Optimal Linear Crossover for Mitigating Negative Transfer in Evolutionary Multitasking

Zhaobo Liu, Jianhua Yuan, Haili Zhang, Tao Zeng, and Zexuan Zhu, *Senior Member, IEEE*

***Abstract*—Evolutionary multitasking algorithms use information exchange among individuals in a population to solve multiple optimization problems simultaneously. Negative transfer is a critical factor that affects the performance of evolutionary multitasking algorithms. In this study, we propose an innovative approach to mitigate negative transfer in evolutionary multitasking algorithms. The proposed approach is grounded in rigorous theoretical analysis, which provides valuable theoretical insights into the design of an optimal linear crossover operator for mitigating negative transfer. By identifying interpretable conditions, we establish a solid theoretical foundation to prevent negative transfer in diverse scenarios. Building upon these findings, we theoretically derive a closed-form expression for the optimal crossover operator and propose practical design methods based on approximations. Furthermore, we integrate the proposed optimal crossover operator into a fundamental evolutionary multitasking algorithm framework. The resultant algorithm is comparable or superior to other state-of-the-art methods. Empirical validation through comprehensive experiments confirms the effectiveness of our theoretical findings.**

***Index Terms*—Evolutionary algorithms, multitask optimization, optimal crossover**

I. Introduction

EVOLUTIONARY algorithms (EAs) are metaheuristic optimization algorithms inspired by the principles of biological evolution [1]. EAs simulate the biological evolution process mainly via crossover, mutation, selection, and evaluation. EAs exhibit robust search capabilities and scalability, making them applicable in various research domains, including artificial intelligence, machine learning, environmental management, and complex optimization problems [2]–[7]. Within the realm of optimization problems, EAs have spawned numerous innovative and practical algorithms customized for various optimization task requirements. Notably, multi-task optimization (MTO), a recent development, have garnered significant attention and research interest in the evolutionary computation community. The primary goal of MTO is to simultaneously optimize multiple problems, fostering mutual influence and knowledge transfer among different tasks to achieve superior results compared with individual task optimization. To tackle this emerging challenge, evolutionary multitasking (EMT) employs EAs to simultaneously solve MTO tasks. By exploiting the advantages of knowledge transfer, EMT has been demonstrated to outperform single-task EAs on various optimization problems and real-world applications.

The multifactorial evolutionary algorithm (MFEA) [8] and its variants [9], [10] represent the most widely used EMT algorithms. Fundamentally, the key to design efficient MFEAs is to enhance knowledge transfer efficiency or alleviate the threat of negative transfer. Various methods have been proposed for selecting the most suitable individuals for knowledge transfer. For instance, decision trees have been used in [11] to identify promising individuals that enhance transfer efficiency, while the work [12] introduced the concept of "skill membership" to select individuals best suited for each task, using the convergence rate as a guideline to prevent population from being trapped in local optima. Unlike the process of selecting appropriate individuals, Ma et al. [13] utilized the Kullback-Leibler divergence to calculate a finer-grained genetic similarity. Surrogate models have also been employed to create helper problems with smoother and simpler functions, facilitating efficient knowledge transfer [14], [15]. In MFEA, a parameter known as the random mating probability (*rmp*) is also usually used to regulate the frequency of knowledge transfer leading to better transfer efficiency. Employing a fixed *rmp* value without considering the differences between tasks can easily lead to negative transfer. Therefore, adaptive *rmp* setting approaches have been explored in [16]–[18]. Similar to the idea of using adaptive *rmp*, Xu et al. [19] calculated the probability of knowledge transfer based on historical knowledge, with the aim of controlling the frequency of knowledge transfer. Some studies mitigated negative transfer by calculating the similarity between tasks and reducing the influence among dissimilar tasks. For example, Da et al. [20] developed a mixture model to identify similarities between source and target. Mapping

This work was supported in part by the National Nature Science Foundation of China under Grants 12401585 and 62471310; in part by the National Natural Science Foundation of China under Grant 12401393; in part by the National Science Foundation of Guangdong Province of China under Grant 2024A1515011542; in part by the Guangdong Provincial Ordinary Universities Youth Innovative Talent Project (Natural Science) under Grant 2023KQNCX064, and in part by the Open Funding of Key Laboratory of Dependable Service Computing in Cyber Physical Society, Ministry of Education of China, under Grant CPSDSC202004. *(Corresponding author: Zexuan Zhu.)*

Zhaobo Liu is with the Institute for Advanced Study, Shenzhen University, Shenzhen 518060, China (e-mail: liuzhaobo@szu.edu.cn).

Jianhua Yuan is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China (e-mail: yuanjianhua2022@email.szu.edu.cn).

Haili Zhang is with the School of Undergraduate Education, Shenzhen Polytechnic University, Shenzhen 518055, China (e-mail: zhanghl@szpu.edu.cn).

Tao Zeng is with the National Engineering Laboratory for Big Data System Computing Technology, Shenzhen University, Shenzhen 518060 (e-mail: szuztao@163.com).

Zexuan Zhu is with the National Engineering Laboratory for Big Data System Computing Technology, Shenzhen University, Shenzhen 518060, China, also with Shenzhen City Key Laboratory of Embedded System Design, Shenzhen University, Shenzhen 518060, and also with Key Laboratory of Dependable Service Computing in Cyber Physical Society, Ministry of Education of China, Chongqing University, Chongqing 400044, China (e-mail: zhuzx@ szu.edu.cn).

methods have also been employed to reduce negative transfer from the source domain to the target domain in [21]–[26].

Among the aforementioned MFEAs, knowledge transfer between different tasks is often accomplished through the utilization of crossover operators. Therefore, the selection or design of crossover operators plays a key role in facilitating positive knowledge transfer and mitigating negative transfer. By carefully choosing or designing appropriate crossover operators, the efficiency of positive knowledge transfer can be enhanced, and the occurrence of negative transfer can be effectively alleviated. Adaptive selection of crossover operators based on information collected during evolution has been explored in [27]. Li et al. [28] adaptively selected Binomial crossover and SBX crossover to ensure that each task is assigned the most suitable crossover operator. In addition, to the best of our knowledge, while designing crossover operators is common in single-task scenarios [29], [30], it is relatively rare in multitask environments. Based on diffusion gradient descent, Liu et al. [31] developed a new crossover operator that can interpret knowledge transfer in MTO as the transfer of convexity between different tasks. In [32], individuals are divided into multiple blocks, leading to the enhancement of the crossover operator based on this division. In summary, while these adaptive strategies or design methods of crossover operators have shown the potential to enhance algorithm performance or provide certain interpretability, the relationship between negative transfer and crossover operators remains unclear. Currently, there is a lack of rigorous theoretical guidance on how to design crossover operators to avoid negative transfer.

In this study, we introduce a novel theoretical framework for the design of optimal crossover operators, aiming to enhance knowledge transfer efficiency and mitigate negative transfer in MFEAs. When prior knowledge of global optimal solutions is available, we evaluate the occurrence of negative transfer by measuring the reduction in the Euclidean distance between the population after crossover and the global optimal solutions. By minimizing this Euclidean distance, we theoretically derive the optimal design of crossover operators, thereby optimizing the efficiency of knowledge transfer. However, because the global optimal solution for each task is usually unknown in real-world problems, we propose a practical crossover operator as an approximation of the theoretically optimal counterpart. Building upon this novel optimal crossover (OC) operator, we develop a simple yet efficient EMT algorithm namely MFEA-OC that exhibits remarkable performance through extensive experimentation. The contributes of this work are highlighted as follows:

1) We propose the conditions under which a linear crossover operator can avoid negative transfer in different scenarios. These conditions have theoretical support and offer valuable guidance for the design of EMT algorithms.
2) Based on rigorous theoretical analysis, a closed-form expression for the OC operator is derived, and we provide some practical design methods based on this formula. This could represents a substantial advancement, as the design of crossover operators often lacks strong theoretical foundations.
3) We integrate the proposed OC operator into the basic MFEA framework and validate our theoretical findings through extensive experiments. This demonstrates the potential of OC operators for applications in MTO. The proposed OC operatore can also seamlessly applicable to other EMT algorithms.

The remainder of this article is organized as follows. Section II provides a brief introduction to MTO and MFEA. Section III presents the theoretical foundations of the optimal crossover operator and the corresponding design principles. Based on this investigation, the details of the proposed MFEA-OC are presented in Section IV. The experimental results and discussion are contained in Section V. Finally, Section VI concludes this study.

II. PRELIMINARIES

In this section, we briefly introduce the preliminaries of MTO and classical MFEA to make this article self-contained.

A. Multitask Optimization Problem

MTO refers to the process of simultaneously solving multiple related optimization tasks. Let us consider an MTO problem with n tasks that need to be addressed concurrently. For the i-th task, we define its objective function as $f_i(\theta_i)$, where $\theta_i \in \mathbb{R}^{d_i}$ represents the decision variable vector for the i-th task. Our goal is to find a set of solutions $\{\theta_1^*, \theta_2^*, \ldots, \theta_n^*\}$ with each for a corresponding task, such that all objective functions of the tasks can be minimized simultaneously. Mathematically, an MTO problem can be formulated as the following minimization problem:

$$\{\arg\min f_1(\theta_1), \arg\min f_2(\theta_2), \ldots, \arg\min f_n(\theta_n)\}.$$

By leveraging knowledge transfer among different tasks, MTO problem solvers like EMT algorithms provide a distinct advantage by enabling faster and more efficient task solving compared to their single-task optimization counterparts. For instance, some scholars have applied EMT algorithms to solve classical NP-hard problems such as the traveling salesman problem and vehicle scheduling dilemmas [33]–[35]. These applications demonstrate superiority of MTO problem solvers to the single-task counterparts and the potential applicability of MTO in addressing real-world issues.

The challenge in solving MTO problems lies in effectively managing the interaction and knowledge transfer among the tasks. There can be both positive transfer, where the optimization process of one task benefits the others, and negative transfer, where it hampers other tasks. To leverage positive transfer and minimize negative transfer among tasks, various strategies have been proposed, such as parameter model sharing, knowledge sharing, and designing specific crossover operators.

B. Multifactorial Evolutionary Algorithm

MFEA solves MTO problems by utilizing evolutionary principles and knowledge transfer. To realize knowledge transfer between individuals, combined with the algorithmic process of

EAs, the authors of the MFEA proposed corresponding definitions to quantify the attributes possessed by each individual as follows:

1) Factorial cost: Let T_j be a given task. The factorial cost Ψ_j^i of individual p_i for T_j is defined as $\Psi_j^i = \lambda \cdot \delta_j^i + f_j^i$, where λ is a large penalizing multiplier, f_j^i represents the objective value of p_i, and δ_j^i represents the total constraint violation of p_i for T_j. If p_i is feasible with respect to T_j (i.e., has zero constraint violation), then $\Psi_j^i = f_j^i$.
2) Factorial rank: the factorial rank r_j^i of individual p_i on task T_j denotes the index of p_i in the sorted population with the factorial cost Ψ_i^j in ascending order.
3) Scalar fitness: The scalar fitness φ_i of an individual p_i is calculated by $\varphi_i = 1/\min_{j=1,\ldots,n}\{r_j^i\}$. If $\varphi_i > \varphi_j$, it means that p_i domain p_j.
3) Skill factor: The skill factor τ_i denote the task that individual p_i is most effective, i.e., $\tau_i = \min_{j=1,\ldots,n}\{r_j^i\}$.

Algorithm 1: The Pseudo-code of MFEA

Input: N (population size), rmp (random mating probability)
Output: A series of solutions

```
Initial the population P with size N;
while the stop condition is not satisfied do
    if the selected parents have different skill factor or rand < rmp then
        Perform vertical cultural transmission to generate offspring individuals;
    else
        Perform variation to generate offspring individuals;
    Evaluate the generated offspring population O;
    Select the N fittest individuals from P ∪ O;
```

The MFEA framework, outlined in Algorithm 1, begins with the generation of a population in a unified expression space with dimension $d = \max_i d_i$. Each individual is randomly assigned a skill factor and evaluated based on factorial cost. Assortative mating and vertical cultural transmission mechanisms are employed in each generation to generate offspring through crossover and mutation operators. Genetic information exchange facilitates knowledge sharing between tasks, and individuals with different skill factors can mate with a certain probability to optimize each task. Following the generation of offspring, the algorithm updates the factorial cost, factorial rank, scalar fitness, and skill factor of each individual. An elite-based environmental selection process is used to create a new population from the union of parent and offspring populations. This evolutionary procedure continues until a stopping criterion is met. For a more comprehensive understanding and detailed insights, the reader is referred to [8].

III. Theoretic Foundations

In this section, we clarify the theoretical foundations of our proposed algorithm, which are crucial for understanding the design of the optimal crossover operator. Although extensive research has been conducted on knowledge transfer in MTO, there is no universally recognized standard for evaluating the effectiveness of a crossover operator in transferring knowledge or detecting negative transfer. Therefore, we adopt a novel perspective to address this issue.

Given n optimization tasks, denoted as $f_1, \ldots, f_n$, and their corresponding optimal solutions as $\theta_1^*, \ldots, \theta_n^*$, respectively. With a given parent population P consisting of some d-dimensional vectors, we define the crossover operator $\mathcal{C}$ as a mapping: $\mathbb{A} \to \mathbb{A}$ (i.e., the domain and range of the mapping are both $\mathbb{A}$), where

$$\mathbb{A} = \bigcup_{m=1}^{\infty} \left\{ X \mid X = \{x_1, \ldots, x_m\}, x_i \in \mathbb{R}^d, i = 1, \ldots, m \right\}.$$

$\mathcal{C}(P)$ represents the offspring population. $\mathbb{A}$ represents a set that contains all possible parents or offspring, and m represents the number of individuals in the parent or offspring population. Since we consider the general case, m can take any positive integer, i.e., we allow the number of individuals in P and $\mathcal{C}(P)$ to be equal to any positive integer.

Next, we denote $\text{dist}(x, S)$ as the distance between a point x and a set S, specifically as $\inf_{y \in S} \|x - y\|$. Note that S may be an open set, in which case there is no point in S strictly closest to x. Therefore, using $\inf$, i.e., the infimum of a set, guarantees the rigour of the definition of "distance" in this context. Ideally, if we can design $\mathcal{C}$ such that the loss function

$$\sum_{i=1}^{n} \text{dist}^2(\theta_i^*, \mathcal{C}(P)) = \sum_{i=1}^{n} \inf_{y \in \mathcal{C}(P)} \|\theta_i^* - y\|^2$$

equals zero, then we have found the optimal solutions for all tasks. However, directly finding such a mapping is impossible. Therefore, we approach the problem by considering the following optimization problem:

$$\min_{\mathcal{C} \in \mathcal{K}} \sum_{i=1}^{n} \text{dist}^2(\theta_i^*, \mathcal{C}(P)) \tag{1}$$

where $\mathcal{K}$ is a specific class of crossover operators.

The problem in Eq. (1) is unsolvable directly due to the unknown values of $\theta_1^*, \ldots, \theta_n^*$. However, by assuming that $\theta_1^*, \ldots, \theta_n^*$ are known, we can solve Eq. (1) to derive the optimal form of the crossover operator under ideal conditions. This approach could enhance the understanding of the problem and facilitate the design of the crossover operator. In this paper, we simplify the problem by considering $\mathcal{K}$ as a class of linear crossover operators, denoted as

$$\mathcal{K} = \{\mathcal{C} \mid \mathcal{C}(P) \subset \text{aff}(P)\}, \tag{2}$$

where $\text{aff}(P)$ is the affine hull of P:

$$\text{aff}(P) = \bigcup_{k \geq 1} \left\{ \sum_{i=1}^{k} \alpha_i x_i : \alpha_i \in \mathbb{R}, x_i \in P, \text{ and } \sum_{i=1}^{k} \alpha_i = 1 \right\}. \tag{3}$$

Note that the linear crossover operators defined in Eq. (2), if restricted to $\alpha_i \geq 0$ and $k = 2$ in Eq. (3), is equivalent to the Geometric crossover in [36] under the metric space being Euclidean. We consider crossover between any number

of parents in general instead of only two. Since we do not restrict $\alpha_i \geq 0$, the scope of the defined operators goes beyond convex evolutionary search. Moreover, inspired by [36], it might be possible to consider the problems in this paper under other metrics as well, such as Hamming distance, Manhattan distance, Swap distance, and Levenshtein distance.

A. Negative Transfer

Before addressing problem Eq. (1), we consider the issue of negative transfer within this framework. We say that $\mathcal{C}$ induces negative transfer if the following inequality holds:

$$\sum_{i=1}^{n} \text{dist}^2(\theta_i^*, \mathcal{C}(P)) > \sum_{i=1}^{n} \text{dist}^2(\theta_i^*, P). \tag{4}$$

Therefore, a natural question arises: which types of $\mathcal{C}$ can prevent negative transfer? In this subsection, we aim to provide a partial answer to this question for $\mathcal{C}$ belonging to $\mathcal{K}$ defined in Eq. (2).

Before presenting the main theoretical results, we explain the reason for choosing Eq. (4) to define negative transfer. The definition implies that we prioritize "finding the global optimum" over "finding better fitness values" as the primary goal of population evolution. Referring to [37, Section 3], these two goals correspond to heuristic search and genetic algorithm search, respectively. The proximity between these two goals depends on the degree of fitness distance correlation in the problem, with a value close to 1 indicating a strong correlation [37]. In general, for difficult problems, these two goals are very different. Even if the offspring has worse fitness than the parents after crossover, our definition considers it a positive transfer as long as the offspring are closer to the global optima. Eq. (4) focuses on whether the evolution of the population as a whole is correctly approaching the global optima. Thus, positive transfer has the potential to produce offspring with poor fitness, which may be a limitation of this definition. However, the definition in Eq. (4) has the following two key benefits:

- There always exists a linear crossover operator $\mathcal{C} \in \mathcal{K}$ such that Eq. (4) does not hold, i.e., it is always possible to avoid negative transfer by using some linear crossover operator. This property is crucial to the study of how to avoid negative transfer. If the existence of such crossover operators is not even guaranteed, discussions about their design methods will be futile. As shown in Fig. 1, there may not be a linear crossover that avoids negative transfer when we use fitness as a measure of whether negative transfer occurs, but such an embarrassment will not occur under our definition.
- This definition focuses on the correct direction of population evolution rather than changes in population's fitnesses, thereby aiding in the avoidance of individuals getting trapped in local optima. Focusing only on fitness may cause the population to spend a lot of time searching for local optima.

Next, we delve into what kind of linear crossover operator avoids negative transfer. Firstly, let us assume that the number of individuals in the parent population P utilized for crossover

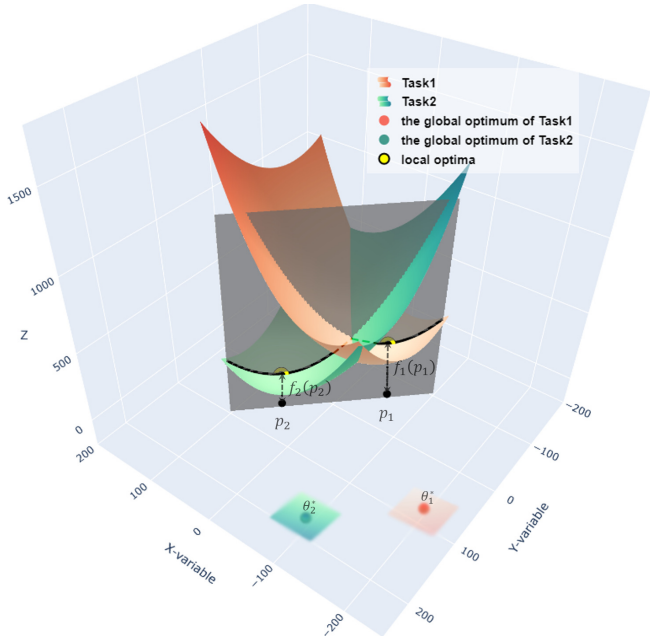


Fig. 1. An example where negative transfer cannot be avoided from the fitness perspective. The images of the two functions $f_1(x, y)$ and $f_2(x, y)$ are represented by the red surface $z = f_1(x, y)$ and the green surface $z = f_2(x, y)$, respectively. θ_1^* and θ_2^* represent the global optimums of the functions, while p_1 and p_2 correspond to their respective local optima. The height of the black curve consists of the fitness values of all linear combinations of p_1 and p_2 on their respective best-performing task. Clearly, the black curve can be divided into two segments, where neither segment has points with heights lower than $f_1(p_1)$ and $f_2(p_2)$, respectively. Thus, linear crossover between p_1 and p_2 must yield offspring with worse fitness.

is less than or equal to n, i.e., $|P| \leq n$. This constraint arises from the observation that crossover typically occurs between individuals in the parent population that excel at different tasks. Consequently, the number of parent individuals involved in crossover should not exceed the number of tasks. Moreover, when $|P| < n$, we can construct a new P' to satisfy $|P'| = n$ by reusing individuals in P for crossover according to the rule: for $i = 1, \ldots, n$, if $\text{dist}(\theta_i^*, P) = \|x - \theta_i^*\|$ for some $x \in P$, then let the i-th element of P' be x. So, it follows that

$$\sum_{i=1}^{n} \text{dist}^2(\theta_i^*, P) = \sum_{i=1}^{n} \text{dist}^2(\theta_i^*, P'). \tag{5}$$

For any $\mathcal{C} \in \mathcal{K}$, let map $\mathcal{C}'(\cdot)$ satisfy $\mathcal{C}'(P') = \mathcal{C}(P) \subset \text{aff}(P) = \text{aff}(P')$, then

$$\sum_{i=1}^{n} \text{dist}^2(\theta_i^*, \mathcal{C}(P)) = \sum_{i=1}^{n} \text{dist}^2(\theta_i^*, \mathcal{C}'(P')). \tag{6}$$

Therefore, we only need to investigate what kind of $\mathcal{C}'$ makes $\sum_{i=1}^{n} \text{dist}^2(\theta_i^*, \mathcal{C}'(P')) \leq \sum_{i=1}^{n} \text{dist}^2(\theta_i^*, P')$ hold, which in turn gives the conditions that $\mathcal{C}$ needs to satisfy. Based on the above analysis, it suffices to consider the case where the parent population size is equal to n. The case $|P| < n$ can be transformed into this case.

Now, consider the typical case $|P| = n$. Define a set $Q(P) \triangleq \{p \in P | \text{dist}(\theta_i^*, P) = \|p - \theta_i^*\| \text{ for some } 1 \leq i \leq n\}$. If $|Q(P)| \neq n$, i.e., $P \setminus Q(P) \neq \emptyset$, we can form a new population P' by retaining only the individuals in $Q(P)$ and

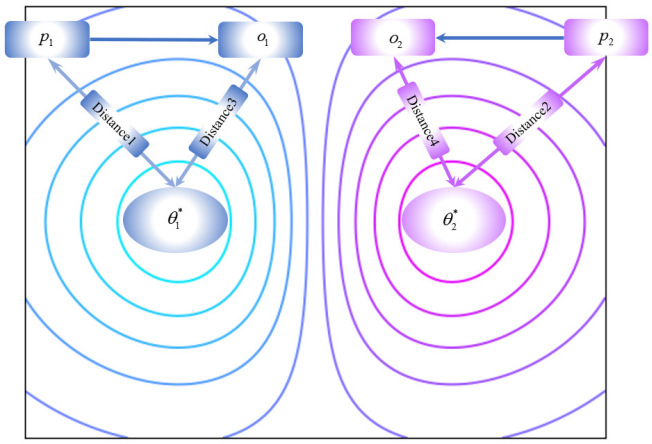


Fig. 2. An example of avoiding negative transfer from the distance perspective. Distances 1-4 denote the distances between the parent and offspring individuals from the global optima, respectively.

making $|P \setminus Q(P)|$ copies of the individuals in $Q(P)$. Clearly, P' satisfies $|Q(P')| = n$ and Eq. (5). Therefore, as long as a linear crossover operator $\mathcal{C}'$ can prevent negative transfer for P', since $\mathcal{C}(P) = \mathcal{C}'(P') \subset \text{aff}(Q(P))$ holds for some linear crossover operator $\mathcal{C}$, then Eq. (6) is satisfied and the operator $\mathcal{C}$ prevent negative transfer for P as well. Thus, we only need to find $\mathcal{C}$ that prevents negative transfer under the assumption $|Q(P)| = n$, and the corresponding linear crossover operator for other cases can be obtained easily .

Under $|Q(P)| = n$, without loss of generality, we can suppose $P = \{p_1, \ldots, p_n\}$, where $\text{dist}(\theta_i^*, P) = \|p_i - \theta_i^*\|$, $i = 1, \ldots, n$. This deduces

$$\sum_{i=1}^{n} \text{dist}^2(\theta_i^*, P) = \sum_{i=1}^{n} \|p_i - \theta_i^*\|^2.$$

Furthermore, we select the some individuals in $\mathcal{C}(P)$ as $o_1, \ldots, o_n$(possibly with duplicates), where each o_i satisfies $\text{dist}(\theta_i^*, \mathcal{C}(P)) = \|o_i - \theta_i^*\|$. Thus, Eq. (4) is equivalent to

$$\sum_{i=1}^{n} \|o_i - \theta_i^*\|^2 > \sum_{i=1}^{n} \|p_i - \theta_i^*\|^2. \tag{7}$$

Fig. 2 illustrates the positions of the parents and offspring. If the offspring are closer to the optimal solutions than the parents, negative transfer does not occur. It is clear that Distance1 > Distance3 and Distance2 > Distance4, therefore the crossover in Fig. 2 is a positive transfer.

Now, by $\mathcal{C} \in \mathcal{K}$, there exists a matrix $\mathcal{A} = \{a_{ij}\} \in \mathbb{R}^{n \times n}$ such that $\mathcal{O} = \mathcal{P}\mathcal{A}$, where $\mathcal{O} = (o_1, \ldots, o_n)$, $\mathcal{P} = (p_1, \ldots, p_n)$. By conducting straightforward calculations, we arrive at the following theorem.

Theorem 1. *Suppose $\mathcal{A} = \mathcal{A}^T$, if $\forall i \neq j$,*

$$\begin{aligned} &\left(\sum_{l=1}^{n} a_{il}a_{jl} - a_{ji}\right) \|p_i - p_j\|^2 \\ \geq\ & 2a_{ji}\left(\theta_i^* - \theta_j^*\right) \cdot (p_i - p_j), \end{aligned} \tag{8}$$

then

$$\sum_{i=1}^{n} \|o_i - \theta_i^*\|^2 \leq \sum_{i=1}^{n} \|p_i - \theta_i^*\|^2.$$

Conversely, if $\forall i \neq j$, (8) does not hold, then

$$\sum_{i=1}^{n} \|o_i - \theta_i^*\|^2 > \sum_{i=1}^{n} \|p_i - \theta_i^*\|^2.$$

Proof. The proof is presented in Appendix A. □

Remark 1. *Denote*

$$b_{ij} = \frac{(\theta_i^* - \theta_j^*) \cdot (p_i - p_j)}{\|p_i - p_j\|^2}, \quad 1 \leq i < j \leq n,$$

where $b_{i,j}$ can be interpreted as a similarity measure capturing both the length and direction between vectors $p_i - p_j$ and $\theta_i^* - \theta_j^*$. Thus, Eq. (8) can be rewritten as

$$\sum_{l=1}^{n} a_{il}a_{jl} \geq (1 + 2b_{ij})a_{ji}. \tag{9}$$

Theorem 1 implies that negative transfer can be avoided if the strength of interaction between the two individuals and the rest of the population, relative to the strength of interaction between the two individuals, is greater than 1 plus twice their similarity to the global optimum in terms of relative position.

Remark 2. *We consider the scenario where $p_i - p_j \approx \theta_i^* - \theta_j^*$, which often arises in the later phases of algorithm execution, then $b_{ij} \approx 1$. According to Theorem 1 and Eq. (9), the condition $\sum_{l=1}^{n} a_{il}a_{jl} \geq 3a_{ji}$ for all $i \neq j$ is required to avoid negative transfer. It is notable that, this condition generally implies that there is some $a_{ij} \leq 0$. For instance, when $n = 2$, since $\{i, j\} = \{1, 2\}$ and $a_{11} = a_{22} = 1 - a_{12} = 1 - a_{21}$, it is straightforward to observe that $\sum_{l=1}^{n} a_{il}a_{jl} = a_{i1}a_{j1} + a_{i2}a_{j2} = 2a_{12}(1 - a_{12})$, which together with $\sum_{l=1}^{n} a_{il}a_{jl} \geq 3a_{12}$ deduces $a_{12} \leq 0$. A non-positive crossover coefficient between two parents indicates the presence of a symmetric mapping operation within the crossover operator. Similar ideas have appeared in methods such as opposition-based learning [38]–[40], but it is generally believed that symmetric mapping is used to expand the population search range. However, in this study, we propose a novel explanation from the perspective of avoiding negative transfer.*

Theorem 2. *Assume that $\sum_{l=1}^{n} a_{il}a_{jl} > a_{ji}$ for any $i \neq j$. Let A_{ij} be the event that Eq. (8) is true, $i \neq j$. If the d components of p_i are all independently generated by a Gaussian distribution $N(0, 1)$, and $\theta_i^* \in [0, 1]^d$, $i = 1, \ldots, n$, then we have*

$$P\left(\cap_{i<j} A_{ij}\right) > 1 - 2n^2 e^{-d\delta}, \tag{10}$$

where $\delta > 0$ is a constant depends on $\mathcal{A}$.

Proof. The proof is contained in Appendix B. □

Remark 3. *Since $d \gg \log n$ usually holds, Theorem 2 reveals an intriguing observation: if matrix $\mathcal{A}$ satisfies $\min_{i<j}(\sum_{l=1}^{n} a_{il}a_{jl} - a_{ji}) > 0$, negative transfer occurs with a very small probability for randomly generated parents. This phenomenon can be attributed to the almost orthogonality of independent random vectors in high-dimensional spaces. The orthogonality between the high-dimensional vectors $p_i - p_j$ and $\theta_i^* - \theta_j^*$ ensures the fulfillment of Eq. (8),*

which distinguishes high-dimensional scenarios from their low-dimensional counterparts. While conventional wisdom suggests that evolutionary algorithms struggle to find optimal solutions in high-dimensional spaces, Theorem 2 surprisingly demonstrates that negative transfers are likely to be rare in such settings. This observation proves beneficial for addressing MTO problems from the perspective of mitigating negative transfer. Furthermore, the setting of normal distribution in Theorem 2 can be extended to sub-Gaussian distributions, encompassing a broader range of distributions, such as the uniform distribution. The proof process remains almost identical, and in this paper, we adopt the normal distribution just to simplify the description of the theorem.

Since the analysis presented above, during the later phases of algorithm execution, when the parents have already explored the vicinity of the optimal solutions, we can design the crossover operator based on $\min_{i<j}(\sum_{l=1}^{n} a_{il}a_{jl} - 3a_{ji}) > 0$ to avoid negative transfer, making it necessary to incorporate mechanisms such as symmetric mapping. On the other hand, in the early phases of algorithm execution, there is typically a substantial disparity between the parents and the optimal solutions, resembling a randomly generated population. In such cases, it is possible to devise a suitable crossover operator based on $\min_{i<j}(\sum_{l=1}^{n} a_{il}a_{jl} - a_{ji}) > 0$ to ensure that negative transfer is unlikely to occur.

B. The Optimal Crossover Operator

We now turn our attention to the design problem of the optimal crossover operator. In the last subsection, we propose a favorable approach to effectively overcome negative transfer. However, it does not guarantee the optimality of knowledge transfer. For the problem defined in Eq. (1), similar to the analytical framework presented in Section III-A, we simplify it to the following constrained optimization problem:

$$\min_{\mathcal{A}} \sum_{i=1}^{n} \|o_i - \theta_i^*\|^2 \quad \text{subject to} \quad \mathbf{1^T}\mathcal{A} = \mathbf{1^T}, \tag{11}$$

where the constraint $\mathbf{1^T}\mathcal{A} = \mathbf{1^T}$ is due to our restriction that the linear crossover operator is an affine combination of individuals in P. Obtaining a solution with a closed form for this problem is generally not straightforward. In this study, we tackle the problem in a typical setting. We assume that crossover occurs between a pair of parents, generating two children symmetrically. Then, $n = 2$, and we restrict

$$\mathcal{A} = \begin{pmatrix} 1-b & b \\ b & 1-b \end{pmatrix}. \tag{12}$$

By doing so, we immediately obtain the following equality from Theorem 1:

$$\begin{aligned} & \sum_{i=1}^{n} \|p_i - \theta_i^*\|^2 - \sum_{i=1}^{n} \|o_i - \theta_i^*\|^2 \\ = \;& \left(b - 2b^2\right) \|p_1 - p_2\|^2 - 2b\left(\theta_1^* - \theta_2^*\right)^T \cdot (p_1 - p_2) \\ \triangleq \;& g(b). \end{aligned}$$

Next, we can maximize $g(b)$ by setting the derivative of $g(b)$ equal to zero and solving for b.

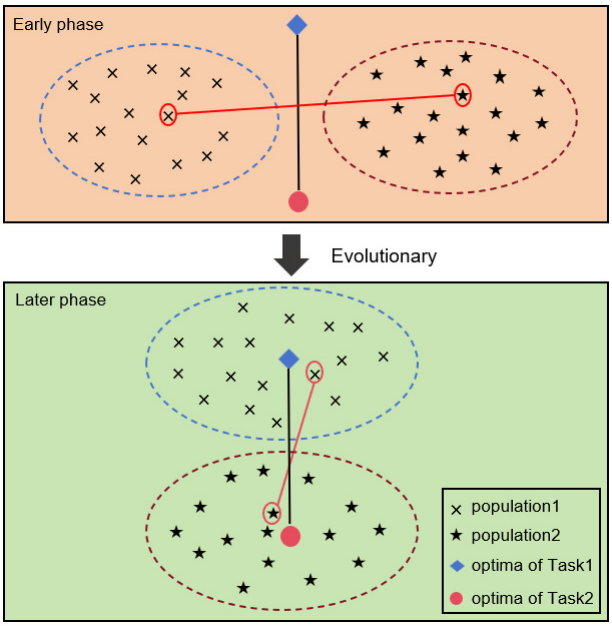


Fig. 3. Changes in the angle between $\theta_1^* - \theta_2^*$ and $p_1 - p_2$ in the algorithmic early phase and later phase.

Theorem 3. *Suppose $\mathcal{A}$ satisfies Eq.* (12)*, then the optimal value of b is*

$$b^* = \frac{1}{4} - \frac{1}{2}\frac{\left(\theta_1^* - \theta_2^*\right)^T \cdot (p_1 - p_2)}{\|p_1 - p_2\|^2}, \tag{13}$$

and

$$g(b^*) = 2\left(\frac{\|p_1 - p_2\|}{4} - \frac{\left(\theta_1^* - \theta_2^*\right)^T \cdot (p_1 - p_2)}{2\|p_1 - p_2\|}\right)^2.$$

Remark 4. *According to Theorem 3, if we denote the angle between $\theta_1^* - \theta_2^*$ and $p_1 - p_2$ as ψ, then*

$$b^* = \frac{1}{4} - \frac{1}{2}\frac{\|\theta_1^* - \theta_2^*\| \cdot \cos\psi}{\|p_1 - p_2\|}. \tag{14}$$

From the previous analysis, we know that $\psi \approx \pi/2$ during the early phase of population evolution, while $\psi \approx 0$ when the parents have already explored the vicinity of the optimal solutions during the later phase, see Fig. 3. Therefore, in these two cases, we have

$$b^* \approx \frac{1}{4} \quad \text{and} \quad b^* \approx \frac{1}{4} - \frac{1}{2}\frac{\|\theta_1^* - \theta_2^*\|}{\|p_1 - p_2\|},$$

respectively. Of course, we point out that in the whole process of population evolution, it is not limited to the above two cases, and it is possible for ψ to take any value in $[0, \pi]$. We just heuristically choose two typical cases to facilitate the understanding of Eq. (14)*.*

While Eq. (14) provides a specific formula for the optimal crossover operator, the unknown nature of $\theta_1^* - \theta_2^*$ poses a challenge in practical design. To address this issue, we can start by estimating $\|\theta_1^* - \theta_2^*\|$, denoted as r.

During the early phase, since $\psi \approx \pi/2$, we generate a random number v_1 from a normal distribution $N(\frac{\pi}{2}, \frac{1}{d})$, and set

$$\widehat{b}^* = \frac{1}{4} - \frac{1}{2}\frac{r\cos v_1}{\|p_1 - p_2\|}. \tag{15}$$

This choice is based on Lemma 1 defined in the Supplementary Materials, which states that for nonzero vector $\theta_1^* - \theta_2^*$, if n random vectors are generated in $\mathbb{R}^d$ with a spherical distribution, the empirical distribution of the angle formed by their pairwise differences with respect to vector $\theta_1^* - \theta_2^*$, tends to the normal distribution $N(\frac{\pi}{2}, \frac{1}{d})$ as d and n tend to infinity. Since in the early phase, the parent individuals can be approximated as independently and identically generated, and the spherical distribution contains many classical distributions (e.g., normal, t-distribution), the empirical distribution of ψ in Eq. (14) can be modelled by $N(\frac{\pi}{2}, \frac{1}{d})$ when the population size and the dimensionality d are both very large. As a result, the optimal crossover operator Eq. (14) has almost the same statistical properties as Eq. (15).

On the other hand, consider the case that the algorithm is not in the early phase. Based on the analysis above (7), it follows that $\|p_1 - \theta_1^*\| \leq \|p_2 - \theta_1^*\|$, $\|p_2 - \theta_2^*\| \leq \|p_1 - \theta_2^*\|$, and then

$$\begin{cases} \|p_1\|^2 - \|p_2\|^2 \leq 2(\theta_1^*)^T \cdot (p_1 - p_2) \\ \|p_2\|^2 - \|p_1\|^2 \leq 2(\theta_2^*)^T \cdot (p_2 - p_1) \end{cases},$$

which implies $(\theta_2^* - \theta_1^*)^T \cdot (p_2 - p_1) \geq 0$, then $\cos\psi \geq 0$. Besides knowing that $\cos\psi \in [0, 1]$, we lack information about the specific distribution of ψ. At this point, $\cos\psi$ is equally likely to take any value in the interval $[0, 1]$. Then, we can generate a number v_2 from a uniform distribution $U(0, 1)$ and define

$$\widehat{b}^* = \frac{1}{4} - \frac{1}{2}\frac{rv_2}{\|p_1 - p_2\|}, \tag{16}$$

as an alternative formula for Eq. (14). It should be clarified that the above design based on the uniform distribution is only a conservative choice when we do not know the distribution of $\cos\psi$. It is possible to replace the uniform distribution with a better distribution if partial information about the distribution of $\cos\psi$ can be obtained from the historical information of the population through more detailed analysis. This part is to be followed up to continue in-depth research.

IV. Proposed Method

A. Overall Framework

Building upon the theoretical analysis in the preceding section, we propose the MFEA-OC based on the introduction of new crossover operators. The pseudocode for MFEA-OC is presented in Algorithm 2, where the function $OC()$ denotes the approximate design of the optimal crossover operator (Algorithms 3). The main distinction between the proposed MFEA-OC and the conventional MFEA lies in the introduction of the new crossover operator. As depicted in Algorithm 2, the workflow of MFEA-OC can be summarized as follows.

1) Randomly initialize the population and assign each individual a skill factor.
2) In each evolutionary generation, an adaptive mechanism (Algorithm 3) is employed to determine the phase of optimization: early phase ($s = 0$) or later phase ($s = 1$). This determination is crucial for selecting the appropriate optimal crossover operator. The adaptive mechanism is based on the variation in information entropy, specifically the pairwise angles between individuals in the population. Therefore, the information entropy is recalculated at the start of each generation.
3) Next, parent individuals p_1 and p_2 are randomly selected from the population P. Only when their skill factors differ, a probability rmp is assigned to employ the approximate strategy $OC()$ of the optimal crossover operator for offspring generation (steps 7–10). In all other scenarios, the conventional SBX crossover operator is directly applied. This choice is motivated by the theoretical analysis of the optimal crossover operator described in the previous section, which assumes that parent individuals should excel in different tasks. After generating o_1 and o_2 through the crossover operator, each offspring undergoes an independent polynomial mutation (PM) to obtain the final offspring.
4) Following the generation and evaluation of the offspring population, the next-generation population is formed using the elite-based environmental selection operator.

Algorithm 2: The Pseudo-code of MFEA-OC

Input: N (population size), rmp (random mating probability)
Output: A series of solutions

1 Randomly initialize population P and assign skill factor τ for every individual;
2 **while** *the stop condition is not satisfied* **do**
3 Calculate the information entropy of the pairwise angles between individuals in P;
4 Use Algorithm 4 to obtain a number $s \in \{0, 1\}$;
5 **for** $i = 1, \ldots, N/2$ **do**
6 Randomly select two parent individuals p_1 and p_2;
7 **if** $\tau_1 \neq \tau_2$ *and rand*$<$ rmp **then**
8 $\widehat{b} \leftarrow OC(P, s, p_1, p_2)$;
9 $o_1 \leftarrow (1 - \widehat{b}) \cdot p_1 + \widehat{b} \cdot p_2$;
10 $o_2 \leftarrow \widehat{b} \cdot p_1 + (1 - \widehat{b}) \cdot p_2$;
11 **else**
12 Generate o_1 and o_2 by SBX;
13 $o_1 \leftarrow$ local variation (mutation) of o_1;
14 $o_2 \leftarrow$ local variation (mutation) of o_2;
15 Each offspring is randomly assigned skill factor τ_1 or τ_2;
16 Evaluate offspring population O and generate new population $NP = P \cup O$;
17 Selection the fittest population P from NP;

Algorithm 2 uses the information entropy of pairwise angles between individuals to determine the algorithmic phase. Actually, simpler criteria can also be used for phase determination. Furthermore, the rmp plays a non-essential role in Algorithm 2. Its inclusion is based on the MFEA framework, allowing us to emphasize the significance of strategy $OC()$ by substituting the conventional crossover operator utilized in classical algorithms. We retain the SBX operator in MFEA, as stated in line 12 of Algorithm 2, when $\tau_1 = \tau_2$. This decision stems from the theoretical derivation of the optimal crossover operator presented in Section III, which assumes that the skill factors of the parent individuals are not equal. Otherwise, we would be unable to transform the design of crossover operators into an optimization problem as described in Theorem 3, nor could we apply the approximation analysis of Eqs. (15) and (16) outlined in Section IV-B. In the following sections, we will provide detailed explanations of the modules in Algorithm 2 and their fundamental principles.

B. Optimal Crossover Strategy

We propose a method for designing the optimal operator based on formulas (15) and (16), as presented in Algorithm 3. By the analysis in Section III-B, Eqs. (15)–(16) necessitate an estimate for $\|\theta_1^* - \theta_2^*\|$. To address this, we select individuals $\bar{p}_1$ and $\bar{p}_2$, the most proficient in tasks τ_1 and τ_2 within the population P, respectively, and calculate $\|\bar{p}_1 - \bar{p}_2\|$ to estimate $\|\theta_1^* - \theta_2^*\|$. The rationale behind this selection is explained as follows:

- During the later phase of algorithm execution, when $\bar{p}_1$ and $\bar{p}_2$ are both close to the global optimums, $\|\theta_1^* - \theta_2^*\|$ is approximately equal to $\|\bar{p}_1 - \bar{p}_2\|$. When $\bar{p}_1$ and $\bar{p}_2$ are in the vicinity of some local optima, since they are the best performing individuals in tasks τ_1 and τ_2, respectively, we can assume that they are close to some specific high-quality local optimal solutions q_1 and q_2, respectively. At this point, if tasks τ_1 and τ_2 have certain similarities, such as a similar relative position between the global optimum and the high-quality local optima, i.e., $\theta_1^* - q_1 \approx \theta_2^* - q_2$, then we obtain $\|\bar{p}_1 - \bar{p}_2\| \approx \|q_1 - q_2\| \approx \|\theta_1^* - \theta_2^*\|$. From the above analysis, it is clear that the key in determining whether $\|\bar{p}_1 - \bar{p}_2\|$ is a good estimate of $\|\theta_1^* - \theta_2^*\|$ is not whether $\bar{p}_1$ and $\bar{p}_2$ are close to the global optimums, but rather how similar the landscapes of the two tasks are.
- Unlike the later phase, in the early phase, perhaps $\|\bar{p}_1 - \bar{p}_2\|$ is not a valid estimate of $\|\theta_1^* - \theta_2^*\|$ due to the randomness of the parents. Fortunately, we can show that there is almost no difference between b^* and $\widehat{b}^*$ in Eq. (15) when d is large. Actually,

$$|\widehat{b}^* - b^*| \le \frac{\|\theta_1^* - \theta_2^*\| \cdot |\cos\psi| + \|\bar{p}_1 - \bar{p}_2\| \cdot |\cos v_1|}{2\|p_1 - p_2\|} \le \frac{\|\theta_1^* - \theta_2^*\| \cdot |\frac{\pi}{2} - \psi| + \mathrm{diam}(P) \cdot |\frac{\pi}{2} - v_1|}{2\|p_1 - p_2\|}$$

where $\mathrm{diam}(P) \triangleq \sup_{p',p'' \in P} \|p' - p''\|$. Based on the analysis in Section III-B, in the early phase it can be approximated that ψ in Eq. (14) satisfies the distribution $N(\frac{\pi}{2}, \frac{1}{d})$, then $|\frac{\pi}{2} - \psi| = O_p(d^{-1/2})$. The notation $a_d = O_p(b_d)$ means that the set of values a_d/b_d is stochastically bounded. Likewise, we have $|\frac{\pi}{2} - v_1| = O_p(d^{-1/2})$. Moreover, by Eq. (19) in Appendix B, it follows that $\|p_1 - p_1\| \approx (2d)^{1/2}$ and $\mathrm{diam}(P) = O_p(d^{1/2})$ as $d \gg \log^{1/2} n$. So, we obtain

$$|\widehat{b}^* - b^*| = O_p(d^{-1/2}) \quad \text{as } d \gg \log^{1/2} n. \tag{17}$$

Thus, even if $\|\bar{p}_1 - \bar{p}_2\| \not\approx \|\theta_1^* - \theta_2^*\|$, according to Eq. (17), we still have $\widehat{b}^* \approx b^*$ when d is large. Therefore, during the early phase, the choice of $r = \|\bar{p}_1 - \bar{p}_2\|$ for Eq. (15) does not affect the validity of $\widehat{b}^*$. Regardless of whether $\|\bar{p}_1 - \bar{p}_2\|$ is approximately equal to $\|\theta_1^* - \theta_2^*\|$ or not, we can get an approximate design of the optimal crossover operator.

Algorithm 3: Optimal Crossover Strategy

Input: Individuals p_1 and p_2, a number $s \in \{0, 1\}$, the population P
Output: A crossover coefficient b

1 Select individuals $\bar{p}_1$ and $\bar{p}_2$ from P who excel most in tasks τ_1 and τ_2, respectively;
2 **if** $s = 0$ **then**
3 Generate a number v using the normal distribution $N(\frac{\pi}{2}, \frac{1}{d})$;
4 $b \leftarrow \frac{1}{4} - \frac{1}{2}\frac{\cos v \|\bar{p}_1 - \bar{p}_2\|}{\|p_1 - p_2\|}$;
5 **else**
6 Generate a number v using the uniform distribution $U(0, 1)$;
7 $b \leftarrow \frac{1}{4} - \frac{1}{2}\frac{v\|\bar{p}_1 - \bar{p}_2\|}{\|p_1 - p_2\|}$;

C. Adaptive strategy for phase determination

We propose a phase determination algorithm based on monitoring the changes in the information entropy of the pairwise angles among individuals in the population(see Algorithm 4). As discussed in Section III-B, during the early phase, the empirical distribution of these angles approximately follows a normal distribution. However, as the population explores local optima, the level of population diversity gradually decreases, resulting in a corresponding decrease in the information entropy of these angles. If the population becomes trapped in local optima, the information entropy of these angles will plateau. Therefore, a natural approach is to consider a decrease in information entropy as an indication that the population has not yet reached a local optimum, signifying the early phase of optimization. Conversely, if the information entropy does not decrease, it suggests that a significant number of individuals in the population have converged to local optima, marking the later phase of optimization. In Algorithm 4, the parameter m, which is an adjustable integer, controls the switching frequency of the adaptive strategy. A larger value of m leads to a lower switching frequency.

V. Experiments

In the preceding sections, we have elucidated the underlying principle and practical design method of the optimal crossover

Algorithm 4: Entropy-based Adaptive Strategy

Input: m information entropies are computed based on the most recent m generations of populations, respectively
Output: A number s

```
if generation < m then
    s = 0;
else
    Let e_c and e_p denote the information entropy of the present population and the average entropy of the past m − 1 generations, respectively ;
    if e_c < e_p then
        s=0;
    else
        s=1;
```

operator. This section experimentally validates our claims through the following aspects:

1) Demonstrating the highly competitive performance of the proposed algorithm, MFEA-OC.
2) Showing the superiority of the proposed crossover operator design method over various alternative designs.
3) Validation of the rationale for distinguishing between early and late phases in the design of optimal crossover operators.
4) Assessing the effectiveness of the adaptive strategy implemented in Algorithm 4.

A. Experimental Settings

Following the outlined plan, we conducted targeted experiments to validate our approach. First, we selected several algorithms for comparison, including the initially proposed algorithm MFEA [8] and its subsequent version, MFEA-II [16], which incorporates the adaptive rmp parameter. These two algorithms hold prominent positions in the field of MTO. In addition, we considered the adaptive knowledge transfer algorithm MFEA-AKT [27] and the algorithm MFEA-GHS [40], which is based on genetic transformation and hyper-rectangle search. Furthermore, we included the algorithm SREMTO [41], which incorporates adaptive control of knowledge transfer intensity, and AT-MFEA [25], which mitigates confusion arising from different task time mappings through affine mapping. Through comprehensive comparisons with these algorithms, we rigorously validated the effectiveness of the proposed approach.

To ensure a fair comparison, a grid search methodology was employed to determine the optimal parameters for the algorithms across three distinct test suites. The search space included 45 different parameter combinations, emphasizing the common parameters among the algorithms. For more detailed information regarding the parameter tuning experiment, please refer to the Section II of the Supplementary Materials. Table I summarizes the optimal parameter configurations determined for the algorithms based on the conducted experiments. For the MFEA-AKT, several specific parameters are used. The parameter for Arithmetical Crossover is $\lambda = 0.25$. Similarly, the Geometrical Crossover parameter ω is set to 0.25. Additionally, the BLX-α Crossover utilizes an α value of 0.3. The population size for each task is 100. The maximum number of function evaluations (FEs) depends on the test suite we select and the comparison experiments, which we will describe separately later.

B. Test Problems

Two suites of test problems were used in the experiments: WCCI2020-MTSO and WCCI2024-MTSO. The first suite includes 10 MTO complex problems extracted from the test suite used in the WCCI2020 Competition on MTO[1]. Each problem in this suite consists of two distinct single-objective optimization tasks. WCCI2020-MTSO represents a more challenging set of problems compared to previous benchmark suites, aimed at investigating the efficacy of algorithms in handling intricate multitasking scenarios. Similar to the first suite, the second suite focuses on a set of 2-task benchmark problems designed in WCCI2024 Competition on MTO[2]

To further validate the performance of our proposed method, we had chosen a test suite from a real-world application: parameter extraction of photovoltaic (PV) models [42]. Parameter extraction is crucial for accurately simulating and optimizing the performance of PV systems. It involves determining the electrical parameters that best describe the behavior of a solar cell or module. Accurate parameter extraction is crucial for improving the efficiency of PV systems, diagnosing faults, and optimizing system design and operation. The commonly used PV models include the single-diode model (SDM), the double-diode model (DDM), and the single-diode model based on PV-module (SMM). Each of these models provides different levels of complexity and accuracy in representing the PV cell characteristics. In [42], the parameter extraction of these three PV models is considered a multi-task optimization problem. In this paper, we use these three PV models to construct a 3-task test problem.

C. Results and Discussion

The comparative experimental results in this section were obtained using Friedman's test at a significance level of 0.05. In addition, as an intermediate output of the Friedman test, the average ranking of each comparison algorithm is presented as $meanrank$. For the two benchmark test suites and the real-world application test suite, we employed the Wilcoxon signed-rank test at the significant level of 95% to compare experimental outcomes. To handle the high error rate, we used the Bonferroni correction to adjust the p-values. The test results were categorized into win/lose/tie denoted as '$+/-/=$'. The best performance values are shown in bold. All experiments were tested on the MToP platform [43], a detailed description of which can be found at https://github.com/intLyc/MTO-Platform.

[1]http://www.bdsc.site/websites/MTO_competition_2020/MTO_Competition_WCCI_2020.html

[2]https://www.hou-yq.com/WCCI_CEC_2024_COMPETITION_EMO.html

TABLE I
THE PARAMETERS USED IN EACH ALGORITHM.

	MFEA	MFEA-II	MFEA-AKT	MFEA-GHS	SREMTO	AT-MFEA	MFEA-OC
SBX crossover probability	1	1	1	1	1	1	1
distribution index (SBX)	2-2-2[1]	10-10-10	2-2-10	2-2-10	2-15-2	10-10-15	15-15-15
PM probability	1/d	1/d	1/d	1/d	1	1/d	1/d
distribution index (PM)	40-30-5	40-40-40	40-40-15	20-40-5	20-30-20	40-40-40	40-40-40
rmp	0.3-0.7-0.7	learned online	0.7-0.7-0.3	0.5-0.7-0.5	0.5-0.7-0.7	0.5-0.7-0.7	0.5-0.5-0.7

[1] 2-2-2 indicates the optimal parameters for test suite 1, test suite 2, and test suite 3, respectively.

1) Performance of MFEA-OC: To comprehensively evaluate the performance of MFEA-OC, we perform comparisons with several state-of-the-art multitask algorithms using three test suites introduced in Section V-B. The first two test suites are configured with a maximum of 200000 FEs, while the third test suite is configured with a maximum of 300000 FEs. The results in terms of the mean and standard deviation of the best-achieved FEs over 20 runs for each component task in each MTO problem from the two test suite are presented in Tables II and III. The results for the test suite 3 are presented in Table IV.

As illustrated in Tables II and III, the proposed MFEA-OC exhibits remarkable performance on the two continuous MTO benchmarks in terms of the average objective value. Unlike the MFEA, which encounters inevitable negative transfer because of random knowledge transfer, the proposed MFEA-OC incorporates an optimal crossover operator to mitigate negative transfer and enhance positive transfer, outperforming or matching the MFEA on 17 of 20 tasks in both two test suites. Although MFEA-II optimizes the knowledge transfer probability between tasks, its ability to overcome negative transfer remains inferior to that of MFEA-OC. For 40 tasks in the two test suites, MFEA-II outperforms MFEA-OC on only 5 tasks. Moreover, the proposed method outperforms MFEA-AKT in 9 of 20 tasks and 10 of 20 tasks in the two test suites, respectively. This indicates that although MFEA-AKT can dynamically select appropriate crossover operators from SBX, arithmetic, geometric, and BLX-α, the optimally designed crossover operator in MFEA-OC facilitates a more effective search for global optima. The remaining three comparison algorithms, namely MFEA-GHS, SREMTO, and AT-MFEA, employ diverse mechanisms to improve knowledge transfer efficiency among tasks. However, they consistently exhibit inferior performance compared with the MFEA-OC algorithm in most of the tasks. Overall, MFEA-OC demonstrated superior performance among all the comparison algorithms and consistently achieved the lowest mean rank. we provide the average convergence trajectories for all problems in test suites 1 and 2. The convergence trajectories for test suites 1 and 2 can be found in the Supplementary Materials. The x-axis represents the number of FEs, and the y-axis represents the average target value on a logarithmic scale.

Although MFEA-OC performs best overall on test suites 1 and 2, it can be observed from Tables II and III that for problem MTSO8 in both test suites, MFEA-OC performs the worst among all algorithms. One possible explanation is that the function of MTSO8 in both WCCI20 and WCCI24 are generated by the Ackley function. The characteristic of this function is that, except for the global optimum, the difference between the function values of most local optima is very small, and the local optima are almost uniformly distributed in the search space. Therefore, for q_1 and q_2 in the approximate analysis of OC in Section IV-B, they are likely to randomly fall into positions of uniformly distributed local optima, leading to significant differences in the directions of vectors $\theta_1^* - q_1$ and $\theta_2^* - q_2$. As a result, $\theta_1^* - q_1 \not\approx \theta_2^* - q_2$. This hinders the effectiveness of the approximation process and prevents us from accurately approaching the true OC.

Table IV presents the averaged standard objective values of different compared methods on the test suite 3. For task 1 (PEPVM-T1), MFEA-OC demonstrates the best performance, significantly outperforming the other methods. For the PEPVM-T2, MFEA-OC again outperforms all other methods, which exhibit a range of performances with none coming close to it. MFEA-OC and AT-MFEA are comparable, with both methods showing superior performance relative to the others for the final task. The *meanrank* obtained via Friedman's test shows that MFEA-OC has the best overall performance with a rank of 1.33. This comprehensive analysis highlights our approach as the most effective method among the seven evaluated for the given real-world application tasks.

2) The superiority of the proposed crossover operator: According to our previous experimental results, the MFEA-OC algorithm demonstrated outstanding performance on three distinct test suites. To further assess the capabilities of the optimal crossover operator, we conducted comparative experiments to investigate the parameters associated with the designed crossover operator.

The new crossover operator employed in this study, which involves selecting a parameter b in Eq. (12) that satisfies Eqs. (15) and (16), is derived mathematically and involves several approximation steps. This endeavor raised several pertinent questions: Does the specific design formula employed in our optimal crossover operator genuinely outperform randomly selecting b or utilizing a fixed value of b? In addition, what would be the effect of using only the SBX operator? To address these inquiries, we have replaced the section on the optimal crossover operator design in MFEA-OC with three alternative designs for comparative analysis: using a fixed value of b, a randomly generated b, and only the SBX operator. For the first design, we set $b = 0.25$, as previous analyses indicated that the optimal crossover operator would theoretically possess a

TABLE II
THE AVERAGED STANDARD OBJECTIVE VALUE OF SEVEN COMPARED METHODS, OVER 20 INDEPENDENT RUNS ON THE SINGLE-OBJECTIVE MTO TEST SUITE 1. THE *meanrank* IS OBTAINED VIA FRIEDMAN'S TEST.

	MFEA	MFEA-II	MFEA-AKT	MFEA-GHS	SREMTO	AT-MFEA	MFEA-OC
WCCI20-MTSO1-T1	6.4663e+02 (8.20e+00) -	6.1210e+02 (3.09e+00) -	6.1386e+02 (2.27e+00) -	6.4114e+02 (6.89e+00) -	6.5822e+02 (4.90e+00) -	**6.0265e+02 (1.63e+00) +**	6.0663e+02 (2.20e+00)
WCCI20-MTSO1-T2	6.4421e+02 (7.09e+00) -	6.1252e+02 (2.78e+00) -	6.1531e+02 (2.56e+00) -	6.4421e+02 (6.66e+00) -	6.5602e+02 (4.34e+00) -	**6.0244e+02 (1.08e+00) +**	6.0713e+02 (1.51e+00)
WCCI20-MTSO2-T1	7.0001e+02 (5.65e-03) +	7.0006e+02 (1.79e-02) +	**7.0000e+02 (5.70e-03) +**	7.0009e+02 (5.71e-02) =	7.0097e+02 (2.80e-02) -	7.0002e+02 (7.45e-03) +	7.0011e+02 (2.35e-02)
WCCI20-MTSO2-T2	**7.0000e+02 (5.08e-03) +**	7.0006e+02 (1.60e-02) +	7.0001e+02 (4.59e-02) +	7.0009e+02 (5.64e-02) =	7.0097e+02 (4.10e-02) -	7.0002e+02 (6.18e-03) +	7.0011e+02 (2.40e-02)
WCCI20-MTSO3-T1	1.4530e+06 (7.29e+05) -	1.4111e+06 (8.31e+05) -	2.0348e+05 (1.64e+05) =	2.6480e+06 (1.32e+06) -	1.7057e+06 (9.90e+05) -	1.2928e+06 (6.32e+05) -	**1.9473e+05 (1.17e+05)**
WCCI20-MTSO3-T2	1.5284e+06 (1.12e+06) -	1.1959e+06 (5.34e+05) -	2.3500e+05 (1.73e+05) =	2.6244e+06 (1.39e+06) -	1.5822e+06 (8.50e+05) -	1.2970e+06 (5.46e+05) -	**2.0663e+05 (1.05e+05)**
WCCI20-MTSO4-T1	1.3005e+03 (9.67e-02) -	1.3004e+03 (7.21e-02) =	1.3005e+03 (5.86e-02) -	1.3005e+03 (1.25e-01) -	1.3006e+03 (8.48e-02) -	1.3004e+03 (4.87e-02) =	**1.3004e+03 (6.17e-02)**
WCCI20-MTSO4-T2	1.3005e+03 (7.00e-02) -	1.3004e+03 (5.79e-02) =	1.3004e+03 (7.13e-02) =	1.3005e+03 (5.71e-02) -	1.3005e+03 (6.73e-02) -	1.3004e+03 (4.41e-02) =	**1.3003e+03 (7.97e-02)**
WCCI20-MTSO5-T1	1.5350e+03 (8.09e+00) -	1.5122e+03 (4.37e+00) =	1.5251e+03 (6.59e+00) -	1.5312e+03 (7.23e+00) -	1.5592e+03 (9.94e+00) -	1.5202e+03 (9.11e+00) =	**1.5111e+03 (2.90e+00)**
WCCI20-MTSO5-T2	1.5285e+03 (6.24e+00) -	**1.5151e+03 (8.06e+00) =**	1.5217e+03 (5.23e+00) -	1.5261e+03 (5.37e+00) -	1.5564e+03 (1.08e+01) -	1.5208e+03 (9.92e+00) =	1.5151e+03 (5.94e+00)
WCCI20-MTSO6-T1	8.6671e+05 (5.39e+05) =	8.5365e+05 (5.22e+05) =	9.1132e+05 (6.42e+05) =	1.1134e+06 (4.24e+05) -	1.6103e+06 (9.19e+05) -	8.8495e+05 (7.49e+05) =	**6.8931e+05 (3.25e+05)**
WCCI20-MTSO6-T2	6.6753e+05 (2.79e+05) =	6.4872e+05 (4.00e+05) =	7.0815e+05 (4.68e+05) =	1.4148e+06 (9.37e+05) -	7.5569e+05 (3.53e+05) =	**4.3971e+05 (2.02e+05) =**	5.4306e+05 (4.13e+05)
WCCI20-MTSO7-T1	3.6214e+03 (4.66e+02) -	3.0684e+03 (4.45e+02) -	2.8982e+03 (2.52e+02) -	3.3398e+03 (4.47e+02) -	3.6106e+03 (3.01e+02) -	2.9967e+03 (4.46e+02) -	**2.4054e+03 (2.05e+02)**
WCCI20-MTSO7-T2	3.3906e+03 (3.40e+02) -	3.1705e+03 (3.91e+02) -	3.0435e+03 (2.75e+02) -	3.3219e+03 (3.63e+02) -	3.5879e+03 (4.39e+02) -	2.8837e+03 (4.30e+02) -	**2.5285e+03 (2.13e+02)**
WCCI20-MTSO8-T1	**5.2002e+02 (1.98e-02) +**	5.2117e+02 (4.38e-02) =	5.2013e+02 (6.47e-02) +	5.2007e+02 (8.13e-02) +	5.2007e+02 (1.75e-02) +	5.2118e+02 (4.69e-02) =	5.2118e+02 (3.38e-02)
WCCI20-MTSO8-T2	**5.2002e+02 (5.05e-02) +**	5.2118e+02 (5.89e-02) =	5.2011e+02 (5.16e-02) +	5.2017e+02 (1.29e-01) +	5.2008e+02 (3.28e-02) +	5.2118e+02 (3.44e-02) =	5.2116e+02 (3.57e-02)
WCCI20-MTSO9-T1	8.1762e+03 (7.29e+02) =	1.2250e+04 (3.25e+03) =	8.0611e+03 (6.18e+02) =	**7.9639e+03 (1.08e+03) =**	7.9924e+03 (6.02e+02) =	1.2770e+04 (2.85e+03) -	1.0633e+04 (3.25e+03)
WCCI20-MTSO9-T2	1.6214e+03 (5.19e-01) =	1.6220e+03 (5.59e-01) -	**1.6213e+03 (4.95e-01) =**	1.6214e+03 (4.90e-01) =	1.6215e+03 (3.86e-01) =	1.6221e+03 (2.58e-01) -	1.6214e+03 (3.07e-01)
WCCI20-MTSO10-T1	2.8909e+04 (1.24e+04) -	2.0793e+04 (7.45e+03) -	2.4765e+04 (1.07e+04) -	4.4463e+04 (1.62e+04) -	3.4922e+04 (1.21e+04) -	1.6539e+04 (5.53e+03) -	**1.0915e+04 (2.64e+03)**
WCCI20-MTSO10-T2	1.4766e+06 (7.77e+05) =	1.5448e+06 (6.52e+05) =	1.2003e+06 (1.04e+06) =	2.2481e+06 (1.10e+06) -	1.9322e+06 (1.32e+06) =	1.3285e+06 (7.51e+05) =	**1.1685e+06 (7.04e+05)**
meanrank	4.28	3.70	3.15	5.15	5.83	3.40	2.50
+ / - / =	4 / 11 / 5	2 / 8 / 10	4 / 8 / 8	2 / 14 / 4	2 / 14 / 4	4 / 7 / 9	Base

TABLE III
THE AVERAGED STANDARD OBJECTIVE VALUE OF SEVEN COMPARED METHODS, OVER 20 INDEPENDENT RUNS ON THE SINGLE-OBJECTIVE MTO TEST SUITE 2. THE *meanrank* IS OBTAINED VIA FRIEDMAN'S TEST.

	MFEA	MFEA-II	MFEA-AKT	MFEA-GHS	SREMTO	AT-MFEA	MFEA-OC
WCCI24-MTSO1-T1	6.3586e+02 (5.48e+00) -	6.1253e+02 (2.30e+00) -	6.1435e+02 (2.36e+00) -	6.4084e+02 (6.89e+00) -	6.5785e+02 (4.35e+00) -	**6.0239e+02 (1.34e+00) +**	6.0688e+02 (1.15e+00)
WCCI24-MTSO1-T2	6.3806e+02 (7.48e+00) -	6.1188e+02 (3.27e+00) -	6.1487e+02 (2.23e+00) -	6.3808e+02 (7.01e+00) -	6.5693e+02 (2.85e+00) -	**6.0243e+02 (1.10e+00) +**	6.0692e+02 (1.29e+00)
WCCI24-MTSO2-T1	**7.0000e+02 (6.52e-03) +**	7.0005e+02 (1.59e-02) +	7.0001e+02 (7.86e-03) +	7.0006e+02 (2.90e-02) +	7.0098e+02 (5.71e-02) -	7.0002e+02 (7.15e-03) +	7.0009e+02 (2.74e-02)
WCCI24-MTSO2-T2	7.0000e+02 (4.09e-03) +	7.0006e+02 (1.28e-02) +	**7.0000e+02 (3.84e-03) +**	7.0006e+02 (2.50e-02) +	7.0096e+02 (4.40e-02) -	7.0002e+02 (8.12e-03) +	7.0011e+02 (2.22e-02)
WCCI24-MTSO3-T1	2.2914e+06 (1.46e+06) -	1.2878e+06 (6.50e+05) -	1.9228e+05 (1.14e+05) =	2.4174e+06 (1.35e+06) -	1.7811e+06 (9.38e+05) -	1.6693e+06 (1.35e+06) -	**1.5394e+05 (8.24e+04)**
WCCI24-MTSO3-T2	1.5744e+06 (9.92e+05) -	1.4753e+06 (1.17e+06) -	2.3738e+05 (2.25e+05) =	2.6499e+06 (1.20e+06) -	1.9547e+06 (9.65e+05) -	1.6313e+06 (1.06e+06) -	**1.7097e+05 (7.20e+04)**
WCCI24-MTSO4-T1	1.3005e+03 (8.10e-02) -	1.3005e+03 (7.83e-02) =	1.3005e+03 (7.19e-02) =	1.3005e+03 (9.96e-02) -	1.3006e+03 (1.05e-01) -	**1.3004e+03 (5.42e-02) =**	1.3004e+03 (5.36e-02)
WCCI24-MTSO4-T2	1.3004e+03 (7.34e-02) -	1.3004e+03 (5.62e-02) -	1.3004e+03 (7.44e-02) -	1.3004e+03 (6.62e-02) -	1.3005e+03 (7.60e-02) -	1.3004e+03 (5.11e-02) -	**1.3003e+03 (4.42e-02)**
WCCI24-MTSO5-T1	1.5251e+03 (4.95e+00) -	1.5132e+03 (4.70e+00) =	1.5238e+03 (5.13e+00) -	1.5289e+03 (6.66e+00) -	1.5512e+03 (1.17e+01) -	1.5179e+03 (8.97e+00) =	**1.5110e+03 (2.59e+00)**
WCCI24-MTSO5-T2	1.5234e+03 (4.83e+00) -	**1.5136e+03 (6.16e+00) =**	1.5212e+03 (4.25e+00) -	1.5232e+03 (4.35e+00) -	1.5554e+03 (9.11e+00) -	1.5209e+03 (8.72e+00) =	1.5157e+03 (7.71e+00)
WCCI24-MTSO6-T1	1.0398e+06 (6.87e+05) =	1.0563e+06 (6.35e+05) =	7.8700e+05 (5.53e+05) =	1.2910e+06 (9.81e+05) =	1.3809e+06 (8.35e+05) -	8.7966e+05 (3.48e+05) =	**7.2042e+05 (2.64e+05)**
WCCI24-MTSO6-T2	8.9946e+05 (4.86e+05) =	6.9928e+05 (4.19e+05) =	**4.3058e+05 (1.85e+05) =**	1.2227e+06 (6.73e+05) -	6.8732e+05 (3.23e+05) =	5.4003e+05 (4.39e+05) =	5.6023e+05 (3.60e+05)
WCCI24-MTSO7-T1	3.4919e+03 (4.16e+02) -	3.0112e+03 (4.11e+02) -	3.0844e+03 (2.97e+02) -	3.3726e+03 (3.31e+02) -	3.6312e+03 (3.19e+02) -	2.8486e+03 (4.90e+02) -	**2.4183e+03 (2.06e+02)**
WCCI24-MTSO7-T2	3.3232e+03 (3.59e+02) -	3.0963e+03 (4.69e+02) -	2.9682e+03 (3.16e+02) -	3.5545e+03 (3.71e+02) -	3.8512e+03 (3.92e+02) -	2.8339e+03 (4.14e+02) =	**2.4925e+03 (2.43e+02)**
WCCI24-MTSO8-T1	5.2011e+02 (5.45e-02) +	5.2118e+02 (3.66e-02) =	**5.2004e+02 (2.17e-02) +**	5.2012e+02 (9.92e-02) +	5.2008e+02 (4.04e-02) +	5.2117e+02 (3.58e-02) =	5.2119e+02 (3.69e-02)
WCCI24-MTSO8-T2	5.2012e+02 (7.02e-02) +	5.2117e+02 (4.58e-02) =	**5.2005e+02 (3.22e-02) +**	5.2010e+02 (9.56e-02) +	5.2009e+02 (2.72e-02) +	5.2119e+02 (2.69e-02) =	5.2119e+02 (2.95e-02)
WCCI24-MTSO9-T1	8.2329e+03 (1.08e+03) =	1.2998e+04 (2.69e+03) -	8.1883e+03 (9.66e+02) =	8.2344e+03 (1.11e+03) =	8.2001e+03 (1.03e+03) =	1.1184e+04 (3.58e+03) -	**7.3229e+03 (1.97e+03)**
WCCI24-MTSO9-T2	1.6214e+03 (5.95e-01) =	1.6219e+03 (6.02e-01) -	**1.6211e+03 (7.29e-01) =**	1.6214e+03 (5.08e-01) =	1.6217e+03 (6.73e-01) =	1.6221e+03 (3.04e-01) -	1.6214e+03 (4.08e-01)
WCCI24-MTSO10-T1	3.9578e+04 (1.35e+04) -	2.4053e+04 (8.54e+03) -	2.5232e+04 (1.15e+04) -	4.2741e+04 (1.75e+04) -	2.9905e+04 (1.05e+04) -	1.7969e+04 (1.08e+04) -	**1.0401e+04 (2.66e+03)**
WCCI24-MTSO10-T2	1.5991e+06 (9.07e+05) -	1.1731e+06 (5.76e+05) =	1.1914e+06 (6.70e+05) =	2.5276e+06 (1.44e+06) -	1.5743e+06 (1.04e+06) -	1.6836e+06 (9.45e+05) -	**8.5840e+05 (3.55e+05)**
meanrank	4.45	3.85	2.60	5.60	5.65	3.35	2.50
+ / - / =	4 / 12 / 4	2 / 9 / 9	4 / 8 / 8	4 / 13 / 3	2 / 15 / 3	4 / 7 / 9	Base

TABLE IV
THE AVERAGED STANDARD OBJECTIVE VALUE OF SEVEN COMPARED METHODS, OVER 20 INDEPENDENT RUNS ON THE SINGLE-OBJECTIVE MTO TEST SUITE 3. THE *meanrank* IS OBTAINED VIA FRIEDMAN'S TEST.

	MFEA	MFEA-II	MFEA-AKT	MFEA-GHS	SREMTO	AT-MFEA	MFEA-OC
PEPVM-T1	1.7630e-03 (4.23e-04) -	1.7027e-03 (4.75e-04) -	1.5687e-03 (4.39e-04) -	1.6629e-03 (5.22e-04) -	1.6027e-03 (4.55e-04) -	1.5227e-03 (3.89e-04) -	**1.1213e-03 (1.02e-04)**
PEPVM-T2	1.6832e-03 (3.89e-04) -	1.4446e-03 (4.03e-04) =	1.5440e-03 (4.39e-04) -	1.5733e-03 (5.08e-04) -	1.6599e-03 (5.25e-04) -	1.4334e-03 (3.64e-04) -	**1.1096e-03 (9.17e-05)**
PEPVM-T3	2.7755e-03 (4.88e-04) =	2.8620e-03 (6.03e-04) =	3.5269e-03 (1.35e-03) -	2.7956e-03 (4.49e-04) =	3.6886e-03 (1.26e-03) -	**2.5362e-03 (6.15e-05) =**	2.5585e-03 (1.00e-04)
meanrank	5.67	4.67	4.33	4.67	5.67	1.67	1.33
+ / - / =	0 / 2 / 1	0 / 1 / 2	0 / 3 / 0	0 / 2 / 1	0 / 3 / 0	0 / 2 / 1	Base

value of b in the approximate vicinity of 0.25 during the early phase of the algorithm. For the second design, we generate b values from a uniform distribution $U(0,1)$ for each crossover.

The results of our comparative experiments are presented in Table V. These experiments were conducted using the same parameter configurations as test suite 1, and 20 independent runs were performed to obtain average values and variances for comparison. The table reveals that the original MFEA-OC algorithm outperforms the other three versions. In addition, convergence trend figures are presented in the Supplementary Materials. These figures illustrate that our proposed MFEA-OC exhibits a distinct advantage over the other three variants of MFEA-OC during the early phase characterized by rapid fitness decline. As the algorithm progresses, the original

TABLE V
THE AVERAGED STANDARD OBJECTIVE VALUE OF FOUR VERSIONS OF THE MFEA-OC ALGORITHM OVER 20 INDEPENDENT RUNS ON THE SINGLE-OBJECTIVE MTO TEST SUITE 1.

	MFEA-OC	MFEA-OC(b=rand())	MFEA-OC(only use SBX)	MFEA-OC (b=0.25)
WCCI20-MTSO1-T1	**6.0663e+02 (2.20e+00)**	6.0872e+02 (1.50e+00) -	6.0793e+02 (1.27e+00) -	6.1089e+02 (2.07e+00) -
WCCI20-MTSO1-T2	**6.0713e+02 (1.51e+00)**	6.0842e+02 (1.32e+00) -	6.0792e+02 (1.22e+00) =	6.1212e+02 (1.81e+00) -
WCCI20-MTSO2-T1	**7.0011e+02 (2.35e-02)**	7.0071e+02 (1.46e-01) -	7.0065e+02 (1.45e-01) -	7.0069e+02 (1.30e-01) -
WCCI20-MTSO2-T2	**7.0011e+02 (2.40e-02)**	7.0064e+02 (1.08e-01) -	7.0073e+02 (1.38e-01) -	7.0076e+02 (1.37e-01) -
WCCI20-MTSO3-T1	**1.9473e+05 (1.17e+05)**	2.7795e+05 (1.44e+05) -	2.6692e+05 (1.38e+05) -	3.3293e+06 (2.09e+06) -
WCCI20-MTSO3-T2	**2.0663e+05 (1.05e+05)**	3.0349e+05 (1.75e+05) -	2.5818e+05 (1.36e+05) =	3.4434e+06 (2.05e+06) -
WCCI20-MTSO4-T1	**1.3004e+03 (6.17e-02)**	1.3005e+03 (4.94e-02) -	1.3005e+03 (5.37e-02) -	1.3005e+03 (8.72e-02) -
WCCI20-MTSO4-T2	**1.3003e+03 (7.97e-02)**	1.3004e+03 (4.75e-02) -	1.3004e+03 (5.78e-02) -	1.3005e+03 (5.99e-02) -
WCCI20-MTSO5-T1	**1.5111e+03 (2.90e+00)**	1.5151e+03 (4.40e+00) -	1.5158e+03 (5.57e+00) -	1.5164e+03 (4.08e+00) -
WCCI20-MTSO5-T2	**1.5151e+03 (5.94e+00)**	1.5167e+03 (6.44e+00) =	1.5174e+03 (5.30e+00) =	1.5160e+03 (4.22e+00) =
WCCI20-MTSO6-T1	**6.8931e+05 (3.25e+05)**	1.2295e+06 (5.56e+05) -	9.4364e+05 (5.03e+05) =	1.8679e+06 (1.30e+06) -
WCCI20-MTSO6-T2	**5.4306e+05 (4.13e+05)**	1.0953e+06 (7.42e+05) -	6.7500e+05 (5.76e+05) =	1.8601e+06 (1.20e+06) -
WCCI20-MTSO7-T1	2.4054e+03 (2.05e+02)	2.4094e+03 (2.42e+02) =	**2.4018e+03 (1.95e+02) =**	3.1246e+03 (3.14e+02) -
WCCI20-MTSO7-T2	2.5285e+03 (2.13e+02)	2.5572e+03 (2.32e+02) =	**2.4379e+03 (1.52e+02) =**	3.0284e+03 (3.70e+02) -
WCCI20-MTSO8-T1	**5.2118e+02 (3.38e-02)**	5.2119e+02 (3.01e-02) =	5.2119e+02 (4.43e-02) =	5.2119e+02 (3.95e-02) =
WCCI20-MTSO8-T2	**5.2116e+02 (3.57e-02)**	5.2118e+02 (4.91e-02) =	5.2119e+02 (4.63e-02) =	5.2117e+02 (3.93e-02) =
WCCI20-MTSO9-T1	1.0633e+04 (3.25e+03)	**1.0546e+04 (3.41e+03) =**	1.1626e+04 (3.11e+03) =	1.3793e+04 (1.49e+03) -
WCCI20-MTSO9-T2	1.6214e+03 (3.07e-01)	1.6213e+03 (3.36e-01) =	**1.6210e+03 (5.60e-01) +**	1.6221e+03 (2.55e-01) -
WCCI20-MTSO10-T1	**1.0915e+04 (2.64e+03)**	1.2355e+04 (3.21e+03) =	1.2303e+04 (1.95e+03) -	3.0855e+04 (8.63e+03) -
WCCI20-MTSO10-T2	1.1685e+06 (7.04e+05)	**1.0391e+06 (7.12e+05) =**	1.2380e+06 (5.06e+05) =	2.3469e+06 (9.73e+05) -
+ / - / =	Base	0 / 11 / 9	1 / 8 / 11	0 / 17 / 3

MFEA-OC algorithm maintains its advantages over other approaches. This observation emphasizes the pivotal role of the design formula for the optimal crossover operator in the later phase of the algorithm, surpassing the performance achievable with a fixed or randomly selected b. Moreover, employing only the SBX operator yields the slowest convergence rate. Consequently, we not only present a theoretical design formula for the optimal crossover operator but also empirically validate its superiority over the widely adopted SBX operator or other fundamental design approaches.

3) The Early and Later Phases: In the design of the optimal crossover operator, we partitioned the design formula into two categories Eqs. (15) and (16), namely the early phase and later phase, based on theoretical analysis. To determine the necessity of distinguishing different phases, we conducted a comparative experiment, while disregarding the determination strategy for the early and later phases. Specifically, we compared two settings: using only the early phase design Eq. (15) and using only the late phase design Eq. (16). We summarize the experimental results of the average convergence trend for the initial 80 generations in the Supplementary Materials.

The figures demonstrate the superior performance of design Eq. (15) over Eq. (16) in the early phase of population evolution, validating our theoretical analysis. However, as the population evolves further, the advantage of using Eq. (15) gradually diminishes and may even be surpassed. Therefore, it may be more advantageous to employ Eq. (16) in the later phase, because some individuals get trapped in local optima, decreasing the randomness of the population distribution. Thus, exclusively relying on either Eq. (15) or (16), the entire process is suboptimal. An optimal design should be able to use Eq. (15) in the early phase and transition to Eq. (16) in the later phase. To address this requirement, the adaptive mechanism Algorithm 4 was introduced into the MFEA-OC algorithm.

4) Different Switching Strategies: In this section, we conduct experiments to assess the effectiveness of the adaptive switching mechanism in MFEA-OC. To achieve this, we set up several control groups in which we replaced the adaptive switching mechanism of Eqs. (15) and (16) in MFEA-OC with fixed switching at predetermined generations. The objective of this experiment was to compare the performance of our proposed adaptive switching mechanism with that of direct switching. Specifically, we selected the 50th, 100th, 150th, and 200th generations as the fixed switching moments, transitioning from the early phase Eq. (15) to the late phase Eq. (16). We compared these results with those obtained from our algorithm using non-adaptive switching strategies, and the experimental outcomes are presented in Table VI. As observed in the table, irrespective of the chosen fixed switching moment, the final results were significantly inferior to those achieved using MFEA-OC with the adaptive switching mechanism. Thus, the switching mechanism introduced in Algorithm 4 effectively discriminates between the early and late phases of population evolution. This phenomenon can be readily explained from a theoretical standpoint. Although a fixed switching moment may be suitable for a specific task, the population evolution varies across different tasks. Therefore, when dealing with multiple tasks, using a fixed generation cannot effectively distinguish between the early and late phases of population evolution.

VI. CONCLUSION

This study investigates the phenomenon of negative transfer in MTO from the perspective of crossover operator design. We perform a mathematical analysis to determine the type of crossover operator that can effectively mitigate negative

TABLE VI
THE AVERAGED STANDARD OBJECTIVE VALUE OF DIFFERENT SWITCHING STRATEGIES OF THE MFEA-OC ALGORITHM OVER 20 INDEPENDENT RUNS ON THE SINGLE-OBJECTIVE MTO TEST SUITE 1.

	MFEA-OC	MFEA-OC(50Gen)	MFEA-OC(100Gen)	MFEA-OC(150Gen)	MFEA-OC(200Gen)
WCCI20-MTSO1-T1	**6.0663e+02 (2.20e+00)**	6.0815e+02 (1.65e+00) -	6.0897e+02 (1.95e+00) -	6.0869e+02 (1.49e+00) -	6.0875e+02 (1.92e+00) -
WCCI20-MTSO1-T2	**6.0713e+02 (1.51e+00)**	6.0878e+02 (1.75e+00) -	6.0885e+02 (1.34e+00) -	6.0866e+02 (1.48e+00) -	6.0893e+02 (1.55e+00) -
WCCI20-MTSO2-T1	**7.0011e+02 (2.35e-02)**	7.0068e+02 (1.19e-01) -	7.0070e+02 (8.98e-02) -	7.0068e+02 (1.48e-01) -	7.0064e+02 (1.20e-01) -
WCCI20-MTSO2-T2	**7.0011e+02 (2.40e-02)**	7.0071e+02 (1.19e-01) -	7.0068e+02 (1.41e-01) -	7.0065e+02 (1.38e-01) -	7.0071e+02 (1.67e-01) -
WCCI20-MTSO3-T1	**1.9473e+05 (1.17e+05)**	3.4559e+05 (1.64e+05) -	2.6964e+05 (1.44e+05) =	2.5681e+05 (1.02e+05) -	2.6339e+05 (1.10e+05) -
WCCI20-MTSO3-T2	**2.0663e+05 (1.05e+05)**	3.0939e+05 (1.31e+05) -	2.5528e+05 (1.65e+05) =	2.5745e+05 (1.28e+05) =	2.8988e+05 (1.68e+05) =
WCCI20-MTSO4-T1	**1.3004e+03 (6.17e-02)**	1.3005e+03 (5.95e-02) -	1.3005e+03 (4.64e-02) -	1.3005e+03 (5.59e-02) -	1.3005e+03 (5.22e-02) -
WCCI20-MTSO4-T2	**1.3003e+03 (7.97e-02)**	1.3004e+03 (6.02e-02) -	1.3004e+03 (5.12e-02) -	1.3004e+03 (5.89e-02) -	1.3004e+03 (6.66e-02) -
WCCI20-MTSO5-T1	**1.5111e+03 (2.90e+00)**	1.5143e+03 (4.55e+00) -	1.5142e+03 (3.78e+00) -	1.5142e+03 (3.13e+00) -	1.5164e+03 (4.77e+00) -
WCCI20-MTSO5-T2	**1.5151e+03 (5.94e+00)**	1.5173e+03 (6.23e+00) =	1.5161e+03 (5.04e+00) =	1.5154e+03 (3.35e+00) =	1.5164e+03 (6.34e+00) =
WCCI20-MTSO6-T1	**6.8931e+05 (3.25e+05)**	1.1379e+06 (1.08e+06) =	1.1365e+06 (6.06e+05) -	9.5206e+05 (6.29e+05) =	1.0393e+06 (6.74e+05) =
WCCI20-MTSO6-T2	5.4306e+05 (4.13e+05)	8.1795e+05 (4.58e+05) -	8.0745e+05 (8.27e+05) =	5.7639e+05 (4.10e+05) =	**4.7160e+05 (2.93e+05) =**
WCCI20-MTSO7-T1	**2.4054e+03 (2.05e+02)**	2.4873e+03 (2.85e+02) =	2.4536e+03 (1.96e+02) =	2.4612e+03 (2.76e+02) =	2.5586e+03 (2.85e+02) =
WCCI20-MTSO7-T2	**2.5285e+03 (2.13e+02)**	2.5796e+03 (3.83e+02) =	2.5834e+03 (2.34e+02) =	2.5619e+03 (2.55e+02) =	2.5425e+03 (2.37e+02) =
WCCI20-MTSO8-T1	5.2118e+02 (3.38e-02)	5.2118e+02 (4.44e-02) =	5.2118e+02 (3.49e-02) =	**5.2117e+02 (4.08e-02) =**	5.2120e+02 (4.02e-02) -
WCCI20-MTSO8-T2	**5.2116e+02 (3.57e-02)**	5.2119e+02 (4.00e-02) =	5.2118e+02 (2.96e-02) =	5.2118e+02 (4.85e-02) =	5.2120e+02 (3.04e-02) -
WCCI20-MTSO9-T1	**1.0633e+04 (3.25e+03)**	1.1768e+04 (3.42e+03) =	1.2798e+04 (2.85e+03) -	1.1947e+04 (3.21e+03) =	1.2969e+04 (2.45e+03) -
WCCI20-MTSO9-T2	**1.6214e+03 (3.07e-01)**	1.6215e+03 (4.20e-01) =	1.6214e+03 (3.15e-01) =	1.6215e+03 (4.70e-01) =	1.6214e+03 (5.36e-01) =
WCCI20-MTSO10-T1	**1.0915e+04 (2.64e+03)**	1.4659e+04 (3.37e+03) -	1.2151e+04 (3.68e+03) =	1.2929e+04 (3.79e+03) -	1.1511e+04 (2.57e+03) =
WCCI20-MTSO10-T2	1.1685e+06 (7.04e+05)	1.4391e+06 (8.18e+05) =	1.3482e+06 (7.00e+05) =	1.2894e+06 (8.72e+05) =	**1.0615e+06 (4.72e+05) =**
+ / - / =	Base	0 / 11 / 9	0 / 9 / 11	0 / 9 / 11	0 / 11 / 9

transfer and present design formulas for the optimal crossover operator. Because this formula relies on information about the global optima of different tasks, which is not known beforehand, a practical approximation design method is proposed. Furthermore, a novel algorithm named MFEA-OC is introduced, which incorporates the design of the optimal crossover operator into the conventional MFEA framework while maintaining a straightforward logical structure. Experimental results demonstrate that the new algorithm surpasses many state-of-the-art algorithms on various test suites, underscoring the potential of the optimal crossover operator in the field of evolutionary multitasking.

Several avenues for future research on the optimal crossover operator can be explored. For example, the approximation method used in this study divides the algorithm into early and late phases and employs an adaptive switching mechanism to approximate the optimal crossover. However, this approach introduces some inaccuracies. Enhancing the approximation methods can improve the performance of the optimal crossover operator. Moreover, the optimal crossover operator can be integrated into various multitask EAs, going beyond the scope of the MFEA, even though the integration process may vary. Finally, the framework used in this study defines negative transfer based on the Euclidean distance. Exploring alternative definitions of negative transfer could lead to different understandings of the optimal crossover operator and the corresponding design formulas.

The source code of MFEA-OC written in MATLAB is available online at https://github.com/Dashla/MFEA_OC/tree/main.

APPENDIX

A. Proof of Theorem 1

For $j = 1, \ldots, n$, denote $T_j = o_j - p_j$, by $\sum_{i=1}^{n} a_{ij} = 1$, we have $T_j = \sum_{i \neq j} a_{ij}(p_i - p_j)$. Next, we compute

$$\begin{aligned}
&\sum_{i=1}^{n} \|p_i - \theta_i^*\|^2 - \sum_{i=1}^{n} \|o_i - \theta_i^*\|^2 \\
=&\sum_{l=1}^{n} \left(2\left(\theta_l^* - p_l\right)^T T_l - \|T_l\|^2\right) \\
=&\underbrace{\sum_{l=1}^{n} 2\left(\theta_l^*\right)^T T_l}_{K_1} + \underbrace{\sum_{l=1}^{n} \left(-\left(p_l\right)^T T_l - \|T_l\|^2\right)}_{K_2}.
\end{aligned}$$

By the symmetry of $\mathcal{A}$, we derive

$$\begin{aligned}
K_1 &= \sum_{l=1}^{n} 2\left(\theta_l^*\right)^T \sum_{j \neq l} a_{jl}\left(p_j - p_l\right) \\
&= \sum_{l=1}^{n} \sum_{j=1}^{n} 2a_{jl}\left(\theta_l^*\right)^T \left(p_j - p_l\right) \\
&= \sum_{l<j} 2a_{jl}\left(\theta_l^* - \theta_j^*\right)^T \left(p_j - p_l\right).
\end{aligned}$$

Moreover, by applying Lagrange's Identities,

$$\begin{aligned}
K_2 &= -\sum_{l=1}^{n}\left(\theta_l^k\right)^T\sum_{j\neq l}a_{jl}\left(p_j-p_l\right)-\sum_{l=1}^{n}\left\|\sum_{j\neq l}a_{jl}\left(p_j-p_l\right)\right\|^2\\
&=\frac{1}{2}\sum_{l=1}^{n}\sum_{j\neq l}a_{jl}\left\|p_j-p_l\right\|^2-\sum_{l=1}^{n}\left\|\sum_{j\neq l}a_{jl}\left(\theta_j^k-\theta_l^k\right)\right\|^2\\
&=\sum_{l=1}^{n}\left(\sum_{j\neq l}a_{jl}\left\|p_j-p_l\right\|^2-\left\|\sum_{j\neq l}a_{jl}\left(p_j-p_l\right)\right\|^2\right)\\
&\quad-\frac{1}{2}\sum_{l=1}^{n}\sum_{j\neq l}a_{jl}\left\|p_j-p_l\right\|^2\\
&=\sum_{l=1}^{n}\sum_{i<j}a_{il}a_{jl}\left\|p_i-p_j\right\|^2-\sum_{i<j}a_{ij}\left\|p_i-p_j\right\|^2,
\end{aligned}$$

then, we deduce

$$\begin{aligned}
&\sum_{i=1}^{n}\left\|p_i-\theta_i^*\right\|^2-\sum_{i=1}^{n}\left\|o_i-\theta_i^*\right\|^2\\
=&\sum_{i<j}\left\|p_i-p_j\right\|\cdot\left(\left(\sum_{l=1}^{n}a_{il}a_{jl}-a_{ij}\right)\left\|p_i-p_j\right\|\right.\\
&\left.-2a_{ij}\left(\theta_i^*-\theta_j^*\right)^T\cdot\frac{\left(p_i-p_j\right)}{\left\|p_i-p_j\right\|}\right),
\end{aligned}$$

which implies Theorem 1.

B. Proof of Theorem 2

It suffices to show there is a constant δ such that for any $i\neq j$,

$$P\left(A_{ij}\right)>1-4e^{-d\delta}. \tag{18}$$

Select $0<\epsilon<\min_{i<j}(\sum_{l=1}^{n}a_{il}a_{jl}-a_{ji})$, by [44, Example 2.11], we derive

$$P\left(\left|\frac{\left\|p_i-p_j\right\|^2}{2d}-1\right|>\epsilon\right)\leq 2e^{-d^2\epsilon^2/4}. \tag{19}$$

On the other hand, by Hoeffding's bound [44, Proposition 2.5],

$$\begin{aligned}
&P\left(a_{ji}\left(\theta_i^*-\theta_j^*\right)^T\cdot\left(p_i-p_j\right)>d\epsilon(1-\epsilon)\right)\\
\leq&e^{-d^2\epsilon^2(1-\epsilon)^2/4\left\|\theta_i^*-\theta_j^*\right\|^2}\leq e^{-d\epsilon^2(1-\epsilon)^2/4}.
\end{aligned} \tag{20}$$

Note that

$$\begin{aligned}
A_{ij}^c\subset&\left\{\left|\left\|p_i-p_j\right\|^2-2d\right|>2d\epsilon\right\}\\
&\cup\left\{a_{ji}\left(\theta_i^*-\theta_j^*\right)^T\cdot\left(p_i-p_j\right)>d\epsilon(1-\epsilon)\right\},
\end{aligned}$$

then by (19) and (20),

$$P\left(A_{ij}^c\right)\leq\quad 2e^{-d^2\epsilon^2/4}+e^{-d\epsilon^2(1-\epsilon)^2/4}<3e^{-d\epsilon^2/8},$$

which is exactly (18) for $\delta=\epsilon^2/8$.

REFERENCES

[1] T. Back, U. Hammel, and H.-P. Schwefel, "Evolutionary computation: comments on the history and current state," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 3–17, 1997.

[2] M. Amir Haeri, M. M. Ebadzadeh, and G. Folino, "Statistical genetic programming for symbolic regression," *Applied Soft Computing*, vol. 60, pp. 447–469, 2017.

[3] D. Datta, K. Deb, C. M. Fonseca, F. Lobo, and P. Condado, "Multi-objective evolutionary algorithm for land-use management problem," *International Journal of Computational Intelligence Research*, vol. 3, no. 4, pp. 1–24, 2007.

[4] B. M. Baker and M. Ayechew, "A genetic algorithm for the vehicle routing problem," *Computers & Operations Research*, vol. 30, no. 5, pp. 787–800, 2003.

[5] T.-L. Lin, S.-J. Horng, T.-W. Kao, Y.-H. Chen, R.-S. Run, R.-J. Chen, J.-L. Lai, and I.-H. Kuo, "An efficient job-shop scheduling algorithm based on particle swarm optimization," *Expert Systems with Applications*, vol. 37, no. 3, pp. 2629–2636, 2010.

[6] P. Junjie and W. Dingwei, "An ant colony optimization algorithm for multiple travelling salesman problem," in *First International Conference on Innovative Computing, Information and Control-Volume I (ICICIC'06)*, vol. 1. IEEE, 2006, pp. 210–213.

[7] B. J. Park, H. R. Choi, and H. S. Kim, "A hybrid genetic algorithm for the job shop scheduling problems," *Computers & industrial engineering*, vol. 45, no. 4, pp. 597–613, 2003.

[8] A. Gupta, Y.-S. Ong, and L. Feng, "Multifactorial evolution: toward evolutionary multitasking," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 3, pp. 343–357, 2015.

[9] T. Wei, S. Wang, J. Zhong, D. Liu, and J. Zhang, "A review on evolutionary multi-task optimization: Trends and challenges," *IEEE Transactions on Evolutionary Computation (Early Access)*, pp. 1–20, 2021.

[10] K. C. Tan, L. Feng, and M. Jiang, "Evolutionary transfer optimization - a new frontier in evolutionary computation research," *IEEE Computational Intelligence Magazine*, vol. 16, no. 1, pp. 22–33, 2021.

[11] W. Li, X. Gao, and L. Wang, "Multifactorial evolutionary algorithm with adaptive transfer strategy based on decision tree," *Complex & Intelligent Systems*, pp. 1–32, 2023.

[12] X. Wang, Q. Kang, M. Zhou, Z. Fan, and A. Albeshri, "A knowledge sharing and individually guided evolutionary algorithm for multi-task optimization problems," *Applied Sciences*, vol. 13, no. 1, p. 602, 2023.

[13] X. Ma, Y. Zheng, Z. Zhu, X. Li, L. Wang, Y. Qi, and J. Yang, "Improving evolutionary multitasking optimization by leveraging inter-task gene similarity and mirror transformation," *IEEE Computational Intelligence Magazine*, vol. 16, no. 4, pp. 38–53, 2021.

[14] X. Yang, W. Li, R. Wang, and K. Yang, "Helper objective-based multifactorial evolutionary algorithm for continuous optimization," *Swarm and Evolutionary Computation*, vol. 78, p. 101279, 2023.

[15] S. Yang, Y. Qi, R. Yang, X. Ma, and H. Zhang, "A surrogate assisted evolutionary multitasking optimization algorithm," *Applied Soft Computing*, vol. 132, p. 109775, 2023.

[16] K. K. Bali, Y.-S. Ong, A. Gupta, and P. S. Tan, "Multifactorial evolutionary algorithm with online transfer parameter estimation: MFEA-II," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 1, pp. 69–83, 2019.

[17] S. Li, W. Gong, L. Wang, and Q. Gu, "Evolutionary multitasking via reinforcement learning," *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2023.

[18] H. T. Thanh Binh, N. Quoc Tuan, and D. C. Thanh Long, "A multi-objective multi-factorial evolutionary algorithm with reference-point-based approach," in *2019 IEEE Congress on Evolutionary Computation (CEC)*, 2019, pp. 2824–2831.

[19] H. Xu, A. K. Qin, and S. Xia, "Evolutionary multitask optimization with adaptive knowledge transfer," *IEEE Transactions on Evolutionary Computation*, vol. 26, no. 2, pp. 290–303, 2022.

[20] B. Da, A. Gupta, and Y.-S. Ong, "Curbing negative influences online for seamless transfer evolutionary optimization," *IEEE transactions on cybernetics*, vol. 49, no. 12, pp. 4365–4378, 2018.

[21] L. Feng, L. Zhou, J. Zhong, A. Gupta, Y.-S. Ong, K.-C. Tan, and A. K. Qin, "Evolutionary multitasking via explicit autoencoding," *IEEE transactions on cybernetics*, vol. 49, no. 9, pp. 3457–3470, 2018.

[22] W. Lin, Q. Lin, L. Feng, and K. C. Tan, "Ensemble of domain adaptation-based knowledge transfer for evolutionary multitasking," *IEEE Transactions on Evolutionary Computation*, 2023.

[23] R. Lim, A. Gupta, Y.-S. Ong, L. Feng, and A. N. Zhang, "Non-linear domain adaptation in transfer evolutionary optimization," *Cognitive Computation*, vol. 13, pp. 290–307, 2021.

[24] K. K. Bali, A. Gupta, L. Feng, Y. S. Ong, and T. P. Siew, "Linearized domain adaptation in evolutionary multitasking," in *2017 IEEE Congress on evolutionary computation (CEC)*. IEEE, 2017, pp. 1295–1302.

[25] X. Xue, K. Zhang, K. C. Tan, L. Feng, J. Wang, G. Chen, X. Zhao, L. Zhang, and J. Yao, "Affine transformation-enhanced multifactorial optimization for heterogeneous problems," *IEEE Transactions on Cybernetics*, vol. 52, no. 7, pp. 6217–6231, 2020.

[26] Z. Liang, J. Zhang, L. Feng, and Z. Zhu, "A hybrid of genetic transform and hyper-rectangle search strategies for evolutionary multi-tasking," *Expert Systems with Applications*, vol. 138, p. 112798, 2019.

[27] L. Zhou, L. Feng, K. C. Tan, J. Zhong, Z. Zhu, K. Liu, and C. Chen, "Toward adaptive knowledge transfer in multifactorial evolutionary computation," *IEEE transactions on cybernetics*, vol. 51, no. 5, pp. 2563–2576, 2020.

[28] Y. Li, W. Gong, and S. Li, "Multitasking optimization via an adaptive solver multitasking evolutionary framework," *Information Sciences*, vol. 630, pp. 688–712, 2023.

[29] B. Koohestani, "A crossover operator for improving the efficiency of permutation-based genetic algorithms," *Expert Systems with Applications*, vol. 151, p. 113381, 2020.

[30] S. Molaei, H. Moazen, S. Najjar-Ghabel, and L. Farzinvash, "Particle swarm optimization with an enhanced learning strategy and crossover operator," *Knowledge-Based Systems*, vol. 215, p. 106768, 2021.

[31] Z. Liu, G. Li, H. Zhang, Z. Liang, and Z. Zhu, "Multifactorial evolutionary algorithm based on diffusion gradient descent," *IEEE Transactions on Cybernetics*, pp. 1–13, 2023.

[32] Y. Jiang, Z.-H. Zhan, K. C. Tan, and J. Zhang, "Block-level knowledge transfer for evolutionary multitask optimization," *IEEE Transactions on Cybernetics*, vol. 54, no. 1, pp. 558–571, 2024.

[33] L. Feng, Y. Huang, L. Zhou, J. Zhong, A. Gupta, K. Tang, and K. C. Tan, "Explicit evolutionary multitasking for combinatorial optimization: A case study on capacitated vehicle routing problem," *IEEE transactions on cybernetics*, vol. 51, no. 6, pp. 3143–3156, 2020.

[34] P. D. Thanh, H. T. T. Binh, and T. B. Trung, "An efficient strategy for using multifactorial optimization to solve the clustered shortest path tree problem," *Applied Intelligence*, vol. 50, pp. 1233–1258, 2020.

[35] L. Zhou, L. Feng, J. Zhong, Y.-S. Ong, Z. Zhu, and E. Sha, "Evolutionary multitasking in combinatorial search spaces: A case study in capacitated vehicle routing problem," in *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2016, pp. 1–8.

[36] A. Moraglio and D. Sudholt, "Principled design and runtime analysis of abstract convex evolutionary search," *Evolutionary computation*, vol. 25, no. 2, pp. 205–236, 2017.

[37] T. Jones, S. Forrest *et al.*, "Fitness distance correlation as a measure of problem difficulty for genetic algorithms." in *ICGA*, vol. 95, 1995, pp. 184–192.

[38] H. R. Tizhoosh, "Opposition-based learning: a new scheme for machine intelligence," in *International conference on computational intelligence for modelling, control and automation and international conference on intelligent agents, web technologies and internet commerce (CIMCA-IAWTIC'06)*, vol. 1. IEEE, 2005, pp. 695–701.

[39] X. Ma, Y. Zheng, Z. Zhu, X. Li, L. Wang, Y. Qi, and J. Yang, "Improving evolutionary multitasking optimization by leveraging inter-task gene similarity and mirror transformation," *IEEE Computational Intelligence Magazine*, vol. 16, no. 4, pp. 38–53, 2021.

[40] Z. Liang, J. Zhang, L. Feng, and Z. Zhu, "A hybrid of genetic transform and hyper-rectangle search strategies for evolutionary multi-tasking," *Expert Systems with Applications*, vol. 138, p. 112798, 2019.

[41] X. Zheng, A. K. Qin, M. Gong, and D. Zhou, "Self-regulated evolutionary multitask optimization," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 1, pp. 16–28, 2019.

[42] Y. Li, W. Gong, and S. Li, "Evolutionary competitive multitasking optimization via improved adaptive differential evolution," *Expert Systems with Applications*, vol. 217, p. 119550, 2023.

[43] Y. Li, W. Gong, F. Ming, T. Zhang, S. Li, and Q. Gu, "Mtop: A matlab optimization platform for evolutionary multitasking," *arXiv preprint arXiv:2312.08134*, 2023.

[44] M. J. Wainwright, *High-dimensional statistics: A non-asymptotic viewpoint*. Cambridge university press, 2019, vol. 48.

Zhaobo Liu received the B.S. degree in mathemat ics and applied mathematics from Peking University, Beijing, China, in 2015, and the Ph.D. degree in control theory from the Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, in 2020. He is currently an Assistant Professor with the Institute for Advanced Study, Shenzhen University, Shenzhen, China.

His current research interests include adaptive control, data science, and computational intelligence.

Jianhua Yuan received the B.E. degree from Taiyuan University of Technology in 2018. He is currently pursuing his master's degree with College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China.

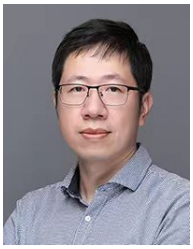
His current research interests include computational intelligence and reinforcement learning.

Haili Zhang obtained her B.S. degree in Mathematics and Applied Mathematics from the University of Science and Technology Beijing, in 2015, and her Ph.D. degree in Statistics from the Academy of Mathematics and Systems Science at the Chinese Academy of Sciences, Beijing, in 2020. She is presently serving as a lecturer at Shenzhen Polytechnic University.

Her research interests include model averaging, with a particular focus on functional data analysis, distributed data analysis, and high-dimensional data.

Tao Zeng received the B.E. degree from Jiangxi Agricultural University and the M.S. degree from Nanchang Institute of Technology, China, in 2019 and 2023, respectively. He is currently pursuing a Ph.D. degree with the National Engineering Laboratory for Big Data System Computing Technology, Shenzhen University, Shenzhen, China.

His current research interests include computational intelligence and large language models.



Zexuan Zhu (Senior Member, IEEE) received the B.S. degree in computer science and technology from Fudan University, China, in 2003, and the Ph.D. degree in computer engineering from Nanyang Technological University, Singapore, in 2008. He is currently a Professor with the National Engineering Laboratory for Big Data System Computing Technology, Shenzhen University, Shenzhen, China. His research interests include computational intelligence, machine learning, and bioinformatics.