Accelerating Large-Scale Multiobjective Optimization via Problem Reformulation

Cheng He, Lianghao Li, Ye Tian, Xingyi Zhang, *Senior Member, IEEE*, Ran Cheng, *Member, IEEE*,
Yaochu Jin, *Fellow, IEEE*, and Xin Yao, *Fellow, IEEE*

***Abstract*—In this paper, we propose a framework to accelerate the computational efficiency of evolutionary algorithms on large-scale multiobjective optimization. The main idea is to track the Pareto optimal set (PS) directly via problem reformulation. To begin with, the algorithm obtains a set of reference directions in the decision space and associates them with a set of weight variables for locating the PS. Afterwards, the original large-scale multiobjective optimization problem is reformulated into a low-dimensional single-objective optimization problem. In the reformulated problem, the decision space is reconstructed by the weight variables and the objective space is reduced by an indicator function. Thanks to the low dimensionality of the weight variables and reduced objective space, a set of quasi-optimal solutions can be obtained efficiently. Finally, a multiobjective evolutionary algorithm is used to spread the quasi-optimal solutions over the approximate Pareto optimal front evenly. Experiments have been conducted on a variety of large-scale multiobjective problems with up to 5000 decision variables. Four different types of representative algorithms are embedded into the proposed framework and compared with their original versions, respectively. Furthermore, the proposed framework has been compared with two state-of-the-art algorithms for large-scale multiobjective optimization. The experimental results have demonstrated the significant improvement benefited from the framework in terms of its performance and computational efficiency in large-scale multiobjective optimization.**

***Index Terms*—Evolutionary algorithms, large-scale optimization, multiobjective optimization, problem reformulation.**

I. Introduction

MANY real-world optimization problems involve multiple conflicting objectives [1], [2], known as multiobjective optimization problems (MOPs), which can be mathematically formulated as follows:

$$\begin{aligned}&\text{minimize} \quad F(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x}))\\&\text{subject to} \quad \mathbf{x} \in X\end{aligned} \tag{1}$$

where X is the search space of decision variables with $\mathbf{x} = (x_1, \ldots, x_D)$ denoting the decision vector [3]. Due to the conflicting nature of the objectives, there does not typically exist a single solution that can minimize all the objectives simultaneously. Instead, a set of nondominated solutions can be obtained as the trade-offs between different objectives [4]. Suppose $\mathbf{x_A}$ and $\mathbf{x_B}$ are two solutions of an MOP illustrated by (1), solution $\mathbf{x_A}$ is known to Pareto dominate solution $\mathbf{x_B}$ (denoted as $\mathbf{x_A} \prec \mathbf{x_B}$), if and only if $f_i(\mathbf{x_A}) \leq f_i(\mathbf{x_B})$ $(\forall i \in \{1, 2, \ldots, M\})$ and there exists at least one objective f_j $(j \in \{1, 2, \ldots, M\})$ satisfying $f_j(\mathbf{x_A}) < f_j(\mathbf{x_B})$. The collection of all the Pareto optimal solutions in the decision space is called the Pareto optimal set (PS), and the projection of PS in the objective space is called the Pareto optimal front (PF).

To solve MOPs, a variety of multiobjective evolutionary algorithms (MOEAs) have been proposed during the past two decades [5], including the Pareto-based MOEAs [6]–[8], the decomposition-based MOEAs [9], [10], and the indicator-based MOEAs [11], [12], etc. Despite that most existing MOEAs have been well assessed on the MOPs with a small number of decision variables, their performance degenerates dramatically on MOPs with hundreds or even thousands of decision variables, i.e., the large-scale MOPs (LSMOPs) [13]. As the number of decision variables increases linearly, the volume (as well as complexity) of the search

Manuscript received June 14, 2018; revised September 21, 2018 and November 26, 2018; accepted January 21, 2019. Date of publication January 28, 2019; date of current version November 27, 2019. This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFC0804002, in part by the Program for Guangdong Introducing Innovative and Entrepreneurial Teams under Grant 2017ZT07X386, in part by the Shenzhen Peacock Plan under Grant KQTD2016112514355531, in part by the Science and Technology Innovation Committee Foundation of Shenzhen under Grant ZDSYS201703031748284, in part by the Program for University Key Laboratory of Guangdong Province under Grant 2017KSYS008, in part by the National Natural Science Foundation of China under Grant 61320106005 and Grant 61772214, and in part by Engineering and Physical Sciences Research Council under Grant EP/M017869/1. *(Corresponding author: Ran Cheng.)*

C. He, R. Cheng, and X. Yao are with the Shenzhen Key Laboratory of Computational Intelligence, University Key Laboratory of Evolving Intelligent Systems of Guangdong Province, Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, China (e-mail: chenghehust@gmail.com; chengr@sustc.edu.cn; xiny@sustc.edu.cn).

L. Li is with the Key Laboratory of Image Information Processing and Intelligent Control of Education Ministry of China, School of Automation, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: lianghaoli.hust@gmail.com).

Y. Tian is with the Institute of Physical Science and Information Technology, Anhui University, Hefei 230601, China (e-mail: field910921@gmail.com).

X. Zhang is with the Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, School of Computer Science and Technology, Anhui University, Hefei 230039, China (e-mail: xyzhanghust@gmail.com).

Y. Jin is with the Shenzhen Key Laboratory of Computational Intelligence, University Key Laboratory of Evolving Intelligent Systems of Guangdong Province, Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, China, and also with the Department of Computer Science, University of Surrey, Guildford GU2 7XH, U.K. (e-mail: yaochu.jin@surrey.ac.uk).

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

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/TEVC.2019.2896002

space will increase exponentially, and thus leading to the curse of dimensionality [14], [15]. In recent years, there has been an increasing interest in large-scale multiobjective optimization [16], [17]. Existing approaches for large-scale multiobjective optimization can be roughly classified into three different categories as follows.

The first category is known as the decision variable analysis-based approaches. A representative algorithm of this category is the MOEA based on decision variable analysis (MOEA/DVA) [18], where the original LSMOP is decomposed into a number of simpler sub-MOPs. Then, the decision variables in each sub-MOP is optimized as an independent subcomponent. Similarly, the decision variable clustering-based large-scale evolutionary algorithm (LMEA) [19] also divides the decision variables into two types using a clustering method. Then, the convergence-related and diversity-related variables are optimized using two different strategies by focusing on convergence and diversity, respectively.

The second category applies the cooperative coevolution (CC) framework [20]. For example, the third-generation cooperative coevolutionary differential evolution algorithm [21] maintains several independent subpopulations. Each subpopulation is a subset of the equal-length decision variables obtained by variable grouping (e.g., random grouping [22], linear grouping [23], ordered grouping [24], or differential grouping (DG) [25]). All the subpopulations work cooperatively to optimize the LSMOPs in a divide-and-conquer manner.

The third category is based on the problem transformation, where the original LSMOP is transformed into a simpler MOP with a relatively small number of decision variables. The weighted optimization framework (WOF) is representative in this category [26]. In WOF, the decision variables are divided into a number of groups, each of which is assigned with a weight variable. As a consequence, the optimization of the weight variables in the same group can be regarded as the optimization of a subproblem in a subspace of the original decision space.

There are also some other approaches that do not fall into the above three categories, e.g., the recently proposed competition mechanism-based multiobjective particle swarm algorithm (CMOPSO) [27]. Instead of adopting explicit decision variable analysis or grouping, the algorithm is motivated to implicitly enhance the swarm diversity of particle swarm optimizer (PSO) for solving LSMOPs using a pairwise competition strategy [28]. Despite that these existing approaches as introduced above can improve the performance of MOEAs on LSMOPs to some extent, the development of large-scale multiobjective optimization is still in its infancy. Particularly, most of the existing algorithms suffer from a low computational efficiency, in terms of both computation time and function evaluations. To accelerate the computational efficiency of existing MOEAs on large-scale multiobjective optimization, we propose a generic framework, termed large-scale multiobjective optimization framework (LSMOF). The main new contributions are summarized as follows.

1) A novel problem reformulation method is proposed. It reformulates the original LSMOP into a low-dimensional single-objective optimization problem (SOP) with some direction vectors and weight variables, aimed at guiding the population toward the PS. Different from existing dimension reduction techniques, there is no decision variable analysis or grouping process in our proposed method. Since the reformulated problem characterized by the weight variables has a lower dimensionality than the original problem, the computational efficiency can be significantly improved.
2) A bi-directional weight variable association strategy is proposed to enhance the performance of the proposed framework for tracking the PS in the decision space effectively. This strategy not only increases the population diversity of the reformulated problem to avoid local optima, but also eliminates the potential nonuniform search caused by the divergence of the unidirectional vectors.
3) A two-stage strategy is adopted in our proposed LSMOF. At the first stage, the decision space reconstruction-based single-objective optimization is used to push the population toward the PS efficiently. Then, the second stage spreads the candidate solutions over the approximate PS evenly.

The rest of this paper is organized as follows. In Section II, we briefly recall some related work on large-scale multiobjective optimization, and the motivation of this paper is also elaborated. The details of the proposed LSMOF for large-scale multiobjective optimization are presented in Section III. Experimental settings and comparisons of LSMOF with the state-of-the-art heuristic algorithms on the benchmark problems are presented in Section IV. Finally, the conclusion is drawn in Section V.

II. Related Work and Motivation

In this section, we first recall some concepts and definitions in large-scale multiobjective optimization. Then some related work about the decision variable analysis, decision variable grouping, and problem transformation are illustrated. Finally, the motivation of this paper is elaborated.

A. Decision Variable Analysis

Definition 1: $f(\mathbf{x})$ is called a partially separable with k components *iff* [25], [29]

$$\arg\min_{\mathbf{x}} f(\mathbf{x}) = \left(\arg\min_{\mathbf{x_1}} f(\mathbf{x_1}, \ldots), \ldots, \arg\min_{\mathbf{x_k}} f(\ldots, \mathbf{x_k})\right)$$

where $\mathbf{x} = (x_1, \ldots, x_D)$ is a decision vector and $\mathbf{x_1}, \ldots, \mathbf{x_k}$ ($k \in [2, D]$) are disjoint subvectors of $\mathbf{x}$.

Definition 2: Two decision variables x_i and x_j are interacting if there exit $\mathbf{x}$, a_1, a_2, b_1, and b_2 satisfying

$$\begin{aligned} &f(\mathbf{x})|_{x_i=a_2,x_j=b_1} < f(\mathbf{x})|_{x_i=a_1,x_j=b_1} \wedge \\ &f(\mathbf{x})|_{x_i=a_2,x_j=b_2} > f(\mathbf{x})|_{x_i=a_1,x_j=b_2} \end{aligned} \tag{2}$$

where

$$f(\mathbf{x})|_{x_i=a_2,x_j=b_1} \triangleq f\big(x_1, \ldots, x_{i-1}, a_2, \ldots, x_{j-1}, b_1, \ldots, x_D\big).$$

According to Definition 1, an SOP is known as partially separable if the decision variables can be divided into a number of subcomponents and optimized independently. The main idea of decision variable analysis is intuitive. First, the interdependence between the pairwise decision variables is detected based on Definition 2 by different techniques, e.g., perturbation [30], interaction adaption [31], [32], modeling [33], or randomization [34]. Then, the relationship between a specific decision variable and the optimization problem is analyzed. To be specific, a decision variable can be related to convergence, diversity, or both of them. Finally, the decision variables of different types can be optimized using different strategies.

In MOEA/DVA [18], the decision variables are divided into three types according to their control properties of convergence or/and spread. The three types of decision variables are defined as follows.

1) *Position Variable:* Decision variable x_j in $\mathbf{x}$ is called a position variable *iff* changing x_j in $\mathbf{x}$ will generate a new solution $\mathbf{x}'$ satisfying that $\mathbf{x}' \nprec \mathbf{x}$ and $\mathbf{x} \nprec \mathbf{x}'$.
2) *Distance Variable:* Decision variable x_j in $\mathbf{x}$ is called a distance variable *iff* changing x_j in $\mathbf{x}$ will generate a new solution $\mathbf{x}'$ satisfying that $\mathbf{x}' \prec \mathbf{x}$ or $\mathbf{x} \prec \mathbf{x}'$.
3) *Mixed Variable:* Decision variables do not fall into any of above two types.

Similarly, the decision variables in LMEA are also clustered into convergence-related variables and diversity-related variables [19]. By dividing the decision variables into different types, the algorithms are able to adopt different optimization strategies to focus on convergence and diversity, respectively. Nevertheless, a crucial disadvantage of the decision variable analysis is the high computational cost, especially when there is a large number of decision variables. For example, it takes up to 7 577 615 function evaluations for LMEA to perform decision variable analysis on an LSMOP with 1000 decision variables (i.e., the 1000D DTLZ1 problem), which is prohibitively expensive in practice.

B. Grouping Techniques in CC

In the CC-based algorithms, the decision variables are divided into a number of groups and optimized in a cooperative coevolutionary manner. Assuming that the grouping technique aims to divide D decision variables into k groups, some representative grouping strategies are summarized as follows.

1) *Random Grouping [22]:* The decision variables are randomly divided into k even groups.
2) *Linear Grouping [23]:* The decision variables are assigned to k groups in order, i.e., $x_1, \ldots, x_{D/k}$ are assigned to the first group, $x_{D/k+1}, \ldots, x_{2D/k}$ are assigned to the second group, and so forth.
3) *Ordered Grouping [24]:* For a selected solution, the decision variables are sorted by their absolute values in ascending order. The D/k decision variables with the smallest absolute decision variables are assigned to the first group, and the rest may be deduced by analogy.
4) *DG [35]:* In contrast to the above three grouping techniques which are based on some heuristic strategies, DG techniques take the variable interactions into consideration when performing grouping [25], where the interacting decision variables are divided into the same group.

Without prior knowledge about the interactions among the decision variables or the number of groups, the performance of CC-based algorithms can be influenced by the selection of different grouping techniques.

C. Problem Transformation

Inspired by the grouping mechanism in the CC framework, the problem transformation strategy is proposed to improve the efficiency of CC-based algorithms on large-scale multiobjective optimization [26]. Instead of optimizing different subpopulations with fixed decision variables, the problem transformation strategy assigns a weight variable to the original decision variables in each group. Then the optimization of the decision variables is transformed to the optimization of the weight variables, which has significantly improved the efficiency of the algorithm.

Given a candidate solution $\mathbf{x} = \mathbf{x_1}, \ldots, \mathbf{x_k}$ (refer to Definition 1), the original optimization problem $f(\mathbf{x})$ can be reformulated into a new optimization problem $f(\psi(\omega, \mathbf{x}))$ by a linear function $\psi(\omega, \mathbf{x})$

$$\psi(\omega, \mathbf{x}) = \left(\underbrace{w_1 x_1, \ldots, w_1 x_{D/k}}_{w_1}, \ldots, \underbrace{w_k x_{D-k+1}, \ldots, w_1 x_D}_{w_k} \right) \tag{3}$$

where w_i $(i \in [1, k])$ is a weight variable and k is the number of groups. In this way, the optimization of the D decision variables is transformed to the optimization of a problem with k decision variables [26].

Despite that the transformation strategy is able to reduce the dimensionality of the decision space to a certain extent, it suffers from two main drawbacks. First, since the transformed subproblems are optimized separately, the correlations between different weight variables are not considered. Second, since the performance of the transformation strategy heavily depends on the grouping technique adopted therein, its computational efficiency and stability remain to be improved.

D. Motivation

While most existing approaches in the literature mainly focused on the optimization performance, little work has been dedicated to improving the computational efficiency. As a result, the computational budgets for solving LSMOPs could be expensive in terms of computation time as well as the number of FEs.

For example, an experimental comparison is conducted on the bi-objective LSMOP8 problem with 200 decision variables using LMEA, MOEA/DVA, and NSGA-II, where the decision variables of the test problem are mixed/interacting. The plot of the convergence profiles of the mean IGD values achieved by NSGA-II, MOEA/DVA, and LMEA on the problem is displayed in Fig. 1. It can be observed that NSGA-II performs

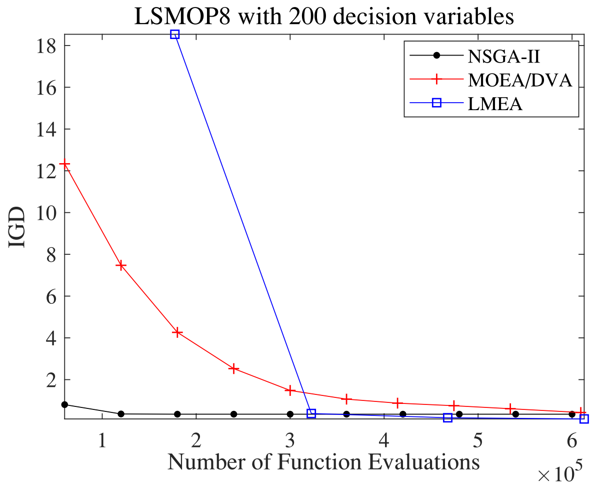


Fig. 1. Convergence profiles of the mean IGD values achieved by NSGA-II, MOEA/DVA, and LMEA on bi-objective LSMOP8 with 200 decision variables.

much better in the early stage of the evolution. Meanwhile, NSGA-II performs overall better than MOEA/DVA and similarly to LMEA, which implies that the decision variable analysis adopted by MOEA/DVA and LMEA do not work effectively. However, in order to perform the variable analysis, it will cost much more additional FEs and computation time [19]. Moreover, since the grouping-based approaches are highly dependent on the grouping results, an unsuitable grouping may lead to complete failure of an algorithm [20].

To address the above issues, this paper proposes a problem reformulation-based framework, termed LSMOF, for large-scale multiobjective optimization. Without using any grouping technique or decision variable analysis method, our LSMOF shows competitive optimization performance and computational efficiency compared to the existing approaches in the literature.

III. Proposed Framework

The main scheme of the proposed LSMOF is presented in Algorithm 1. To begin with, the population of the embedded MOEA is initialized. Then a two-stage strategy is adopted, where the first stage aims to find several quasi-optimal solutions near the PS and the second stage spreads them over the approximate PS evenly. At the first stage, the decision space is reconstructed with the assistance of population P, and the original LSMOP is reformulated into a low-dimensional SOP Z'; then, a single-objective optimizer (e.g., the differential evolution (DE) algorithm [36]) is used to optimize Z'. The above problem reformulation and single-objective optimization repeat until the maximum number of FEs is reached. For simplicity, we allocate 50% of the whole FEs to each stage, i.e., tr is set to 0.5. At the second stage, the original LSMOP is optimized by the embedded MOEA with the population P obtained at the first stage. Note that the LSMOF framework shares the same environmental selection operator with the embedded MOEA, and thus we will not enter the details of it. In the following sections, we will introduce the other two main components in Algorithm 1, i.e., problem reformulation and single-objective optimization.

Algorithm 1 Main Framework of the Proposed LSMOF

```
Input: $Z$ (original LSMOP), $FE_{max}$ (total FEs), $Alg$ (embedded MOEA), $N$ (population size for $Alg$), $r$ (number of reference solutions), $tr$ (threshold).
Output: $P$ (final population).
1:  $P \leftarrow$ Initialization$(N, Z)$
2:  /***********First Stage***********/
3:  while $t \leq tr \times FE_{max}$ do
4:      $Z' \leftarrow$ Problem_Reformulation$(P, r, Z)$
5:      $A, \Delta t \leftarrow$ Single_Objective_Optimization$(Z')$
6:      $P \leftarrow$ Environmental_Selection$(A \bigcup P, N)$
7:      $t \leftarrow t + \Delta t$
8:  end while
9:  /*********Second Stage*********/
10: $P \leftarrow$ Embedded_MOEA $(P, N, Alg, Z)$
```

A. Problem Reformulation

Problem reformulation is a crucial component of the proposed LSMOF (step 4 in Algorithm 1), which reformulates the original LSMOP into an SOP with relatively small-scale weight variables. To be specific, the proposed problem reformulation consists of three steps: 1) the bi-direction weight variable association; 2) the weight variable-based subproblem construction; and 3) the objective space reduction, where the first two steps aim to reconstruct the decision space.

1) Weight Variable Association: To guide the search of the algorithm toward the PS, a set of well converged and evenly distributed candidate solutions is used during the decision space reconstruction. For simplicity, we directly use the environmental selection in the embedded MOEA to select r solutions from the current population P as the reference solution set. Afterwards, each reference solution is associated with two direction vectors and two weight variables. This operation aims to specify the search directions in the decision space and guide the population toward the PS.

Fig. 2 illustrates the relationship between the reference solutions, the direction vectors, and the weight variables. In this example, a reference solution $\mathbf{s_1} = (x_1, \ldots, x_d)$ is located in a 2-D decision space; $\mathbf{o}$ and $\mathbf{t}$ are the lower and upper boundary points of X; $\mathbf{v}_l$ and $\mathbf{v}_u$ are vectors starting from $\mathbf{o}$ and $\mathbf{t}$ and pointing to $\mathbf{s}_1$, respectively

$$\begin{aligned}\mathbf{v}_l &= \mathbf{s}_1 - \mathbf{o}\\ \mathbf{v}_u &= \mathbf{t} - \mathbf{s}_1\end{aligned} \tag{4}$$

where $l_{\max} = ||\mathbf{t} - \mathbf{o}||$ is the maximum diagram length in X. Assume that points $\mathbf{p}_1$ and $\mathbf{p}_2$ are intersections between vectors $\mathbf{v}_l$ and $\mathbf{v}_u$ and the PS, and the distances from $\mathbf{o}$ to $\mathbf{p}_1$ and $\mathbf{t}$ to $\mathbf{p}_2$ are $\lambda_{11}(\mathbf{v}_l/||\mathbf{v}_l||)l_{\max}$ and $\lambda_{12}(\mathbf{v}_u/||\mathbf{v}_u||)l_{\max}$, respectively, the values of $\mathbf{p}_1$ and $\mathbf{p}_2$ can be calculated as

$$\begin{aligned}\mathbf{p}_1 &= \mathbf{o} + \lambda_{11}\frac{\mathbf{v}_l}{||\mathbf{v}_l||}l_{\max}\\ \mathbf{p}_2 &= \mathbf{t} - \lambda_{12}\frac{\mathbf{v}_u}{||\mathbf{v}_u||}l_{\max}\end{aligned} \tag{5}$$

where λ_{11} and λ_{11} are two weight variables. Note that since the weight variables are between 0 and 0.5, each weight variable

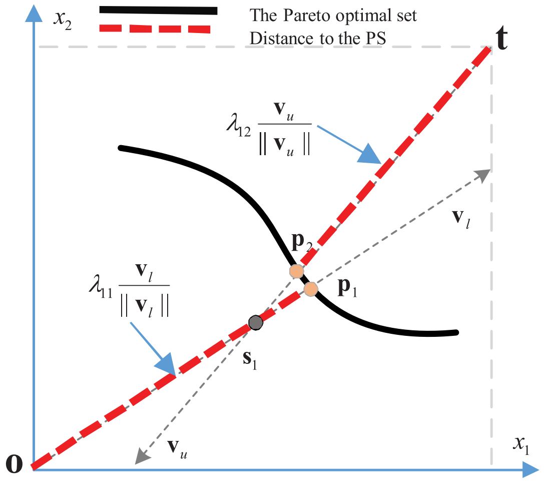


Fig. 2. Example of the bi-directional weight variable association. In this example, $\mathbf{s}_1$ is a selected solution, **o** and **t** are the lower and upper boundary points, and $\mathbf{p}_1$ and $\mathbf{p}_2$ are intersections between the direction vectors and the PS. Besides, two weight variables λ_{11} and λ_{12} and two direction vectors $\mathbf{v_l}$ and $\mathbf{v_u}$ are associated with this solution.

will only range in half of the original search space. Hence, the bi-directional weight variables will cover the entire search space without overlapping, which also enables the parallelization of the search for enhancing the efficiency of the proposed algorithm.

To be specific, each reference solution is associated with bi-directional vectors instead of a unidirectional vector, which is out of two main considerations. First, the bi-directional vectors can enhance the population diversity, thus reducing the possibility that there is no intersection between a given directional vector and the PS. For instance, if the reference solution locates around the boundary of the PS or the PS locates around a corner of the decision space, the unidirectional vector may disjoint with the PS, but the bi-directional vectors are more likely to have at least one intersection with the PS. Second, the bi-directional vectors can eliminate the nonuniform search caused by the divergence of the unidirectional vectors. To further illustrate the advantage of the proposed bi-directional weight variable association strategy, an example is given in Fig. 3. Generally, it has a better chance to locate the Pareto optimal solutions on the PS by adopting the bi-directional vectors than the unidirectional vectors. Furthermore, the experimental comparisons between our proposed algorithm with bi-directional vectors (LSMOF) and that with unidirectional vector (LSMOFU) are given in the supplementary materials. The results indicate that LSMOF has an overall superiority over LSMOFU, thus verifying the effectiveness of adopting bi-directional vectors.

2) Subproblem Construction: Given a reference solution set of size r, once each reference solution is associated with two direction vectors and two weight variables, a total number of $2r$ subproblems can be constructed. Taking the first reference solution $\mathbf{s_1}$ for example, two subproblems can be constructed as follows:

$$z_{11}(\lambda_{11}) = F\left(\mathbf{o} + \lambda_{11}\frac{\mathbf{v}_l}{||\mathbf{v}_l||}l_{\max}\right)$$

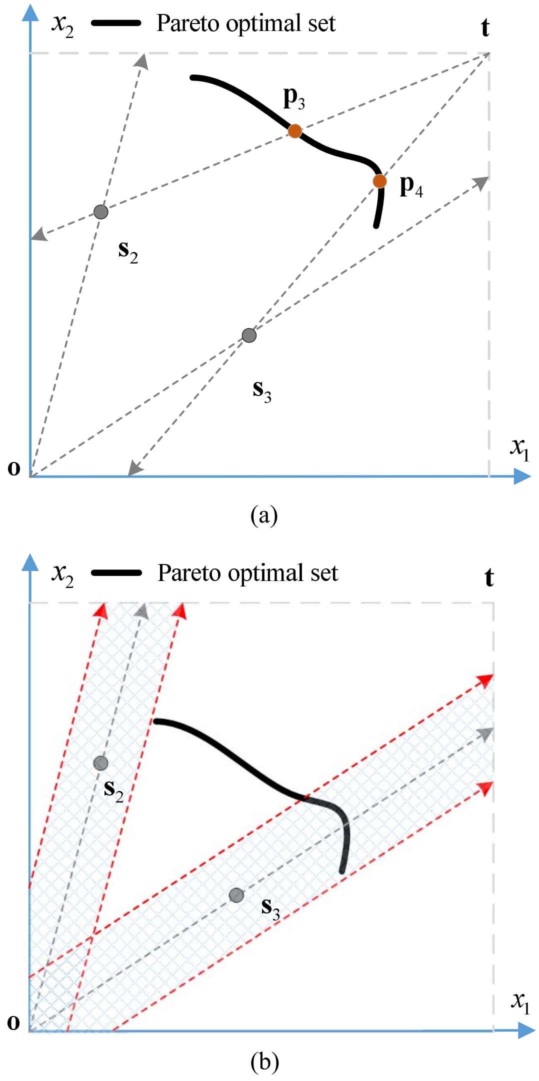


Fig. 3. Example illustrating the advantage of the proposed bi-directional weight variable association strategy in a 2-D decision space. $\mathbf{s}_2$ and $\mathbf{s}_3$ are the reference solutions, $\mathbf{p}_3$ and $\mathbf{p}_4$ are the intersections, and **o**, **t** are the lower and upper boundary points, respectively. Illustration of the (a) proposed bi-directional weight variable association strategy in a 2-D decision space and (b) unidirectional weight variable association strategy in a 2-D decision space.

$$z_{12}(\lambda_{12}) = F\left(\mathbf{t} - \lambda_{12}\frac{\mathbf{v}_u}{||\mathbf{v}_u||}l_{\max}\right) \tag{6}$$

where λ_{11} and λ_{12} are two 1-D weight variables, $\mathbf{v}_l$ and $\mathbf{v}_u$ are the reference directions calculated by (4), and **o**, **t**, and $l_{\max}$ are elements in (5). The constructed subproblems are $Z'(\Lambda) = \{z_{11}(\lambda_{11}), z_{12}(\lambda_{12}), \ldots, z_{r1}(\lambda_{r1}), z_{r2}(\lambda_{r2})\}$, where the weight vector $\Lambda = \{\lambda_{11}, \lambda_{12}, \ldots, \lambda_{r1}, \lambda_{r2}\}$ is in the reconstructed decision space.

It is worth noting that there are two main differences between our proposed subproblem construction and the problem transformation in WOF. First, while the decision variables in WOF are divided using grouping techniques, the decision variables in the proposed LSMOF are controlled by the weight variables and optimized as a whole. On one hand, it can explicitly save the computational cost of variable analysis; on the other hand, it can implicitly take the variable interactions into consideration during the evolution. Second, while

Algorithm 2 Fitness Assignment Strategy in LSMOF

```
Input: $\Lambda$ (weight vector), $R$ (reference solution set), $P$ (current population).
Output: $fit(\Lambda)$ (fitness value of $\Lambda$).
1: $Nad \leftarrow$ Calculate the nadir point of $P$ in objective space
2: for $i \leftarrow 1 : r$ do
3:     $\mathbf{v}_l, \mathbf{v}_u \leftarrow$ Calculate direction vectors using (4)
4:     $\mathbf{p}_1, \mathbf{p}_2 \leftarrow$ Calculate the weight variable associated solutions using (5)
5:     $z_{i1}(\lambda_{i1}), z_{i2}(\lambda_{i2}) \leftarrow$ Calculate the objective vectors using (6)
6: end for
7: $Z'(\Lambda) \leftarrow \{z_{11}(\lambda_{11}), z_{12}(\lambda_{12}), \ldots, z_{r1}(\lambda_{r1}), z_{r2}(\lambda_{r2})\}$
8: $fit(\Lambda) \leftarrow$ Hypervolume$(Z'(\Lambda), Nad)$
```

the direction vectors in WOF are unidirectional, those in the proposed LSMOF are bi-directional. In general, the coverage of the bi-directional search in LSMOF is larger than that of WOF, which enhances the exploration ability of the algorithm and maintains better population diversity.

3) Objective Space Reduction: Once the subproblems are reconstructed, the optimization of the decision vector $\mathbf{x}$ in the original decision space is transformed to the optimization of the weight vector Λ in the reconstructed decision space. Correspondingly, the objective space can be reduced and the new optimization problem can be reformulated as

$$\begin{aligned} &\text{maximize} \;\; G(\Lambda) = H\big(Z'(\Lambda)\big) \\ &\text{subject to} \;\; \Lambda \in \Re^{2r} \end{aligned} \tag{7}$$

where H is a function to assess the quality of the $2r$ multiobjective solutions, and H can be any performance indicator, e.g., the hypervolume (HV) indicator [37] as adopted in this paper. By using such a reformulation scheme, the scale of the original problem can be substantially reduced. For example, an LSMOP with 1000 decision variables can be reformulated into an SOP with only ten decision variables as in our experiments.

To assess the quality of the reconstructed decision vectors, we propose a fitness assignment strategy for the evaluation of (7) in a reduced objective space. As given by the pseudo code of the fitness assignment procedure in Algorithm 2, the fitness of a weight vector (i.e., a sequence formed by the weight variables) in the reconstructed decision space can be calculated by two main stages. At the first stage, the objective vectors in accordance with the weight vector Λ are calculated (steps 1–6). Then, at the second stage, the fitness value of the objective vector is calculated using the HV indicator. In this way, the algorithm is able to return a scalar value as the fitness of the reconstructed decision vector Λ.

B. Single-Objective Optimization

Once the original LSMOP is reformulated, the proposed LSMOF is expected to perform single-objective optimization of the weight variables in the reconstructed decision space and the reduced objective space. For simplicity, we adopt the widely used DE [38] as the single-objective optimizers in this paper. Note that we use DE in this framework due to its efficiency and simplicity, and any other single-objective optimization algorithm (e.g., the PSO [39]) is also compatible with the proposed framework.

Algorithm 3 Single-Objective Optimization in LSMOF

```
Input: $NI$ (population size of DE), $g$ (maximum number of iterations), $CR$ (cross constant), $F_m$ (scaling factor).
Output: $A$ (population), $\Delta t$ (number of FEs).
$\Delta t \leftarrow 0$
$P_\Lambda \leftarrow \{\Lambda_1, \ldots, \Lambda_{NI}\}$ /*Initialization*/
$fit(\Lambda_1), \ldots, fit(\Lambda_{NI}) \leftarrow$ Calculate the fitnesses of elements in $P_\Lambda$ using Algorithm 2
$A \leftarrow$ Collet the generated candidate solutions during the fitness assignment
$\Delta t \leftarrow \Delta t + |A|$ /*$|A|$ denotes the element size of $A$*/
for $r \leftarrow 1 : g$ do
    for $i \leftarrow 1 : NI$ do
        $c_1, c_2, c_3 \leftarrow$ Randomly select three indices in $[1, NI]$
        $\mathbf{a} \leftarrow \Lambda_{c_1} + F_m(\Lambda_{c_2} - \Lambda_{c_3})$
        for $j \leftarrow 1 : |\Lambda_1|$ do
            if $rand_j[0, 1) \leq CR$ or $j = j_{rand}$ then
                $\mathbf{b}_i \leftarrow$ Choose the $j$th element of $\Lambda_i$
            else
                $\mathbf{b}_i \leftarrow$ Choose the $j$th element of $\mathbf{a}$
            end if
        end for
        $fit(\mathbf{b}) \leftarrow$ Calculate $\mathbf{b}$'s fitness using Algorithm 2
        $A' \leftarrow$ Collet the generated candidate solutions during the fitness assignment
        $\Delta t \leftarrow \Delta t + |A'|$
        if $fit(\mathbf{b}) \geq fit(\Lambda_i)$ then
            $\Lambda_i \leftarrow \mathbf{b}$
        end if
        $A \leftarrow A \cup A'$
    end for
end for
```

The details of the DE-based single-objective optimization are presented in Algorithm 3. In this algorithm, a set of weight vectors P_Λ are first initialized in range $[0, 0.5]$ as defined in (5), and their fitness values are calculated using Algorithm 2. For each weight vector Λ_i in P_Λ, three different weight vectors are randomly selected from P_Λ to form a trail vector $\mathbf{a}$ for crossover, where $\mathbf{a}$ is associated with the weight vector to generate an offspring $\mathbf{b}$ according to a probability rate CR. If the fitness of offspring $\mathbf{a}$ is better than that of Λ_i, the weight vector Λ_i is replaced by $\mathbf{b}$. The reproduction and replacement procedures repeat for a number of g iterations. All the candidate solutions generated during the evolution of the weight vectors are merged into the archive A, which will be used as the initial population of the embedded MOEA at step 10 of Algorithm 1.

It is worth noting that during the optimization of the weight variables, a number of candidate solutions of the original MOP are generated by Algorithm 2 (e.g., $\mathbf{p_1}$ and $\mathbf{p_2}$). In other words, the optimization of the weight variables naturally leads to the optimization of the original problem.

IV. Empirical Studies

To empirically investigate the performance of the proposed LSMOF framework, four representative MOEAs, namely NSGA-II [6], MOEA/D-DE [10], SMS-EMOA [12], and CMOPSO [27], are embedded into LSMOF and compared with their original versions on nine test problems from the LSMOP test suite [13]. Here we adopt these four algorithms as they represent different types of MOEAs as discussed in Section I, and the embedding of these algorithms could reveal the potential advantages of our proposed LSMOF. Then, two state-of-the-art large-scale MOEAs, namely WOF [26][1] and MOEA/DVA [18], are also compared with our proposed LSMOF.[2]

In the remainder of this section, we first present a brief introduction to the adopted performance indicator, and then we give the parameter settings of the compared algorithms and our proposed LSMOF. Afterwards, each algorithm is run for 20 times on each test problem independently, and the Wilcoxon rank sum test [42] is used to compare the results obtained by the proposed LSMOF and the compared algorithms at a significance level of 0.05. Symbols "+," "−," and "≈" indicate the compared algorithm is significantly better than, significantly worse than, and statistically tied by LSMOF.

A. Performance Indicator

In the experiments, a widely used performance indicator, the inverted generational distance (IGD) [43], is adopted for evaluating the performance of the compared algorithms.

Suppose that P^* is a set of evenly distributed reference points on the PF and Ω is the set of obtained nondominated solutions, IGD is defined as follows:

$$\mathrm{IGD}\big(P^*, \Omega\big) = \frac{\sum_{\mathbf{x}\in P^*} \mathrm{dis}(\mathbf{x}, \Omega)}{|P^*|} \tag{8}$$

where $\mathrm{dis}(\mathbf{x}, \Omega)$ is the minimum Euclidean distance between $\mathbf{x}$ and points in Ω and $|P^*|$ the number of elements in P^*. A smaller value of IGD will indicate a better performance of the algorithm. In this paper, the size of P^* is set to 10 000 (or a close number) for the IGD calculations.

Note that the HV [37] values of the obtained nondominated solutions are presented in the supplementary materials. In this paper, we use the IGD indicator instead of the HV indicator since the PFs of LSMOP test problems are relatively simple and regular. Meanwhile, the reference solution sets in PlatEMO [44], [45] are evenly sampled, which enables the IGD indicator to well assess the qualities of the obtained solution sets.

B. Experimental Settings

For fair comparisons, we adopt the recommended parameter settings for the compared algorithms that have achieved the best performance as reported in the literature. All the compared algorithms are implemented in PlatEMO [44].

1) Reproduction Operators: In this paper, the simulated binary crossover [4] and the polynomial mutation (PM) [46] are adopted in the compared algorithms for offspring generation in NSGA-II and SMS-EMOA. The distribution index of crossover is set to n_c =20 and the distribution index of mutation is set to n_m =20, as recommended in [4]. The crossover probability p_c is set to 1.0 and the mutation probability p_m is set to $1/D$, where D is the number of decision variables. In MOEA/D-DE and MOEA/DVA, DE operator [36] and PM are used for offspring generation, where the control parameters are set to CR =1, F =0.5, $p_m = 1/d$, and η=20 as recommended in [10]. As for CMOPSO, the particle swarm operator [47] and PM are used, where parameters R_1 and R_2 are randomly selected from [0, 1] with γ set to 10 as recommended in [27].

2) Population Size: The population size is set to 100 for test instances with two objectives and 105 for test instances with three objectives.

3) Specific Parameter Settings in Each Algorithm: In MOEA/D-DE, the neighborhood size T is set to 20, the probability of choosing parents locally δ is set to 0.9, and the maximum number of solutions replaced by each offspring n_r is set to 2. In WOF, the number of FEs for the optimization of each original problem t_1 is set to 500, and for the transferred problem, t_2 is set to 250, parameter q is set to 3, the number of groups γ is set to 4, and the ordered grouping is adopted as the grouping method [26]. Meanwhile, NSGA-II is embedded in both WOF and LSMOF to be compared on LSMOP problems.[3] In MOEA/DVA, the number of sampling solutions to recognize the control properties of the decision variables is set to 20, and the maximum number of trails required to judge the interaction between two variables is set to 6. In the proposed LSMOF, the number of reference solutions r is set to 10, the population size for the single-objective optimization is set to 30, and the mutation factor F_m in DE is set to 0.8.

4) Termination Condition: A total number of 50 000 FEs is adopted as the termination condition for all the test instances. The number of FEs is relatively small for existing MOEAs, but it is practical for real-world applications. It is attributed to the fact that the number of FEs is always limited by the economic and/or computational cost, especially for the large-scale optimization problems.

C. General Performance

To investigate the effect of LSMOF on different MOEAs, four representative algorithms, i.e., NSGA-II, MOEA/D-DE, SMS-EMOA, and CMOPSO, are embedded into the proposed framework. Pairwise comparisons are conducted between the heuristic algorithm and its LSMOF version in terms of both solution quality and algorithm runtime. The experimental results obtained by these compared algorithms are displayed in Table I, where LS-*Alg* denotes the LSMOF with algorithm *Alg* embedded.

[1] The implementation of WOF is adapted from the codes available at http://www.is.ovgu.de/Team/Heiner+Zille.html.

[2] More experimental results on test problems selected from DTLZ [40] and WFG [41] are displayed in the supplementary materials.

[3] Experiments on DTLZ and LSMOP problems using WOF-SMPSO [26] are conducted in Section II of the supplementary materials.

TABLE I
STATICS OF IGD RESULTS OBTAINED BY EIGHT COMPARED ALGORITHMS ON 54 TEST INSTANCES FROM LSMOP TEST SUITE. THE BEST RESULTS IN EACH TWO COLUMNS ARE HIGHLIGHTED

Problem	M	D	NSGA-II	LS-NSGA-II	MOEA/D-DE	LS-MOEA/D-DE	SMS-EMOA	LS-SMS-EMOA	CMOPSO	LS-CMOPSO
LSMOP1	2	200	9.17E-1(2.54E-1)−	5.78E-1(5.32E-2)	3.59E-1(1.85E-2)−	2.13E-1(3.44E-2)	6.24E-1(6.25E-2)−	5.30E-1(6.69E-2)	4.38E-1(2.32E-1)+	5.75E-1(4.58E-2)
		500	2.73E+0(2.77E-1)−	6.14E-1(2.54E-2)	1.07E+0(9.70E-2)−	3.10E-1(3.44E-2)	2.09E+0(5.51E-1)−	5.98E-1(3.35E-2)	1.50E+0(1.61E-1)−	6.18E-1(2.35E-2)
		1000	4.21E+0(2.70E-1)−	6.37E-1(1.97E-2)	1.64E+0(1.17E-1)−	4.26E-1(5.07E-2)	3.72E+0(2.83E-1)−	6.22E-1(2.66E-2)	2.50E+0(1.31E-1)−	6.37E-1(1.99E-2)
	3	200	2.08E+0(2.12E-1)−	5.24E-1(1.35E-2)	1.57E+0(1.56E-1)−	5.26E-1(3.84E-2)	4.58E-1(3.02E-2)+	5.04E-1(1.13E-2)	2.12E+0(3.42E-1)−	5.20E-1(2.66E-2)
		500	5.05E+0(5.73E-1)−	5.96E-1(1.08E-2)	1.80E+0(1.59E-1)−	6.54E-1(4.31E-2)	2.94E+0(3.50E-1)−	5.84E-1(4.14E-2)	4.23E+0(5.58E-1)−	6.16E-1(1.55E-2)
		1000	6.93E+0(6.64E-1)−	6.33E-1(1.34E-2)	1.86E+0(1.97E-1)−	6.68E-1(6.41E-2)	6.29E+0(3.95E-1)−	7.09E-1(1.12E-1)	6.80E+0(5.29E-1)−	6.94E-1(2.13E-2)
LSMOP2	2	200	1.02E-1(2.93E-3)−	3.85E-2(1.08E-3)	9.64E-2(2.38E-3)−	2.71E-2(1.54E-3)	9.19E-2(2.90E-3)−	3.55E-2(2.01E-3)	9.82E-2(2.50E-3)−	3.70E-2(1.14E-3)
		500	6.20E-2(1.16E-3)−	2.32E-2(6.90E-4)	4.89E-2(1.68E-3)−	1.38E-2(1.17E-3)	5.41E-2(1.21E-3)−	1.65E-2(4.67E-4)	5.54E-2(1.42E-3)−	2.14E-2(6.83E-4)
		1000	3.70E-2(3.16E-4)−	1.81E-2(5.41E-4)	2.75E-2(9.26E-4)−	9.15E-3(1.27E-3)	3.30E-2(3.87E-4)−	9.73E-3(2.00E-4)	3.72E-2(7.32E-4)−	1.54E-2(8.72E-4)
	3	200	1.27E-1(4.79E-3)+	1.38E-1(2.76E-3)	1.05E-1(2.83E-3)−	8.51E-2(2.95E-3)	1.23E-1(1.97E-3)+	1.25E-1(5.01E-3)	1.21E-1(9.02E-4)−	1.17E-1(2.27E-3)
		500	8.25E-2(5.49E-3)+	8.71E-2(3.29E-3)	7.41E-2(8.49E-4)−	6.55E-2(9.76E-4)	7.98E-2(2.11E-3)+	8.14E-2(2.98E-3)	6.83E-2(2.81E-4)+	7.20E-2(9.77E-3)
		1000	6.72E-2(3.63E-3)+	7.05E-2(3.08E-3)	6.35E-2(2.54E-4)−	5.97E-2(4.12E-4)	6.55E-2(2.63E-3)+	6.64E-2(1.65E-3)	5.18E-2(3.66E-4)+	5.22E-2(5.09E-4)
LSMOP3	2	200	1.42E+1(2.56E+0)−	1.54E+0(1.43E-3)	5.82E+0(1.03E+0)−	1.53E+0(5.83E-3)	1.73E+1(2.63E+0)−	1.54E+0(1.12E-3)	3.85E+0(6.91E-1)−	1.52E+0(3.30E-3)
		500	1.92E+1(1.62E+0)−	1.57E+0(1.05E-3)	1.33E+1(1.29E+0)−	1.56E+0(1.41E-3)	2.21E+1(1.26E+0)−	1.57E+0(9.70E-4)	2.86E+1(1.24E+0)−	1.56E+0(2.01E-3)
		1000	2.22E+1(1.12E+0)−	1.57E+0(2.28E-4)	1.83E+1(1.22E+0)−	1.57E+0(3.30E-4)	2.35E+1(1.04E+0)−	1.57E+0(2.31E-4)	3.06E+1(1.06E+0)−	1.57E+0(8.81E-4)
	3	200	7.30E+0(1.37E+0)−	8.40E-1(2.51E-2)	7.77E+0(9.45E-1)−	8.27E-1(4.68E-2)	2.65E+0(7.63E-1)−	8.24E-1(3.15E-2)	9.46E+0(8.41E-1)−	8.60E-1(2.45E-3)
		500	1.53E+1(2.62E+0)−	8.59E-1(3.26E-3)	1.00E+1(7.92E-1)−	8.19E-1(4.79E-2)	7.81E+0(1.30E+0)−	1.60E+0(3.09E+0)	1.31E+1(8.51E-1)−	8.61E-1(1.14E-6)
		1000	1.95E+1(3.27E+0)−	8.61E-1(7.03E-5)	1.08E+1(5.73E-1)−	8.41E-1(3.55E-2)	1.63E+1(5.24E+0)−	5.97E+0(1.63E+1)	1.49E+1(7.86E-1)−	8.61E-1(1.14E-6)
LSMOP4	2	200	1.51E-1(3.96E-3)−	9.87E-2(1.69E-3)	1.59E-1(1.01E-2)−	6.99E-2(6.41E-3)	1.41E-1(2.25E-3)−	9.65E-2(1.56E-3)	1.31E-1(2.43E-3)−	9.41E-2(2.27E-3)
		500	9.71E-2(2.47E-3)−	5.05E-2(1.14E-3)	9.18E-2(1.28E-3)−	4.18E-2(2.52E-3)	7.84E-2(1.17E-3)−	4.66E-2(7.59E-4)	9.00E-2(2.21E-3)−	5.06E-2(9.20E-4)
		1000	6.26E-2(9.58E-4)−	3.20E-2(9.49E-4)	5.42E-2(9.08E-4)−	2.42E-2(1.43E-3)	4.83E-2(5.52E-4)−	2.50E-2(4.18E-4)	6.50E-2(1.08E-3)−	3.12E-2(9.27E-4)
	3	200	3.20E-1(6.48E-3)−	2.92E-1(8.37E-3)	2.87E-1(6.01E-3)−	2.31E-1(8.50E-3)	2.97E-1(1.09E-2)−	2.73E-1(1.30E-2)	3.27E-1(1.05E-2)−	2.72E-1(7.12E-3)
		500	1.93E-1(4.24E-3)+	2.13E-1(4.72E-3)	1.65E-1(1.76E-3)−	1.29E-1(3.44E-3)	1.90E-1(4.21E-3)−	1.83E-1(9.10E-3)	1.94E-1(2.27E-3)−	1.68E-1(4.74E-3)
		1000	1.29E-1(4.51E-3)+	1.41E-1(3.63E-3)	1.09E-1(1.50E-3)−	8.83E-2(2.32E-3)	1.26E-1(2.24E-3)+	1.32E-1(2.59E-3)	1.18E-1(1.42E-3)−	1.10E-1(1.88E-3)
LSMOP5	2	200	2.18E+0(4.38E-1)−	7.42E-1(1.14E-6)	6.40E-1(4.20E-2)+	7.42E-1(1.14E-6)	1.59E+0(4.72E-1)−	7.42E-1(1.14E-6)	6.33E-1(1.53E-1)+	7.42E-1(1.14E-6)
		500	8.21E+0(4.68E-1)−	7.42E-1(1.14E-6)	2.30E+0(2.69E-1)−	7.42E-1(1.14E-6)	7.33E+0(9.18E-1)−	7.42E-1(1.14E-6)	5.02E+0(3.62E-1)−	7.42E-1(1.14E-6)
		1000	1.12E+1(8.52E-1)−	7.42E-1(1.14E-6)	3.16E+0(1.86E-1)−	7.42E-1(1.14E-6)	1.10E+1(8.88E-1)−	7.42E-1(1.14E-6)	7.31E+0(5.20E-1)−	7.42E-1(1.14E-6)
	3	200	5.19E+0(5.61E-1)−	4.88E-1(5.13E-2)	2.79E+0(4.11E-1)−	4.99E-1(4.33E-2)	1.00E+0(3.95E-1)−	6.14E-1(9.91E-2)	3.35E+0(1.80E+0)−	6.30E-1(1.69E-1)
		500	1.17E+1(9.56E-1)−	5.35E-1(1.23E-2)	3.59E+0(3.91E-1)−	5.41E-1(2.47E-3)	9.42E+0(1.15E+0)−	9.51E-1(2.57E-1)	1.16E+1(1.21E+0)−	7.37E-1(2.02E-1)
		1000	1.62E+1(8.65E-1)−	5.49E-1(2.83E-2)	3.78E+0(2.09E-1)−	5.42E-1(1.60E-4)	1.75E+1(2.25E+0)−	9.02E-1(2.06E-2)	1.47E+1(1.77E+0)−	8.04E-1(1.98E-1)
LSMOP6	2	200	8.97E-1(8.91E-3)−	3.59E-1(2.37E-3)	7.59E-1(5.31E-2)−	3.32E-1(1.64E-2)	9.00E-1(8.66E-3)−	3.58E-1(4.24E-3)	9.60E-1(6.99E-1)−	3.58E-1(1.68E-3)
		500	8.09E-1(1.76E-3)−	3.22E-1(4.69E-4)	7.34E-1(8.50E-2)−	2.87E-1(3.04E-2)	8.08E-1(7.01E-4)−	3.22E-1(1.28E-3)	7.80E-1(6.42E-2)−	3.22E-1(2.63E-4)
		1000	7.75E-1(4.05E-4)−	3.14E-1(6.41E-4)	6.98E-1(1.23E-1)−	2.87E-1(2.74E-2)	7.71E-1(1.61E-2)−	3.14E-1(7.02E-4)	7.35E-1(8.16E-2)−	3.14E-1(1.70E-4)
	3	200	9.64E+1(1.55E+2)−	6.97E-1(1.63E-2)	3.05E+0(1.30E+0)−	6.76E-1(2.25E-2)	3.07E+0(1.03E+0)−	1.62E+0(9.13E-2)	5.13E+1(8.58E+1)−	8.37E-1(3.69E-1)
		500	3.76E+3(1.38E+3)−	7.42E-1(1.70E-2)	2.21E+1(1.72E+1)−	6.78E-1(4.09E-2)	8.44E+1(5.20E+1)−	2.31E+0(1.27E+0)	2.60E+3(1.05E+3)−	7.37E-1(2.11E-2)
		1000	1.24E+4(2.36E+3)−	7.45E-1(2.06E-2)	1.80E+2(8.33E+1)−	7.00E-1(1.47E-2)	1.61E+3(4.90E+2)−	2.05E+0(4.84E-1)	4.95E+3(2.10E+3)−	8.87E-1(6.58E-1)
LSMOP7	2	200	6.15E+1(8.08E+1)−	1.48E+0(2.65E-3)	4.04E+0(7.20E-1)−	1.48E+0(1.82E-3)	2.02E+1(5.37E+1)−	1.48E+0(1.71E-3)	2.52E+0(6.97E-1)−	1.47E+0(3.99E-3)
		500	1.45E+3(1.98E+3)−	1.50E+0(8.71E-4)	2.88E+1(4.97E+0)−	1.50E+0(6.11E-4)	4.74E+2(4.38E+2)−	1.50E+0(1.26E-3)	8.29E+1(1.36E+2)−	1.50E+0(1.35E-3)
		1000	8.24E+3(3.61E+3)−	1.51E+0(4.22E-4)	2.20E+2(4.85E+1)−	1.51E+0(3.19E-4)	4.15E+3(1.90E+3)−	1.51E+0(7.46E-4)	2.05E+3(5.98E+2)−	1.51E+0(7.37E-4)
	3	200	1.78E+0(8.52E-2)−	9.67E-1(2.51E-2)	1.17E+0(6.62E-2)−	8.97E-1(3.29E-2)	3.93E+1(1.85E+1)−	1.05E+0(1.71E-1)	1.89E+0(8.59E-2)−	1.04E+0(7.82E-2)
		500	1.29E+0(1.30E-2)−	8.96E-1(6.81E-3)	1.15E+0(9.17E-3)−	8.51E-1(3.19E-2)	3.98E+3(1.30E+3)−	1.03E+0(9.99E-2)	5.11E+1(2.23E+2)−	9.47E-1(7.64E-2)
		1000	1.10E+0(2.50E-3)−	8.68E-1(1.13E-2)	1.05E+0(2.96E-3)−	8.23E-1(6.78E-2)	3.17E+4(9.76E+3)−	9.75E-1(8.31E-2)	9.32E+2(3.64E+3)−	9.24E-1(8.98E-2)
LSMOP8	2	200	8.88E-1(5.54E-2)−	7.42E-1(1.14E-6)	3.79E-1(1.14E-1)+	7.40E-1(7.96E-3)	8.49E-1(6.41E-2)−	7.42E-1(1.14E-6)	6.66E-1(1.93E-1)≈	7.42E-1(1.14E-6)
		500	3.40E+0(2.81E-1)−	7.42E-1(1.14E-6)	6.34E-1(3.22E-2)+	7.42E-1(1.14E-6)	2.98E+0(3.05E-1)−	7.42E-1(1.14E-6)	2.84E+0(2.05E-1)−	7.42E-1(1.14E-6)
		1000	6.83E+0(4.47E-1)−	7.42E-1(1.14E-6)	1.26E+0(8.71E-2)−	7.42E-1(1.14E-6)	6.23E+0(3.12E-1)−	7.42E-1(1.14E-6)	4.89E+0(2.28E-1)−	7.42E-1(1.14E-6)
	3	200	5.70E-1(7.28E-2)−	3.63E-1(1.38E-2)	7.56E-1(1.02E-1)−	3.37E-1(2.79E-2)	4.42E-1(5.72E-2)+	5.34E-1(4.84E-2)	3.39E-1(4.42E-2)≈	3.56E-1(1.07E-2)
		500	9.64E-1(1.12E-2)−	3.53E-1(4.70E-2)	5.51E-1(6.05E-3)−	3.27E-1(3.14E-2)	1.74E+0(1.41E+0)−	5.40E-1(1.13E-2)	8.36E-1(9.70E-2)−	3.16E-1(3.95E-2)
		1000	9.52E-1(1.82E-2)−	3.60E-1(4.27E-2)	5.35E-1(5.24E-3)−	3.02E-1(4.71E-2)	2.43E+0(3.19E+0)−	5.35E-1(2.17E-2)	9.59E-1(2.61E-4)−	3.01E-1(2.93E-2)
LSMOP9	2	200	1.78E+0(4.84E-2)−	8.10E-1(1.14E-6)	4.44E-1(1.06E-2)+	8.10E-1(1.14E-6)	1.76E+0(2.74E-2)−	8.10E-1(2.25E-3)	1.54E+0(1.91E-1)−	8.10E-1(1.14E-6)
		500	1.38E+0(4.94E-2)−	8.10E-1(6.01E-4)	4.93E-1(2.47E-2)+	8.09E-1(8.96E-4)	1.32E+0(3.73E-2)−	8.09E-1(4.53E-4)	1.23E+0(7.34E-3)−	8.09E-1(8.64E-4)
		1000	4.80E+0(6.96E-1)−	8.08E-1(1.49E-3)	9.43E-1(1.22E-1)−	8.09E-1(1.88E-3)	4.02E+0(6.33E-1)−	8.08E-1(1.08E-3)	1.22E+0(8.51E-2)−	8.07E-1(1.29E-3)
	3	200	3.66E+0(4.05E-1)−	1.54E+0(4.56E-6)	1.29E+0(3.37E-1)≈	1.15E+0(1.46E-3)	3.60E+0(7.43E-2)−	1.37E+0(5.29E-2)	2.54E+0(1.78E-1)−	1.15E+0(4.00E-4)
		500	9.17E+0(1.32E+0)−	1.54E+0(4.56E-6)	5.25E+0(6.43E-1)−	1.16E+0(7.52E-3)	7.20E+0(8.65E-1)−	1.43E+0(1.42E-1)	3.23E+0(7.16E-1)−	1.15E+0(2.89E-4)
		1000	2.04E+1(1.53E+0)−	1.38E+0(1.97E-1)	1.33E+1(1.27E+0)−	1.16E+0(1.14E-2)	2.34E+1(2.39E+0)−	1.17E+0(6.76E-2)	2.59E+1(2.44E+0)−	1.15E+0(7.36E-4)
+/ − / ≈			5/49/0	—-	5/48/1	—-	6/48/0	—-	4/48/2	—-

In Table I, the four original algorithms are outperformed by the LSMOF-based versions over 48 out of 56 test instances. On one hand, NSGA-II performs better than LS-NSGA-II on tri-objective LSMOP2 and LSMOP4, MOEA/D-DE mainly outperforms LS-MOEA/D-DE on four test instances with 200 and 500 decision variables, SMS-EMOA outperforms LS-SMS-EMOA on tri-objective LSMOP2 and two test instances with 200 decision variables, and CMOPSO has achieved two better results on LSMOP2 and other two test instances with 200 decision variables compared with LS-CMOPSO. Besides, most of the best results are achieved by LS-MOEA/D-DE, totaling 31 out of 56 test instances. It is mainly attributed to the fact that the LSMOP problems are designed with decision variables linked on the PSs, and the MOEA/D-DE is exactly tailored for such problems. All in all, the pairwise comparisons have demonstrated the capability of our proposed LSMOF in enhancing the performance of existing MOEAs on LSMOPs.

D. Computational Efficiency

Since one important motivation of this paper is to accelerate large-scale multiobjective optimization, we will further investigate the convergence speed and computation time of the compared algorithms.

1) Convergence Speed: The convergence profiles of the eight compared algorithms on LSMOP3 and LSMOP5 with 1000 decision variables are displayed in Fig. 4. As can be observed, the original algorithms converge slowly and thus have failed to achieve an acceptable accuracy level by the end of the evolution. By contrast, the LSMOF-based algorithms have already converged to a promising accuracy level at a very early stage of the evolution (i.e., before 10 000 FEs).

2) Computation Time: In order to investigate the computation time of the proposed LSMOF, we display the average computation time of the eight compared algorithms on LSMOP3, LSMOP6, and LSMOP9.

As shown in Figs. 5 and 6, our proposed LSMOF has accelerated the computation time of MOEA/D-DE and SMS-EMOA on all the test instances. As for NSGA-II and CMOPSO, LSMOF has saved almost 1/3 computation time on problems with 500 and 1000 decision variables.

In conclusion, our proposed LSMOF is capable of reducing the computation time of MOEAs in large-scale multiobjective optimization, and the acceleration improvement is more

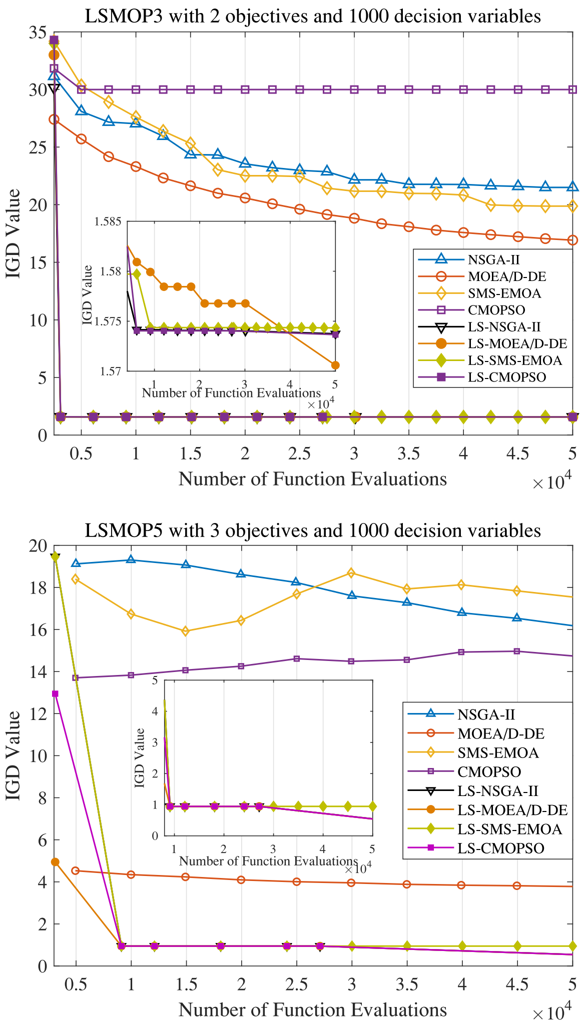


Fig. 4. Convergence profiles of eight compared algorithms on bi-objective LSMOP3 and tri-objective LSMOP5 with 1000 decision variables.

TABLE II
STATICS OF IGD RESULTS ACHIEVED BY THREE COMPARED ALGORITHMS ON 54 TEST INSTANCES FROM LSMOP TEST SUITE. THE BEST RESULT IN EACH ROW IS HIGHLIGHTED

Problem	M	D	MOEA/DVA	WOF-NSGA-II	LS-NSGA-II
LSMOP1	2	200	8.66E+0(8.04E-1)−	6.30E-1(9.36E-2)−	5.78E-1(5.32E-2)
		500	1.91E+1(1.00E+0)−	6.58E-1(6.11E-2)−	6.14E-1(2.54E-2)
		1000	2.39E+1(7.84E-1)−	6.79E-1(4.22E-2)−	6.37E-1(1.97E-2)
	3	200	6.26E+0(4.62E-1)−	6.95E-1(1.32E-1)−	5.24E-1(1.35E-2)
		500	9.42E+0(2.89E-1)−	7.09E-1(8.36E-2)−	5.96E-1(1.08E-2)
		1000	1.08E+1(3.22E-1)−	8.01E-1(7.05E-2)−	6.33E-1(1.34E-2)
LSMOP2	2	200	1.51E-1(6.75E-4)−	7.46E-2(4.63E-4)−	3.85E-2(1.08E-3)
		500	7.27E-2(2.30E-4)−	3.30E-2(3.91E-4)−	2.32E-2(6.90E-4)
		1000	4.04E-2(3.87E-4)−	1.92E-2(3.40E-4)−	1.81E-2(5.41E-4)
	3	200	1.23E-1(2.61E-3)+	1.36E-1(3.84E-3)≈	1.38E-1(2.76E-3)
		500	7.89E-2(2.63E-3)+	8.54E-2(3.82E-3)≈	8.71E-2(3.29E-3)
		1000	6.48E-2(2.46E-3)+	7.00E-2(4.28E-3)≈	7.05E-2(3.08E-3)
LSMOP3	2	200	1.71E+1(1.30E+0)−	1.50E+0(6.88E-2)≈	1.54E+0(1.43E-3)
		500	2.87E+1(8.26E-1)−	1.57E+0(1.47E-3)−	1.57E+0(1.05E-3)
		1000	3.36E+1(6.07E-1)−	1.58E+0(1.61E-3)−	1.57E+0(2.28E-4)
	3	200	2.30E+1(3.53E+0)−	8.61E-1(3.38E-4)−	8.40E-1(2.51E-2)
		500	3.60E+1(2.95E+0)−	8.61E-1(1.30E-4)−	8.59E-1(3.26E-3)
		1000	4.02E+1(2.09E+0)−	8.61E-1(7.28E-4)≈	8.61E-1(7.03E-5)
LSMOP4	2	200	6.56E-1(9.76E-3)−	1.33E-1(1.51E-2)−	9.87E-2(1.69E-3)
		500	5.44E-1(1.90E-3)−	8.74E-2(6.83E-3)−	5.05E-2(1.14E-3)
		1000	4.61E-1(6.97E-4)−	5.99E-2(5.57E-3)−	3.20E-2(9.49E-4)
	3	200	3.26E-1(2.31E-3)−	3.15E-1(9.10E-3)−	2.92E-1(8.37E-3)
		500	1.94E-1(5.71E-4)+	2.14E-1(6.87E-3)≈	2.13E-1(4.72E-3)
		1000	1.20E-1(1.96E-4)+	1.39E-1(5.80E-3)≈	1.41E-1(3.63E-3)
LSMOP5	2	200	1.42E+1(6.21E-1)−	7.42E-1(1.14E-6)≈	7.42E-1(1.14E-6)
		500	2.09E+1(5.02E-1)−	7.42E-1(1.14E-6)≈	7.42E-1(1.14E-6)
		1000	2.41E+1(3.40E-1)−	7.42E-1(1.14E-6)≈	7.42E-1(1.14E-6)
	3	200	1.17E+1(9.27E-1)−	5.41E-1(1.02E-3)−	4.88E-1(5.13E-2)
		500	1.70E+1(6.15E-1)−	5.41E-1(4.66E-5)−	5.35E-1(1.23E-2)
		1000	1.91E+1(5.97E-1)−	5.41E-1(7.27E-5)≈	5.49E-1(2.83E-2)
LSMOP6	2	200	7.36E+2(6.12E+2)−	6.42E-1(7.36E-2)−	3.59E-1(2.37E-3)
		500	2.24E+3(2.14E+3)−	7.33E-1(1.76E-1)−	3.22E-1(4.69E-4)
		1000	2.99E+3(2.33E+3)−	6.82E-1(9.03E-4)−	3.14E-1(6.41E-4)
	3	200	1.77E+4(3.58E+3)−	1.22E+0(3.15E-3)−	6.97E-1(1.63E-2)
		500	3.05E+4(6.34E+3)−	1.29E+0(2.01E-3)−	7.42E-1(1.70E-2)
		1000	3.68E+4(7.07E+3)−	1.31E+0(1.31E-3)−	7.45E-1(2.06E-2)
LSMOP7	2	200	5.58E+4(6.03E+3)−	1.48E+0(2.34E-3)−	1.48E+0(2.65E-3)
		500	1.06E+5(5.12E+3)−	1.51E+0(1.18E-3)−	1.50E+0(8.71E-4)
		1000	1.33E+5(4.14E+3)−	1.51E+0(1.19E-3)−	1.51E+0(4.22E-4)
	3	200	1.80E+0(3.92E-2)−	9.78E-1(4.70E-2)≈	9.67E-1(2.51E-2)
		500	1.27E+0(9.73E-3)−	9.48E-1(1.26E-1)−	8.96E-1(6.81E-3)
		1000	1.10E+0(2.56E-3)−	9.23E-1(1.38E-1)−	8.68E-1(1.13E-2)
LSMOP8	2	200	1.40E+1(8.86E-1)−	7.42E-1(1.14E-6)≈	7.42E-1(1.14E-6)
		500	2.11E+1(4.21E-1)−	7.42E-1(1.14E-6)≈	7.42E-1(1.14E-6)
		1000	2.39E+1(4.73E-1)−	7.42E-1(1.14E-6)≈	7.42E-1(1.14E-6)
	3	200	6.69E-1(1.07E-2)−	3.65E-1(4.56E-3)−	3.63E-1(1.38E-2)
		500	6.51E-1(6.13E-3)−	3.55E-1(1.59E-2)−	3.53E-1(4.70E-2)
		1000	6.49E-1(4.56E-3)−	3.56E-1(9.05E-3)+	3.60E-1(4.27E-2)
LSMOP9	2	200	2.26E+1(1.92E+0)−	8.10E-1(1.14E-6)≈	8.10E-1(1.14E-6)
		500	4.32E+1(1.36E+0)−	8.10E-1(3.21E-4)≈	8.10E-1(6.01E-4)
		1000	5.24E+1(1.03E+0)−	8.09E-1(4.10E-4)−	8.08E-1(1.49E-3)
	3	200	6.70E+1(5.47E+0)−	7.74E-1(3.80E-1)+	1.54E+0(4.56E-6)
		500	1.15E+2(5.42E+0)−	8.21E-1(4.13E-1)+	1.54E+0(4.56E-6)
		1000	1.37E+2(3.51E+0)−	1.08E+0(4.00E-1)+	1.38E+0(1.97E-1)
+/ − / ≈			5/49/0	4/32/18	

significant on LSMOPs with a larger number of decision variables, e.g., LSMOPs with more than 500 decision variables. There are two main reasons that our proposed LSMOF performs efficiently in terms of both convergence rate and computation time. First of all, since the original LSMOP is reformulated into a low-dimensional SOP, the search complexity is significantly reduced. Second, since the proposed framework guides the early search toward the PS by the reference directions, the population exhibits more stable and efficient convergence behaviors.

E. Comparisons With State-of-the-Arts

In this section, we compare our proposed LSMOF with another two state-of-the-art large-scale MOEAs, namely MOEA/DVA and WOF, in terms of both optimization performance and computational efficiency. In both WOF and LSMOF, NSGA-II is embedded for fair comparisons. The statistics of IGD results achieved by MOEA/DVA, WOF-NSGA-II, and LS-NSGA-II are displayed in Table II.

As can be observed, LS-NSGA-II has achieved 32 out of 54 best results, WOF-NSGA-II has achieved six best results, and MOEA/DVA has achieved five best results. To be specific, LS-NSGA-II has achieved the best results mainly on LSMOP1, LSMOP3, LSMOP5–LSMOP8, and bi-objective LSMOP2 and LSMOP9; WOF-NSGA-II has achieved the best results mainly on LSMOP5 and tri-objective LSMOP9; meanwhile, MOEA/DVA has achieved the best results on tri-objective LSMOP2 and tri-objective LSMOP4.

It should be noted that MOEA/DVA has achieved some results far from the PFs on LSMOP6 and bi-objective LSMOP7. This may be attributed to the failure of decision variables analysis which has caused the significant performance degeneration. After comparing the performance of these three algorithms on LSMOPs, their convergence rates on bi-objective LSMOP1 and tri-objective LSMOP6 with 1000

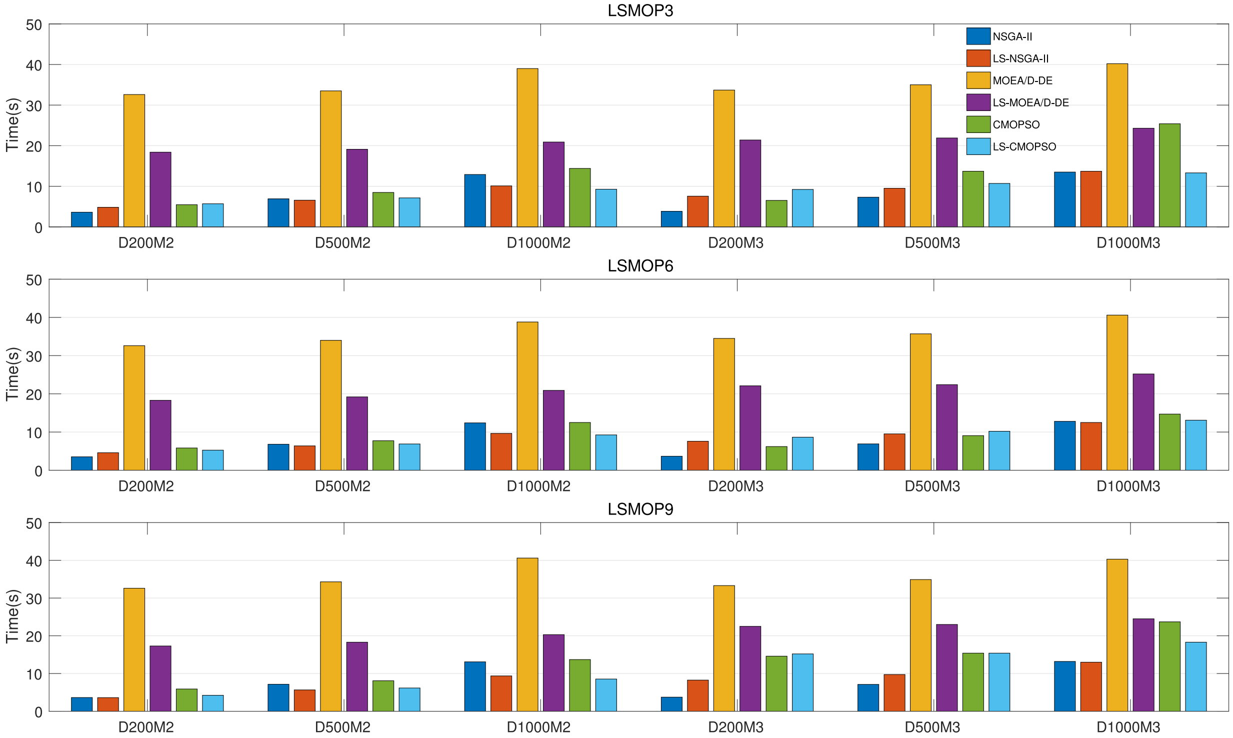


Fig. 5. Average computation time of NSGA-II, MOEA/D-DE, CMOPSO, and their LSMOF-based versions on LSMOP3, LSMOP6, and LSMOP9, where M denotes the number of objectives and D denotes the number of decision variables.

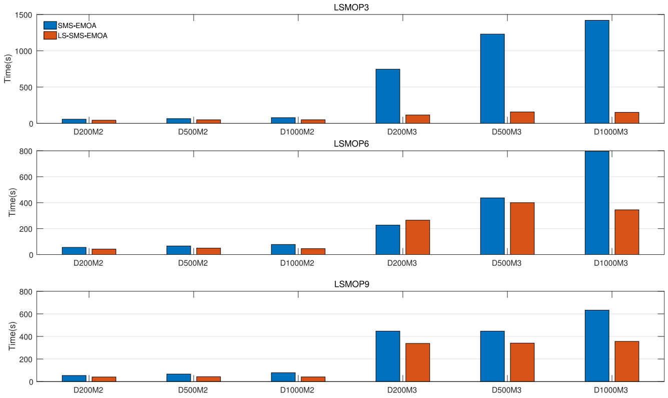


Fig. 6. Average computation time of SMS-EMOA and LS-SMS-EMOA on LSMOP3, LSMOP6, and LSMOP9, where M denotes the number of objectives and D denotes the number of decision variables.

decision variables are presented in Fig. 7, and the average computation time on LSMOP1 is displayed in Fig. 8. It can be observed from these two figures that LS-NSGA-II has the fastest convergence rate on those two test instances, while its computation time is similar to that of MOEA/DVA and WOF-NSGA-II. Besides, it can be observed that LS-NSGA-II shows fast convergence at the early stage of the evolution (within 10 000 FEs) and stops converging before the second stage starts. This is attributed to the fact that LSMOF obtains the quasi-optimal solutions at the first stage, and then the embedded MOEA spreads the obtained solutions over the entire PS. This can be confirmed by the continuous improvement of the IGD values after the first stage.

TABLE III
STATICS OF IGD RESULTS ACHIEVED BY LSMOF WITH DIFFERENT SETTINGS OF THRESHOLD VALUES

Problem	Obj.	tr =0.2	tr =0.4	tr =0.6	tr =0.8
LSMOP1	2	6.39E-1(2.07E-2)	6.31E-1(1.90E-2)	6.31E-1(2.24E-2)	6.36E-1(2.14E-2)
	3	6.05E-1(7.61E-3)	6.21E-1(7.88E-3)	6.42E-1(1.54E-2)	6.76E-1(1.36E-2)
LSMOP2	2	2.00E-2(3.56E-8)	2.00E-2(3.56E-8)	2.00E-2(3.56E-8)	1.85E-2(3.66E-3)
	3	7.00E-2(4.59E-3)	6.90E-2(3.08E-3)	7.05E-2(3.94E-3)	7.00E-2(4.59E-3)
LSMOP3	2	1.57E+0(0.00E+0)	1.57E+0(3.08E-3)	1.57E+0(2.24E-3)	1.57E+0(0.00E+0)
	3	8.60E-1(2.28E-6)	8.60E-1(2.28E-6)	8.60E-1(2.28E-6)	8.60E-1(2.28E-6)
LSMOP4	2	3.00E-2(1.78E-7)	3.00E-2(1.78E-7)	3.00E-2(1.78E-7)	3.00E-2(1.78E-7)
	3	1.42E-1(3.66E-3)	1.42E-1(3.66E-3)	1.40E-1(5.62E-3)	1.32E-1(4.10E-3)
LSMOP5	2	7.40E-1(1.14E-6)	7.40E-1(1.14E-6)	7.40E-1(1.14E-6)	7.40E-1(1.14E-6)
	3	5.43E-1(1.59E-2)	5.51E-1(3.45E-2)	5.62E-1(4.06E-2)	7.31E-1(1.46E-1)
LSMOP6	2	3.10E-1(1.14E-6)	3.11E-1(2.24E-3)	3.11E-1(2.24E-3)	3.10E-1(1.14E-6)
	3	7.59E-1(3.20E-2)	7.51E-1(2.96E-2)	7.59E-1(1.57E-2)	7.87E-1(1.42E-2)
LSMOP7	2	1.51E+0(6.83E-6)	1.51E+0(6.83E-6)	1.51E+0(6.83E-6)	1.51E+0(6.83E-6)
	3	8.62E-1(3.66E-3)	8.53E-1(3.23E-2)	8.67E-1(1.08E-2)	9.13E-1(4.51E-2)
LSMOP8	2	7.40E-1(1.14E-6)	7.40E-1(1.14E-6)	7.40E-1(1.14E-6)	7.40E-1(1.14E-6)
	3	3.47E-1(3.89E-2)	3.55E-1(2.72E-2)	3.90E-1(6.22E-2)	4.98E-1(6.76E-2)
LSMOP8	2	8.10E-1(2.28E-6)	8.10E-1(2.28E-6)	8.10E-1(2.28E-6)	8.10E-1(2.28E-6)
	3	1.44E+0(1.74E-1)	1.38E+0(1.99E-1)	1.41E+0(1.88E-1)	1.48E+0(1.17E-1)

In conclusion, the proposed LSMOF shows a competitive performance and similar computational efficiencies in comparison with MOEA/DVA and WOF on these LSMOPs. The competitiveness of LSMOF in comparison with the state-of-the-arts is verified.

F. Parameter Sensitivity Analysis

In our proposed LSMOF, a threshold tr is used to control the number of evaluations used by the first stage. To analyze the effect of the threshold value on the performance of LSMOF, we conduct experiments on a set of LSMOP problems with 1000 decision variables. The threshold is set to 0.2, 0.4, 0.6,

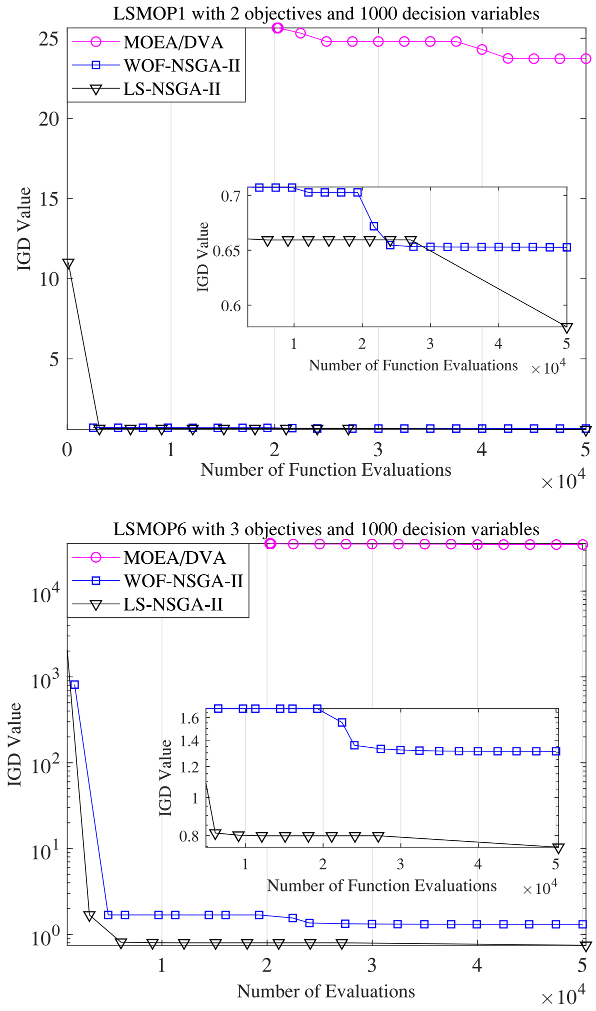


Fig. 7. Convergence rates of three compared algorithms on bi-objective LSMOP1 and tri-objective LSMOP6 with 1000 decision variables.

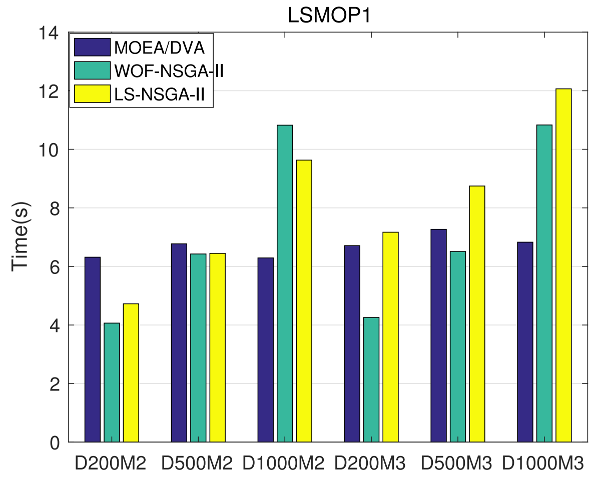


Fig. 8. Average computation time of MOEA/DVA, WOF-NSGA-II, and LS-NSGA-II on LSMOP1, where M is the number of objectives and D is the number of decision variables.

and 0.8, respectively. The experimental results are shown in Table III. It can be observed that different settings of tr do not significantly affect the performance of the proposed algorithm. This can be attributed to the fact that the algorithm converges so fast during the first stage that it does not need too many FEs. Therefore, the performance of the proposed LSMOF is not sensitive to the setting of the threshold. For simplicity, we set the threshold value to 0.5 in all the other experiments in this paper.

V. Conclusion

In this paper, we have proposed a general framework for large-scale multiobjective optimization, termed LSMOF. The proposed LSMOF adopts a two-stage strategy, where the first stage conducts the problem reformulation for obtaining a set of quash-optimal solutions near the PS, and the second stage spreads these solutions over the approximate PS uniformly by an embedded MOEA.

At the first stage of the proposed LSMOF, the decision space is first reconstructed by associating a set of reference solutions with a set of weight variables in the decision space. Then, a series of subproblems are constructed by taking the weight variables as the input, where each weight variable is aimed at tracking a specific point on the PS. Meanwhile, a performance indicator is adopted to assess the quality of the reconstructed decision vector for objective space reduction, and the original LSMOP is thus reformulated into a low-dimensional SOP. The DE algorithm is adopted to optimize the SOP by using an indicator-based fitness assignment strategy, and the candidate solutions obtained therein are used as the initial population of the embedded MOEA at the second stage.

To assess the performance of the proposed LSMOF, a variety of empirical comparisons have been conducted on a set of LSMOPs. The general performance of our proposed LSMOF is tested by embedding four MOEAs, namely NSGA-II, MOEA/D-DE, SMS-EMOA, and CMOPSO into it. The statistical results indicate that LSMOF has accelerated the convergence speed and saved computation time of the embedded algorithms on most of the test instances. More importantly, the performance of the MOEAs has also been significantly improved. The second experiment assesses the performance of the proposed LSMOF in comparison with two state-of-the-art large-scale MOEAs, namely MOEA/DVA and WOF. The superiority of the proposed LSMOF over the other two algorithms is also verified by the experimental results.

The proposed LSMOF has shown good potential in large-scale multiobjective optimization. Future work on developing more efficient problem reformulation method is highly desirable. It is also interesting to adapt our proposed LSMOF to real-world LSMOPs with more decision variables by parallel (e.g., GPU-based) computing.

References

[1] A. Ponsich, A. L. Jaimes, and C. A. Coello Coello, "A survey on multiobjective evolutionary algorithms for the solution of the portfolio optimization problem and other finance and economics applications," *IEEE Trans. Evol. Comput.*, vol. 17, no. 3, pp. 321–344, Jun. 2013.

[2] H. Ishibuchi and T. Murata, "A multi-objective genetic local search algorithm and its application to flowshop scheduling," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 28, no. 3, pp. 392–403, Aug. 1998.

[3] Y. Tian, H. Wang, X. Zhang, and Y. Jin, "Effectiveness and efficiency of non-dominated sorting for evolutionary multi- and many-objective optimization," *Complex Intell. Syst.*, vol. 3, no. 4, pp. 247–263, 2017.

[4] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. Chichester, U.K.: Wiley, 2001.

[5] 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.

[6] 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.

[7] K. Deb, M. Mohan, and S. Mishra, "Evaluating the ϵ-domination based multi-objective evolutionary algorithm for a quick computation of Pareto-optimal solutions," *Evol. Comput.*, vol. 13, no. 4, pp. 501–525, 2005.

[8] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization," in *Proc. 5th Conf. Evol. Methods Design Optim. Control Appl. Ind. Probl.*, 2001, pp. 95–100.

[9] 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.

[10] 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.

[11] Y. Tian, X. Zhang, R. Cheng, and Y. Jin, "A multi-objective evolutionary algorithm based on an enhanced inverted generational distance metric," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Vancouver, BC, Canada, 2016, pp. 5222–5229.

[12] 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.

[13] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "Test problems for large-scale multiobjective and many-objective optimization," *IEEE Trans. Cybern.*, vol. 47, no. 12, pp. 4108–4121, Dec. 2017.

[14] L. Parsons, E. Haque, and H. Liu, "Subspace clustering for high dimensional data: A review," *ACM SIGKDD Explor. Newslett.*, vol. 6, no. 1, pp. 90–105, 2004.

[15] H. Wang, L. Jiao, R. Shang, S. He, and F. Liu, "A memetic optimization strategy based on dimension reduction in decision space," *Evol. Comput.*, vol. 23, no. 1, pp. 69–100, 2015.

[16] M. N. Omidvar, X. Li, Y. Mei, and X. Yao, "Cooperative co-evolution with differential grouping for large scale optimization," *IEEE Trans. Evol. Comput.*, vol. 18, no. 3, pp. 378–393, Jun. 2014.

[17] S. Mahdavi, M. E. Shiri, and S. Rahnamayan, "Metaheuristics in large-scale global continues optimization: A survey," *Inf. Sci.*, vol. 295, pp. 407–428, Feb. 2015.

[18] X. Ma *et al.*, "A multiobjective evolutionary algorithm based on decision variable analyses for multiobjective optimization problems with large-scale variables," *IEEE Trans. Evol. Comput.*, vol. 20, no. 2, pp. 275–298, Apr. 2016.

[19] X. Zhang, Y. Tian, R. Cheng, and Y. Jin, "A decision variable clustering-based evolutionary algorithm for large-scale many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 22, no. 1, pp. 97–112, Feb. 2016.

[20] L. M. Antonio and C. A. Coello Coello, "Coevolutionary multiobjective evolutionary algorithms: Survey of the state-of-the-art," *IEEE Trans. Evol. Comput.*, vol. 22, no. 6, pp. 851–865, Dec. 2018.

[21] L. M. Antonio and C. A. Coello Coello, "Use of cooperative coevolution for solving large scale multiobjective optimization problems," in *Proc. IEEE Congr. Evol. Comput.*, Cancún, Mexico, 2013, pp. 2758–2765.

[22] M. N. Omidvar, X. Li, Z. Yang, and X. Yao, "Cooperative co-evolution for large scale optimization through more frequent random grouping," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Barcelona, Spain, 2010, pp. 1–8.

[23] S. Van Aelst, X. S. Wang, R. H. Zamar, and R. Zhu, "Linear grouping using orthogonal regression," *Comput. Stat. Data Anal.*, vol. 50, no. 5, pp. 1287–1312, 2006.

[24] W. Chen, T. Weise, Z. Yang, and K. Tang, "Large-scale global optimization using cooperative coevolution with variable interaction learning," in *Proc. 11th Int. Conf. Parallel Problem Solving Nat. (PPSN)*, Kraków, Poland, 2010, pp. 300–309.

[25] M. N. Omidvar, M. Yang, Y. Mei, X. Li, and X. Yao, "DG2: A faster and more accurate differential grouping for large-scale black-box optimization," *IEEE Trans. Evol. Comput.*, vol. 21, no. 6, pp. 929–942, Dec. 2017.

[26] H. Zille, H. Ishibuchi, S. Mostaghim, and Y. Nojima, "A framework for large-scale multiobjective optimization based on problem transformation," *IEEE Trans. Evol. Comput.*, vol. 22, no. 2, pp. 260–275, Apr. 2018.

[27] X. Zhang, X. Zheng, R. Cheng, J. Qiu, and Y. Jin, "A competitive mechanism based multi-objective particle swarm optimizer with fast convergence," *Inf. Sci.*, vol. 427, pp. 63–76, Feb. 2018.

[28] R. Cheng and Y. Jin, "A competitive swarm optimizer for large scale optimization," *IEEE Trans. Cybern.*, vol. 45, no. 2, pp. 191–204, Feb. 2015.

[29] T. Weise, R. Chiong, and K. Tang, "Evolutionary optimization: Pitfalls and booby traps," *J. Comput. Sci. Technol.*, vol. 27, no. 5, pp. 907–936, 2012.

[30] M. Munetomo and D. E. Goldberg, "Identifying linkage groups by nonlinearity/non-monotonicity detection," in *Proc. Genet. Evol. Comput. Conf.*, 1999, pp. 433–440.

[31] Y. Chen, "Extending the scalability of linkage learning genetic algorithms: Theory and practice," Ph.D. dissertation, Dept. Comput. Sci., Univ. Illinois Urbana–Champaign, Urbana, IL, USA, 2004.

[32] K. Li, M. N. Omidvar, K. Deb, and X. Yao, "Variable interaction in multi-objective optimization problems," in *Proc. Int. Conf. Parallel Problem Solving Nat.*, Edinburgh, U.K., 2016, pp. 399–409.

[33] Q. Zhang and H. Muhlenbein, "On the convergence of a class of estimation of distribution algorithms," *IEEE Trans. Evol. Comput.*, vol. 8, no. 2, pp. 127–136, Apr. 2004.

[34] Z. Yang, K. Tang, and X. Yao, "Large scale evolutionary optimization using cooperative coevolution," *Inf. Sci.*, vol. 178, no. 15, pp. 2985–2999, 2008.

[35] Y. Ling, H. Li, and B. Cao, "Cooperative co-evolution with graph-based differential grouping for large scale global optimization," in *Proc. IEEE 12th Int. Conf. Nat. Comput. Fuzzy Syst. Knowl. Disc. (ICNC-FSKD)*, Changsha, China, 2016, pp. 95–102.

[36] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Glob. Optim.*, vol. 11, no. 4, pp. 341–359, 1997.

[37] L. While, P. Hingston, L. Barone, and S. Huband, "A faster algorithm for calculating hypervolume," *IEEE Trans. Evol. Comput.*, vol. 10, no. 1, pp. 29–38, Feb. 2006.

[38] J. Zhang and A. C. Sanderson, "JADE: Adaptive differential evolution with optional external archive," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 945–958, Oct. 2009.

[39] X. Li and X. Yao, "Cooperatively coevolving particle swarms for large scale optimization," *IEEE Trans. Evol. Comput.*, vol. 16, no. 2, pp. 210–224, Apr. 2012.

[40] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable multi-objective optimization test problems," in *Proc. Congr. Evol. Comput.*, Honolulu, HI, USA, 2002, pp. 825–830.

[41] S. Huband, P. Hingston, L. Barone, and 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.

[42] W. Haynes, "Wilcoxon rank sum test," in *Encyclopedia of Systems Biology*. New York, NY, USA: Springer, 2013, pp. 2354–2355.

[43] A. Zhou, Y. Jin, Q. Zhang, B. Sendhoff, and E. Tsang, "Combining model-based and genetics-based offspring generation for multi-objective optimization using a convergence criterion," in *Proc. IEEE Congr. Evol. Comput.*, Vancouver, BC, Canada, 2006, pp. 892–899.

[44] Y. Tian, R. Cheng, X. Zhang, and Y. Jin, "PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [educational forum]," *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 73–87, Nov. 2017.

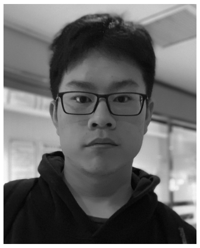
[45] Y. Tian, X. Xiang, X. Zhang, R. Cheng, and Y. Jin, "Sampling reference points on the Pareto fronts of benchmark multi-objective optimization problems," in *Proc. IEEE World Congr. Comput. Intell. (WCCI)*, Rio de Janeiro, Brazil, 2018, pp. 1–6.

[46] K. Deb and M. Goyal, "A combined genetic adaptive search (GeneAS) for engineering design," *Comput. Sci. Informat.*, vol. 26, no. 4, pp. 30–45, 1996.

[47] C. A. Coello Coello and M. S. Lechuga, "MOPSO: A proposal for multiple objective particle swarm optimization," in *Proc. Congr. Evol. Comput. IEEE World Congr. Comput. Intell.*, Honolulu, HI, USA, 2002, pp. 1051–1056.

Cheng He received the B.Eng. degree from the Wuhan University of Science and Technology, Wuhan, China, in 2012, and the Ph.D. degree from the Huazhong University of Science and Technology, Wuhan, in 2018.

He is currently a Post-Doctoral Research Fellow with the Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, China. His current research interests include model-based evolutionary algorithms, multiobjective optimization, large-scale optimization, deep learning, and their application.



Lianghao Li received the B.Eng. degree from the Huazhong University of Science and Technology, Wuhan, China, in 2015, where he is currently pursuing the M.Eng. degree with the School of Automation.

His current research interests include multiobjective optimization methods and their application.

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 interests include multiobjective optimization methods and their application.

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

Xingyi Zhang (SM'18) received the B.Sc. degree from Fuyang Normal College, Fuyang, China, in 2003, and the M.Sc. and Ph.D. degrees from the Huazhong University of Science and Technology, Wuhan, China, in 2006 and 2009, respectively.

He is currently a Professor with the School of Computer Science and Technology, Anhui University, Hefei, China. His current research interests include unconventional models and algorithms of computation, multiobjective optimization, and membrane computing.

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

Ran Cheng (M'16) received the B.Sc. degree from Northeastern University, Shenyang, China, in 2010, and the Ph.D. degree from the University of Surrey, Guildford, U.K., in 2016.

He is currently an Assistant Professor with the Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, China. His current research interests include evolutionary multiobjective optimization, model-based evolutionary algorithms, large-scale optimization, swarm intelligence, and deep learning.

Dr. Cheng was a recipient of the 2018 IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION Outstanding Paper Award, and the 2019 IEEE Computational Intelligence Society Outstanding Ph.D. Dissertation Award. He is the Founding Chair of the IEEE Symposium on Model Based Evolutionary Algorithms.

Yaochu Jin (M'98–SM'02–F'16) received the B.Sc., M.Sc., and Ph.D. degrees from Zhejiang University, Hangzhou, China, in 1988, 1991, and 1996, respectively, and the Dr.-Ing. degree from Ruhr University Bochum, Bochum, Germany, in 2001.

He is currently a Professor of Computational Intelligence with the Department of Computer Science, University of Surrey, Guildford, U.K., where he heads the Nature Inspired Computing and Engineering Group. He is also a Finland Distinguished Professor funded by the Finnish Funding Agency for Innovation (Tekes), Finland, and a Changjiang Distinguished Visiting Professor appointed by the Ministry of Education, China. He has (co)-authored over 300 peer-reviewed journal and conference papers and holds eight patents on evolutionary optimization. His current research interests include computational intelligence, computational neuroscience, computational systems biology, application of nature-inspired algorithms to solving real-world optimization, learning, and self-organization problems.

Dr. Jin was a recipient of the 2015 and 2017 *IEEE Computational Intelligence Magazine* Outstanding Paper Award, and the 2018 IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION Outstanding Paper Award. He is the Editor-in-Chief of the IEEE TRANSACTIONS ON COGNITIVE AND DEVELOPMENTAL SYSTEMS and *Complex and Intelligent Systems*. Since 2017, he has been an IEEE Distinguished Lecturer. He was the Vice President for the Technical Activities of the IEEE Computational Intelligence Society from 2014 to 2015.

Xin Yao (M'91–SM'96–F'03) received the B.Sc. and Ph.D. degrees from the University of Science and Technology of China, Hefei, China, in 1982 and 1990, respectively.

He is a Chair Professor of Computer Science with the Southern University of Science and Technology, Shenzhen, China, and a part-time Professor of Computer Science with the University of Birmingham, Birmingham, U.K. He has been researching multi-objective optimization since 2003, when he published a well-cited EMO'03 paper on many-objective optimization. His current research interests include evolutionary computation, ensemble learning, and their applications in software engineering.

Dr. Yao was a recipient of the 2001 IEEE Donald G. Fink Prize Paper Award, the 2010, 2016, and 2017 IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION Outstanding Paper Awards, the 2010 BT Gordon Radley Award for Best Author of Innovation (Finalist), the 2011 IEEE TRANSACTIONS ON NEURAL NETWORKS Outstanding Paper Award, the Prestigious Royal Society Wolfson Research Merit Award in 2012, the IEEE Computational Intelligence Society (CIS) Evolutionary Computation Pioneer Award in 2013, and several other best paper awards. From 2003 to 2008, he was the Editor-in-Chief of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION. He was the President of the IEEE CIS from 2014 to 2015. He is a Distinguished Lecturer of the IEEE CIS.