Hyperplane Assisted Evolutionary Algorithm for Many-Objective Optimization Problems

Huangke Chen, Ye Tian, Witold Pedrycz, *Fellow, IEEE*, Guohua Wu, Rui Wang, and Ling Wang

***Abstract*—In many-objective optimization problems (MaOPs), forming sound tradeoffs between convergence and diversity for the environmental selection of evolutionary algorithms is a laborious task. In particular, strengthening the selection pressure of population toward the Pareto-optimal front becomes more challenging, since the proportion of nondominated solutions in the population scales up sharply with the increase of the number of objectives. To address these issues, this paper first defines the nondominated solutions exhibiting evident tendencies toward the Pareto-optimal front as prominent solutions, using the hyperplane formed by their neighboring solutions, to further distinguish among nondominated solutions. Then, a novel environmental selection strategy is proposed with two criteria in mind: 1) if the number of nondominated solutions is larger than the population size, all the prominent solutions are first identified to strengthen the selection pressure. Subsequently, a part of the other nondominated solutions are selected to balance convergence and diversity and 2) otherwise, all the nondominated solutions are selected; then a part of the dominated solutions are selected according to the predefined reference vectors. Moreover, based on the definition of prominent solutions and the new selection strategy, we propose a hyperplane assisted evolutionary algorithm, referred here as *hpaEA*, for solving MaOPs. To demonstrate the performance of *hpaEA*, extensive experiments are conducted to compare it with five state-of-the-art many-objective evolutionary algorithms on 36 many-objective benchmark instances. The experimental results show the superiority of *hpaEA* which significantly outperforms the compared algorithms on 20 out of 36 benchmark instances.**

***Index Terms*—Continuous optimization, evolutionary computations, hyperplane, many-objective optimization, selection pressure.**

I. Introduction

OPTIMIZATION problems encountered in real-world problems often involve many conflicting objectives. Taking the problem of scheduling workflows in cloud computing [1] as an example, several conflicting factors (e.g., makespan, cost, security, etc.) are required to be optimized simultaneously. Similarly, in automotive engine calibration problem [2], one needs to optimize multiple conflicting objectives. Such optimization problems including multiple objectives are called multiobjective optimization problems (MOPs) [3], [4], which can be defined as follows:

$$\begin{aligned} \min \ \ & F(\mathbf{x}) = \left[f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})\right] \\ \text{s.t.} \ \ & \mathbf{x} \in \Omega \end{aligned} \tag{1}$$

where $\Omega \subseteq \mathbb{R}^n$ and $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \Omega$ denote the decision space and the decision vector, respectively. The parameter n represents the count of decision variables. The objective function is formulated in terms of F, that is, $F : \mathbb{R}^n \to \mathbb{R}^m$, in which m represents the number of objectives. Generally, the MOPs with more than three objectives are regarded as many-objective optimization problems (MaOPs) [5]–[7]. That is to say, the problem (1) is called an MaOP if the number of its objectives satisfies the condition $m > 3$.

Due to the conflicting nature of the objectives present in MOPs, improving one objective typically cannot be achieved without the deterioration of at least one of the other objectives [8]. Thus, there rarely exists a single solution being capable of making all the objectives to be optimal, and a number of Pareto-optimal solutions can be found for each MOP. For two feasible solutions $\mathbf{x}_1, \mathbf{x}_2 \in \Omega$ considered in the MOP, we say that $\mathbf{x}_1$ dominates $\mathbf{x}_2$, if all the objectives of $\mathbf{x}_1$ are not larger than those of $\mathbf{x}_2$ and at least one objective of $\mathbf{x}_1$ is less than that of $\mathbf{x}_2$. A solution $\mathbf{x}^* \in \Omega$ is Pareto optimal if there exists no solution that dominates it. In addition, all the Pareto-optimal solutions constitute the Pareto set (PS) in the decision space, and the Pareto front (PF) in the objective space.

To solve the MOPs, multiobjective evolutionary algorithms (MOEAs) have attracted a great deal of attention and were rapidly developed resulting in a considerable number of interesting studies [9]. The MOEAs have exhibited promising

Manuscript received August 22, 2018; revised December 6, 2018 and February 3, 2019; accepted February 11, 2019. This work was supported in part by the National Science Foundation of China under Grant 61603404, Grant 71801218, and Grant 61873328, in part by Natural Science Fund for Distinguished Young Scholars of Hunan Province under Grant 2019JJ20026, in part by the National Defense Technology Innovation Special Zone Projects under Grant 18-163-11-ZT-003-008-01 and Grant 18-163-11-ZT-003-008-02, and in part by the National Natural Science Fund for Distinguished Young Scholars of China under Grant 61525304. This paper was recommended by Associate Editor S. Yang. *(Corresponding author: Guohua Wu.)*

H. Chen and R. Wang are with the College of Systems Engineering, National University of Defense Technology, Changsha 410073, China (e-mail: hkchen@nudt.edu.cn; ruiwangnudt@gmail.com).

Y. Tian is with the Institute of Physical Science and Information Technology, Anhui University, Heifei 230039, China (e-mail: field910921@gmail.com).

W. Pedrycz is with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6G 2V4, Canada, also with the Department of Electrical and Computer Engineering, Faculty of Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia, and also with the Systems Research Institute, Polish Academy of Sciences, 01447 Warsaw, Poland (e-mail: wpedrycz@ualberta.ca).

G. Wu is with the School of Traffic and Transportation Engineering, Central South University, Changsha 410073, China (e-mail: guohuawu@csu.edu.cn).

L. Wang is with the Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: wangling@mail.tsinghua.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2019.2899225

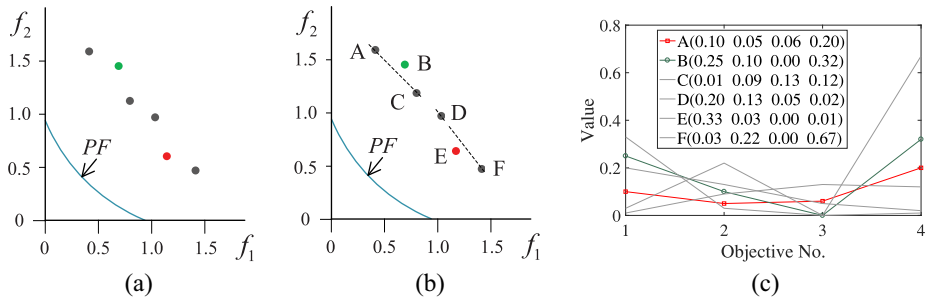


Fig. 1. Illustration of prominent solutions. (a) Population in a 2-D space. (b) Prominent solution in a 2-D space. (c) Prominent solution in a 4-D space.

performance in solving MOPs with two or three objectives. However, their performance deteriorates considerably when solving MOPs with more than three objectives, that is, MaOPs. Thus, MaOPs have attracted intensive attention, resulting in a large number of many-objective evolutionary algorithms (MaOEAs) [10].

Till now, it is an ongoing challenging task of forming sound tradeoffs between convergence and diversity, especially enhancing the selection pressure for the population of an MaOEA. The reason for this challenge is twofold. First, the objective space expands exponentially with the increase of the number of the objectives, which makes the limited solutions being distributed more sparsely in the objective space [11], and challenges the diversity preservation for the population. Second, with the increasing number of objectives in MaOPs, the proportion of nondominated solutions in a population scales up fleetly, and dominance-based sorting approach cannot discriminate between such nondominated solutions [10], [12]. This hinders the convergence of the population.

Motivation: As a matter of fact, some nondominated solutions exhibit an evident tendency toward the PF in comparison with other adjacent nondominated solutions. These solutions can be distinguished by the hyperplane formed by their neighboring solutions. Here, we use two visual examples in Fig. 1 to illustrate this phenomenon. For the nondominated solutions, shown in Fig. 1(a), it is intuitive that the solution shown as the red point is obviously much closer to the PF than the one being depicted by the green point. This can be identified by the relationship between them and the hyperplanes (lines in a 2-D space) formed by their neighboring solutions: 1) from Fig. 1(b), we can see that the green point B is located above the line between its neighboring solutions A and C and 2) regarding the red point E, it is located below the line formed by its neighbors D and F.

Similar property can also be observed in higher-dimensional space. Consider a 4-D space as an example shown in Fig. 1(c), which is displayed by parallel coordinates [13]. The parallel coordinate transforms m-dimensional vectors into a 2-D graph with m parallel axes. For an m-dimensional vector, its value obtained along with the ith dimension is plotted as a vertex on the ith axes, and then a polyline is employed to link the vertices on the axes. For instance, the objective vector of solution A is represented as the red polyline in Fig. 1(c). Although solution A and solution B (shown in green polyline) are all nondominated, solution A seems to be better than B in terms of convergence, since the objective values of solution A are much lower than those of solution B except for the third objective. Furthermore, the hyperplane ($f_4 = 0.8926 - 2.6151 \times f_1 - 0.6554 \times f_2 - 5.2884 \times f_3$) formed by their neighbors from C to F can also distinguish between the two solutions A and B: solution A is located below the hyperplane (i.e., $0.20 < 0.8926 - 2.6151 \times 0.10 - 0.6554 \times 0.05 - 5.2884 \times 0.06 = 0.281$), while solution B is positioned above the hyperplane (i.e., $0.32 > 0.8926 - 2.6151 \times 0.25 - 0.6554 \times 0.10 - 5.2884 \times 0.0 = 0.173$). Motivated by the above observations, this paper focuses on further distinguishing the nondominated solutions using hyperplanes, and thereby strengthening selection pressure for the population.

Here, we define the nondominated solution having an evident tendency toward the PF as a prominent solution, which is given as follows.

Definition 1 (Neighboring Solution Set): For a solution $\mathbf{x}_i$ with m objectives, its neighboring solution set is defined as $B(\mathbf{x}_i) = \{\mathbf{x}_i^1, \mathbf{x}_i^2, \ldots, \mathbf{x}_i^m\}$, where $\mathbf{x}_i^j = \arg\max_{1 \le k \le N}\{f_j(\mathbf{x}_k) | f_j(\mathbf{x}_k) < f_j(\mathbf{x}_i)\}, j = 1, 2, \ldots, m$.

Definition 2 (Prominent Solution): For a solution $\mathbf{x}_i$, it is defined as a prominent solution if there exist m different elements in its neighboring solution set and its objective vector $[f_1(\mathbf{x}_i), f_2(\mathbf{x}_i), \ldots, f_m(\mathbf{x}_i)]$ is positioned below the hyperplane (denoted as $f_m = a_1 \times f_1 + a_2 \times f_2 + \cdots + a_{m-1} \times f_{m-1} + a_m$) formed by its neighboring solution set, that is, $f_m(\mathbf{x}_i) < a_1 \times f_1(\mathbf{x}_i) + a_2 \times f_2(\mathbf{x}_i) + \cdots + a_{m-1} \times f_{m-1}(\mathbf{x}_i) + a_m$.

Here, we use the example shown in Fig. 1(c) to illustrate the definition of the prominent solution. Assume a population consists of six solutions $\{A, B, C, D, E, F\}$, and their objective vectors are shown in Fig. 1(c). Since solution F has smaller and the closest value to that of solution A in the first objective, F is one neighboring solution of A on the basis of Definition 1. Similarly, we can know that solutions E, D, and C are also the neighboring solutions of A. Then, the neighboring solution set of solution A is $\{C, D, E, F\}$, and the hyperplane formed by the solution set can be obtained as $f_4 = 0.8926 - 2.6151 \times f_1 - 0.6554 \times f_2 - 5.2884 \times f_3$. Since the objective vector of solution A satisfies the relationship $0.20 < 0.8926 - 2.6151 \times 0.10 - 0.6554 \times 0.05 - 5.2884 \times 0.06 = 0.281$, we call A a prominent solution.

So far, major existing selection strategies employ either convergence first [12] or diversity first [14] strategies to balance the convergence and diversity. However, the performance of these two classes of strategies is determined by the

characteristics of MaOPs. If the problem is not difficult in terms of convergence, then the diversity is critical and the diversity-first strategy will be more effective, and vice versa. It is worth noting that assume the MaOPs are black-box problems, it is difficult to evaluate their characteristics. Thus, this paper also focuses on how to aggregate both the convergence and diversity factors into the environmental selection, rather than giving priority to either of them.

Contributions: We address the challenges in MaOEAs with the following technical contributions.

1) The prominent solutions are defined, based on the hyperplane formed by their neighboring solutions, to further distinguish nondominated solutions in a population.
2) A novel environmental selection strategy is proposed to balance the convergence and diversity. This strategy emphasizes retaining prominent solutions to strengthen the selection pressure.
3) Based on the definition of the prominent solution, and the new environmental selection strategy, we develop a hyperplane assisted evolutionary algorithm, namely, *hpaEA*, to tackle MaOPs. In the proposed algorithm, the prominent solutions are given more evolutionary efforts in each iteration, thereby generating better offsprings.
4) Extensive experiments are conducted to compare the proposed *hpaEA* with five representative algorithms in the context of 36 test instances. The experimental results demonstrate the superiority of *hpaEA*.

This paper is organized as follows. The literature on MOEAs and MaOEAs is briefly reviewed in Section II. Section III presents the algorithm *hpaEA*, followed by extensive experiments and comparison studies in Section IV. In Section **??**, the conclusions and future directions are covered.

II. Related Studies

During the past two decades, a variety of MOEAs have been put forward and improved. On the basis of their environmental selection strategies, the existing MOEAs can be roughly divided into four categories [9], [10], [15]: 1) dominance-based; 2) indicator-based; 3) decomposition-based; and 4) others.

The Pareto-dominance-based MOEAs are the representatives of dominance-based MOEAs, they first sort of all the candidate solutions in the population, on the basis of their Pareto-dominance relations, and then sort the solutions in last accepted level using a second criterion. For example, *NSGA-II* is a typical representative in this direction [16]. But, with the increase of the number of objectives, the performance of Pareto-dominance-based MOEAs deteriorates seriously due to the dominance resistance. One avenue to address this shortcoming is to modify the Pareto dominance, and a series of relevant works have been reported. For instance, Wagner *et al.* [17] proposed an ϵ-dominance-based environmental selection strategy, and used the proximity of population to the PF to guide the evolutionary process. He *et al.* [18] defined a new dominance relation using fuzzy logic to distinguish Pareto nondominated solutions. Yang *et al.* [19] developed a grid-dominance-based approach to solve MaOPs. Tian *et al.* [20] suggested a strengthened dominance relation for many-objective optimization. In addition to the modification of the Pareto dominance, another avenue are the diversity improvement strategies that consider both the convergence and diversity, and there are also many works in this respect. For instance, Adra and Fleming [21] added two diversity control mechanisms to make the *NSGA-II* suitable for solving MaOPs. Li *et al.* [22] improved the *NSGA-II* to being well suitable for MaOPs with a shift-based density measurement strategy covering both the diversity and convergence of solutions. Xiang *et al.* [23] followed the framework of *NSGA-II* to design a new strategy. In this strategy, a worse-elimination principle is used to replace the solutions with worse convergence when selecting the solutions in the last accepted level.

Indicator-based MOEAs often compare solutions using new low-dimensional indicators, rather than the original high-dimensional objective vectors. For instance, Zitzler and Künzli [24] combined decision makers' preference information to define indicator to measure solutions, and developed a framework for indicator-based MOEAs. Beume *et al.* [25] presented a hypervolume indicator-based MOEA. Lin *et al.* [26] combined both the convergence and diversity into being a fitness for each solution when updating external archive, and improved particle swarm optimization to generate a new offspring population. Tian *et al.* [27] developed a new MOEA using an improved inverted generational distance (IGD) indicator, and proposed a strategy to adjust the reference points adaptively according to the contributions of the selected solutions. Liu *et al.* [28] designed a convergence indicator and a distribution indicator for the environmental selection strategy to balance the convergence and diversity. Pamulapati *et al.* [29] suggested an indicator for solving MaOPs, and the indicator combined both the sum of objective values and shift-based density value.

Zhang and Li [30] published a decomposition-based MOEA, that is, *MOEA/D*, in 2007, this branch of MOEAs has become vigorous, and a large number of studies have been reported [31]. For instance, Molina *et al.* [32] used the reference points of decision makers to convert the MOP into a single-objective optimization problem. Wang *et al.* [33] suggested a replacement strategy that assigns the new solutions to the most suitable subproblems. Jiang *et al.* [34] developed the multiplicative and penalty-based scalarizing functions to improve the performance of decomposition-based MOEAs in making tradeoffs between diversity and convergence. Li *et al.* [35] employed the covariance matrix adaptation evolution strategy as local search in MOEA/D to deal with biased MOPs. To handle the complex PFs, Jiang and Yang [36] improved *MOEA/D* with a two-phase strategy and a niche-guided strategy. To conquer the shortcoming of *NSGA-II* in solving MaOPs, Deb and Jain [12] proposed a decomposition-based diversity-preservation operator to balance both the convergence and diversity. Wang *et al.* [37] suggested a preference-inspired strategy that adaptively adjusts the weights for decomposition-based MOEAs to solve MaOPs with intricate PFs. Asafuddoula *et al.* [38] used systematic sampling to generate uniformly distributed weights for the

decomposition-based approach to solve MaOPs. Li *et al.* [39] improved decomposition-based approaches using dominance-based strategy to achieve sound tradeoffs between convergence and diversity when solving MaOPs. Cai *et al.* [40] employed angle-based-selection to improve the performance of decomposition-based MOEA for many-objective optimization. A branch of decomposition-based MOEA is to use a set of reference vectors to partition the objective space into a series of subspaces, and select solutions from each subspace. For instance, Liu *et al.* [41] designed an algorithm, namely, *MOEA/D-M2M*, to divide the objective space into many subspaces, where each subspace retains a population. To further improve the performance of the *MOEA/D-M2M*, Liu *et al.* [42] proposed an adaptive strategy to adjust the reference vectors according to the distribution of the current population. Cheng *et al.* [11] proposed an algorithm *RVEA* that divides the solutions into a series of subspaces, and then selects a solution with the minimal penalized distance from each subspace. Jiang and Yang [14] designed a new algorithm that divides the objective space into many subspaces, and selects solutions from each subspace based on their fitness values.

There also exist substantial studies that do not fall under the above categories. For instance, Li *et al.* [43] designed an ingenious strategy to transform many-objective optimization into bi-objective optimization. Chen *et al.* [44] focused on large-scale many-objective optimization, and leveraged covariance matrix adaptation evolution strategy to drive scalable subpopulations to approach the PF. In addition, the knee points refer to the preferred tradeoff solutions, and have been used to assist Pareto-based MOEAs [45]. For instance, Du *et al.* [46] defined the knee point as the solution with the largest positive bend-angle, and then retained knee point preferentially in environmental selection. Zhang *et al.* [47] divided the solutions into many groups, and defined the knee point in each group as the solution with the minimal distance to the hyperplane formed by the extreme solutions. Based on the above description, we know that the concept of prominent solution in this paper is quite different from the concept of knee point. The prominent solutions refer to the solutions having evident tendencies toward the PF, and are identified using the hyperplane formed by their neighboring solutions.

In contrast to the aforementioned selection strategies, the environmental selection strategy proposed in this paper first retains the prominent solutions, striving to strengthen the selection pressure. Then, the proposed environmental selection strategy uses two criteria to select population according to the relationship between the population size and the count of nondominated solutions.

III. Proposed Algorithm: *hpaEA*

In this section, the core ideas of the proposed algorithm *hpaEA* are presented. On the basis of the definition of prominent solution, a new mating selection strategy, that gives more evolutionary efforts to prominent solutions at the stage of mating selection, is introduced in the main procedure of the proposed *hpaEA*. In the sequel, the novel environmental selection strategy is elaborated.

Algorithm 1: Main Procedure of the Proposed *hpaEA*

Input: MOP; maximum function evaluations (*MFEs*); population size N;
Output: The final population P;

1 Generate N uniformly distributed m-dimensional unit vectors as $V \leftarrow \{\mathbf{v}^1, \mathbf{v}^2, \cdots, \mathbf{v}^N\}$;
2 Initialize a population P randomly and the $\mathbf{z} \leftarrow \mathbf{z}^{max}$;
3 Record the used function evaluation number as $FEs \leftarrow N$;
4 Initialize the number of prominent solutions as $K \leftarrow 0$;
5 **while** $FEs < MFEs$ **do**
6 $I \leftarrow$ Randomly generate $N - K$ integers between 1 and $|P|$;
7 $I \leftarrow I \bigcup \{1, 2, \cdots, K\}$; Disorder the elements in I;
8 $P' \leftarrow GenerateOffsprings(P(I))$;
9 $Q \leftarrow P \cup P'$;
10 Remove the solutions that cannot dominate $\mathbf{z}$;
11 Update the used function evaluations as $FEs = FEs + N$, and the $\mathbf{z} \leftarrow \min\{\mathbf{z}, \mathbf{z}^{max}\}$;
12 $[P, K] \leftarrow popSelectionStrategy(Q, V, N)$;

A. Main Procedure of hpaEA

In practice, the conflicting objectives in the MaOP often have different units of measurement, and the objectives may be scaled using different ranges. Thus, a natural way to avoid the disparities in different objectives is to normalize them to be dimensionless. Similar to the studies in [23] and [37], the objective vector $F(\mathbf{x}_i)$ of a certain solution $\mathbf{x}_i$ in current population P (i.e., $\mathbf{x}_i \in P$) is normalized by

$$F'(\mathbf{x}_i) = \frac{F(\mathbf{x}_i) - \mathbf{z}^{\min}}{\mathbf{z}^{\max} - \mathbf{z}^{\min}} \tag{2}$$

where $\mathbf{z}^{\min}$ denotes the ideal point, that is, $\mathbf{z}_i^{\min} = \min_{j=1}^{|P|} f_i(\mathbf{x}_j), i \in \{1, \ldots, m\}$; $\mathbf{z}^{\max}$ represents the worst point, that is, $\mathbf{z}_i^{\max} = \max_{j=1}^{|P|} f_i(\mathbf{x}_j), i \in \{1, \ldots, m\}$.

In *hpaEA*, we employ the widely used framework of Pareto-based MOEAs, which is described as Algorithm 1.

As shown in Algorithm 1, four parameters are first initialized: 1) reference vectors V (line 1); 2) an initial population P (line 2); 3) the used function evaluations (line 3); and 4) the number of prominent solutions (line 4), followed by the main loop of the proposed *hpaEA* (lines 5–12). During each generation, the algorithm performs the following three steps: 1) mating selection; 2) offspring population generation; and 3) environmental selection. For the mating selection used in this paper, it strives to allocate more evolutionary efforts to the prominent solutions in the current population, enabling better solutions in terms of convergence being generated. Since the prominent solutions will be first selected and placed in the front of the population during environmental selection and the number of prominent solutions is denoted as K, the set $\{1, 2, \ldots, K\}$ represents the indexes of the prominent solutions. The parameter I is an array standing for the indexes of the solutions selected for mating. To form the mating pool, $N - K$ solutions in the current populations are first randomly

selected (line 6), and then all the K prominent solutions are selected (line 7) despite that some of them have been selected at random in the former step. Afterward, all the elements in the array I are shuffled (line 7). Similar to the studies [11], [12], the proposed *hpaEA* uses the simulated binary crossover and the polynomial mutation to generate the offspring population (line 8). Then, the environmental selection strategy is performed on population Q (line 12), which can balance the convergence and diversity of population while enhancing the selection pressure for pushing the population toward the PF. The environmental selection strategy is one of the main contributions in this paper, and will be elaborated in the following section.

B. Environmental Selection Approach

The purpose of the environmental selection strategy is to balance the convergence and diversity, especially to strengthen selection pressure. When the number of nondominated solutions in the combined population is larger than the population size, these nondominated solutions are incomparable based on the Pareto-dominance relations. How to distinguish these nondominated solutions becomes a challenging task. To address this issue, we attempt to first select the prominent solutions to strengthen selection pressure, which is helpful for improving the convergence. Then, other nondominated solutions will be selected based on acute angles among solutions to improve the diversity. Otherwise, if the number of nondominated solutions is less than the population size, these nondominated solutions determine the convergence of the population, and are first selected to improve the convergence, and then some dominated solutions will be selected to maintain the diversity. The details of the new environmental selection strategy are presented as Algorithm 2.

Algorithm 2: Function *popSelectionStrategy*(Q, V, N)

Input: Combined population Q; unit vector set V; population size N;
Output: The selected population P; number of prominent solutions K;

1 Initialize the number of prominent solutions as: $K \leftarrow 0$;
2 $[L_1, L_2, \cdots] \leftarrow$ Non-dominated-sort(Q);
3 **if** $|L_1| > N$ **then**
4 Initialize an empty set: $P_s \leftarrow \emptyset$;
5 Normalize all the solutions in level L_1 based on formulation (2), and denote as M;
6 **for** $j = 1 \rightarrow |L_1|$ **do**
7 $O_j \leftarrow \emptyset$;
8 **for** $i = 1 \rightarrow m$ **do**
9 $I_{1\times|L_1|} \leftarrow$ Get the arrangement of the solutions in L_1 in an ascending order on the i-th objective;
10 **for** $j = 2 \rightarrow |L_1|$ **do**
11 $l \leftarrow I(j)$; $k \leftarrow I(j-1)$; $p \leftarrow M_k$;
12 $O_l \leftarrow O_l \bigcup \{p\}$;
13 **for** $j = 1 \rightarrow |L_1|$ **do**
14 **if** O_j *contains* m *different points* **then**
15 $H \leftarrow$ Calculate the hyperplane based on the points in O_j;
16 **if** *objective vector of* $\boldsymbol{x}_j$ *is below hyperplane* H **then**
17 $P_s \leftarrow P_s \bigcup \{L_1^j\}$;
18 $[P, K] \leftarrow$ *SelectSolutions*(P_s, L_1, N);
19 **else**
20 Construct the candidate solution set as: $R \leftarrow Q \setminus L_1$;
21 $P' \leftarrow$ *MaintainDiversity*(R, V);
22 $P \leftarrow L_1 \bigcup P'$;

As described in Algorithm 2, this strategy first initializes the count of prominent solutions (line 1). Then, the combined population Q will be classified into different nondominated levels ($L_1, L_2, \ldots$) by nondominated sorting (line 2). If the number of solutions in L_1 is larger than the population size N (line 3), we propose a novel strategy aims to balance the convergence and diversity, especially strengthen the selection pressure of the population. To begin with, this strategy identifies all the prominent solutions (lines 4–17), which are helpful for enhancing selection pressure of the population toward the PF. The detailed steps to identify the prominent solutions are described as follows: 1) initialize an empty set to record the prominent solutions (line 4); 2) all the objective values of solutions in L_1 are normalized and denoted as M, where each row of M (denoted as M_i) stands for the normalized objective vector of a solution (line 5); 3) for each solution in L_1, an empty set is initialized to record the objective vectors of its neighboring solutions (lines 6 and 7); 4) find the neighboring solutions for each solution and record the corresponding objective vectors (lines 8–12); and 5) for each solution, if it has m different neighboring solutions (line 14), the hyperplane is formed (line 15) to judge whether the solution is positioned below it (line 16), if so, the solution will be regarded as a prominent solution and added to the population P_s (line 17), where L_1^j denotes the jth solution in L_1. After that, Function *SelectSolutions*() is triggered to select solutions (line 18), which is given in Algorithm 4. If the number of solutions in L_1 is less than the population size N, the solutions in the first level determine the convergence of the population, and all these solutions will be selected in (line 20). To balance the convergence and diversity, a part of dominated solutions are selected by Function *MaintainDiversity*() (line 21), which is presented in Algorithm 3.

Once the prominent solutions have been identified, the Function *SelectSolutions*() is triggered to balance the convergence and diversity, especially to prioritize the choice of prominent solution to strengthen the selection pressure. This function first evenly divides each objective variable into many intervals, and then selects a single solution from each interval. If an interval contains prominent solutions, one of the prominent solutions will be selected. After that, the other nondominated solutions are selected based on their acute cosines to selected solutions until the number of selected solutions reaches the population size. The details of this function are given in Algorithm 4.

Algorithm 3: Function *MaintainDiversity*(R, V)

Input: The candidate population R; unit vector set V;
Output: The selected population P';

1 Initialize an empty population as $P' \leftarrow \emptyset$;
2 Initialize $|V|$ empty sets as: $S_i \leftarrow \emptyset, i = 1, 2, \cdots, |V|$;
3 Normalize all the solutions in R based on formulation (2), and denote the normalized objective matrix as M;
4 **for** $j = 1 \rightarrow |R|$ **do**
5 $maxCos \leftarrow 0$; $v \leftarrow 1$;
6 **for** $i = 1 \rightarrow |V|$ **do**
7 $cos\theta_{j,i} = \frac{M_j \cdot \mathbf{v}^i}{\|M_j\| \cdot \|\mathbf{v}^i\|}$;
8 **if** $cos\theta_{j,i} > maxCos$ **then**
9 $maxCos \leftarrow cos\theta_{j,i}$; $v \leftarrow j$;
10 $S_v \leftarrow S_v \bigcup \{R_j\}$;
11 Get the worst point: $\mathbf{z}^{max}$, where $\mathbf{z}_i^{max} = \max_{j=1}^{|R|} f_i(\mathbf{x}_j), i \in \{1, \cdots, m\}$;
12 **for** $i = 1 \rightarrow |V|$ **do**
13 $maxHV \leftarrow 0$; $S \leftarrow \emptyset$;
14 **for** $j = 1 \rightarrow |S_i|$ **do**
15 Calculate the hypervolume HV between S_i^j and $\mathbf{z}^{max}$;
16 **if** $HV > maxHV$ **then**
17 $maxHV \leftarrow HV$; $S \leftarrow S_i^j$;
18 $P' \leftarrow P' \bigcup \{S\}$;

Algorithm 4: Function *SelectSolutions*(P_s, C, N)

Input: The set of prominent solutions P_s; candidate population C; population size N;
Output: The updated selected population P; The number of selected prominent solutions K;

1 $P \leftarrow \emptyset$; $T \leftarrow \lfloor \frac{N}{m} \rfloor$;
2 **for** $i = 1 \rightarrow m$ **do**
3 $z_i^{min} \leftarrow \min_{p \in L_1} f_i(p)$; $z_i^{max} \leftarrow \max_{p \in L_1} f_i(p)$;
4 $len \leftarrow (z_i^{max} - z_i^{min})/(T-1)$;
5 $S_t (1 \le t \le T) \leftarrow \emptyset$;
6 **for** $j = 1 \rightarrow |L_1|$ **do**
7 $k \leftarrow \lfloor \frac{f_i(C_j) - z_i^{min}}{len} \rfloor + 1$;
8 $S_k \leftarrow S_k \bigcup \{C_j\}$;
9 $P' \leftarrow \emptyset$;
10 **for** $t = 1 \rightarrow T$ **do**
11 **if** $S_k \bigcap P == \emptyset$ **then**
12 $P_c \leftarrow S_k \bigcap P_s$;
13 **if** $P_c != \emptyset$ **then**
14 $p \leftarrow$ Select one solution with the smallest i-th objective from P_c;
15 **else**
16 $p \leftarrow$ Select one solution with the smallest i-th objective from S_k;
17 $P' \leftarrow P' \bigcup \{p\}$;
18 $P \leftarrow P \bigcup P'$;
19 $P_s \leftarrow P_s \bigcap P$; $K \leftarrow |P_s|$;
20 Move selected prominent solutions P_s to the front of P;
21 $Q \leftarrow C \setminus P$;
22 Initialize an empty set as: $A \leftarrow \emptyset$;
23 **for** $j = 1 \rightarrow |Q|$ **do**
24 $c_j \leftarrow 0$;
25 **for** $k = 1 \rightarrow |P|$ **do**
26 Calculate the cosine between solution Q_j and P_k as $cos\theta_{j,k}$;
27 **if** $cos\theta_{j,k} > c_j$ **then**
28 $c_j \leftarrow cos\theta_{j,k}$;
29 $A \leftarrow A \bigcup \{(Q_j, c_j)\}$;
30 **for** $j = 1 \rightarrow N - |P|$ **do**
31 $V_m \leftarrow 1$; $r \leftarrow 1$;
32 **for** $(Q_j, c_j) \in A$ **do**
33 **if** $c_j < V_m$ **then**
34 $V_m \leftarrow c_j$; $r \leftarrow j$;
35 $P \leftarrow P \bigcup \{Q_r\}$;
36 $Q \leftarrow Q \setminus \{Q_r\}$;
37 $A \leftarrow A \setminus \{(Q_r, c_r)\}$;
38 **for** $i = 1 \rightarrow |Q|$ **do**
39 Calculate the cosine between Q_i and Q_r as $cos\theta_{i,r}$;
40 **if** $cos\theta_{i,r} > c_i$ **then**
41 Replace the element (Q_i, c_i) in A with $(Q_i, cos\theta_{i,r})$;

As shown in Algorithm 4, the set of selected solutions P are initialized as empty, and the number of intervals T in each objective dimension is given (line 1). For each objective dimension, the length of each interval is calculated (line 4), and then each solution in C is associated to an interval (lines 5–8), where S_t records the solutions in the t-th interval and C_j denotes the jth solution in C. The interval index of a solution is calculated in line 7. After that, if there exist solutions in this interval and none of the solutions in an interval has been selected (line 11), one solution in this interval will selected (lines 12 and 13). If this interval contains prominent solutions (line 13), the one with the minimal ith objective value will be selected (line 14). Otherwise, one solution with the minimal ith objective value in this interval will be selected (line 16). Then, the selected prominent solutions are separated as P_s and moved to the front of the selected population (line 20). Also, the number of selected prominent solutions K is updated (line 19). Next, the candidate solutions are recorded by Q (line 21), and an empty set A is initialized (line 22), where each element in A records a candidate solution in Q and its maximal cosine to all the solutions in the selected population P. Then, for each candidate solution in Q, its maximal cosine is calculated (lines 23–29), and recorded in set A, where Q_j and P_k stand for the jth candidate solution in Q and the kth solution in P, respectively. Afterward, the following steps will be iterated for $N - |P|$ times to select solutions from the candidate population Q: 1) one candidate solution with the minimal cosine Q_r is moved from Q to P (lines 31–35); 2) remove the selected solution Q_r from the population Q (line 36), and the corresponding element (Q_r, c_r) from set A (line 37); and 3) update the maximal cosine of each candidate solution in Q to the selected solutions in P (lines 38–41).

On the other hand, for the case that the number of nondominated solutions is less than the population size, that is, $|L_1| \le N$, in addition to all the nondominated solutions being selected, some dominated solutions should be selected by

Function *MaintainDiversity*() (line 21, Algorithm 2). To balance the convergence and diversity, this function divides the objective space into a series of subspaces using a set of unit vectors. Next, each dominated solution is associated to a subspace, and then one solution having the best convergence in each subspace is selected. The pseudocode of this function is described in Algorithm 3.

As shown in Algorithm 3, the selected population P' is first initialized to an empty set (line 1). Then, $|V|$ empty sets are initialized (line 2), and each set S_i is used to record the solutions associated with unit vector $\mathbf{v}^i$. Next, all the candidate solutions in R are normalized (line 3). After that, each candidate solution is associated with a unit vector that is the closest (i.e., has the maximal value of cosine) to the normalized objective vector of the solution (lines 4–10). Finally, for all the solutions associated with a unit vector, the one that has the maximal hypervolume to the worst point is selected (lines 11–18).

C. Time Complexity Analysis

During each generation, the computational cost of *hpaEA* is mainly determined by the population selection strategy, which is described in Algorithm 2. It takes $O(N\log^{m-1}N)$ to sort the nondominated solutions (line 2, Algorithm 2).

When the number of nondominated solutions $|L_1|$ is larger than the population size N, the time complexity of normalizing the solutions is $O(mN)$ (line 5, Algorithm 2). It takes $O(mN\log N)$ to select the neighboring solution set for all the solutions (lines 8–12, Algorithm 2), where the time complexity of line 9 is $O(N\log N)$. Then, since it takes $O(m^3)$ to obtain the parameters of a hyperplane (line 15, Algorithm 2), the time complexity of identifying the prominent solutions is $O(m^3N)$ (lines 13–17, Algorithm 2). For the Function *SelectSolutions*, it takes $O(mN)$ to evenly select solutions from each objective dimension (lines 2–18, Algorithm 4). Then, the time complexity of calculating the cosines among solutions is $O(mN^2)$ (lines 23–29, Algorithm 4). Besides, it costs $O(N^2)$ to select solutions (lines 30–41, Algorithm 4). Thus, the time complexity of Function *SelectSolutions* is $O(mN^2)$, which is the time complexity of line 18 of Algorithm 2. Then, the time complexity of lines 3–18 of Algorithm 2 is $O(mN + mN\log N + m^3N + mN^2) = O(m^3N + mN^2)$.

When the number of solutions in L_1 is less than the population size N, the Function *MaintainDiversity* will be used. For this function, the time complexity of normalizing the solutions is $O(mN)$ (line 3, Algorithm 3). Associating each solution to a unit vector costs $O(mN^2)$ (lines 4–10, Algorithm 3). Besides, the time complexity of selecting a solution from each unit vector is $O(mN^2)$ (lines 12–18, Algorithm 3). Thus, the time complexity of Function *MaintainDiversity* is $O(mN+mN^2+mN^2) = O(mN^2)$. Then, line 21 of Algorithm 2 requires $O(mN^2)$ operations.

In conclusion, at each generation, the time complexity of the proposed *hpaEA* is $O(N\log^{m-1}N + mN^2 + m^3N + mN^2) = O(N\log^{m-1}N + mN^2 + m^3N)$.

TABLE I
SETTING OF POPULATION SIZE

m	H	*NSGA-III*	*MOEA/D*
3	12	92	91
5	6	212	210
8	3	156	156
10	3	276	275

IV. PERFORMANCE EVALUATION

To empirically investigate the effectiveness of the proposed *hpaEA*, we compare it with the following five representative algorithms: 1) *NSGA-III* [12]; 2) *RVEA* [11]; 3) *MOEA/DD* [39]; 4) *SPEA/R* [14]; and 5) *SPEA2+SDE* [22].

The algorithms used in the comparative analysis were implemented in MATLAB, and embedded into the evolutionary multiobjective optimization platform PlatEMO, which is freely available to the public.[1] Except for special instructions, we follow the original settings of the algorithms and problems in their original literature. The proposed algorithm *hpaEA* is also coded in MATLAB and embedded into PlatEMO. All the experiments are run on a PC with 8.0-GB RAM and two Intel Xeon CPU i7-4790 and 64-bit Window 7 operating system. In addition, the MATLAB R2016a has been installed on this PC to run the experiments.

A. Experimental Settings

1) Benchmark Problems: The performance of the six algorithms are compared in the context of the nine benchmarks, that is, MaF1–MaF9 [48], with 3, 5, 8, and 10 objectives. These nine benchmarks are the improved version of the widely used benchmarks DTLZ [49], and they have more complicated properties, e.g., complicated PS, irregular PF outline, multimodal, etc. In this paper, a benchmark with a specific number of objectives is called a test instance, e.g., benchmark MaF1 with three objectives.

2) Population Size: In *MOEA/D* [30], the number of reference points of m-objective problems is suggested to set as $N' = C_{m-1}^{H+m-1}$, where H is a user-defined integer parameter and C_k^l represents the number of k-combinations from a set of l elements, that is, $C_k^l = l!/k!(l-k)!$ Then, the population size (N) is set to be the same with the number of reference points, that is, $N = N'$. Since *MOEA/DD* and *RVEA* require the reference points as *MOEA/D*, the setting of their population sizes keeps the same to *MOEA/D*.

According to [12], the population size (N) in *NSGA-III* is suggested to set as the smallest multiple of four and larger than the number of reference points (N'). The population sizes of *SPEA/R*, *SPEA2+SDE*, and *hpaEA* are set the same to *NSGA-III*. The population size N of the six algorithms in the context of the different number of objectives m is summarized in Table I.

3) Termination Condition: The maximal number of function evaluations (MFEs) is selected as the termination condition for all the six algorithms. Since benchmark MaF3 has a lot of local optimal solutions, it is difficult to solve and

[1] https://github.com/BIMK/PlatEMO

TABLE II
COMPARISON OF NHV AMONG THE SIX ALGORITHMS ON BENCHMARKS MAF1–MAF9 WITH 3, 5, 8, AND 10 OBJECTIVES

MaOP	m	*NSGA-III*	*RVEA*	*MOEA/DD*	*SPEA/R*	*SPEA2+SDE*	*hpaEA*
MaF1	3	1.863e-1 (1.49e-3)−	1.683e-1 (4.71e-4)−	1.693e-1 (9.65e-4)−	1.695e-1 (4.40e-4)−	2.006e-1 (4.37e-4)−	2.014e-1 (4.92e-4)
	5	1.645e-2 (1.01e-3)−	1.030e-2 (7.03e-4)−	1.383e-2 (2.57e-4)−	1.575e-2 (1.80e-4)−	2.751e-2 (1.79e-4)+	2.716e-2 (1.87e-4)
	8	1.166e-3 (8.96e-5)−	3.118e-4 (8.99e-5)−	4.644e-4 (4.39e-5)−	5.679e-4 (1.43e-4)−	2.489e-3 (4.71e-5)+	2.399e-3 (4.24e-5)
	10	1.033e-4 (6.56e-6)−	2.196e-5 (4.99e-6)−	2.892e-5 (3.98e-6)−	3.292e-5 (7.58e-6)−	2.672e-4 (1.46e-5)+	2.452e-4 (1.03e-5)
MaF2	3	2.220e-1 (1.29e-3)−	2.143e-1 (1.17e-3)−	1.904e-1 (4.84e-3)−	2.205e-1 (1.03e-3)−	2.291e-1 (6.39e-4)+	2.264e-1 (6.35e-4)
	5	2.348e-1 (2.55e-3)−	2.277e-1 (1.93e-3)−	2.014e-1 (4.45e-3)−	2.220e-1 (1.67e-3)−	2.604e-1 (1.32e-3)+	2.443e-1 (1.71e-3)
	8	3.428e-1 (1.18e-2)−	2.958e-1 (1.45e-2)−	3.070e-1 (5.14e-3)−	3.203e-1 (3.67e-3)−	3.681e-1 (2.44e-3)+	3.522e-1 (4.28e-3)
	10	3.581e-1 (8.58e-3)−	2.943e-1 (1.28e-2)−	3.110e-1 (4.29e-3)−	3.214e-1 (2.68e-3)−	3.699e-1 (2.36e-3)+	3.610e-1 (3.38e-3)
MaF3	3	9.546e-1 (1.14e-3)−	9.567e-1 (1.18e-3)−	9.536e-1 (1.02e-3)−	6.709e-1 (2.64e-1)−	9.449e-1 (5.88e-3)−	9.577e-1 (7.54e-4)
	5	9.988e-1 (1.55e-3)−	8.294e-1 (3.60e-1)−	9.923e-1 (2.84e-3)−	3.410e-2 (1.28e-1)−	9.927e-1 (1.77e-3)−	9.995e-1 (2.36e-4)
	8	3.973e-1 (4.90e-1)−	9.554e-1 (1.29e-1)−	9.570e-1 (1.81e-1)−	0.000e+0 (0.00e+0)−	9.967e-1 (1.01e-3)−	9.999e-1 (1.25e-4)
	10	0.000e+0 (0.00e+0)−	9.997e-1 (8.72e-4)+	9.914e-1 (3.28e-2)+	0.000e+0 (0.00e+0)−	9.988e-1 (3.67e-4)+	9.543e-1 (1.40e-1)
MaF4	3	4.984e-1 (5.88e-3)−	4.768e-1 (4.64e-2)−	4.865e-1 (1.73e-3)−	3.344e-1 (1.34e-1)−	5.096e-1 (6.15e-3)−	5.120e-1 (2.38e-3)
	5	1.109e-1 (9.57e-3)−	6.125e-2 (1.37e-2)−	6.750e-2 (7.49e-3)−	4.588e-2 (2.85e-2)−	1.393e-1 (6.17e-3)−	1.542e-1 (3.52e-3)
	8	1.509e-2 (1.16e-3)−	1.059e-3 (5.18e-4)−	1.011e-3 (1.08e-4)−	2.066e-3 (2.79e-3)−	4.312e-3 (9.25e-4)−	1.680e-2 (1.24e-3)
	10	2.970e-3 (1.12e-4)−	8.587e-5 (4.75e-5)−	7.124e-5 (1.09e-5)−	2.221e-4 (3.33e-4)−	4.180e-4 (5.30e-5)−	3.330e-3 (2.37e-4)
MaF5	3	4.686e-1 (1.02e-1)+	5.274e-1 (3.99e-2)+	4.822e-1 (8.72e-2)+	5.338e-1 (4.09e-4)+	4.746e-1 (9.36e-2)+	3.581e-1 (1.19e-1)
	5	8.620e-1 (3.97e-4)+	8.620e-1 (3.55e-4)+	7.468e-1 (2.88e-2)−	8.606e-1 (3.23e-4)+	8.415e-1 (1.46e-2)+	8.314e-1 (5.89e-2)
	8	9.833e-1 (1.43e-4)+	9.678e-1 (4.81e-2)−	7.842e-1 (3.79e-2)−	9.816e-1 (3.47e-4)+	9.599e-1 (6.20e-3)−	9.689e-1 (1.73e-2)
	10	9.948e-1 (3.19e-3)+	9.918e-1 (4.82e-4)≈	8.015e-1 (9.86e-3)−	9.905e-1 (1.57e-3)−	9.796e-1 (8.33e-3)−	9.914e-1 (9.62e-4)
MaF6	3	1.725e-1 (1.21e-3)−	1.583e-1 (1.73e-3)−	1.621e-1 (4.76e-4)−	1.611e-1 (3.91e-3)−	1.790e-1 (2.07e-4)−	1.795e-1 (8.34e-5)
	5	1.850e-1 (1.76e-3)−	1.732e-1 (2.16e-3)−	1.719e-1 (1.39e-3)−	1.716e-1 (9.84e-4)−	1.870e-1 (3.58e-4)−	1.878e-1 (4.08e-4)
	8	2.082e-1 (9.72e-2)−	2.566e-1 (1.78e-3)−	2.551e-1 (1.04e-3)−	2.522e-1 (4.17e-3)−	2.366e-1 (6.91e-2)−	2.674e-1 (5.06e-4)
	10	4.578e-2 (7.44e-2)−	2.528e-1 (1.16e-3)−	2.451e-1 (9.42e-3)−	2.316e-1 (6.30e-2)−	2.018e-2 (6.59e-2)−	2.606e-1 (5.14e-4)
MaF7	3	2.521e-1 (5.66e-3)+	2.457e-1 (1.17e-3)−	1.989e-1 (1.99e-2)−	2.500e-1 (7.28e-4)≈	2.538e-1 (1.08e-2)+	2.495e-1 (1.71e-2)
	5	3.191e-1 (3.61e-3)−	2.848e-1 (3.29e-3)−	1.521e-1 (2.59e-2)−	3.136e-1 (1.43e-3)−	3.422e-1 (2.77e-3)+	3.277e-1 (4.86e-3)
	8	3.784e-1 (6.78e-3)−	3.224e-1 (1.35e-2)−	4.286e-2 (5.17e-2)−	3.724e-1 (5.02e-3)−	2.336e-1 (6.59e-2)−	3.872e-1 (2.15e-3)
	10	3.794e-1 (3.23e-3)+	2.996e-1 (1.22e-2)−	1.861e-3 (5.01e-4)−	3.533e-1 (5.20e-3)−	1.079e-1 (1.47e-2)−	3.671e-1 (3.28e-3)
MaF8	3	2.416e-1 (3.55e-3)−	2.234e-1 (3.06e-3)−	2.308e-1 (2.99e-3)−	6.801e-2 (9.54e-2)−	2.532e-1 (4.64e-3)+	2.503e-1 (2.60e-2)
	5	1.375e-1 (3.35e-3)−	1.037e-1 (5.52e-3)−	1.088e-1 (3.99e-3)−	1.851e-2 (3.53e-2)−	1.595e-1 (6.51e-4)−	1.599e-1 (1.99e-3)
	8	6.791e-2 (2.65e-3)−	3.820e-2 (4.88e-3)−	4.657e-2 (7.19e-4)−	1.004e-2 (1.43e-2)−	8.252e-2 (3.93e-4)+	8.176e-2 (2.69e-3)
	10	3.292e-2 (1.07e-3)−	1.512e-2 (2.02e-3)−	1.998e-2 (5.00e-4)−	4.163e-3 (6.42e-3)−	3.884e-2 (1.55e-4)−	3.891e-2 (2.21e-4)
MaF9	3	8.281e-1 (5.09e-4)+	8.216e-1 (2.79e-2)+	8.282e-1 (3.27e-4)+	5.290e-1 (3.76e-2)−	8.112e-1 (4.55e-3)+	8.087e-1 (4.42e-2)
	5	3.005e-1 (3.71e-2)−	2.836e-1 (1.12e-2)−	3.082e-1 (2.15e-3)−	1.255e-1 (5.10e-2)−	3.783e-1 (1.15e-3)+	3.708e-1 (2.49e-2)
	8	5.901e-2 (3.27e-2)+	5.988e-2 (7.93e-3)+	9.141e-2 (4.51e-3)+	2.392e-3 (4.97e-3)−	1.374e-1 (4.56e-4)+	3.496e-2 (4.48e-2)
	10	3.664e-2 (6.83e-3)−	2.176e-2 (4.96e-3)−	3.032e-2 (5.73e-4)−	9.732e-3 (5.99e-3)−	6.818e-2 (3.00e-4)+	4.626e-2 (1.70e-2)
+/ − / ≈		8 / 28 / 0	5 / 30 / 1	4 / 32 / 0	3 / 32 / 1	18 / 18 / 0	

the MFEs of the six algorithms to solve the test instances from this benchmark are set to be 150 000. For other benchmarks, that is, MaF1–MaF2 and MaF4–MaF9, with 3, 5, 8, and 10 objectives, the MFEs of the six algorithms are set to be 100 000.

4) Performance Metric: The two widely used indicators, that is, hypervolume (HV) [50] and IGD [51], [52], are chosen for comparison.

1) The HV refers to the volume of the space decided by a reference point and a set of solutions in objective space, is a prevalent metric to measure the performance of MOEAs, since it can simultaneously reflect both convergence and diversity of the population obtained by the MOEAs. How to specify the reference point, the work in [53] has reported comprehensive discussions, and gave the following suggestions: a) all the obtained solutions are normalized by the ideal point and the worst point of the PF and b) then the reference point $\mathbf{r}_{1\times m} = \{r, r, \ldots, r\}$ is suggested as $r = 1 + 1/H$, where H satisfies that $C_{m-1}^{H+m-1} \leq N < C_{m-1}^{H+m}$. On the basis of $N = C_{m-1}^{H+m-1}$ and the above setup of population sizes, we can obtain the parameter H for problems with different number of objectives, which are displayed in the second column of Table I. Thus, the parameter $r = 1 + 1/H$ in reference point is set as: 1.08 for 3-objectives, 1.17 for 5-objectives, 1.33 for 8-objectives, and 1.33 for 10-objectives. Since the HV indicator is calculated for normalized solutions, we call this indicator NHV. On the basis of the reference points and the normalized solutions, for test instances with 3 or 5 objectives, the exact NHV values are calculated directly. For 8- and 10-objective test instances, the Monte Carlo method with 1 000 000 sampling points is used to approximate the exact NHV values. Note that larger values of NHV imply better performance considering both the convergence and diversity.
2) The metric IGD can be calculated as follows. We sample a set of uniformly distributed points on the PF, and denote them as P^*. Suppose the objective vectors of the

TABLE III
COMPARISON OF IGD AMONG THE SIX ALGORITHMS ON BENCHMARKS MAF1–MAF9 WITH 3, 5, 8, AND 10 OBJECTIVES

MaOP	m	*NSGA-III*	*RVEA*	*MOEA/DD*	*SPEA/R*	*SPEA2+SDE*	*hpaEA*
MaF1	3	6.184e-2 (1.67e-3)−	8.202e-2 (8.78e-4)−	8.139e-2 (1.10e-3)−	8.103e-2 (5.18e-4)−	4.434e-2 (7.48e-4)−	4.320e-2 (5.20e-4)
	5	1.897e-1 (1.28e-2)−	2.727e-1 (1.45e-2)−	2.096e-1 (2.49e-3)−	1.888e-1 (1.24e-3)−	1.060e-1 (7.16e-4)+	1.089e-1 (7.73e-4)
	8	2.726e-1 (1.29e-2)−	5.738e-1 (7.25e-2)−	4.404e-1 (1.45e-2)−	3.853e-1 (4.51e-2)−	2.097e-1 (5.37e-4)+	2.104e-1 (1.54e-3)
	10	2.619e-1 (6.05e-3)−	5.775e-1 (6.52e-2)−	4.767e-1 (1.72e-2)−	4.720e-1 (5.56e-2)−	2.222e-1 (1.37e-3)−	2.180e-1 (1.24e-3)
MaF2	3	3.648e-2 (8.03e-4)−	4.292e-2 (1.11e-3)−	5.570e-2 (3.11e-3)−	3.788e-2 (5.94e-4)−	3.262e-2 (6.46e-4)−	3.221e-2 (5.81e-4)
	5	1.146e-1 (2.95e-3)−	1.158e-1 (7.37e-4)−	1.336e-1 (3.67e-3)−	1.186e-1 (8.11e-4)−	9.713e-2 (1.82e-3)−	9.451e-2 (1.59e-3)
	8	2.121e-1 (3.33e-2)+	2.264e-1 (5.08e-2)+	2.161e-1 (1.36e-2)+	2.035e-1 (9.75e-4)+	1.599e-1 (3.30e-3)+	2.786e-1 (2.09e-2)
	10	2.015e-1 (1.52e-2)+	2.522e-1 (3.28e-2)+	2.632e-1 (2.37e-2)+	2.072e-1 (5.84e-3)+	1.656e-1 (3.09e-3)+	3.180e-1 (1.92e-2)
MaF3	3	4.718e-2 (9.02e-4)−	4.136e-2 (1.05e-3)−	5.277e-2 (1.13e-3)−	3.777e-1 (6.42e-1)−	6.185e-2 (1.01e-2)−	3.756e-2 (1.26e-3)
	5	8.088e-2 (1.17e-2)−	5.358e-1 (1.02e+0)−	1.008e-1 (4.09e-3)−	2.186e+1 (7.38e+1)−	8.292e-2 (7.34e-3)−	6.227e-2 (5.93e-3)
	8	1.725e+2 (5.82e+2)−	1.661e-1 (1.43e-1)−	2.593e-1 (7.44e-1)−	1.427e+4 (1.78e+4)−	1.172e-1 (9.66e-3)−	8.470e-2 (4.23e-3)
	10	2.015e-1 (1.52e-2)+	2.522e-1 (3.28e-2)+	2.632e-1 (2.37e-2)+	2.072e-1 (5.84e-3)+	1.048e-1 (4.65e-3)+	3.180e-1 (1.92e-2)
MaF4	3	3.484e-1 (1.72e-2)−	4.489e-1 (1.88e-1)−	5.540e-1 (8.98e-3)−	1.050e+0 (5.71e-1)−	5.065e-1 (7.41e-2)−	3.330e-1 (9.37e-3)
	5	2.430e+0 (1.75e-1)−	3.445e+0 (6.04e-1)−	5.649e+0 (3.38e-1)−	6.103e+0 (2.48e+0)−	2.898e+0 (2.88e-1)−	1.953e+0 (5.39e-2)
	8	2.896e+1 (2.47e+0)−	5.562e+1 (1.94e+1)−	9.672e+1 (3.09e+0)−	1.144e+2 (1.26e+2)−	4.096e+1 (6.21e+0)−	1.726e+1 (9.70e-1)
	10	9.521e+1 (1.01e+1)−	1.974e+2 (4.97e+1)−	3.889e+2 (1.48e+1)−	3.900e+2 (4.16e+2)−	1.425e+2 (1.45e+1)−	4.883e+1 (2.28e+0)
MaF5	3	7.022e-1 (7.02e-1)+	3.011e-1 (2.27e-1)+	5.378e-1 (5.59e-1)+	2.602e-1 (5.73e-4)+	7.875e-1 (6.93e-1)+	1.583e+0 (1.14e+0)
	5	1.969e+0 (3.25e-3)+	1.970e+0 (2.61e-3)+	4.094e+0 (2.81e-1)−	1.974e+0 (6.22e-3)+	2.184e+0 (2.68e-1)+	2.349e+0 (1.20e+0)
	8	2.120e+1 (2.06e-1)−	2.458e+1 (5.44e+0)−	7.895e+1 (2.49e+0)−	2.155e+1 (2.83e-1)−	1.672e+1 (6.96e-1)+	1.868e+1 (4.03e+0)
	10	7.953e+1 (8.13e+0)−	9.045e+1 (9.55e+0)−	2.928e+2 (1.31e+1)−	7.976e+1 (5.59e+0)−	6.460e+1 (3.79e+0)−	6.023e+1 (2.50e+0)
MaF6	3	1.546e-2 (1.51e-3)−	3.740e-2 (5.17e-3)−	3.250e-2 (1.00e-3)−	3.689e-2 (4.96e-3)−	1.082e-2 (9.37e-4)−	5.130e-3 (1.09e-4)
	5	1.781e-2 (1.54e-2)−	7.387e-2 (1.08e-2)−	6.648e-2 (5.58e-3)−	8.901e-2 (1.22e-2)−	5.169e-3 (4.97e-4)−	2.589e-3 (8.00e-5)
	8	2.787e-1 (5.69e-1)−	9.659e-2 (2.35e-2)−	1.116e-1 (8.12e-3)−	1.590e-1 (9.31e-2)−	7.094e-3 (1.22e-3)−	4.814e-3 (6.09e-4)
	10	5.365e-1 (2.00e-1)−	9.908e-2 (2.17e-2)−	9.759e-2 (2.12e-2)−	3.012e+0 (1.42e+1)−	9.699e-1 (5.44e-1)−	2.698e-3 (1.17e-4)
MaF7	3	8.540e-2 (5.02e-2)+	1.052e-1 (1.11e-3)+	4.685e-1 (2.42e-1)−	9.585e-2 (1.35e-3)+	9.444e-2 (9.30e-2)+	1.166e-1 (1.21e-1)
	5	2.806e-1 (1.06e-2)≈	4.995e-1 (9.45e-3)−	2.834e+0 (6.33e-1)−	3.562e-1 (5.10e-3)−	2.747e-1 (2.48e-2)+	2.811e-1 (4.86e-2)
	8	7.568e-1 (5.13e-2)−	1.585e+0 (2.89e-1)−	1.725e+0 (5.00e-1)−	1.171e+0 (2.29e-2)−	7.725e-1 (1.20e-3)−	7.421e-1 (1.80e-2)
	10	1.171e+0 (1.21e-1)≈	1.920e+0 (4.39e-1)−	2.418e+0 (3.06e-1)−	2.096e+0 (4.95e-2)−	8.846e-1 (5.83e-2)+	1.185e+0 (1.45e-1)
MaF8	3	1.121e-1 (7.75e-3)−	1.415e-1 (7.26e-3)−	1.341e-1 (6.69e-3)−	5.061e+2 (8.71e+2)−	8.722e-1 (1.44e-2)−	1.018e-1 (9.35e-2)
	5	1.702e-1 (1.28e-2)−	3.191e-1 (2.75e-2)−	2.853e-1 (1.66e-2)−	8.017e+2 (1.05e+3)−	8.403e-2 (3.80e-3)−	8.165e-2 (1.42e-2)
	8	3.365e-1 (4.19e-2)−	7.720e-1 (1.04e-1)−	7.399e-1 (1.78e-2)−	8.923e+2 (1.36e+3)−	1.450e-1 (6.67e-3)+	1.575e-1 (5.47e-2)
	10	3.117e-1 (6.32e-2)−	9.202e-1 (1.13e-1)−	9.142e-1 (2.12e-2)−	9.263e+2 (1.03e+3)−	1.154e-1 (4.26e-3)−	1.072e-1 (3.23e-3)
MaF9	3	6.207e-2 (3.44e-4)+	7.055e-2 (4.06e-2)+	6.195e-2 (1.23e-4)+	4.809e-1 (2.83e-2)−	6.584e-2 (2.05e-3)+	9.286e-2 (7.75e-2)
	5	2.541e-1 (1.09e-1)−	2.725e-1 (2.26e-2)−	2.253e-1 (3.15e-3)−	8.783e-1 (3.02e-1)−	7.164e-2 (6.19e-4)+	9.407e-2 (5.95e-2)
	8	1.020e+0 (9.14e-1)+	6.685e-1 (1.12e-1)+	3.800e-1 (2.48e-2)+	4.332e+0 (1.64e+0)−	1.114e-1 (2.03e-4)+	2.122e+0 (1.55e+0)
	10	5.361e-1 (1.71e-1)−	8.777e-1 (2.08e-1)−	5.929e-1 (9.10e-3)−	1.820e+0 (1.24e+0)−	9.646e-2 (6.13e-4)+	4.054e-1 (2.95e-1)
+/ − / ≈		8 / 26 / 2	8 / 28 / 0	6 / 30 / 0	6 / 30 / 0	16 / 20 / 0	

output population is P. Then, the IGD is calculated as follows:

$$\mathrm{IGD}(P^*, P) = \frac{\sum_{v \in P^*} D_{\min}(v, P)}{|P^*|} \tag{3}$$

where $D_{\min}(v, P)$ represents the minimum Euclidean distance from the point v to all the points belonging to P. From (3), we know that the metric IGD can also quantify both the convergence and diversity, and smaller values of IGD imply better performance of the corresponding algorithms. In the experiments, the default settings in the platform PlatEMO is employed to sample 8000 points for calculating IGD.

For each test instance, the mean and standard deviation of NHV and IGD over 30 independently runs for the six algorithms are reported and analyzed.

B. Experimental Results and Analysis

The statistical results of the six algorithms realized in the context of nine benchmarks are summarized in Tables II and III. The results refer to the mean and standard deviation (in parentheses) of the NHV and IGD of the final populations. For each test instance, the best result (the largest NHV value and lowest IGD value) among the six algorithms is highlighted in gray. Similar to [11], we also choose the Wilcoxon rank-sum test with $\alpha = 0.05$ to verify significant differences between the results produced by proposed *hpaEA* and those of the compared algorithms. The symbols $+$, $-$, and $\approx$ represent that the metric value of the corresponding algorithm is significantly better than, worse than, and similar to the proposed *hpaEA*, respectively.

As shown in Table II, among the nine 3-objective test instances, the proposed *hpaEA* performs significantly better than the first four compared algorithms (i.e., *NSGA-III*, *RVEA*, *MOEA/DD*, and *SPEA/R*) on the six test instances, while the comparative algorithms *NSGA-III*, *MOEA/DD*, and *SPEA/R*, exhibits better performance on the other three test instances, that is, 3-objective MaF5, MaF7, and MaF9. For the 5-objective test instances, *hpaEA* outperforms the first four compared algorithms on 8 out of 9 test instances. The similar superiority of *hpaEA* can also be observed in the 8- and 10-objective test instances. The compared algorithm *SPEA2+SDE*

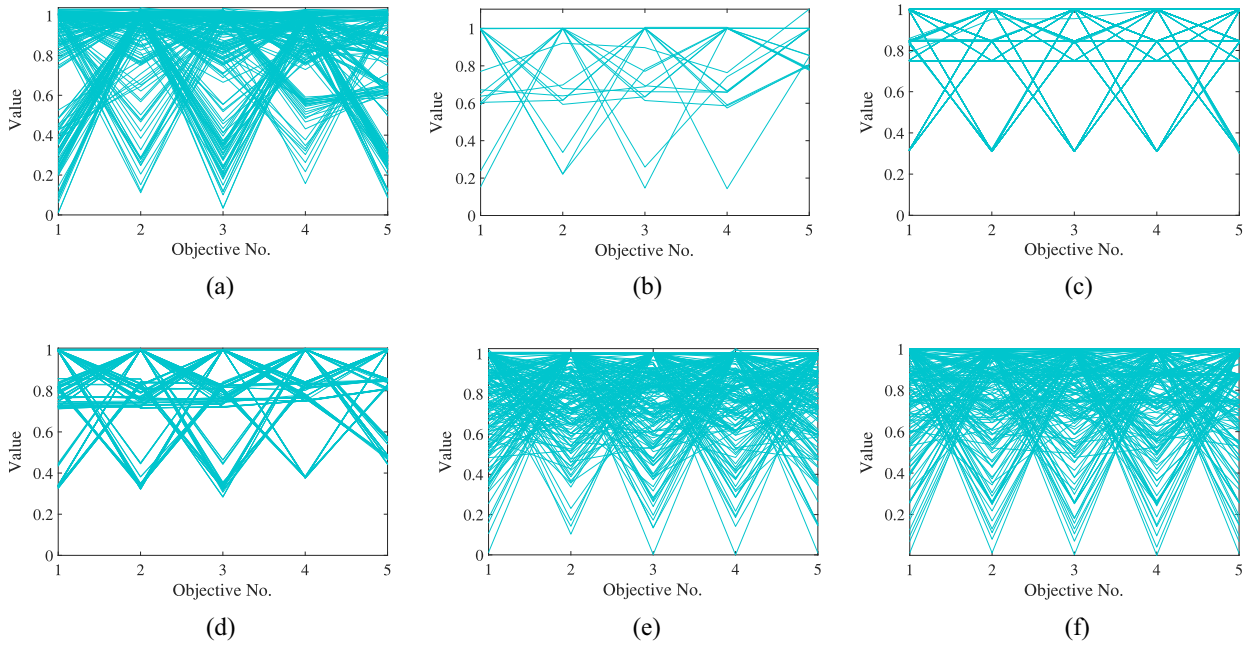


Fig. 2. Output populations of six algorithms on 5-objective MaF1. (a) *NSGA-III* on MaF1. (b) *RVEA* on MaF1. (c) *MOEA/DD* on MaF1. (d) *SPEA/R* on MaF1. (e) *SPEA2+SDE* on MaF1. (f) *hpaEA* on MaF1.

has comparative performance in dealing with complicated PF shapes of MaOPs. Comparing with *SPEA2+SDE*, the NHV values of the proposed *hpaEA* are significantly better than it on 18 out of the 36 test instances, and are very near to it on seven test instances coming from MaF1 and MaF2. In summary, with respect to the NHV metric, the proposed *hpaEA* has better performance on 18 out of the 36 test instances. More details, *hpaEA* significantly outperforms *NSGA-III*, *RVEA*, *MOEA/DD*, *SPEA/R*, and *SPEA2+SDE* on 28, 30, 32, 32, and 18 out of the 36 test instances, respectively.

From Table III, we observe that the proposed *hpaEA* also shows promising performance. In terms of IGD metric, the *hpaEA* significantly outperforms the five compared algorithms on 20 out of the 36 test instances. In more detail, *hpaEA* significantly outperforms *NSGA-III*, *RVEA*, *MOEA/DD*, *SPEA/R*, and *SPEA2+SDE* on 26, 28, 30, 30, and 20 out of the 36 test instances, respectively.

Such superior performance of *hpaEA* can be attributed to the following facts. The proposed *hpaEA* leverages the hyperplane to distinguish the nondominated solutions to strengthen the selection stress, which is helpful for improving the convergence of *hpaEA*. After the prominent solutions have been identified, the interval between the minimal and maximal values of each objective is divided into many subintervals, and then a single solution coming from each subinterval is selected to maintain the diversity, especially prioritizing the choice of prominent solution to strengthen the selection pressure. In addition, acute angles among solutions are also exploited to select nondominated solutions to guarantee that the number of selected solutions is equal to the population size.

From the compared results reported in Table II, we can observe that *hpaEA* never surpasses all the five compared algorithms on the four test instances coming from benchmark MaF5. These test instances are easy for all the six algorithms to converge to the PF. In addition, due to the normalized PFs of these test instances are similar to the hyperplane that the reference vectors are produced, each reference vector of algorithms *NSGA-III*, *RVEA*, *MOEA/DD*, and *SPEA/R* is associated with one single best solution, which is helpful for obtaining the well-distributed solutions covering the PFs. Thus, these compared algorithms generate larger NHV than *hpaEA*. It is worth noting that with the increase in the number of objectives, the NHV of *hpaEA* becomes closer or even similar to that of the compared algorithms.

To visually compare the performance of the six algorithms, their output populations with the largest NHV values among the 30 runs on 5-objective MaF1 and MaF3 are plotted in parallel coordinates, which are illustrated in Figs. 2 and 3.

For 5-objective MaF1, the interval of its PF in each dimension is between 0 and 1 [48]. From Fig. 2, we can observe that all the algorithms can converge to the PF. While for the diversity, the three algorithms *RVEA*, *MOEA/DD*, and *SPEA/R* have poor performance in all the five objectives. The algorithms *NSGA-III*, *SPEA2+SDE*, and *hpaEA* have the similar diversity. On the basis of the NHV and IGD values on 5-objective MaF1 in Tables II and III, we can know that with respect to both the convergence and diversity, the proposed *hpaEA* outperforms *NSGA-III*. Although the proposed *SPEA2+SDE* performs very well on MaOPs with complicated PF shapes [15], the proposed *hpaEA* is very near to it considering both the convergence and diversity. It is worth noting that despite all the algorithms has the same population size, the count of objective vectors of *hpaEA* seems to be more than that of *RVEA* and *MOEA/DD*. This is because *RVEA* selects at most one solution to a reference vector, and some reference vectors may be associated with no solution. Besides, some solutions in *MOEA/DD* are

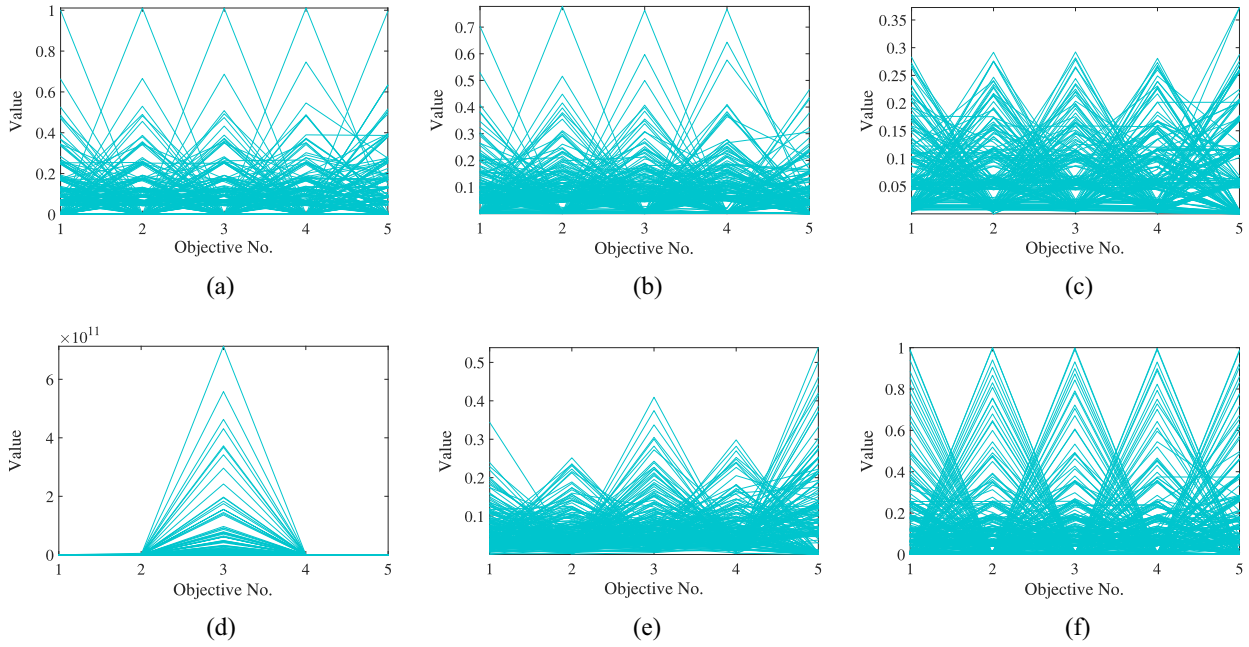


Fig. 3. Output populations of six algorithms on 5-objective MaF3. (a) *NSGA-III* on MaF3. (b) *RVEA* on MaF3. (c) *MOEA/DD* on MaF3. (d) *SPEA/R* on MaF3. (e) *SPEA2+SDE* on MaF3. (f) *hpaEA* on MaF3.

very close in the objective space, and the dots representing these solutions are overlapped. The experimental results in Fig. 2 demonstrate that the proposed *hpaEA* has promising performance in balancing convergence and diversity compared with the five existing algorithms.

For the PF of test instance 5-objective MaF3, it ranges from 0 to 1 in each dimension [48]. As illustrated in Fig. 3, some solutions of *SPEA/R* do not converge on the third objective. This is because benchmark MaF3 is a multimodal problem with a series of local optima, and *SPEA/R* falls into the local optima. For the other five algorithms, they can escape from local optima and converge to the PF. But, they exhibit great differences in terms of diversity. For *RVEA*, the values of its population in each objective dimension distribute between [0, 0.8], rather than 0 and 1. Comparing with Fig. 3(b) and (c), it is obvious that *MOEA/DD* is much worse than *RVEA* in terms of diversity. As shown in Fig. 3(e), the values of the population in each objective dimension distribute between [0, 0.55], rather than 0 and 1, thus the diversity performance of *SPEA2+SDE* is very poor. Comparing the algorithms *RVEA*, *MOEA/DD*, and *SPEA2+SDE*, we can find that the proposed *hpaEA* has the best diversity on all the five objectives. In addition, the *NSGA-III* and the proposed *hpaEA* have the similar diversity, since the values of their population range from 0 to 1. According to the results on 5-objective MaF3 in Tables II and III, we can further conclude that the proposed *hpaEA* has better diversity. This can be explained by the fact that *hpaEA* evenly divides the interval between minimal and maximal values of each objective into many subintervals, and selects solution from each subinterval, which is beneficial for diversity.

The benchmark MaF8 [48] is designed to calculate the Euclidean distance from a 2-D point $\mathbf{x}$ to a set of m 2-D vertexes of a polygon, where m refers to the number of objectives. The optimization goal of MaF8 is to minimize the m Euclidean distance simultaneously. The distributions of the 2-D decision variables of the six algorithms on 5-objective MaF8 are illustrated in Fig. 4. It is intuitive that the diversity performance of algorithms *NSGA-III*, *RVEA*, *MOEA/DD*, and *SPEA/R* are very poor, and many solutions are crowded in several subregions. By contrast, the *SPEA2+SDE* and the *hpaEA* have better diversity. Regarding the NHV and IGD values of these two algorithms on 5-objective MaF8, we can find that the proposed *hpaEA* has better performance with respect to both the convergence and diversity. This result demonstrates the effectiveness of the proposed *hpaEA* once again.

C. Distribution of Prominent Solutions

To verify the performance of the proposed *hpaEA* in identifying prominent solutions from nondominated solutions, we determined statistics on the number of prominent solutions during the whole evolutionary process. The results are shown in Fig. 5. The three subgraphs in Fig. 5 correspond to the statistical results obtained by *hpaEA* when solving the benchmark MaF1, MaF2, and MaF3, respectively.

In Fig. 5, the maximum number of evolutionary generations are quite different when the proposed algorithm solving test instances with different objectives. This is mainly because the population size for solving the test instances with different objectives varies. Besides, we can note that the fluctuation curve does not start from the first generation, which is in particular obvious for the 3-objective MaF3. This can be explained as follows. For the proposed *hpaEA*, only when the number of nondominated solutions is greater than the population size, this will identify the prominent solutions from the nondominated solutions. Once the number of nondominated

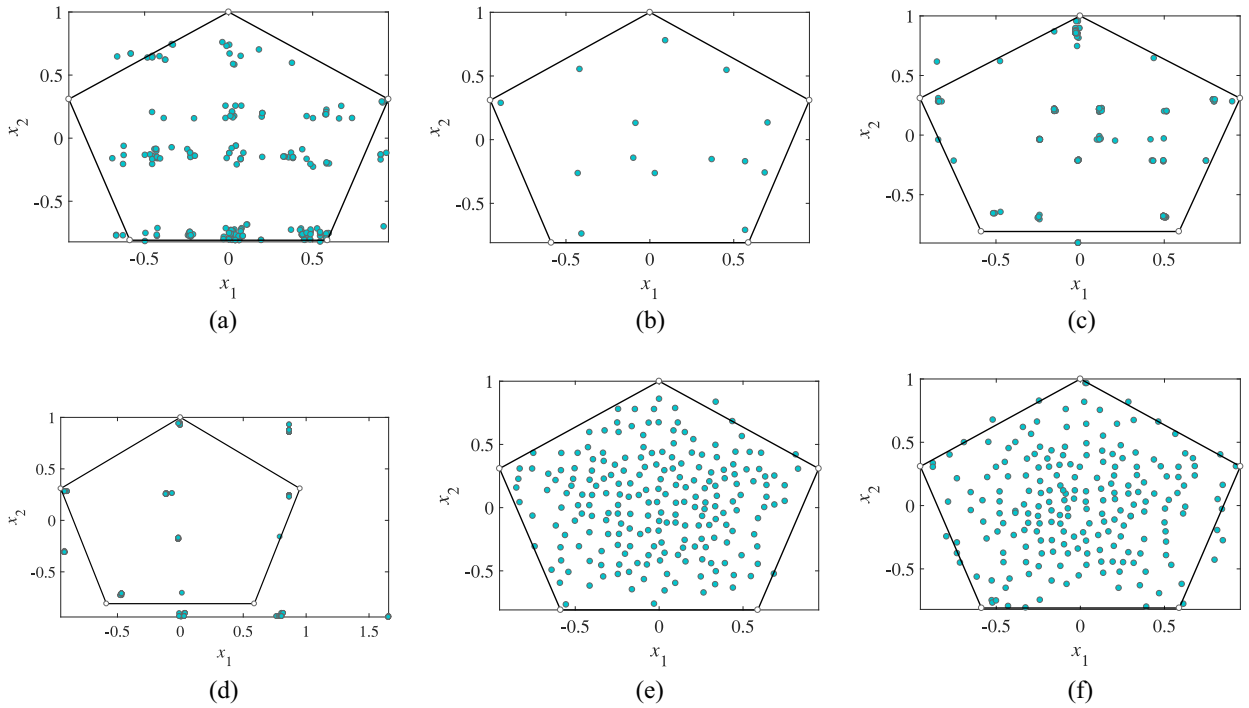


Fig. 4. Populations with the best NHV values for the six algorithms in the context of 5-objective MaF8. (a) *NSGA-III* on MaF8. (b) *RVEA* on MaF8. (c) *MOEA/DD* on MaF8. (d) *SPEA/R* on MaF8. (e) *SPEA2+SDE* on MaF8. (f) *hpaEA* on MaF8.

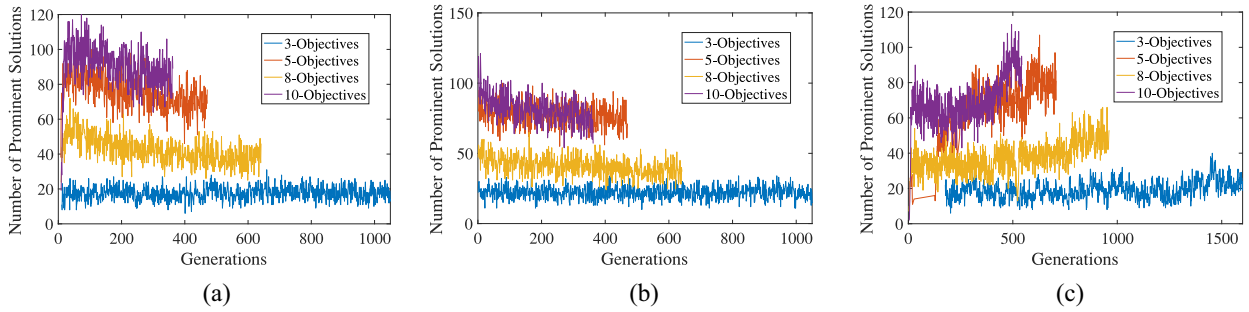


Fig. 5. Distribution of the prominent solutions over the evolutionary process. (a) MaF1. (b) MaF2. (c) MaF3.

solutions is lower than the population size, the proposed algorithm is not triggered to identify prominent solutions, especially at the early stages of the evolutionary process.

From Fig. 5, we can observe that the number of prominent solutions fluctuates during the evolutionary process. The results in this figure demonstrate that the proposed *hpaEA* can further distinguish nondominated solutions when the number of nondominated solutions is larger than the population size. Moreover, the proportions of prominent solutions to population size are 0.1898, 0.3567, 0.2725, and 0.3285 for 3, 5, 8, and 10-objective MaF1, respectively. Similar trend also appears when solving the other two benchmarks, that is, MaF2 and MaF3. These results show that the proposed algorithm can handle the optimization problems with many objectives.

V. Conclusion

This paper focuses on balancing the convergence and diversity for MaOEAs, while strengthening selection pressure for their populations toward the PFs. We first defines the prominent solution to further distinguish among nondominated solutions, which is helpful for strengthening selection pressure. Then, a novel environmental selection strategy is proposed to balance the convergence and diversity. In the proposed selection strategy, if the number of nondominated solutions is larger than the population size, all the prominent solutions will be first identified to enhance the selection pressure, then a minimum-cosine-first rule is designed to select other nondominated solutions for balancing the convergence and diversity. Otherwise, all the nondominated solutions are first selected to promote the convergence, and then reference-based selection operator is employed to select other dominated solutions to improve the diversity. On the basis of the above definition and selection strategy, we propose an MaOEA *hpaEA*. To test the performance improvement of the proposed *hpaEA*, extensive experiments are conducted to compare it with the five existing algorithms in the context of 36 test instances. The experimental results demonstrate that the proposed *hpaEA* is

comparative to the five compared algorithms with respect to both the NHV and IGD metrics.

Many real-world optimization problems involve the scalability to both objectives and decision variables [54]. However, the solution spaces of the optimization problems expand exponentially with the number of decision variables increasing, which dramatically deteriorates the performance of evolutionary algorithms. To tackle this challenge, we plan to investigate algorithms to solve MaOPs with large-scale decision variables.

REFERENCES

[1] H. Chen, X. Zhu, D. Qiu, L. Liu, and Z. Du, "Scheduling for workflows with security-sensitive intermediate data by selective tasks duplication in clouds," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 9, pp. 2674–2688, Sep. 2017.

[2] R. J. Lygoe, M. Cary, and P. J. Fleming, "A real-world application of a many-objective optimisation complexity reduction process," in *Proc. Int. Conf. Evol. Multi Crit. Optim.*, 2013, pp. 641–655.

[3] H. Chen *et al.*, "An adaptive resource allocation strategy for objective space partition based multiobjective optimization," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published. doi: 10.1109/TSMC.2019.2898456.

[4] Z. Wang, Y.-S. Ong, and H. Ishibuchi, "On scalable multiobjective test problems with hardly-dominated boundaries," *IEEE Trans. Evol. Comput.*, to be published. doi: 10.1109/TEVC.2018.2844286.

[5] M. Farina and P. Amato, "On the optimal solution definition for many-criteria optimization problems," in *Proc. IEEE Annu. Meeting North Amer. Fuzzy Inf. Process. Soc.*, 2002, pp. 233–238.

[6] H. Xu, W. Zeng, X. Zeng, and G. G. Yen, "An evolutionary algorithm based on Minkowski distance for many-objective optimization," *IEEE Trans. Cybern.*, to be published. doi: 10.1109/TCYB.2018.2856208.

[7] R. Wang, Z. Zhou, H. Ishibuchi, T. Liao, and T. Zhang, "Localized weighted sum method for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 22, no. 1, pp. 3–18, Feb. 2018.

[8] M. Ming, R. Wang, Y. Zha, and T. Zhang, "Pareto adaptive penalty-based boundary intersection method for multi-objective optimization," *Inf. Sci.*, vol. 414, pp. 158–174, Nov. 2017.

[9] A. Zhou *et al.*, "Multiobjective evolutionary algorithms: A survey of the state of the art," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 32–49, 2011.

[10] B. Li, J. Li, K. Tang, and X. Yao, "Many-objective evolutionary algorithms: A survey," *ACM Comput. Surveys*, vol. 48, no. 1, p. 13, 2015.

[11] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A reference vector guided evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 20, no. 5, pp. 773–791, Oct. 2016.

[12] 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.

[13] M. Li, L. Zhen, and X. Yao, "How to read many-objective solution sets in parallel coordinates," *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 88–100, Nov. 2017.

[14] S. Jiang and S. Yang, "A strength Pareto evolutionary algorithm based on reference direction for multi-objective and many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 21, no. 3, pp. 329–346, Jun. 2017.

[15] K. Li, R. Wang, T. Zhang, and H. Ishibuchi, "Evolutionary many-objective optimization: A comparative study of the state-of-the-art," *IEEE Access*, vol. 6, pp. 26194–26214, 2018.

[16] K. Deb, A. Pratap, S. Agarwal, 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.

[17] M. Wagner, K. Bringmann, T. Friedrich, and F. Neumann, "Efficient optimization of many objectives by approximation-guided evolution," *Eur. J. Oper. Res.*, vol. 243, no. 2, pp. 465–479, 2015.

[18] Z. He, G. G. Yen, and J. Zhang, "Fuzzy-based Pareto optimality for many-objective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 18, no. 2, pp. 269–285, Apr. 2014.

[19] S. Yang, M. Li, X. Liu, and J. Zheng, "A grid-based evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 17, no. 5, pp. 721–736, Oct. 2013.

[20] Y. Tian, R. Cheng, X. Zhang, Y. Su, and Y. Jin, "A strengthened dominance relation considering convergence and diversity for evolutionary many-objective optimization," *IEEE Trans. Evol. Comput.*, to be published. doi: 10.1109/TEVC.2018.2866854.

[21] S. F. Adra and P. J. Fleming, "Diversity management in evolutionary many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 15, no. 2, pp. 183–195, Apr. 2011.

[22] M. Li, S. Yang, and X. Liu, "Shift-based density estimation for Pareto-based algorithms in many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 18, no. 3, pp. 348–365, Jun. 2014.

[23] Y. Xiang, Y. Zhou, M. Li, and Z. Chen, "A vector angle-based evolutionary algorithm for unconstrained many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 21, no. 1, pp. 131–152, Feb. 2017.

[24] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Proc. Int. Conf. Parallel Probl. Solving Nat.*, 2004, pp. 832–842.

[25] N. Beume, B. Naujoks, and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *Eur. J. Oper. Res.*, vol. 181, no. 3, pp. 1653–1669, 2007.

[26] Q. Lin *et al.*, "Particle swarm optimization with a balanceable fitness estimation for many-objective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 22, no. 1, pp. 32–46, Feb. 2018.

[27] Y. Tian, R. Cheng, X. Zhang, F. Cheng, and Y. Jin, "An indicator-based multiobjective evolutionary algorithm with reference point adaptation for better versatility," *IEEE Trans. Evol. Comput.*, vol. 22, no. 4, pp. 609–622, Aug. 2018.

[28] Y. Liu, D. Gong, J. Sun, and Y. Jin, "A many-objective evolutionary algorithm using a one-by-one selection strategy," *IEEE Trans. Cybern.*, vol. 47, no. 9, pp. 2689–2702, Sep. 2017.

[29] T. Pamulapati, R. Mallipeddi, and P. N. Suganthan, "ISDE+—An indicator for multi and many-objective optimization," *IEEE Trans. Evol. Comput.*, to be published. doi: 10.1109/TEVC.2018.2848921.

[30] 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.

[31] A. Trivedi, D. Srinivasan, K. Sanyal, and A. Ghosh, "A survey of multiobjective evolutionary algorithms based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 21, no. 3, pp. 440–462, Jun. 2017.

[32] J. Molina, L. V. Santana, A. G. Hernández-Díaz, C. A. Coello Coello, and R. Caballero, "G-dominance: Reference point based dominance for multiobjective metaheuristics," *Eur. J. Oper. Res.*, vol. 197, no. 2, pp. 685–692, 2009.

[33] 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.

[34] S. Jiang, S. Yang, Y. Wang, and X. Liu, "Scalarizing functions in decomposition-based multiobjective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 22, no. 2, pp. 296–313, Apr. 2018.

[35] H. Li, Q. Zhang, and J. Deng, "Biased multiobjective optimization and decomposition algorithm," *IEEE Trans. Cybern.*, vol. 47, no. 1, pp. 52–66, Jan. 2017.

[36] S. Jiang and S. Yang, "An improved multiobjective optimization evolutionary algorithm based on decomposition for complex Pareto fronts," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp. 421–437, Feb. 2016.

[37] R. Wang, R. C. Purshouse, and P. J. Fleming, "Preference-inspired co-evolutionary algorithms using weight vectors," *Eur. J. Oper. Res.*, vol. 243, no. 2, pp. 423–441, 2015.

[38] M. Asafuddoula, T. Ray, and R. Sarker, "A decomposition-based evolutionary algorithm for many objective optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 3, pp. 445–460, Jun. 2015.

[39] 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.

[40] X. Cai, Z. Yang, Z. Fan, and Q. Zhang, "Decomposition-based-sorting and angle-based-selection for evolutionary multiobjective and many-objective optimization," *IEEE Trans. Cybern.*, vol. 47, no. 9, pp. 2824–2837, Sep. 2017.

[41] 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.

[42] H.-L. Liu, L. Chen, Q. Zhang, and K. Deb, "Adaptively allocating search effort in challenging many-objective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 22, no. 3, pp. 433–448, Jun. 2018.

[43] M. Li, S. Yang, and X. Liu, "Bi-goal evolution for many-objective optimization problems," *Artif. Intell.*, vol. 228, pp. 45–65, Nov. 2015.

[44] H. Chen, R. Cheng, J. Wen, H. Li, and J. Weng, "Solving large-scale many-objective optimization problems by covariance matrix adaptation evolution strategy with scalable small subpopulations," *Inf. Sci.*, to be published. doi: 10.1016/j.ins.2018.10.007.

[45] K. Deb and S. Gupta, "Understanding knee points in bicriteria problems and their implications as preferred solution principles," *Eng. Optim.*, vol. 43, no. 11, pp. 1175–1204, 2011.
[46] W. Du, S. Y. S. Leung, and C. K. Kwong, "Time series forecasting by neural networks: A knee point-based multiobjective evolutionary algorithm approach," *Expert Syst. Appl.*, vol. 41, no. 18, pp. 8049–8061, 2014.
[47] X. Zhang, Y. Tian, and Y. Jin, "A knee point-driven evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 6, pp. 761–776, Dec. 2015.
[48] R. Cheng *et al.*, "A benchmark test suite for evolutionary many-objective optimization," *Complex Intell. Syst.*, vol. 3, no. 1, pp. 67–81, 2017.
[49] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable multi-objective optimization test problems," in *Proc. Congr. Evol. Comput.*, 2002, pp. 825–830.
[50] 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.
[51] 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.
[52] Z. Wang, Y.-S. Ong, J. Sun, A. Gupta, and Q. Zhang, "A generator for multiobjective test problems with difficult-to-approximate Pareto front boundaries," *IEEE Trans. Evol. Comput.*, to be published. doi: 10.1109/TEVC.2018.2872453.
[53] H. Ishibuchi, R. Imada, Y. Setoguchi, and Y. Nojima, "Reference point specification in hypervolume calculation for fair comparison and efficient search," in *Proc. Genet. Evol. Comput. Conf.*, 2017, pp. 585–592.
[54] H. Chen, X. Zhu, G. Liu, and W. Pedrycz, "Uncertainty-aware online scheduling for real-time workflows in cloud service environment," *IEEE Trans. Services Comput.*, to be published. doi: 10.1109/TSC.2018.2866421.

Huangke Chen received the B.S. and M.S. degrees from the College of Information and System Management, National University of Defense Technology, Changsha, China, in 2012 and 2014, respectively, where he is currently pursuing the Ph.D. degree with the College of Systems Engineering.

He was a visiting Ph.D. student with the University of Alberta, Edmonton, AB, Canada, from 2017 to 2018. His current research interests include computational intelligence, multiobjective evolutionary algorithms, large-scale optimization, task, and workflow scheduling.

Ye Tian received the B.Sc., M.Sc., and Ph.D. degrees from Anhui University, Hefei, China, in 2012, 2015, and 2018, respectively.

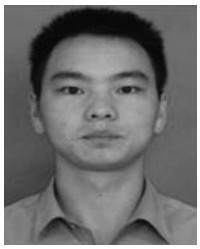
He is currently a Lecturer with the Institute of Physical Science and Information Technology, Anhui University. His current research interest includes multiobjective optimization methods and their application.

Dr. Tian was a recipient of the 2018 IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION Outstanding Paper Award.

Witold Pedrycz (F'99) received the M.Sc., Ph.D., and D.Sc. degrees from the Silesian University of Technology, Gliwice, Poland.

He is a Professor and the Canada Research Chair (CRC-Computational Intelligence) with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada, and also with the Department of Electrical and Computer Engineering, Faculty of Engineering, King Abdulaziz University, Jeddah, Saudi Arabia. He is also with the Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland. His current research interests include computational intelligence, fuzzy modeling and granular computing, knowledge discovery and data mining, fuzzy control, pattern recognition, knowledge-based neural networks, relational computing, and software engineering. He has published numerous papers in the above areas.

Dr. Pedrycz is the Editor-in-Chief of *Information Sciences* and serves as an Associate Editor for the IEEE TRANSACTIONS ON SYSTEM MAN CYBERNETICS: SYSTEM and the IEEE TRANSACTIONS ON FUZZY SYSTEMS. He is intensively involved in editorial activities. He is also a member of a number of editorial boards of other international journals.



Guohua Wu received the B.S. degree in information systems and the Ph.D. degree in operations research from the National University of Defense Technology, Changsha, China, in 2008 and 2014, respectively.

From 2012 to 2014, he was a visiting Ph.D. student with the University of Alberta, Edmonton, AB, Canada. From 2014 to 2017, he was a Lecturer with the College of Information Systems and Management, National University of Defense Technology. He is currently a Professor with the School of Traffic and Transportation Engineering, Central South University, Changsha. He has authored over 50 refereed papers, including those published in the IEEE TRANSACTIONS ON SYSTEM MAN CYBERNETICS: SYSTEM, *Information Sciences*, *Computers and Operations Research*, and *Applied Soft Computing*. His current research interests include planning and scheduling, evolutionary computation, and machine learning.

Dr. Wu served as an Associate Editor for *Swarm and Evolutionary Computation* and an Editorial Board Member of the *International Journal of Bio-Inspired Computation*. He is a regular Reviewer of over 20 journals, including the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, the IEEE TRANSACTIONS ON CYBERNETICS, *Information Sciences*, and *Applied Soft Computing*.

Rui Wang received the B.S. degree from the National University of Defense Technology, Changsha, China, in 2008 and the Ph.D. degree from the University of Sheffield, Sheffield, U.K., in 2013.

His current research interests include evolutionary computation, multiobjective optimization, machine learning, and optimization methods on energy Internet network.

Ling Wang received the B.Sc. degree in automation and the Ph.D. degree in control theory and control engineering from Tsinghua University, Beijing, China, in 1995 and 1999, respectively.

Since 1999, he has been with the Department of Automation, Tsinghua University, where he became a Full Professor in 2008. He has authored 5 academic books and over 260 refereed papers. His current research interests include intelligent optimization and production scheduling.

Prof. Wang was a recipient of the National Natural Science Fund for Distinguished Young Scholars of China, the National Natural Science Award (Second Place) in 2014, the Science and Technology Award of Beijing City in 2008, the Natural Science Award (First Place in 2003 and Second Place in 2007) nominated by the Ministry of Education of China. He is currently the Editor-in-Chief of the *International Journal of Automation and Control* and an Associate Editor of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION.