IEEE Access
Multidisciplinary Rapid Review Open Access Journal

Received November 28, 2019, accepted December 9, 2019, date of publication December 19, 2019, date of current version December 31, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2960874

Many-Objective Brain Storm Optimization Algorithm

YALI WU, XINRUI WANG, YULONG FU, AND GUOTING LI
School of Automation and Information Engineering, Xi'an University of Technology, Xi'an 710048, China
Shaanxi Province Key Laboratory of Complex System Control and Intelligent Information Processing, Xi'an 710048, China
Corresponding author: Xinrui Wang (xinruiwang6@163.com)

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFB1703000, and in part by the National Youth Foundation of China under Grant 61503299 and Grant 61502385.

ABSTRACT In recent years, many evolutionary algorithms and population-based algorithms have been developed for solving many-objective optimization problems. Inspired by the human brainstorming conference, Brain Storming Optimization (BSO) algorithm was guided by the cluster centers and other individuals with probability, which can balance convergence and diversity greatly. In this paper, the authors propose a novel brain storm optimization algorithm for many-objective optimization problem. The algorithm adopts the decision variable clustering method to divides the variables into convergence-related variables and diversity-related variables. The decomposition strategy is designed to increases selection pressure for the convergence-related variables, while the reference point's strategy is adopted for the diversity-related variables to update the population and increase the diversity. Experimental results show that the proposed many-objective brain storm optimization algorithm is a very promising algorithm for solving many-objective optimization problems.

INDEX TERMS Brain storm optimization, decision variable clustering method, decomposition strategy, reference point, many-objective optimization.

I. INTRODUCTION

As a typical complex optimization problem, multi-objective optimization problems (MOPs) are widely exist in engineering and real life, such as groundwater monitoring [1], PID controller design [2], power quality monitoring [3] and so on. Nowadays, multi-objective optimization algorithms have made great progress, and various strategies were also gradually matured. With the development of the science and technology, the problems faced by reality are becoming more and more complex. Most of them show the characteristics of dynamic, multi-constrained, large-scale, and increased number of objectives. Generally, we call the MOPs with more than three objectives as the many-objective optimization problems (MaOPs).

Compared with the single-objective optimization problems, MaOPs are obviously more practical but more complicated. Due to the conflicting nature of the objectives, it is impossible to find the best solution that is able to optimize all the objectives simultaneously. Therefore, Pareto-optimal solutions, a set of optimal solutions representing the trade-offs between different objectives, can be achieved, which is called Pareto Set (PS) in the decision space. The set of objective functions corresponding to the Pareto-optimal solutions are known as the Pareto Front (PF). As the number of objectives increases, most multi-objective evolutionary algorithms (MOEAs) for MOPs which involve only two or three objectives, cannot easily deal with MaOPs for the loss of convergence pressure and the difficulty in diversity maintenance [4]–[8]. Different methods have been proposed to tackle the above two issues [9], [10].

Due to the proportion of non-dominated solutions in the population increases sharply with the number of objectives increases, nearly all the solutions in a population become non-dominated with one another [11] in MOPs, which decrease the convergence pressure. Fig.1 gives the relationship between the proportions of non-dominated solutions in the population and the number of objectives. It can be seen that as the number of objectives increases to 6, about half of individuals in the population are non-dominated solutions. And when the number is greater than 12, nearly all the individuals in the population become non-dominated solutions.

The associate editor coordinating the review of this manuscript and approving it for publication was Bilal Alatas.

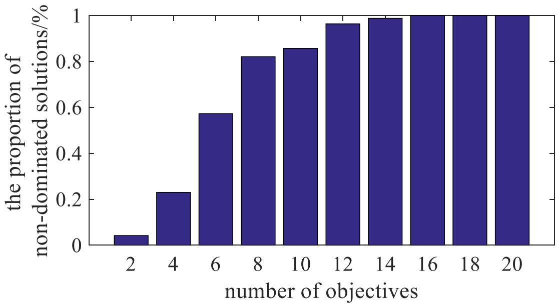

FIGURE 1. The relationship between the proportion of non-dominated solutions in the population and the number of the objective.

It means that the advantages and disadvantages of individuals in the population can hardly be distinguished if the dominance strategy is adopted as the criterion for population selection. The main approaches to solve this problem can be divided into two categories: one is to modify the traditional Pareto dominance definition to increase the selection pressure toward the Pareto front, such as ž-dominance [12], L-optimality [13], fuzzy dominance [14], grid dominance [15], θ-dominance [16]. The other is to adopt performance indicators as selection criterion to distinguish non-dominated solutions which cannot be distinguished by traditional Pareto dominance, such as the indicator based evolutionary algorithm [17], the S-metric selection-based evolutionary multi-objective algorithm [18], a dynamic neighborhood MOEA based on hypervolume (HV) indicator [19], the fast HV-based EA (HypE) [20], and the enhanced inverted generational distance (IGD-NS) indicator based MOEA [21].

It is simple and effective to maintain the diversity of PS when the number of objectives is 2 or 3 in MaOPs as the Pareto front is a curve (surface). The candidate solutions will sparsely be distributed in the high dimensional space with the increase of objectives, which will lead to the ineffectiveness of the traditional diversity maintenance strategy. The decomposition method, which decomposes MaOPs into a set of subproblems [22]–[24], is proposed to solve this problem. There are mainly two categories of decomposition approaches. One is to decompose MaOPs into a set of single-objective problems (SOPs), such as MOEA/D algorithm [22], and several variants of MOEA/D [25]–[29]. The other is to decompose MaOPs into a set of simple MOPs, such as MOEA/D-M2M [30], reference-point based many objective NSGA-II (NSGA-III) [31], and the recently proposed reference vector guided evolutionary algorithm (RVEA) [32].

During the last decades, a number of population-based methods, especially evolutionary algorithms and swarm intelligence algorithms such as RVEA [32], MOEA/DD [23] MMOPSO [33], NMPSO [34], SetGA [35], Nondominated sorting genetic algorithm, have been successfully used to solve MaOPs. Although these MaOEAs showed competitive performance, they still cannot achieve good result for the more difficult MaOPs with discontinuous and irregular PFs, especially for the MaF test problems recently proposed in [36].

Some previous interesting works-on embedding clustering into MOEAs, such as CA-MOEA [37], MaOEA/C [38], MOEA/DVA [24], LMEA [39], has paid more attention in recent year for dealing with large-size MaOPs. These algorithms have shown the potential ability for MaOPs with discontinuous and irregular PFs.

Brain Storm Optimization (BSO) was proposed in 2011 by Professor Shi Yuhui. The algorithm is guided by the cluster centers and other individuals with probability, which can balance convergence and diversity greatly. On the basis of classical algorithm, different researchers have derived a variety of Multi-objective Brain Storm Optimization algorithms (MOBSO) focused on their clustering, mutation and the updating archive sets [40]–[43]. Clustering in the objective space and elite strategy were taken to improve the convergence performance of MOBSO in [40]. An improved Multi-objective Brain Storm Optimization algorithm (MMOBSO) was proposed in which DBSCAN is taken to replace the k-means clustering and differential mutation to Gaussian mutation [41]. A cyclic crowding distance is designed to update the archive set to improve the diversity of the archive set (SMOBSO) [42]. Furthermore, MOBSO was also used to solve the environmental and economic dispatch problem of combined heat and power [44].

Based on the evolutionary advantages of BSO and the selection strategy of clustering, a novel optimization algorithm named Many-objective Brain Storm Optimization (MaOBSO) algorithm is proposed in this paper to solve many-objective optimization problems. In the proposed algorithm, the new clustering methods with the reference point allocation and decision variable clustering are implemented. The reference point allocation method is performed to increase the diversity of individuals. The decision variable clustering is designed to divide the decision variables into convergence-related variables and the diversity-related variables, and different mutation operations designed to optimize different categories variable, accordingly it is beneficial to get the reasonable distribution of PS. The performance of MaOBSO is evaluated using the DTLZ [45] benchmark function and the MaF [36] test problems (CEC2018) with a number of objectives. As we will see later on, when compared to five state-of-the-art MOEAs for many-objective optimization, MOEA/DD [23], NSGA-III [31], LMEA [39], AR-MOEA [21], and RVEA [32], MaOBSO performed better on most of the test problems adopted.

To conclude, the main contributions of this paper are as follows:

- The adaptive clustering is performed according to the angles between the individuals and the reference points in the iteration. The reference points are generated by the NBI method to enhance the guidance of the algorithm. Moreover, the reference points corresponding to the individual is taken as a selection in the updating process of convergence optimization and diversity optimization. Therefore, the reference points in the proposed algorithm are the learning direction of the individual.

- The corner points are adopted as the clustering centers and the clustering operation is designed in the objective space. A new adaptive clustering method corresponding to the reference point allocation is designed in the proposed algorithm for MaOBSO. The diversity of the proposed algorithm is enhanced by learning from the clustering centers with a certain probability.
- The index of the convergence evolutionary is designed when the convergence degree of the algorithm reaches a certain level. The diversity optimization plays an important role in the iteration process.

The rest of this paper is organized as follows: the related works are described in Section II. The principle and implement process of MaOBSO are listed in Section III. The comparison of the simulation experiments with the current algorithms is discussed in Section IV. The conclusions and outlooks are given in Section V.

II. RELATED WORK

A. BRAIN STORM OPTIMIZATION

BSO algorithm was proposed in 2011 and it was inspired by the human brainstorming conference [46]. Every thought generated is regarded as the potential solution. The facilitator facilitates the process of generating ideas which obey the Osborn's original four rules, that is, ''Suspend Judgment'', ''Anything Goes'', ''Cross-fertilize (Piggyback)'', ''Go for Quantity'', which produce many possible solutions. In BSO, two basic operations, which are converging operation (individual clustering) and diverging operation (individual updating), are implemented in each iteration [40]. Converging operation is a kind of techniques that divides individuals into several groups (clusters). The goal of clustering is to make the individuals being similar (or related) to one another are in the same cluster, and being different from (or related to) each other in different clusters. The clustering operation could refine a search area, and a probability value is used to replacing a cluster center by a randomly generated solution. This could avoid the premature convergence and the local optima. Diverging operation is new individual generation process. A new individual can be generated by one or two parent individuals. One cluster could refine a search region and improve the exploitation ability. Two clusters may be from these clusters and improve the diversity of population.

At present, BSO is widely paid more attention by researchers coming from various fields. Different new versions have been proposed for different problems to improve the performance of BSO algorithm. A modified BSO, named VGLBSO, adopts the random grouping scheme in the grouping operator to reduce the algorithm computational burden [47]. MIIBSO algorithm with multi-information interaction strategy is proposed to avoid the algorithm premature convergence [48]. A hybrid algorithm which is integrated the simulated annealing process into the brain storm optimization algorithm is proposed to solve continuous optimization problems in [49].

B. MULTI-OBJECTIVE BRAIN STORM OPTIMIZATION

MOBSO algorithm is proposed firstly in [40]. The algorithm introduces the strategy of archive set with non-dominated solutions in each iteration to get a group of solution close enough to Pareto front and uniform solution. A grid-based method and a hybrid mutation strategy integrating above traditional Gaussian-, Cauchy- and Chaotic-based mutation replace k-means clustering and Gaussian mutation in [50] to enhance the diversity and avoiding the premature convergence. A random probabilistic decision-making of river formation dynamics scheme to select optimal cluster centroids during population generation step, and an adaptive mutation operator were taken in [51] to improve the performance of IMBSO. A multi-objective brainstorming algorithm based on multiple indicators (MIBSO) is proposed in [52] to extend BSO algorithms. And the combination of the decomposition technology and multi-objective brain storm optimization algorithm (MBSO/D) is proposed to improve the search efficiency in [53].

From the discussion above, we can see that MOBSOs have shown a promising performance when solving MOPs, such as MOBSO [40], MMOBSO [41], SMOBSO [42] and so on. But the performance of MOBSOs is seldom investigated in solving MaOPs.

C. MOTIVATIONS OF OUR APPROACH

Due to the size of the non-dominated solutions in the population increases sharply as the number of objectives increase, the advantages and disadvantages of individuals in the population can hardly be distinguished when the classical update strategy of the archive set of original MOBSO. Inspired by the achievement of other swarm intelligent optimization algorithm for MaOPs, a novel Many-objective Brain Storm Optimization Algorithm is proposed in this paper. In the proposed algorithm, the decision variable classification strategy is designed to divide the decision variables into the convergence-related variables and the diversity-related variables. The different optimization methods are carried out according to their characteristics respectively. For the convergence-related variables, we use the idea of the divide-and-conquer to improve the optimization efficiency. The penalty-based boundary intersection (PBI) method is adopted to evaluate the individuals and make trade-offs. For the diversity-related variables, the individuals in the population are guided by the cluster centers and learn from each other by the probability to enhance the diversity of the population. The benchmark functions of DTLZ and MaF are used to evaluate the performance of MaOBSO.

III. MANY-OBJECTIVE BRAIN STORM OPTIMIZATION ALGORITHM

In this section, the details of MaOBSO are described. We present the main framework of MaOBSO firstly. Then the main three components, i.e., the initialization procedure, the converging operation, and the diverging operation, are discussed in the following sections.

Algorithm 1 The Main Framework of MaOBSO

Input: M (numbers of objectives), E_{max} (maximum number of evaluation), P_o (Probability of selecting a new individual with an individual), P_c (Probability of selecting new individuals with a cluster center);

Output: final population;

1: $P \leftarrow$ Initalize(M);
2: Generate reference vectors w with NBI;
3: [w, *cluster*] $\leftarrow$ VectorAllocation(P, w, M);
4: [DV,CV] $\leftarrow$ VariableCluster(P);
5: *subCVs* $\leftarrow$ InteractionAnalysis(P, CV);
6: **while** termination criterion is not fulfilled **do**
7: **if** Convergence condition is not fulfilled **then**
8: $P \leftarrow$ ConverOptimi(P, *subCVs*);
9: **end if**
10: $P \leftarrow$ DiverOptimi(P, P_o, P_c, DV, w, *cluster*);
11: **end while**

A. THE MAIN FRAMEWORK OF MaOBSO

The main framework of MaOBSO is listed in Algorithm 1. As the same as the other swarm intelligent optimization algorithm, the initialization procedure generates population P while the reference points w by NBI firstly. Then the converging operations including the individual clustering and decision variables clustering are designed secondly. The individual clustering adopts the allocation strategy of the reference points. The corner points are used as the clustering centers to cluster the reference points. The decision variable classification strategy divides the variables into the convergence-related variables and diversity-related variables. Lastly, the diverging operation, which includes the convergence-related variables and diversity-related variables, are optimized respectively. The convergence-related variables optimization uses roulette method to select parents' individuals and binary crossover to generate new individuals, the updating individuals use PBI as the standard. The smaller the PBI value, the better the individual. The diversity-related variables optimization uses the process of selection and mutation in original BSO. And the updating of the population is given by the reference points. The implementation details of each component in MaBSO will be explained in the following subsections.

B. THE INITIALIZATION PROCEDURE

The initialization procedure of MaOBSO contains two main aspects. One is the initialization of parent population P (line 1 in Algorithm 1), the other is the generation of reference vectors w (line 2 in Algorithm 1). To be specific, the initial parent population P is randomly sampled via a uniform distribution. The reference points in the objective space are generated by the Normal-Boundary Intersection (NBI) method proposed

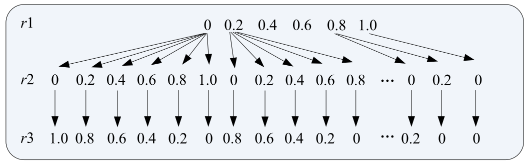

FIGURE 2. The process of generating reference points.

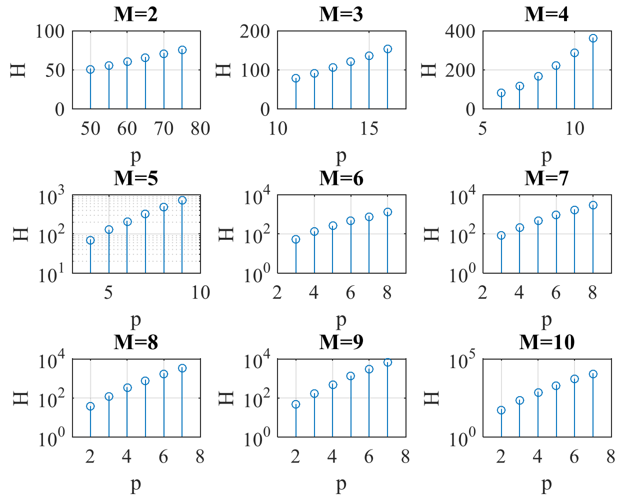

FIGURE 3. The relationship between ***H***, ***M*** and ***p***.

by Das et al [54]. The process of generating reference points with 3 objectives is shown in Fig.2, where $r1$, $r2$, and $r3$ are the dimensions of the objective, respectively. If p divisions are considered along each objective, the total number of reference points (H) in an M-objective problem is given by:

$$H = C_{M+p-1}^{M-1} \tag{1}$$

It can be seen from the formula (1) that H will increases with p when M is fixed. The relationship between H, M and p is shown in Fig.3.

From the Fig.3, we can see that H is very sensitive to p, as the number of objective increases. H will close to 100 when $M = 7, p = 3$, and even $H = 364$ when $M = 10, p = 3$. This leads more pressure on the computational burden. If we solve this problem by lowering H, it will make the reference points sparely lay along the boundary. Fig.4(a) shows the reference points of 3 objectives by single-layer. So we present a two-layer reference point generation method the same as the reference [31]. The main process includes three steps. Firstly the boundary reference points $B = \{B_1, B_2, \ldots, B_{H1}\}$ and the inside reference points $I = \{I_1, I_2, \ldots, I_{H2}\}$ is produced, the number of reference points are $H1 = Cp1\ M + p1 - 1$ and $H2 = Cp2\ M + p2 - 1$. Then the parameter I inward is reduced by a shrinkage factor $\tau \in [0, 1]$ which computed by the formula $I = (1-\tau)/m+\tau * I$. Finally, the reference points is obtained by $P = I \cup B$, and the total number of the reference points is $H = H1 + H2$. Fig.4(b) shows the reference points of $M = 3, p1 = 1, p2 = 1, \tau = 0.5$ by two-layer.

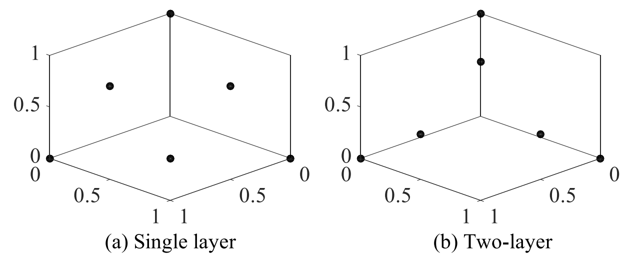

FIGURE 4. Reference point schematic of M=3, p=2.

C. CONVERGING OPEREATION

The converging operation is the clustering procedure. It consists of two parts: individual clustering and decision variables clustering. Individual clustering means the clustering methods in the original BSO algorithm. The objective of individual clustering is to cluster the population, and find the clustering center in the iteration at first. Then the clustering center is served as a guide and the other individuals learn from them. The individual clustering realizes the multiple classes parallel local search process. It increases the diversity of individuals and improves the search efficiency of brainstorming process. The decision variables' clustering is the processing of variables in MaOPs. Different variables affect the Pareto frontier with the different degree. So, the variables should be classified and optimized with different mutation operations according to their different categories.

1) REFERENCE POINT ALLOCATION

As the discussion above, the individual clustering procedure adopts the reference point allocation. The principle of the allocation of the individual is corresponded to the reference points by the angle between the individual and the reference points. From the point of mathematical model, the corner points play an important role in the diversity of algorithms, especially the breadth of distribution. Therefore, the corner points can be indirectly more potential. Compared with other individuals, it is more positive role that the corner point serves as the clustering center in the evolution of BSO algorithm. When the reference points are allocated, the individual is also clustered as followed. Algorithm 2 (line 3 in Algorithm 1) is a specific process of clustering and allocating reference points. The corner point in the reference points is serves as the cluster center, and the reference points w are clustered by the k-means clustering method. The angles are calculated between all individuals and all reference points, and we obtain the angles matrix of $N \times N$. Then the reference points correspond to the individuals which the smallest angle in the matrix. Finally, the clustering results based on reference points are corresponding to the individual, so the process achieves the adaptive clustering of the population.

Fig.5 shows the clustering result of the reference points when $M = 3 \sim 8$ in the benchmark function DTLZ1, where the solid points and the bold lines are the cluster center, and the higher probability of these points and lines has a positive effect on the scalability of the archive set in the new individual.

Algorithm 2 Vector Allocation (P, w, M)

Input: P (population), w (reference point /vector), M(objectives);
Output: w (allocated vector), *cluster*(clustering result);
1: *cluster* $\leftarrow$ k-means clustering of w with corners as clustering centers;
2: *Angle* $\leftarrow$ calculate the angles of each individual in P to w; /∗*Angle* is a matrix of $N \times N$, *Angle*[i][j] is the angle of ith individual with jth *vector*, and N is population size ∗/
3: **for** $k = 1$ to N **do**
4: *sortindex*[i] $\leftarrow j$; /∗[i, j] is the subscript of the minimum in *Angle*∗/
5: $Q1 \leftarrow$ row i of Angle;
6: *Angle* $\leftarrow$*Angle*/$Q1$;
7: $Q2 \leftarrow$ column j of *Angle*;
8: *Angle* $\leftarrow$*Angle*/$Q2$;
9: $w'[k] \leftarrow w[sortindex[i]]$;
10: *cluster*'[k] $\leftarrow$*cluster*[*sortindex*[i]];
11: **end for**
12: $w \leftarrow w'$;
13: *cluster* $\leftarrow$*cluster*';

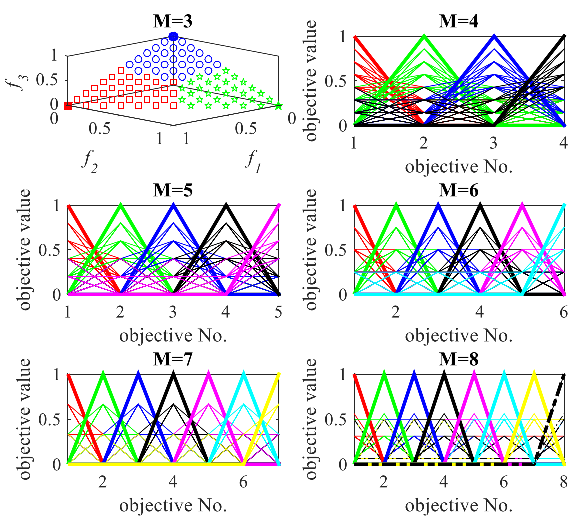

FIGURE 5. The result of reference point clustering.

2) DECISION VARIABLE CLUSTERING

Different variables of the problem have different effects on the final PS. It is very sensitive to decision variables changes for some problems. If a variable is changed slightly, the fitness value is far away from the real Pareto front or has a great impact on the overall uniformity. Take the following MOP as an example:

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

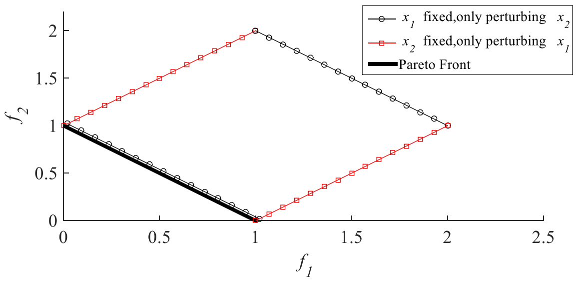

FIGURE 6. The fitness distribution obtained by perturbing x_1 and x_2 respectively.

Fig.6 shows the sampling points obtained on the MOP formulated in (2) by perturbing one variable between [0,1] while fixing the others. If changing the value of x_2 within the defined domain when x_1 is fixed, it can be seen that the fitness to be close or far from the Pareto front. And changing x_2 in $x = (x_1, x_2)$ can only result in a decision vector which equals x, dominates x or is dominated by x. So x_2 affects the convergence of the MOP. If changing x_1 when x_2 is fixed, it can be seen that the step size of x_1 has an effect on the uniform distribution of PS. And changing x_1 in $x = (x_1, x_2)$ can only result in a decision vector which is incomparable or equivalent to x. So x_1 affects the diversity of the MOP.

As a consequence, there is a huge difference of variables in MaOP, the variables affect the convergence which is called convergence-related variables and the variables affects the diversity which is called diversity-related variables. In terms of convergence-related variables, it is necessary to find the optimal value to make the PS close to or even distributed on the Pareto front. And in terms of diversity-related variables, it makes the Pareto solution reasonable and more uniform distribution (uniform distribution of diversity-related variables does not lead to uniform distribution of PS due to the MOP is not linear problem). Therefore, different optimize strategies for convergence-related variables and diversity-related variables are adopted, which are helpful to the reasonable distribution of the final PS.

In MOEA/DVA [24], decision variables are divided into position variable (homogeneity related variables), distance variable (convergence related variables), and mixed variable (mixed related variables). New individuals obtained by changing a dimension of the individual, and new individuals are non-dominated sorted. If the number of optimal front individuals is equal to the total number of individuals, this decision variable is position variable, while the number of optimal front individuals is 1 as distance variable. Otherwise it is mixed variable. In LMEA [39], decision variables are divided into two categories, uniformity-related variables and diversity-related variables, mixed variables were analyzed their impact on the diversity and convergence of the problem. Perturbing the values of each variable are normalized and the sample solutions generated, a line is generated to fit each set of normalized sample solutions, the angle between each fitted line and the normal line of hyperplane $f_1 + f_2 + \ldots + f_M = 1$ is calculated, a larger angle indicates the variables have a great impact on diversity, and a smaller angle means that the variables have more contribution to convergence.

Our proposed decision variable clustering method divides the variables into convergence-related variables and diversity-related variables initially. Algorithm 3 (line 4 in Algorithm 1) gives the detail of the proposed decision variable clustering method. An individual is randomly select and one variable is perturbed n times while the other variable is fixed, we can obtain a population of $n \times 1$ related to this variable. The non-dominated sorting algorithms are used to obtain non-dominated fronts and reflect the features of this variable. The sequential search strategy (ENS-SS) and the tree-based efficient non-dominated sorting (T-ENS) are designed for MaOP by Zhang *et.al.* [55], [56]. At last, the k-means clustering method is adopted to divide the decision variables into two clusters based on the features of each variable. Therefore, the variables in the cluster having a smaller non-dominated front are identified as diversity-related and the other are identified as convergence-related.

Algorithm 3 Variable Cluster (P)

Input: P(population);
Output:DV(diversity-related variables); CV (convergence-related variables);
1: *sample* $\leftarrow$ random select a individual from P;
2: $D \leftarrow$ the length of decision variables;
3: **for** $i = 1$ to D **do**
4: *Sample* $\leftarrow$ perturb the ith decision variable of *sample* for n times to generate a matrix *Sample*; /*n is a user-defined parameter(e.g. 20), *Sample* is a matrix of $n \times D$, which is only different in ith column */
5: *FrontNo*$[i]$ $\leftarrow$ nondominated sorting for *Sample* to obtain nondominated fronts; /**FrontNo*$[i]$ is the fronts of individual in *Sample**/
6 **end for**
7 $[CV,DV]$ $\leftarrow$ k-means is used to cluster the decision variables into two sets based on *FrontNo*;
8: **if** mean (*FrontNo*$[CV]$) > mean(*FrontNo*$[DV]$) **then**
9: $CV \leftarrow DV$;
10: $DV \leftarrow \{j = 1, \ldots, D | j \notin CV\}$;
11: **end if**

Table 1 shows the clustering results of different algorithm which is including MOTL/DVA, LMEA and MaOBSO on 16 decision variables with 5 objectives and 15 decision variables with 10 objectives about the DTLZ benchmark function set. It can be seen that MaOBSO has the same performance as LMEA. And the proposed decision variable clustering method of MaOBSO has lower computational complexity. Compared with MOEA/DVA, the clustering results are the same for DTLZ1$\sim$4 and DTLZ7 when the Pareto front is a spatial surface. For the benchmark function of DTLZ5 and DTLZ6, the PF are spatial curves, all of the algorithms have

TABLE 1. The decision variables clustering result of three algorithms on DTLZ benchmark function set.

Problem	Obj.	MOEA/DVA			LMEA		MaOBSO	
		Diversity	Convergence	both	Diversity	Convergence	Diversity	Convergence
DTLZ1	5	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$	Φ	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$
	10	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$	Φ	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$
DTLZ2	5	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$	Φ	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$
	10	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$	Φ	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$
DTLZ3	5	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$	Φ	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$
	10	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$	Φ	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$
DTLZ4	5	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$	Φ	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$
	10	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$	Φ	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$
DTLZ5	5	$\{x_1,x_2,x_3,x_4\}$	$\{x_7,x_9,\cdots,x_{11},x_{13},\cdots,x_{15}\}$	$\{x_5,x_6,x_8,x_{12},x_{16}\}$	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$
	10	$\{x_1,\cdots,x_9\}$	$\{x_{11}\}$	$\{x_{10},x_{12},\cdots,x_{15}\}$	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$
DTLZ6	5	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{11},x_{13},x_{15},x_{16}\}$	$\{x_{12},x_{14}\}$	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$
	10	$\{x_1,\cdots,x_9\}$	$\{x_{12},x_{13}\}$	$\{x_{10},x_{11},x_{14},x_{15}\}$	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$
DTLZ7	5	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$	Φ	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$	$\{x_1,x_2,x_3,x_4\}$	$\{x_5,\cdots,x_{16}\}$
	10	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$	Φ	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$	$\{x_1,\cdots,x_9\}$	$\{x_{10},\cdots,x_{15}\}$

same diversity variables, while partial convergence-related variables in MaOBSO are identified as mixed variables in MOEA/DVA.

D. DIVERGING OPERATION

1) VARIABLE INTERACTION ANALYSIS

In order to solve large-scale problem, it is necessary to decompose a large-scale optimization problem into several small-scale optimization problems. If the independence of variables is known in advance, variables can be divided into many low-dimensional variables subcomponents, so it is helpful to solve high-dimensional variables problems. The separability and nonseparability of decision variables in SOP can be definite as follows:

$f(x)$ is called a separable function if and only if each decision variable $x_i, i = 1, \ldots, n$ can be optimized independently.

$$\begin{aligned}\arg\min_{(x_1,\ldots,x_n)} f(x_1,\ldots,x_n) &= [\arg\min_{x_1} f(x_1,\ldots,x_i,\ldots,x),\ldots,\\ &\quad \arg\min_{x_i} f(x_1,\ldots,x_i,\ldots,x),\ldots,\\ &\quad \arg\min_{x_n} f(x_1,\ldots,x_i,\ldots,x)]. \end{aligned} \tag{3}$$

Otherwise, $f(x)$ is called a nonseparable function.

The variables of separable functions can be optimized separately, which can improve the efficiency of optimization. For convergence-related variables, it is necessary to find the optimal value to make the PS close to or even distributed on the Pareto front, so the optimization of convergence-related variables can be similar to single-objective optimization.

The proposed MaOBSO algorithm adopts multiple judgments to analyze the variable interaction of convergence-related variables. The definition of the interaction for two decision variables x_i and x_j as follows: if there exist x, a_1, a_2, b_1, and b_2 meeting

$$\begin{aligned} &f(\mathrm{x})|_{x_i=a_2,x_j=b_1} < f(\mathrm{x})|_{x_i=a_1,x_j=b_1} \wedge\\ &f(\mathrm{x})|_{x_i=a_2,x_j=b_2} > f(\mathrm{x})|_{x_i=a_1,x_j=b2}\\ &f(\mathrm{x})|_{x_i=a_2,x_j=b_1} = f(x_1,\ldots,x_{i-1},a_2,\ldots,x_{j-1},b_1,\ldots,x_n) \end{aligned} \tag{4}$$

Algorithm 4 (line 5 in Algorithm 1) is the interaction analysis process of convergence-related variables. Firstly, an empty set *subCVs* of interacted variable subgroups is initialized. And then, the convergence-related variables in *CV* are assigned to different subgroups based on the formula (4) to judge the interaction [39]. If a variable has interaction with at least one existing variable in *subCVs*, the variable interaction analysis will no longer be conducted, which improve computational efficiency. And the two variables are putted into the same subgroup. Otherwise, it is putted a new subgroup. Each convergence-related variable is putted into a subgroup by the process above. To be noted that if the convergence-related variables are fully separable, the number of subgroups is $|CV|$; if the variables are fully non-separable, the number of subgroups is one.

2) VARIABLE OPTIMIZATION

When the clustering procedure and variable interaction analysis are completed, MaOBSO starts to optimize different categories variables. The convergence-related variables are optimized by a convergence optimization method as detailed in Algorithm 5 (line 8 in Algorithm 1). While the diversity-related variables by diversity optimization method detailed in Algorithm 6 (line 10 in Algorithm 1).

In the convergence optimization method, a new individual is generated by binary crossover (*SBX*) between the parent

Algorithm 4 Interaction Analysis(P, CV)

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

$subCVs \leftarrow \Phi$;
for all the $v \in CV$ **do**
  $CorSet \leftarrow \Phi$;
  for all the $Group \in subCVs$ **do**
    for all the $u \in Group$ **do**
      $flag \leftarrow$ **false**;
      for $i = 1$ to $nCor$ **do**
            /*$nCor$ is a user-defined parameter */
        $p \leftarrow$ Randomly select an individual from P;
        if v is interacted with u in p **then**
          $flag \leftarrow$ **true**;
          $CorSet \leftarrow CorSet \cup \{Group\}$;
          break;
          end if
        end for
        if $flag$ **then**
          break;
          end if
      end for
    end for
  if $CorSet == \Phi$ **then**
    $subCVs \leftarrow subCVs \cup \{\{v\}\}$;
  else
    $subCVs \leftarrow subCVs / CorSet$;
    $Group \leftarrow$ all variables in $CorSet$ and v;
    $subCVs \leftarrow subCVs \cup \{Group\}$;
  end if
end for

Algorithm 5 ConverOptimi (P, $subCVs$)

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

for $i = 1$ to N **do**/*N is population size */
  $[p1, p2] \leftarrow$ select parents by roulette wheel based on the PBI of P;
  for $j = 1$ to C **do** /*C is $subCVs$ size */
  for $d = 1$ to B **do** /*B is $subCVs[j]$ size */
    $p' \leftarrow$ SBX($p1[d]p2[d]$);
      $offspring \leftarrow s[d] = p'$;
          /*s is the ith individual of P*/
      end for
    $P \leftarrow$ the better one is retained to update P by comparing the PBI of s and $offspring$;
  end for
end for

populations which are selected by roulette according to the fittest survival principle in evolutionary theory. As a promising performance for many-objective optimization reported in [31]. PBI, as a variant of the normal-boundary intersection method [22], is adopted as the standard, where equality constraint is handled by a penalty function. The algorithm is defined as

$$\min \ g^{pbi}\left(x|w, z^{\min}\right) = d_1 + \theta d_2 \tag{5}$$

where:

$$d_1 = \frac{\left\| \left(f(x) - z^{\min}\right)^T w \right\|}{\|w\|} \tag{6}$$

$$d_2 = \left\| f(x) - \left(z^{\min} + d_1 \frac{w}{\|w\|}\right) \right\| \tag{7}$$

where d_1 is the distance from point f of all objectives to point of the minimum of all objectives, which can measure the convergence of x; and d_2 represents the distance between point f and points w, which can measure the diversity of x. The relationship between $f(x)$, d_1 and d_2 is shown in Fig.7.

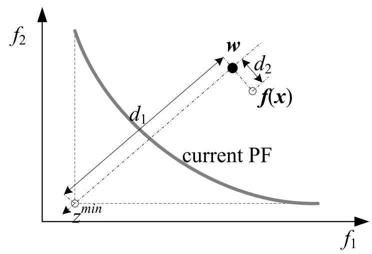

FIGURE 7. The relationship between $f(x)$, d_1, and d_2.

θ is a user-defined penalty parameter which can be used to balance the convergence and diversity of the solution in formula (5). Therefore, PBI can measure the pros and cons of a solution more comprehensively with a single value. The smaller the PBI value, the better the individual. In addition, as described in the previous section, after the independence of the convergence-related variables is judged, each independent variable is optimized one by one.

Algorithm 5 (line 8 in Algorithm 1) is the convergence optimization process of convergence-related variables. When two parents by roulette wheel based on the PBI of P are selected, the new variables in $subCVs$ are generated by SBX between the parents which are selected by roulette wheel, so a new individual $offspring$ is generated and P is updated by comparing the PBI of parent and $offspring$.

The objective of the diversity optimization method is to generate a new individual which has the variation of single individual so as to maintain the diversity of the population. Here we draw on the idea of generating as many individuals as possible without constraints in the brain storm optimization algorithm. Algorithm 6 (line 10 in Algorithm 1) describes the diversity optimization process of diversity-related variables in detail. The process is listed in the following.

Firstly, one or several existing individuals randomly selected by the probability P_o and the probability P_c in the

Algorithm 6 DiverOptimi (P, P_o, P_c, DV, w, *cluster*)

Input: P (current population), P_o(Probability of selecting an individual with an individual), P_c(Probability of selecting an individual with a cluster center),DV(diversity-related variables), w(reference point/vector),*cluster*(cluster of each individual)

Output: P (new population)

1: **for** $i = 1$ to N **do**/*N is population size */
2: **if** random < P_o **then**
3: **if** random < P_c **then**
4: *pselect* ← the corner of the cluster of s; /*s is the ith individual of P*/
5: **else**
6: *pselect* ← randomly select an individual in the cluster of s;
7: **end if**
8: **else**
9: *pselect* ← randomly select two individuals from P and weighted summation;
10: **end if**
11: *offspring*[i] ← mutation of *p select*;
12: **end for**
13: *POP* ← $P \cup$ *offspring*;
14: P ← Selecting the *POP* based on the reference point to generate new P

corner point. Secondly, the new individuals are generated by adding noise to the DV of the randomly select individuals with mutation operation. In original brain storm optimization algorithm, the mutation operation adopts Gaussian mutation. During the process of searching, it is central to balance the ability of the exploration and the exploitation. Gaussian variation cannot make full use of information in the current population and the operation complexity is high. Therefore, we adopt differential evolution mutation to generate new individual based on the selected individual(s), and setting the open probability p_r to enhance the diversity of the algorithm. In section IV, we give the simulation result comparison between the two mutation operations. The update process of the population is similar to the reference point assignment process, which was given in Algorithms 2. The angles of all points are calculated and selected one by one.

Moreover, the convergence optimization index is set in the MaOBSO algorithm for avoiding the unnecessary search. When the mean variance of the convergence-related variables is less than a certain threshold, the convergence optimization process will stop. The certain threshold is set to 0.01 according to the empiric value in this paper.

IV. SIMLATION EXPERIMENTS AND ANALYSIS

In this section, five state-of-the-art MOEAs for many-objective optimization, namely MOEA/DD [23], NSGA-III [31], LMEA [39], AR-MOEA [21], and RVEA [32] is simulated to compare with our proposed MaOBSO algorithm. The performance of MaOBSO in solving MaOPs is verified by different kinds of benchmark function.

In the following subsections, the benchmark functions and the relating performance metrics for comparison analysis is introduced firstly. And then the parameter settings are listed. The simulation results with different algorithms are discussed in details and the results are analyzed lastly.

A. BENCHAMARK FUNCTION

The classic DTLZ benchmark function set, which was proposed by Deb in 1999 [45], and the MaF test problems, which was proposed in [36], are selected to valid the performance of the given algorithms in this paper. For each DTLZ instance, the number of decision variables is set to $D = M + K$-1, where M is the objective number, $K = 5$ is used for DTLZ1, $K = 10$ is used for DTLZ2-DTLZ6, and $K = 20$ is used for DTLZ7. For MaF1-MaF7 instance, the number of decision variables is set to $D = M + K - 1$, $K = 10$ is used for MaF1-MaF6, and $K = 20$ is used for MaF7. For MaF8 and MaF9 instance, the number of decision variable is D=2. For MaF10-MaF12 instance, the number of decision variables is set to $D = K + L$, with K denoting the number of position variables and L denoting the number of distance variables. And the parameter settings are $K = M - 1$, $L = 10$. For MaF13 instance, the number of decision variable is D=5. For MaF14 and MaF15 instance, the number of decision variables is set to $D = 20 \times M$. In our experiments, the number of objectives varies from 3 to 10, i.e., $M \in \{3, 5, 8, 10\}$.

B. PERFORMANCE METRICS

The Inverted Generational Distance (IGD) [57] is used to evaluate the convergence performance and diversity performance of the algorithm, which is defined as:

$$IGD\left(S, P^{*}\right) = \frac{\sum_{x^{*} \in P^{*}} dist\left(x^{*}, S\right)}{|P^{*}|} \tag{8}$$

where the PS of S is obtained by the algorithm, $P*$ is the point on the true Pareto front of the benchmark function. In theory, the scale of $P*$ is infinite. But the true Pareto front is represented by a finite set of points in practice. The IGD value is the average distance between a point on the true Pareto front and the PS by MaOBSO algorithm. The larger is the distance between the true Pareto front and PS by MaOBSO algorithm, the big is the IGD value, so the IGD value reflect the convergence performance of MaOBSO. If PS is inhomogeneous, that means no solutions in local area. The parameter IGD is becoming larger as the shortest distance from the Pareto front to this area increases. Therefore, IGD can also reflect the diversity of algorithms. According to the analysis above, the smaller the IGD value, the better the overall performance of MaOBSO algorithm.

C. PARAMETER SETTING

The parameters of the different kinds of algorithms, especially the MaOBSO algorithm proposed in this paper, are set in this subsection. For comparison, some parameters such

TABLE 2. Setting of population size in MaOBSO, where *p* 1 and *p* 2 are parameters controlling the numbers of reference points.

M	$(p1, p2)$	Population size
3	(13,0)	105
5	(5,0)	126
8	(2,2)	72
10	(2,1)	65

as the population size and termination condition are set as the same value as the comparative algorithms suggested in the corresponding references. Some parameters such as the objective number are set differently, according to the complexity of the problem. And some parameters such as the probability are set according to our experience. Therefore, we present the parameters of the six algorithms, which are summarized as follows.

1) POPULATION SIZE

The population size of MaOBSO, MOEA/DD, NSGA-III, LMEA, RVEA is equal to the number of the reference points. As recommended in [31] and [39], for objectives with $M \geq 8$, the two-layer reference points generation method is adopted. The setting of the population size in MaOBSO is presented in Table 2, where $p1$ and $p2$ are the number of divisions in the boundary layer and inside layer respectively. The other algorithms adopt the same population size settings with the same number of objectives.

2) TERMINATION CONDITION

The termination condition is the maximum number of evaluations. The maximum number of evaluations is set to 100000 for all considered algorithms.

3) CROSSOVER AND MUTATION

For the *SBX*, the distribution index is set to $\eta_c = 30$ and the crossover probability is $p_c = 1.0$. For the polynomial mutation, the mutation probability is $p_m = 1/\mathrm{D}$ and its distribution index is $\eta_m = 20$. For DE mutation, the mutation probability is $p_r = 0.6$.

4) THE OTHER PARAMETERS

The penalty parameter θ in PBI is set to 5 for all algorithms. For MOEA/DD, the neighborhood size is T=20, and the neighborhood selection probability is $\delta = 0.9$. 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. For MOEA/DVA, the number of interaction analysis and the number of control property analysis are set to *NIA*=6 and *NCA*=50. For RVEA, the index of control the change rate of the penalty function and the frequency of the reference point adaptation are set to $\alpha = 2$ and $f_r = 0.1$ respectively. For MaOBSO, the probability of selecting an individual with an individual is $P_o = 0.6$. The probability of selecting an individual with a cluster center is $P_c = 0.3$.

D. COMPARISON WITH FIVE COMPETITIVE MAOEAS

1) COMPARISON RESULTS ON DTLZ

Table 3 shows the comparison results in the mean IGD value of MaOBSO and the other five algorithms after 30 independent runs, where the best result on each test instance is shown in a gray background. From the second last row of Table 3, we can see that LMEA obtains the best results in 10 out of 28 cases. MaOBSO and MaOBSO-GS (MaOBSO adopt Gaussian mutation) respectively obtain the best results in 8 and 7 case, while MOEA/DD, NSGA-III and RVEA are unable to perform best on any test problem and AR-MOEA performed best in 3 case, which proves the advantage of MaOBSO on DTLZ1-DTLZ7 problems. The last row of Table 3 shows that the rank of MaOBSO is better than MOEA/DD, NSGA-III, LMEA, AR-MOEA, RVEA and MaOBSO-GS. Therefore, it is concluded that MaOBSO showed a superior performance over its six competitors on DTLZ1-DTLZ7.

As can be seen from the table, the overall performance of MaOBSO is relatively good. Compared with MOEA/DD, NSGA-III, LMEA, AR-MOEA, RVEA, MaOBSO-GS, and MaOBSO algorithms have the same order of magnitude, the performance of MaOBSO is better than the other algorithms. In the same benchmark function, the IGD value increases with the increase of the number of objectives, which indicates that the challenge of optimization by increase of the number of objectives. With the increasing of the number of objectives, the population is more distributed discretely in the objective space. This makes the diversity of population more difficult to be maintained. In the different benchmark function, DTLZ1-DTLZ4 owes more standardized PF, most algorithm shows good performance. But for DTLZ5-DTLZ7, algorithms are more influenced by the number of objectives.

From the table, we can get that the performance of these algorithms are similar for DTLZ1 and DTLZ2. Owing to DTLZ3 have a large number of local extreme points, the performance of NSGA-III decreases significantly as the number of objectives increases. Compared with other algorithms, MaOBSO shows better ability in maintaining the diversity for the function DTLZ4. For DTLZ5 to DTLZ7, the performance of MOEA/DD, NSGA-III, LMEA, AR-MOEA, RVEA, MaOBSO-GS and MaOBSO are better and stable because all of them have different retention mechanism, which can get more uniform frontier. In terms of algorithm stability, the standard deviation of MaOBSO is small, which indicates that the algorithm has good stability.

For further observation, Fig.8 shows the non-dominated solutions by the seven algorithms among 30 runs in the objective space on 10-objective. For DTLZ1 and DTLZ2, the six algorithms have good performance. But NSGA-III is poor in terms of diversity. AR-MOEA and RVEA are bad in terms of diversity. For DTLZ3, NSGA-III is poor in

TABLE 3. The Comparison of IGD between MaOBSO several algorithms on DTLZ.

function	M	MOEA/DD	NSGA-III	LMEA	AR-MOEA	RVEA	MaOBSO-GS	MaOBSO
DTLZ1	3	2.0559e-2(2.87e-6)$_{(3)}$	2.0566e-2(9.53e-6)$_{(5)}$	2.0943e-2(2.28e-4)$_{(7)}$	2.0607e-2(1.78e-5)$_{(6)}$	2.0561e-2(8.00e-6)$_{(4)}$	2.0204e-2(2.58e-4)$_{(2)}$	**1.8988e-2(7.93e-6)$_{(1)}$**
	5	6.8087e-2(6.91e-5)$_{(5)}$	6.8113e-2(7.78e-5)$_{(6)}$	6.6469e-2(6.63e-4)$_{(3)}$	6.8113e-2(4.97e-5)$_{(6)}$	6.8081e-2(2.76e-5)$_{(4)}$	**6.0126e-2(5.91e-4)$_{(1)}$**	6.3105e-2(6.50e-5)$_{(2)}$
	8	1.0850e-1(1.96e-4)$_{(2)}$	1.0878e-1(1.71e-3)$_{(3)}$	1.1506e-1(2.56e-3)$_{(5)}$	**1.0844e-1(1.96e-4)$_{(1)}$**	1.0947e-1(2.56e-3)$_{(4)}$	1.1756e-1(1.16e-3)$_{(6)}$	1.1931e-1(6.58e-4)$_{(7)}$
	10	1.5336e-1(6.44e-4)$_{(5)}$	1.8203e-1(4.49e-2)$_{(6)}$	**1.3449e-1(6.11e-3)$_{(1)}$**	1.5561e-1(5.18e-3)$_{(6)}$	1.5033e-1(3.41e-3)$_{(4)}$	1.3800e-1(1.31e-3)$_{(2)}$	1.4306e-1(7.89e-4)$_{(3)}$
DTLZ2	3	5.4464e-2(1.99e-7)$_{(4)}$	5.4465e-2(1.65e-6)$_{(5)}$	5.3739e-2(4.69e-4)$_{(3)}$	5.4464e-2(1.22e-6)$_{(4)}$	5.4464e-2(4.15e-7)$_{(4)}$	5.2112e-2(4.34e-4)$_{(2)}$	**5.0469e-2(7.16e-5)$_{(1)}$**
	5	2.1221e-1(3.43e-6)$_{(4)}$	2.1221e-1(5.98e-6)$_{(4)}$	2.0430e-1(1.62e-3)$_{(3)}$	2.1221e-1(2.40e-5)$_{(4)}$	2.1222e-1(8.60e-6)$_{(7)}$	**1.8937e-1(1.25e-3)$_{(1)}$**	1.9425e-1(4.29e-4)$_{(2)}$
	8	3.8693e-1(1.13e-5)$_{(3)}$	4.1907e-1(6.62e-2)$_{(6)}$	3.8290e-1(2.51e-3)$_{(2)}$	**3.8147e-1(7.57e-4)$_{(1)}$**	3.8708e-1(1.13e-4)$_{(4)}$	4.0553e-1(2.66e-3)$_{(5)}$	4.4655e-1(3.18e-2)$_{(7)}$
	10	5.0027e-1(1.07e-4)$_{(5)}$	5.6664e-1(7.69e-2)$_{(7)}$	**4.7109e-1(1.76e-3)$_{(1)}$**	4.9941e-1(5.59e-3)$_{(4)}$	5.0630e-1(1.47e-2)$_{(6)}$	4.9767e-1(2.85e-3)$_{(3)}$	4.7322e-1(1.42e-3)$_{(2)}$
DTLZ3	3	5.4607e-2(1.64e-4)$_{(4)}$	5.4784e-2(3.36e-4)$_{(7)}$	5.4159e-2(7.59e-4)$_{(3)}$	5.4626e-2(1.65e-4)$_{(5)}$	5.4728e-2(6.07e-4)$_{(6)}$	5.1970e-2(3.94e-4)$_{(2)}$	**5.0465e-2(7.56e-5)$_{(1)}$**
	5	2.1261e-1(3.00e-4)$_{(4)}$	2.1334e-1(9.13e-4)$_{(6)}$	2.0529e-1(1.61e-3)$_{(3)}$	2.1393e-1(1.62e-3)$_{(7)}$	2.1284e-1(7.59e-4)$_{(5)}$	**1.8896e-1(1.05e-3)$_{(1)}$**	1.9422e-1(3.30e-4)$_{(2)}$
	8	3.8943e-1(1.65e-3)$_{(3)}$	1.1330e+0(1.21e+0)$_{(7)}$	**3.8309e-1(2.33e-3)$_{(1)}$**	4.0062e-1(1.13e-2)$_{(4)}$	3.8887e-1(1.71e-3)$_{(2)}$	4.0537e-1(2.49e-3)$_{(5)}$	4.1955e-1(9.87e-4)$_{(6)}$
	10	5.0210e-1(1.44e-3)$_{(4)}$	3.4232e+0(2.43e+0)$_{(7)}$	4.7692e-1(3.13e-2)$_{(2)}$	5.3647e-1(1.23e-2)$_{(6)}$	5.1413e-1(2.66e-2)$_{(5)}$	4.9591e-1(4.34e-3)$_{(3)}$	**4.7353e-1(1.16e-3)$_{(1)}$**
DTLZ4	3	7.0779e-2(8.94e-2)$_{(4)}$	1.3569e-1(1.85e-1)$_{(6)}$	7.2436e-2(5.75e-2)$_{(5)}$	3.4627e-1(2.51e-1)$_{(7)}$	5.4464e-2(6.53e-7)$_{(3)}$	5.3029e-2(8.86e-4)$_{(2)}$	**5.0776e-2(1.86e-4)$_{(1)}$**
	5	2.2704e-1(5.64e-2)$_{(4)}$	2.7401e-1(1.04e-1)$_{(6)}$	2.7677e-1(1.32e-1)$_{(7)}$	2.2603e-1(5.24e-2)$_{(2)}$	2.4085e-1(7.42e-2)$_{(5)}$	2.4466e-1(9.52e-3)$_{(3)}$	**2.2464e-1(1.18e-2)$_{(1)}$**
	8	4.1241e-1(4.32e-2)$_{(3)}$	5.1010e-1(1.13e-1)$_{(7)}$	5.0814e-1(1.31e-1)$_{(6)}$	**3.8671e-1(1.61e-4)$_{(1)}$**	4.6015e-1(9.46e-2)$_{(5)}$	4.2477e-1(4.24e-3)$_{(4)}$	4.0265e-1(1.04e-3)$_{(2)}$
	10	5.3488e-1(5.48e-2)$_{(4)}$	6.2561e-1(9.46e-2)$_{(7)}$	5.3721e-1(9.28e-2)$_{(5)}$	5.0565e-1(4.31e-3)$_{(2)}$	5.4121e-1(4.86e-2)$_{(6)}$	5.0745e-1(4.61e-3)$_{(3)}$	**4.9896e-1(1.89e-3)$_{(1)}$**
DTLZ5	3	3.1367e-2(7.90e-4)$_{(5)}$	1.2129e-2(1.87e-3)$_{(4)}$	4.6623e-3(1.70e-4)$_{(3)}$	5.3671e-2(1.46e-4)$_{(6)}$	6.2711e-2(2.71e-3)$_{(7)}$	**4.2339e-3(9.44e-5)$_{(1)}$**	4.4031e-3(1.17e-4)$_{(2)}$
	5	8.7960e-2(1.01e-2)$_{(5)}$	1.1260e-1(5.21e-2)$_{(6)}$	5.5556e-3(3.47e-4)$_{(2)}$	7.3074e-2(9.64e-3)$_{(4)}$	2.3237e-1(4.29e-2)$_{(7)}$	1.1656e-2(4.11e-2)$_{(3)}$	**4.2074e-3(1.22e-4)$_{(1)}$**
	8	1.9294e-1(3.15e-2)$_{(5)}$	3.3088e-1(1.02e-1)$_{(6)}$	**6.8352e-3(5.58e-4)$_{(1)}$**	1.0278e-1(2.68e-2)$_{(4)}$	6.0644e-1(1.65e-1)$_{(7)}$	9.1309e-3(5.40e-4)$_{(3)}$	8.9492e-3(4.23e-4)$_{(2)}$
	10	1.5320e-1(1.60e-3)$_{(5)}$	3.1655e-1(1.07e-1)$_{(6)}$	**7.5308e-3(5.34e-4)$_{(1)}$**	1.1909e-1(2.46e-2)$_{(4)}$	3.9476e-1(1.74e-1)$_{(7)}$	3.8767e-2(1.53e-1)$_{(3)}$	1.0970e-2(8.61e-4)$_{(2)}$
DTLZ6	3	3.3817e-2(5.48e-4)$_{(6)}$	1.9818e-2(2.51e-3)$_{(5)}$	4.5812e-3(1.31e-4)$_{(3)}$	5.1079e-3(3.19e-4)$_{(4)}$	9.1442e-2(2.21e-2)$_{(7)}$	**4.2158e-3(6.64e-5)$_{(1)}$**	4.2717e-3(6.34e-5)$_{(2)}$
	5	9.1180e-2(1.38e-2)$_{(5)}$	2.6499e-1(6.12e-2)$_{(7)}$	**5.0916e-3(3.94e-4)$_{(1)}$**	7.8842e-2(1.15e-2)$_{(4)}$	2.2645e-1(5.17e-2)$_{(6)}$	7.2129e-3(5.91e-3)$_{(2)}$	2.2930e-2(4.45e-2)$_{(3)}$
	8	1.8747e-1(3.05e-2)$_{(3)}$	1.0875e+0(7.60e-1)$_{(5)}$	**5.7316e-3(8.31e-4)$_{(1)}$**	1.2643e-1(3.84e-2)$_{(2)}$	3.0836e-1(1.11e-1)$_{(4)}$	1.3480e+0(3.43e+0)$_{(6)}$	1.7571e+0(2.96e+0)$_{(7)}$
	10	1.5384e-1(2.84e-4)$_{(4)}$	3.0677e+0(1.30e+0)$_{(6)}$	**6.3166e-3(1.38e-3)$_{(1)}$**	1.1092e-1(2.72e-2)$_{(2)}$	1.9113e-1(3.51e-2)$_{(3)}$	4.6629e+0(5.04e+0)$_{(7)}$	3.0427e+0(3.81e+0)$_{(5)}$
DTLZ7	3	5.2702e-1(2.50e-1)$_{(7)}$	7.6846e-2(3.46e-3)$_{(4)}$	5.8590e-2(1.11e-3)$_{(2)}$	9.5123e-2(8.88e-2)$_{(5)}$	1.0535e-1(1.43e-3)$_{(6)}$	**5.6869e-2(9.49e-4)$_{(1)}$**	5.8870e-2(1.00e-3)$_{(3)}$
	5	2.7686e+0(5.87e-1)$_{(7)}$	3.7384e-1(1.70e-2)$_{(5)}$	3.4907e-1(6.68e-3)$_{(3)}$	3.5122e-1(6.17e-3)$_{(4)}$	5.0684e-1(4.41e-3)$_{(6)}$	**3.2012e-1(1.28e-2)$_{(1)}$**	3.2916e-1(1.25e-2)$_{(2)}$
	8	1.9071e+0(5.04e-1)$_{(7)}$	9.4230e-1(5.48e-2)$_{(3)}$	**7.5623e-1(2.17e-2)$_{(1)}$**	1.7642e+0(9.92e-2)$_{(6)}$	1.7343e+0(2.41e-1)$_{(5)}$	8.6037e-1(2.74e-2)$_{(2)}$	9.5301e-1(3.77e-2)$_{(4)}$
	10	2.4667e+0(3.56e-1)$_{(5)}$	1.6996e+0(2.14e-1)$_{(4)}$	**1.0680e+0(2.98e-2)$_{(1)}$**	3.4084e+0(2.43e-1)$_{(7)}$	3.1222e+0(5.37e-1)$_{(6)}$	1.2544e+0(8.27e-2)$_{(2)}$	1.2692e+0(4.97e-2)$_{(3)}$
best/all	-	0/28	0/28	10/28	3/28	0/28	7/28	8/28
rank	-	125(4)	156(6)	77(2)	118(3)	145(5)	77(2)	76(1)

terms of both convergence and diversity. The six algorithms have good convergence performance for DTLZ4. From the perspective of diversity, the NSGA-III and LMEA have poor diversity performance. The distribution breadth and uniformity of the optimal solution set are not ideal in the objective space. And the most of Pareto optimal solutions are distributed in the border. In addition, some objectives are ''missing'', which make the optimal solution set is difficultly distributed on the whole non-dominant frontier. However, due to MaOBSO-GS and MaOBSO ability to generate new individuals without constraint, the performance of MaOBSO-GS and MaOBSO are best, while MOEA/DD, AR-MOEA and RVEA showed slightly worse performance. For DTLZ5 and DTLZ6 with degenerate frontier, the performance of LMEA than the other algorithms, which reflect the real frontier completely and uniformly. MaOBSO-GS and MaOBSO has good convergence but poor diversity, and the other algorithms have poor convergence and diversity. For DTLZ7 with discontinuous frontier, the performance of LMEA is also the best. NSGA-III, MaOBSO-GS and MaOBSO also perform better, while the diversity of another algorithm is poor.

To sum up, compared with several current popular algorithms, MaOBSO shows good competitiveness. For the many-objective optimization problem with regular frontier, MaOBSO performs better. For the problem with irregular frontier, the proposed algorithm is slightly inferior to LMEA in diversity, but the overall performance is better than the other algorithms.

2) COMPARISON RESULTS ON MAF

Table 4 shows the mean IGD comparison results of MaOBSO and the other algorithms on MaF test problems, where the best result on each test instance is shown in a gray background. From the second last row of Table 4, MaOBSO shows a superior performance, it's the best results in 15 out of 45 cases. MOEA/DD, NSGA-III, LMEA, AR-MOEA, and RVEA are respectively best in 3, 1, 14, 6, and 6 cases. From the last row of Table 4, the overall performance of AR-MOEA is best, LMEA is second, and MaOBSO is better than MOEA/DD, NSGA-III and RVEA. Therefore, MaOBSO showed competitive performance over its five competitors in most MaF problems.

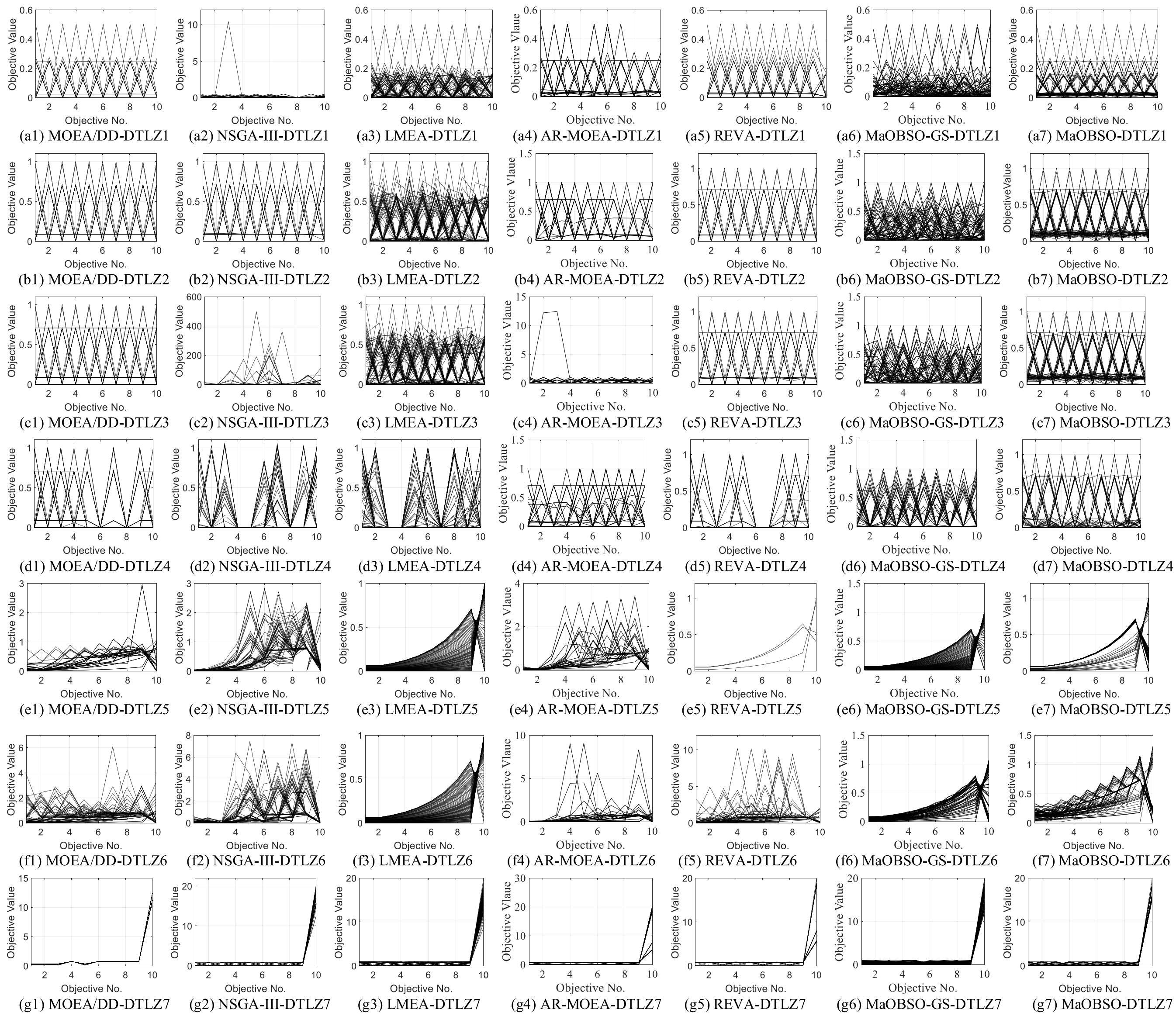

FIGURE 8. Results of several algorithms under the 10 objectives DTLZ benchmark function set.

MaF1 is gained by inverting the PF of DTLZ1, LMEA shows the best performance on all the objectives. MaF2 is modification DTLZ2 by increasing the difficulty of convergence, and all the objective have to be optimized simultaneously in order to reach the true PF. The proposed algorithm MaOBSO perform best in all case of MaF2. MaF3 has a convex PF and there are a large number of local fronts. RVEA can solve the problem well. MaF4's shape by inverting the shape of DTLZ3, and MaF5 has a badly-scaled PF and a highly biased distribution, MaOBSO solve it very well. Regarding MaF6 with degenerate PF, AR-MOEA and LMEA gain the best results in the case of 5, 8 and 10 objectives, respectively. MaF7 and MaF11 have a characteristic with a disconnected PF, MaOBSO shows a competitive performance, as it obtains the best results in MaF7 of 5 objective and MaF11 of 5 and 8 objectives, while it is outperformed by LMEA in MaF7 with 8 and 10 objectives and AR-MOEA in MaF11 with 10 objectives. One important characteristic of MaF8 is its Pareto optimal region in the decision space is typically a 2D manifold, LMEA obtain the best result in the instance with all objectives. Regarding MaF9, it is characterized with the points in the regular polygon (including the boundaries) and their objective images are similar in the sense of Euclidean geometry, AR-MOEA is best in the case of 5 objective, while LMEA perform best in the cases of 8 and 10 objectives. For MaF10 with PF of complicated mixed geometries, RVEA and AR-MOEA obtain the best results in the case of 5 and 10 objectives, respectively. on MaF12 which has scaled concave PF together with complicated fitness landscapes, LMEA can best solve this problem in all objectives. Concerning

TABLE 4. The Comparison of IGD between MaOBSO several algorithms on MaF.

function	M	MOEA/DD	NSGA-III	LMEA	AR-MOEA	RVEA	MaOBSO
MaF1	5	2.7344e-1 (2.62e-2)$_{(4)}$	2.4541e-1 (2.89e-2)$_{(3)}$	**1.3354e-1 (8.25e-4)$_{(1)}$**	1.5184e-1 (2.96e-3)$_{(2)}$	3.6166e-1 (9.96e-2)$_{(5)}$	1.0632e+1 (3.61e-3)$_{(6)}$
	8	4.4529e-1 (4.24e-2)$_{(4)}$	2.9570e-1 (7.80e-3)$_{(3)}$	**2.3235e-1 (1.98e-3)$_{(1)}$**	2.6704e-1 (1.24e-3)$_{(2)}$	6.7367e-1 (8.31e-2)$_{(5)}$	3.7582e+0 (3.63e-3)$_{(6)}$
	10	6.3952e-1 (5.13e-2)$_{(4)}$	3.2592e-1 (1.15e-2)$_{(3)}$	**2.8007e-1 (4.38e-3)$_{(1)}$**	2.9848e-1 (9.09e-3)$_{(2)}$	6.7172e-1 (7.58e-2)$_{(5)}$	3.1099e+0 (4.72e-3)$_{(6)}$
MaF2	5	1.5540e-1 (2.77e-3)$_{(6)}$	1.4465e-1 (4.15e-3)$_{(5)}$	1.2200e-1 (5.29e-2)$_{(3)}$	1.2183e-1 (1.56e-3)$_{(2)}$	1.4417e-1 (1.41e-3)$_{(4)}$	**9.9560e-2 (1.41e-3)$_{(1)}$**
	8	2.5340e-1 (8.69e-2)$_{(3)}$	2.6239e-1 (4.05e-2)$_{(4)}$	3.8788e-1 (1.60e-1)$_{(5)}$	2.0035e-1 (3.55e-3)$_{(2)}$	4.8649e-1 (1.55e-1)$_{(6)}$	**1.8888e-1 (3.38e-3)$_{(1)}$**
	10	3.7669e-1 (2.91e-2)$_{(4)}$	3.2954e-1 (5.20e-2)$_{(3)}$	4.2823e-1 (1.60e-1)$_{(5)}$	2.4359e-1 (1.08e-2)$_{(2)}$	6.8246e-1 (1.40e-1)$_{(6)}$	**2.0622e-1 (3.15e-3)$_{(1)}$**
MaF3	5	1.1697e-1 (2.18e-3)$_{(6)}$	1.0949e-1 (3.97e-2)$_{(4)}$	1.1175e-1 (1.49e-2)$_{(5)}$	9.8567e-2 (1.57e-3)$_{(2)}$	**9.1608e-2 (2.87e-2)$_{(1)}$**	1.0599e-1 (1.01e-2)$_{(3)}$
	8	1.2874e-1 (3.03e-3)$_{(2)}$	1.5570e+0 (4.74e+0)$_{(6)}$	1.3468e-1 (1.47e-2)$_{(3)}$	1.3498e-1 (8.95e-3)$_{(4)}$	**1.1284e-1 (6.91e-3)$_{(1)}$**	1.3896e-1 (8.03e-3)$_{(5)}$
	10	1.3742e-1 (1.09e-1)$_{(3)}$	1.0370e+2 (2.97e+2)$_{(5)}$	7.9170e+3 (4.34e-4)$_{(6)}$	1.1213e-1 (8.19e-3)$_{(2)}$	**1.0270e-1 (1.05e-2)$_{(1)}$**	1.7793e-1 (1.98e-1)$_{(4)}$
MaF4	5	7.5650e+0 (5.37e-1)$_{(6)}$	3.6962e+0 (4.21e-1)$_{(4)}$	2.2969e+0 (5.73e-2)$_{(2)}$	2.8500e+0 (1.41e-1)$_{(3)}$	4.9743e+0 (1.23e+0)$_{(5)}$	**2.0440e+0 (3.62e-2)$_{(1)}$**
	8	1.0314e+2 (2.99e+0)$_{(6)}$	3.4757e+1 (2.14e+0)$_{(4)}$	1.8362e+1 (1.08e+0)$_{(2)}$	3.0956e+1 (2.70e+0)$_{(3)}$	6.4620e+1 (1.49e+1)$_{(5)}$	**1.7335e+1 (5.56e-1)$_{(1)}$**
	10	4.7617e+2(1.64e+1)$_{(6)}$	1.4876e+2 (1.84e+1)$_{(4)}$	6.6462e+1 (3.99e+0)$_{(2)}$	1.4243e+2 (1.11e+1)$_{(3)}$	2.1997e+2 (6.62e+1)$_{(5)}$	**6.4626e+1 (3.57e+0)$_{(1)}$**
MaF5	5	5.9982e+0 (1.36e+0)$_{(5)}$	2.6072e+0 (7.72e-1)$_{(3)}$	8.9777e+0 (6.15e+0)$_{(6)}$	2.5412e+0 (6.65e-1)$_{(2)}$	2.8211e+0 (7.58e-1)$_{(4)}$	**2.3331e+0 (8.44e-2)$_{(1)}$**
	8	7.6852e+1 (5.25+0)$_{(6)}$	2.7716e+1 (1.64e+0)$_{(2)}$	6.6298e+1 (3.28e+1)$_{(5)}$	2.9790e+1 (2.57e+0)$_{(4)}$	2.9384e+1 (4.11e+0)$_{(3)}$	**2.2763e+1(1.90e+0)$_{(1)}$**
	10	2.9461e+2 (1.23e+1)$_{(6)}$	1.3340e+2 (1.28e+1)$_{(2)}$	1.7857e+2 (1.22e+2)$_{(5)}$	1.6437e+2 (4.82e+0)$_{(4)}$	1.3870e+2 (1.50e+1)$_{(3)}$	**1.0023e+2 (1.27e+1)$_{(1)}$**
MaF6	5	8.4712e-2 (4.89e-3)$_{(5)}$	6.1344e-2 (2.43e-2)$_{(3)}$	5.4320e-3 (3.30e-4)$_{(2)}$	**5.1073e-3 (5.54e-5)$_{(1)}$**	8.1072e-2 (1.38e-2)$_{(4)}$	6.2498e-1 (2.46e-6)$_{(6)}$
	8	1.1689e-1 (3.34e-3)$_{(3)}$	1.5046e-1 (1.57e-1)$_{(4)}$	6.7551e-3 (5.82e-4)$_{(2)}$	**6.3809e-3 (4.67e-4)$_{(1)}$**	3.3228e-1 (2.72e-1)$_{(5)}$	7.4370e-1 (4.31e-6)$_{(6)}$
	10	9.6563e-2 (9.88e-3)$_{(3)}$	2.9130e-1 (1.03e-1)$_{(5)}$	**7.6375e-3 (6.75e-4)$_{(1)}$**	1.2132e-2 (8.02e-3)$_{(2)}$	2.3351e-1 (1.89e-1)$_{(4)}$	9.9498e-1 (1.10e+0)$_{(6)}$
MaF7	5	2.7974e+0 (5.75e-1)$_{(6)}$	3.8120e-1 (2.81e-2)$_{(4)}$	3.6470e-1 (5.75e-2)$_{(3)}$	3.4757e-1 (6.40e-3)$_{(2)}$	5.0802e-1 (1.39e-2)$_{(5)}$	**3.3433e-1 (8.95e-3)$_{(1)}$**
	8	2.1206e+0 (6.37e-1)$_{(6)}$	9.3471e-1 (4.59e-2)$_{(3)}$	**8.1213e-1 (6.45e-2)$_{(1)}$**	1.7487e+0 (1.46e-1)$_{(5)}$	1.5891e+0 (2.70e-1)$_{(4)}$	9.1137e-1 (3.08e-2)$_{(2)}$
	10	2.4937e+0 (4.25e-1)$_{(4)}$	1.6632e+0 (2.52e-1)$_{(3)}$	**1.1407e+0 (1.41e-1)$_{(1)}$**	3.4455e+0 (1.48e-1)$_{(6)}$	3.0655e+0 (5.77e-1)$_{(5)}$	1.2623e+0 (4.95e-2)$_{(2)}$
MaF8	5	3.7264e-1 (1.34e-2)$_{(5)}$	2.8127e-1 (4.40e-2)$_{(3)}$	**1.2470e-1 (4.28e-3)$_{(1)}$**	1.3984e-1 (4.74e-3)$_{(2)}$	4.6620e-1 (6.05e-2)$_{(6)}$	2.8601e-1 (2.15e-2)$_{(4)}$
	8	7.3585e-1 (1.70e-2)$_{(5)}$	4.2430e-1 (5.51e-2)$_{(3)}$	**1.8050e-1 (6.41e-3)$_{(1)}$**	2.1707e-1 (1.17e-2)$_{(2)}$	9.6370e-1 (1.54e-1)$_{(6)}$	4.6762e-1 (3.88e-2)$_{(4)}$
	10	1.1615e+0 (7.03e-3)$_{(6)}$	4.8221e-1 (6.93e-2)$_{(4)}$	**2.0485e-1 (8.04e-3)$_{(1)}$**	2.5163e-1 (1.15e-2)$_{(3)}$	1.0000e+0 (1.48e-1)$_{(5)}$	2.5103e-1 (4.90e-2)$_{(2)}$
MaF9	5	2.9689e-1 (2.96e-3)$_{(3)}$	4.0295e-1 (1.32e-1)$_{(4)}$	5.0136e-1 (3.48e-1)$_{(6)}$	**1.4227e-1 (7.04e-3)$_{(1)}$**	4.1090e-1 (1.13e-1)$_{(5)}$	1.5407e-1 (6.17e-2)$_{(2)}$
	8	4.8461e-1 (2.01e-2)$_{(3)}$	1.4498e+0 (1.84e+0)$_{(6)}$	**2.3675e-1 (1.54e-2)$_{(1)}$**	2.5799e-1 (1.59e-2)$_{(2)}$	9.8022e-1 (2.20e-1)$_{(5)}$	6.6120e-1 (5.64e-1)$_{(4)}$
	10	4.0869e-1 (3.33e-2)$_{(3)}$	1.4239e+0 (1.57e+0)$_{(5)}$	**2.5778e-1 (1.92e-2)$_{(1)}$**	3.2725e-1 (1.77e-2)$_{(2)}$	1.2737e+0 (3.80e-1)$_{(4)}$	8.9443e+0 (1.79e+1)$_{(6)}$
MaF10	5	7.6720e-1 (1.12e-1)$_{(4)}$	4.9512e-1 (1.65e-2)$_{(2)}$	1.2927e+0 (1.83e-1)$_{(5)}$	5.1060e-1 (1.76e-2)$_{(3)}$	**4.6984e-1 (1.34e-2)$_{(1)}$**	1.6642e+0 (9.89e-2)$_{(6)}$
	8	1.5083e+0 (1.25e-1)$_{(4)}$	**1.0962e+0 (1.07e-1)$_{(1)}$**	1.7629e+0 (2.11e-1)$_{(5)}$	1.1860e+0 (9.86e-2)$_{(2)}$	1.2547e+0 (2.82e-1)$_{(3)}$	1.9297e+0 (1.08e-1)$_{(6)}$
	10	1.9058e+0 (8.13e-2)$_{(3)}$	1.9938e+0 (4.30e-1)$_{(4)}$	2.0001e+0 (2.63e-1)$_{(5)}$	**1.5498e+0 (9.81e-2)$_{(1)}$**	1.6051e+0 (3.34e-1)$_{(2)}$	2.2797e+0 (1.66e-1)$_{(6)}$
MaF11	5	4.3036e+0 (3.04e-1)$_{(6)}$	8.2010e-1 (1.54e-2)$_{(2)}$	9.2477e-1 (2.20e-1)$_{(4)}$	8.2684e-1 (7.46e-3)$_{(3)}$	1.7973e+0 (4.53e-1)$_{(5)}$	**8.1875e-1 (8.05e-2)$_{(1)}$**
	8	7.5018e+0 (2.53e-2)$_{(6)}$	3.8225e+0 (1.10e+0)$_{(4)}$	2.4835e+0 (2.96e-1)$_{(3)}$	1.9556e+0 (2.22e-1)$_{(2)}$	4.3574e+0 (7.55e-1)$_{(5)}$	**1.8324e+0 (1.51e-1)$_{(1)}$**
	10	1.5239e+1 (2.73e-2)$_{(6)}$	6.6816e+0 (2.85e+0)$_{(4)}$	2.8756e+0 (2.91e-1)$_{(3)}$	**2.2254e+0 (2.02e-1)$_{(1)}$**	8.3677e+0 (2.65e+0)$_{(5)}$	2.6246e+0 (6.70e-1)$_{(2)}$
MaF12	5	1.3646e+0 (2.67e-2)$_{(6)}$	1.2040e+0 (9.36e-3)$_{(3)}$	**1.1785e+0 (6.11e-2)$_{(1)}$**	1.2067e+0 (2.43e-3)$_{(4)}$	1.2128e+0 (6.00e-3)$_{(5)}$	1.1995e+0 (3.69e-2)$_{(2)}$
	8	4.0440e+0 (1.57e-1)$_{(6)}$	3.5527e+0 (4.25e-2)$_{(4)}$	**3.3073e+0 (6.44e-2)$_{(1)}$**	3.5432e+0 (2.31e-2)$_{(3)}$	3.4701e+0 (3.01e-2)$_{(2)}$	4.0124e+0 (1.24e-1)$_{(5)}$
	10	7.3356e+0 (3.10e-1)$_{(6)}$	5.8305e+0 (1.43e-1)$_{(5)}$	**4.8155e+0 (4.29e-2)$_{(1)}$**	5.7755e+0 (4.06e-2)$_{(3)}$	5.7782e+0 (6.10e-2)$_{(4)}$	5.7157e+0 (1.64e-1)$_{(2)}$
MaF13	5	3.3183e+0 (2.07e-1)$_{(6)}$	2.6446e-1 (3.06e-2)$_{(4)}$	1.7465e-1 (5.77e-2)$_{(3)}$	1.4637e-1 (7.60e-3)$_{(2)}$	5.6565e-1 (1.91e-1)$_{(5)}$	**1.2841e-1 (2.43e-2)$_{(1)}$**
	8	3.8659e-1 (3.84e-2)$_{(5)}$	3.6443e-1 (7.63e-2)$_{(4)}$	2.2054e-1 (6.70e-2)$_{(3)}$	1.9693e-1 (1.33e-2)$_{(2)}$	8.7379e-1 (2.82e-1)$_{(6)}$	**1.9692e-1 (5.64e-2)$_{(1)}$**
	10	5.8577e-1 (7.53e-2)$_{(5)}$	4.1626e-1 (1.19e-1)$_{(4)}$	2.3001e-1 (7.25e-2)$_{(3)}$	2.2905e-1 (1.63e-2)$_{(2)}$	9.9086e-1 (2.38e-1)$_{(6)}$	**2.0072e-1 (4.49e-2)$_{(1)}$**
MaF14	5	**6.6113e-1 (1.30e-1)$_{(1)}$**	2.6815e+0 (1.26e+0)$_{(4)}$	3.1455e+0 (1.10e+0)$_{(5)}$	6.9543e-1 (1.83e-1)$_{(2)}$	8.1702e-1 (1.54e-1)$_{(3)}$	7.7551e+0 (1.25e+0)$_{(6)}$
	8	**9.1684e-1 (1.86e-1)$_{(1)}$**	8.8913e+0 (3.72e+0)$_{(4)}$	6.3861e+3 (2.88e+3)$_{(6)}$	9.7103e-1 (2.38e-1)$_{(2)}$	1.0321e+0 (2.08e-1)$_{(3)}$	1.4383e+2 (3.00e+2)$_{(5)}$
	10	1.0684e+0 (7.27e-2)$_{(2)}$	9.9039e+0 (4.21e+0)$_{(4)}$	6.7520e+3 (4.57e+3)$_{(6)}$	1.2315e+0 (4.82e-1)$_{(3)}$	**1.0631e+0 (8.48e-2)$_{(1)}$**	4.8899e+2 (1.10e+3)$_{(5)}$
MaF15	5	**5.2237e-1 (4.20e-2)$_{(1)}$**	1.3565e+0 (2.70e-1)$_{(5)}$	1.1923e+0 (1.80e+0)$_{(4)}$	6.1264e-1 (6.11e-2)$_{(3)}$	6.0187e-1 (5.68e-2)$_{(2)}$	2.5537e+0 (1.46e+0)$_{(6)}$
	8	1.1243e+0 (1.08e-1)$_{(3)}$	2.2142e+0 (7.15e-1)$_{(5)}$	5.2098e+1 (3.59e+0)$_{(6)}$	**9.3027e-1 (7.30e-2)$_{(1)}$**	9.9321e-1 (5.02e-2)$_{(2)}$	1.3089e+0 (2.32e+1)$_{(4)}$
	10	1.1141e+0 (6.36e-2)$_{(3)}$	4.0523e+0 (1.85+0)$_{(4)}$	6.3996e+1 (4.14e+0)$_{(6)}$	1.0713e+0 (8.17e-2)$_{(2)}$	**1.0347e+0 (4.99e-2)$_{(1)}$**	1.8419e+1 (2.59e+1)$_{(5)}$
best/all	-	3/45	1/45	14/45	6/45	6/45	15/45
rank	-	197(6)	169(4)	143(2)	109(1)	178(5)	149(3)

MaF13 with degenerate PFs and complicated variable linkages, MaOBSO solve it very well in the cases of all objectives. At last, MaF14 and MaF15 with complicated fitness landscape with mixed variable separability, especially in large-scale cases, MaOBSO gave a median performance among all the compared MaOEAs.

Based on the above comparison results, we can conclude that the proposed MaOBSO shows more competitive performance in irregular frontier.

V. CONCLUSIONS AND OUTLOOKS

MaOBSO is proposed in this paper to improve the convergence and diversity of MaOPs. The proposed algorithm adopts decision variable clustering method to divides the variables into convergence-related variables and diversity-related variables. The different optimization strategies suitable to different categories variable are designed respectively. A new adaptive clustering strategy according to the characteristics of corner points is proposed for many-objective brainstorming algorithm. The simulation results show that the algorithm has good performance.

In the future, it is interesting to investigate the performance of MaOBSO for a wider range of problem, such as complicated Pareto frontier problems, real-life problems, large-scale many-objective optimization problems and so on. Moreover, improving the universality of MaOBSO is also considered as part of our future work.

REFERENCES

[1] 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 Resour. Res.*, vol. 47, no. 2, pp. 155–170, Feb. 2010.

[2] H. Freire, P. B. M. Oliveira, and E. J. S. Pires, "From single to many-objective PID controller design using particle swarm optimization," *Int. J. Control Autom. Syst.*, vol. 15, no. 2, pp. 918–932, Feb. 2017.

[3] M. F. Shaaban, A. H. Osman, and F. A. Aseeri, "A multi-objective allocation approach for power quality monitoring devices," *IEEE Access*, vol. 1, pp. 40866–40877, Mar. 2019.

[4] K. Ikeda, H. Kita, and S. Kobayashi, "Failure of Pareto-based MOEAs: Does non-dominated really mean near to optimal?" in *Proc. IEEE Int. Conf. Congr. Evol. Comput.*, Seoul, South Korea, May 2001, pp. 957–962.

[5] V. Khare, X. Yao, and K. Deb, "Performance scaling of multi-objective evolutionary algorithms," in *Proc. 2nd Int. Conf. Evol. Multi-Criterion Optim.* Berlin, Germany: Springer, 2003, pp. 376–390.

[6] D. Brockhoff, T. Friedrich, N. Hebbinghaus, C. Klein, F. Neumann, and E. Zitzler, "On the effects of adding objectives to plateau functions," *IEEE Trans. Evol. Comput.*, vol. 13, no. 3, pp. 591–603, Jun. 2009.

[7] R. C. Purshouse and P. J. Fleming, "Evolutionary many-objective optimisation: An exploratory analysis," in *Proc. IEEE Int. Conf. Congr. Evol. Comput.*, Dec. 2003, pp. 2066–2073.

[8] 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.

[9] S. Chand and M. Wagner, "Evolutionary many-objective optimization: A quick-start guide," *Surv. Oper. Res. Manage. Sci.*, vol. 20, no. 2, pp. 35–42, Dec. 2015.

[10] B. Li, J. Li, K. Tang, and X. Yao, "Many-objective evolutionary algorithms: A survey," *ACM Comput. Surv.*, vol. 48, no. 1, pp. 1–35, 2015.

[11] H. Ishibuchi, N. Tsukamoto, and Y. Nojima, "Evolutionary many-objective optimization: A short review," in *Proc. IEEE Int. Conf. Congr. Evol. Comput.*, Hong Kong, Jun. 2008, pp. 2419–2426.

[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] 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.

[14] G. Wang and H. Jiang, "Fuzzy-dominance and its application in evolutionary many objective optimization," in *Proc. Int. Conf. Comput. Intell. Secur.*, 2007, pp. 195–198.

[15] 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.

[16] 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.

[17] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Proc. 8th Int. Conf. Parallel Problems Solving Nat.*, Birmingham, U.K., 2004, pp. 832–842.

[18] 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, Sep. 2007.

[19] K. Li, S. Kwong, J. Cao, M. Li, J. Zheng, and R. Shen, "Achieving balance between proximity and diversity in multi-objective evolutionary algorithm," *Inf. Sci.*, vol. 182, no. 1, pp. 220–242, Jan. 2012.

[20] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 1, pp. 45–76, Mar. 2011.

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

[22] 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.

[23] 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.

[24] X. Ma, F. Liu, Y. Qi, X. Wang, L. Li, L. Jiao, M. Yin, and M. Gong, "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.

[25] 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. 2013.

[26] Y. Qi, X. L. Ma, F. Liu, L. Jiao, J. Sun, and J. Wu, "MOEA/D with adaptive weight adjustment," *Evol. Comput.*, vol. 22, no. 2, pp. 231–264, 2014.

[27] 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.

[28] 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.

[29] 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.

[30] H.-L. Liu, F. Gu, and Q. Zhang, "Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems," *IEEE Trans. Evol. Comput.*, vol. 18, no. 3, pp. 450–455, Jun. 2014.

[31] 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.

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

[33] Q. Lin, J. Li, Z. Du, J. Chen, and Z. Ming, "A novel multi-objective particle swarm optimization with multiple search strategies," *Eur. J. Oper. Res.*, vol. 247, no. 3, pp. 732–744, Dec. 2015.

[34] Q. Lin, S. Liu, Q. Zhu, C. Tang, R. Song, J. Chen, C. A. C. Coello, K.-C. Wong, and J. Zhang, "Particle swarm optimization with a balanceable fitness estimation for many-objective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 22, no. 1, pp. 32–46, Feb. 2018.

[35] D. Gong, J. Sun, and Z. Miao, "A set-based genetic algorithm for interval many-objective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 22, no. 1, pp. 47–60, Feb. 2018.

[36] R. Cheng, M. Li, Y. Tian, X. Zhang, S. Yang, Y. Jin, and X. Yao, "A benchmark test suite for evolutionary many-objective optimization," *Complex Intell. Syst.*, vol. 3, no. 1, pp. 67–81, Mar. 2017.

[37] Q. Lin, S. Liu, K.-C. Wong, M. Gong, C. A. C. Coello, J. Chen, and J. Zhang, "A clustering-based evolutionary algorithm for many-objective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 23, no. 3, pp. 391–405, Jun. 2019.

[38] Q. Lin, S. Liu, K.-C. Wong, M. Gong, C. A. C. Coello, J. Chen, and J. Zhang, "A clustering-based adaptive evolutionary algorithm for multiobjective optimization with irregular Pareto fronts," *IEEE Trans. Evol. Comput.*, vol. 23, no. 3, pp. 391–405, Jun. 2019.

[39] X. Zhang, Y. Tian, R. Cheng, and Y. Jin, "A decision variable clustering-based evolutionary algorithm for large-scale many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 22, no. 1, pp. 97–112, Feb. 2018.

[40] J. Xue, Y. Wu, Y. Shi, and S. Cheng, "Brain storm optimization algorithm for multi-objective optimization problems," in *Proc. Int. Conf. Swarm Intell.*, in Lecture Notes in Computer Science, Jun. 2012, vol. 7331, no. 1, pp. 513–519.

[41] L. Xie and Y. Wu, "A modified multi-objective optimization based on brain storm optimization algorithm," in *Proc. Int. Conf. Swarm Intell.* Cham, Switzerland: Springer, 2014, pp. 328–339.

[42] X. Guo, Y. Wu, L. Xie, S. Cheng, and J. Xin, "An adaptive brain storm optimization algorithm for multiobjective optimization problems," in *Proc. Int. Conf. Swarm Intell.*, in Lecture Notes in Computer Science, 2015, vol. 9140, no. 4, pp. 365–372.

[43] Y. Wu, L. Xie, and Q. Liu, "Multi-objective brain storm optimization based on estimating in knee region and clustering in objective-space," *Adv. Swarm Intell.*, pp. 479–490, Jun. 2016.

[44] W. Yali, "Mult-objective brain storm optimization for environment economic dispatch," in *Brain Storm Optimization Algorithm Theory and Application*, 8th ed. Beijing, China: Science Press, 2017, pp. 187–201.

[45] K. Deb, "Multi-objective genetic algorithms: Problem difficulties and construction of test problems," *Evol. Comput.*, vol. 7, no. 3, pp. 205–230, 1999.

[46] Y. Shi, "An optimization algorithm based on brainstorming process," *Int. J. Swarm Intell. Res.*, vol. 2, no. 4, pp. 35–62, Dec. 2011.

[47] C. Li, D. Hu, and Z. Song, "A vector grouping learning brain storm optimization algorithm for global optimization problems," *IEEE Access*, vol. 6, pp. 78193–78213, Dec. 2018.

[48] C. Li, Z. Song, and J. Fan, "A brain storm optimization with multiinformation interactions for global optimization problems," *IEEE Access*, vol. 6, pp. 19304–19323, Mar. 2018.

[49] J. Zhengxuan, D. Haibin, and S. Yuhui, "Hybrid brain storm optimization and simulated annealing algorithm for continuous optimization problems," *Int. J. Bio-Inspired Comput.*, vol. 8, no. 2, pp. 109–121, 2016.

[50] Y. Guo *et al.*, "Grid-based dynamic robust multi-objective brain storm optimization algorithm," *Soft Comput.*, pp. 1–21, Sep. 2019.

[51] D. Satyabrata, J. Deepak, and T. Gaurav, "Multiobjective analog/RF circuit sizing using an improved brain storm optimization algorithm," *Memetic Comput.*, vol. 10, no. 4, pp. 423–440, Dec. 2018.

[52] X. Zhang, G.-M. Yang, T. Zhang, and M.-L. Shi, "Multi-indicators brain storm optimization algorithm for multi-objective optimization problems," *DEStech Trans. Comput. Sci. Eng.*, pp. 61–70, Mar. 2019.

[53] C. Dai and X. Lei, "A multiobjective brain storm optimization algorithm based on decomposition," *Complexity*, vol. 12, pp. 1–11, Jan. 2019.

[54] 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, Jul. 2000.

[55] X. Zhang, Y. Tian, R. Cheng, and Y. Jin, "An efficient approach to nondominated sorting for evolutionary multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 2, pp. 201–213, Apr. 2015.

[56] X. Zhang, Y. Tian, R. Cheng, and Y. Jin, "Empirical analysis of a tree-based efficient non-dominated sorting approach for many-objective optimization," in *Proc. IEEE Int. Conf. Comput. Intell.*, Athens, Greece, Dec. 2016, pp. 1–8.

[57] P. A. N. Bosman and D. Thierens, "The balance between proximity and diversity in multiobjective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 174–188, Apr. 2003.

YALI WU received the Ph.D. degree in control science and engineering from Xi'an Jiaotong University, Xi'an, China, in 2003. She is currently a Professor with the School of Automation and Information Engineering, Xi'an University of Technology. She has authored more than 30 refereed articles. Her main research areas are intelligent optimization algorithm theory and application, and complex system modeling and optimization.

XINRUI WANG received the M.Sc. degree in systems engineering from the Xi'an University of Technology. Her research interests include intelligent optimization algorithm theory and application, and deep learning.

YULONG FU received the M.Sc. degree in control engineering from the Xi'an University of Technology. His research interest includes intelligent optimization algorithm theory and application.

GUOTING LI received the M.Sc. degree in systems engineering from the Xi'an University of Technology. Her research interest includes deep learning.