (3.22e-3)	4.3341e-2 (3 (3.06e-3)	4.3463e-3 (3 (7.52e-5)
	10	20	2.1255e-2 (8.85e-4)	2.0257e-1 (2.19e-2)	1.3684e-1 (1.07e-2)	2.2658e-1 (1.26e-2)	7.6357e-3 (7.20e-4)	4.3897e-1 (5.23e-2)	1.3333e-2 (2 (2.56e-3)	5.1887e-2 (3 (3.02e-3)	6.0958e-3 (2 (3.23e-4)	
	10	40	4.6831e-3 (1.99e-4)	1.1919e-1 (1.73e-2)	5.3996e-2 (8.09e-3)	1.0693e-1 (9.17e-3)	4.4742e-3 (3.184e-4)	1.8025e-1 (3.05e-2)	9.1499e-2 (3 (2.31e-3)	4.1198e-2 (2 (3.4e-3)	3.5247e-3 (3 (6.92e-5)	
	5	20	6.5881e-2 (1.65e-2)	1.7913e-1 (1.47e-2)	1.4784e-1 (1.24e-2)	5.5644e-1 (6.99e-2)	3.3357e-2 (2 (2.58e-2)	2.1905e-1 (5.69e-2)	9.9933e-2 (2 (1.96e-2)	1.4355e-2 (2 (1.25e-3)	3.1537e-2 (2 (3.23e-3)	
	5	40	3.3173e-2 (7.61e-3)	6.0784e-2 (7.69e-3)	6.7439e-2 (1.42e-2)	1.8565e-1 (3.09e-2)	2.4537e-2 (1.58e-2)	7.4925e-2 (3.04e-2)	6.3289e-2 (2 (1.90e-2)	1.1823e-2 (2 (1.10e-3)	1.2843e-2 (2 (1.38e-3)	
	10	20	7.2509e-2 (2.53e-2)	1.9378e-1 (2.36e-2)	9.9882e-2 (1.64e-2)	2.7459e-1 (4.24e-2)	3.6498e-2 (2.31e-2)	1.6149e-1 (3.27e-2)	8.0305e-2 (3 (1.9e-2)	1.6046e-2 (2 (2.51e-3)	2.4403e-2 (2 (3.35e-3)	
	10	40	3.6549e-2 (2 (1.08e-2)	6.2987e-2 (2 (1.00e-2)	9.8055e-2 (2 (2.61e-2)	1.4563e-1 (2.39e-2)	5.4257e-2 (3 (3.1e-2)	5.0559e-2 (2 (2.04e-2)	5.8979e-2 (2 (2.86e-2)	1.3161e-2 (2 (2.77e-3)	1.2632e-2 (2 (3.97e-3)	
	DCP9	5	20	1.8216e-1 (2.95e-2)	1.1130e-1 (9.48e-3)	3.2004e-1 (2.37e-2)	9.5693e-1 (7.34e-2)	5.6129e-2 (1.34e-2)	1.7056e-1 (2 (2.52e-2)	8.7767e-2 (2 (1.88e-2)	1.1397e-1 (2 (5.9e-2)	3.4689e-2 (2 (5.60e-3)
		5	40	6.4244e-2 (2 (2.76e-2)	3.6730e-2 (4.23e-3)	1.0215e-1 (1.49e-2)	5.2918e-1 (5.01e-2)	1.5552e-2 (1.32e-3)	4.9865e-2 (5 (4.7e-3)	2.6506e-2 (1.12e-2)	1.7934e-2 (2 (3.91e-3)	1.1361e-2 (2 (2.51e-4)
10		20	1.0293e-1 (2.90e-2)	1.0013e-1 (7.15e-3)	1.5493e-1 (1.16e-2)	6.8512e-1 (3.90e-2)	5.3526e-2 (2 (4.2e-3)	1.4618e-1 (2 (3.03e-2)	7.5218e-2 (2 (1.82e-2)	2.7604e-2 (1.12e-2)	2.2104e-2 (2 (4.45e-3)	
10		40	2.9614e-2 (1.87e-2)	3.8728e-2 (2 (2.91e-3)	4.7576e-2 (6.19e-3)	3.9016e-1 (3.80e-2)	1.4778e-2 (3 (4.2e-3)	5.3049e-2 (4 (81e-3)	3.2145e-2 (2 (1.57e-2)	1.1152e-2 (2 (5.06e-4)	1.0629e-2 (2 (2.87e-4)	
-		-	-	-	-	-	-	-	-	-	-	
-		-	-	-	-	-	-	-	-	-	-	
-		-	-	-	-	-	-	-	-	-	-	
-		-	-	-	-	-	-	-	-	-	-	
-		-	-	-	-	-	-	-	-	-	-	
-		-	-	-	-	-	-	-	-	-	-	
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-	-	-		

Table 2
Mean and standard deviation of MHV obtained by AMPF and eight compared algorithms on two DCMOP test suites under various environmental settings.

Problem	n_t	τ_t	SKTEA	DCNSGAIL-A	DCNSGAIL-B	DCNSGA-III	MP-DFR	DC-MOEA	HATC	TDCEA	AMPF
DCF1	5	20	3.5387e-1 (7.16e-4)	3.2787e-1 (3.08e-3)	3.4292e-1 (2.01e-3)	3.1868e-1 (3.36e-3)	3.5676e-1 (4.84e-4)	3.3974e-1 (1.81e-3)	3.5998e-1 (3.35e-3)	3.5757e-1 (2.91e-4)	3.5765e-1 (2.92e-4)
	5	40	3.6268e-1 (1.85e-4)	3.5085e-1 (1.86e-3)	3.5094e-1 (1.01e-3)	3.4178e-1 (1.67e-3)	3.6319e-1 (4.98e-4)	3.5353e-1 (1.81e-3)	3.6974e-1 (2.22e-3)	3.6260e-1 (2.01e-4)	3.6327e-1 (2.32e-4)
	10	20	3.5474e-1 (1.29e-3)	3.3859e-1 (2.87e-3)	3.3966e-1 (5.04e-3)	3.2214e-1 (3.38e-3)	3.5570e-1 (1.04e-3)	3.3261e-1 (3.51e-3)	3.6134e-1 (2.59e-4)	3.5832e-1 (3.19e-4)	3.5771e-1 (4.26e-4)
	10	40	3.6260e-1 (2.72e-4)	3.5066e-1 (1.10e-3)	3.4643e-1 (3.21e-3)	3.3781e-1 (3.21e-3)	3.6258e-1 (8.72e-4)	3.4992e-1 (2.41e-3)	3.6804e-1 (5.61e-3)	3.6281e-1 (1.75e-4)	3.6323e-1 (1.49e-4)
	5	20	3.8362e-1 (5.05e-3)	3.3060e-1 (5.22e-3)	2.3626e-1 (8.33e-3)	2.4642e-1 (3.32e-3)	3.4936e-1 (1.90e-2)	4.1133e-1 (4.98e-3)	4.2095e-1 (2.95e-3)	4.2686e-1 (9.02e-4)	4.2686e-1 (9.02e-4)
DCF2	5	40	3.4227e-1 (2.60e-3)	4.2185e-1 (2.44e-3)	4.0669e-1 (4.48e-3)	3.3753e-1 (5.03e-3)	4.3680e-1 (6.45e-4)	4.2334e-1 (2.36e-3)	4.3252e-1 (2.08e-3)	4.3565e-1 (2.88e-4)	4.3672e-1 (1.23e-4)
	10	20	4.0556e-1 (5.13e-3)	3.3859e-1 (6.05e-3)	3.8121e-1 (5.64e-3)	2.9330e-1 (8.00e-3)	4.1608e-1 (3.36e-3)	3.6032e-1 (1.78e-2)	4.0820e-1 (5.80e-3)	4.2915e-1 (8.94e-4)	4.3216e-1 (8.94e-4)
	10	40	4.3439e-1 (2.55e-3)	4.2211e-1 (2.61e-3)	4.2198e-1 (2.61e-3)	3.8167e-1 (6.32e-3)	4.3634e-1 (1.08e-3)	4.2661e-1 (1.82e-3)	4.3374e-1 (1.59e-3)	4.3644e-1 (8.00e-5)	4.3721e-1 (8.68e-5)
	5	20	3.3096e-1 (2.17e-3)	1.6787e-1 (8.82e-3)	2.2620e-1 (5.34e-3)	8.5738e-2 (5.70e-3)	3.3743e-1 (1.19e-3)	1.8398e-1 (2.63e-2)	3.3627e-1 (1.48e-3)	3.3357e-1 (1.28e-3)	3.3929e-1 (4.42e-4)
	5	40	3.4342e-1 (1.40e-4)	3.1285e-1 (1.89e-3)	3.2941e-1 (1.42e-3)	2.2521e-1 (4.80e-3)	3.4417e-1 (4.10e-4)	3.2480e-1 (2.88e-3)	3.4358e-1 (4.94e-4)	3.4367e-1 (8.50e-5)	3.4486e-1 (1.06e-4)
DCF3	10	20	3.3866e-1 (6.00e-4)	1.6601e-1 (7.69e-3)	3.0857e-1 (2.39e-3)	1.7358e-1 (1.16e-2)	3.3989e-1 (9.91e-4)	1.6042e-1 (3.57e-2)	3.3904e-1 (1.23e-3)	3.4118e-1 (2.51e-4)	3.4328e-1 (1.98e-4)
	10	40	3.4468e-1 (1.07e-4)	3.1361e-1 (2.90e-3)	3.3778e-1 (8.09e-4)	3.0986e-1 (2.45e-3)	3.4481e-1 (4.20e-4)	3.3731e-1 (8.50e-4)	3.4420e-1 (4.38e-4)	3.4440e-1 (7.42e-5)	3.4569e-1 (7.14e-5)
	5	20	4.6465e-1 (6.03e-3)	3.4931e-1 (1.09e-2)	4.4271e-1 (6.80e-3)	3.9253e-1 (1.36e-2)	4.4839e-1 (1.14e-2)	4.3616e-1 (9.06e-3)	3.8805e-1 (2.34e-2)	4.7536e-1 (2.57e-3)	4.7940e-1 (4.36e-4)
	5	40	4.8121e-1 (1.53e-3)	3.8185e-1 (8.47e-3)	4.5155e-1 (7.12e-3)	4.0753e-1 (2.13e-2)	4.7434e-1 (6.40e-3)	4.4847e-1 (5.36e-3)	4.1344e-1 (2.28e-2)	4.8137e-1 (1.79e-3)	4.8438e-1 (1.79e-3)
	10	20	4.7047e-1 (1.56e-2)	3.5715e-1 (6.10e-3)	4.5257e-1 (7.29e-3)	4.0444e-1 (2.46e-2)	4.7135e-1 (7.18e-3)	4.4613e-1 (4.86e-3)	3.8627e-1 (3.41e-2)	4.8580e-1 (1.96e-3)	4.8862e-1 (3.17e-4)
DCF4	10	40	4.8969e-1 (1.96e-3)	3.9132e-1 (9.79e-3)	4.6356e-1 (7.89e-3)	4.2570e-1 (1.91e-2)	4.8523e-1 (3.41e-3)	4.5578e-1 (5.82e-3)	4.2995e-1 (2.65e-2)	4.9067e-1 (1.18e-3)	4.9275e-1 (1.63e-4)
	5	20	4.9981e-1 (3.09e-3)	3.7819e-1 (4.50e-3)	3.9320e-1 (5.82e-3)	3.9253e-1 (1.36e-2)	4.4839e-1 (1.14e-2)	4.3616e-1 (9.06e-3)	3.8805e-1 (2.34e-2)	4.7536e-1 (2.57e-3)	4.7940e-1 (4.36e-4)
	5	40	5.2393e-1 (2.38e-4)	4.8426e-1 (4.66e-3)	4.8568e-1 (4.71e-3)	2.5790e-1 (1.53e-2)	5.2536e-1 (7.81e-4)	4.7931e-1 (8.16e-3)	5.2512e-1 (1.10e-3)	5.2233e-1 (5.55e-4)	5.2567e-1 (1.98e-4)
	10	20	5.1575e-1 (8.53e-4)	3.8248e-1 (6.81e-3)	4.6422e-1 (3.58e-3)	1.3171e-1 (4.97e-2)	5.2195e-1 (1.46e-3)	4.0689e-1 (3.07e-2)	5.2126e-1 (1.16e-3)	5.2087e-1 (5.93e-4)	5.2282e-1 (4.18e-4)
	10	40	5.2819e-1 (1.38e-4)	4.8522e-1 (3.08e-3)	5.1858e-1 (1.83e-3)	4.1470e-1 (1.66e-2)	5.2870e-1 (7.13e-4)	5.0627e-1 (5.50e-3)	5.2826e-1 (1.01e-3)	5.2880e-1 (1.13e-4)	5.2906e-1 (1.47e-4)
DCF5	5	20	3.0073e-1 (4.72e-3)	1.5838e-1 (6.48e-3)	1.8077e-1 (8.26e-3)	6.3848e-3 (2.80e-3)	3.2575e-1 (1.65e-3)	1.5610e-1 (2.66e-2)	5.1369e-1 (1.38e-3)	4.4520e-1 (6.57e-3)	3.2505e-1 (9.77e-4)
	5	40	3.2699e-1 (3.92e-3)	2.4383e-1 (6.18e-3)	2.6217e-1 (9.29e-3)	8.5286e-2 (1.37e-2)	3.3402e-1 (1.24e-3)	2.4374e-1 (8.81e-3)	3.2767e-1 (2.95e-3)	3.2811e-1 (7.40e-4)	3.3559e-1 (2.00e-4)
	10	20	3.2203e-1 (1.10e-3)	1.5899e-1 (7.92e-3)	2.7676e-1 (4.51e-3)	2.1525e-2 (2.43e-2)	3.2810e-1 (2.25e-3)	1.8479e-1 (4.07e-2)	3.1953e-1 (5.12e-3)	3.2521e-1 (5.21e-4)	3.3105e-1 (2.69e-4)
	10	40	3.3274e-1 (2.13e-4)	2.4884e-1 (7.57e-3)	3.1544e-1 (4.27e-3)	1.8184e-1 (2.41e-2)	3.3161e-1 (2.57e-3)	2.8060e-1 (1.07e-2)	3.2568e-1 (4.93e-3)	3.3335e-1 (1.34e-4)	3.3519e-1 (7.25e-5)
	5	20	3.7395e-1 (1.65e-3)	1.8482e-1 (8.19e-3)	1.9201e-1 (5.95e-3)	7.1962e-2 (7.78e-3)	2.7836e-1 (1.78e-3)	1.3906e-1 (8.30e-3)	2.7967e-1 (2.50e-3)	2.7930e-1 (8.10e-4)	2.8001e-1 (1.04e-3)
DCF6	5	40	3.8302e-1 (3.34e-4)	2.5630e-1 (3.77e-3)	2.6043e-1 (2.62e-3)	1.8917e-1 (1.07e-2)	2.8426e-1 (7.08e-4)	2.5856e-1 (5.00e-3)	2.8474e-1 (2.80e-3)	2.8430e-1 (1.90e-4)	2.8501e-1 (2.90e-4)
	10	20	2.7262e-1 (8.38e-4)	1.8701e-1 (6.37e-3)	2.4368e-1 (3.82e-3)	1.5229e-1 (1.07e-2)	2.7029e-1 (1.53e-3)	1.3449e-1 (7.31e-3)	2.7172e-1 (2.61e-3)	2.7497e-1 (3.18e-4)	2.7570e-1 (3.65e-4)
	10	40	2.7674e-1 (3.71e-4)	2.5654e-1 (2.77e-3)	2.6814e-1 (1.12e-3)	2.4130e-1 (5.27e-3)	2.7542e-1 (1.12e-3)	2.5660e-1 (2.80e-3)	2.7784e-1 (2.30e-3)	2.7676e-1 (1.77e-4)	2.7738e-1 (1.39e-4)
	5	20	5.7192e-1 (5.40e-4)	4.7530e-1 (1.31e-2)	5.2301e-1 (2.53e-2)	4.1333e-1 (1.81e-3)	5.7174e-1 (1.81e-3)	5.0248e-1 (1.67e-2)	5.7127e-1 (1.46e-3)	5.7354e-1 (1.31e-4)	5.7327e-1 (1.20e-4)
	5	40	5.7558e-1 (7.59e-5)	5.5366e-1 (5.05e-3)	5.6624e-1 (9.86e-3)	5.3286e-1 (1.25e-2)	5.7608e-1 (2.32e-4)	5.6619e-1 (2.63e-3)	5.7556e-1 (6.68e-4)	5.7611e-1 (6.17e-5)	5.7601e-1 (7.66e-5)
DCF7	10	20	5.7218e-1 (2.04e-4)	4.7347e-1 (1.78e-2)	5.5662e-1 (7.78e-3)	5.1906e-1 (8.10e-3)	5.6615e-1 (5.64e-3)	5.3610e-1 (1.52e-2)	5.6597e-1 (5.61e-3)	5.7375e-1 (1.17e-4)	5.7375e-1 (1.17e-4)
	10	40	5.7534e-1 (1.09e-4)	5.5507e-1 (4.73e-3)	5.7041e-1 (3.39e-3)	5.5421e-1 (5.49e-3)	5.7498e-1 (1.17e-3)	5.6703e-1 (3.99e-3)	5.7494e-1 (9.89e-4)	5.7541e-1 (7.47e-5)	5.7606e-1 (6.84e-5)
	5	20	5.3129e-1 (2.35e-2)	3.0345e-1 (1.83e-2)	3.6813e-1 (7.19e-2)	2.3759e-1 (1.20e-2)	5.7071e-1 (2.74e-3)	3.5754e-1 (1.39e-2)	5.6157e-1 (1.15e-2)	5.7479e-1 (9.66e-4)	5.7938e-1 (5.70e-4)
	5	40	5.4200e-1 (2.02e-2)	4.7910e-1 (1.93e-2)	4.5131e-1 (7.9e-3)	3.8481e-1 (1.43e-2)	5.8054e-1 (2.32e-3)	5.0477e-1 (1.05e-2)	5.7776e-1 (4.02e-3)	5.8489e-1 (1.93e-4)	5.8541e-1 (1.85e-4)
	10	20	5.2316e-1 (2.38e-2)	2.8899e-1 (1.57e-2)	4.8146e-1 (2.96e-2)	3.4722e-1 (1.68e-2)	5.5889e-1 (4.38e-3)	3.6337e-1 (1.77e-2)	5.5576e-1 (8.69e-3)	5.7191e-1 (4.21e-4)	5.7197e-1 (5.88e-4)
DCF8	10	40	5.3361e-1 (2.36e-2)	4.7431e-1 (1.17e-2)	5.4977e-1 (6.10e-2)	4.7867e-1 (1.83e-2)	5.6938e-1 (4.10e-3)	5.4346e-1 (1.21e-2)	5.6704e-1 (5.14e-3)	5.7628e-1 (2.12e-4)	5.7663e-1 (5.51e-4)
	5	20	5.3129e-1 (2.35e-2)	3.0345e-1 (1.83e-2)	3.6813e-1 (7.19e-2)	2.3759e-1 (1.20e-2)	5.7071e-1 (2.74e-3)	3.5754e-1 (1.39e-2)	5.6157e-1 (1.15e-2)	5.7479e-1 (9.66e-4)	5.7938e-1 (5.70e-4)
	5	40	5.4200e-1 (2.02e-2)	4.7910e-1 (1.93e-2)	4.5131e-1 (7.9e-3)	3.8481e-1 (1.43e-2)	5.8054e-1 (2.32e-3)	5.0477e-1 (1.05e-2)	5.7776e-1 (4.02e-3)	5.8489e-1 (1.93e-4)	5.8541e-1 (1.85e-4)
	10	20	5.2316e-1 (2.38e-2)	2.8899e-1 (1.57e-2)	4.8146e-1 (2.96e-2)	3.4722e-1 (1.68e-2)	5.5889e-1 (4.38e-3)	3.6337e-1 (1.77e-2)	5.5576e-1 (8.69e-3)	5.7191e-1 (4.21e-4)	5.7197e-1 (5.88e-4)
	10	40	5.3361e-1 (2.36e-2)	4.7431e-1 (1.17e-2)	5.4977e-1 (6.10e-2)	4.7867e-1 (1.83e-2)	5.6938e-1 (4.10e-3)	5.4346e-1 (1.21e-2)	5.6704e-1 (5.14e-3)	5.7628e-1 (2.12e-4)	5.7663e-1 (5.51e-4)

Table 2
Mean and standard deviation of MHV obtained by AMPF and eight compared algorithms on two DCMOP test suites under various environmental settings.

Problem	n_t	SKTEA	DCNSGAIL-A	DCNSGAIL-B	DCNSGA-III	MP-DFR	DC-MOEA	HATC	TDCEA	AMPF				
DCF10	5	20	2.7085e-1	(2.03e-2)	(1.3238e-2)	(4.05e-3)	(4.0782e-2)	(5.05e-3)	(1.4551e-2)	(4.08e-3)	(3.1889e-1)	(1.02e-3)	(3.1600e-1)	(2.79e-3)
	5	40	3.2843e-1	(1.01e-2)	(2.2027e-1)	(1.06e-2)	(1.8356e-1)	(1.23e-2)	(4.2566e-2)	(6.02e-3)	(3.4006e-1)	(1.01e-3)	(1.9535e-1)	(1.12e-3)
	5	40	2.9368e-1	(9.58e-3)	(1.2349e-2)	(4.09e-3)	(4.9231e-2)	(1.85e-2)	(2.5264e-2)	(5.98e-3)	(3.2910e-1)	(8.28e-4)	(2.8736e-2)	(1.54e-2)
	10	40	3.3035e-1	(4.01e-3)	(2.2663e-1)	(1.20e-2)	(2.4146e-2)	(1.63e-2)	(8.7847e-2)	(1.73e-2)	(3.4149e-1)	(1.583e-4)	(2.1472e-2)	(1.47e-2)
	5	20	2.0838e-1	(1.754e-3)	(1.4513e-1)	(8.42e-3)	(1.7381e-1)	(6.10e-3)	(9.0524e-2)	(7.73e-3)	(3.0017e-1)	(1.65e-3)	(1.4518e-1)	(6.44e-3)
	5	40	2.7280e-1	(3.90e-3)	(2.6321e-1)	(3.64e-3)	(2.8361e-1)	(3.32e-3)	(1.7337e-1)	(7.39e-3)	(3.0995e-1)	(6.12e-4)	(2.2742e-1)	(5.76e-3)
DCP1	5	20	2.5992e-1	(1.933e-3)	(1.3509e-1)	(1.04e-2)	(2.5677e-1)	(1.57e-3)	(1.4388e-1)	(1.00e-2)	(2.9002e-1)	(1.96e-3)	(1.2822e-1)	(6.85e-3)
	10	20	2.9410e-1	(2.15e-3)	(2.5451e-1)	(3.81e-3)	(2.8987e-1)	(1.67e-3)	(2.4064e-1)	(5.53e-3)	(2.9781e-1)	(9.63e-4)	(1.2822e-1)	(8.89e-5)
	5	20	2.4740e-1	(6.38e-3)	(1.6042e-1)	(1.07e-2)	(1.6135e-1)	(7.49e-3)	(9.1564e-2)	(1.23e-2)	(4.6923e-1)	(2.31e-3)	(1.6068e-1)	(1.06e-2)
	5	40	2.7280e-1	(3.90e-3)	(2.6321e-1)	(3.64e-3)	(2.8361e-1)	(3.32e-3)	(1.7337e-1)	(7.39e-3)	(3.0995e-1)	(6.12e-4)	(2.2742e-1)	(5.76e-3)
	5	40	2.5992e-1	(1.933e-3)	(1.3509e-1)	(1.04e-2)	(2.5677e-1)	(1.57e-3)	(1.4388e-1)	(1.00e-2)	(2.9002e-1)	(1.96e-3)	(1.2822e-1)	(6.85e-3)
	10	20	2.9410e-1	(2.15e-3)	(2.5451e-1)	(3.81e-3)	(2.8987e-1)	(1.67e-3)	(2.4064e-1)	(5.53e-3)	(2.9781e-1)	(9.63e-4)	(1.2822e-1)	(8.89e-5)
DCP2	5	20	2.4740e-1	(6.38e-3)	(1.6042e-1)	(1.07e-2)	(1.6135e-1)	(7.49e-3)	(9.1564e-2)	(1.23e-2)	(4.6923e-1)	(2.31e-3)	(1.6068e-1)	(1.06e-2)
	5	40	2.4687e-1	(5.35e-3)	(3.2159e-1)	(1.67e-2)	(2.4502e-1)	(1.09e-2)	(1.7081e-1)	(1.31e-2)	(4.8836e-1)	(5.08e-3)	(2.8496e-1)	(2.06e-2)
	10	20	4.1809e-1	(1.256e-3)	(1.7147e-1)	(1.01e-2)	(2.1977e-1)	(1.31e-2)	(9.9923e-2)	(2.96e-2)	(4.8148e-1)	(1.271e-3)	(2.0527e-1)	(1.288e-2)
	10	40	4.2190e-1	(8.55e-3)	(3.2445e-1)	(1.34e-2)	(3.1241e-1)	(1.73e-2)	(2.0822e-1)	(2.27e-2)	(4.8983e-1)	(4.83e-3)	(3.0704e-1)	(2.28e-2)
	5	20	4.4190e-3	(3.54e-3)	(4.2911e-6)	(1.90e-5)	(1.7376e-7)	(1.91e-7)	(0.0000e+0)	(0.00e+0)	(7.6588e-2)	(7.83e-3)	(1.1274e-2)	(1.37e-2)
	5	40	4.0361e-2	(1.05e-2)	(4.2554e-4)	(9.02e-4)	(3.4777e-5)	(8.79e-5)	(2.5669e-7)	(1.92e-7)	(2.0833e-1)	(1.09e-2)	(3.4916e-2)	(1.92e-2)
DCP3	5	20	1.9071e-1	(3.284e-3)	(1.9765e-8)	(7.72e-8)	(1.2672e-4)	(4.566e-4)	(2.6331e-6)	(1.16e-5)	(1.0521e-1)	(1.720e-3)	(8.4747e-4)	(1.94e-3)
	10	40	3.1402e-2	(2.114e-2)	(2.5974e-4)	(6.47e-4)	(1.9094e-3)	(3.80e-3)	(3.6255e-4)	(1.30e-3)	(2.0311e-1)	(9.26e-3)	(1.1880e-2)	(7.99e-3)
	5	20	4.4190e-3	(3.54e-3)	(4.2911e-6)	(1.90e-5)	(1.7376e-7)	(1.91e-7)	(0.0000e+0)	(0.00e+0)	(7.6588e-2)	(7.83e-3)	(1.1274e-2)	(1.37e-2)
	5	40	4.0361e-2	(1.05e-2)	(4.2554e-4)	(9.02e-4)	(3.4777e-5)	(8.79e-5)	(2.5669e-7)	(1.92e-7)	(2.0833e-1)	(1.09e-2)	(3.4916e-2)	(1.92e-2)
	5	40	1.9071e-1	(3.284e-3)	(1.9765e-8)	(7.72e-8)	(1.2672e-4)	(4.566e-4)	(2.6331e-6)	(1.16e-5)	(1.0521e-1)	(1.720e-3)	(8.4747e-4)	(1.94e-3)
	10	40	3.1402e-2	(2.114e-2)	(2.5974e-4)	(6.47e-4)	(1.9094e-3)	(3.80e-3)	(3.6255e-4)	(1.30e-3)	(2.0311e-1)	(9.26e-3)	(1.1880e-2)	(7.99e-3)
DCP4	5	20	3.0925e-1	(1.19e-4)	(8.4013e-2)	(9.05e-3)	(2.9830e-1)	(3.07e-3)	(2.6092e-1)	(1.18e-2)	(3.0369e-1)	(1.86e-3)	(2.9914e-1)	(3.41e-3)
	5	40	3.1040e-1	(1.388e-5)	(2.5067e-1)	(5.60e-3)	(3.0854e-1)	(1.70e-3)	(2.8921e-1)	(6.50e-3)	(3.0934e-1)	(1.10e-3)	(3.0725e-1)	(2.26e-3)
	10	20	3.0425e-1	(1.67e-4)	(8.4484e-2)	(9.76e-3)	(2.8658e-1)	(2.86e-3)	(2.3794e-1)	(4.48e-3)	(3.0055e-1)	(2.14e-3)	(2.925e-1)	(3.95e-3)
	10	40	3.0548e-1	(4.20e-5)	(2.4673e-1)	(4.43e-3)	(3.0102e-1)	(2.09e-3)	(2.6663e-1)	(4.64e-3)	(3.0586e-1)	(4.69e-4)	(3.0175e-1)	(1.63e-3)
	5	20	3.130e-1	(6.54e-3)	(3.8306e-2)	(5.60e-3)	(2.7183e-1)	(1.54e-3)	(2.3168e-1)	(1.31e-2)	(2.9939e-1)	(2.96e-2)	(2.6694e-1)	(9.43e-3)
	5	40	2.7605e-1	(5.70e-3)	(4.4150e-1)	(5.74e-3)	(2.5183e-1)	(5.22e-3)	(2.5404e-1)	(4.32e-2)	(3.1504e-1)	(3.19e-2)	(2.7776e-1)	(9.89e-3)
DCP5	5	20	3.2428e-1	(9.06e-3)	(3.9320e-2)	(6.25e-3)	(2.5513e-1)	(5.60e-3)	(2.2737e-1)	(6.86e-3)	(3.0887e-1)	(2.85e-2)	(2.6053e-1)	(8.98e-3)
	10	40	2.8384e-1	(5.62e-3)	(1.4299e-1)	(5.29e-3)	(2.6344e-1)	(1.49e-2)	(2.4484e-1)	(9.24e-3)	(3.2467e-1)	(3.39e-2)	(2.7637e-1)	(1.07e-2)
	5	20	3.130e-1	(6.54e-3)	(3.8306e-2)	(5.60e-3)	(2.7183e-1)	(1.54e-3)	(2.3168e-1)	(1.31e-2)	(2.9939e-1)	(2.96e-2)	(2.6694e-1)	(9.43e-3)
	5	40	2.7605e-1	(5.70e-3)	(4.4150e-1)	(5.74e-3)	(2.5183e-1)	(5.22e-3)	(2.5404e-1)	(4.32e-2)	(3.1504e-1)	(3.19e-2)	(2.7776e-1)	(9.89e-3)
	5	20	3.2428e-1	(9.06e-3)	(3.9320e-2)	(6.25e-3)	(2.5513e-1)	(5.60e-3)	(2.2737e-1)	(6.86e-3)	(3.0887e-1)	(2.85e-2)	(2.6053e-1)	(8.98e-3)
	10	40	2.8384e-1	(5.62e-3)	(1.4299e-1)	(5.29e-3)	(2.6344e-1)	(1.49e-2)	(2.4484e-1)	(9.24e-3)	(3.2467e-1)	(3.39e-2)	(2.7637e-1)	(1.07e-2)
DCP6	5	20	4.0750e-1	(9.70e-4)	(4.2478e-1)	(1.25e-2)	(3.6785e-1)	(1.37e-2)	(3.0994e-1)	(1.16e-2)	(4.0240e-1)	(4.00e-3)	(3.7742e-1)	(4.50e-3)
	5	40	1.769e-1	(3.45e-4)	(3.7665e-1)	(5.40e-3)	(3.9154e-1)	(1.97e-3)	(3.3264e-1)	(1.60e-2)	(4.1845e-1)	(1.64e-3)	(3.9698e-1)	(4.32e-3)
	10	20	4.0523e-1	(8.99e-4)	(2.4394e-1)	(1.20e-2)	(3.6548e-1)	(8.57e-3)	(3.1018e-1)	(1.96e-2)	(3.9679e-1)	(4.72e-3)	(3.6404e-1)	(6.61e-3)
	10	40	1.387e-1	(3.24e-4)	(3.7199e-1)	(5.20e-3)	(3.7606e-1)	(6.94e-3)	(3.2627e-1)	(1.53e-2)	(4.1334e-1)	(2.79e-3)	(3.8931e-1)	(3.06e-3)
	5	20	5.3170e-1	(8.79e-3)	(3.2390e-1)	(1.97e-2)	(3.2110e-1)	(9.94e-3)	(2.1952e-1)	(1.62e-2)	(5.5498e-1)	(9.98e-4)	(2.6278e-1)	(1.48e-2)
	5	40	5.5840e-1	(8.65e-4)	(4.3456e-1)	(1.15e-2)	(4.3705e-1)	(5.19e-3)	(3.4789e-1)	(1.08e-2)	(5.6309e-1)	(2.62e-4)	(3.7751e-1)	(7.35e-3)
DCP7	5	20	3.9697e-1	(1.08e-2)	(1.9783e-1)	(1.04e-2)	(2.4777e-1)	(1.32e-2)	(1.0336e-1)	(1.32e-2)	(4.1859e-1)	(6.21e-3)	(2.0351e-1)	(3.21e-2)
	5	40	4.1203e-1	(3.05e-2)	(3.6613e-1)	(7.21e-2)	(3.4510e-1)	(1.48e-2)	(2.3637e-1)	(1.97e-2)	(4.2739e-1)	(1.63e-3)	(3.5361e-1)	(2.83e-2)
	10	20	5.3877e-1	(1.23e-2)	(1.3106e-1)	(1.60e-2)	(4.1088e-1)	(9.62e-3)	(3.1021e-1)	(1.11e-2)	(5.5869e-1)	(1.01e-3)	(2.5165e-1)	(1.64e-2)
	10	40	5.6292e-1	(2.85e-4)	(4.1964e-1)	(1.47e-2)	(5.0458e-1)	(6.60e-3)	(4.4090e-1)	(6.07e-3)	(5.6421e-1)	(2.06e-4)	(3.8584e-1)	(1.83e-2)
	5	20	3.9697e-1	(1.08e-2)	(1.9783e-1)	(1.04e-2)	(2.4777e-1)	(1.32e-2)	(1.0336e-1)	(1.32e-2)	(4.1859e-1)	(6.21e-3)	(2.0351e-1)	(3.21e-2)
	5	40	4.1203e-1	(3.05e-2)	(3.6613e-1)	(7.21e-2)	(3.4510e-1)	(1.48e-2)	(2.3637e-1)	(1.97e-2)	(4.2739e-1)	(1.63e-3)	(3.5361e-1)	(2.83e-2)
DCP8	5	20	4.1202e-1	(1.23e-2)	(1.8133e-1)	(1.49e-2)	(3.1345e-1)	(1.40e-2)	(2.6351e-1)	(1.90e-2)	(4.1202e-1)	(4.93e-3)	(2.3792e-1)	(2.49e-2)
	10	40	1.724e-1	(7.20e-3)	(3.5192e-1)	(9.83e-3)	(3.5298e-1)	(1.51e-2)	(3.0403e-1)	(1.90e-2)	(4.1763e-1)	(6.56e-3)	(3.7629e-1)	(1.25e-2)
	5	20	3.4179e-1	(9.85e-3)	(3.3169e-1)	(8.34e-3)	(1.7558e-1)	(1.12e-2)	(3.8124e-2)	(5.53e-3)	(4.1060e-1)	(7.09e-3)	(3.1373e-1)	(1.25e-2)
	5	40	2.4742e-1	(9.94e-3)	(4.2444e-1)	(3.19e-3)	(3.7062e-1)	(7.62e-3)	(1.5491e-1)	(4.76e-3)	(4.4314e-1)	(4.19e-3)	(4.1314e-1)	(4.19e-3)
	10	20	3.7949e-1	(1.03e-2)	(3.4303e-1)	(6.35e-3)	(2.8207e-1)	(8.69e-3)	(6.6370e-2)	(1.16e-2)	(4.089e-1)	(7.45e-3)	(3.0926e-1)	(1.42e-2)
	10	40	3.829e-1	(6.72e-3)	(4.2012e-1)	(2.44e-3)	(4.1158e-1)	(1.51e-2)	(1.8032e-1)	(1.06e-2)	(4.4557e-1)	(1.28e-2)	(4.0946e-1)	(1.39e-2)
+/-/-/ =											0/76/0	0/76/0	0/76/0	3/69/4

Algorithm 3 Dynamic tri-population optimization framework.

Input: Population, Offspring, N (population size), N_v (size of convergence auxiliary population)

Output: Population, Offspring

```

1:  $N_s \leftarrow \lfloor N/3 \rfloor$ ;
2: Population  $\leftarrow$  Population  $\cup$  Offspring;
3: Update  $Z_{\min}$ ;
4: MainPop  $\leftarrow$  Update Population by main task;
5: ConPop  $\leftarrow$  Update Population by unconstrained
   auxiliary task;
6: DivPop  $\leftarrow$  Update Population by diversity
   enhancement task;
7: Population  $\leftarrow$  MainPop  $\cup$  ConPop  $\cup$  DivPop;
8: Offspring1  $\leftarrow$  Generate  $N_s$  offspring individuals by
   MainPop;
9: Offspring2  $\leftarrow$  Generate  $N_v$  offspring individuals by
   ConPop;
10: Offspring3  $\leftarrow$  Generate  $2N_s - N_v$  offspring individuals by
   DivPop;
11: Offspring  $\leftarrow$  Offspring1  $\cup$  Offspring2  $\cup$  Offspring3;
```

these offspring sets are merged to form the complete offspring population for the next generation (lines 8–11).

The allocation of offspring among the two auxiliary populations is dynamically controlled by the detected change intensity. The convergence auxiliary population generates N_v offspring individuals, while the diversity auxiliary population generates $2N_s - N_v$ offspring individuals. When the environmental change is severe, a larger N_v is assigned to the convergence auxiliary population, allowing more computational resources to be used for quickly approaching the new UPF and adapting to the shifted CPF. This is beneficial because drastic changes may cause the current population to deviate substantially from the new promising region. In contrast, when the environmental change is mild, the current population is usually close to the new CPF, and excessive convergence-oriented search may waste evaluations. In this case, the framework reduces the size of the convergence auxiliary population and assigns more resources to the diversity auxiliary population, which supports finer exploration of the CPF and improves the uniformity of the obtained approximation. Therefore, the proposed dynamic allocation strategy enables AMPF to adaptively balance convergence and diversity under different environmental change patterns.

4.5. Computational complexity

AMPF comprises three key parts, including the change intensity detection module, static optimizer, and adaptive memory-prediction fusion mechanism. The change intensity detection module evaluates the individuals and calculates the intensity indicators, with a computational complexity of $O(MN)$. Static optimizer has a computational complexity of $O(MN^2) + O(MN \log(N))$. In the adaptive memory-prediction fusion mechanism, the primary computational overhead arises from distance calculations in the objective space, resulting in a complexity of $O(MN^2)$. Consequently, the overall computational complexity of AMPF is $O(MN^2)$, where N represents the population size and M denotes the number of objectives.

5. Experiments

5.1. Experimental settings

To comprehensively evaluate AMPF, experiments are conducted from two perspectives. Firstly, AMPF is compared with eight representative DCMOEAs on widely used DCMOP benchmark suites. Secondly, ablation studies are performed to analyze the effectiveness of its key

components. All algorithms are implemented in MATLAB R2024b and executed using parallel computing on a workstation equipped with a 13th Gen Intel® Core™ i9-13900HX CPU at 2.20 GHz and 64 GB RAM.

5.1.1. Benchmark test suites

The DCF (Guo et al., 2022) and DCP (Chen et al., 2024b) test suites are adopted to assess the performance of the compared algorithms. The DCF suite contains 10 test problems, while the DCP suite contains 9 test problems. All 19 problems are dynamic constrained bi-objective optimization problems. According to the source of environmental variation, these problems can be divided into three categories. The first category involves dynamic changes only in the objective functions, including DCF2, DCF5, DCF7, DCP1, and DCP2. The second category involves dynamic changes only in the constraints, including DCF1, DCF4, DCP4, DCP5, and DCP6. The third category involves simultaneous changes in both the objective functions and the constraints, covering the remaining problems. This classification enables a more detailed evaluation of algorithmic performance under different types of environmental dynamics.

5.1.2. Compared algorithms

Eight state-of-the-art DCMOEAs are selected to evaluate the performance of the proposed algorithm, including DC-MOEA (Azzouz et al., 2018), DCNSGAI-A (Deb et al., 2007), DCNSGAI-B (Deb et al., 2007), DCNSGA-III (Chen et al., 2024c), HATC (Zhang et al., 2025a), TDCEA (Chen et al., 2024b), SKTEA (Chen et al., 2024a), and MP-DFR (Zhang et al., 2025b). Among these, DC-MOEA employs a dynamic optimizer to enhance optimization performance. DCNSGAI-A randomly reinitializes the individuals, whereas DCNSGAI-B perturbs 20% of the individuals to enhance diversity. DCNSGA-III employs a multi-population strategy to simultaneously balance feasibility and convergence. HATC is a history-assisted two-state auxiliary task collaboration approach for DCMOPs, which uses historical environmental information and auxiliary-task collaboration to improve the evolutionary process. TDCEA adopts a two-stage diversity compensation strategy with collaborative search characteristics. SKTEA divides the objective space into feasible and infeasible subspaces, enhancing adaptability by transferring knowledge from solutions within feasible subspaces to infeasible subspaces. MP-DFR incorporates point set pairing techniques from computer vision into DCMOEA, enabling more granular capture of environmental changes. To ensure fairness, all parameters for the compared algorithms are configured according to the original papers.

5.1.3. Parameter settings

To ensure a fair comparison, the same population size is used for AMPF and all compared algorithms. Following the common practice in evolutionary multi-objective optimization and dynamic multi-objective optimization studies (Tian et al., 2020; Zhang et al., 2025b), N is set to 100 in all experiments. The dimension of the decision space is set to $D = 10$, consistent with the default parameters in the original articles (Chen et al., 2024b; Guo et al., 2022). Each test problem is independently run 20 times. The parameters n_t and τ_t are used to control the severity and frequency of environmental changes, respectively. A larger n_t corresponds to milder environmental changes, while a larger τ_t indicates a lower change frequency. To evaluate algorithmic performance under different dynamic scenarios, four parameter combinations are considered: $(n_t, \tau_t) \in \{(5, 20), (5, 40), (10, 20), (10, 40)\}$. These four combinations are used to cover four different dynamic scenarios. Specifically, (5, 20), (5, 40), (10, 20), and (10, 40) correspond to the severe and frequent changes, the severe but less frequent changes, the mild but frequent changes, and the mild and less frequent changes, respectively. Therefore, the same test problem under different (n_t, τ_t) settings may present different levels of difficulty and emphasize different algorithmic abilities, such as rapid response, convergence recovery, diversity maintenance, and robustness to outdated historical information. Before the first environmental change, each algorithm is executed for 60 generations to obtain sufficient initial search information. Subsequently, 30

environmental changes are introduced. Therefore, the total number of generations is set to $60 + 30 \cdot \tau_i$.

5.1.4. Performance indicators

Two widely used performance indicators for DCMOPs are adopted, namely the mean inverse generational distance (MIGD) (Jiang et al., 2021) and the mean hypervolume (MHV) (Zitzler & Thiele, 1999). MIGD measures the average distance between each point on the reference CPF and its nearest corresponding point in the solution set. A smaller MIGD indicates that the solution set obtained by the algorithm is closer to the CPF. MIGD can be calculated as follows:

$$\text{MIGD}(CPF_t^*, CPF_t) = \frac{\sum_{i \in T} \text{IGD}(CPF_t^*, CPF_t)}{|T|}, \quad (16)$$

with

$$\text{IGD}(CPF_t^*, CPF_t) = \frac{\sum_{p^* \in CPF_t^*} \min_{p \in CPF_t} \|p^* - p\|}{|CPF_t^*|}, \quad (17)$$

where T represents a set of discrete time steps and CPF_t^* represents the true CPF at time t . CPF_t denotes the approximated solution set obtained by the algorithm at time t . MHV evaluates the performance of an algorithm by measuring the hypervolume enclosed by the approximated set and the reference point z^r . A higher MHV value indicates better performance of the algorithm. The MHV is calculated as follows:

$$\text{MHV}(CPF_t) = \frac{\sum_{i \in T} \text{HV}(CPF_t)}{|T|}, \quad (18)$$

with

$$\text{HV}(CPF_t) = L(\cup_{x \in CPF_t} [x_1, z_1^r] \times \cdots \times [x_M, z_M^r]). \quad (19)$$

5.2. Comparative results

Table 1 reports the mean and standard deviation of MIGD obtained by AMPF and the eight compared algorithms on all test instances. The results of the Wilcoxon rank-sum test are provided, where “+”, “−”, and “=” indicate that the corresponding compared algorithm performs significantly better than, significantly worse than, and statistically equivalent to AMPF, respectively. AMPF obtains the best MIGD values on 63 out of 76 test instances, demonstrating its strong capability in tracking DCPFs. Compared with SKTEA, DCNSGAI-A, DCNSGAI-B, DCNSGA-III, MP-DFR, DC-MOEA, HATC, and TDCEA, AMPF achieves significantly better MIGD results on 76, 76, 76, 76, 71, 76, 75, and 60 instances, respectively. These results indicate that AMPF is highly competitive in terms of convergence and distribution under dynamic constrained environments. The favorable MIGD performance of AMPF can be mainly attributed to two aspects. Firstly, the adaptive memory-prediction fusion mechanism exploits both recent change trends and information from perturbed historical elite solutions, which helps construct a more suitable initial population after environmental changes. Secondly, the dynamic tri-population optimization framework adaptively coordinates feasibility exploitation, convergence enhancement, and diversity maintenance, thereby improving the search efficiency on complex DCMOPs. However, AMPF does not obtain the best MIGD value in every dynamic configuration. For example, MP-DFR achieves the best result on one configuration of DCF5, while TDCEA obtains better MIGD values on several configurations of DCF7, DCF8, DCP4, DCP5, and DCP8. These exceptions indicate that some problem-specific dynamic characteristics can reduce the advantage of AMPF. In particular, AMPF may produce a relatively conservative response intensity for problems where the objective functions change little and the shapes of DCPFs are primarily determined by constraint pruning. Thus, its convergence performance may be inferior to TDCEA on certain instances. Similarly, MP-DFR can be competitive on some configurations where maintaining diverse feasible search directions is especially important. Therefore, the MIGD results suggest that AMPF has strong overall tracking ability, but its advantage is not uniform across all dynamic configurations.

Table 2 presents the comparative results in terms of MHV. This indicator evaluates the overall quality of the obtained solution set from the perspective of the dominated region in the objective space, and a larger MHV value usually indicates better convergence and diversity. AMPF obtains the best MHV values on 57 out of 76 test instances. Although this confirms that AMPF still has a clear overall advantage under the MHV indicator, the advantage is less dominant than that observed for MIGD. According to the Wilcoxon rank-sum test, AMPF significantly outperforms SKTEA, DCNSGAI-A, DCNSGAI-B, DCNSGA-III, MP-DFR, DC-MOEA, HATC, and TDCEA on 76, 76, 76, 76, 64, 76, 63, and 69 instances, respectively. The MHV results also reveal several problem-specific cases where other algorithms perform better. For example, HATC obtains better MHV values on several configurations of DCF1 and DCP1. One possible reason is that HATC uses a history-assisted auxiliary task and similar historical CPF information, which can help preserve a broader dominated region on these instances. MP-DFR achieves better MHV values on some configurations of DCF5, DCF6, DCF8, and DCF10, indicating that its diversity-related mechanism can be effective when the dominated-volume coverage is sensitive to the spread of solutions. TDCEA also performs well on several configurations of DCF8 and DCP4, which suggests that its two-stage diversity compensation strategy can be beneficial for maintaining solution spread on specific dynamic landscapes.

Fig. 4 presents the convergence curves of IGD values under the dynamic settings $n_i = 5$ and $\tau_i = 20$, where each curve corresponds to one compared algorithm. As can be seen, a lower curve indicates better performance under the IGD indicator, while a smoother curve signifies greater stability during the dynamic tracking process. In most cases, AMPF produces curves that are lower and smoother than those of the compared algorithms, suggesting that it can guide the population to approach the DCPFs rapidly and stably after environmental changes. However, some compared algorithms may show competitive curves on specific problems, especially when their diversity compensation or auxiliary-task mechanisms match the characteristics of the dynamic landscape.

To further illustrate the ability of the compared algorithms to handle changes in DCPF shape and distribution, Fig. 5 presents the approximated DCPF distributions obtained on DCF4 under the setting $\tau_i = 20$ and $n_i = 5$. It can be observed that AMPF produces a solution set with better proximity and more uniform distribution along the target front, whereas several compared algorithms show more obvious gaps, local aggregation, or insufficient coverage in some regions. This result suggests that AMPF can better adapt to complex DCPF changes even when the DCPF shape is not explicitly known in advance. The main reason is that the adaptive memory-prediction fusion mechanism does not require an explicit fixed DCPF model, but instead exploits reliable recent trends and similar historical information to guide population initialization. Meanwhile, the dynamic tri-population framework further helps maintain convergence, diversity, and feasibility during the subsequent evolutionary process.

5.3. Parameter analysis

The proposed algorithm includes two key parameters: the constraint violation weight w_{CV} in the change intensity detection module and the MAD threshold parameter k_{MAD} in the adaptive memory-prediction fusion mechanism. The former controls whether constraint violation variation should be incorporated into the estimation of environmental change intensity, while the latter controls the strictness of the filtering criterion for predicted individuals.

The parameter w_{CV} is used to control the contribution of constraint variation in the change intensity detection module. Specifically, a larger w_{CV} assigns more importance to the change in constraint violation, while $w_{CV} = 0$ means that the change intensity is estimated only according to the variation in the objective space. To investigate the influence of this parameter, five variants are designed:

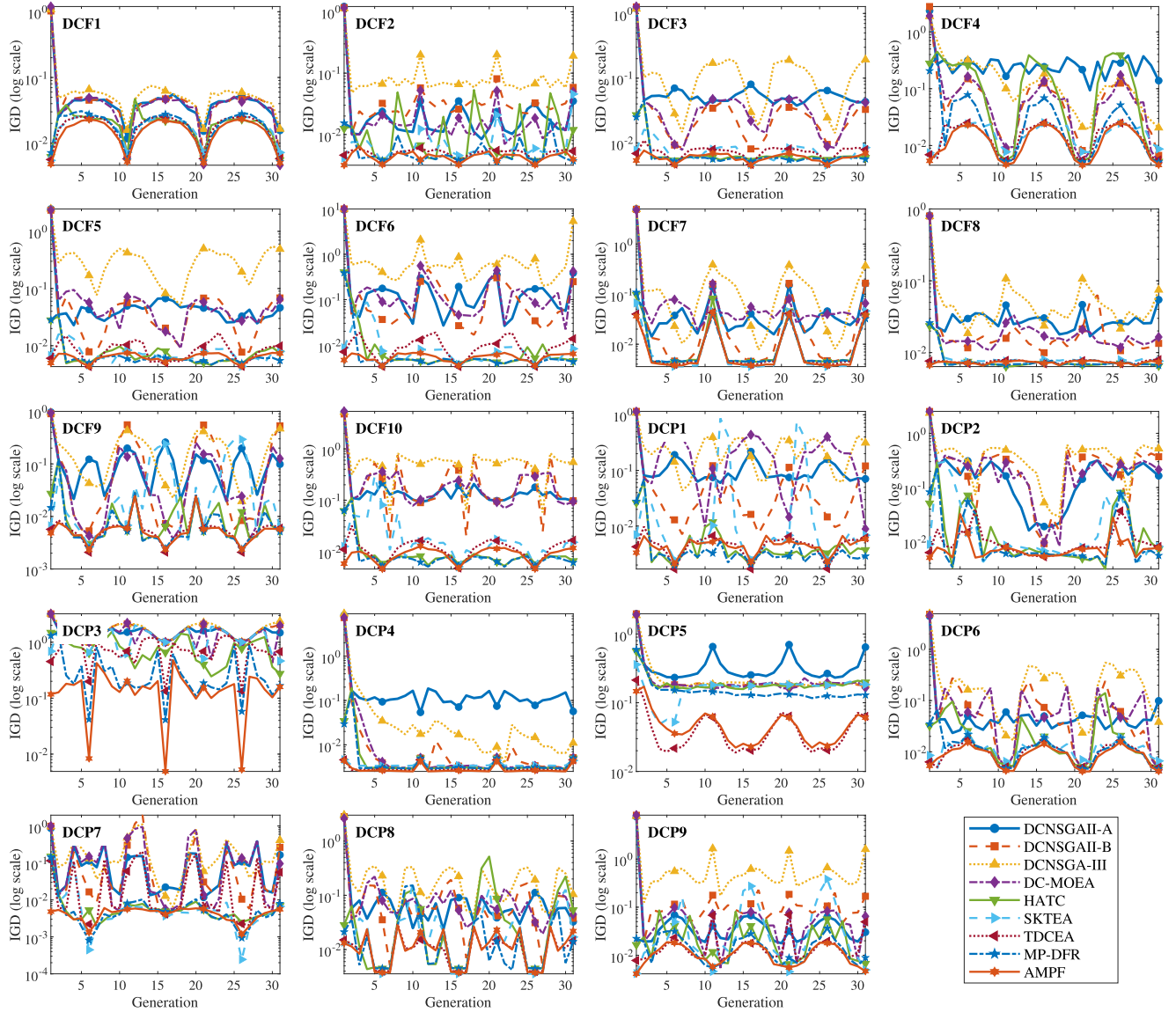


Fig. 4. Curves of average IGD values over environmental changes obtained by AMPF and the compared algorithms on DCF and DCP test suites with $\tau_i = 20$ and $n_i = 5$.

$w_{CV} \in \{1.00, 0.75, 0.50, 0.25, 0.00\}$. The results in Tables 3 and 4 show that $w_{CV} = 0.00$ achieves the best overall performance on both IGD and HV. For IGD, $w_{CV} = 0.00$ achieves the best IGD values on 13 out of 19 test instances. This indicates that the overall performance tends to deteriorate when more constraint-violation information is incorporated into the change-intensity estimation. A similar trend is observed in the HV results. The setting $w_{CV} = 0.00$ achieves the best results on 12 test instances. These results further demonstrate that directly introducing constraint variation into the change intensity detection module does not improve the overall performance.

The main reason is that objective space variation and constraint variation have different numerical scales and search implications. The detected change intensity in AMPF is mainly used to guide the dynamic response strength and population reconstruction after environmental changes. If constraint variation is directly mixed into the estimation of change intensity, the response strength may be dominated by unstable or scale-sensitive constraint changes, which can lead to an inappropriate dynamic response. In contrast, using only objective space variation provides a more stable estimate of the movement degree of the environment. The influence of constraints can then be handled by the sub-

sequent constrained optimization process, including the main population and auxiliary populations. Therefore, $w_{CV} = 0.00$ is adopted in the final version of AMPF. This setting means that the environmental change intensity is calculated only from the objective-space variation, while the constraint-related influence is addressed during the evolutionary search stage rather than in the change intensity estimation stage.

The parameter k_{MAD} controls the strictness of the validity criterion for trusted matches in the adaptive memory-prediction fusion. A smaller k_{MAD} imposes a stricter filtering threshold, so that only matched pairs close to the robust central tendency are retained. In contrast, a larger k_{MAD} relaxes the threshold and allows more predicted individuals to participate in the subsequent search. Therefore, this parameter affects the balance between prediction utilization and reliability control. To conduct a more comprehensive parameter analysis, seven settings are considered: $k_{MAD} \in \{0.5, 1, 1.5, 2, 2.5, 3, 3.5\}$. The results of the Friedman test in Figs. 6 and 7 show that $k_{MAD} = 1$ achieves the best overall performance. For IGD, $k_{MAD} = 1$ obtains the best average ranking of 2.789. For HV, $k_{MAD} = 1$ also achieves the best average ranking of 2.789. The setting $k_{MAD} = 0.5$ is competitive, with an average ranking of 3.263, but it is still inferior to $k_{MAD} = 1$ in terms of overall ranking and robust-

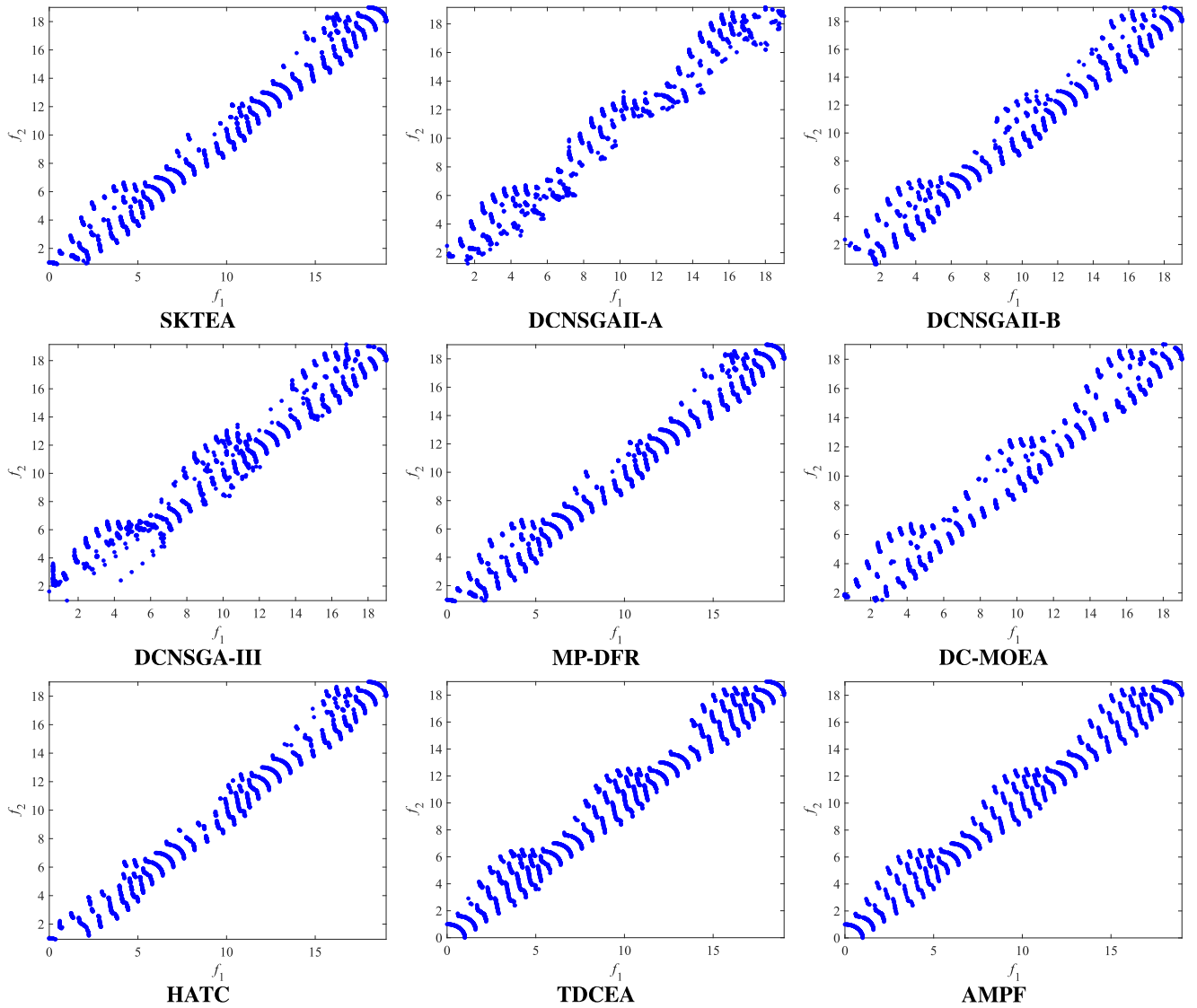


Fig. 5. Approximated DCPF solution distributions obtained by SKTEA, DCNSGAI-A, DCNSGAI-B, DCNSGA-III, MP-DFR, DC-MOEA, HATC, TDCEA, and AMPF on DCF4 with $\tau_i = 20$ and $n_i = 5$.

ness. When k_{MAD} increases beyond 1, the overall performance generally deteriorates. In particular, large settings such as $k_{MAD} = 2.5$, $k_{MAD} = 3$, and $k_{MAD} = 3.5$ obtain worse average rankings on both IGD and HV, indicating that a loose filtering threshold may retain too many unreliable predicted individuals. These results suggest that both overly strict and overly loose k_{MAD} values are not optimal. When k_{MAD} is too small, useful predicted individuals may be removed. When k_{MAD} is too large, inaccurate predicted individuals may be retained and mislead the subsequent evolutionary search. Moreover, the trusted matching rate is further used to adjust the fusion between trend-based prediction and historical elite reuse. Therefore, matching quality affects not only the reliability of the predicted initial population but also the subsequent optimization performance. By contrast, $k_{MAD} = 1$ provides a better trade-off between retaining useful predicted solutions and suppressing unreliable ones.

Overall, the parameter sensitivity analysis shows that AMPF performs best when $w_{CV} = 0.00$ and $k_{MAD} = 1$. Therefore, these two values are adopted as the default settings in all experiments. The former suggests that objective-space variation is more suitable for estimating environmental change intensity, while the latter indicates that a moderately strict MED-MAD threshold is more effective for controlling the reliability of predicted solutions.

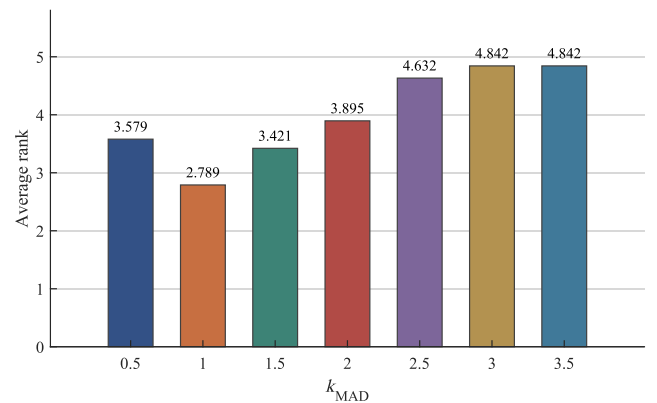


Fig. 6. Average Friedman rankings in terms of IGD for the sensitivity analysis of the MAD threshold parameter.

Table 3

IGD results of the sensitivity experiment on the constraint violation weight in environmental change intensity detection.

Problem	$w_{CV} = 1.00$	$w_{CV} = 0.75$	$w_{CV} = 0.50$	$w_{CV} = 0.25$	$w_{CV} = 0.00$
DCF1	3.1259e-02 (8.96e-04) -	3.0977e-02 (9.33e-04) -	3.0143e-02 (6.95e-04) -	2.9511e-02 (8.04e-04) -	2.3455e-02 (6.56e-04)
DCF2	1.2586e-02 (1.65e-03) -	1.2619e-02 (1.38e-03) -	1.3051e-02 (1.52e-03) -	1.2002e-02 (1.37e-03) -	1.1001e-02 (8.83e-04)
DCF3	1.1812e-02 (5.77e-04) =	1.1681e-02 (7.26e-04) =	1.1805e-02 (6.87e-04) =	1.1780e-02 (5.96e-04) =	1.1490e-02 (5.24e-04)
DCF4	2.5821e-02 (6.24e-04) -	2.5701e-02 (1.03e-03) -	2.5273e-02 (9.07e-04) -	2.5246e-02 (1.08e-03) -	2.1234e-02 (5.51e-04)
DCF5	2.1851e-02 (1.08e-03) -	2.2832e-02 (1.29e-03) -	2.2418e-02 (1.24e-03) -	2.2037e-02 (1.68e-03) -	1.7302e-02 (8.30e-04)
DCF6	2.1664e-02 (1.03e-03) -	2.1603e-02 (1.14e-03) -	2.0851e-02 (1.17e-03) -	2.0523e-02 (1.36e-03) -	1.2330e-02 (5.29e-04)
DCF7	1.3553e-02 (6.98e-04) =	1.3692e-02 (9.57e-04) -	1.3253e-02 (7.31e-04) =	1.3129e-02 (6.87e-04) =	1.3097e-02 (6.96e-04)
DCF8	1.0707e-02 (1.25e-04) -	1.0708e-02 (1.87e-04) -	1.0744e-02 (1.95e-04) -	1.0709e-02 (1.71e-04) -	9.9151e-03 (1.52e-04)
DCF9	1.3031e-02 (7.78e-04) -	1.3022e-02 (7.49e-04) -	1.3217e-02 (9.40e-04) -	1.3046e-02 (1.24e-03) -	1.1231e-02 (6.15e-04)
DCF10	2.9671e-02 (2.42e-03) =	2.9349e-02 (1.82e-03) =	2.9942e-02 (1.81e-03) =	2.9552e-02 (1.92e-03) =	3.0122e-02 (1.70e-03)
DCP1	1.2434e-02 (1.48e-03) =	1.1777e-02 (1.18e-03) =	1.1841e-02 (9.97e-04) =	1.2101e-02 (7.52e-04) =	1.2048e-02 (6.67e-04)
DCP2	2.1080e-02 (1.97e-03) =	2.1165e-02 (1.39e-03) =	2.0961e-02 (1.43e-03) =	2.0523e-02 (1.45e-03) =	2.0818e-02 (2.24e-03)
DCP3	5.4820e-01 (5.27e-02) +	5.3851e-01 (4.29e-02) +	5.5434e-01 (5.15e-02) +	5.6258e-01 (4.30e-02) +	6.7241e-01 (4.81e-02)
DCP4	3.1407e-03 (1.31e-04) =	3.1346e-03 (1.50e-04) =	3.0963e-03 (1.59e-04) =	3.0905e-03 (1.02e-04) =	3.0739e-03 (1.24e-04)
DCP5	9.8548e-02 (4.49e-02) =	9.7754e-02 (3.54e-02) -	8.8318e-02 (3.64e-02) =	8.4307e-02 (3.31e-02) =	7.4534e-02 (6.95e-03)
DCP6	1.7124e-02 (6.56e-04) -	1.7078e-02 (6.71e-04) -	1.6571e-02 (4.94e-04) -	1.6499e-02 (4.72e-04) -	1.3792e-02 (4.33e-04)
DCP7	1.0219e-02 (4.59e-04) =	1.0162e-02 (5.00e-04) =	1.0028e-02 (4.18e-04) =	1.0107e-02 (4.30e-04) =	1.0170e-02 (5.88e-04)
DCP8	3.2126e-02 (3.55e-03) =	3.1516e-02 (3.77e-03) =	3.0860e-02 (2.98e-03) =	3.1517e-02 (3.52e-03) =	3.1938e-02 (3.47e-03)
DCP9	4.5719e-02 (8.71e-03) -	4.4217e-02 (1.16e-02) =	4.4037e-02 (9.45e-03) =	4.3760e-02 (1.09e-02) =	3.8936e-02 (7.20e-03)
+/-/=	1/9/9	1/10/8	1/8/10	1/8/10	

Table 4

HV results of the sensitivity experiment on the constraint violation weight in environmental change intensity detection.

Problem	$w_{CV} = 1.00$	$w_{CV} = 0.75$	$w_{CV} = 0.50$	$w_{CV} = 0.25$	$w_{CV} = 0.00$
DCF1	3.5282e-01 (4.97e-04) -	3.5292e-01 (5.60e-04) -	3.5330e-01 (4.80e-04) -	3.5363e-01 (4.38e-04) -	3.5736e-01 (3.01e-04)
DCF2	4.2456e-01 (2.43e-03) -	4.2447e-01 (2.04e-03) -	4.2382e-01 (2.23e-03) -	4.2529e-01 (2.06e-03) -	4.2691e-01 (1.22e-03)
DCF3	3.3873e-01 (6.39e-04) =	3.3893e-01 (8.06e-04) =	3.3885e-01 (6.75e-04) =	3.3876e-01 (6.07e-04) =	3.3907e-01 (5.45e-04)
DCF4	4.7645e-01 (5.79e-04) -	4.7662e-01 (7.26e-04) -	4.7662e-01 (6.84e-04) -	4.7677e-01 (7.54e-04) -	4.7926e-01 (6.07e-04)
DCF5	5.0274e-01 (1.61e-03) -	5.0132e-01 (1.96e-03) -	5.0186e-01 (1.89e-03) -	5.0258e-01 (2.57e-03) -	5.0943e-01 (1.23e-03)
DCF6	3.1301e-01 (1.26e-03) -	3.1333e-01 (1.62e-03) -	3.1414e-01 (1.72e-03) -	3.1439e-01 (1.80e-03) -	3.2486e-01 (7.55e-04)
DCF7	2.8040e-01 (7.45e-04) =	2.8023e-01 (8.91e-04) =	2.8066e-01 (6.86e-04) =	2.8062e-01 (7.61e-04) =	2.8064e-01 (5.53e-04)
DCF8	5.7284e-01 (1.74e-04) -	5.7288e-01 (1.53e-04) -	5.7281e-01 (1.83e-04) -	5.7285e-01 (1.65e-04) -	5.7323e-01 (1.76e-04)
DCF9	5.7902e-01 (8.54e-04) =	5.7960e-01 (8.18e-04) =	5.7903e-01 (1.24e-03) =	5.7936e-01 (7.53e-04) =	5.7963e-01 (6.35e-04)
DCF10	3.1624e-01 (2.97e-03) =	3.1661e-01 (2.38e-03) =	3.1557e-01 (2.48e-03) =	3.1675e-01 (2.82e-03) =	3.1541e-01 (2.30e-03)
DCP1	3.0403e-01 (1.51e-03) -	3.0471e-01 (8.61e-04) =	3.0453e-01 (1.01e-03) =	3.0432e-01 (7.27e-04) -	3.0488e-01 (5.49e-04)
DCP2	4.9040e-01 (1.94e-03) =	4.9061e-01 (1.56e-03) =	4.9106e-01 (1.88e-03) =	4.9192e-01 (1.88e-03) =	4.9138e-01 (1.98e-03)
DCP3	9.9717e-02 (1.28e-02) +	1.0545e-01 (1.31e-02) +	1.0095e-01 (1.55e-02) +	1.0378e-01 (1.12e-02) +	8.1150e-02 (1.30e-02)
DCP4	3.1056e-01 (1.89e-04) -	3.1056e-01 (2.04e-04) -	3.1060e-01 (2.31e-04) =	3.1061e-01 (1.57e-04) -	3.1073e-01 (1.88e-04)
DCP5	3.6332e-01 (3.34e-02) -	3.6555e-01 (2.73e-02) -	3.7426e-01 (2.86e-02) -	3.7892e-01 (2.70e-02) =	3.9007e-01 (4.80e-03)
DCP6	4.2647e-01 (5.67e-04) +	4.2645e-01 (5.12e-04) +	4.2606e-01 (4.83e-04) +	4.2537e-01 (8.17e-04) +	4.2385e-01 (5.53e-04)
DCP7	5.5459e-01 (6.67e-04) =	5.5464e-01 (7.34e-04) =	5.5491e-01 (5.91e-04) =	5.5474e-01 (7.02e-04) =	5.5475e-01 (8.58e-04)
DCP8	4.6962e-01 (3.94e-03) +	4.6885e-01 (4.73e-03) +	4.6832e-01 (3.04e-03) +	4.6803e-01 (3.52e-03) +	4.6470e-01 (5.22e-03)
DCP9	4.1795e-01 (9.37e-03) =	4.1935e-01 (1.35e-02) =	4.2031e-01 (1.04e-02) =	4.1993e-01 (1.33e-02) =	4.2362e-01 (9.52e-03)
+/-/=	3/10/6	3/8/8	3/7/9	3/8/8	

5.4. Ablation study

To evaluate the contribution of each component, six variants of AMPF are designed. In particular, we examine the effects of the change intensity detection module, the historical elite perturbation strategy, the recent trend prediction strategy, the adaptive memory-prediction fusion mechanism, the dynamic tri-population optimization framework, and the strengthened cooperation mechanism among the three populations.

Specifically, AMPFV0 uses a weak cooperation mechanism to facilitate knowledge transfer between populations. AMPFV1 removes the change intensity detection mechanism. In this variant, the algorithm no longer adaptively estimates the severity of environmental changes, and thus cannot adjust the response intensity according to the degree of change. AMPFV2 removes the historical elite perturbation strategy, so that previously promising solutions are not reused to generate perturbed candidate solutions after environmental changes. AMPFV3 removes the recent trend prediction strategy, and the recent movement of the DCPF across recent environments is not explicitly exploited. AMPFV4 removes the entire adaptive memory-prediction fusion mechanism, which means that the algorithm does not use the fusion of historical elite perturbation and trend prediction to construct the initial population after

environmental changes. AMPFV5 removes the dynamic tri-population optimization framework, and thus the coordinated search among feasibility exploitation, convergence-oriented exploration, and diversity enhancement is disabled. Finally, AMPF introduces a stronger cooperation mechanism among the three populations. Unlike the offspring-level cooperation of AMPFV0, AMPF merges the main population, the convergence auxiliary population, the diversity auxiliary population, and the offspring into a unified shared pool, from which the three populations are synchronously reselected according to their own environmental selection criteria. Although the three populations share the same candidate pool, their role specialization is preserved because they are updated by different selection criteria.

The Friedman average rankings obtained by AMPF and its variants are shown in Figs. 8 and 9, where a smaller ranking indicates better overall performance. In terms of IGD, AMPF obtains the best average ranking of 2.053, followed by AMPFV0 with a ranking of 2.474. The remaining variants obtain worse rankings. Similar observations can be made for the HV results. AMPF again achieves the best ranking of 2.158, while AMPFV0 obtains the second-best ranking of 2.737. The rankings of AMPFV1, AMPFV2, AMPFV3, AMPFV4, and AMPFV5 are 2.947, 4.105, 4.263, 4.947, and 6.842, respectively. These results show that the

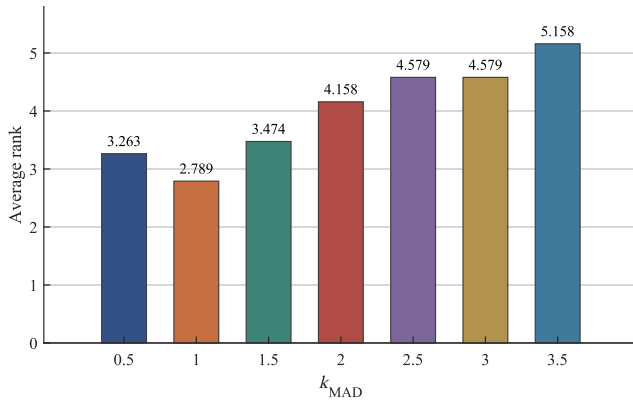


Fig. 7. Average Friedman rankings in terms of HV for the sensitivity analysis of the MAD threshold parameter.

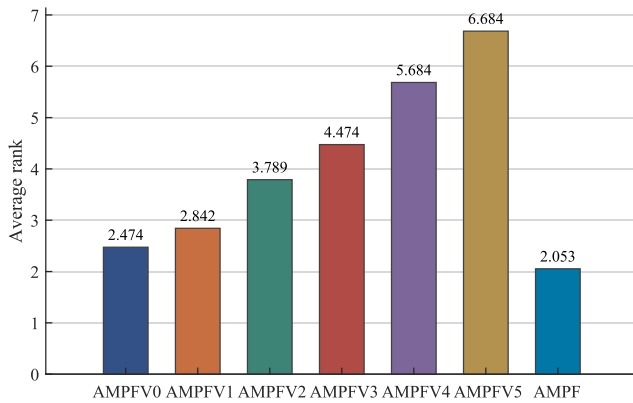


Fig. 8. Average Friedman rankings in terms of IGD for ablation study.

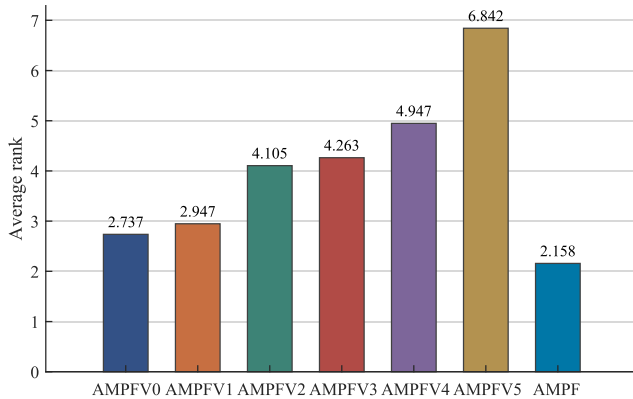


Fig. 9. Average Friedman rankings in terms of HV for ablation study.

shared-pool cooperation mechanism provides a more effective way for the three populations to exchange useful knowledge than the weak cooperation mechanism. Based on this observation, the strong cooperation mechanism is incorporated into the revised version of AMPF.

The results of AMPFV1 show that removing the change intensity detection mechanism leads to less stable performance. Although AMPFV1 obtains competitive results on some instances, its average rankings are worse than those of AMPF and AMPFV0 on both IGD and HV. This indicates that dynamically estimating the severity of environmental changes is useful for controlling the response strength. Without this mechanism, the algorithm may either overreact to mild changes or respond insuf-

Table 5

Statistics of performance comparisons on the real-world problem according to the MHV indicator.

Indicator	Algorithm	$\tau_i = 80$			$\tau_i = 100$		
		Best	Mean	Worst	Best	Mean	Worst
MHV	SKTEA	0.043	0.031	0.020	0.043	0.028	0.019
	DCNSGAI-A	0.034	0.024	0.018	0.032	0.023	0.016
	DCNSGAI-B	0.032	0.020	0.012	0.022	0.015	0.011
	DCNSGA-III	0.073	0.056	0.036	0.069	0.054	0.032
	MP-DFR	0.056	0.041	0.023	0.050	0.035	0.024
	DC-MOEA	0.066	0.038	0.024	0.072	0.039	0.015
	TDCEA	0.059	0.033	0.017	0.052	0.029	0.017
	HATC	0.063	0.043	0.034	0.091	0.041	0.016
	AMPF	0.089	0.073	0.058	0.080	0.068	0.058

ficiently to severe changes, which weakens its robustness across different dynamic scenarios. AMPFV2 and AMPFV3 are used to separately evaluate the two key components of the adaptive memory-prediction fusion mechanism. AMPFV2 performs worse than AMPFV0 and AMPF, demonstrating that historical elite perturbation is important for reusing previously promising solutions and accelerating adaptation after environmental changes. AMPFV3 also obtains inferior average rankings, which shows that recent trend prediction provides useful directional information for tracking the movement of the optimal region. These results indicate that historical elite perturbation and trend prediction contribute from different perspectives. The former improves local recovery around high-quality historical regions, while the latter enhances the ability to anticipate environmental dynamics. AMPFV4 removes the entire adaptive memory-prediction fusion mechanism and obtains substantially worse average rankings than AMPFV0 and AMPF. This confirms that simply relying on the current population or random diversity introduction is insufficient for constructing a well-adapted initial population after environmental changes. By contrast, the proposed adaptive memory-prediction mechanism provides more informative guidance by combining perturbed historical elites with predicted search directions. Therefore, the fusion of memory and prediction is essential for effective dynamic response. AMPFV5 shows the worst performance among all compared variants on both IGD and HV. It obtains the largest Friedman average rankings, namely 6.684 on IGD and 6.842 on HV. This result demonstrates the importance of the dynamic tri-population optimization framework. Without the coordinated search among feasibility exploitation, convergence-oriented exploration, and diversity enhancement, the algorithm has difficulty in balancing convergence, feasibility, and diversity in dynamic constrained multi-objective environments.

Overall, the ablation results verify the effectiveness of each component of AMPF. The change intensity detection mechanism improves the adaptiveness of the environmental response. The historical elite perturbation and recent trend prediction strategies provide complementary knowledge for population reinitialization. The adaptive memory-prediction fusion mechanism integrates these two sources of information to guide the search after environmental changes. The dynamic tri-population framework plays a fundamental role in balancing feasibility, convergence, and diversity. Moreover, the newly introduced shared-pool cooperation mechanism further enhances information exchange among the three populations and leads to the best overall performance. These results demonstrate that the proposed components are complementary and jointly contribute to the effectiveness of the AMPF.

5.5. Real-world experiments on dynamic raw ore allocation

To further evaluate the practical applicability of AMPF, an additional experiment is conducted on a real-world dynamic raw ore allocation problem in mineral processing. Raw ore allocation aims to assign different quantities of raw ores under changing production conditions, to optimize multiple conflicting production objectives, such as concentrate yield, concentrate grade, metal recovery, beneficiation ratio, and pro-

duction cost. In practical mineral processing systems, the production environment is typically nonstationary. Variations in equipment capacity, equipment runtime, maintenance status, and processing capability may change the objective functions, constraint boundaries, and model parameters over time. Therefore, raw ore allocation can be formulated as a dynamic multi-objective optimization problem (Ding et al., 2019). In this real-world problem, the environmental changes are mainly caused by variations in the capacity and runtime of the ball mill grinding process, which is a bottleneck subprocess in mineral processing. Compared with synthetic benchmark problems, this problem has more realistic characteristics, including multiple conflicting production objectives, time-varying equipment-capacity constraints, and irregular changes in the feasible regions.

Because the true PF of this real-world problem is unknown, MHV is adopted to evaluate the performance of the compared algorithms. A larger MHV value indicates better overall performance in terms of convergence and diversity across dynamic environments. Two change-frequency settings are considered: $\tau_i = 80$ and $\tau_i = 100$. The statistical results are reported in Table 5. When $\tau_i = 80$, AMPF achieves the best MHV values among all compared algorithms. Specifically, AMPF obtains a mean MHV of 0.073, while the second-best mean value is obtained by DCNSGA-III with 0.056. This indicates that AMPF can obtain better overall solution quality under relatively frequent environmental changes. Moreover, the worst MHV value of AMPF is 0.058, which is higher than the worst-case values of all compared algorithms. This suggests that AMPF has better robustness across independent runs. When $\tau_i = 100$, AMPF still achieves the best mean and worst-case MHV values. Although HATC obtains the best single-run MHV value of 0.091, its mean and worst MHV values are 0.041 and 0.016, respectively. In contrast, AMPF obtains a mean MHV of 0.068 and a worst MHV of 0.058. This result indicates that AMPF provides more stable performance rather than only achieving occasional high-quality solutions. Compared with DCNSGA-III, which obtains the second-best mean MHV of 0.054, AMPF still shows a clear advantage under this setting.

The competitive performance of AMPF on this real-world problem can be attributed to the synergy between its dynamic response mechanism and tri-population optimization framework. The memory-prediction fusion mechanism enables the reuse of historical information to accelerate the response to environmental changes. Meanwhile, the main population, the convergence auxiliary population, and the diversity auxiliary population jointly refine the population after each environmental change. Therefore, AMPF can maintain good dynamic tracking ability even when the problem involves practical equipment-capacity variations and time-varying constraints. Overall, the real-world raw ore allocation experiment demonstrates that AMPF is applicable beyond synthetic benchmark problems. It can be applied to practical dynamic optimization scenarios with changing constraints and conflicting production objectives.

6. Conclusion

This paper proposes AMPF, an adaptive memory-prediction fusion-based evolutionary algorithm for dynamic constrained multi-objective optimization. AMPF introduces a change intensity detection module to identify environmental changes and quantify their severity using the change intensity parameter C_i . Based on this parameter, the adaptive memory-prediction fusion mechanism exploits recent change trends and perturbed historical elite solutions to construct a suitable initial population for the new environment. The fusion design alleviates the bias caused by relying solely on either prediction or memory and improves adaptability under different environmental change patterns. AMPF also develops a dynamic tri-population framework to coordinate feasibility, convergence, and diversity during the subsequent evolutionary search. The main population solves the original constrained task using constrained Pareto dominance. The convergence and diversity auxiliary populations promote convergence and distribution maintenance

through unconstrained Pareto dominance and reference-vector-based association, respectively. By dynamically allocating computational resources between the two auxiliary populations according to the detected change intensity, AMPF better balances convergence enhancement and diversity preservation. Experimental results on 76 benchmark test instances and a real-world raw-ore allocation problem demonstrate that AMPF achieves competitive or superior performance compared with eight state-of-the-art DCMOEAs. Moreover, the change intensity detection module, dynamic response mechanism, and tri-population framework provide transferable design ideas for other dominance-based DCMOEAs.

Although AMPF shows promising performance, it has several limitations. Because AMPF uses a memory-prediction fusion mechanism, its prediction component is more effective when consecutive environments have useful temporal correlations. When environmental changes are abrupt, noisy, non-recurrent, or weakly related to previous environments, prediction accuracy may decrease, and inaccurate predicted solutions may mislead the search if they are not properly filtered. In addition, the performance of AMPF depends on reliable change detection. Delayed detection may cause the algorithm to search around outdated promising regions, whereas false detections may trigger unnecessary population reconstruction and disturb the evolutionary process. The limitations suggest several future directions, including prediction confidence estimation, adaptive switching between prediction-based and memory-based responses, and surrogate- or transfer-learning-assisted acceleration.

CRedit authorship contribution statement

Huaiwen Liu: Conceptualization, Methodology; **Baosheng Li:** Software, Writing – original draft; **Hongbiao Tian:** Supervision; **Zhiyang Zhang:** Methodology, Writing – original draft; **Qianlong Dang:** Funding acquisition, Validation, Visualization; **Junhu Ruan:** Writing – review & editing, Software.

Declaration of competing interest

In this research activity, all the authors were involved in the data collection and preprocessing phase, model constructing, empirical research, results analysis and discussion, and manuscript preparation. All authors have approved the submitted manuscript. The authors declare no conflict of interest.

Acknowledgement

This work was supported in part by the National Nature Science Foundation of China under Grant 72271202 and 12501711, in part by the Youth Project of the [Natural Science Foundation of Shaanxi Province](#) under Grant 2025JC-YBQN-972, in part by the Postdoctoral Research Project of Shaanxi Province under Grant 2025BSHEDZZ012, in part by the [China Postdoctoral Science Foundation](#) under Grant 2024M762632, in part by the [Social Science Foundation of Shaanxi](#) under Grant 2024R027, in part by [Chinese Universities Scientific Fund](#) under Grant 2452026010 and 2452026073, in part by the Fundamental Research Funds for the [Central Universities](#) under Grant 2452026134.

Data availability

No data was used for the research described in the article.

References

- Azzouz, R., Bechikh, S., & Said, L. B. (2017). A dynamic multi-objective evolutionary algorithm using a change severity-based adaptive population management strategy. *Soft Computing*, 21(4), 885–906.

- Azzouz, R., Bechikh, S., Said, L. B., & Trabelsi, W. (2018). Handling time-varying constraints and objectives in dynamic evolutionary multi-objective optimization. *Swarm and Evolutionary Computation*, 39, 222–248. <https://doi.org/10.1016/j.swevo.2017.10.005>
- Bao, Q., Wang, M., Yang, S., Dai, G., & Chen, X. (2025). A coevolutionary response framework for dynamic constrained multi-objective optimization problems. *IEEE Transactions on Evolutionary Computation*, (pp. 1–1). <https://doi.org/10.1109/TEVC.2025.3595410>
- Branke, J. (1999). Memory enhanced evolutionary algorithms for changing optimization problems. In *Proceedings of the 1999 congress on evolutionary computation-CEC99* (cat. no. 99TH8406) (pp. 1875–1882). (vol. 3). <https://doi.org/10.1109/CEC.1999.785502>
- Cao, L., Xu, L., Goodman, E. D., Bao, C., & Zhu, S. (2020). Evolutionary dynamic multi-objective optimization assisted by a support vector regression predictor. *IEEE Transactions on Evolutionary Computation*, 24(2), 305–319. <https://doi.org/10.1109/TEVC.2019.2925722>
- Chen, G., Guo, Y., Jiang, M., Yang, S., Zhao, X., & Gong, D. (2024a). A subspace-knowledge transfer based dynamic constrained multiobjective evolutionary algorithm. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 8(2), 1500–1512. <https://doi.org/10.1109/TETCI.2023.3336918>
- Chen, G., Guo, Y., Wang, Y., Liang, J., Gong, D., & Yang, S. (2024b). Evolutionary dynamic constrained multiobjective optimization: Test suite and algorithm. *IEEE Transactions on Evolutionary Computation*, 28(5), 1381–1395. <https://doi.org/10.1109/TEVC.2023.3313689>
- Chen, Q., Ding, J., Yang, S., & Chai, T. (2020). A novel evolutionary algorithm for dynamic constrained multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 24(4), 792–806. <https://doi.org/10.1109/TEVC.2019.2958075>
- Chen, Q., Ding, J., Yen, G. G., Yang, S., & Chai, T. (2024c). Multipopulation evolution-based dynamic constrained multiobjective optimization under diverse changing environments. *IEEE Transactions on Evolutionary Computation*, 28(3), 763–777. <https://doi.org/10.1109/TEVC.2023.3241762>
- Dang, Q., Shang, W., Huang, Z., & Yang, S. (2025). Constrained multi-objective optimization assisted by convergence and diversity auxiliary tasks. *Engineering Applications of Artificial Intelligence*, 139, 109546. <https://doi.org/10.1016/j.engappai.2024.109546>
- Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2), 182–197. <https://doi.org/10.1109/4235.996017>
- Deb, K., Udaya Bhaskara, R. N., & Karthik, S. (2007). Dynamic multi-objective optimization and decision-making using modified NSGA-II: A case study on hydro-thermal power scheduling. In *Evolutionary multi-criterion optimization* (pp. 803–817). Springer Berlin Heidelberg.
- Ding, J., Yang, C., Xiao, Q., Chai, T., & Jin, Y. (2019). Dynamic evolutionary multiobjective optimization for raw ore allocation in mineral processing. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 3(1), 36–48. <https://doi.org/10.1109/TETCI.2018.2812897>
- Fan, Z., Li, W., Cai, X., Li, H., Wei, C., Zhang, Q., Deb, K., & Goodman, E. (2019). Push and pull search for solving constrained multi-objective optimization problems. *Swarm and Evolutionary Computation*, 44, 665–679.
- Feng, L., Zhou, W., Liu, W., Ong, Y.-S., & Tan, K. C. (2022). Solving dynamic multiobjective problem via autoencoding evolutionary search. *IEEE Transactions on Cybernetics*, 52(5), 2649–2662. <https://doi.org/10.1109/TCYB.2020.3017017>
- Gao, X., Dang, Q., Feng, X., Li, X., Liu, Q., & Zhan, T. (2026). Multiknowledge adaptive transfer framework and auxiliary value assessment strategy for constrained multiobjective optimization. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, (pp. 1–14). <https://doi.org/10.1109/TSMC.2026.3669395>
- Goh, C.-K., & Tan, K. C. (2009). A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 13(1), 103–127. <https://doi.org/10.1109/TEVC.2008.920671>
- Guo, Y., Chen, G., Yue, C., Liang, J., Wang, Y., & Yang, S. (2022). Benchmark problems for CEC2023 competition on dynamic constrained multiobjective optimization. Technical report, School of Computer Science and Informatics, De Montfort University, Leicester, UK, 1–12.
- Jain, H., & Deb, K. (2014). An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: Handling constraints and extending to an adaptive approach. *IEEE Transactions on Evolutionary Computation*, 18(4), 602–622. <https://doi.org/10.1109/TEVC.2013.2281534>
- Jiang, M., Huang, Z., Qiu, L., Huang, W., & Yen, G. G. (2018). Transfer learning-based dynamic multiobjective optimization algorithms. *IEEE Transactions on Evolutionary Computation*, 22(4), 501–514. <https://doi.org/10.1109/TEVC.2017.2771451>
- Jiang, M., Wang, Z., Qiu, L., Guo, S., Gao, X., & Tan, K. C. (2021). A fast dynamic evolutionary multiobjective algorithm via manifold transfer learning. *IEEE Transactions on Cybernetics*, 51(7), 3417–3428. <https://doi.org/10.1109/TCYB.2020.2989465>
- Li, K., Chen, R., Fu, G., & Yao, X. (2019). Two-archive evolutionary algorithm for constrained multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 23(2), 303–315. <https://doi.org/10.1109/TEVC.2018.2855411>
- Liang, J., Ban, X., Yu, K., Qu, B., & Qiao, K. (2021). Differential evolution with rankings-based fitness function for constrained optimization problems. *Applied Soft Computing*, 113, 108016. <https://doi.org/10.1016/j.asoc.2021.108016>
- Liang, J., Ban, X., Yu, K., Qu, B., Qiao, K., Yue, C., Chen, K., & Tan, K. C. (2023). A survey on evolutionary constrained multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 27(2), 201–221. <https://doi.org/10.1109/TEVC.2022.3155533>
- Liang, J., Lin, H., Yue, C., Ban, X., & Yu, K. (2024). Evolutionary constrained multi-objective optimization: A review. *Vicinagearth*, 1(1), 5.
- Liang, Z., Zheng, S., Zhu, Z., & Yang, S. (2019). Hybrid of memory and prediction strategies for dynamic multiobjective optimization. *Information Sciences*, 485, 200–218. <https://doi.org/10.1016/j.ins.2019.01.066>
- Liu, Z., Han, F., Ling, Q., Han, H., & Jiang, J. (2025). Constraint-Pareto dominance and diversity enhancement strategy based evolutionary algorithm for solving constrained multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 29(6), 2771–2784. <https://doi.org/10.1109/TEVC.2024.3525153>
- Ma, H., Wei, H., Tian, Y., Cheng, R., & Zhang, X. (2021). A multi-stage evolutionary algorithm for multi-objective optimization with complex constraints. *Information Sciences*, 560, 68–91. <https://doi.org/10.1016/j.ins.2021.01.029>
- Mala-Jetmarova, H., Sultanova, N., & Savic, D. (2017). Lost in optimisation of water distribution systems? a literature review of system operation. *Environmental Modelling & Software*, 93, 209–254.
- Ming, F., Gong, W., Wang, L., & Gao, L. (2022). Constrained multiobjective optimization via multitasking and knowledge transfer. *IEEE Transactions on Evolutionary Computation*, 28(1), 77–89.
- Nguyen, T. T., & Yao, X. (2009). Benchmarking and solving dynamic constrained problems. In *2009 IEEE Congress on evolutionary computation* (pp. 690–697). <https://doi.org/10.1109/CEC.2009.4983012>
- Nguyen, T. T., & Yao, X. (2012). Continuous dynamic constrained optimization-the challenges. *IEEE Transactions on Evolutionary Computation*, 16(6), 769–786. <https://doi.org/10.1109/TEVC.2011.2180533>
- Peng, Z., Zheng, J., Zou, J., & Liu, M. (2015). Novel prediction and memory strategies for dynamic multiobjective optimization. *Soft Computing*, 19(9), 2633–2653.
- Qiao, K., Liang, J., Liu, Z., Yu, K., Yue, C., & Qu, B. (2023). Evolutionary multitasking with global and local auxiliary tasks for constrained multi-objective optimization. *IEEE/CAA Journal of Automatica Sinica*, 10(10), 1951–1964.
- Ray, T., Tai, K., & Seow, K. C. (2001). Multiobjective design optimization by an evolutionary algorithm. *Engineering Optimization*, 33(4), 399–424. <https://doi.org/10.1080/03052150108940926>
- Takahama, T., & Sakai, S. (2010). Constrained optimization by the ϵ constrained differential evolution with an archive and gradient-based mutation. In *IEEE congress on evolutionary computation* (pp. 1–9). <https://doi.org/10.1109/CEC.2010.5586484>
- Tan, B., Ma, H., Mei, Y., & Zhang, M. (2018). Evolutionary multi-objective optimization for web service location allocation problem. *IEEE Transactions on Services Computing*, 14(2), 458–471.
- Tian, Y., Zhang, T., Xiao, J., Zhang, X., & Jin, Y. (2020). A coevolutionary framework for constrained multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 25(1), 102–116.
- Wang, J., Zhou, Y., Wang, Y., Zhang, J., Chen, C. L. P., & Zheng, Z. (2016). Multiobjective vehicle routing problems with simultaneous delivery and pickup and time windows: Formulation, instances, and algorithms. *IEEE Transactions on Cybernetics*, 46(3), 582–594. <https://doi.org/10.1109/TCYB.2015.2409837>
- Wang, Y., Hu, C., Ming, F., Li, Y., Gong, W., & Gao, L. (2025). A diversity-enhanced tri-stage framework for constrained multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 29(5), 2047–2061. <https://doi.org/10.1109/TEVC.2024.3458079>
- Wei, J., & Wang, Y. (2012). Hyper rectangle search based particle swarm algorithm for dynamic constrained multi-objective optimization problems. In *2012 IEEE Congress on evolutionary computation* (pp. 1–8). <https://doi.org/10.1109/CEC.2012.6256137>
- Woldeesenbet, Y. G., & Yen, G. G. (2009). Dynamic evolutionary algorithm with variable relocation. *IEEE Transactions on Evolutionary Computation*, 13(3), 500–513. <https://doi.org/10.1109/TEVC.2008.2009031>
- Woldeesenbet, Y. G., Yen, G. G., & Tessema, B. G. (2009). Constraint handling in multi-objective evolutionary optimization. *IEEE Transactions on Evolutionary Computation*, 13(3), 514–525. <https://doi.org/10.1109/TEVC.2008.2009032>
- Xia, Z., Liu, Y., Lu, J., Cao, J., & Rutkowski, L. (2021). Penalty method for constrained distributed quaternion-variable optimization. *IEEE Transactions on Cybernetics*, 51(11), 5631–5636. <https://doi.org/10.1109/TCYB.2020.3031687>
- Xu, X., Tan, Y., Zheng, W., & Li, S. (2018). Memory-enhanced dynamic multi-objective evolutionary algorithm based on Lp decomposition. *Applied Sciences*, 8(9), 1673.
- Yu, K., Zhang, D., Liang, J., Qu, B., Liu, M., Chen, K., Yue, C., & Wang, L. (2024). A framework based on historical evolution learning for dynamic multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 28(4), 1127–1140. <https://doi.org/10.1109/TEVC.2023.3290485>
- Zapotecas-Martínez, S., & Ponsich, A. (2020). Constraint handling within MOEA/D through an additional scalarizing function. In *Proceedings of the 2020 genetic and evolutionary computation conference GECCO '20* (pp. 595–602). New York, NY, USA: Association for Computing Machinery. <https://doi.org/10.1145/3377930.3390240>
- Zhang, D., Yu, K., Liang, J., Qiao, K., Qu, B., Chen, K., Yue, C., & Wang, L. (2025a). History-assisted two-state auxiliary task collaboration approach for dynamic constrained multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 29(6), 2386–2400. <https://doi.org/10.1109/TEVC.2024.3425756>
- Zhang, D., Zhang, K., Yu, K., Liang, J., Qiao, K., Qu, B., Chen, K., & Yue, C. (2025b). Dynamic constrained multiobjective evolutionary algorithm with multipopulation prediction and dynamic fusion ranking. *IEEE Transactions on Evolutionary Computation*, (pp. 1–1). <https://doi.org/10.1109/TEVC.2025.3569387>
- Zhou, A., Jin, Y., & Zhang, Q. (2014). A population prediction strategy for evolutionary dynamic multiobjective optimization. *IEEE Transactions on Cybernetics*, 44(1), 40–53. <https://doi.org/10.1109/TCYB.2013.2245892>
- Zhou, B., & Huang, Y. (2023). Dynamic cyclic kitting part-feeding scheduling for mixed-model assembly line by a hybrid quantum-behaved particle swarm optimization. *Robotic Intelligence and Automation*, 43(3), 267–289.
- Zhu, Q., Zhang, Q., & Lin, Q. (2020). A constrained multiobjective evolutionary algorithm with detect-and-escape strategy. *IEEE Transactions on Evolutionary Computation*, 24(5), 938–947. <https://doi.org/10.1109/TEVC.2020.2981949>
- Zitzler, E., & Thiele, L. (1999). Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Transactions on Evolutionary Computation*, 3(4), 257–271. <https://doi.org/10.1109/4235.797969>