Matching-Based Selection With Incomplete Lists for Decomposition Multiobjective Optimization

Mengyuan Wu, Ke Li, Sam Kwong, *Fellow, IEEE*, Yu Zhou, and Qingfu Zhang, *Fellow, IEEE*

***Abstract*—The balance between convergence and diversity is the cornerstone of evolutionary multiobjective optimization (EMO). The recently proposed stable matching-based selection provides a new perspective to handle this balance under the framework of decomposition multiobjective optimization. In particular, the one-one stable matching between subproblems and solutions, which achieves an equilibrium between their mutual preferences, is claimed to strike a balance between convergence and diversity. However, the original stable marriage model has a high risk of matching a solution with an unfavorable subproblem, which finally leads to an imbalanced selection result. In this paper, we introduce the concept of incomplete preference lists into the stable matching model to remedy the loss of population diversity. In particular, each solution is only allowed to maintain a partial preference list consisting of its favorite subproblems. We implement two versions of stable matching-based selection mechanisms with incomplete preference lists: one achieves a two-level one-one matching and the other obtains a many-one matching. Furthermore, an adaptive mechanism is developed to automatically set the length of the incomplete preference list for each solution according to its local competitiveness. The effectiveness and competitiveness of our proposed methods are validated and compared with several state-of-the-art EMO algorithms on 62 benchmark problems.**

***Index Terms*—Adaptive mechanism, convergence and diversity, decomposition, multiobjective optimization, stable matching with incomplete lists.**

I. Introduction

THE MULTIOBJECTIVE optimization problem (MOP) considered in this paper is defined as follows [1]:

$$\text{minimize} \quad \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_m(\mathbf{x}))^T$$
$$\text{subject to} \quad \mathbf{x} \in \Omega \tag{1}$$

where $\mathbf{x} = (x_1, \ldots, x_n)^T$ is a n-dimensional decision vector and $\mathbf{F}(\mathbf{x})$ is a m-dimensional objective vector. $\Omega \subseteq \mathbb{R}^n$ is the feasible region of the decision space, while $\mathbf{F} : \Omega \rightarrow \mathbb{R}^m$ is the corresponding attainable set in the objective space $\mathbb{R}^m$. Given two solutions $\mathbf{x}^1, \mathbf{x}^2 \in \Omega$, $\mathbf{x}^1$ is said to dominate $\mathbf{x}^2$, denoted by $\mathbf{x}^1 \preceq \mathbf{x}^2$, if and only if $f_i(\mathbf{x}^1) \leq f_i(\mathbf{x}^2)$ for all $i \in \{1, \ldots, m\}$ and $\mathbf{F}(\mathbf{x}^1) \neq \mathbf{F}(\mathbf{x}^2)$. A solution $\mathbf{x}^* \in \Omega$ is said to be Pareto optimal if and only if no solution $\mathbf{x} \in \Omega$ dominates it. All Pareto optimal solutions constitute the Pareto-optimal set (PS) and the corresponding Pareto-optimal front (PF) is defined as $\text{PF} = \{\mathbf{F}(\mathbf{x}) | \mathbf{x} \in \text{PS}\}$.

Evolutionary multiobjective optimization (EMO) algorithms, which are capable of approximating the PS and PF in a single run, have been widely accepted as a major approach for multiobjective optimization. Convergence and diversity are two cornerstones of multiobjective optimization: the former means the closeness to the PF while the latter indicates the spread and uniformity along the PF. Selection, which determines the survival of the fittest, plays a key role in balancing convergence and diversity. According to different selection mechanisms, the existing EMO algorithms can be roughly classified into three categories, i.e., Pareto-based methods [2]–[4], indicator-based methods [5]–[7], and decomposition-based methods [8]–[10].

This paper focuses on the decomposition-based methods, especially the multiobjective evolutionary algorithm based on decomposition (MOEA/D) [8]. The original MOEA/D employs a steady-state selection mechanism, where the population is updated immediately after the generation of an offspring. In particular, this offspring is able to replace its neighboring parents when it has a better aggregation function value for the corresponding subproblem. To avoid a superior solution overwhelmingly occupying the whole population, Li and Zhang [11] suggested to restrict the maximum number of replacements taken by an offspring. More recently, Li *et al.* [10] developed a new perspective to understand the selection process of MOEA/D. Specifically, the selection process of MOEA/D is modeled as a one-one matching problem, where subproblems and solutions are treated as two sets of matching agents whose mutual preferences are defined as the convergence and diversity, respectively. Therefore, a stable matching between subproblems and solutions achieves an equilibrium between their mutual preferences, leading to a balance between convergence and diversity. However,

Manuscript received August 21, 2016; revised December 20, 2016; accepted January 7, 2017. Date of publication January 31, 2017; date of current version July 27, 2017. This work was supported in part by the Hong Kong RGC General Research Fund GRF Grant 9042038 (CityU 11205314), in part by the ANR/RCC Joint Research Scheme through the Research Grants Council of the Hong Kong Special Administrative Region and France National Research Agency under Project A-CityU101/16 and in part by the National Science Foundation of China under Grant 61672443 and Grant 61473241.

M. Wu is with the Department of Computer Science, City University of Hong Kong, Hong Kong (e-mail: mengyuan.wu@my.cityu.edu.hk).

S. Kwong and Q. Zhang are with the Department of Computer Science, City University of Hong Kong, and also with City University of Hong Kong Shenzhen Research Institute, Shenzhen, China (e-mail: cssamk@cityu.edu.hk; qingfu.zhang@cityu.edu.hk).

K. Li is with the College of Engineering, Mathematics and Physical Sciences, University of Exeter, Exeter EX4 4QF, U.K. (e-mail: k.li@exeter.ac.uk).

Y. Zhou is with the Department of Computer Science, City University of Hong Kong, Hong Kong, and also with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China (e-mail: yzhou57-c@my.cityu.edu.hk).

This paper has supplementary downloadable multimedia material available at http://ieeexplore.ieee.org provided by the authors.

Digital Object Identifier 10.1109/TEVC.2017.2656922

as discussed in [12] and [13], partially due to the overrated convergence property, both original MOEA/D and stable matching-based selection mechanism fail to maintain the population diversity when solving problems with complicated properties, e.g., imbalanced problem [13], [14] and many objectives. Bearing these considerations in mind, Li *et al.* [12] modified the mutual preference definition and developed a straightforward but more effective selection mechanism based on the interrelationship between subproblems and solutions. Later on, Wang *et al.* [15] proposed an adaptive replacement strategy, which adjusts the replacement neighborhood size dynamically, to assign solutions to their most suitable subproblems. It is also interesting to note that some works took the advantages of the Pareto dominance- and decomposition-based selection mechanisms in a single paradigm [16]–[18].

To achieve a good balance between convergence and diversity, this paper suggests to introduce the concept of incomplete preference lists into the stable matching model. Specifically, borrowing the idea from the stable matching with incomplete preference lists [19], we restrict the number of subproblems with which a solution is allowed to match. In this case, a solution can only be assigned to one of its favorite subproblems. However, due to the restriction on the preference list, the stable marriage model, which results in a one-one matching, may leave some subproblems unmatched. To remedy this situation, this paper implements two different versions of stable matching-based selection mechanisms with incomplete preference lists.

1) The first one achieves a two-level one-one matching. At the first level, we find the stable solutions for subproblems according to the incomplete preference lists. Afterwards, at the second level, the remaining unmatched subproblems are matched with suitable solutions according to the remaining preference information.
2) The second one obtains a many-one matching. In such a way, the unmatched subproblems give the matching opportunities to other subproblems that have already matched with a solution but still have openings.

Note that the length of the incomplete preference list has a significant impact on the performance and is problem dependent [20]. By analyzing the underlying mechanism of the proposed stable matching-based selection mechanisms in depth, we develop an adaptive mechanism to set the length of the incomplete list for each solution on the fly. Comprehensive experiments on 62 benchmark problems fully demonstrate the effectiveness and competitiveness of our proposed methods.

The rest of this paper is organized as follows. Section II introduces some preliminaries of this paper. Thereafter, the proposed algorithm is described step by step in Section III. Sections IV and V provide the experimental settings and the analysis of the empirical results. Finally, Section VI concludes this paper and provides some future directions.

II. Preliminaries

In this section, we first introduce some background knowledge of MOEA/D and the stable matching-based selection. Then, our motivations are developed by analyzing their underlying mechanisms and drawbacks.

A. MOEA/D

As a representative of the decomposition-based algorithms, MOEA/D has become an increasingly popular choice for *posterior* multiobjective optimization. Generally speaking, there are two basic components in MOEA/D: one is *decomposition* and the other is *collaboration*. The following paragraphs give some general descriptions of each component separately.

1) Decomposition: The basic idea of decomposition is transforming the original MOP into a single-objective optimization subproblem. There are many established decomposition methods developed for classic multiobjective optimization [21], among which the most popular ones are weighted sum, Tchebycheff (TCH) and boundary intersection approaches. Without loss of generality, this paper considers the inverted TCH approach [10], which is defined as follows:

$$\begin{aligned} &\text{minimize} \quad g^{\text{tch}}(\mathbf{x}|\mathbf{w}, \mathbf{z}^*) = \max_{1\le i\le m} \left\{|f_i(\mathbf{x}) - z_i^*|/w_i\right\} \\ &\text{subject to} \quad \mathbf{x} \in \Omega \end{aligned} \tag{2}$$

where $\mathbf{w} = (w_1, \ldots, w_m)^T$ is a user specified weight vector, $w_i \ge 0$ for all $i \in \{1, \ldots, m\}$ and $\sum_{i=1}^m w_i = 1$. In practice, w_i is set to be a very small number, say 10^{-6}, when $w_i = 0$. $\mathbf{z}^* = (z_1^*, \ldots, z_m^*)^T$ is an Utopian objective vector where $z_i^* = \min_{\mathbf{x}\in\Omega} f_i(x)$, $i \in \{1, \ldots, m\}$. Note that the search direction of the inverted TCH approach is $\mathbf{w}$, and the optimal solution of (2) is a Pareto-optimal solution of the MOP defined in (1) under some mild conditions. We can expect to obtain various Pareto-optimal solutions by using (2) with different weight vectors. In MOEA/D, a set of uniformly distributed weight vectors are sampled from a unit simplex.

2) Collaboration: As discussed in [8], the neighboring subproblems, associated with the geometrically close weight vectors, tend to share similar optima. In other words, the optimal solution of $g^{\text{tch}}(\cdot|\mathbf{w}^1, \mathbf{z}^*)$ is close to that of $g^{\text{tch}}(\cdot|\mathbf{w}^2, \mathbf{z}^*)$, given $\mathbf{w}^1$ and $\mathbf{w}^2$ are close to each other. In MOEA/D, each solution is associated with a subproblem. During the optimization process, the solutions cooperate with each other via a well-defined neighborhood structure and they solve the subproblems in a collaborative manner. In practice, the collaboration is implemented as a restriction on the mating and update procedures. More specifically, the mating parents are selected from neighboring subproblems and a newly generated offspring is only used to update its corresponding neighborhood. Furthermore, since different subproblems might have various difficulties, it is more reasonable to dynamically allocate the computational resources to different subproblems than treating all subproblems equally important. In [22], a dynamic resource allocation scheme is developed to allocate more computational resources to those promising ones according to their online performance.

B. Stable Matching-Based Selection

Stable marriage problem (SMP) was originally introduced in [23] and its related work won the 2012 Nobel Prize in Economics. In a nutshell, the SMP is about how to establish a stable one-one matching between two sets of agents, say men and women, which have mutual preferences over each other. A stable matching should not contain a man and a woman

who are not matched together but prefer each other to their assigned spouses.

In MOEA/D, subproblems and solutions can be treated as two sets of agents which have mutual preferences over each other. In particular, a subproblem prefers a solution that optimizes its underlying single-objective optimization problem as much as possible; while a solution prefers to have a well distribution in the objective space. The ultimate goal of selection is to select the best solution for each subproblem, and vice versa. In this case, we can treat the selection procedure as a one-one matching procedure between subproblems and solutions. To the best of our knowledge, MOEA/D-STM [10] is the first one that has modeled the selection procedure of MOEA/D as an SMP, and encouraging results have been reported therein. The framework of the stable matching-based selection contains two basic components, i.e., *preference settings* and *matching model*. The following paragraphs briefly describe these two components.

1) Preference Settings: The preference of a subproblem p on a solution $\mathbf{x}$ is defined as

$$\Delta_P(p, \mathbf{x}) = g^{\text{tch}}\left(\mathbf{x}|\mathbf{w}, \mathbf{z}^*\right) \tag{3}$$

where $\mathbf{w}$ is the weight vector of p. Consequently, $\Delta_P(p, \mathbf{x})$ measures the convergence of $\mathbf{x}$ with respect to p. The preference of a solution $\mathbf{x}$ on a subproblem p is defined as

$$\Delta_X(\mathbf{x}, p) = \left\| \overline{F}(\mathbf{x}) - \frac{\mathbf{w}^{\mathrm{T}} \cdot \overline{F}(\mathbf{x})}{\mathbf{w}^{\mathrm{T}} \cdot \mathbf{w}} \mathbf{w} \right\| \tag{4}$$

where $\overline{F}(\mathbf{x})$ is the normalized objective vector of $\mathbf{x}$ and $\|\cdot\|$ is the ℓ_2-norm. Since the weight vectors are usually uniformly distributed, it is desirable that the optimal solution of each subproblem has the shortest perpendicular distance to its corresponding weight vector. For the sake of simplicity, $\Delta_X(\mathbf{x}, p)$ can be used to measure the diversity of a solution [10].

2) Matching Model: Based on the above preference settings, [10] employed the classic deferred acceptance procedure (DAP) developed in [23] to find a stable matching between subproblems and solutions. The pseudo code of this stable matching-based selection mechanism is given in Algorithm 1 and the DAP function is presented in Algorithm 2. Ψ_P and Ψ_X are the preference matrices of subproblems and solutions, each row of which represents the preference list of a subproblem over all solutions, and vice versa. In particular, a preference list is built by sorting the preference values in an ascending order. M indicates the set of all the constructed matching pairs. It is worth noting that the convergence and diversity have been aggregated into the preference settings, thus the stable matching between subproblems and solutions strikes the balance between convergence and diversity.

Algorithm 1: STM(P, S, Ψ_P, Ψ_X)

Input:
- subproblem set P and solution set S
- sets of preference lists Ψ_P and Ψ_X

Output: stable matching set M

```
P_u ← P, M ← ∅;
while P_u ≠ ∅ do
    p ← Randomly pick a subproblem from P_u;
    x ← First solution on p's preference list;
    Remove x from p's preference list;
    M ← DAP(p, x, P_u, M, Ψ_P, Ψ_X);
return M
```

Algorithm 2: DAP(p, $\mathbf{x}$, P_u, M, Ψ_P, Ψ_X)

Input:
- current subproblem p and solution $\mathbf{x}$
- unmatched subproblem set P_u
- current stable matching set M
- sets of preference lists Ψ_P and Ψ_X

Output: stable matching set M

```
if x ∉ M then
    M ← M ∪ (p, x);            // match p and x
    P_u ← P_u \ p;
else
    p' ← M(x);                 // current partner of x
    if x prefers p to p' then
        M ← M ∪ (p, x) \ (p', x);
        P_u ← P_u ∪ p' \ p;
return M
```

C. Drawbacks of MOEA/D and MOEA/D-STM

In this section, we discuss some drawbacks of the selection mechanisms of MOEA/D and MOEA/D-STM.

1) MOEA/D: The update mechanism of the original MOEA/D is simple and efficient, yet greedy. In a nutshell, each subproblem simply selects its best solution according to the corresponding scalar optimization function value. As discussed in [22], since different parts of the PF might have various difficulties, some subproblems might be easier than the others for finding the optimal solutions. During some intermediate stages of the optimization process, the currently elite solutions of some relatively easier subproblems might also be good candidates for the others. In this case, these elite solutions can easily take over all subproblems. In addition, it is highly likely that the offspring solutions generated from these elite solutions crowd into the neighboring areas of the corresponding subproblems. Therefore, this purely fitness-driven selection mechanism can be severely harmful for the population diversity and may lead to the failure of MOEA/D on some challenging problems [13]. Let us consider an example shown in Fig. 1(a), where five out of ten solutions need to be selected for five subproblems. Since $\mathbf{x}^1$ is currently the best solution for $\{p^1, p^2, p^3\}$ and $\mathbf{x}^{10}$ is the current best candidate for $\{p^4, p^5\}$, these two elite solutions finally take over all five subproblems. Obviously, the population diversity of this selection result is not satisfied.

2) MOEA/D-STM: As discussed in [24], the DAP maximizes the satisfactions of the preferences of men and women in order to maintain the stable matching relationship. According to the preference settings for subproblems, solutions closer to

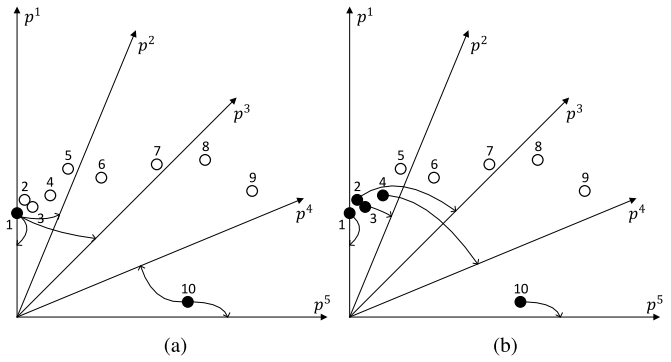


Fig. 1. Comparisons of (a) MOEA/D and (b) MOEA/D-STM.

the PF are always on the front of the subproblems' preference lists. In this case, the DAP might make some solutions match themselves with subprobolems lying on the rear of their preference lists. Even worse, as discussed in Section II-C1, these currently well converged solutions may crowd in a narrow area. This obviously goes against the population diversity. Let us consider the same example discussed in Fig. 1(a). The preference matrices of subproblems and solutions are

$$\Psi_P = \begin{array}{l} p^1: [\ 1\ \ 2\ \ 3\ \ 4\ \ 5\ \ 6\ \ 7\ \ 8\ \ 10\ \ 9\ ] \\ p^2: [\ 1\ \ 3\ \ 2\ \ 4\ \ 5\ \ 6\ \ 7\ \ 8\ \ 10\ \ 9\ ] \\ p^3: [\ 1\ \ 3\ \ 2\ \ 4\ \ 6\ \ 5\ \ 7\ \ 10\ \ 8\ \ 9\ ] \\ p^4: [\ 10\ \ 1\ \ 3\ \ 2\ \ 4\ \ 9\ \ 6\ \ 5\ \ 7\ \ 8\ ] \\ p^5: [\ 10\ \ 1\ \ 3\ \ 2\ \ 4\ \ 9\ \ 6\ \ 5\ \ 7\ \ 8\ ] \end{array} \tag{5}$$

$$\Psi_X = \begin{array}{l} \mathbf{x}^1: [\ 1\ \ 2\ \ 3\ \ 4\ \ 5\ ] \\ \mathbf{x}^2: [\ 1\ \ 2\ \ 3\ \ 4\ \ 5\ ] \\ \mathbf{x}^3: [\ 1\ \ 2\ \ 3\ \ 4\ \ 5\ ] \\ \mathbf{x}^4: [\ 2\ \ 1\ \ 3\ \ 4\ \ 5\ ] \\ \mathbf{x}^5: [\ 2\ \ 1\ \ 3\ \ 4\ \ 5\ ] \\ \mathbf{x}^6: [\ 2\ \ 3\ \ 1\ \ 4\ \ 5\ ] \\ \mathbf{x}^7: [\ 3\ \ 2\ \ 4\ \ 1\ \ 5\ ] \\ \mathbf{x}^8: [\ 3\ \ 4\ \ 2\ \ 5\ \ 1\ ] \\ \mathbf{x}^9: [\ 4\ \ 3\ \ 5\ \ 2\ \ 1\ ] \\ \mathbf{x}^{10}: [\ 5\ \ 4\ \ 3\ \ 2\ \ 1\ ] \end{array}. \tag{6}$$

From (5), we can clearly see that $\mathbf{x}^1$ to $\mathbf{x}^4$ dominate the top positions of the preference lists of all subproblems. By using Algorithm 1, we have the selection/matching result shown in Fig. 1(b), where $\mathbf{x}^1$ to $\mathbf{x}^4$ crowd in a narrow area between p^1 and p^2. This is obviously harmful for the population diversity as well.

From the above discussions, we find that the original selection mechanism of MOEA/D is a *convergence first and diversity second* strategy [13], which might give excessive priority to the convergence requirement. On the other hand, although the stable matching-based selection mechanism intends to achieve an equilibrium between convergence and diversity, the stable matching between subproblems and solutions may fail to keep the population diversity. This is because no restriction has been given to the subproblem with which a solution can match. In other words, a solution can match with an unfavorable subproblem in the resulting stable matching. To relieve this side effect, the next section suggests a strategy to take advantages of some partial information from the preference lists when finding the stable matching between subproblems and solutions.

Algorithm 3: STMIC(P, S, Ψ_P, Ψ_X, R)

Input:
- subproblem set P, solution set S
- preference matrices Ψ_P and Ψ_X
- length of solution's preference list set R

Output: stable matching set M

1 $P_u \leftarrow P$, $M \leftarrow \emptyset$;
2 **for** $i \leftarrow 1$ to $|S|$ **do**
3 Keep the first r^i subproblems on $\mathbf{x}^i$'s complete preference list and remove the remainders;
4 **while** $P_u \neq \emptyset$ **do**
5 $p \leftarrow$ Randomly select a subproblem from P_u;
6 **if** *p's preference list* $\neq \emptyset$ **then**
7 $\mathbf{x} \leftarrow$ First solution on p's preference list;
8 Remove $\mathbf{x}$ from p's preference list;
9 **if** *p is on* $\mathbf{x}$*'s preference list* **then**
10 $M \leftarrow$ DAP(p, $\mathbf{x}$, P_u, M, Ψ_P, Ψ_X);
11 **else**
12 $P_u \leftarrow P_u \setminus p$;
13 **Return** M;

III. Adaptive Stable Matching-Based Selection With Incomplete Preference Lists

In the canonical SMP, each man/woman holds a complete and strictly ordered preference list over all agents from the other side. However, in practice, it may happen that a man/woman declares some unacceptable partners [24], and this results in an SMP with incomplete lists [25]. By these means, a man/woman is only allowed to match with a matching agent that appears on his/her incomplete preference list. Due to the restriction from the incomplete preference lists, there is no guarantee that all agents can have a stable matching mate. A stable matching for an SMP with incomplete lists does not contain such a pair of man and woman: 1) they are acceptable to each other but not matched together and 2) they either do not match with anyone else or prefer each other to their current matching mates. To overcome the drawbacks discussed in Section II-C, here we implement two versions of stable matching-based selection mechanisms with incomplete preference lists: one achieves a two-level one-one matching while the other obtains a many-one matching.

A. Two-Level One-One Stable Matching-Based Selection

In the first level, let us assume that there are N subproblems and Q solutions, where $N < Q$. After obtaining the complete preference lists of all subproblems and solutions (lines 1 and 2 of Algorithm 4), we only keep the first r^i, where $i \in \{1, \ldots, Q\}$ and $0 < r^i \leq N$, subproblems on the preference list of each solution $\mathbf{x}^i$, while the remaining ones are not considered any

Algorithm 4: SelectionOOSTM2L(P, S)

Input: subproblem set P and solution set S
Output: solution set S
/* First-level stable matching */
1 Compute Ψ_p and $\Psi_{\mathbf{x}}$ for P and S;
2 $R \leftarrow$ Set the length of each solution's preference list;
3 $M \leftarrow \mathsf{STMIC}(P, S, \Psi_P, \Psi_X, R)$;
/* Second-level stable matching */
4 $(P_m, S_m) \leftarrow M$;
5 $P_u \leftarrow P \setminus P_m$;
6 $S_u \leftarrow S \setminus S_m$;
7 Compute Ψ'_P and Ψ'_X for P_u and S_u;
8 $M' \leftarrow \mathsf{STM}(P_u, S_u, \Psi'_P, \Psi'_X)$;
/* Combine the stable matching pairs */
9 $M \leftarrow M \cup M'$;
10 **Return** M;

longer (lines 2 and 3 of Algorithm 3). In this case, each solution is only allowed to match with its first several favorite subproblems which are close to itself according to (4). In contrast, the preference lists of subproblems are kept unchanged. Given the incomplete preference information, we employ the DAP to find a stable matching between subproblems and solutions (lines 4–12 of Algorithm 3). By these means, we can expect that the population diversity is strengthened during the first-level stable matching. This is because a solution is not allowed to match with an unfavorable subproblem which lies out of its incomplete preference list. The pseudo code of the stable matching with incomplete lists is given in Algorithm 3.

During the first-level stable matching, not all subproblems are assigned with a stable solution due to the incomplete preference information. To remedy this issue, the second-level stable matching with complete preference lists is developed to find a stable solution for each unmatched subproblem. At first, we compute the preference matrices of the unmatched subproblems and solutions (line 7 of Algorithm 4). Afterwards, we employ Algorithm 1 to find a stable matching between them (line 8 of Algorithm 4). In the end, the matching pairs of both levels of stable matching are gathered together to form the final selection results (line 9 of Algorithm 4). The pseudo code of the two-level stable matching-based selection mechanism is given in Algorithm 4.

B. Many-One Stable Matching-Based Selection

Many-one stable matching problem is an extension of the standard SMP, where a matching agent from one side is allowed to have more than one matching mates from the other side. For example, in the college admission problem (CAP) [23], the colleges and applicants are two sets of matching agents. Each college has a preference list over all applicants and vice versa. Different from the SMP, each applicant is only allowed to enter one college, whereas each college has a positive integer quota being the maximum number of applicants that it can admit.

As the other implementation of the stable matching-based selection with incomplete preference lists, here we model the

Algorithm 5: SelectionMOSTM(P, S)

Input: subproblem set P and solution set S
Output: stable matching set M
1 Compute Ψ_P and Ψ_X for P and S;
2 $R \leftarrow$ Set the length of each solution's preference list;
3 $S_u \leftarrow S$, $M \leftarrow \emptyset$;
4 **for** $i \leftarrow 1$ to Q **do**
5 Keep the first r^i subproblems on $\mathbf{x}^i$'s complete preference list and remove the remainders;
6 **while** $S_u \neq \emptyset$ **do**
7 $\mathbf{x} \leftarrow$ Randomly select a solution from S_u;
8 **if** $\mathbf{x}$*'s preference list* $\neq \emptyset$ **then**
9 $p \leftarrow$ First subproblem on $\mathbf{x}$'s preference list;
10 Remove p from $\mathbf{x}$'s preference list;
11 $M \leftarrow M \cup (p, \mathbf{x})$;
12 **if** $|M| > N$ **then**
13 $\overline{P} \leftarrow \arg\max_{p \in P} |M(p)|$; // $|M(p)|$ is the cardinality of $M(p)$
14 $\overline{P} \leftarrow \arg\max_{p \in \overline{P}} \{\max_{\mathbf{x} \in M(p)} rank(p, \mathbf{x})\}$; // $rank(p, \mathbf{x})$ is the rank of $\mathbf{x}$ on p's preference list
15 $p' \leftarrow$ Randomly select a subproblem from $\overline{P}$;
16 $\mathbf{x}' \leftarrow \arg\max_{\mathbf{x} \in M(p')} rank(p', \mathbf{x})$;
17 $M \leftarrow M \setminus (p', \mathbf{x}')$;
18 $S_u \leftarrow S_u \cup \mathbf{x}'$;
19 **else**
20 $S_u \leftarrow S_u \setminus \mathbf{x}$;
21 **return** M;

selection process of MOEA/D as a CAP with a common quota [26]. More specifically, subproblems and solutions are treated as colleges and applicants, respectively. A solution is only allowed to match with one subproblem while a subproblem is able to match with more than one solution. In particular, we do not limit the separate quota for every subproblem but assign a common quota for all subproblems, which equals the number of subproblems (i.e., N). In other words, N subproblems can at most match with N solutions in this many-one matching. Note that a matching is stable if there does not exist any pair of subproblem p and solution $\mathbf{x}$ where:

1) p and $\mathbf{x}$ are acceptable to each other but not matched together;
2) $\mathbf{x}$ is unmatched or prefers p to its assigned subproblem;
3) the common quota is not met or p prefers $\mathbf{x}$ to at least one of its assigned solutions.

The pseudo code of the many-one stable matching-based selection mechanism is given in Algorithm 5. The initialization process (lines 1–5 of Algorithm 5) is the same as the one-one stable matching discussed in Section III-A. During the main while-loop, an unmatched solution $\mathbf{x} \in S_u$ at first matches with its current favorite subproblem p according to

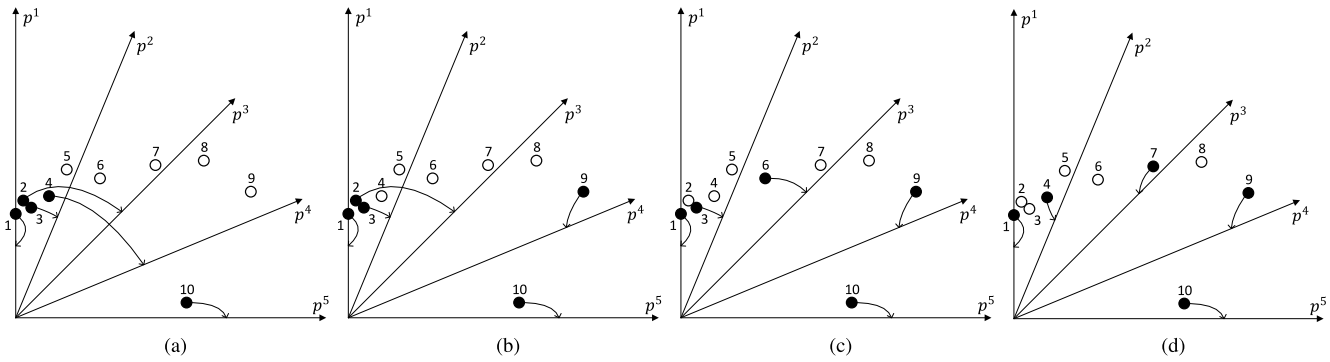


Fig. 2. Comparisons of the two-level stable matching-based selection with incomplete lists using different r settings. (a) STM2L ($r = 4$). (b) STM2L ($r = 3$). (c) STM2L ($r = 2$). (d) STM2L ($r = 1$).

its preference list (lines 7–11 of Algorithm 5). If the number of current matching pairs $|M|$ is larger than N, we find a substitute subproblem p' and adjust its matching pairs by releasing the matching relationship with its least preferred solution $\mathbf{x}'$ (lines 12–18 of Algorithm 5). In particular, p' is selected according to the following criteria.

1) At first, we choose the subproblems that have the largest number of matched solutions to form $\overline{P}$ (line 13 of Algorithm 5). Its underlying motivation is to reduce the chance for overly exploiting a particular subproblem.
2) If the cardinality of $\overline{P}$ is greater than one, we need to further process $\overline{P}$. Specifically, we investigate the ranks of the solutions matched with subproblems in $\overline{P}$. The subproblems, whose least preferred solution holds the worst rank on that subproblem's preference list, are used to reconstruct $\overline{P}$ (line 14 of Algorithm 5).
3) In the end, p' is randomly chosen from $\overline{P}$ (line 14 of Algorithm 5).

Note that we add $\mathbf{x}'$ back into S_u after releasing its matching relationship with p' (line 18 of Algorithm 5). The matching process terminates when S_u becomes empty.

C. Impacts of the Length of the Incomplete Preference List

As discussed in the previous sections, we expect to improve the population diversity by restricting the length of the preference list of each solution. A natural question is whether this length affects the behavior of our proposed stable matching-based selection mechanisms? Let us consider the example discussed in Fig. 1 again. For the sake of discussion, here we set the length of the incomplete preference list of each solution as a constant (denoted by r). By using different settings of r, Fig. 2 shows the selection results of the two-level stable matching-based selection mechanism. From this figure, we find that the diversity of the selected solutions increases with the decrease of r; on the other hand, the improvement of the diversity is at the expense of the convergence. It is interesting to note that the two-level stable matching-based selection mechanism totally degenerates into the original stable matching-based selection mechanism shown in Fig. 1(a) when using $r = 4$. In a word, r controls the tradeoff between convergence and diversity in the stable matching-based selection with incomplete preference lists. In the next section, we develop an adaptive mechanism to control the length of each solution's preference list on the fly.

D. Adaptive Mechanism

To better understand the proposed adaptive mechanism, here we introduce the concept of *local competitiveness*. At first, all solutions are associated with their closest subproblems having the shortest perpendicular distance between the objective vector of the solution and the weight vector of the subproblem. Afterwards, for each subproblem having more than one associated solutions, we choose the one, which has the best aggregation function value, as its representative solution. A solution is defined as a *locally competitive* solution in case it dominates at least one representative solution of its $\ell \geq 1$ nearest subproblems; otherwise, it is defined as a *locally noncompetitive* solution. In view of the population dynamics of the evolutionary process, we develop an adaptive mechanism to set the length of the incomplete preference list of a solution according to its local competitiveness (Algorithm 6 gives its pseudo code). Briefly speaking, this length is set as the maximum ℓ that keeps the corresponding solution locally noncompetitive.

More specifically, given N subproblems and Q solutions, each solution is associated with its closest subproblem as shown in lines 1 and 2 of Algorithm 6. In particular, $\Phi[i]$ represents the index of the subproblem with which a solution $\mathbf{x}^i$ is associated, $i \in \{1, \ldots, Q\}$. In line 4 of Algorithm 6, we collect the associated solutions of each subproblem p^j, $j \in \{1, \ldots, N\}$, to form a temporary set χ. Then, lines 5–8 of Algorithm 6 determine the representative solution of each subproblem p^j, where $\varphi[j]$ represents the index of its representative solution. Afterwards, for each solution $\mathbf{x}^i$, lines 10–16 of Algorithm 6 gradually increase ℓ until $\mathbf{x}^i$ becomes locally competitive, and this final ℓ is used as the length of $\mathbf{x}^i$'s incomplete preference list. Note that since each solution locates within the subspace between m closest neighboring weight vectors in m-dimensional objective space, it can be associated with any of these m subproblems in principle. Moreover, to avoid unnecessary comparisons, it is desirable to keep the solution's incomplete preference list within a reasonably small length. All in all, the length of $\mathbf{x}^i$'s incomplete preference list is adaptively tuned between m and $\ell_{\max}$. In particular, $\ell_{\max}$ is

Algorithm 6: AdaptiveSetR(P, S, Ψ_X)

Input:
- subproblem set P and solution set S
- solution preference matrix Ψ_X

Output: length of solution's preference list set R

```
1  for i ← 1 to Q do
2      Φ[i] ← Ψ_X[i][1];
3  for j ← 1 to N do
4      χ ← {i|Φ[i] = j,  i ∈ 1, 2, ..., Q};
5      if χ = ∅ then
6          φ[j] ← −1;
7      else
8          φ[j] ← arg min_{i∈χ} g^tch(x^i|λ^j, z*);
9  for i ← 1 to Q do
10     r^i ← m;
11     for ℓ ← m + 1 to ℓ_max do
12         t ← φ[Ψ_X[i][ℓ]];
13         if t ≠ −1 then
14             if x^i ≺ x^t then
15                 break;
16         r^i ← ℓ;
17 return R;
```

set as the neighborhood size T used in MOEA/D, where the mating parents are selected from.

Let us use the example shown in Fig. 1(b) to explain the underlying principle of our proposed adaptive mechanism. In this example, solutions $\mathbf{x}^2$ and $\mathbf{x}^3$ become locally competitive when $\ell > 1$; while solutions $\mathbf{x}^7$ and $\mathbf{x}^9$ are locally noncompetitive for all ℓ settings. It is worth noting that neither $\mathbf{x}^2$ nor $\mathbf{x}^3$ is the representative solution of any subproblem; in the meanwhile, they are crowded in a narrow area. Since these locally competitive solutions have better ranks in the preference lists than those less competitive ones, the original stable matching-based selection tends to give them higher priorities to form the matching pairs. However, this selection result is obviously harmful for the population diversity. In addition, we also notice that $\mathbf{x}^7$ and $\mathbf{x}^9$ are the representative solutions of p^3 and p^4, thus they should contain some relevant information for optimizing these subproblems. In contrast, although $\mathbf{x}^2$ and $\mathbf{x}^3$ have better aggregation function values, they are far away from p^3 and p^4 and should be less relevant to them. To resolve these issues, our proposed adaptive mechanism adaptively restricts the length of the preference list of each solution $\mathbf{x}$ by removing subproblems whose representative solution is dominated by $\mathbf{x}$. By these means, we can make sure that each solution does not consider a subproblem which prefers this solution to its own representative solution. Thus each subproblem is prevented from matching with a less relevant solution. Note that this adaptive mechanism can be readily plugged into both of our proposed two versions of stable matching-based selection mechanisms by using Algorithm 6 to replace line 2 of Algorithms 4 and 5, respectively. The adaptive two-level one-one stable matching-based selection mechanism and the adaptive many-one stable matching-based selection mechanism are denoted by AOOSTM and AMOSTM for short.

E. Time Complexity of AOOSTM and AMOSTM

In this section, we analyze the complexity of AOOSTM and AMOSTM. For both selection mechanisms, the calculation of $\Delta_P(p, \mathbf{x})$ and $\Delta_X(\mathbf{x}, p)$ cost $\mathcal{O}(NQ \log Q)$ computations [10]. In Algorithm 6, the association operation between subproblems and solutions costs $\mathcal{O}(Q)$ calculations (lines 1 and 2). As for lines 3–8 of Algorithm 6, the identification of the representative solution for each subproblem requires $\mathcal{O}(mNQ)$ computations. Thereafter, the computation of R in lines 9–16 of Algorithm 6 costs $\mathcal{O}(mQ(\ell_{\max} - m))$ computations in the worst case. Considering the two-level one-one stable matching in Algorithm 4, the complexity of the one-one stable matching with the incomplete lists in line 3 is $\mathcal{O}(N\ell_{\max})$, which is simpler than the original stable matching with complete preference lists [10]. Next, the complexity of lines 4–6 of Algorithm 4 is $\mathcal{O}(N + Q)$. During the second-level stable matching (lines 7 and 8 of Algorithm 4), same complexity analysis can be done for the remaining subproblems and solutions. Overall, the total complexity of AOOSTM is $\mathcal{O}(\max(NQ \log Q, mQ(\ell_{\max} - m)))$. When it comes to AMOSTM, since Algorithm 5 is solution-oriented, the computational complexity of lines 3–19 is $\mathcal{O}(Q\ell_{\max})$. The total complexity of AMOSTM is still $\mathcal{O}(\max(NQ \log Q, mQ(\ell_{\max} - m)))$.

F. Incorporation With MOEA/D

Similar to [10], we choose the MOEA/D-DRA [22] as the base framework and replace the update mechanism by the AOOSTM and AMOSTM selection mechanisms developed in Section III-D. The resulted algorithms are denoted by MOEA/D-AOOSTM and MOEA/D-AMOSTM, of which the pseudo code is given in Algorithm 7. Note that the normalization scheme proposed in [27] is adopted to handle MOPs with different scales of objectives. In the following paragraphs, some important components of MOEA/D-AOOSTM/AMOSTM are further illustrated.

1) Initialization: Without any prior knowledge of the landscape, the initial population $S = \{\mathbf{x}^1, \ldots, \mathbf{x}^N\}$ is randomly sampled from Ω. Same as the original MOEA/D, we use the classic method suggested in [28] to generate a set of uniformly distributed weight vectors $W = \{\mathbf{w}^1, \ldots, \mathbf{w}^N\}$ on a unit simplex. In addition, for each weight vector $\mathbf{w}^i$, $i \in \{1, \ldots, N\}$, we assign its T, $1 \leq T \leq N$, closest weight vectors as its neighbors.

2) Reproduction: According to the underlying test problem, here we employ the widely used differential evolution (DE) [29], simulated binary crossover (SBX) [30], and polynomial mutation [31] for offspring generation.

3) Utility of Subproblem [22]: The utility of subproblem p^i, denoted by π^i, $i \in \{1, \ldots, N\}$, measures the improvement rate of p^i. We make some modifications on π^i to fit our proposed

Algorithm 7: MOEA/D–AOOSTM/AMOSTM

Input: algorithm parameters
Output: final population S

Initialize the population S, a set of weight vectors W and their neighborhood structure B;
$M \leftarrow$ Random one-one matching between P and S;
$neval \leftarrow 0$, $iteration \leftarrow 0$;
while *Stopping criterion is not satisfied* **do**
  Select the current active subproblems to form I;
  for *each* $i \in I$ **do**
    if $uniform(0, 1) < \delta$ **and** $|E| >= T$ **then**
      $E \leftarrow \{M(p)|p \in B(i)\}$;
    else
      $E \leftarrow S$;
    Randomly select mating solutions from E and generate an offspring $\overline{\mathbf{x}}$, $S \leftarrow S \cup \overline{\mathbf{x}}$;
    Evaluate $\mathbf{F}(\overline{\mathbf{x}})$, $neval$++;
  $M \leftarrow$ SelectionOOSTM2L/MOSTM(P, S);
  $S \leftarrow \{M(p)|p \in P\}$;
  $iteration$++;
  if $mod(iteration, 30) = 0$ **then**
    Update the utility of each subproblem;
return S;

MOEA/D-AOOSTM/AMOSTM

$$\pi^i = \begin{cases} 1 & \text{if } \Delta^i > 0.001 \\ 0.95 \times \pi^i & \text{if } \Delta^i < 0 \\ \left(0.95 + 0.05 \times \frac{\Delta^i}{0.001}\right) \times \pi^i & \text{otherwise} \end{cases} \tag{7}$$

where Δ^i represents the relative decrease of the scalar objective value of p^i and is evaluated as

$$\Delta^i = \begin{cases} \frac{g^{\text{tch}}(\mathbf{x}^{i,\text{old}}|\mathbf{w}^i,\mathbf{z}^*) - g^{\text{tch}}(\mathbf{x}^{i,\text{best}}|\mathbf{w}^i,\mathbf{z}^*)}{g^{\text{tch}}(\mathbf{x}^{i,\text{old}}|\mathbf{w}^i,\mathbf{z}^*)} & \text{if } M(p^i) \neq \emptyset \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

where $\mathbf{x}^{i,\text{best}} = \arg\min_{\mathbf{x} \in M(p^i)} g^{\text{tch}}(\mathbf{x}|\mathbf{w}^i, \mathbf{z}^*)$ is the best solution matched with p^i in the current generation and $\mathbf{x}^{i,\text{old}}$ is the previously saved value of $\mathbf{x}^{i,\text{best}}$.

IV. Experimental Settings

This section presents the general setup of our empirical studies, including the benchmark problems, algorithms in comparisons, parameter settings and performance metrics.

A. Benchmark Problems

Three popular benchmark suites, i.e., MOP [13], UF [32], and WFG [33], 62 problem instances in total, are chosen as the benchmark set in our empirical studies. These problem instances have various characteristics, e.g., nonconvexity, deceptive, and multimodality. According to the recommendations in the original references, the number of decision variables is set as: $n = 10$ for the MOP instances and $n = 30$ for the UF instances. As the WFG instances are scalable to any number of objectives, here we consider $m \in \{2, 3, 5, 8, 10\}$. In particular, when $m = 2$, $n = k + l$ [12], where the position-related variable $k = 2$ and the distance-related variable $l = 4$; while for $m \geq 3$, we use the recommended settings in [16] and [27], i.e., $k = 2 \times (m - 1)$ and $l = 20$.

TABLE I
Settings of Population Size

Benchmark Problem	m	Population Size
UF1 to UF7	2	600
UF8 to UF10	3	1,000
MOP1 to MOP5	2	100
MOP6 to MOP7	3	300
WFG1 to WFG9	2	250
WFG1 to WFG9	3	91
WFG1 to WFG9	5	210
WFG1 to WFG9	8	156
WFG1 to WFG9	10	275

B. Algorithms in Comparisons

Nine state-of-the-art EMO algorithms, i.e., MOEA/D-STM, MOEA/D-IR [12], gMOEA/D-AGR [15], MOEA/D-M2M [13], MOEA/D-DRA, HypE [7], NSGA-III [27], PICEA-g [34], and MOEA/DD [16], are considered in our empirical studies. In particular, the first seven algorithms are used for comparative studies on problems with complicated PSs; while the latter five are chosen to investigate the scalability on problems with more than three objectives. The characteristics of these algorithms are briefly described in the supplementary material.[1]

C. Parameter Settings

Referring to [10], [12], and [16], the settings of the population size N for different benchmark problems are shown in Table I. The stopping condition of each algorithm is the predefined number of function evaluations. In particular, it is set to 300 000 for the UF and MOP instances [10], and 25 000 for the bi-objective WFG instances [12]. As for the many-objective WFG instances, where $m \in \{3, 5, 8, 10\}$, the number of function evaluations is set as $400 \times N$, $750 \times N$, $1,500 \times N$ and $2,000 \times N$, respectively [16]. The parameters of our proposed MOEA/D-AOOSTM and MOEA/D-AMOSTM are set as follows.

1) *Reproduction Operators:* As for problems with complicated properties, we use the DE operator and polynomial mutation for offspring generation. As recommended in [12], we set CR = 1.0 and $F = 0.5$ for the UF and MOP instances; while CR = $F = 0.5$ for the bi-objective WFG instances. The mutation probability p_m is set to be $1/n$ and its distribution index η_m equals 20. For problems with more than three objectives, we use

[1] https://coda-group.github.io/publications/suppASTM.pdf

the SBX operator to replace the DE operator, where the crossover probability $p_c = 1$ and its distribution index $\eta_c = 30$ [27]. Note that other MOEA/D variants in our experimental studies share the same settings for the reproduction operators.

2) *Neighborhood Size:* $T = 20$ [10], [12].
3) *Probability to Select B in the Neighborhood:* $\delta = 0.9$ [22].

D. Performance Metrics

To assess the performance of different algorithms, we choose the following two widely used performance metrics.

1) *Inverted Generational Distance (IGD) [35]:* Given P^* as a set of points uniformly sampled along the PF and P as the set of solutions obtained from an EMO algorithm. The IGD value of P is calculated as

$$\mathrm{IGD}\big(P, P^*\big) = \frac{\sum_{\mathbf{z}\in P^*} \mathrm{dist}(\mathbf{z}, P)}{|P^*|} \tag{9}$$

where $\mathrm{dist}(\mathbf{z}, P)$ is the Euclidean distance of $\mathbf{z}$ to its nearest point in P.

2) *Hypervolume (HV) [36]:* Let $\mathbf{z}^r = (z_1^r, \ldots, z_m^r)^T$ be a point dominated by all the Pareto optimal objective vectors. The HV of P is defined as the volume of the objective space dominated by the solutions in P and bounded by $\mathbf{z}^r$

$$\mathrm{HV}(P) = \mathsf{VOL}\left(\bigcup_{\mathbf{z}\in P}\left[z_1, z_1^r\right] \times \cdots \times \left[z_m, z_m^r\right]\right) \tag{10}$$

where VOL indicates the Lebesgue measure.

Since the objective functions of WFG instances are in different scales, we normalize their PFs and the obtained solutions in the range of [0, 1] before calculating the performance metrics. In this case, we constantly set $\mathbf{z}^r = (1.2, \ldots, 1.2)^T$ in the HV calculation. Note that both IGD and HV can evaluate the convergence and diversity simultaneously. A smaller IGD value or a larger HV value indicates a better approximation to the PF. Each algorithm is independently run 51 times. The mean and standard deviation of the IGD and HV values are presented in the corresponding tables, where the ranks of each algorithms on each problems are also given by sorting the mean metric values. The best metric values are highlighted in boldface with a gray background. To have a statistically sound conclusion, we use the Wilcoxon's rank sum test at a significant level of 5% to evaluate whether the proposed MOEA/D-AOOSTM and MOEA/D-AMOSTM are significantly better or worse than the others. In addition, we use the two-sample Kolmogorov-Smirnov test at a significant level of 5% to summarize the relative performance of all considered EMO algorithms.

V. Empirical Studies

In this section, we first analyze the comparative results for problems with complicated properties. Afterwards, we investigate the effectiveness of the adaptive mechanism. In the end, we summarize the experimental studies in a statistical point of view. Due to the page limits, the empirical studies on problems with more than three objectives are given in the supplementary material.

A. Performance Comparisons on MOP Instances

As discussed in [13], MOP benchmark suite, in which different parts of the PF have various difficulties, poses significant challenges for maintaining the population diversity. Tables II and III demonstrate the IGD and HV results of the nine EMO algorithms. From the IGD results shown in Table II, it can be seen that MOEA/D-AMOSTM shows the best overall performance and MOEA/D-AOOSTM, obtaining a slightly lower total rank than MOEA/D-M2M, ranks in the third place. In terms of the mean IGD values, MOEA/D-M2M gives the best results on MOP1–MOP4, while MOEA/D-AOOSTM ranks the first on MOP5 and MOEA/D-AMOSTM beats all other EMO algorithms on MOP6 and MOP7. When it comes to the Wilcoxon's rank sum test results, both MOEA/D-AMOSTM and MOEA/D-AOOSTM are significantly better the others on MOP2, MOP3, and MOP5–MOP7. They are only beaten by MOEA/D-M2M on MOP1 and MOP4. This is because MOEA/D-AMOSTM and MOEA/D-AOOSTM achieve better performance on MOP2 and MOP3 than MOEA/D-M2M but the former two have large variances. Comparing MOEA/D-AOOSTM and MOEA/D-AMOSTM, they have no significant differences on five problems but the former is outperformed by the latter on MOP4 and MOP7. Following the best three algorithms, MOEA/D-IR and gMOEA/D-AGR are able to obtain a set of non-dominated solutions moderately covering the entire PF. As for MOEA/D-DRA, MOEA/D-STM, NSGA-III, and HypE, they can only obtain some solutions lying on the boundaries. Table III shows similar results in HV tests, except that MOEA/D-AMOSTM obtains better performance than MOEA/D-AOOSTM on MOP5.

We plot the final solution sets with the best IGD values among 51 runs on all test instances in the supplementary material. From Figs. 1–4 of the supplementary material, we can see that although MOEA/D-M2M obtains slightly better mean IGD and HV metric values than MOEA/D-AMOSTM and MOEA/D-AOOSTM, the solutions obtained by MOEA/D-AMOSTM and MOEA/D-AOOSTM have a more uniform distribution along the PF. This can be explained by the density estimation method, i.e., the crowding distance of NSGA-II, used in MOEA/D-M2M, which is too coarse to guarantee the population diversity. Nevertheless, the convergence ability of MOEA/D-M2M is satisfied, thus contributing to promising IGD values on MOP1 to MOP4. According to [15], gMOEA/D-AGR uses a sigmoid function to assign a same replacement neighborhood size to all subproblems. However, since different parts of the PF require various efforts, this same setting might not be appropriate for all subproblems. From Figs. 3 and 7 of the supplementary material, we can obverse that the solutions obtained by gMOEA/D-AGR may miss some segments of the PF. This can be explained by the replacement neighborhood that grows too fast for the corresponding subproblems. In order to emphasize the population diversity, for each subproblem, MOEA/D-IR selects the appropriate solution

TABLE II
IGD RESULTS ON MOP TEST INSTANCES

Problem	IGD	DRA	STM	IR	AGR	M2M	NSGA-III	HypE	AOOSTM	AMOSTM
MOP1	Mean	3.380E-1	3.509E-1	4.726E-2	3.189E-2	**1.614E-2**	3.652E-1	8.013E-1	2.407E-2	2.390E-2
	Std	5.908E-2	2.786E-2	2.811E-3	9.792E-3	**4.586E-4**	3.337E-3	1.060E-2	2.907E-3	2.551E-3
	Rank	$6-\downarrow$	$7-\downarrow$	$5-\downarrow$	$4-\downarrow$	$1+\uparrow$	$8-\downarrow$	$9-\downarrow$	$3\parallel$	2
MOP2	Mean	2.836E-1	3.083E-1	3.200E-2	6.846E-2	**1.061E-2**	3.436E-1	5.980E-1	2.034E-2	3.115E-2
	Std	7.028E-2	6.782E-2	2.798E-2	7.344E-2	**1.578E-3**	1.478E-2	2.155E-1	4.301E-2	6.203E-2
	Rank	$6-\downarrow$	$7-\downarrow$	$4-\downarrow$	$5-\downarrow$	$1-\downarrow$	$8-\downarrow$	$9-\downarrow$	$2\parallel$	3
MOP3	Mean	4.927E-1	4.913E-1	4.267E-2	6.785E-2	**1.269E-2**	3.869E-1	6.094E-1	4.140E-2	3.203E-2
	Std	2.885E-2	3.391E-2	3.691E-2	8.518E-2	**3.924E-3**	1.337E-16	1.742E-1	7.378E-2	6.527E-2
	Rank	$8-\downarrow$	$7-\downarrow$	$4-\downarrow$	$5-\downarrow$	$1-\downarrow$	$6-\downarrow$	$9-\downarrow$	$3\parallel$	2
MOP4	Mean	3.068E-1	3.136E-1	3.843E-2	3.934E-2	**7.774E-3**	3.147E-1	7.107E-1	2.025E-2	1.414E-2
	Std	2.749E-2	1.840E-2	2.928E-2	4.065E-2	**7.983E-4**	1.845E-2	1.041E-2	3.284E-2	1.155E-2
	Rank	$6-\downarrow$	$7-\downarrow$	$4-\downarrow$	$5-\downarrow$	$1+\uparrow$	$8-\downarrow$	$9-\downarrow$	$3\downarrow$	2
MOP5	Mean	3.168E-1	3.135E-1	5.573E-2	2.379E-2	2.195E-2	2.911E-1	1.023E+0	**2.035E-2**	2.042E-2
	Std	7.241E-3	1.268E-2	2.524E-3	3.323E-3	2.489E-3	2.422E-2	2.343E-1	**1.692E-3**	1.803E-3
	Rank	$8-\downarrow$	$7-\downarrow$	$5-\downarrow$	$4-\downarrow$	$3-\downarrow$	$6-\downarrow$	$9-\downarrow$	$1\parallel$	2
MOP6	Mean	3.061E-1	3.046E-1	1.146E-1	8.016E-2	8.547E-2	3.065E-1	5.750E-1	5.398E-2	**5.328E-2**
	Std	2.161E-8	9.552E-3	7.590E-3	1.015E-2	3.941E-3	4.459E-4	1.620E-2	3.094E-3	**2.917E-3**
	Rank	$7-\downarrow$	$6-\downarrow$	$5-\downarrow$	$3-\downarrow$	$4-\downarrow$	$8-\downarrow$	$9-\downarrow$	$2\parallel$	1
MOP7	Mean	3.501E-1	3.512E-1	1.778E-1	2.458E-1	1.171E-1	3.514E-1	6.377E-1	8.186E-2	**7.912E-2**
	Std	7.648E-3	1.463E-7	1.052E-2	3.239E-2	8.566E-3	9.279E-4	9.311E-3	2.778E-3	**2.619E-3**
	Rank	$6-\downarrow$	$7-\downarrow$	$4-\downarrow$	$5-\downarrow$	$3-\downarrow$	$8-\downarrow$	$9-\downarrow$	$2\downarrow$	1
	Total Rank	47	48	31	31	14	52	63	16	13
	Final Rank	6	7	4	4	2	8	9	3	1

According to Wilcoxon's rank sum test, $+$, $-$ and $\approx$ indicate that the corresponding EMO algorithm is significantly better than, worse than or similar to MOEA/D-AOOSTM, while $\uparrow$, $\downarrow$ and $\parallel$ indicate that the corresponding EMO algorithm is significantly better than, worse than or similar to MOEA/D-AMOSTM.

TABLE III
HV RESULTS ON MOP TEST INSTANCES

Problem	HV	DRA	STM	IR	AGR	M2M	NSGA-III	HypE	AOOSTM	AMOSTM
MOP1	Mean	0.564	0.540	1.027	1.062	**1.080**	0.515	0.292	1.071	1.072
	Std	1.097E-1	5.309E-2	4.908E-3	1.202E-2	**9.058E-4**	8.867E-3	1.355E-2	3.882E-3	3.267E-3
	Rank	$6-\downarrow$	$7-\downarrow$	$5-\downarrow$	$4-\downarrow$	$1+\uparrow$	$8-\downarrow$	$9-\downarrow$	$3\parallel$	2
MOP2	Mean	0.476	0.466	0.717	0.680	**0.756**	0.445	0.320	0.745	0.731
	Std	4.459E-2	4.257E-2	3.322E-2	9.570E-2	**2.340E-3**	8.938E-3	9.798E-2	5.037E-2	7.825E-2
	Rank	$6-\downarrow$	$7-\downarrow$	$4-\downarrow$	$5-\downarrow$	$1-\downarrow$	$8-\downarrow$	$9-\downarrow$	$2\parallel$	3
MOP3	Mean	0.240	0.240	0.595	0.560	**0.637**	0.440	0.316	0.606	0.617
	Std	1.665E-16	1.665E-16	5.361E-2	1.240E-1	**4.858E-3**	2.201E-16	9.708E-2	7.281E-2	6.263E-2
	Rank	$8-\downarrow$	$9-\downarrow$	$4-\downarrow$	$5-\downarrow$	$1-\downarrow$	$6-\downarrow$	$7-\downarrow$	$3\parallel$	2
MOP4	Mean	0.578	0.578	0.917	0.912	**0.945**	0.570	0.337	0.931	0.939
	Std	2.040E-2	1.746E-2	4.097E-2	5.434E-2	**2.076E-3**	9.318E-3	1.097E-2	4.521E-2	1.518E-2
	Rank	$6-\downarrow$	$7-\downarrow$	$4-\downarrow$	$5-\downarrow$	$1+\uparrow$	$8-\downarrow$	$9-\downarrow$	$3\downarrow$	2
MOP5	Mean	0.635	0.636	1.006	1.067	1.067	0.648	0.060	1.073	**1.074**
	Std	4.447E-9	8.201E-3	9.637E-3	8.135E-3	4.295E-3	2.991E-2	1.806E-1	3.038E-3	**3.196E-3**
	Rank	$8-\downarrow$	$7-\downarrow$	$5-\downarrow$	$3-\downarrow$	$4-\downarrow$	$6-\downarrow$	$9-\downarrow$	$2\parallel$	1
MOP6	Mean	1.221	1.224	1.418	1.463	1.439	1.216	0.682	1.494	**1.495**
	Std	5.183E-7	1.470E-2	1.843E-2	1.639E-2	1.100E-2	5.601E-3	3.505E-2	6.155E-3	**5.671E-3**
	Rank	$7-\downarrow$	$6-\downarrow$	$5-\downarrow$	$3-\downarrow$	$4-\downarrow$	$8-\downarrow$	$9-\downarrow$	$2\parallel$	1
MOP7	Mean	0.939	0.939	1.038	1.005	1.047	0.933	0.538	1.084	**1.088**
	Std	2.763E-3	1.317E-6	2.296E-2	4.975E-2	2.397E-2	5.768E-3	6.204E-3	5.196E-3	**4.578E-3**
	Rank	$6-\downarrow$	$7-\downarrow$	$4-\downarrow$	$5-\downarrow$	$3-\downarrow$	$8-\downarrow$	$9-\downarrow$	$2\downarrow$	1
	Total Rank	47	50	31	30	15	52	61	17	12
	Final Rank	6	7	5	4	2	8	9	3	1

According to Wilcoxon's rank sum test, $+$, $-$ and $\approx$ indicate that the corresponding EMO algorithm is significantly better than, worse than or similar to MOEA/D-AOOSTM, while $\uparrow$, $\downarrow$ and $\parallel$ indicate that the corresponding EMO algorithm is significantly better than, worse than or similar to MOEA/D-AMOSTM.

from a couple of related ones. However, its preference setting, which encourages the selection in a less crowded area, tends to result in an unstable selection result. In this case, some solutions far away from the PF can be selected occasionally. The reason behind the poor performance of NSGA-III, HypE, and MOEA/D-DRA is that their convergence first and diversity second selection strategies may easily trap the population in some narrow areas. As discussed in Section II-C, the stable matching model used in MOEA/D-STM can easily match a solution with an unfavorable subproblem, thus resulting in an unbalanced selection.

B. Performance Comparisons on UF Instances

The comparison results on the IGD and HV metrics between MOEA/D-AOOSTM, MOEA/D-AMOSTM, and the

TABLE IV
IGD RESULTS ON UF TEST INSTANCES

Problem	IGD	DRA	STM	IR	AGR	M2M	NSGA-III	HypE	AOOSTM	AMOSTM
UF1	Mean	1.071E-3	1.043E-3	2.471E-3	1.813E-3	7.076E-3	9.457E-2	9.902E-2	**9.631E-4**	9.696E-4
	Std	2.583E-4	7.870E-5	1.180E-4	8.699E-5	2.785E-3	1.200E-2	1.089E-2	**4.650E-5**	5.158E-5
	Rank	$4-\downarrow$	$3-\downarrow$	$6-\downarrow$	$5-\downarrow$	$7-\downarrow$	$8-\downarrow$	$9-\downarrow$	$1\parallel$	2
UF2	Mean	4.601E-3	3.024E-3	5.475E-3	5.256E-3	3.957E-3	2.993E-2	2.119E-1	**2.270E-3**	2.577E-3
	Std	9.338E-3	9.309E-4	1.172E-3	7.183E-4	5.099E-4	2.629E-3	6.301E-2	**5.587E-4**	5.649E-4
	Rank	$5\approx\parallel$	$3-\downarrow$	$7-\downarrow$	$6-\downarrow$	$4-\downarrow$	$8-\downarrow$	$9-\downarrow$	$1\uparrow$	2
UF3	Mean	1.772E-2	7.757E-3	1.642E-2	8.141E-3	1.549E-2	2.078E-1	1.805E-1	7.296E-3	**4.110E-3**
	Std	1.500E-2	6.213E-3	1.289E-2	8.673E-3	5.495E-3	4.775E-2	5.100E-2	8.380E-3	**3.128E-3**
	Rank	$7-\downarrow$	$3\approx\downarrow$	$6-\downarrow$	$4\approx\downarrow$	$5-\downarrow$	$9-\downarrow$	$8-\downarrow$	$2\parallel$	1
UF4	Mean	5.320E-2	5.076E-2	5.623E-2	5.025E-2	**3.994E-2**	4.297E-2	4.899E-2	5.269E-2	5.043E-2
	Std	3.115E-3	2.857E-3	2.818E-3	2.874E-3	**3.705E-4**	8.311E-4	7.077E-3	3.523E-3	2.803E-3
	Rank	$8\approx\downarrow$	$6+\parallel$	$9-\downarrow$	$4+\parallel$	$1+\uparrow$	$2+\uparrow$	$3+\uparrow$	$7\downarrow$	5
UF5	Mean	3.033E-1	2.397E-1	2.574E-1	2.625E-1	**1.795E-1**	2.107E-1	2.289E-1	2.514E-1	2.392E-1
	Std	7.779E-2	3.369E-2	4.334E-2	1.102E-1	**3.013E-2**	2.131E-2	4.852E-2	1.766E-2	2.220E-2
	Rank	$9-\downarrow$	$5+\parallel$	$7\approx\downarrow$	$8\approx\parallel$	$1+\uparrow$	$2+\uparrow$	$3+\uparrow$	$6\downarrow$	4
UF6	Mean	1.504E-1	7.805E-2	1.073E-1	1.126E-1	8.990E-2	2.134E-1	2.312E-1	8.146E-2	**6.876E-2**
	Std	1.224E-1	4.305E-2	4.600E-2	7.840E-2	5.355E-2	6.523E-2	6.828E-2	4.048E-2	**3.300E-2**
	Rank	$7-\downarrow$	$2+\parallel$	$5-\downarrow$	$6-\downarrow$	$4\approx\downarrow$	$8-\downarrow$	$9-\downarrow$	$3\downarrow$	1
UF7	Mean	1.245E-3	**1.123E-3**	3.707E-3	2.145E-3	6.234E-3	6.856E-2	2.622E-1	1.150E-3	1.148E-3
	Std	2.371E-4	**7.371E-5**	5.295E-4	3.221E-4	1.867E-3	8.357E-2	4.540E-2	1.095E-4	1.481E-4
	Rank	$4\approx\downarrow$	$1\approx\parallel$	$6-\downarrow$	$5-\downarrow$	$7-\downarrow$	$8-\downarrow$	$9-\downarrow$	$3\parallel$	2
UF8	Mean	3.104E-2	3.019E-2	6.467E-2	4.715E-2	9.655E-2	1.674E-1	3.116E-1	**2.921E-2**	5.393E-2
	Std	4.020E-3	8.706E-3	1.070E-2	9.477E-3	8.181E-3	2.670E-3	3.417E-2	**5.154E-3**	9.528E-3
	Rank	$3-\uparrow$	$2\approx\uparrow$	$6-\downarrow$	$4-\uparrow$	$7-\downarrow$	$8-\downarrow$	$9-\downarrow$	$1\uparrow$	5
UF9	Mean	4.779E-2	**2.373E-2**	5.794E-2	5.861E-2	1.148E-1	1.767E-1	2.353E-1	3.704E-2	3.769E-2
	Std	3.446E-2	**1.112E-3**	3.960E-2	4.572E-2	3.045E-2	3.924E-2	3.018E-2	3.125E-2	4.306E-2
	Rank	$4-\downarrow$	$1+\parallel$	$5-\downarrow$	$6-\downarrow$	$7-\downarrow$	$8-\downarrow$	$9-\downarrow$	$2\downarrow$	3
UF10	Mean	5.184E-1	1.701E+0	7.216E-1	4.168E-1	5.572E-1	**2.257E-1**	2.568E-1	1.028E+0	2.426E+0
	Std	6.698E-2	2.849E-1	1.202E-1	7.165E-2	5.950E-2	**5.700E-2**	6.938E-2	2.943E-1	1.868E-1
	Rank	$4+\uparrow$	$8-\uparrow$	$6+\uparrow$	$3+\uparrow$	$5+\uparrow$	$1+\uparrow$	$2+\uparrow$	$7\uparrow$	9
	Total Rank	55	34	63	51	48	62	70	33	34
	Final Rank	6	2	8	5	4	7	9	1	2

According to Wilcoxon's rank sum test, $+$, $-$ and $\approx$ indicate that the corresponding EMO algorithm is significantly better than, worse than or similar to MOEA/D-AOOSTM, while $\uparrow$, $\downarrow$ and $\parallel$ indicate that the corresponding EMO algorithm is significantly better than, worse than or similar to MOEA/D-AMOSTM.

other EMO algorithms on UF benchmark suite are presented in Tables IV and V. Different from the MOP benchmark suite, the major source of difficulty for the UF benchmark suite is not the diversity preservation but the complicated PS. Generally speaking, the overall performance of MOEA/D-AOOSTM ranks the first on the UF benchmark suite, followed by MOEA/D-AMOSTM and their predecessor MOEA/D-STM. More specifically, for both IGD and HV metrics, MOEA/D-AOOSTM performs the best on UF1, UF2, and UF8 and acts as the top three algorithm on all instances except for UF4, UF5, and UF10. For UF3 and UF7, the performance of MOEA/D-AOOSTM does not show significant difference with the best performing algorithms. MOEA/D-AMOSTM shows similar rankings to MOEA/D-AOOSTM. It is significantly better than MOEA/D-AOOSTM on UF4–UF6 and UF9 in terms of both IGD and HV metrics. In contrast, MOEA/D-AOOSTM wins on UF2, UF8, and UF10 according to Wilcoxon's rank sum test of the IGD results and wins on UF1, UF2, UF7, UF8, and UF10 in the HV tests.

According to the performance of different algorithms on the UF test instances, the analysis can be divided into three groups. For UF4 and UF5, MOEA/D-M2M, NSGA-III and HypE are able to provide better performance than all other MOEA/D variants. All these three algorithms use the Pareto dominance as the major driving force in the environmental selection, which can improve the convergence to a great extent. For UF1–UF3 and UF6–UF9, all the MOEA/D variants outperform NSGA-III and HypE. In particular, The three variants with stable matching-based selection, i.e., MOEA/D-AOOSTM, MOEA/D-AMOSTM, and MOEA/D-STM, have shown very promising results on these six test instances. The superior performance can be attributed to the well balance between convergence and diversity achieved by the stable matching relationship between subproblems and solutions. gMOEA/D-AGR has shown a medium performance for the former two groups of problem instances. This might be due to its adaptive mechanism that can hardly make a satisfied prediction of the replacement neighborhood size. UF10 is a difficult tri-objective problem, where none of these eight EMO algorithms are able to obtain a well approximation to the PF within the given number of function evaluations. Nevertheless, it is worth noting that the empirical studies in [10] demonstrate that the stable matching-based selection mechanism can offer a competitive result in case the maximum number of function evaluations is doubled.

C. Performance Comparisons on Bi-Objective WFG Instances

From the comparison results shown in Tables I and II of the supplementary material, it can be seen that MOEA/D-AOOSTM and MOEA/D-AMOSTM are the best

TABLE V
HV RESULTS ON UF TEST INSTANCES

Problem	HV	DRA	STM	IR	AGR	M2M	NSGA-III	HypE	AOOSTM	AMOSTM
UF1	Mean	1.104	1.104	1.101	1.102	1.092	0.945	0.941	**1.104**	1.104
	Std	6.732E-4	4.643E-4	6.668E-4	4.079E-4	5.167E-3	2.713E-2	2.800E-2	**3.180E-4**	4.627E-4
	Rank	4 − ‖	3 − ‖	6 − ↓	5 − ↓	7 − ↓	8 − ↓	9 − ↓	1 ↑	2
UF2	Mean	1.097	1.100	1.093	1.096	1.099	1.054	0.889	**1.101**	1.101
	Std	1.238E-2	1.889E-3	3.835E-3	1.866E-3	1.903E-3	6.402E-3	4.580E-2	**1.730E-3**	1.395E-3
	Rank	5 − ‖	3 − ‖	7 − ↓	6 − ↓	4 − ↓	8 − ↓	9 − ↓	1 ↑	2
UF3	Mean	1.073	1.093	1.075	1.090	1.079	0.732	0.793	1.094	**1.099**
	Std	2.867E-2	1.066E-2	2.590E-2	1.848E-2	8.270E-3	5.290E-2	6.111E-2	1.513E-2	**5.436E-3**
	Rank	7 − ↓	3 − ↓	6 − ↓	4 ≈ ↓	5 − ↓	9 − ↓	8 − ↓	2 ‖	1
UF4	Mean	0.672	0.679	0.667	0.680	**0.701**	0.698	0.685	0.676	0.679
	Std	5.996E-3	5.759E-3	5.444E-3	5.213E-3	**7.356E-4**	1.257E-3	1.459E-2	6.500E-3	5.755E-3
	Rank	8 − ↓	6 ≈ ‖	9 − ↓	4 + ‖	1 + ↑	2 + ↑	3 + ↑	7 ↓	5
UF5	Mean	0.353	0.437	0.414	0.455	**0.574**	0.536	0.519	0.411	0.434
	Std	8.320E-2	7.346E-2	8.085E-2	1.201E-1	**6.204E-2**	3.906E-2	7.893E-2	3.560E-2	4.814E-2
	Rank	9 − ↓	5 + ‖	7 ≈ ‖	4 + ‖	1 + ↑	2 + ↑	3 + ↑	8 ↓	6
UF6	Mean	0.591	0.645	0.610	0.647	**0.685**	0.621	0.592	0.646	0.666
	Std	1.152E-1	8.742E-2	8.839E-2	6.045E-2	**4.863E-2**	2.851E-2	6.576E-2	8.337E-2	6.562E-2
	Rank	9 − ↓	5 ≈ ‖	7 − ↓	3 ≈ ↓	1 ≈ ‖	6 − ↓	8 − ↓	4 ↓	2
UF7	Mean	0.937	**0.937**	0.931	0.935	0.928	0.831	0.610	0.937	0.937
	Std	9.245E-4	**4.244E-4**	1.589E-3	1.674E-3	3.597E-3	1.055E-1	3.176E-2	6.227E-4	6.386E-4
	Rank	4 − ‖	1 ≈ ↑	6 − ↓	5 − ↓	7 − ↓	8 − ↓	9 − ↓	2 ↑	3
UF8	Mean	1.127	1.125	1.050	1.088	0.938	0.777	0.783	**1.143**	1.073
	Std	9.272E-3	1.428E-2	2.737E-2	2.234E-2	2.383E-2	4.349E-3	3.482E-3	**1.291E-2**	2.411E-2
	Rank	2 − ↑	3 − ↑	6 − ↓	4 − ↑	7 − ↓	9 − ↓	8 − ↓	1 ↑	5
UF9	Mean	1.402	**1.462**	1.400	1.395	1.243	0.971	0.855	1.455	1.453
	Std	6.489E-2	**3.831E-3**	7.704E-2	9.007E-2	5.711E-2	7.193E-2	8.419E-2	6.385E-2	8.895E-2
	Rank	4 − ↓	1 − ↓	5 − ↓	6 − ↓	7 − ↓	8 − ↓	9 − ↓	2 ↓	3
UF10	Mean	0.188	0.000	0.063	0.311	0.172	**0.653**	0.612	0.015	0.000
	Std	4.694E-2	3.781E-4	4.914E-2	6.402E-2	3.272E-2	**1.121E-1**	1.286E-1	3.031E-2	0.000E+0
	Rank	4 + ↑	8 − ‖	6 + ↑	3 + ↑	5 + ↑	1 + ↑	2 + ↑	7 ↑	9
	Total Rank	56	38	65	44	45	61	68	35	38
	Final Rank	6	2	8	4	5	7	9	1	2

According to Wilcoxon's rank sum test, +, − and ≈ indicate that the corresponding EMO algorithm is significantly better than, worse than or similar to MOEA/D-AOOSTM, while ↑, ↓ and ‖ indicate that the corresponding EMO algorithm is significantly better than, worse than or similar to MOEA/D-AMOSTM.

two algorithms in overall performance on the bi-objective WFG instances. Comparing with the seven existing algorithms, MOEA/D-AOOSTM and MOEA/D-AMOSTM achieves significant better performance in 56 and 57 out of 63 IGD comparisons, respectively. As for HV results, they both wins in 57 comparisons. In particular, MOEA/D-AOOSTM and MOEA/D-AMOSTM obtain the best mean metric values on WFG1, WFG3, WFG6, WFG7, and WFG9 and obtain very promising results on WFG3 and WFG5. Even though the mean HV metric values of MOEA/D-AOOSTM and MOEA/D-AMOSTM on WFG2 rank the fifth and sixth, all the other algorithms are significantly worse than them. It is interesting to note that MOEA/D-AOOSTM and MOEA/D-AMOSTM are significantly better than MOEA/D-STM on all WFG instances except for WFG8. One possible reason is the proper normalization method used in MOEA/D-AOOSTM/AMOSTM. However, we also notice that the performance of NSGA-III fluctuates significantly on difficult problem instances, though it uses the same normalization method. The other variants of MOEA/D perform more or less the same on all test instances except MOEA/D-M2M significantly outperforms all other EMO algorithms on WFG8. The indicator-based algorithm HypE performs the worst on all nine problems. Comparing the IGD and HV results, the algorithm comparisons are consistent on most test instances, except when the performance of two algorithms are very close, the ranking of IGD and HV metric values may change slightly. However, it is worth noting that the algorithms perform quite differently on WFG2 under the IGD and HV assessments. NSGA-III shows the best mean IGD value but gives the second worst mean HV value. In contrast, the mean HV value of gMOEA/D-AGR ranks the first among all algorithms but its mean IGD value only obtains a rank of 7. This is probably because WFG2 has a discontinuous PF, which makes the distinction between IGD and HV more obvious.

D. Effectiveness of the Adaptive Mechanism

To show the functionality of the adaptive mechanism proposed in Section III-D, we choose MOP1 as an example and plot the trajectories of r of the selected solutions of four different subproblems, i.e., p^1, p^{34}, p^{67}, and p^{100}, controlled by the local competitiveness-based adaptive mechanism. From trajectories shown in Fig. 3, we notice that the r value fluctuates significantly at the early stages of the evolution. Afterwards, it almost convergences to the threshold $\ell_{\max}$. This is because the local competitiveness varies dramatically when the population is far away from but heading to the PF. With the progress of evolution, the selected solutions gradually become nondominated from each other. As a consequence, the value of ℓ, which keeps the solution locally noncompetitive, grows and finally settles at $\ell_{\max}$. All in all, we can see that

TABLE VI
FINAL RANKS OF MEAN METRIC VALUES ON MOP, UF, AND BI-OBJECTIVE WFG TEST INSTANCES

Metric	Rank	DRA	STM	IR	AGR	M2M	NSGA-III	HypE	AOOSTM	AMOSTM
IGD	Total Rank	146	124	160	123	115	154	214	68	66
	Final Rank	6	5	8	4	3	7	9	2	1
HV	Total Rank	147	132	165	104	109	157	209	72	75
	Final Rank	6	5	8	3	4	7	9	1	2

TABLE VII
FINAL RANKS OF MEAN METRIC VALUES ON 3-, 5-, 8-, AND 10-OBJECTIVE WFG TEST INSTANCES

Metric	Rank	DRA	MOEA/DD	PICEA-g	NSGA-III	HypE	AOOSTM	AMOSTM
HV	Total Rank	223	153	133	120	226	70	83
	Final Rank	6	5	4	3	7	1	2

TABLE VIII
AVERAGE PERFORMANCE SCORES ON MOP, UF, AND BI-OBJECTIVE WFG TEST INSTANCES

Metric	Problem	DRA	STM	IR	AGR	M2M	NSGA-III	HypE	AOOSTM	AMOSTM
IGD	MOP	6.14	5.86	3.86	3.43	1.57	6.43	8.00	0.57	**0.29**
	UF	3.90	2.30	5.40	3.30	4.00	5.20	5.90	2.30	**2.10**
	WFG	3.44	3.78	6.78	3.56	5.44	3.33	7.89	1.22	**0.67**
	Overall	4.35	3.77	5.46	3.42	3.85	4.88	7.15	1.46	**1.12**
HV	MOP	5.57	5.57	4.00	3.00	2.00	6.71	8.00	0.57	**0.29**
	UF	4.00	2.40	5.50	3.30	3.90	5.30	5.80	**1.90**	2.40
	WFG	3.67	4.00	6.89	2.67	5.00	3.78	7.89	**0.56**	0.89
	Overall	4.31	3.81	5.58	3.00	3.77	5.15	7.12	**1.08**	1.31

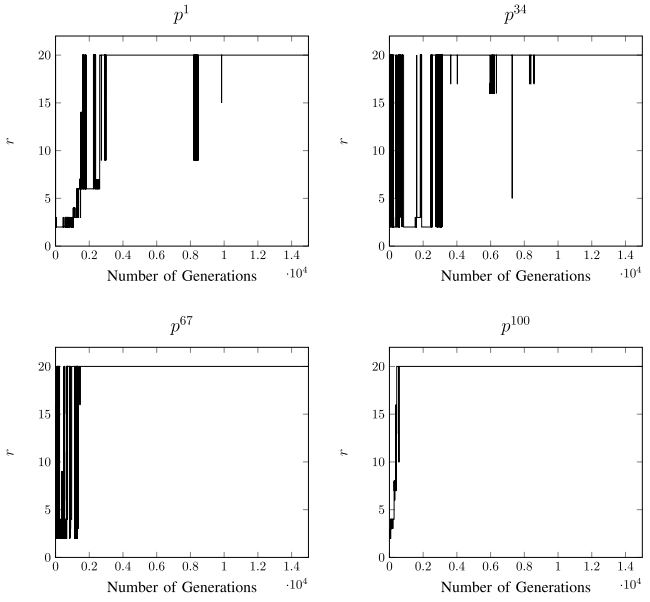


Fig. 3. Illustration of the r values of the solutions selected by subproblems p^1, p^{34}, p^{67}, and p^{100}.

different solutions have different local competitiveness, thus it is meaningful to have a different length of the preference list.

In order to further investigate the effectiveness brought by our proposed adaptive mechanism, we develop two variants, denoted by MOEA/D-AOOSTM-v and MOEA/D-AMOSTM-v, in which all solutions share the same static r setting. From the our offline parameter studies, we finally find that $r = 4$ and $r = 8$ are the best settings for MOP and UF benchmark suites, respectively. Thereafter, in Table VII of the supplementary material, we show the IGD results of MOEA/D-AOOSTM-v and MOEA/D-AMOSTM-v with the best r settings. Comparing with the best static settings of r, we can see that both MOEA/D-AOOSTM and MOEA/D-AMOSTM obtain better mean IGD values than their variant on 11 out of 17 problems. Even though MOEA/D-AOOSTM-v and MOEA/D-AMOSTM-v perform better on some test instances, in most of the cases, our proposed adaptive mechanism achieve comparable results to the best settings of r. Note that the optimal r settings are obtained from a series of comprehensive try-and-error experiments, which are not as intelligent and flexible as our proposed adaptive mechanism. Therefore, we conclude that our proposed adaptive mechanism based on the local competitiveness is generally effective for dynamically setting the length of the solution's preference list.

E. Summaries

We summarize the total ranks of different algorithms based on the mean metric values in Tables VI and VII and compute their final ranks over all test instances. As can be seen from the tables, the proposed MOEA/D-AOOSTM and MOEA/D-AMOSTM remain the two best algorithms on all test instances. It seems that MOEA/D-AOOSTM achieves better performance under the HV metric while MOEA/D-AMOSTM obtains better IGD results.

To analyze the relative performance of different algorithms in a statistical point of view, we adopt the performance score [7] to qualify the algorithms. Given K algorithms $\{A_1, \ldots, A_K\}$, the performance score of each algorithm A_i, $i \in \{1, \ldots, K\}$ on a certain test instance, is defined as

$$P(A_i) = \sum_{j=1, j\neq i}^{K} \delta_{i,j} \tag{11}$$

TABLE IX
AVERAGE PERFORMANCE SCORES ON 3-, 5-, 8-, AND 10-OBJECTIVE WFG TEST INSTANCES

Metric	Problem	DRA	MOEA/DD	PICEA-g	NSGA-III	HypE	AOOSTM	AMOSTM
HV	$m = 3$	4.78	3.67	2.33	1.89	5.89	**0.56**	0.67
	$m = 5$	5.00	3.56	2.00	3.00	5.67	**0.11**	0.89
	$m = 8$	5.22	3.56	2.89	2.22	4.78	**1.44**	1.78
	$m = 10$	5.78	3.56	2.56	2.11	4.78	**1.00**	1.67
	Overall	5.19	3.58	2.44	2.31	5.28	**0.78**	1.25

where $\delta_{i,j} = 1$ when A_i is significantly outperformed by A_j; otherwise, $\delta_{i,j} = 0$. This time, we use the Kolmogorov-Smirnov test to examine whether an algorithm is significantly outperformed by another. The performance score indicates how many other algorithms are significantly better than the corresponding algorithm on a certain test instance. Thus, the smaller the performance score, the better the algorithm. Tables VIII and IX present the average performance scores of different algorithms over all test instances, where we add a gray background to the top two algorithms and highlight the best algorithms in boldface. Coincident with the total ranks on mean metric values, MOEA/D-AOOSTM and MOEA/D-AMOSTM obtain the best average performance scores on all test instances with different number of objectives. Comparing with all other algorithms, the leads of the two proposed algorithms employing stable matching-based selection with incomplete lists are statistically significant.

VI. CONCLUSION

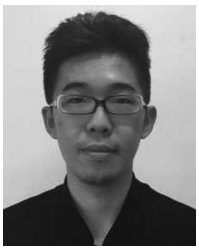
The stable matching-based selection mechanism paves a new avenue to address the balance between convergence and diversity from the perspective of achieving the equilibrium between the preferences of subproblems and solutions. However, considering the population diversity, it might not be appropriate to allow each solution to match with any subproblem on its preference list. This paper introduced the incomplete preference lists into the stable matching-based selection model, in which the length of the preference list of each solution is restricted so that a solution is only allowed to match with one of its favorite subproblems. Based on the partial preference information, a two-level one-one stable matching-based selection mechanism and a many-one stable matching-based selection mechanism are proposed and integrated into MOEA/D. Note that the length of the preference list of each solution is problem dependent and is related to the difficulty of the corresponding subproblem. To address this issue, an adaptive mechanism is proposed to dynamically control the length of the preference list of each solution according to the local competitiveness information. Comprehensive experiments are conducted on 62 benchmark problems which cover various characteristics, e.g., multimodality, deceptive, complicated PSs, and many objectives. From the experimental studies in Section V, we can clearly observe the competitive performance obtained by our proposed MOEA/D-AOOSTM and MOEA/D-AMOSTM, comparing with a variety of state-of-the-art EMO algorithms.

Although our proposed MOEA/D-AOOSTM and MOEA/D-AMOSTM have shown very competitive performance in the empirical studies, we also notice that the stable matching relationship between subproblems and solutions may sacrifice the convergence property of the population to some extent. One possible reason might be both two-level one-one matching and many-one matching restricts that each solution can only be selected by at most one subproblem. Future work could be focused on assigning higher priorities for elite solutions to produce offspring solutions or allowing elite solutions to be matched with more than one subproblem. It is also interesting to apply the proposed algorithms to real-world application scenarios.

REFERENCES

[1] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. New York, NY, USA: Wiley, 2001.
[2] K. Deb, A. Pratap, S. Agrawal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
[3] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization," in *Proc. Evol. Methods Design Optim. Control*, Athens, Greece, 2002, pp. 95–100.
[4] J. D. Knowles and D. W. Corne, "Approximating the nondominated front using the Pareto archived evolution strategy," *Evol. Comput.*, vol. 8, no. 2, pp. 149–172, 2000.
[5] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Proc. 8th Int. Conf. Parallel Problem Solving Nat. (PPSN)*, Birmingham, U.K., 2004, pp. 832–842.
[6] N. Beume, B. Naujoks, and M. T. M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *Eur. J. Oper. Res.*, vol. 181, no. 3, pp. 1653–1669, 2007.
[7] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *Evol. Comput.*, vol. 19, no. 1, pp. 45–76, 2011.
[8] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.
[9] K. Li, Á. Fialho, S. Kwong, and Q. Zhang, "Adaptive operator selection with bandits for a multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 18, no. 1, pp. 114–130, Feb. 2014.
[10] K. Li, Q. Zhang, S. Kwong, M. Li, and R. Wang, "Stable matching-based selection in evolutionary multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 18, no. 6, pp. 909–923, Dec. 2014.
[11] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, Apr. 2009.
[12] K. Li, S. Kwong, Q. Zhang, and K. Deb, "Interrelationship-based selection for decomposition multiobjective optimization," *IEEE Trans. Cybern.*, vol. 45, no. 10, pp. 2076–2088, Oct. 2015.
[13] H.-L. Liu, F. Gu, and Q. Zhang, "Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems," *IEEE Trans. Evol. Comput.*, vol. 18, no. 3, pp. 450–455, Jun. 2014.
[14] H.-L. Liu, L. Chen, K. Deb, and E. D. Goodman, "Investigating the effect of imbalance between convergence and diversity in evolutionary multi-objective algorithms," *IEEE Trans. Evol. Comput.*, vol. PP, no. 99, pp. 1–1, Sept. 2016.
[15] Z. Wang, Q. Zhang, A. Zhou, M. Gong, and L. Jiao, "Adaptive replacement strategies for MOEA/D," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp. 474–486, Feb. 2016.
[16] K. Li, K. Deb, Q. Zhang, and S. Kwong, "An evolutionary many-objective optimization algorithm based on dominance and decomposition," *IEEE Trans. Evol. Comput.*, vol. 19, no. 5, pp. 694–716, Oct. 2015.

[17] X. Cai, Y. Li, Z. Fan, and Q. Zhang, "An external archive guided multiobjective evolutionary algorithm based on decomposition for combinatorial optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 4, pp. 508–523, Aug. 2015.
[18] K. Li, S. Kwong, and K. Deb, "A dual-population paradigm for evolutionary multiobjective optimization," *Inf. Sci.*, vol. 309, pp. 50–72, Jul. 2015.
[19] Q. Liu, G. J. Mailath, A. Postlewaite, and L. Samuelson, "Stable matching with incomplete information," *Econometrica*, vol. 82, no. 2, pp. 541–587, 2014.
[20] M. Wu *et al.*, "Two-level stable matching-based selection in MOEA/D," in *Proc. IEEE Int. Conf. Syst. Man. Cybern. (SMC)*, Hong Kong, 2015, pp. 1720–1725.
[21] K. Miettinen, *Nonlinear Multiobjective Optimization*, vol. 12. Boston, MA, USA: Kluwer, 1999.
[22] Q. Zhang, W. Liu, and H. Li, "The performance of a new version of MOEA/D on CEC09 unconstrained MOP test instances," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Trondheim, Norway, May 2009, pp. 203–208.
[23] D. Gale and L. S. Shapley, "College admissions and the stability of marriage," *Amer. Math. Monthly*, vol. 120, no. 5, pp. 386–391, 2013.
[24] D. Gusfield and R. W. Irving, *The Stable Marriage Problem—Structure and Algorithms* (Foundations of Computing). Cambridge, MA, USA: MIT Press, 1989.
[25] K. Iwama, S. Miyazaki, Y. Morita, and D. Manlove, "Stable marriage with incomplete lists and ties," in *Proc. 26th Int. Colloquium Autom. Lang. Program. (ICALP)*, Prague, Czech Republic, Jul. 1999, pp. 443–452.
[26] P. Biró, T. Fleiner, R. W. Irving, and D. F. Manlove, "The college admissions problem with lower and common quotas," *Theor. Comput. Sci.*, vol. 411, nos. 34–36, pp. 3136–3153, 2010.
[27] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 577–601, Aug. 2014.
[28] I. Das and J. E. Dennis, "Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems," *SIAM J. Optim.*, vol. 8, no. 3, pp. 631–657, 1998.
[29] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 4–31, Feb. 2011.
[30] R. B. Agrawal, K. Deb, and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex Syst.*, vol. 9, no. 2, pp. 115–148, 1994.
[31] K. Deb and M. Goyal, "A combined genetic adaptive search (GeneAS) for engineering design," *Comput. Sci. Informat.*, vol. 26, no. 2, pp. 30–45, 1996.
[32] Q. Zhang *et al.*, "Multiobjective optimization test instances for the CEC 2009 special session and competition," Dept. Mech. Eng., Univ. Essex, Colchester, U.K., and Nanyang Technol. Univ., Singapore, Tech. Rep. CES-487, 2008.
[33] S. Huband, P. Hingston, L. Barone, and R. L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Trans. Evol. Comput.*, vol. 10, no. 5, pp. 477–506, Oct. 2006.
[34] R. Wang, R. C. Purshouse, and P. J. Fleming, "Preference-inspired coevolutionary algorithms for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 17, no. 4, pp. 474–494, Aug. 2013.
[35] P. A. N. Bosman and D. Thierens, "The balance between proximity and diversity in multiobjective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 174–188, Apr. 2003.
[36] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257–271, Nov. 1999.



Mengyuan Wu received the B.Sc. and M.Sc. degrees in industrial sciences from KU Leuven, Leuven, Belgium, in 2013 and 2014, respectively. He is currently pursuing the Ph.D. degree in computer science with the City University of Hong Kong, Hong Kong.

His research interests include evolutionary computation, multi-objective optimization, stable matching and machine learning.

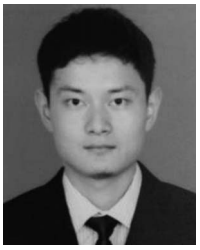
Ke Li received the B.Sc. and M.Sc. degrees in computer science and technology from Xiangtan University, Xiangtan, China, in 2007 and 2010, respectively, and the Ph.D. degree in computer science from the City University of Hong Kong, Hong Kong, in 2014.

He was a Post-Doctoral Research Associate with Michigan State University, East Lansing, MI, USA, and a Research Fellow with the University of Birmingham, Birmingham, U.K. He is currently a Lecturer (Assistant Professor) of Data Analytics with the University of Exeter, Exeter, U.K. He has published several research papers in renowned journals, including the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION and the IEEE TRANSACTIONS ON CYBERNETICS. His current research interests include the evolutionary multiobjective optimization, large-scale optimization, statistical machine learning, and applications in software engineering and industrial design.

Sam Kwong (M'93–SM'04–F'14) received the B.Sc. degree in electrical engineering from the State University of New York, Buffalo, NY, USA, in 1983, the M.Sc. degree in electrical engineering from the University of Waterloo, Waterloo, ON, Canada, in 1985, and the Ph.D. degree from the University of Hagen, Hagen, Germany, in 1996.

He was a Diagnostic Engineer with Control Data Canada, Mississauga, ON, Canada, from 1985 to 1987. He later joined Bell Northern Research Canada, Ottawa, ON, Canada, as a Member of Scientific Staff. He joined the Department of Electronic Engineering, City University of Hong Kong, Hong Kong, as a Lecturer in 1990, where he is currently a Professor with the Department of Computer Science. His current research interests include video coding, pattern recognition, and evolutionary algorithms.

Prof. Kwong is an Associate Editor of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, the IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS and the IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS.



Yu Zhou received the B.Sc. degree in electronics and information engineering and the M.Sc. degree in circuits and systems from Xidian University, Xi'an, China, in 2009 and 2012, respectively, and the Ph.D. degree in computer science from the City University of Hong Kong, Hong Kong, in 2017.

His current research interests include compressed sensing, image processing, machine learning, and computational intelligence.

Qingfu Zhang (M'01–SM'06–F'17) received the B.Sc. degree in mathematics from Shanxi University, Taiyuan, China, in 1984, and the M.Sc. degree in applied mathematics and the Ph.D. degree in information engineering from Xidian University, Xi'an, China, in 1991 and 1994, respectively.

He is a Professor with the Department of Computer Science, City University of Hong Kong, Hong Kong, where he is currently leading the Metaheuristic Optimization Research Group. His current research interests include evolutionary computation, optimization, neural networks, data analysis, and their applications. Multiobjective evolutionary algorithm based on decomposition developed in his group is one of the most widely used multiobjective evolutionary algorithmic framework.

Prof. Zhang was a recipient of the 2010 IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION Outstanding Paper Award, and the 2016 Highly Cited Researchers Award in computer science by Thomson Reuters. He is an Associate Editor of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, the IEEE TRANSACTIONS ON CYBERNETICS, and other international journals.