

A Decision Variable Clustering-Based Evolutionary Algorithm for Large-Scale Many-Objective Optimization

Xingyi Zhang, Ye Tian, Ran Cheng, and Yaochu Jin, *Fellow, IEEE*

Abstract—The current literature of evolutionary many-objective optimization is merely focused on the scalability to the number of objectives, while little work has considered the scalability to the number of decision variables. Nevertheless, many real-world problems can involve both many objectives and large-scale decision variables. To tackle such large-scale many-objective optimization problems (MaOPs), this paper proposes a specially tailored evolutionary algorithm based on a decision variable clustering method. To begin with, the decision variable clustering method divides the decision variables into two types: 1) convergence-related variables and 2) diversity-related variables. Afterward, to optimize the two types of decision variables, a convergence optimization strategy and a diversity optimization strategy are adopted. In addition, a fast nondominated sorting approach is developed to further improve the computational efficiency of the proposed algorithm. To assess the performance of the proposed algorithm, empirical experiments have been conducted on a variety of large-scale MaOPs with up to ten objectives and 5000 decision variables. Our experimental results demonstrate that the proposed algorithm has significant advantages over several state-of-the-art evolutionary algorithms in terms of the scalability to decision variables on MaOPs.

Index Terms—Clustering, evolutionary multiobjective optimization, large-scale optimization, many-objective optimization, nondominated sorting, tree.

I. INTRODUCTION

MANY-OBJECTIVE optimization problems (MaOPs) refer to the problems that involve more than three conflicting objectives to be optimized simultaneously, which

widely exist in real-world applications, such as engineering design [1], air traffic control [2], groundwater monitoring [3], and molecular design [4]. Generally speaking, MaOPs cannot be easily solved by most multiobjective evolutionary algorithms (MOEAs) designed for dealing with traditional multiobjective optimization problems (MOPs) which involve only two or three objectives due to two major issues [5]–[9]. The first issue is the loss of convergence pressure, which is mainly caused by the phenomenon called *dominance resistance* [1]. This is due to the fact that in many-objective optimization, most candidate solutions become nondominated with each other, thus causing failure of dominance-based selection strategies in traditional MOEAs. The other issue is the difficulty in diversity management. In many-objective optimization, the candidate solutions are sparsely distributed in the high-dimensional objective space, which leads to the ineffectiveness of traditional diversity management strategies. To tackle the above two issues, a number of approaches have been proposed [10], [11], which can be roughly divided into four categories.

The first category covers various convergence enhancement based approaches. To enhance the convergence pressure, the most intuitive idea is to directly modify the definition of traditional Pareto dominance, such as ϵ -dominance [12], [13], L -optimality [14], fuzzy dominance [15], preference order ranking [16], and θ -dominance [17]. Another idea belonging to this category is to combine the traditional dominance with an additional convergence-related metric, and MOEAs belonging to this category include the substitute distance assignment based NSGA-II [18], grid-based evolutionary algorithm [19], preference-inspired coevolutionary algorithm [20], many-objective evolutionary algorithm based on directional diversity and favorable convergence [21], and knee point driven evolutionary algorithm (KnEA) [22].

The second category directly adopts performance indicators as selection criterion to distinguish nondominated solutions which cannot be distinguished by traditional Pareto dominance. Among many others [23]–[26], some representative approaches of this category are indicator-based evolutionary algorithm [27], S -metric selection based evolutionary multiobjective optimization algorithm [28], and hypervolume (HV)-based evolutionary algorithm [29].

The third category is known as the decomposition-based approaches, where the idea is to decompose an MaOP into a group of simple subproblems and then solve them in a

Manuscript received March 14, 2016; revised June 27, 2016; accepted August 12, 2016. Date of publication August 16, 2016; date of current version January 26, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 61272152, Grant 61502004, Grant 61502001 and Grant 61672033, in part by the Joint Research Fund for Overseas Chinese, Hong Kong and Macao Scholars of the National Natural Science Foundation of China (Grant No. 61428302), and in part by the U.K. EPSRC Under Grant EP/M017869/1. The work of Y. Jin was supported in part by EC FP7, and in part by the U.K. EPSRC and industry. (*Corresponding author: Ran Cheng.*)

X. Zhang and Y. Tian are with the Institute of Bio-Inspired Intelligence and Mining Knowledge, School of Computer Science and Technology, Anhui University, Hefei 230039, China (e-mail: xyzhanghust@gmail.com; field910921@gmail.com).

R. Cheng is with the School of Computer Science, University of Birmingham, Birmingham, B15 2TT, U.K. (e-mail: ranchengcn@gmail.com).

Y. Jin is 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 multimedia material available at <http://ieeexplore.ieee.org> provided by the authors.

Digital Object Identifier 10.1109/TEVC.2016.2600642

collaborative manner. To be specific, one type of decomposition is to decompose an MaOP into a group of single-objective problems (SOPs). One popular decomposition-based approach of this type is the MOEA/D algorithm [30], as well as its variants, such as stable matching model based MOEA/D [31], MOEA/D with adaptive weight vector adjustment [32], external archive guided MOEA/D [33], MOEA/D with a distance-based updating [34], and online diversity metric based MOEA/D [35]. Another type of decomposition is to decompose an MaOP into a group of simple MOPs. Some representative MOEAs of this type are MOEA/D-M2M [36], reference-point based many-objective NSGA-II (NSGA-III) [37], dominance and decomposition based MOEA [38], and the recently proposed reference vector guided evolutionary algorithm [39].

The main idea of the fourth category is to convert an MaOP into an MOP, so that it can be directly solved using existing MOEAs. There are two major branches along this direction. One branch is to use the objective reduction approaches to eliminate redundant or irrelevant objectives, such as dominance relation preservation based algorithms [40], [41], unsupervised feature selection based algorithms [42], Pareto corner search based algorithms [43], machine learning based objective reduction [44]–[46], and the recently proposed nonlinear correlation information entropy based objective reduction [47]. The other branch is to replace the original objectives with two or three newly defined objectives. Representative approaches of this type include bi-goal evolution [48] and summation of normalized objective values and diversified selection based MOEA [49].

Apart from the approaches proposed to enhance the search abilities of MOEAs in many-objective optimization, there is also some work motivated to improve the computational efficiency. For example, Bringmann *et al.* [50] suggested to use the Monte Carlo method to improve the computational efficiency of HV calculations in the multiobjective covariance matrix adaptation evolution strategy. Some fast nondominated sorting approaches were also developed to reduce the computational cost of Pareto-based MOEAs for solving MaOPs, such as deductive sort [51], corner sort [52], M-front [53], efficient nondominated sort (ENS) [54], and approximate nondominated sort [55].

However, in spite of the various approaches that are focused on the scalability of MOEAs to the number of objectives, little work has been reported to consider the large number of decision variables in multiobjective optimization. While large-scale optimization has already attracted certain interests in the single-objective literature [56]–[62], the development of large-scale multiobjective optimization is still in the infancy [63].

Recently, Ma *et al.* [64] proposed a decision variable analysis based MOEA, known as MOEA/DVA, for solving large-scale MOPs. In MOEA/DVA, a decision variable analysis method based on dominance relationships is designed to divide the decision variables into three groups: 1) convergence-related variables; 2) diversity-related variables; and 3) variables related to both convergence and diversity. The convergence-related variables and diversity-related variables

are optimized with different strategies separately, while the variables related to both convergence and diversity are simply treated as diversity-related variables. It has been shown that such a decision variable analysis method works efficiently on large-scale MOPs with two or three objectives, although its performance has not yet been assessed on any MaOP.

Following the basic idea of MOEA/DVA proposed in [64], in this paper, we propose an evolutionary algorithm for tackling large-scale MaOPs based on a decision variable clustering method, where the main new contributions are summarized as follows.

- 1) A decision variable clustering method is proposed to divide the decision variables into convergence-related and diversity-related ones. While the decision variable analysis method in MOEA/DVA [64] is based on dominance-based relationships, the proposed decision variable clustering method adopts the k -means method with features measured by the angles between the sample solutions and the direction of convergence, where smaller angles indicate more contributions to convergence and larger angles to diversity. Consequently, a decision variable can only be divided as either convergence-related or diversity-related, thus addressing the open issue that decision variables related to both convergence and diversity cannot be further distinguished [64].
- 2) An evolutionary algorithm for large-scale many-objective optimization (LMEA) is proposed based on the decision variable clustering method. In LMEA, two optimization strategies, i.e., the convergence optimization strategy and the diversity optimization strategy, are proposed to deal with the two types of decision variables separately. Both strategies adopt nondominated sorting as the first selection criterion but differ in the secondary selection criterion: to enhance convergence, the secondary selection in the convergence optimization strategy is based on the Euclidean distances from the candidate solutions to the ideal point; to manage diversity, the secondary selection in the diversity optimization strategy is based on the angles between the candidate solutions.
- 3) To reduce the computational cost of the two optimization strategies in LMEA, a computational efficient tree-based nondominated sorting approach, termed T-ENS, is developed on the basis of the recently proposed ENS [54]. In T-ENS, the information for identifying the nondominance relationship is recorded in the nodes of a tree, with which a large number of nondominance comparisons between solutions can be inferred. As a result, only a subset of the solutions, rather than all, that have been assigned to a nondominated front needs to be compared. Theoretical analysis indicates that the developed T-ENS holds a time complexity of $O(MN \log N / \log M)$, where N denotes the population size and M denotes the number of objectives, which is more efficient than the average complexity of $O(MN^2)$ held by most existing approaches for solving MaOPs where most solutions in the populations are nondominated with each other.

- 4) To assess the performance of the proposed LMEA, empirical evaluations have been conducted on a variety of benchmark problems in comparison with several state-of-the-art MOEAs for solving MaOPs as well as large-scale MOPs. Our experimental results demonstrate that LMEA is well suited for solving large-scale MaOPs having up to 5000 decision variables.

The rest of this paper is organized as follows. In Section II, we briefly recall some related work on MOEAs for solving large-scale MOPs. The motivation of this paper is also elaborated in Section II, mainly with respect to the recently proposed MOEA/DVA. In Section III, we describe the details of the proposed LMEA for large-scale MaOPs. Simulation results are presented in Section IV to empirically assess performance of the LMEA on large-scale MaOPs. Finally, the conclusions and future work are drawn in Section V.

II. RELATED WORK AND MOTIVATION

In this section, we first review several representative MOEAs for solving large-scale MOPs, and then elaborate the motivation of this paper by two illustrative examples. Note that, the current literature only covers a very limited number of MOEAs that can achieve a good performance on large-scale MOPs, some of which were even not explicitly designed for solving large-scale MOPs.

Antonio and Coello [63] suggested a cooperative coevolution framework for solving large-scale MOPs. The main idea of this algorithm is to randomly divide the large number of decision variables into several small subcomponents of equal size and coevolve these subcomponents cooperatively for a predefined number of cycles. Experimental results confirmed the competitiveness of the algorithm on large-scale MOPs with two and three objectives. Following this line, in [65], the idea of cooperative coevolution was also adopted to deal with large-scale multiobjective capacitated arc routing problems.

Most recently, an MOEA for solving large-scale MOPs, termed MOEA/DVA, has been suggested by Ma *et al.* [64], where a decision variable analysis strategy is adopted to divide the decision variables into different groups by checking the dominance relationships between solutions generated by perturbing the values of variables. To be specific, a decision variable is considered to be as follows.

- 1) Convergence-related, if and only if the solutions generated by the perturbed decision variable values are dominated one by one.
- 2) Diversity-related, if and only if the solutions generated by the perturbed decision variable values are nondominated with each other.
- 3) Related to both convergence and diversity, otherwise.

Once the decision variables are successfully divided into different groups, MOEA/DVA performs a two-stage optimization process: first, the convergence-related variables are optimized until the candidate solutions reach the Pareto front; afterward, the diversity-related variables are optimized to achieve a wide distribution of the candidate solutions on the Pareto front.

For the variables related to both convergence and diversity, MOEA/DVA simply treats them as diversity-related variables,

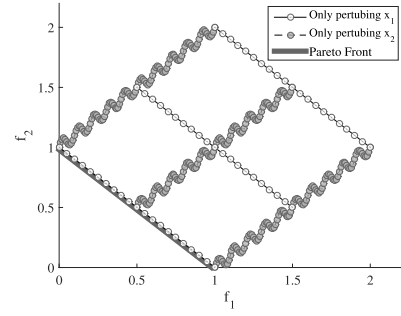


Fig. 1. Sampling points obtained on the MOP formulated in (1) by perturbing one variable and fixing the other to 0, 0.5, and 1, respectively.

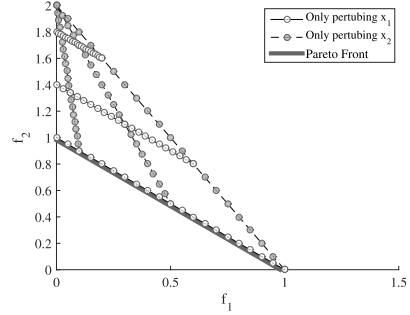


Fig. 2. Sampling points obtained on the MOP formulated in (2) by perturbing one variable between $[0, 1]$ while fixing the other as 0.2, 0.6, and 1, respectively.

as the variable analysis strategy is not able to further distinguish them [64]. Nevertheless, our empirical observation indicates that, these decision variables still can be further divided into two groups, as some of them contribute more to convergence while others contribute more to diversity. To illustrate such an observation, let us consider the following bi-objective MOP:

$$\begin{cases} \min f_1 = x_1 + x_2 \\ \min f_2 = (1 - x_1) + x_2 + \sin(20\pi x_2)/20 \\ \text{subject to: } x_i \in [0, 1], i = 1, 2. \end{cases} \quad (1)$$

Fig. 1 shows the sampling points obtained on the MOP formulated in (1) by perturbing one variable between $[0, 1]$ while fixing the other to 0, 0.5, and 1, respectively. According to the variable analysis strategy introduced above, MOEA/DVA labels x_1 and x_2 as a diversity-related variable and a variable related to both convergence and diversity, respectively. Nevertheless, since x_2 contributes more to convergence than diversity, it will be more beneficial to optimize x_2 as a convergence-related variable rather than a diversity-related one.

It should be noted that, even for certain diversity-related variables, sometime it is also essential to consider them as convergence-related to guarantee convergence of the population. Let us take the following bi-objective MOP as an example:

$$\begin{cases} \min f_1 = x_1 x_2 \\ \min f_2 = (1 - x_1 x_2) + 1 - x_2 \\ \text{subject to: } x_i \in [0, 1], i = 1, 2. \end{cases} \quad (2)$$

Fig. 2 shows the sampling points obtained on the MOP formulated in (2) by perturbing one variable between $[0, 1]$

Algorithm 1: Main Framework of LMEA

Input: N (population size), $nSel$ (number of selected solutions for decision variable clustering), $nPer$ (number of perturbations on each solution for decision variable clustering), $nCor$ (number of selected solutions for decision variable interaction analysis)

Output: P (final population)

```

1  $P \leftarrow \text{Initialize}(N)$ ;
2  $[DV, CV] \leftarrow \text{VariableClustering}(P, nSel, nPer)$ ;
3  $subCVs \leftarrow \text{InteractionAnalysis}(P, CV, nCor)$ ;
4 while termination criterion not fulfilled do
5    $P \leftarrow \text{ConvergenceOptimization}(P, subCVs)$ ;
6    $P \leftarrow \text{DiversityOptimization}(P, DV)$ ;
```

while fixing the other to 0.2, 0.6, and 1, respectively. According to the variable analysis strategy introduced above, MOEA/DVA labels both x_1 and x_2 as diversity-related variables. Nevertheless, in this case, it is more beneficial to label x_2 as a convergence-related variable due to the fact that optimizing x_2 will guide the population toward the Pareto front.

To address the above issue, this paper suggests a decision variable clustering based MOEA, termed LMEA, for large-scale MaOPs. In LMEA, a decision variable clustering method is proposed to distinguish convergence-related and diversity-related variables by measuring their contributions to convergence and diversity, respectively. With the proposed decision variable clustering method, variables similar to x_2 in the above two examples can be properly categorized.

III. PROPOSED ALGORITHM: LMEA

In this section, we first present the main framework of the proposed LMEA, then detail two important components in it, i.e., the decision variable clustering method and the fast tree-based nondominated sorting approach.

A. Main Framework of LMEA

Algorithm 1 presents the main framework of LMEA, which consists of the following five components. First, a population of N candidate solutions is randomly initialized. Second, the developed decision variable clustering method is applied to divide the variables into two groups, convergence-related variables and diversity-related variables. Third, the convergence-related variables are further divided into several subgroups based on the interaction between these variables, where the variables are interacted with each other inside one subgroup but not interacted with those in any other subgroups. The variables in each subgroup are also known as *interacting* variables, as they cannot be optimized separately due to the interactions between each other. Algorithm 2 presents the details of the interaction analysis for convergence-related variables, which adopts the strategy developed in [64]. Note that similar strategies have also been widely used for

Algorithm 2: InteractionAnalysis($P, CV, nCor$)

Input: P (current population), CV (set of convergence-related variables), $nCor$ (number of selected solutions)

Output: $subCVs$ (set of subgroups of interacted decision variables in CV)

```

1  $subCVs \leftarrow \emptyset$ ;
2 forall the  $v \in CV$  do
3    $CorSet \leftarrow \emptyset$ ;
4   forall the  $Group \in subCVs$  do
5     forall the  $u \in Group$  do
6        $flag \leftarrow \text{false}$ ;
7       for  $i = 1$  to  $nCor$  do
8         Randomly select a solution  $p$  from  $P$ ;
9         if  $v$  is interacted with  $u$  in  $p$  then
10            $flag \leftarrow \text{true}$ ;
11            $CorSet \leftarrow CorSet \cup \{Group\}$ ;
12           break;
13       if  $flag$  then
14         break;
15   if  $CorSet == \emptyset$  then
16      $subCVs \leftarrow subCVs \cup \{v\}$ ;
17   else
18      $subCVs \leftarrow subCVs / CorSet$ ;
19      $Group \leftarrow$  all variables in  $CorSet$  and  $v$ ;
20      $subCVs \leftarrow subCVs \cup \{Group\}$ ;
```

variable interaction detection in single-objective large-scale optimization [56], [66].

Algorithm 2 first initializes an empty set $subCVs$ of interacted variable subgroups, and afterward, the convergence-related variables in CV are assigned to different subgroups based on the pairwise interactions between them, where the interactions are defined as follows: given an MOP $\min \mathbf{f} = \min(f_1, f_2, \dots, f_m)$, if there exist $\mathbf{x}, a_1, a_2, b_1, b_2$ and at least one f_k , $1 \leq k \leq m$, such that $f_k(\mathbf{x})|_{x_i=a_2, x_j=b_1} < f_k(\mathbf{x})|_{x_i=a_1, x_j=b_1}$ and $f_k(\mathbf{x})|_{x_i=a_2, x_j=b_2} > f_k(\mathbf{x})|_{x_i=a_1, x_j=b_2}$, then the two decision variables x_i and x_j are regarded to be interacted, where $f_k(\mathbf{x})|_{x_i=a_2, x_j=b_1} = f_k(x_1, \dots, x_{i-1}, a_2, \dots, x_{j-1}, b_1, \dots, x_n)$. To be specific, if a variable has interaction with at least one existing variable in $subCVs$, then the two variables are assigned into the same subgroup; otherwise, it is assigned to a new subgroup. This operation is repeated until each convergence-related variable has been assigned to a subgroup. Therefore, in the extreme cases, there will be at most a number of $|CV|$ subgroups, which means that the convergence-related variables are fully separable; while if the variables are fully nonseparable, there will be only one subgroup.

Once the interaction analysis is done in Algorithm 2, LMEA starts to optimize each subgroup of variables one by one using a convergence optimization strategy as detailed in Algorithm 3, while the diversity-related variables are optimized using a diversity optimization strategy as detailed in Algorithm 4.

Algorithm 3: *ConvergenceOptimization*(P , $subCVs$)

Input: P (current population), $subCVs$ (set of groups of convergence-related variables)
Output: P (next population)

```

1  $Front \leftarrow NondominatedSort(P)$ ;
2 Calculate the distance between each solution in  $P$  and the
  origin in objective space;
3 forall the  $Group \in subCVs$  do
4    $nEvaluated \leftarrow 0$ ;
5   while  $nEvaluated < |P|$  do
6      $S \leftarrow \emptyset$ ;
7     for  $i = 1$  to  $|P|$  do
8       if  $rand() < Front[i]/max(Front)$  then
9          $S \leftarrow S \cup \{i\}$ ;
10     $O \leftarrow \emptyset$ ;
11    for  $s \in S$  do
12      Choose  $p_1$  and  $p_2$  from  $P$  by binary
        tournament, where the front number and the
        distance of each solution are used as the first
        and second criteria;
13       $/s'(Group)$  denotes a vector consisting of
        values of  $s'$  on decision variables in  $Group^*/$ 
14       $s'(Group) \leftarrow$ 
         $recombination(p_1(Group), p_2(Group))$ ;
15       $s'' \leftarrow s$ ;
16       $s''(Group) \leftarrow s'(Group)$ ;
17       $O \leftarrow O \cup \{s''\}$ ;
18     $nEvaluated \leftarrow nEvaluated + |O|$ ;
19     $Front \leftarrow NondominatedSort(P \cup O)$ ;
20    Calculate the distance between each solution in  $O$ 
    and the origin in objective space;
21    Use each solution in  $O$  to replace the
    corresponding solution in  $P$  according to the front
    number and the distance;
```

In the convergence optimization strategy, the proposed T-ENS is used to perform nondominated sorting on the parent population, and the Euclidean distance from each candidate solution to the ideal point¹ is calculated. Afterward, the offspring candidate solutions are generated by optimizing the convergence-related variables inside each subgroup independently. For the variables in each subgroup, two candidate solutions are randomly chosen from population P using the binary tournament, and afterward, an offspring candidate solution is generated by replacing the values of the variables in the same subgroup with those generated by recombination operators, while leaving the rest variables unchanged. If an offspring candidate solution has a better convergence than its parent, it will be selected to be passed to the next generation; otherwise, the offspring candidate solution is discarded and the parent candidate solution will survive. To

¹In this paper, the candidate solutions are all transformed to be inside the first quadrant of the objective space, such that the ideal point is always the axis origin.

Algorithm 4: *DiversityOptimization*(P , DV)

Input: P (current population), DV (diversity-related variables);
Output: Q (next population);

```

1  $O \leftarrow \emptyset$ ;
2 forall the  $p \in P$  do
3   Randomly choose  $p_1$  and  $p_2$  from  $P$ ;
4    $p'(DV) \leftarrow recombination(p_1(DV), p_2(DV))$ ;
5    $p'' \leftarrow p$ ;
6    $p''(DV) \leftarrow p'(DV)$ ;
7    $O \leftarrow O \cup \{p''\}$ ;
8  $F \leftarrow NondominatedSort(P \cup O)$ ;
9  $Q \leftarrow F_1 \cup F_2 \cup \dots \cup F_{k-1}$ , where  $k$  is the smallest value
  satisfying  $|F_1 \cup F_2 \cup \dots \cup F_k| > |P|$ ;
10 if  $Q = \emptyset$  then
11    $Q \leftarrow$  all the extreme solutions in  $F_k$ ;
12    $F_k \leftarrow F_k / Q$ ;
13 Calculate the angle  $Angle$  in objective space between
  each two solutions in  $Q \cup F_k$ ;
14 while  $|Q| < |P|$  do
15    $p \leftarrow \operatorname{argmax}_{x \in F_k} \min_{y \in Q} Angle[x][y]$ ;
16    $Q \leftarrow Q \cup \{p\}$ ;
```

be specific, an offspring candidate solution is considered to have better convergence quality than its parent solution if and only if it has a smaller nondominate front number in the nondominated sorting, or it has the same nondominated front number but a smaller Euclidean distance to the ideal point.

In the diversity optimization strategy, $|P|$ offsprings are first generated from population P by optimizing all diversity-related variables as a whole using the recombination operators, then the offspring population is combined with the parent population and the environmental selection performed on the combined population consists of two stages: to begin with, the candidate solutions from the first $k - 1$ fronts are selected, such that k is the smallest value satisfying $|F_1 \cup F_2 \cup \dots \cup F_k| > |P|$; afterward, the remaining candidate solutions of the best diversity are selected one by one from front F_k until the population size $|P|$ is reached, where the diversity is measured by the angles between the candidate solutions in the objective space. As reported by Cheng *et al.* [39], angle is a more efficient diversity measurement compared to Euclidean distance in many-objective optimization. Algorithms 3 and 4 are repeated until a termination condition is met.

It should be noted that, in both optimization strategies, any commonly adopted recombination operators such as simulated binary crossover (SBX) [67], polynomial mutation [68], and differential evolution (DE) [21], [69], [70] can be used to generate offspring candidate solutions.

In the following sections, we will describe in detail the decision variable clustering method and fast tree-based nondominated sorting approach, which are two most important components in the proposed LMEA.

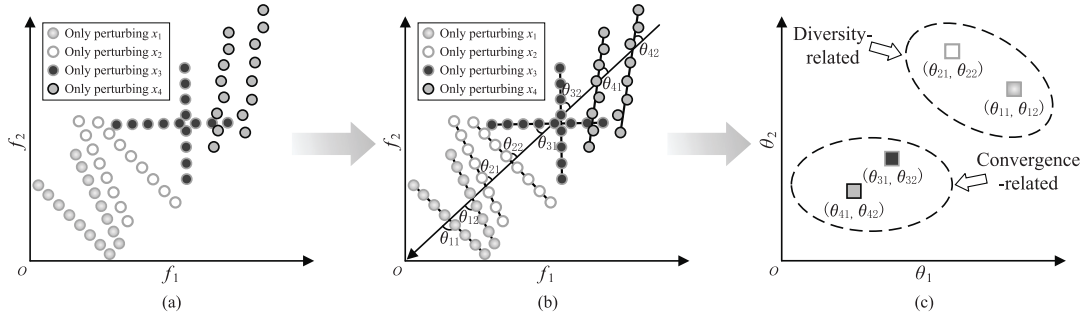


Fig. 3. Example to illustrate how to identify the convergence-related variables and diversity-related variables in LMEA for a bi-objective minimization problem with four decision variables, denoted as x_1, x_2, x_3 , and x_4 . In this example, x_1 and x_2 are identified as diversity-related variables, and x_3 and x_4 are identified as convergence-related variables in LMEA. (a) Objective values of sample solutions generated by the perturbations. (b) Angles between the fitted lines and the normal line of the hyperplane. (c) Clustering results on the four decision variables x_1, x_2, x_3 , and x_4 .

B. Decision Variable Clustering Method

Fig. 3 presents an example to illustrate the main idea of the proposed decision variable clustering method, where a bi-objective minimization problem with four decision variables x_1, x_2, x_3 , and x_4 is considered. To identify if the decision variables are convergence-related or diversity-related, a number of $nSel$ (two in this example) candidate solutions are first randomly selected from the population. Then, a number of $nPer$ (eight in this example) perturbations are performed on each of the four variables of the selected candidate solutions. Fig. 3(a) plots the objective values of sample solutions generated by the perturbations.

Afterward, the sample solutions generated by perturbing the values of each variable are normalized, and a line L is generated to fit each set of normalized sample solutions. With the normalized sample solutions and the fitted line L , the angle between each fitted line L and the normal line of hyperplane $f_1 + \dots + f_M = 1$ is calculated, where the normal line represents the direction of convergence and M is the number of objectives. In this way, each variable is associated with several angles, where the number is dependent on the number of selected candidate solutions for the decision variable clustering. Fig. 3(b) depicts the angles between the fitted lines and the normal line of the hyperplane. Since two candidate solutions are used for decision variable clustering in the example, each variable x_i , $1 \leq i \leq 4$, is associated with two angles, denoted as θ_{i1} and θ_{i2} as shown in Fig. 3(b). In the proposed decision variable clustering method, the angles associated with each variable are measurements that indicate its contributions to convergence and diversity. To be specific, a larger angle means that the associated variable contributes more to diversity, while a smaller angle indicates more contribution to convergence. The more solutions are used for decision variable clustering, the more angles are associated with each variable, thus providing more precise measurements.

Finally, the k -means clustering method is adopted to divide the decision variables into two clusters based on the features of each variable. As a consequence, the variables in the cluster having a smaller mean of angles are identified as convergence-related, while those in the other cluster are identified as diversity-related. Fig. 3(c) presents the clustering results on the four decision variables x_1, x_2, x_3 , and x_4 . As a consequence, x_1 and x_2 are identified as diversity-related variables

Algorithm 5: VariableClustering($P, nSel, nPer$)

Input: P (current population), $nSel$ (number of selected solutions for decision variable clustering), $nPer$ (number of perturbations on each solution for decision variable clustering)

Output: DV (diversity-related variables), CV (convergence-related variables)

```

1  $D \leftarrow$  the length of decision variables;
2 for  $i = 1$  to  $D$  do
3    $S \leftarrow$  randomly select  $nSel$  solutions from  $P$ ;
4   for  $j = 1$  to  $nSel$  do
5     Perturb the  $i$ -th decision variable of  $S[j]$  for  $nPer$ 
       times to generate a population  $SP$ ;
6     Normalize  $SP$ ;
7     fit a line  $L$  in the objective space for  $SP$ ;
8      $Angle[i][j] \leftarrow$  angle between  $L$  and the normal
       line of hyperplane  $f_1 + \dots + f_M = 1$ ,  $M$  is the
       number of objectives of MaOPs;
9      $MSE[i][j] \leftarrow$  mean square error of the fitting;
10  $CV \leftarrow \{i = 1, \dots, D | \text{mean}(MSE[i]) < 1e - 2\}$ ;
11  $[S1, S2] \leftarrow$  use  $k$ -means to cluster the decision variables
    into two sets based on  $Angle$ ;
12 if  $CV \cap S1 \neq \emptyset$  and  $CV \cap S2 \neq \emptyset$  then
13    $CV \leftarrow CV \cap S$ ,  $S$  is the one of  $S1$  and  $S2$  which has
     a smaller mean of  $Angle$ ;
14  $DV \leftarrow \{j = 1, \dots, D | j \notin CV\}$ ;

```

whereas x_3 and x_4 are identified as convergence-related variables, which is consistent with the observations in Fig. 3(a). Algorithm 5 presents the detailed procedure of the proposed decision variable clustering method.

It should be noted that, compared to the variable analysis method suggested in MOEA/DVA [64], the decision variable clustering method proposed in this paper is significantly different in the following two aspects.

- 1) The decision variable analysis method in MOEA/DVA divides the variables into three categories, namely, convergence-related variables, diversity-related variables, and variables related to both convergence and diversity. By contrast, the decision variable clustering

TABLE I
DECISION VARIABLE GROUPING RESULTS OBTAINED BY THE DECISION VARIABLE ANALYSIS METHOD IN MOEA/DVA
AND THE DECISION VARIABLE CLUSTERING METHOD IN LMEA ON DTLZ1–DTLZ7, WFG3, UF9, AND UF10

Problem	Obj.	MOEA/DVA			LMEA		Remark
		Diversity	Convergence	Both	Diversity	Convergence	
DTLZ1	5	$\{x_1, x_2, x_3, x_4\}$	$\{x_5, \dots, x_{16}\}$	\emptyset	$\{x_1, x_2, x_3, x_4\}$	$\{x_5, \dots, x_{16}\}$	Same results
	10	$\{x_1, \dots, x_9\}$	$\{x_{10}, \dots, x_{15}\}$	\emptyset	$\{x_1, \dots, x_9\}$	$\{x_{10}, \dots, x_{15}\}$	
DTLZ2	5	$\{x_1, x_2, x_3, x_4\}$	$\{x_5, \dots, x_{16}\}$	\emptyset	$\{x_1, x_2, x_3, x_4\}$	$\{x_5, \dots, x_{16}\}$	
	10	$\{x_1, \dots, x_9\}$	$\{x_{10}, \dots, x_{15}\}$	\emptyset	$\{x_1, \dots, x_9\}$	$\{x_{10}, \dots, x_{15}\}$	
DTLZ3	5	$\{x_1, x_2, x_3, x_4\}$	$\{x_5, \dots, x_{16}\}$	\emptyset	$\{x_1, x_2, x_3, x_4\}$	$\{x_5, \dots, x_{16}\}$	
	10	$\{x_1, \dots, x_9\}$	$\{x_{10}, \dots, x_{15}\}$	\emptyset	$\{x_1, \dots, x_9\}$	$\{x_{10}, \dots, x_{15}\}$	
DTLZ4	5	$\{x_1, x_2, x_3, x_4\}$	$\{x_5, \dots, x_{16}\}$	\emptyset	$\{x_1, x_2, x_3, x_4\}$	$\{x_5, \dots, x_{16}\}$	
	10	$\{x_1, \dots, x_9\}$	$\{x_{10}, \dots, x_{15}\}$	\emptyset	$\{x_1, \dots, x_9\}$	$\{x_{10}, \dots, x_{15}\}$	
DTLZ7	5	$\{x_1, x_2, x_3, x_4\}$	$\{x_5, \dots, x_{16}\}$	\emptyset	$\{x_1, x_2, x_3, x_4\}$	$\{x_5, \dots, x_{16}\}$	
	10	$\{x_1, \dots, x_9\}$	$\{x_{10}, \dots, x_{15}\}$	\emptyset	$\{x_1, \dots, x_9\}$	$\{x_{10}, \dots, x_{15}\}$	
DTLZ5	5	$\{x_1, x_2, x_3, x_4\}$	$\{x_7, x_9, x_{10}, x_{11}, x_{13}, x_{14}, x_{15}\}$	$\{x_5, x_6, x_8, x_{12}, x_{16}\}$	$\{x_1, x_2, x_3, x_4\}$	$\{x_5, \dots, x_{16}\}$	The variables related to both convergence and diversity are labeled as convergence related variables
	10	$\{x_1, \dots, x_9\}$	$\{x_{11}\}$	$\{x_{10}, x_{12}, \dots, x_{15}\}$	$\{x_1, \dots, x_9\}$	$\{x_{10}, \dots, x_{15}\}$	
DTLZ6	5	$\{x_1, x_2, x_3, x_4\}$	$\{x_5, \dots, x_{11}, x_{13}, x_{15}, x_{16}\}$	$\{x_{12}, x_{14}\}$	$\{x_1, x_2, x_3, x_4\}$	$\{x_5, \dots, x_{16}\}$	
	10	$\{x_1, \dots, x_9\}$	$\{x_{12}, x_{13}\}$	$\{x_{10}, x_{11}, x_{14}, x_{15}\}$	$\{x_1, \dots, x_9\}$	$\{x_{10}, \dots, x_{15}\}$	
WFG3	5	$\{x_1, x_2, x_3, x_4\}$	$\{x_8, \dots, x_{13}, x_{15}, x_{16}\}$	$\{x_5, x_6, x_7, x_{14}\}$	$\{x_1, x_2, x_3, x_4\}$	$\{x_5, \dots, x_{16}\}$	
	10	$\{x_1, \dots, x_9\}$	$\{x_{14}, x_{15}\}$	$\{x_{10}, x_{11}, x_{12}, x_{13}\}$	$\{x_1, \dots, x_9\}$	$\{x_{10}, \dots, x_{15}\}$	
UF9	3	\emptyset	$\{x_3, \dots, x_{16}\}$	$\{x_1, x_2\}$	$\{x_1, x_2\}$	$\{x_3, \dots, x_{16}\}$	Labeled as diversity related
UF10	3	\emptyset	$\{x_3, \dots, x_{16}\}$	$\{x_1, x_2\}$	$\{x_1, x_2\}$	$\{x_3, \dots, x_{16}\}$	

method proposed in this paper divides the variables into only two categories, namely, convergence-related variables and diversity-related variables. With the proposed strategy, on the one hand, the variables identified as related to both convergence and diversity in MOEA/DVA are identified as either convergence-related or diversity-related in LMEA; on the other hand, part of the variables identified as diversity-related in MOEA/DVA are also identified as convergence-related in LMEA, as illustrated in Fig. 2.

- 2) The decision variable analysis method in MOEA/DVA determines the category of each variable based on the dominance relationship. By contrast, the proposed decision variable clustering method adopts the k -means method to determine the category of each variable. Compared to the decision variable analysis method in MOEA/DVA, the proposed decision variable clustering method is more robust as its performance is independent of the dominance-based relationships which may probably be ineffective in many-objective optimization due to the issue of dominance resistance.

Table I presents the decision variable grouping results obtained by the decision variable analysis method in MOEA/DVA and the decision variable clustering method in LMEA on DTLZ1–DTLZ7, WFG3, UF9, and UF10, where 16 decision variables are considered for each five-objective DTLZ and WFG problem and all UF problems, and 15 decision variables for each ten-objective DTLZ and WFG problem. For the UF and WFG test suites, here we only list the grouping results on three representative test problems, i.e., UF9, UF10, and WFG3. The grouping results on the other WFG test problems are similar to those obtained on DTLZ1–DTLZ4

and DTLZ7, and the results obtained on all UF problems are the same. As is evidenced in Table I, the proposed decision variable clustering method is able to further divide the variables related to both convergence and diversity as labeled by the decision variable analysis method in MOEA/DVA into convergence-related variables and diversity-related variables, respectively. In Section IV, we will empirically illustrate that the proposed decision variable clustering method in LMEA helps improve the performance of MOEAs for solving large-scale MaOPs. It is worth noting that, in practice, a decision variable can be convergence-related in a region but diversity-related in another, and vice versa. In LMEA, the convergence and diversity property of a decision variable is identified based on the perturbation results on a number of $nSel$ solutions randomly selected from the initial population. The larger value of $nSel$ is used, the more regions will be likely to be sampled, but it does not fully guarantee the global consistency of the identification results.

C. Fast Tree-Based Nondominated Sorting Approach

As illustrated in Algorithms 3 and 4, nondominated sorting is adopted during the optimization of both convergence-related variables and diversity-related variables in LMEA. Therefore, the computational efficiency of LMEA is considerably influenced by nondominated sorting approach thus applied. In multiobjective optimization literature, there are a number of approaches that have been suggested for reducing the complexity of nondominated sorting [51]–[54], [71]. Most of these approaches, however, are not applicable to MaOPs because they hold an average complexity of $O(MN^2)$ (M is the number of objectives and N is the population size) if the majority

Algorithm 6: Main Steps of T-ENS

Input: P (population), M (number of objectives)
Output: F (set of fronts, each front is represented by a tree)

```

1 Sort  $P$  in an ascending order according to the first
  objective;
2  $F \leftarrow \emptyset$ ;
3  $k \leftarrow 0$ ;
4 while  $\text{not\_empty}(P)$  do
5    $k \leftarrow k + 1$ ; /*start to construct a new
     tree*/
6   for all the  $p \in P$  do
7      $\text{objSeq}[p] \leftarrow$  random sequence from 2 to  $M$ ;
8      $\text{update\_tree}(p, F[k], \text{objSeq})$ ;
9 return  $F$ ;

```

of candidate solutions in a population are nondominated with each other, which is very common in many-objective optimization though.

To reduce the computational cost of LMEA for solving large-scale MaOPs, we present a computational efficient tree-based nondominated sorting approach, termed T-ENS, on the basis of the ENS recently proposed in [54]. The proposed T-ENS holds a time complexity of $O(MN \log N / \log M)$ for populations where most candidate solutions are nondominated with each other. It is worth noting that, there are also some other tree-structure-based nondominated sorting approaches. For example, the sorting algorithm proposed in [72] stores the nondominated solutions in an M - d tree (M is the number of objectives), and adds new solution via inserting and deleting nodes from the tree; another sorting algorithm proposed in [73] uses a binary tree to store the dominance relationships between solutions, by adjusting the binary tree the solutions in each front can be identified; and the recently proposed M-front [53] keeps a K - d tree to perform approximate nearest neighbor search for determining whether a new solution is dominated by the existing nondominated solutions. Compared to these existing tree-structure based approaches, the proposed T-ENS holds a considerably less computational complexity as it benefits from the framework of ENS [54], which allows candidate solutions to be inserted into the tree one by one without adjusting the structure of it.

The main idea of T-ENS is to use a tree to represent the solutions in each nondominated front, where the information about the objectives determining the nondominance relationship between solutions are recorded by the position of nodes in the tree in which the solutions are stored. As a result, many nondominance relationships between solutions can be inferred from those that have been assigned to the front (stored in the tree), thereby considerably reducing the number of comparisons between solutions belonging to the same front. This leads to considerable enhancement of the computational efficiency of T-ENS over ENS, in particular for solving MaOPs. Algorithms 6–8 present the detailed steps of T-ENS in pseudo code.

Algorithm 7: $\text{update_tree}(p, \text{tree}, \text{objSeq})$

Input: p (the solution to be checked), tree (the tree to be checked), objSeq (the sequence of objectives for each solution)

Output: -

```

1 if  $\text{empty}(\text{tree})$  then
2    $\text{tree} \leftarrow p$ ; /* $p$  is used as the root of
   the tree*/
3 else if  $\text{check\_tree}(p, \text{tree}, \text{objSeq}, \text{true})$  then
4    $P \leftarrow P \setminus \{p\}$ ;
5 return;

```

To be specific, as done in ENS, T-ENS first sorts the solutions in the population according to an ascending order of the first objective for M -objective minimization problems, $M \geq 2$. Let us assume that the sorted population consists of the following N solutions: $p_1(f_1^1, f_2^1, \dots, f_M^1), p_2(f_1^2, f_2^2, \dots, f_M^2), \dots$, and $p_N(f_1^N, f_2^N, \dots, f_M^N)$, where f_j^i is the j th objective value of the i th solution, $1 \leq j \leq M$ and $1 \leq i \leq N$. Then, T-ENS starts to construct a tree for each nondominated front. T-ENS adopts the first solution p_1 as the root of the tree for the first nondominated front F_1 and all other solutions in the population belonging to F_1 will be stored as the descendants of p_1 . The position of solution p_i , $2 \leq i \leq N$, in the tree is determined by the minimum j ($j > 1$) satisfying that $f_{\text{objSeq}[1][j]}^i < f_{\text{objSeq}[1][j]}^1$, where $\text{objSeq}[1][j]$ is the j th element of a sequence that is randomly generated from 2 to M for solution p_1 . More precisely, a solution p_i with the $\text{objSeq}[1][j]$ th objective satisfying the above condition will be stored as the j th child of the root. If there is more than one solution that satisfies the above condition, the next solution will be stored as a child of p_i , i.e., a grandchild of p_1 . This procedure repeats until all solutions in the population are checked. After the construction of tree for the first nondominated front F_1 is complete, T-ENS starts to construct a tree for the second front F_2 until all solutions in the population are assigned to a tree.

With the above tree constructed to represent solutions in a front, if a solution p to be assigned to the front is nondominated with a solution q in the tree, where p has a smaller value than q on the $\text{objSeq}[q][j]$ th objective (p has a larger value than q on the first objective since q is ranked before p in the sorted population), then the nondominance relationship between p and the solution stored as the k th ($k > j$) child of node q and all descendants of this child can be inferred from the nondominance relationship between q and p , since for each of these solutions s we have $f_{\text{objSeq}[q][j]}^p < f_{\text{objSeq}[q][j]}^q < f_{\text{objSeq}[q][j]}^s$, $j \in \{1, 2, \dots, M-1\}$. Therefore, p does not need to be compared with these solutions in determining the front of p , which enables T-ENS work very efficiently for MaOPs.

To investigate the computational efficiency of the proposed T-ENS, we have conducted theoretical time complexity analysis and empirical evaluation which can be found in the supplementary materials. Compared to most existing nondominated approaches, which hold a time complexity of $O(MN^2)$ on nondominated populations, T-ENS is computationally more

Algorithm 8: *check_tree(p, tree, objSeq, add_pos)*

Input: p (the solution to be checked), $tree$ (the tree to be checked), $objSeq$ (the sequence of objectives for each solution), add_pos (indicates whether p can be added to a node of $tree$ as a ray)

Output: nd (indicates whether p is non-dominated with all solutions that have been assigned to the $tree$)

```

1 if empty(tree) then
2   return true; /*tree is an empty tree*/
3 Find the minimal index  $m$  satisfying that
   $p[objSeq[tree.root][m]] < tree.root[objSeq[tree.root][m]]$ ;
4 if  $m$  not found then
5   return false; /* $p$  is dominated by the
  solution at the root*/
6 else
7   for  $i \leftarrow 1$  to  $m$  do
8     if check_tree( $p$ , tree.branch[ $i$ ], objSeq,
7      $i == m \ \&\& \ add\_pos == false$  then
9       return false; /* $p$  is dominated by a
7       solution in the branch of the
7       tree*/
10  if empty(tree.branch[ $m$ ]) && add_pos then
11    tree.branch[ $m$ ] =  $p$ ; /*add  $p$  to the
11    branch of the tree*/
12  return true;
```

efficient, holding a time complexity of $O(MN \log N / \log M)$, which enables the proposed LMEA to work faster for solving large-scale MaOPs.

IV. SIMULATION EXPERIMENTS AND ANALYSIS

In this section, we verify the performance of LMEA in solving large-scale MaOPs by empirically comparing it with four state-of-the-art MOEAs, namely, MOEA/D [30], NSGA-III [37], KnEA [22], and MOEA/DVA [64]. The MOEA/D, NSGA-III, and KnEA are three well-known MOEAs for solving MaOPs, and MOEA/DVA is the first decision variable analysis based MOEA recently proposed in [64] to solve large-scale MOPs.

The experiments are conducted on ten test problems shown in Table I, which are taken from three widely used test suites, DTLZ [74], WFG [75], and UF [76]. These test problems include DTLZ1–DTLZ7, WFG3, UF9 and UF10, which can be categorized into three groups according to the decision variable grouping results in Table I. The first group contains DTLZ1–DTLZ4 and DTLZ7, on which the proposed LMEA and MOEA/DVA have the same grouping results. The second group consists of DTLZ5, DTLZ6, and WFG3, on which the variables related to both convergence and diversity in MOEA/DVA are identified as convergence-related variables in LMEA. The third group contains UF9 and UF10, on which the variables related to both convergence and diversity in MOEA/DVA are identified as diversity-related variables in LMEA. Due to the limitation of pages, we have not

TABLE II
SETTING OF POPULATION SIZE IN NSGA-III AND MOEA/D, WHERE p_1 AND p_2 ARE PARAMETERS CONTROLLING THE NUMBERS OF REFERENCE POINTS ALONG THE BOUNDARY OF THE PARETO FRONT AND INSIDE IT, RESPECTIVELY

Number of objectives	Parameter (p_1, p_2)	Population size
3	(13, 0)	105
5	(5, 0)	126
10	(3, 2)	275

presented the experimental results on other WFG and UF test problems. Further experiments indicate that all WFG test problems except WFG3 belong to the first group, and all UF test problems belong to the third group. Regarding the number of decision variables, 100, 500, and 1000 decision variables are considered for each test problem, respectively. Regarding the number of objectives, for each DTLZ and WFG test problem, five and ten objectives are considered, respectively, while both UF9 and UF10 are three-objective. Moreover, the proposed LMEA is further challenged on the test problems with 2000 and 5000 decision variables. In this case, all the test problems used in our simulation experiments are large-scale MaOPs, except UF9 and UF10, which are large-scale MOPs.

A. Experimental Setting

For fair comparisons, all compared algorithms adopt the recommended parameter values to achieve the best performance. To be specific, the parameter settings for all conducted experiments are as follows.

1) *Crossover and Mutation*: All compared algorithms adopt the SBX [67] to create offspring for DTLZ and WFG test problems, and DE [69], [70], [77] for UF test problems. For each test problem, the polynomial mutation [68] is also performed in all algorithms. The distribution indexes of crossover and mutation are all set to 20. The crossover probability and mutation probability are set to $p_c = 1.0$ and $p_m = 1/D$, respectively, where D denotes the number of decision variables. The control parameters F and CR in DE are set to $F = 0.5$ and $CR = 1.0$, as recommended in [77].

2) *Population Sizing*: The population size of NSGA-III and MOEA/D cannot be arbitrarily specified, which is equal to the number of uniformly distributed reference points (or weight vectors) generated with the simplex lattice design. As recommended in [37]–[39], the two-layer reference point generation strategy is adopted here. Table II presents the setting of population size in NSGA-III and MOEA/D, where p_1 and p_2 are parameters controlling the numbers of reference points along the outer and inner layers of the Pareto front, respectively. To be consistent with NSGA-III and MOEA/D, KnEA, MOEA/DVA, and LMEA adopt the same population size settings with respect to the number of objectives.

3) *Stopping Condition and Number of Runs*: The number of evaluations is used as the termination criterion for all considered algorithms. For each test instance with 100, 500, 1000, 2000, and 5000 decision variables, the maximum number of evaluations is set to 1 000 000, 6 800 000,

TABLE III
IGD METRIC VALUES OF THE FIVE ALGORITHMS ON DTLZ1–DTLZ4 AND DTLZ7, WHERE THE
BEST RESULT ON EACH TEST INSTANCE IS SHOWN IN A GRAY BACKGROUND

Problem	Obj.	Dec.	MOEA/D	NSGA-III	KnEA	MOEA/DVA	LMEA
DTLZ1	5	100	1.1859e-1(1.63e-3)–	3.9340e+0(1.72e+0)–	6.6202e+0(1.83e+0)–	6.2932e-2(5.72e-5)–	5.9982e-2(4.22e-4)
		500	1.4648e-1(4.30e-2)–	5.2597e+1(5.63e+0)–	7.6436e+1(6.86e+0)–	6.3284e-2(1.63e-4)–	6.1124e-2(5.16e-4)
		1000	1.6833e-1(4.29e-2)–	1.1669e+2(5.08e+0)–	1.3392e+2(1.28e+1)–	6.3442e-2(1.26e-4)–	6.0423e-2(4.09e-4)
	10	100	2.2187e+0(4.72e+0)–	1.1944e+2(2.72e+1)–	5.5749e+0(1.53e+0)–	1.4356e-1(1.79e-2)≈	1.6302e-1(4.87e-3)
		500	1.6987e+2(1.01e+2)–	3.9829e+2(2.77e+1)–	5.0951e+1(1.25e+1)–	1.7047e-1(1.14e-2)≈	1.5995e-1(4.05e-3)
		1000	4.8922e+2(2.04e+2)–	7.9841e+2(4.16e+1)–	1.4841e+2(1.93e+1)–	1.3805e-1(2.18e-2)≈	1.6002e-1(5.24e-3)
DTLZ2	5	100	3.2006e-1(2.26e-8)–	1.9494e-1(7.98e-7)–	2.2045e-1(1.18e-2)–	1.9493e-1(9.26e-8)–	1.8825e-1(2.14e-3)
		500	3.2006e-1(5.31e-8)–	1.9494e-1(5.62e-8)–	2.3629e-1(7.63e-3)–	1.9494e-1(3.17e-8)–	1.8832e-1(2.29e-3)
		1000	3.2006e-1(1.12e-7)–	1.9494e-1(9.86e-7)–	2.2844e-1(1.00e-2)–	1.9494e-1(6.25e-8)–	1.8816e-1(2.53e-3)
	10	100	7.1528e-1(1.81e-2)–	4.2141e-1(1.58e-4)+	4.1230e-1(2.24e-3)+	4.5923e-1(5.13e-2)≈	5.0905e-1(1.48e-2)
		500	7.2369e-1(8.63e-3)–	4.2152e-1(4.79e-5)+	4.1402e-1(5.07e-3)+	4.1976e-1(3.53e-4)+	5.0617e-1(1.78e-2)
		1000	7.3217e-1(1.90e-2)–	4.2172e-1(5.76e-5)+	4.1222e-1(1.85e-3)+	4.2065e-1(1.19e-4)+	5.0689e-1(2.84e-2)
DTLZ3	5	100	3.2877e-1(2.49e-3)–	2.2868e+1(6.14e+0)–	7.6962e-1(4.18e-1)–	1.9505e-1(5.92e-5)–	1.8985e-1(2.14e-3)
		500	4.1776e-1(1.97e-1)–	2.9421e+2(3.34e+1)–	6.1228e-1(1.26e-1)–	1.9536e-1(1.01e-4)≈	1.9035e-1(4.44e-3)
		1000	4.4681e-1(1.40e-2)–	6.8229e+2(5.50e+1)–	8.2477e-1(2.42e-1)–	1.9563e-1(3.70e-4)≈	1.8812e-1(4.10e-3)
	10	100	8.0019e-1(4.87e-2)–	8.2193e+2(1.37e+2)–	6.3652e+0(3.88e+0)–	5.0747e-1(3.77e-2)≈	5.5352e-1(3.56e-2)
		500	1.2941e+2(2.78e+2)–	2.2891e+3(1.35e+2)–	9.4498e+0(4.57e+0)–	5.2820e-1(9.88e-2)≈	5.5126e-1(1.66e-2)
		1000	2.5643e+2(3.27e+2)–	4.4071e+3(1.91e+2)–	4.4621e+0(2.34e+0)–	4.7728e-1(4.26e-2)+	5.4964e-1(1.85e-2)
DTLZ4	5	100	6.2588e-1(2.50e-1)–	2.7298e-1(1.31e-1)+	2.1434e-1(4.33e-3)+	2.6957e-1(1.29e-1)≈	2.6411e-1(1.55e-2)
		500	5.2727e-1(1.18e-1)–	1.9496e-1(2.64e-5)+	2.1571e-1(9.97e-3)+	3.4421e-1(1.29e-1)≈	2.7256e-1(2.46e-2)
		1000	4.3848e-1(1.80e-1)–	2.2829e-1(1.09e-1)≈	2.1380e-1(4.21e-3)+	3.4420e-1(1.29e-1)≈	2.7071e-1(2.36e-2)
	10	100	8.3550e-1(3.18e-2)–	4.2123e-1(1.95e-4)+	4.2764e-1(2.53e-2)+	4.3772e-1(3.33e-2)+	5.0820e-1(2.47e-2)
		500	8.3052e-1(2.93e-2)–	4.2154e-1(1.02e-4)+	4.0104e-1(4.57e-3)+	4.1970e-1(5.83e-5)+	5.2786e-1(3.98e-3)
		1000	8.2082e-1(2.40e-2)–	4.2173e-1(7.92e-5)+	4.0081e-1(2.61e-3)+	4.5695e-1(3.18e-2)+	5.2345e-1(9.26e-3)
DTLZ7	5	100	5.2987e-1(2.57e-2)–	5.2849e-1(1.56e-1)–	2.4790e-1(1.14e-2)+	5.2044e-1(2.51e-6)–	3.0913e-1(1.10e-2)
		500	5.1542e-1(3.73e-7)–	2.3928e+0(1.60e-1)–	2.3191e-1(7.58e-3)+	5.2043e-1(4.57e-7)–	3.2032e-1(8.53e-3)
		1000	5.2120e-1(1.78e-2)–	2.6633e+0(1.38e-1)–	2.3408e-1(1.44e-2)+	5.2043e-1(7.54e-7)–	3.1051e-1(7.59e-3)
	10	100	4.4020e+0(1.37e+0)–	4.6684e+0(4.21e-1)–	1.3854e+0(3.48e-2)–	1.6385e+0(7.84e-2)–	1.0749e+0(6.40e-3)
		500	5.3581e+0(6.94e-1)–	1.1875e+1(8.36e-1)–	1.3572e+0(1.65e-2)–	1.6320e+0(1.02e-1)–	1.0752e+0(3.29e-3)
		1000	5.7150e+0(2.69e-1)–	1.5027e+1(9.39e-1)–	1.3300e+0(9.50e-3)–	1.5249e+0(4.16e-2)–	1.0781e+0(4.58e-3)
+ / – / ≈			0/30/0	8/21/1	12/18/0	6/13/11	

'+', '-' and '≈' indicate that the result is significantly better, significantly worse and statistically similar to that of LMEA, respectively.

17 000 000, 50 000 000, and 230 000 000, respectively. On each test instance, 20 independent runs are performed for each algorithm to obtain statistical results.

4) *Other Parameters*: The range of neighborhood, the maximal number of solutions replaced by each child solution and the probability that parent solutions are selected from the neighborhood in MOEA/D (as well as MOEA/DVA) are set to $T = 0.1N$, $n_r = 0.01N$, and $\delta = 0.9$ for all test problems, respectively, where N is the population size. The Tchebycheff approach is employed as the aggregation function in MOEA/D. For KnEA, the parameter T for controlling the ratio of knee points is set to 0.3 for five-objective DTLZ1 and DTLZ3, 0.4 for five-objective DTLZ5 and DTLZ6, 0.5 for five-objective DTLZ2, DTLZ4 and DTLZ7, as well as UF9 and UF10; T is set to the recommended values for ten-objective DTLZ and WFG test problems [22]. For MOEA/DVA, the number of interaction analysis and the number of control property

analysis are set to the recommended values, namely, $NIA = 6$ and $NCA = 50$. For LMEA, the number of selected solutions and the number of perturbations for each selected solution in decision variable clustering are set to $nSel = 2$ and $nPer = 4$, respectively, and the number of selected solutions in decision variable interaction analysis is set to $nCor = 6$. A detailed sensitivity analysis of LMEA to parameters $nSel$ and $nPer$ can be found in the supplementary materials.

4) *Performance Metric*: Two widely used performance indicators, the inverted generational distance (IGD) [78] and the HV [79], [80] are adopted to evaluate the performance of the compared algorithms. As recommended in [37] and [39], the reference point sets required for IGD calculations are generated by sampling uniformly distributed points using the Das and Dennis method [81], and a set size closest to 5000 is used in this paper for all test instances. For the calculation of HV, we first discard the solutions that are outside

TABLE IV
IGD METRIC VALUES OF THE FIVE ALGORITHMS ON DTLZ5, DTLZ6, WFG3, UF9, AND UF10,
WHERE THE BEST RESULT ON EACH TEST INSTANCE IS SHOWN IN A GRAY BACKGROUND

Problem	Obj.	Dec.	MOEA/D	NSGA-III	KnEA	MOEA/DVA	LMEA
DTLZ5	5	100	4.5161e-2(9.31e-7)–	1.4957e-1(2.58e-2)–	2.7185e-1(4.11e-2)–	2.0440e-1(5.06e-4)–	4.1162e-3(1.44e-4)
		500	4.5161e-2(1.04e-6)–	1.9413e-1(1.78e-2)–	3.1740e-1(6.11e-2)–	2.0469e-1(5.20e-8)–	4.0861e-3(1.48e-4)
		1000	4.5162e-2(3.32e-7)–	2.0606e-1(1.10e-2)–	3.8913e-1(6.77e-2)–	2.0461e-1(1.36e-4)–	4.0729e-3(9.90e-5)
	10	100	4.9994e-2(2.41e-4)–	3.1946e-1(2.03e-2)–	3.6432e-1(5.71e-2)–	1.8877e-1(1.87e-4)–	2.3954e-3(6.95e-5)
		500	5.0407e-2(4.16e-4)–	5.2642e-1(2.04e-2)–	3.6389e-1(5.43e-2)–	1.8866e-1(3.30e-4)–	2.2721e-3(4.47e-5)
		1000	5.0759e-2(2.12e-5)–	6.2093e-1(1.14e-2)–	4.1806e-1(5.07e-2)–	1.8880e-1(2.03e-4)–	2.0713e-3(6.98e-5)
DTLZ6	5	100	1.4970e-1(3.14e-2)–	2.5642e-1(2.29e-2)–	5.8811e-1(1.34e-1)–	1.8236e-1(2.43e-6)–	3.9943e-3(2.14e-4)
		500	1.3010e+0(1.04e-1)–	4.9939e-1(1.89e-2)–	7.2754e-1(1.43e-1)–	1.8236e-1(4.25e-7)–	4.5127e-3(1.22e-3)
		1000	2.7140e+0(1.97e-1)–	6.5774e-1(2.19e-2)–	1.5085e+0(4.53e-1)–	1.8236e-1(5.91e-7)–	3.9747e-3(2.29e-4)
	10	100	6.7510e-2(1.85e-2)–	7.2120e+0(1.35e+0)–	3.7560e+0(9.56e-1)–	1.6531e-1(4.09e-2)–	2.4477e-3(5.11e-4)
		500	1.1735e+0(2.52e-1)–	8.7171e+1(4.88e+0)–	6.3085e+0(2.18e+0)–	1.2750e-1(5.33e-2)–	3.0711e-3(7.20e-4)
		1000	2.6191e+0(5.73e-1)–	1.9202e+2(9.83e+0)–	4.8989e+0(2.54e+0)–	1.1844e-1(2.26e-2)–	3.7077e-3(1.66e-3)
WFG3	5	100	2.0705e+0(8.91e-2)–	7.7137e-1(3.83e-2)–	6.2696e-1(4.24e-1)–	2.3769e+0(7.55e-3)–	1.2581e-1(2.91e-2)
		500	2.2789e+0(6.03e-2)–	8.6982e-1(2.37e-2)–	2.2284e-1(2.56e-2)–	2.4699e+0(9.15e-3)–	1.1736e-1(3.58e-2)
		1000	2.3370e+0(8.10e-2)–	8.8753e-1(2.53e-2)–	4.6414e-1(1.49e-1)–	2.4410e+0(3.52e-2)–	1.2493e-1(2.41e-2)
	10	100	3.4569e+0(1.29e-1)–	3.0344e+0(5.71e-2)–	2.2907e+0(7.93e-1)–	3.4846e+0(2.45e-2)–	1.8542e-1(5.96e-2)
		500	3.8106e+0(8.24e-2)–	3.1112e+0(5.04e-2)–	1.6148e+0(5.53e-1)–	3.5264e+0(9.75e-2)–	4.8685e-1(5.49e-2)
		1000	3.9456e+0(7.23e-2)–	3.1454e+0(4.20e-2)–	1.9861e+0(1.27e+0)–	3.5070e+0(1.17e-1)–	6.9330e-1(1.16e-1)
UF9	3	100	2.9851e-1(1.58e-2)–	2.2030e-1(9.19e-2)–	5.3546e-1(1.39e-1)–	4.3517e-2(2.50e-6)+	5.7008e-2(8.91e-3)
		500	3.1975e-1(2.92e-2)–	3.1029e-1(7.27e-2)–	4.6017e-1(1.19e-1)–	4.3516e-2(9.76e-7)+	5.3626e-2(6.94e-3)
		1000	3.0557e-1(8.39e-2)–	3.7850e-1(4.21e-2)–	5.3607e-1(8.03e-2)–	4.3516e-2(7.00e-7)+	5.1231e-2(4.50e-3)
UF10	3	100	5.9354e-1(1.50e-1)–	3.3482e-1(8.13e-2)–	7.5510e-1(1.49e-1)–	1.1024e-1(2.92e-3)+	1.6632e-1(1.45e-2)
		500	6.3119e-1(1.92e-1)–	3.6779e-1(8.36e-2)–	1.3142e+0(8.69e-1)–	1.0158e-1(8.55e-4)+	1.5547e-1(4.99e-3)
		1000	5.6232e-1(2.48e-1)–	4.2148e-1(1.10e-1)–	9.1794e-1(1.35e-1)–	1.0277e-1(1.01e-3)+	1.6924e-1(9.48e-3)
+ / – / \approx			0/24/0	0/24/0	0/24/0	6/18/0	

'+', '–' and ' \approx ' indicate that the result is significantly better, significantly worse and statistically similar to that of LMEA, respectively.

the box bounded by the ideal point and nadir point of the true Pareto front, and then normalize the remaining solutions inside. Since the same reference point $(1, 1, \dots, 1)$ is used for all test instances, it is guaranteed that all the obtained HV values will be inside the interval $[0, 1]$. Besides, we adopt the exact calculation for test problems with three objectives and the Monte Carlo estimation method when the number of objective is larger than 3. Note that both IGD and HV can account for the convergence as well as the distribution of a solution set. A smaller value of IGD indicates a better quality of the obtained solution set, while a larger value of HV signals a better quality.

B. Performance Comparison Between LMEA and Existing MOEAs on Large-Scale MaOPs

Tables III and IV present the statistical results of the IGD metric values of the five compared algorithms on the 10 test problems with 100, 500, and 1000 decision variables obtained via 20 independent runs, where the best result on each test instance is shown in a gray background. The Wilcoxon rank sum test is also adopted at a significance level of 0.05, where the symbols “+,” “–,” and “ \approx ” indicate that the result is

significantly better, significantly worse and statistically similar to that obtained by LMEA, respectively. From the tables, the following three observations can be made. First, the MOEA/D, NSGA-III, and KnEA can achieve a promising performance on some of the large-scale test instances under consideration, but their general performance will be worse than that of MOEA/DVA and the proposed LMEA that are specifically designed for solving large-scale problems. As can be seen from the tables, the IGD values obtained by MOEA/DVA and LMEA on each test problem are consistently good as the number of decision variables increases from 100 to 1000, which confirms a promising scalability of MOEA/DVA and LMEA.

Second, on the group of test problems DTLZ1–DTLZ4 and DTLZ7 where MOEA/DVA and LMEA obtain the same decision variable grouping results, the proposed LMEA has achieved a similar performance as that of MOEA/DVA with the only exception of DTLZ7. On all the DTLZ7 test instances, the proposed LMEA always obtains a smaller IGD value than MOEA/DVA. This is attributed to the fact that MOEA/DVA adopts MOEA/D for optimizing the diversity-related variables, while such decomposition-based MOEAs are not well suited for solving MOPs or MaOPs with irregular (disconnected or degenerate) Pareto optimal fronts [37], [39], [82].

TABLE V
RUNTIME (S) OF THE FIVE ALGORITHMS AVERAGING OVER ALL TEST INSTANCES WITH 100, 500, AND 1000 VARIABLES

Number of variables	MOEA/D	NSGA-III	KnEA	MOEA/DVA			LMEA		
				Control property analysis	Interaction analysis	Search	Variable clustering	Interaction analysis	Search
100	1.0499e+2	9.2768e+1	2.2953e+2	2.0693e-1	2.0669e+1	1.4598e+2	1.4638e-1	1.4218e+1	3.7283e+1
500	7.4233e+2	2.5760e+3	1.6144e+3	1.2292e+0	5.3845e+2	7.6295e+2	7.9235e-1	4.0784e+2	1.7208e+2
1000	1.8008e+3	7.4880e+3	4.2560e+3	2.9508e+0	2.3653e+3	1.2397e+3	1.4735e+0	1.9820e+3	4.0282e+2

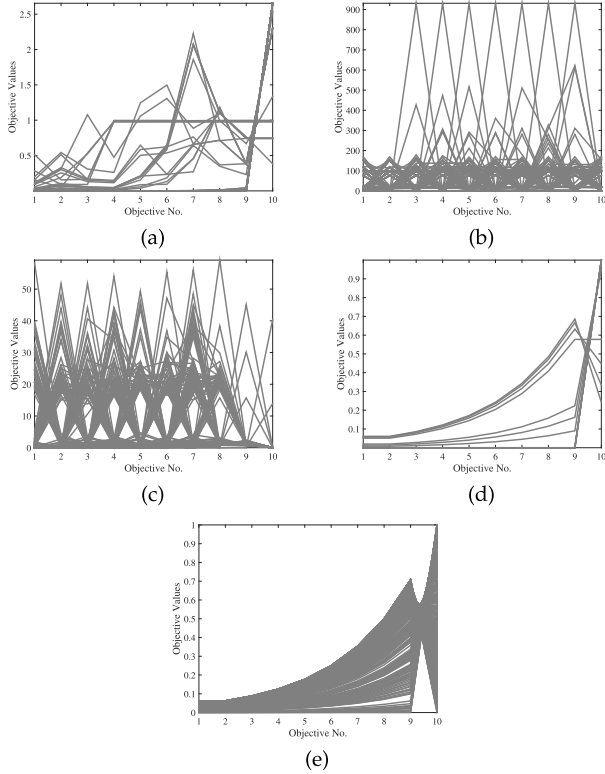


Fig. 4. Nondominated solutions with the lowest IGD metric values obtained by each algorithm among 20 runs in the objective space on ten-objective DTLZ6 problem with 1000 decision variables. (a) MOEA/D on DTLZ6. (b) NSGA-III on DTLZ6. (c) KnEA on DTLZ6. (d) MOEA/DVA on DTLZ6. (e) LMEA on DTLZ6.

By contrast, the distribution management strategy as detailed in Algorithm 4 is capable of handling such irregular Pareto optimal fronts as that of DTLZ7.

Finally, on the group of test problems DTLZ5, DTLZ6, WFG3, UF9, and UF10 where MOEA/DVA and LMEA obtain different decision variable grouping results, LMEA shows generally better performance than MOEA/DVA in terms of IGD. As can be seen from Table IV, on DTLZ5, DTLZ6, and WFG3, the variables related to both convergence and diversity in MOEA/DVA are identified as convergence-related variables in LMEA, thus resulting in significantly smaller IGD values. This observation confirms that the proposed variable clustering strategy in LMEA is able to reasonably identify the decision variables related to both convergence and diversity, which cannot be distinguished by MOEA/DVA.

It is worth noting that, MOEA/DVA and LMEA still achieve similar performance on UF9 and UF10, although different

TABLE VI
HV METRIC VALUES OF LMEA ON DTLZ1–DTLZ7, WFG3, UF9, AND UF10 WITH 2000 AND 5000 DECISION VARIABLES, AVERAGING OVER 20 RUNS

Problem	Obj.	2000-variable	5000-variable
DTLZ1	5	9.3601e-1(6.41e-3)	9.3782e-1(4.81e-3)
	10	8.9433e-1(1.38e-2)	8.7102e-1(1.31e-2)
DTLZ2	5	6.1363e-1(5.08e-3)	6.1221e-1(4.75e-3)
	10	6.4035e-1(2.19e-2)	6.5006e-1(1.41e-2)
DTLZ3	5	6.1373e-1(4.85e-3)	6.1270e-1(1.33e-2)
	10	3.0131e-1(4.37e-2)	1.5543e-1(2.88e-2)
DTLZ4	5	5.8063e-1(4.39e-2)	5.9056e-1(2.04e-2)
	10	7.3539e-1(3.52e-2)	6.6224e-1(3.64e-2)
DTLZ5	5	3.4370e-2(2.75e-4)	3.4090e-2(3.37e-4)
	10	8.3512e-3(2.66e-4)	7.8917e-3(5.19e-4)
DTLZ6	5	3.3410e-2(1.22e-3)	3.4350e-2(1.10e-4)
	10	6.6238e-3(1.49e-3)	6.2831e-3(1.15e-3)
DTLZ7	5	1.2758e-1(4.62e-3)	1.3263e-1(2.81e-3)
	10	5.5823e-2(4.15e-3)	5.7065e-2(7.94e-3)
WFG3	5	5.8731e-1(2.03e-3)	5.8971e-1(3.06e-3)
UF9	3	7.0473e-1(1.29e-2)	7.1139e-1(1.05e-2)
UF10	3	1.7095e-1(7.70e-3)	1.8247e-1(1.40e-2)

decision variable grouping results have been obtained by both algorithms. This is due to the fact that MOEA/DVA optimizes all variables related to both convergence and diversity, e.g., x_1 and x_2 in UF9 and UF10, in the same way as diversity-related variables. As shown in Table I, the variables x_1 and x_2 in UF9 and UF10 are identified as diversity-related variables in LMEA. The slightly better performance of MOEA/DVA than LMEA on UF9 and UF10 is due to the fact that MOEA/DVA adopts a uniform sampling method developed in [83] to initialize the population of diversity-related variables, which is well suited for maintaining the distribution of the candidate solutions when applied to the UF problems.

For further observations, Fig. 4 plots the nondominated solutions with the best IGD metric values obtained by the five algorithms among 20 runs in the objective space on ten-objective DTLZ6 with 1000 decision variables, respectively. It can clearly be seen that the proposed LMEA and MOEA/DVA outperform MOEA/D, NSGA-III, and KnEA on ten-objective DTLZ6 with 1000 decision variables in terms of both convergence and diversity, and the LMEA achieves much more nondominated solutions in the Pareto optimal front than MOEA/DVA, which confirms the effectiveness of decision variable clustering method developed in LMEA.

TABLE VII
IGD METRIC VALUES OF THE FIVE ALGORITHMS ON LSMOP1–LSMOP9, WHERE THE
BEST RESULT ON EACH TEST INSTANCE IS SHOWN IN A GRAY BACKGROUND

Problem	MOEA/D	NSGA-III	KnEA	MOEA/DVA	LMEA
LSMOP1	3.6215e-1(2.77e-2)–	2.0411e-1(3.05e-3)–	6.5295e-1(3.88e-1)–	1.7219e-1(7.47e-3)–	1.5151e-1(9.99e-3)
LSMOP2	2.4541e-1(8.98e-4)–	1.4727e-1(1.67e-3)–	2.3724e-1(7.06e-2)–	1.4212e-1(2.16e-3)–	1.2644e-1(1.45e-3)
LSMOP3	7.0884e-1(4.00e-2)–	4.4176e-1(1.30e-1)≈	7.0241e-1(9.37e-2)–	6.9024e-1(4.55e-2)–	4.1242e-1(4.68e-2)
LSMOP4	2.7326e-1(4.55e-3)–	1.8222e-1(9.13e-3)–	6.0172e-1(1.44e-1)–	1.5548e-1(4.40e-3)≈	1.5585e-1(2.04e-3)
LSMOP5	5.7900e-1(6.32e-2)–	3.2983e-1(1.23e-1)–	1.1584e+0(4.06e-1)–	3.8667e-1(3.92e-2)–	2.6932e-1(1.65e-2)
LSMOP6	1.2119e+0(2.44e-1)+	1.1094e+0(1.22e-1)+	1.8486e+0(1.39e+0)+	2.0992e+0(1.79e-1)+	1.3820e+0(3.91e+3)
LSMOP7	9.9083e-1(1.53e-1)+	1.0033e+0(2.11e-1)+	1.2275e+1(9.00e+0)–	9.1631e-1(2.81e-2)+	1.3542e+0(3.00e-1)
LSMOP8	5.1604e-1(1.67e-2)–	2.9337e-1(1.02e-2)–	4.3832e-1(5.81e-3)–	3.0537e-1(6.19e-2)–	2.3022e-1(7.57e-3)
LSMOP9	1.0600e+0(2.31e-1)–	2.3568e+0(8.52e-2)–	1.3535e+0(6.62e-1)≈	6.5805e-1(1.13e-1)≈	6.1163e-1(3.55e-2)
+ / – / ≈	2 / 7 / 0	2 / 6 / 1	1 / 7 / 1	2 / 5 / 2	

‘+’, ‘–’ and ‘≈’ indicate that the result is significantly better, significantly worse and statistically similar to that of LMEA, respectively.

To verify the computational efficiency of LMEA, Table V summarizes the runtime(s) of the five compared algorithms averaged over all the runs on test instances with 100, 500, and 1000 variables, respectively. Note that the proposed T-ENS is employed as the nondominated approach in all compared algorithms except MOEA/D. The runtime of LMEA and MOEA/DVA consists of three parts, namely, runtime of decision variable clustering (known as control property analysis in MOEA/DVA), runtime of decision variable interaction analysis and runtime of search stage. It can be observed from the table that LMEA is computationally more efficient than MOEA/D, NSGA-III, KnEA, and MOEA/DVA on test problems with 100 and 500 variables, while MOEA/D performs the best on problems with 1000 variables. For LMEA, the variable clustering component is less time-consuming than the other two components, namely decision variable interaction analysis and the search stage. In addition, on the one hand, the runtime of search stage in LMEA is almost linear scaling to the number of decision variables; on the other hand, however, the runtime of variable interaction analysis rapidly increases with the number of decision variables, which is due to the fact that this component holds a time complexity of $O(D^2)$ in the worst case, where D denotes the number of variables.

Based on the above empirical results, we can conclude that the proposed LMEA is more effective and efficient to handle large-scale MaOPs than state-of-the-art MOEAs designed for solving MaOPs or large-scale MOPs.

C. Performance of LMEA on Large-Scale MaOPs With 2000 and 5000 Decision Variables

In the previous section, we have demonstrated the competitive performance of the proposed LMEA on benchmark MaOPs with the number of decision variables ranged from 100 to 1000. In this section, we further challenge the proposed LMEA on MaOPs with larger scales.

Table VI presents the HV metric values obtained by the proposed LMEA on DTLZ1–DTLZ7, WFG3, UF9, and UF10 with 2000 and 5000 decision variables, via 20 runs, showing

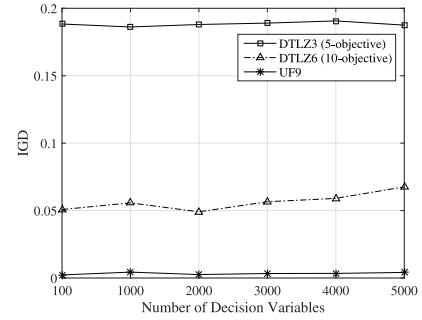


Fig. 5. IGD metric values of LMEA on five-objective DTLZ3, and ten-objective DTLZ6 and UF9 with different numbers of decision variables, averaging over 20 runs.

that the proposed LMEA obtained a similar HV metric value on each test problem with both 2000 and 5000 variables. For further observations, Fig. 5 plots the IGD metric values of LMEA on five-objective DTLZ3, ten-objective DTLZ6 and UF9 with different number of decision variables averaged over 20 runs. As evidenced in Fig. 5, the IGD metric values obtained by LMEA shows little deterioration on the three test problems as the number of decision variables increases from 100 to 5000, which indicates a promising scalability of the algorithm.

D. Performance Comparisons Between LMEA and Existing MOEAs on LSMOP Test Suit

To further assess the performance of LMEA on more challenging large-scale MaOPs, more empirical experiments are conducted on a recently developed test suite for large-scale multiobjective and many-objective optimization, known as the LSMOP test suite [84]. The LSMOP test suite consists of nine test problems (i.e., LSMOP1–LSMOP9), which are constructed based on six widely used large-scale SOPs, namely, sphere function, Schwefel’s problem, Rosenbrock’s function, Rastrigin’s function, Griewank’s function, and Ackley’s function. This class of test problems have the following main characteristics: 1) complicated variable linkages on the Pareto sets;

2) mixed separabilities between decision variables; 3) nonuniform nonseparable variable groups; and 4) nonuniform correlations between decision variables and objective functions, some of which are borrowed from the literature of large-scale single-objective optimization. Hence, LSMOP test problems can better reflect the challenges in large-scale MaOPs than existing test suits.

Table VII presents the statistical results of the IGD values obtained by the five algorithms on LSMOP1–LSMOP9 with five objectives and 500 decision variables via 20 independent runs, where the parameter T for controlling the ratio of knee points in KnEA is set to 0.5, and all the other parameters are set to the same as in Section IV-A. It can be seen that the proposed LMEA is still able to achieve a competitive performance on the LSMOP test problems, which confirms the effectiveness of LMEA in solving challenging large-scale MaOPs.

V. CONCLUSION

In this paper, a decision variable clustering based evolutionary algorithm, termed LMEA, has been suggested for solving large-scale MaOPs. In LMEA, a decision variable clustering method has been developed to divide the decision variables into two groups, namely convergence-related variables and diversity-related variables, according to their contributions to the convergence and diversity, respectively. Based on the grouping results, the two groups of variables are optimized separately. Moreover, a fast nondominated sorting approach with a time complexity of $O(MN \log N / \log M)$, called T-ENS, has been proposed to improve the computational efficiency of LMEA, where M is the number of objectives and N is the size of population to perform nondominated sorting. Empirical results on a variety of large-scale benchmark MaOPs that have up to 5000 decision variables demonstrate the promising scalability of LMEA in comparison with several state-of-the-art MOEAs.

It has been shown that decision variable clustering is effective for solving large-scale MaOPs. While the proposed clustering method in this paper only considers the decision variables to be convergence-related and diversity-related, in the future, we would like to investigate if it is possible to make more classifications of the decision variables, because decision variables in the same group may still have different (strong or weak) correlations with the objectives even if they are only related to convergence or diversity. In addition, we would also like to further improve the computational efficiency of the decision variable interaction analysis procedure, which is the main computational cost of the proposed LMEA.

REFERENCES

- [1] P. J. Fleming, R. C. Purshouse, and R. J. Lygoe, "Many-objective optimization: An engineering design perspective," in *Proc. 3rd Int. Conf. Evol. Multi Criterion Optim.*, Guanajuato, Mexico, 2005, pp. 14–32.
- [2] J. Garcia, A. Berlanga, and J. M. M. López, "Effective evolutionary algorithms for many-specifications attainment: Application to air traffic control tracking filters," *IEEE Trans. Evol. Comput.*, vol. 13, no. 1, pp. 151–168, Feb. 2009.
- [3] J. B. Kollat, P. M. Reed, and R. M. Maxwell, "Many-objective groundwater monitoring network design using bias-aware ensemble Kalman filtering, evolutionary optimization, and visual analytics," *Water Resources Res.*, vol. 47, no. 2, pp. 1–18, 2011.
- [4] J. W. Krusselbrink *et al.*, "Combining aggregation with Pareto optimization: A case study in evolutionary molecular design," in *Proc. 5th Int. Conf. Evol. Multi Criterion Optim.*, Nantes, France, 2009, pp. 453–467.
- [5] K. Ikeda, H. Kita, and S. Kobayashi, "Failure of Pareto-based MOEAs: Does non-dominated really mean near to optimal?" in *Proc. IEEE Congr. Evol. Comput.*, Seoul, South Korea, 2001, pp. 957–962.
- [6] V. Khare, X. Yao, and K. Deb, "Performance scaling of multiobjective evolutionary algorithms," in *Proc. 2nd Int. Conf. Evol. Multi Criterion Optim.*, Faro, Portugal, 2003, pp. 376–390.
- [7] D. Brockhoff *et al.*, "On the effects of adding objectives to plateau functions," *IEEE Trans. Evol. Comput.*, vol. 13, no. 3, pp. 591–603, Jun. 2009.
- [8] R. C. Purshouse and P. J. Fleming, "Evolutionary many-objective optimisation: An exploratory analysis," in *Proc. Congr. Evol. Comput.*, Canberra, ACT, Australia, 2003, pp. 2066–2073.
- [9] R. C. Purshouse and P. J. Fleming, "On the evolutionary optimization of many conflicting objectives," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 770–784, Dec. 2007.
- [10] S. Chand and M. Wagner, "Evolutionary many-objective optimization: A quick-start guide," *Surveys Oper. Res. Manag. Sci.*, vol. 20, no. 2, pp. 35–42, 2015.
- [11] B. Li, J. Li, K. Tang, and X. Yao, "Many-objective evolutionary algorithms: A survey," *ACM Comput. Surveys*, vol. 48, no. 1, pp. 1–37, 2015.
- [12] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, "Combining convergence and diversity in evolutionary multiobjective optimization," *Evol. Comput.*, vol. 10, no. 3, pp. 263–282, 2002.
- [13] D. Hadka and P. Reed, "Borg: An auto-adaptive many-objective evolutionary computing framework," *Evol. Comput.*, vol. 21, no. 2, pp. 231–259, 2013.
- [14] X. Zou, Y. Chen, M. Liu, and L. Kang, "A new evolutionary algorithm for solving many-objective optimization problems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 5, pp. 1402–1412, Oct. 2008.
- [15] G. Wang and H. Jiang, "Fuzzy-dominance and its application in evolutionary many objective optimization," in *Proc. Int. Conf. Comput. Intell. Security Workshops*, Harbin, China, 2007, pp. 195–198.
- [16] F. di Pietro, S.-T. Khu, and D. A. Savić, "An investigation on preference order ranking scheme for multiobjective evolutionary optimization," *IEEE Trans. Evol. Comput.*, vol. 11, no. 1, pp. 17–45, Feb. 2007.
- [17] Y. Yuan, H. Xu, B. Wang, and X. Yao, "A new dominance relation-based evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 20, no. 1, pp. 16–37, Feb. 2016.
- [18] M. Köppen and K. Yoshida, "Substitute distance assignments in NSGA-II for handling many-objective optimization problems," in *Proc. 4th Int. Conf. Evol. Multi Criterion Optim.*, Matsushima, Japan, 2007, pp. 727–741.
- [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] R. Wang, R. C. Purshouse, and P. J. Fleming, "Preference-inspired coevolutionary algorithms for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 17, no. 4, pp. 474–494, Aug. 2013.
- [21] J. Cheng, G. G. Yen, and G. Zhang, "A many-objective evolutionary algorithm with enhanced mating and environmental selections," *IEEE Trans. Evol. Comput.*, vol. 19, no. 4, pp. 592–605, Aug. 2015.
- [22] 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. 2014.
- [23] C. A. R. Villalobos and C. A. C. Coello, "A new multi-objective evolutionary algorithm based on a performance assessment indicator," in *Proc. 14th Annu. Conf. Genet. Evol. Comput. (GECCO)*, Philadelphia, PA, USA, 2012, pp. 505–512.
- [24] T. Ulrich, J. Bader, and L. Thiele, "Defining and optimizing indicator-based diversity measures in multiobjective search," in *Proc. 11th Int. Conf. Parallel Problem Solving Nat.*, Kraków, Poland, 2010, pp. 707–717.
- [25] H. Trautmann, T. Wagner, and D. Brockhoff, "R2-EMOA: Focused multiobjective search using R2-indicator-based selection," in *Proc. 7th Int. Conf. Learn. Intell. Optim.*, Catania, Italy, 2013, pp. 70–74.
- [26] G. Rudolph, O. Schütze, C. Grimme, and H. Trautmann, "An aspiration set EMOA based on averaged hausdorff distances," in *Proc. 8th Int. Conf. Learn. Intell. Optim.*, Gainesville, FL, USA, 2014, pp. 153–156.
- [27] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Proc. 8th Int. Conf. Parallel Problem Solving Nat.*, Birmingham, U.K., 2004, pp. 832–842.

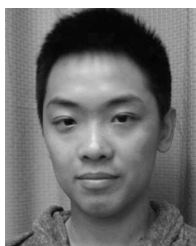
- [28] N. Beume, B. Naujoks, and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *Eur. J. Oper. Res.*, vol. 181, no. 3, pp. 1653–1669, 2007.
- [29] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *Evol. Comput.*, vol. 19, no. 1, pp. 45–76, 2011.
- [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] K. Li, Q. Zhang, S. Kwong, M. Li, and R. Wang, "Stable matching-based selection in evolutionary multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 18, no. 6, pp. 909–923, Dec. 2014.
- [32] Y. Qi *et al.*, "MOEA/D with adaptive weight adjustment," *Evol. Comput.*, vol. 22, no. 2, pp. 231–264, 2014.
- [33] X. Cai, Y. Li, Z. Fan, and Q. Zhang, "An external archive guided multiobjective evolutionary algorithm based on decomposition for combinatorial optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 4, pp. 508–523, Aug. 2015.
- [34] Y. Yuan, H. Xu, B. Wang, B. Zhang, and X. Yao, "Balancing convergence and diversity in decomposition-based many-objective optimizers," *IEEE Trans. Evol. Comput.*, vol. 20, no. 2, pp. 180–198, Apr. 2016.
- [35] S. B. Gee, K. C. Tan, V. A. Shim, and N. R. Pal, "Online diversity assessment in evolutionary multiobjective optimization: A geometrical perspective," *IEEE Trans. Evol. Comput.*, vol. 19, no. 4, pp. 542–559, Aug. 2015.
- [36] H.-L. Liu, F. Gu, and Q. Zhang, "Decomposition of a multiobjective optimization problem into a number of simple multiobjective sub-problems," *IEEE Trans. Evol. Comput.*, vol. 18, no. 3, pp. 450–455, Jun. 2014.
- [37] 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.
- [38] 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.
- [39] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A reference vector guided evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, to be published.
- [40] D. Brockhoff and E. Zitzler, "Are all objectives necessary? On dimensionality reduction in evolutionary multiobjective optimization," in *Proc. 9th Int. Conf. Parallel Problem Solving Nat.*, Reykjavik, Iceland, 2006, pp. 533–542.
- [41] D. Brockhoff and E. Zitzler, "Objective reduction in evolutionary multi-objective optimization: Theory and applications," *Evol. Comput.*, vol. 17, no. 2, pp. 135–166, 2009.
- [42] A. L. Jaimes, C. A. C. Coello, and D. Chakraborty, "Objective reduction using a feature selection technique," in *Proc. 10th Annu. Conf. Genet. Evol. Comput.*, Atlanta, GA, USA, 2008, pp. 673–680.
- [43] H. K. Singh, A. Isaacs, and T. Ray, "A Pareto corner search evolutionary algorithm and dimensionality reduction in many-objective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 15, no. 4, pp. 539–556, Aug. 2011.
- [44] K. Deb and D. Saxena, "Searching for Pareto-optimal solutions through dimensionality reduction for certain large-dimensional multi-objective optimization problems," in *Proc. World Congr. Comput. Intell.*, Vancouver, BC, Canada, 2006, pp. 3352–3360.
- [45] D. K. Saxena and K. Deb, "Non-linear dimensionality reduction procedures for certain large-dimensional multi-objective optimization problems: Employing coreentropy and a novel maximum variance unfolding," in *Proc. 4th Int. Conf. Evol. Multi Criterion Optim.*, Matsushima, Japan, 2007, pp. 772–787.
- [46] D. K. Saxena, J. A. Duro, A. Tiwari, K. Deb, and Q. Zhang, "Objective reduction in many-objective optimization: Linear and nonlinear algorithms," *IEEE Trans. Evol. Comput.*, vol. 17, no. 1, pp. 77–99, Feb. 2013.
- [47] H. Wang and X. Yao, "Objective reduction based on nonlinear correlation information entropy," *Soft Comput.*, vol. 20, no. 6, pp. 2393–2407, 2016.
- [48] M. Li, S. Yang, and X. Liu, "Bi-goal evolution for many-objective optimization problems," *Artif. Intell.*, vol. 228, pp. 45–65, Nov. 2015.
- [49] B.-Y. Qu and P. N. Suganthan, "Multi-objective evolutionary algorithms based on the summation of normalized objectives and diversified selection," *Inf. Sci.*, vol. 180, no. 17, pp. 3170–3181, 2010.
- [50] K. Bringmann, T. Friedrich, C. Igel, and T. Voß, "Speeding up many-objective optimization by Monte Carlo approximations," *Artif. Intell.*, vol. 204, pp. 22–29, Nov. 2013.
- [51] K. McClymont and E. Keedwell, "Deductive sort and climbing sort: New methods for non-dominated sorting," *Evol. Comput.*, vol. 20, no. 1, pp. 1–26, Mar. 2012.
- [52] H. Wang and X. Yao, "Corner sort for Pareto-based many-objective optimization," *IEEE Trans. Cybern.*, vol. 44, no. 1, pp. 92–102, Jan. 2014.
- [53] M. Drozdík, Y. Akimoto, H. Aguirre, and K. Tanaka, "Computational cost reduction of nondominated sorting using the M-front," *IEEE Trans. Evol. Comput.*, vol. 19, no. 5, pp. 659–678, Oct. 2015.
- [54] X. Zhang, Y. Tian, R. Cheng, and Y. Jin, "An efficient approach to non-dominated sorting for evolutionary multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 2, pp. 201–213, Apr. 2015.
- [55] X. Zhang, Y. Tian, and Y. Jin, "Approximate non-dominated sorting for evolutionary many-objective optimization," *Inf. Sci.*, Jun. 2016.
- [56] Z. Yang, K. Tang, and X. Yao, "Large scale evolutionary optimization using cooperative coevolution," *Inf. Sci.*, vol. 178, no. 15, pp. 2985–2999, 2008.
- [57] 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.
- [58] 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.
- [59] 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.
- [60] K. H. Hajikolaie, Z. Pirmoradi, G. H. Cheng, and G. G. Wang, "Decomposition for large-scale global optimization based on quantified variable correlations uncovered by metamodelling," *Eng. Optim.*, vol. 47, no. 4, pp. 429–452, 2015.
- [61] J. Fan, J. Wang, and M. Han, "Cooperative coevolution for large-scale optimization based on kernel fuzzy clustering and variable trust region methods," *IEEE Trans. Fuzzy Syst.*, vol. 22, no. 4, pp. 829–839, Aug. 2014.
- [62] R. Cheng and Y. Jin, "A social learning particle swarm optimization algorithm for scalable optimization," *Inf. Sci.*, vol. 291, pp. 43–60, Jan. 2015.
- [63] L. M. Antonio and C. A. C. Coello, "Use of cooperative coevolution for solving large scale multiobjective optimization problems," in *Proc. IEEE Congr. Evol. Comput.*, Cancún, Mexico, 2013, pp. 2758–2765.
- [64] 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.
- [65] R. Shang, K. Dai, L. Jiao, and R. Stolkin, "Improved memetic algorithm based on route distance grouping for multiobjective large scale capacitated arc routing problems," *IEEE Trans. Cybern.*, vol. 46, no. 4, pp. 1000–1013, Apr. 2016.
- [66] 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.*, Kraków, Poland, 2010, pp. 300–309.
- [67] K. Deb and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex Systems*, vol. 9, no. 4, pp. 115–148, 1995.
- [68] K. Deb and M. Goyal, "A combined genetic adaptive search (GeneAS) for engineering design," *Comput. Sci. Inform.*, vol. 26, no. 4, pp. 30–45, 1996.
- [69] B. V. Babu and M. M. L. Jehan, "Differential evolution for multi-objective optimization," in *Proc. Congr. Evol. Comput.*, vol. 4, Canberra, ACT, Australia, 2003, pp. 2696–2703.
- [70] T. Robič and B. Filipič, "DEMO: Differential evolution for multiobjective optimization," in *Proc. 3rd Int. Conf. Evol. Multi Criterion Optim.*, Guanajuato, Mexico, 2005, pp. 520–533.
- [71] 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.
- [72] O. Schütze, "A new data structure for the nondominance problem in multi-objective optimization," in *Proc. 2nd Int. Conf. Evol. Multi Criterion Optim.*, Faro, Portugal, 2003, pp. 509–518.
- [73] H. Fang, Q. Wang, Y.-C. Tu, and M. F. Horstemyer, "An efficient non-dominated sorting method for evolutionary algorithms," *Evol. Comput.*, vol. 16, no. 3, pp. 355–384, 2008.
- [74] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, *Scalable Test Problems for Evolutionary Multiobjective Optimization*. London, U.K.: Springer, 2005.
- [75] 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.

- [76] Q. Zhang *et al.*, "Multiobjective optimization test instances for the CEC 2009 special session and competition," School Comput. Sci. Elect. Eng., Univ. Essex, Colchester, U.K., Working Rep. CES-887, pp. 1–30, 2008.
- [77] 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.
- [78] 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.
- [79] 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.
- [80] L. M. S. Russo and A. P. Francisco, "Quick hypervolume," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 481–502, Aug. 2014.
- [81] I. Das and J. E. Dennis, "Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems," *SIAM J. Optim.*, vol. 8, no. 3, pp. 631–657, 1998.
- [82] R. Cheng, Y. Jin, and K. Narukawa, "Adaptive reference vector generation for inverse model based evolutionary multiobjective optimization with degenerate and disconnected Pareto fronts," in *Proc. Evol. Multi Criterion Optim.*, Guimarães, Portugal, 2015, pp. 127–140.
- [83] L.-Y. Tseng and C. Chen, "Multiple trajectory search for unconstrained/constrained multi-objective optimization," in *Proc. IEEE Congr. Evol. Comput.*, Trondheim, Norway, 2009, pp. 1951–1958.
- [84] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "Test problems for large-scale multi- and many-objective optimization," *IEEE Trans. Cybern.*, to be published.



Xingyi Zhang 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.



Ye Tian received the B.Sc. and M.Sc. degrees from Anhui University, Hefei, China, in 2012 and 2015, respectively, where he is currently pursuing the Ph.D. degree with the School of Computer Science and Technology.

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



Ran Cheng received the B.Eng. 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 a Research Fellow with the School of Computer Science, University of Birmingham, Birmingham, U.K. His current research interests include multi/many-objective optimization, large-scale optimization, model-based evolutionary algorithms, and swarm intelligence.



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 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 Agency for Innovation (Tekes) and a Changjiang Distinguished Visiting Professor appointed by the Ministry of Education, Beijing, China. He has (co)authored over 200 peer-reviewed journal and conference papers and been granted eight patents on evolutionary optimization. He has delivered 20 invited keynote speeches at international conferences. His current research interests include computational intelligence, computational neuroscience, computational systems biology, and nature-inspired, real-world driven problem-solving.

Dr. Jin was a recipient of the Best Paper Award of the 2010 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology and the 2014 IEEE Computational Intelligence Magazine Outstanding Paper Award. He is the Editor-in-Chief of the IEEE TRANSACTIONS ON COGNITIVE AND DEVELOPMENTAL SYSTEMS and *Complex and Intelligent Systems*. He is also an Associate Editor or an Editorial Board Member of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, the IEEE TRANSACTIONS ON CYBERNETICS, the IEEE TRANSACTIONS ON NANOBIOSCIENCE, *Evolutionary Computation*, *BioSystems*, *Soft Computing*, and *Natural Computing*. He was an IEEE Distinguished Lecturer from 2013 to 2015 and the Vice President for Technical Activities of the IEEE Computational Intelligence Society from 2014 to 2015.