Research Article

An Opposition-Based Evolutionary Algorithm for Many-Objective Optimization with Adaptive Clustering Mechanism

Wan Liang Wang , Weikun Li, and Yu Le Wang

Zhejiang University of Technology, Hangzhou, Zhejiang 310023, China

Correspondence should be addressed to Wan Liang Wang; zjutwml@zjut.edu.cn

Received 7 January 2019; Revised 14 February 2019; Accepted 9 April 2019; Published 2 May 2019

Academic Editor: Michele Migliore

Copyright © 2019 Wan Liang Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Balancing convergence and diversity has become a key point especially in many-objective optimization where the large numbers of objectives pose many challenges to the evolutionary algorithms. In this paper, an opposition-based evolutionary algorithm with the adaptive clustering mechanism is proposed for solving the complex optimization problem. In particular, opposition-based learning is integrated in the proposed algorithm to initialize the solution, and the nondominated sorting scheme with a new adaptive clustering mechanism is adopted in the environmental selection phase to ensure both convergence and diversity. The proposed method is compared with other nine evolutionary algorithms on a number of test problems with up to fifteen objectives, which verify the best performance of the proposed algorithm. Also, the algorithm is applied to a variety of multiobjective engineering optimization problems. The experimental results have shown the competitiveness and effectiveness of our proposed algorithm in solving challenging real-world problems.

1. Introduction

Over the last two decades, evolutionary algorithm (EA) has been proven to be prevalent and efficient to solve real-world optimization problem [1]. Some of these well-known methodologies include genetic algorithms (GAs) [2], evolution strategies (ES) [3], and ant colony optimization (ACO) [4]. However, real-world optimization problems always involve multiple objectives, which means there is no single solution when considering multiple objectives as the goal of the optimization process [5]. In this case, the solutions for a multiobjective problem (MOP), which is the main focus of the algorithm, represent the trade-offs between the objectives due to the nature of such problems [6].

For the evolutionary approach to address multiobjective optimization, which is called multiobjective evolutionary algorithm (MOEA), different varieties of the algorithms have been proposed in recent years. Among them, the algorithms inspired by nature have drawn the attention like improving strength-Pareto evolutionary algorithm (SPEA2) [7], nondominated sorting genetic algorithm version 2 (NSGA-II)

[8], multiobjective particle swarm optimization (MOPSO) [9], multiobjective moth-flame algorithm [10], and multiobjective ant lion optimizer [11].

There is no doubt that MOEA has been proven to be prevalent and efficient to solve optimization problem with less than three objectives [1]. However, with the gradually rising scale of data and the pluralism of target requirements, many real-world optimization problems containing more than three objectives named “many-objective problems” (MaOPs) [12, 13] are widely appearing in engineering [14], traffic [15], and water [16]. Unfortunately, the effectiveness of previous MOEAs tends to deteriorate dramatically with the increase in the number of objectives, which has been verified in [17, 18]. This can be attributed to the situation that almost all solutions in a population become nondominated with one another, and the conflict between convergence and diversity becomes aggravated with the increasing number of objectives in MaOPs [19, 20]. Moreover, computational complexity of calculating some performance metrics and the representation and visualization of the trade-off surface are also the difficulties of MaOPs. To overcome these drawbacks,

a series of many-objective evolutionary algorithms (MaOEAs) have been proposed to address these optimization problems with more than three objectives. In summary, the proposed algorithms can be roughly classified into the following four types.

1.1. New Domination Relation-Based Approach. As the selection criterion based on the standard dominance relationship fails to distinguish solutions in MaOPs, a number of new domination principles have been proposed to adaptively discretize the Pareto-optimal front, for instance, the ϵ -dominance [21, 22], CDAS-dominance [23, 24], α -dominance [25], fuzzy Pareto dominance [26], L -dominance [27], cone-domination [28, 29], and grid-based evolutionary algorithm (GrEA) [30]. Furthermore, the recently proposed generalization of Pareto optimality (GPO) [31] expands the dominance area of solutions to enhance the scalability of existing Pareto-based algorithms [32] and employs a shift-based density estimation strategy (SDE) into the dominance-based criterion, and θ -dominance [33, 34] introduces a new dominance relation to rank solutions, which all have been proven to be more effective than the original Pareto dominance relation. In a word, these algorithms are proposed with several variants of the Pareto dominance to enhance the selection pressure toward the PF. However, the drive toward a more aggressive selection pressure could make diversity maintenance more difficult in these new dominance-based MaOEAs.

1.2. Indicator-Based Approach. Aiming to obtain a desired ordering among the representative PF approximations, indicator-based MOEAs have been widely studied, for example, the indicator-based evolutionary algorithm (IBEA) [35], SMS-EMOA [36], the fast hypervolume-based evolutionary algorithm (HypE) [37], DNMOEA/HI [38], R2 indicator based [39], Δ_p [40], stochastic ranking algorithm based on multiple indicators (SRA) [41], and IGD indicator-based evolutionary algorithm (MaOEA/IGD) [42]. However, the computational cost of some performance calculation could be prohibitively expensive as the number of objectives increases.

1.3. Decomposition-Based Approach. These algorithms always decompose a MaOP into several single-objective optimization problems and use aggregation functions to differentiate many-objective solutions. Among the methods that use a set of weight vectors to generate multiple aggregation functions, multiobjective evolutionary algorithm based on decomposition (MOEA/D) [43] is the most representative algorithm, which aggregates the objectives of an MOP into an aggregation function with a unique weight vector. Around the MOEA/D, several variants have been proposed to strike a better balance between convergence and diversity such as I-DBEA [44], MOEA/D-DD [45], MOEA/D-DU [46], and MOEA/D-LWS [47]. However, the situation that the number of scalarizing functions is usually very limited might cause difficulties

for diversity maintenance of the solution especially when compared with the exponentially increasing objective space.

1.4. Reference Set-Based Approach. The algorithms of this category use a set of reference solutions to measure the quality of solutions. Thus, the search process is guided by the solutions in the reference solution set. Praditwong and Yao [48] and Wang [49] proposed a novel two archive algorithm (TAA) and its improved version (Two_Arch2), in which the convergence archive (CA) can be seen as an online-updated real reference set. The recently proposed vector angle-based evolutionary algorithm (VaEA) [50] uses the population as the reference set to dynamically guide the evolutionary process. As for the algorithm using virtual reference set, NSGA-III [51, 52] is predominant which employs a set of predefined reference points to manage the diversity of the candidate solutions. Hereafter, a number of algorithms have been proposed with the reference point or vector such as the reference vector-guided evolutionary algorithm for many-objective optimization (RVEA) [53], reference vector-guided evolutionary Pareto evolutionary algorithm with reference direction (SPEAR) [54], and many-objective evolutionary optimization based on reference points (RPEA) [55].

The algorithm from the third category and fourth category becomes particularly prevalent for many-objective optimization, which could be attributed to the low cost to achieve a representative subset of the entire PF especially in a limited population size. However, most algorithms belong to these categories simply use the angle or distance solely to measure the quality of the population members with the reference set, which may lose some good solutions due to their simplex selection mechanism. Furthermore, it has been logically proved by the No Free-Lunch (NFL) theorem [56] that none of these algorithms is able to solve all optimization problems, which allows the researchers to propose new methods or improve the current algorithms for better solving the problems [33, 57]. Therefore, this paper proposes an opposition-based multiobjective evolutionary algorithm with an adaptive clustering mechanism, in short named OBEOA to strengthen the selection mechanism through comprehensive consideration of the angle and the distances. The main properties of OBEOA can be summarized as follows:

- (i) A new initialization approach is designed with the assistance of the opposition-based learning (OBL) to generate the population. Due to the fact that random initialization lowers the chance of sampling better regions in algorithms, here we use the OBL to initialize the populations in stand of the previous random method. Moreover, opposition-based learning is also adapted in the evolutionary process with the aim of enhancing the probability of obtaining the better solutions.
- (ii) An adaptive clustering strategy is integrated in this algorithm. In this proposed strategy, the acute angle and perpendicular distance between the candidate solutions and the reference vectors are combined

through an adaptive mechanism to cluster the candidate. Moreover, a novel selective approach is designed to dynamically select the individual with comprehensive consideration of the balance convergence and diversity.

Furthermore, an extensive comparison between the proposed OBEA with nine algorithms is implemented on 60 instances of 14 test problems taken from two well-known test suites. The results indicate that OBEA is a very promising alternative for many-objective optimization. The rest of this paper is organized as follows. Section 2 introduces the background knowledge, and details of the proposed OBEA are described in Section 3. Section 4 presents the numerical results of OBEA on benchmark and the detailed analysis of the proposed algorithm together with nine MaOEAs. Finally, conclusions and future work are given in Section 5.

2. Background

In this section, the main components of multiobjective optimization problem (MOP) are given first, which involve the basic knowledge of optimization and Pareto dominate. Next, a brief description of the reference vector is given, which is used as the underlying mechanism for solving many-objective optimization problems.

2.1. Multiobjective Problem. Generally, a multiobjective optimization problem can be stated as follows:

$$\begin{aligned} &\text{minimize } F(x) = (f_1(x), f_2(x), \dots, f_m(x))^T \\ &\text{subject to } x \in \Omega \subseteq \mathbb{R}^n, \end{aligned} \quad (1)$$

where $x = \{x_1, x_2, \dots, x_n\}$ is the decision vector that satisfies $x \in \Omega$ and Ω stands for the decision space. The objective function vector $F : \Omega \rightarrow \mathbb{R}^m$ consists of m ($m \geq 2$) objectives and $\mathbb{R}^m$ refers to the objective space.

Definition 1. Given two vector $x, y \in \Omega$, x is *Pareto dominate* y (denoted as $x < y$) if and only if $f_i(x) \leq f_i(y)$ for each $i \in (1, 2, \dots, m)$, and $\exists j \in (1, 2, \dots, m)$, $f_j(x) < f_j(y)$.

Definition 2. A decision vector $\tilde{x} \in \Omega$ is said to be *Pareto optimal*, if and only if there is no $x \in \Omega$, $x < \tilde{x}$.

Definition 3. The set of Pareto-optimal solutions (PS) is defined as $PS = \{x \in \Omega \mid x \text{ is Pareto optimal}\}$.

Definition 4. The Pareto-optimal front (PF) is defined as $PF = \{F(x) \mid x \in PS\}$.

2.2. Reference Vector. As an underlying mechanism throughout the algorithm, OBEA uses Das and Dennis's systematic approach [58], and adaptation of this approach generates the reference points and thus forms the reference vectors. The original Das and Dennis's method places points on a normalized hyperplane, where the number of reference points depends on the dimension of objective

space m and positive integer H . The equation can be described as follows:

$$N = \binom{H+m-1}{m-1}. \quad (2)$$

However, the number of reference points would rapidly increase when m is a relatively large number. To address the drawback of the computational burden of the reference point, a number of new approaches have been proposed [45, 59]. OBEA utilizes the two-layered reference point generation approach, as suggested in [33]. The hyperplane is divided into two parts, which are the boundary and inner layers, as shown in Figure 1. The detail of implementation is as follows:

$$N = \binom{H_1+m-1}{m-1} + \binom{H_2+m-1}{m-1}. \quad (3)$$

The reference vector is the fundamental mechanism for the proposed algorithm. On the one hand, the acute angle and perpendicular distance between the candidate solutions and the reference vectors are adopted in OBEA; on the other hand, the proposed algorithm tends to find near Pareto-optimal solutions corresponding to the reference vectors. Furthermore, the Das and Dennis's systematic approach is utilized in 3-objective and 5-objective problems, while the two-layered reference point generation approach is used in other situations ($M > 5$).

3. Proposed Algorithm: OBEA

The pseudocode of the proposed OBEA is presented in Algorithm 1. This algorithm shares a common framework of many evolutionary algorithms and consists of four main phases. First, N solutions and reference vectors are initialized with the OBL. Then, potential solutions are selected into the mating pool. In what follows, a set of offspring solutions is obtained by applying crossover and mutation operations. Finally, solution P is chosen by adopting the environmental selection procedure. The above steps continue until the termination criteria are met. In the following sections, the details of each component in OBEA will be explained step by step (Algorithm 1).

3.1. Initialization with the Opposition-Based Learning. Since the opposition-based learning (OBL) was first proposed by Tizhoosh [60], the OBL has been adopted in various algorithms, which is due to its promising potential to improve the performance of the algorithms [61]. Because of the fact that OBL can obtain fitter starting candidates even when there is no a priori knowledge and enhance the probability of detecting better regions, here we use the OBL to initialize the algorithm. First, the definition of the opposite number and the opposite point is given below.

Definition 5. Let $x \in [a, b]$ be a real number. Its opposite number $\tilde{x}$ is defined as follows [62]:

FIGURE 1: An illustration of the approach to generate the reference point and the reference vector in three objective spaces. As the figure shows, given the $H_1 = 2$ and $H_2 = 1$, respectively, the final number of the reference point is 9.

```

Output: final population  $P$ 
Initialization: create the initial population  $P_0$ , reference vector  $V$ 
while (termination criterion not fulfilled) : do
     $\bar{P} \leftarrow$  Mating-selection ( $P_0$ )
     $Q \leftarrow$  Variation ( $\bar{P}$ )
     $R \leftarrow \bar{P} \cup Q$ 
     $P_{\text{next}} \leftarrow$  Environmental-selection ( $R, V$ )
return  $P$ 

```

ALGORITHM 1: General framework of OBEA.

$$\tilde{x} = a + b - x. \quad (4)$$

Definition 6. Let $x(x_1, x_2, \dots, x_D)$ be a point in D -dimensions space, where $x_i \in [a_i, b_i]$, $i = 1, 2, \dots, D$. Its opposite point $\tilde{x}(\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_D)$ is defined as follows [62]:

$$\tilde{x}_i = a_i + b_i - x_i. \quad (5)$$

The initialization phase of OBEA is shown in Figure 2. Here we first divide population P_N into two parts, and the first half population P_{N1} is generated by a random distribution. Thereafter, the remaining half population P_{N2} is initialized in terms of OBL, as follows:

$$P_{N2} = a_i + b_i - P_{N1}. \quad (6)$$

Finally, the set $\{P_{N1} \cup P_{N2}\}$ is restructured as the initial population P_N^* .

Note that the initialization of OBEA uses the OBL strategy to calculate the opposite point which shares the same idea from our previous work [63]. There are some similarities between them. In detail, both of these algorithms divide the population into two parts and use the random methods and OBL strategy to generate one of them, respectively. Furthermore, both of the algorithms apply the OBL strategy in the process of optimization. However, there are still some

differences between our proposed OBEA and our previous work. For our previous work, the OBL strategy-based initialization in this algorithm is only for the multiobjective optimization problem, which the number of the objective is two or three. As for the initialization in our proposed OBEA, the number of objective is far more than that in our previous work as OBEA is designed to solve the many-objective optimization problem. Moreover, OBL strategy in OBEA is also included to design the individual selection and construct the last front with the aim of improving the probability of detecting better regions, which are shown in the Section 3.4. However, OBL strategy in our previous work is only adopted to calculate opposite individuals.

3.2. Offspring Creation. In the proposed OBEA, two steps are involved: (1) mating selection, which chooses parents for offspring generation, and (2) variation operation, which generates new candidate solutions. Due to the poor performance to generate offspring solutions in high-dimensional objective space, different methods have been proposed. Here the polynomial mutation and the simulated binary crossover (SBX) [64] are employed as in many other algorithms [59, 65].

FIGURE 2: An example showing the initialization phase in OBEA.

3.3. Environmental Selection. Environmental selection aims at surviving N individuals from the current population and their offspring to select the optimal population with the hybrid selection-based nondominated sort. To be specific, the environmental selection consists of four steps: non-dominated sorting, normalization, adaptive clustering, and opposition-based selection. The procedure of environmental selection is shown in Algorithm 2.

3.3.1. Normalization. The normalization procedure incorporated into OBEA is similar to the other algorithms. In normalization, the objective $f_i(x), i = 1, 2, \dots, m$, can be replaced as follows:

$$\tilde{f}_i(x) = \frac{f_i(x) - g_i^{\min}}{g_i^{\max} - g_i^{\min}}, \quad (7)$$

where $g_i^{\min}$ is the ideal point with the aim to find the minimum value of each objective for all solutions in S . Similarly, $g_i^{\max}$ is the nadir point. Moreover, the norm (in the normalized objective space) of each solution in S is also calculated, which is used to calculate the acute angle and the perpendicular distance between the candidate solutions and the reference vectors in the normalized objective space. Given $x_k, x_k \in S$, its norm (denoted as $\text{norm}(x_k)$) is defined as follows:

```

Input:  $R$ , unit reference vector set  $V$ 
Output:  $P_{\text{next}}$ 
 $I \leftarrow \emptyset, i = 1$ 
 $(\bar{F} = \{F_1, F_2, \dots\}) \leftarrow \text{Non\_dominated\_sort}(R)$ 
repeat
   $I = I \cup F_i$  and  $i = i + 1$ 
until  $|I| \geq N$ 
  Last front to be included:  $F_l = F_i$ 
if  $|I_l| = N$  then
   $P_{\text{next}} = I$ , break
else
   $S \leftarrow \text{Normalize}(I)$ 
   $C \leftarrow \text{Adaptive\_Clustering}(S, V)$ 
   $F_l \leftarrow \text{Opposition\_based\_selection}(C)$ 
   $P_{\text{next}} = P_{\text{next}} \cup F_l$ 

```

ALGORITHM 2: Environmental selection.

$$\text{norm}(x_k) = \sqrt{\sum_{i=1}^m \tilde{f}_i(x_k)^2}. \quad (8)$$

3.3.2. Adaptive Clustering Operation. In OBEA, the clustering is done to the population S at each generation. After

the normalization, population S is partitioned into N (N is the number of reference vectors) sub-populations by associating each individual with its closest reference vector (refer to Figure 3). In the normalized objective space, given the normalized objective vector $\tilde{f}_i(x)$ and the reference vector λ_j , the acute angle between an objective vector and a reference vector can be calculated as

$$\theta_{i,j} = \arccos \frac{\tilde{f}_i(x) \cdot \lambda_j}{\text{norm}(\tilde{f}_i(x))}. \quad (9)$$

Furthermore, u is the projection of $\tilde{f}_i(x)$ on λ_j , and the distance between the origin and u , denoted as du , is calculated as

$$du_{i,j} = \text{norm}(\tilde{f}_i(x)) \cdot \cos \theta_{i,j}. \quad (10)$$

The perpendicular distance between $\tilde{f}_i(x)$ and λ_j , denoted as dp , is calculated as

$$dp_{i,j} = \text{norm}(\tilde{f}_i(x)) \cdot \sin \theta_{i,j}. \quad (11)$$

For the clustering operator, as most algorithms only consider dp or θ , here the sum of the du and θ with an adaptive mechanism will be involved together as

$$AD_{i,j} = (1-s) \cdot du_{i,j} + s \cdot \theta_{i,j}, \quad (12)$$

where s is a sigmoid function which can be described as follows:

$$s = \frac{1}{1 + \exp((\mu - (t/t_{\max})) \times 15)}, \quad (13)$$

where t is the current generation number, $t_{\max}$ is the maximal generation number, and $\mu \in [0, 1)$ is the control parameter.

As Algorithm 3 shows, the core concept of the adaptive clustering operation is to allocate individuals to the sub-populations with the AD values. To be more specific, for one individual S_i , we first calculate the $AD_{i,j}$ between the individual and each reference vector. Thereafter, minimum AD is obtained; thus, the related subpopulation C_k is the cluster where the individual S_i belongs to. In this way, an individual S_i is allocated to a subpopulation C_k if and only if the $AD_{i,j}$ is minimal.

3.4. Opposition-Based Selection. To solve the MOPs with good convergence and diversity, the opposition-based selection has been conducted. The main idea of the proposed opposition-based selection is taking the advantages of opposition-based learning and combined the clustering operation into an efficient method to achieve the goal. Following paragraph will describe the selection mechanism in the details, and the general framework is shown in Algorithm 4.

In the opposition-based selection, we first construct the last front F_1 through the cluster C . To be more specific, our motivation is to find the solution on each reference vector that is closest to the ideal point with the better convergence criterion and diversity. Hence, a hybrid distance (HD) is proposed where the acute angle and perpendicular distance

FIGURE 3: An illustration of the acute angle θ , distance du , and distance dp .

```

Input:  $S$ , unit reference vector set  $V$ 
Output:  $C$ 
for  $i = 1$  to  $|S|$  do
  for  $j = 1$  to  $|V|$  do
    Calculate the  $\theta_{i,j}$ ,  $dp_{i,j}$ ,  $AD_{i,j}$ 
  for  $i = 1$  to  $|S|$  do
     $k = \min_{j \in \{1, \dots, N\}} AD_{i,j}$ 
     $C_k = C_k \cup S_i$ 
return  $C$ 

```

ALGORITHM 3: Adaptive clustering operation.

between the candidate solutions and the reference vectors are combined through an adaptive mechanism to select the candidate in the cluster C effectively. The hybrid distance is as follows:

$$HD_{i,j} = \left(1 + \frac{k}{K} * \gamma\right) \cdot \|\tilde{f}_i\|, \quad (14)$$

where k is the current generation number and K is the maximal generation number. γ can be described as $\gamma = (dp_{i,j}/du_{i,j}) + (\theta_{i,j}/\Theta_{i,j})$, and the $\Theta_{i,j}$ is the smallest angle value between the reference vector and the other reference vectors in the current generation.

After constructing the last front F_1 , we calculate the opposite value of individuals in F_1 according to the OBL in Section 3.1. Whereafter, the opposite values are added into F_1 . Finally, the K individuals are randomly selected from F_1 to construct P_{next} .

3.5. Computational Complexity of OBEA. The normalization and the calculation of norm in OBEA require $O(mN)$ additions. The time complexity for clustering is $O(mN^2)$. In addition, the selection holds a computational complexity of $O(mN^2)$. To sum up, the overall worst complexity of one generation of OBEA is approximately $O(mN^2)$.

```

Input:  $C = \{C_1, C_2, \dots\}$ ,  $N_c$  = number of the  $C$ 
Output:  $K$ 
for  $i = 1 : N_c$  do
  for  $j = 1 : |C_i|$  do
    Calculate the  $HD_{i,j}$  (equation (14))
  for  $i = 1 : N_c$  do
     $F_l = F_l \cup (c \text{ in } C_i : \arg \min HD_{i,j})$ 
  Calculate the opposite individuals in  $F_l$  with the OBL according (equations (4)–(6))
  Add the opposite individuals into the  $F_l$ 
  Randomly choose  $K$  individuals from  $F_l : K = N - |P_{\text{next}}|$ 
return  $K$ 

```

ALGORITHM 4: Opposition-based selection.

4. Experiment Description

In this section, experiments with twelve algorithms such as MOEA/D, dMOPSO [66], MOMBI2 [67], ϵ -MOEA [68], NSGA-III, RVEA, MaOEARD [69], MyODEMR [70] MOEA/DD, MOEA/DVA [71], Two_Arch2, and SPEAR have been adopted in order to evaluate the performance of the proposed OBEA algorithm on 15 benchmark test problems taken from two widely used DTLZ test suites [72] and WFG test suites [73]. For each test problem, objective numbers varying from 3 to 15, i.e., $M \in \{3, 5, 8, 10, 15\}$, are considered.

In the following subsections, the test problems and the quality indicators used in our comparative experiments are first presented. Then, the experimental settings adopted in this study are provided. Moreover, thirty independent runs are executed for each test problem to avoid randomness, and the Wilcoxon rank sum test is adopted to compare the results obtained by OBEA and those nine compared algorithms at a significance level of 0.05.

4.1. Test Problems. Aiming to evaluate the performance effectively, two well-known test suites Deb-Thiele-Laumanns-Zitzler (DTLZ) and Walking-Fish-Group (WFG) are involved in the experiments, as shown in Table 1. Since the nature of DTLZ5 and DTLZ6s PFs is unclear beyond three objectives [73], here we only consider DTLZ 1–4 and DTLZ7 problems for the DTLZ test suite. The main features of these problems are summarized in Table 2. For DTLZ1–DTLZ4 and DTLZ7, the total number of decision variables is given by $n = m + k - 1$, where m is the number of objectives and k is set to 5 for DTLZ1, 10 for DTLZ2–DTLZ4, and 20 for DTLZ7. As for all WFG problems, the number of decision variables is set to 24 and the position-related parameter is set to $m - 1$ according to [33, 73].

4.2. Performance Metrics. In our experimental study, two widely used metrics are chosen to evaluate the performance of each algorithm, which are named the Inverted Generational Distance (IGD) [74] and Hypervolume (HV) [75]. The IGD and HV can measure both the convergence and diversity of obtained solutions effectively.

4.2.1. Inverted Generational Distance (IGD). Let P^* denote a set of uniformly distributed solutions in the objective space along the Pareto front. P is an approximation to the PF, which is obtained by the algorithm. The IGD is described as

$$IGD(P, P^*) = \frac{\sum_{i=1}^{|P^*|} \text{dist}(P_i^*, P)}{|P^*|}, \quad (15)$$

where $\text{dist}(P_i^*, P)$ is the Euclidean distance between a point $x^* \in P^*$ and its nearest neighbor in P , and $|P^*|$ is the cardinality of P^* . It can be seen from the definition of IGD that, for a large $|P^*|$, it can cover approximately the entire Pareto front, which is another aspect of metric in terms of diversity.

4.2.2. Hypervolume (HV). Consider the set of final non-dominated points S and a reference point $a = (a_1, a_2, \dots, a_m)^T$ in the objective space which is dominated by any point in the set S . Then, the hypervolume of S with regard to a can be described as follows:

$$HV(S, a) = \text{volume} \left(\bigcup_{f \in A} [f_1, a_1] \times \dots \times [f_m, a_m] \right). \quad (16)$$

It should be noted that choosing r that is slightly larger than the nadir point g_{nad} is suitable [76, 77]. Here, we set a to $1.1g_{\text{nad}}$, which is the same as in [33]. In addition, for problems with no more than 10 objectives, the recently proposed fast hypervolume calculation method is adopted to calculate the exact hypervolume [78]. As for problems having 15 objectives, the Monte Carlo method [37] with 1,000,000 sampling points is adopted, and all hypervolume values presented in this work are all normalized to [0,1].

4.3. Parameter Settings. As for the parameter settings, several general settings for algorithms are given as follows:

- (1) **Population Size.** The setting of the population size N for NSGA-III, MOEA/D, MOEA/DD, MOEA/DVA, and OBEA is controlled by a parameter H . Since MOMBI2 involves a binary tournament selection, we use the same population size as in NSGA-III or OBEA. Moreover, for the other algorithms, the population size is the same as above for ensuring a

TABLE 1: The statistical results (mean and standard deviation) of the HV values obtained by each algorithm on DTLZ1 to DTLZ4 and DTLZ7. The best results are italicized.

Problem	M	MOEA/D	dMOPSO	MOMBI2	ϵ -MOEA	OBEA
DTLZ1	3	3.994e-1 (3.36e-1) -	7.015e-3 (3.84e-2) -	5.819e-1 (3.01e-1) -	2.931e-1 (2.89e-1) -	8.471e-1 (6.13e-3)
	5	5.010e-1 (3.49e-1) -	2.856e-4 (1.56e-3) -	1.406e-1 (2.68e-1) -	8.164e-2 (2.00e-1) =	9.761e-1 (4.15e-4)
	8	5.824e-1 (2.20e-1) -	5.418e-3 (2.51e-2) -	2.923e-1 (3.37e-1) -	3.102e-1 (5.34e-3) -	9.524e-1 (6.84e-3)
	10	4.907e-1 (2.76e-1) -	1.031e-2 (2.31e-2) -	9.039e-1 (9.79e-2) =	8.356e-2 (1.53e-1) -	9.571e-1 (4.18e-3)
	15	3.566e-1 (2.33e-1) -	5.092e-3 (1.76e-2) -	2.195e-1 (2.87e-1) -	4.015e-1 (6.12e-2) -	9.456e-1 (4.46e-6)
DTLZ2	3	5.551e-1 (7.76e-4) -	3.671e-1 (1.60e-2) -	5.572e-1 (5.51e-4) +	5.409e-1 (2.93e-3) -	5.563e-1 (5.70e-4)
	5	7.823e-1 (5.77e-3) =	4.986e-1 (1.69e-2) -	7.881e-1 (3.38e-3) =	8.022e-1 (5.35e-3) +	7.829e-1 (3.48e-3)
	8	8.541e-1 (1.27e-2) -	3.945e-1 (2.67e-2) -	8.643e-1 (1.33e-2) -	8.422e-1 (2.30e-2) -	8.981e-1 (4.96e-3)
	10	8.384e-1 (2.34e-2) -	4.536e-1 (2.17e-2) -	7.993e-1 (2.28e-2) -	8.799e-1 (1.83e-2) -	9.071e-1 (1.22e-2)
	15	8.428e-1 (6.76e-2) -	2.953e-1 (2.68e-2) -	7.273e-1 (6.12e-2) -	7.828e-1 (4.00e-2) -	9.459e-1 (7.02e-3)
DTLZ3	3	3.806e-1 (6.33e-2) -	1.775e-3 (5.02e-3) -	1.001e-1 (1.38e-1) -	4.173e-1 (5.42e-1) -	5.212e-1 (4.53e-2)
	5	2.780e-1 (2.820e-1) -	2.563e-1 (2.52e-1) -	6.709e-1 (1.95e-1) =	2.316e-1 (3.55e-1) -	7.688e-1 (5.15e-2)
	8	2.736e-4 (1.50e-3) -	4.563e-1 (2.63e-2) =	2.670e-1 (3.22e-1) -	4.621e-1 (4.13e-2) =	5.475e-1 (3.15e-1)
	10	3.501e-2 (2.42e-1) -	2.511e-1 (3.44e-1) -	4.073e-1 (2.61e-1) =	4.012e-1 (7.27e-1) -	4.912e-1 (4.38e-1)
	15	5.370e-1 (6.42e-1) -	2.036e-1 (3.54e-1) -	1.996e-1 (2.01e-1) -	5.722e-1 (5.15e-1) -	9.879e-1 (6.17e-2)
DTLZ4	3	3.630e-1 (1.69e-1) -	3.023e-1 (4.80e-2) -	5.359e-1 (6.60e-2) +	3.963e-1 (1.53e-1) -	4.954e-1 (1.03e-1)
	5	5.863e-1 (1.70e-1) -	3.276e-1 (5.26e-2) -	7.872e-1 (2.44e-2) +	7.973e-1 (4.00e-2) +	7.814e-1 (8.95e-3)
	8	6.231e-1 (1.33e-1) -	3.726e-1 (5.68e-2) -	9.102e-1 (3.20e-2) -	9.199e-1 (1.62e-2) +	9.113e-1 (6.95e-3)
	10	6.928e-1 (1.79e-1) -	3.958e-1 (7.36e-2) -	9.505e-1 (8.43e-3) +	9.589e-1 (5.99e-3) +	9.396e-1 (6.00e-3)
	15	6.959e-1 (1.80e-1) -	3.394e-1 (1.02e-1) -	9.769e-1 (9.59e-3) -	9.819e-1 (5.12e-3) -	9.875e-1 (8.11e-4)
DTLZ7	3	2.288e-1 (1.06e-2) -	2.398e-1 (1.27e-2) -	2.620e-1 (1.66e-2) +	2.357e-1 (2.32e-2) -	2.526e-1 (6.40e-3)
	5	1.993e-2 (2.06e-2) -	1.805e-1 (1.10e-2) +	2.069e-1 (1.34e-2) +	1.694e-1 (2.42e-2) +	1.320e-1 (2.08e-2)
	8	2.283e-3 (6.87e-3) -	1.044e-1 (1.27e-2) +	9.639e-2 (2.69e-2) =	5.824e-2 (3.18e-2) -	9.463e-2 (1.67e-2)
	10	8.257e-5 (1.41e-4) -	1.194e-1 (1.63e-2) +	1.503e-1 (3.59e-3) +	6.109e-4 (2.42e-3) -	8.100e-2 (1.87e-3)
	15	8.583e-6 (2.73e-5) -	8.524e-2 (2.63e-2) +	1.159e-1 (3.30e-4) +	4.012e-4 (5.11e-3) -	5.400e-2 (3.06e-4)
w/l/t		0/24/1	4/20/1	8/12/5	5/18/2	

TABLE 2: The features of the test problems.

Problem	Features
DTLZ1	Linear, multimodal
DTLZ2	Concave
DTLZ3	Concave, multimodal
DTLZ4	Concave, biased
DTLZ7	Mixed, disconnected, multimodal
WFG1	Mixed, biased
WFG2	Convex, disconnected, multimodal, nonseparable
WFG3	Linear, degenerate, nonseparable
WFG4	Concave, multimodal
WFG5	Concave, deceptive
WFG6	Concave, nonseparable
WFG7	Concave, biased
WFG8	Concave, biased, nonseparable
WFG9	Concave, biased, multimodal, deceptive, nonseparable

fair comparison. Population sizes N used in this study for different number of objectives are listed in Table 3.

- (2) *Parameters for Operator.* Since the PBI function is involved in MOEA/D, dMOPSO, Two_Arch2, and MOEA/DD, the penalty parameter θ is set to 5 as suggested in [43]. Furthermore, as the simulated binary crossover (SBX) and polynomial mutation are employed to generate offspring solutions, the crossover probability p_c and mutation probability p_m are set to 1.0 and $1/n$, respectively. For the SBX operator,

TABLE 3: The setting of the population size.

Number of objectives (m)	Divisions (H_1, H_2)	Population size (N)
3	(12, 0)	91
5	(6, 0)	210
8	(3, 2)	156
10	(3, 2)	275
15	(2, 1)	135

its distribution index is $\eta_c = 30$, and the distribution index of the mutation operator is $\eta_m = 20$ [45].

- (3) *Parameter Settings for Algorithms.* Besides the parameters mentioned above, algorithms also have their specific parameters. These parameters are set mainly according to the suggestions given by their developers with the purpose of ensuring the impartiality and objectivity of the experiment. The details are shown below.

- (a) Parameter setting in MOEA/D: the neighborhood size T is set to 20 [43].
- (b) Parameter setting in MOEA/DD: the neighborhood size T is set to 20, and the probability used to select in the neighborhood is $\delta = 0.9$ [45].
- (c) Parameter setting in MOEA/DVA: the neighborhood size T is set to 20, the number of sampling solutions in control variable analysis is 20, and the maximum number of tries required to judge the interaction is 6 [71].

- (d) Parameter setting in dMOPSO: the age threshold T_a is set to 2 [66].
- (e) Parameter setting in MOMBI2: following the practice in [67], two parameters in MOMBI2 are set as $\epsilon = 1e-3$ and $\alpha = 0.5$, respectively [67].
- (f) Parameter setting in ϵ -MOEA: the parameter in grid location calculation is $\epsilon = 0.06$ [68].
- (g) Parameter setting in RVEA: in the experimental comparisons, $\alpha = 2$ and $f_r = 0.1$ are used for all test instances [53].
- (h) Parameter setting in Two_Arch2: the CA size is equal to the population size, and the fractional distance p is equal to $1/m$ (where m is the number of the objective) [49].
- (i) Parameter setting in OBEA: $\mu = 0.35$ is used for the adaptive clustering mechanism.

4.4. Experimental Results and Analysis. In this section, OBEA is compared with the four efficient evolutionary algorithms that are proposed in Section 4.4.1. Then, six state-of-the-art evolutionary algorithms are also included as the comparators to investigate the ability of OBEA for solving many-objective optimizations in Section 4.4.2. The analysis of the adaptive strategy design is included in Section 4.4.3, and the discussion of the effectiveness analysis of OBL and adaptive strategy is present in Section 4.4.5, followed by the comparisons between HD, TCH, and PBI in Section 4.4.4. Furthermore, investigation of the evolutionary behavior of OBEA on parts of test problems is also included in Section 4.4.6.

4.4.1. Comparison with the Previous Algorithms. Statistical results of the IGD values obtained by OBEA and four algorithms named MOEA/D, dMOPSO, MOMBI2, and ϵ -MOEA are summarized in Tables 4 and 5, where the best results are italicized. The significance of difference between OBEA and the peer algorithms is determined by using the Wilcoxon rank sum test, where “+,” “−,” and “=” indicate the competitor is better than, worse than, or similar to the proposed OBEA, respectively, and the results are summarized as “ w/l ,” which denotes that corresponding competitor wins on w functions, loses on l functions, and ties on t functions, compared with the proposed OBEA.

It can be observed that OBEA covers the majority of the best values among the compared algorithms on the five original DTLZ test instances, especially on DTLZ1, DTLZ3, and DTLZ7. As for the DTLZ2 test problem, the best values on 5-objective, 8-objective, and 10-objective are obtained by ϵ -MOEA and MOEA/D, respectively. However, the proposed OBEA tends to perform well on 3-objective and 15-objective test instances. While, for the DTLZ4 test problem, the performance of OBEA is little worse than MOMBI2 and ϵ -MOEA on 3-objective, 10-objective, and 15-objective instances, the proposed algorithm still outperforms the others on 5-objective instance and is similar to ϵ -MOEA on the 8-objective test instance according to the Wilcoxon rank sum test.

The statistical results in Table 5 also indicate that OBEA has achieved the best performance among the five algorithms on all WFG1 and WFG6–9 test instances. For WFG2

and WFG3 test problems, ϵ -MOEA and MOMBI2 show best performance on most test instances. However, the proposed OBEA tends to perform well on parts of instances. Furthermore, OBEA still outperforms the other algorithms like MOEA/D and dMOPSO on most instances according to the Wilcoxon rank sum test. As for the WFG4 and WFG5 test instances, ϵ -MOEA and OBEA cover all the best values among five algorithms, which also verify the best performance of the proposed algorithm.

As evidenced by statistical results of the HV values on DTLZ test suits summarized in Table 1, OBEA has covered the best values on DTLZ1 and DTLZ3 test problems. As for DTLZ2, the best values on 8-, 10-, and 15-objective instances have been obtained by OBEA, while MOMBI2 and ϵ -MOEA have achieved the best performance on different instances, respectively. Moreover, the best values on the 15-objective DTLZ4 test instance have been obtained by OBEA, which indicates the best performance of the proposed algorithm for high-dimensional problems. By contrast, the performance of OBEA on the DTLZ7 test functions is not as good as that on the DTLZ1 and DTLZ3 test functions; however, the performance of the proposed algorithm is superior to MOEA/D and ϵ -MOEA on most DTLZ7 test instances.

Similar observations can be made about the results on the WFG test problem, where OBEA has shown the most competitive performance on most test instances in Table 6, while MOMBI2 and ϵ -MOEA have also achieved the best performance on different instances, respectively. Above all, it can be concluded that OBEA can balance the diversity and convergence better than these four algorithms on most instances.

4.4.2. Comparison with the State-of-the-Art Algorithms. In this section, the proposed algorithm is compared with several state-of-the-art algorithms on the DTLZ and WFG test problems. The experimental results of each compared algorithms on the DTLZ and WFG test instances are shown in Tables 7 and 8, in terms of IGD metrics. The experimental results of HV on these instances are also demonstrated in Tables 9 and 10. Moreover, thirty independent runs are executed for these algorithms on the test problems with the Wilcoxon rank sum test at a significance level of 0.05.

It can be observed that OBEA obtained most of the best values among the compared algorithms on the five original DTLZ test instances. For DTLZ1, OBEA and MOEA/DVA show better performance than the other algorithms according to the Wilcoxon rank sum test. For the DTLZ2 and DTLZ7 test problems, although Two_Arch2 and MOEA/DD, together with MOEA/DVA, cover parts of the best values, OBEA still tends to outperform most algorithms on these instances. As for the DTLZ3, although NSGA-III and RVEA share the same idea with OBEA in using the reference line, OBEA is still superior to NSGA-III and RVEA, which may due to the effectiveness of selection strategy in our algorithm. As for the DTLZ7, MOEA/DD, Two_Arch2 and MOEA/DVA shows good performance on parts of instances. However, the proposed OBEA shows competitive results on the other high-dimensional instances.

TABLE 4: The statistical results (mean and standard deviation) of the IGD values obtained by MOEA/D, dMOPSO, MOBI2, ϵ -MOEA, and OBEA on DTLZ1 to DTLZ4 and DTLZ7. The best results are italicized.

Problem	M	MOEA/D	dMOPSO	MOMBI2	ϵ -MOEA	OBEA
DTLZ1	3	$2.569e-1$ ($2.50e-1$) =	$8.076e+0$ ($5.99e+0$) =	$1.421e-1$ ($1.63e-1$) =	$3.775e-1$ ($4.66e-1$) =	<i>$1.121e-2$ ($5.13e-4$)</i>
	5	$2.884e-1$ ($2.88e-1$) =	$1.129e+1$ ($4.53e+0$) =	$6.283e-1$ ($3.21e-1$) =	$8.831e-1$ ($5.86e-1$) =	<i>$5.101e-2$ ($4.27e-4$)</i>
	8	$1.937e-1$ ($9.66e-2$) =	$5.033e+0$ ($3.95e+0$) =	$4.950e-1$ ($2.04e-1$) =	$5.220e+0$ ($2.22e+0$) =	<i>$8.202e-2$ ($2.51e-3$)</i>
	10	$3.256e-1$ ($3.79e-1$) =	$7.812e+0$ ($3.13e+0$) =	$5.716e+0$ ($2.09e+0$) =	$9.107e-1$ ($5.31e-1$) =	<i>$1.312e-1$ ($3.09e-3$)</i>
	15	$2.433e-1$ ($1.20e-1$) =	$7.121e+0$ ($6.43e+0$) =	$7.148e-1$ ($5.39e-1$) =	$6.084e+0$ ($2.37e+0$) =	<i>$1.214e-1$ ($3.01e-3$)</i>
DTLZ2	3	$5.492e-2$ ($2.34e-4$) =	$1.425e-1$ ($8.05e-3$) =	$5.819e-2$ ($1.15e-3$) =	$7.177e-2$ ($2.78e-3$) =	<i>$5.479e-2$ ($9.14e-5$)</i>
	5	$1.668e-1$ ($1.91e-3$) =	$2.619e-1$ ($9.02e-3$) =	$2.094e-1$ ($4.86e-3$) =	$1.289e-1$ ($4.58e-3$) =	<i>$1.696e-1$ ($9.83e-4$)</i>
	8	$3.260e-1$ ($7.38e-3$) +	$5.501e-1$ ($1.88e-2$) =	$4.079e-1$ ($7.15e-3$) =	$3.102e-1$ ($1.52e-2$) +	<i>$3.335e-1$ ($2.51e-3$)</i>
	10	<i>$4.185e-1$ ($2.24e-2$) =</i>	$6.212e-1$ ($2.90e-2$) =	$4.373e-1$ ($1.67e-2$) =	$5.073e-1$ ($1.40e-2$) =	<i>$4.215e-1$ ($3.95e-3$)</i>
	15	$7.053e-1$ ($6.39e-2$) =	$9.488e-1$ ($3.17e-2$) =	$8.965e-1$ ($4.96e-2$) =	$6.734e-1$ ($3.81e-2$) =	<i>$5.885e-1$ ($5.97e-3$)</i>
DTLZ3	3	$1.388e+1$ ($9.64e+0$) =	$4.493e+1$ ($3.14e+1$) =	$6.321e+0$ ($5.45e+0$) =	$1.719e+1$ ($7.00e+0$) =	<i>$5.494e-1$ ($3.21e-2$)</i>
	5	$1.933e+1$ ($1.38e+1$) =	$1.898e+2$ ($3.36e+1$) =	$1.704e+1$ ($5.60e+0$) =	$2.985e+1$ ($1.07e+1$) =	<i>$1.782e-1$ ($2.51e-3$)</i>
	8	$6.610e+0$ ($4.88e+0$) =	$2.032e+2$ ($9.75e+0$) =	$1.459e+1$ ($5.64e+0$) =	$1.310e+2$ ($2.52e+1$) =	<i>$8.911e-1$ ($8.01e-1$)</i>
	10	$6.232e+0$ ($2.12e+0$) =	$2.031e+2$ ($1.22e+1$) =	$1.530e+2$ ($3.41e+1$) =	$3.215e+1$ ($7.78e+0$) =	<i>$1.197e+0$ ($4.32e-1$)</i>
	15	$6.469e+0$ ($3.99e+0$) =	$2.045e+2$ ($7.22e+0$) =	$1.794e+1$ ($9.92e+0$) =	$1.515e+2$ ($3.44e+1$) =	<i>$1.569e+0$ ($4.91e-1$)</i>
DTLZ4	3	$4.602e-1$ ($3.30e-1$) =	$3.173e-1$ ($2.84e-2$) =	<i>$1.085e-1$ ($1.47e-1$) +</i>	$3.695e-1$ ($3.02e-1$) =	<i>$1.851e-1$ ($2.20e-1$)</i>
	5	$5.470e-1$ ($2.31e-1$) =	$5.217e-1$ ($2.20e-2$) =	$2.246e-1$ ($5.14e-2$) =	$1.908e-1$ ($9.86e-2$) =	<i>$1.770e-1$ ($1.44e-2$)</i>
	8	$7.792e-1$ ($1.35e-1$) =	$6.505e-1$ ($2.01e-2$) =	$4.058e-1$ ($6.38e-2$) =	$3.423e-1$ ($5.58e-2$) =	<i>$3.478e-1$ ($1.95e-3$)</i>
	10	$8.279e-1$ ($1.38e-1$) =	$7.178e-1$ ($1.26e-2$) =	$4.213e-1$ ($2.13e-2$) +	$4.789e-1$ ($1.80e-2$) =	<i>$4.631e-1$ ($3.80e-3$)</i>
	15	$9.284e-1$ ($1.14e-1$) =	$8.328e-1$ ($2.13e-2$) =	$6.675e-1$ ($2.36e-2$) =	$5.324e-1$ ($1.82e-2$) +	<i>$6.306e-1$ ($1.38e-3$)</i>
DTLZ7	3	$1.907e-1$ ($1.67e-1$) =	$1.705e-1$ ($1.43e-1$) =	$1.840e-1$ ($1.25e-1$) =	$2.591e-1$ ($2.71e-1$) =	<i>$5.452e-2$ ($3.12e-3$)</i>
	5	$1.169e+0$ ($2.71e-1$) =	$6.097e-1$ ($2.00e-1$) =	$4.823e-1$ ($1.64e-1$) =	$6.667e-1$ ($2.53e-1$) =	<i>$4.141e-1$ ($5.13e-2$)</i>
	8	$1.659e+0$ ($2.34e-1$) =	$1.385e+0$ ($2.73e-1$) =	$4.324e+0$ ($1.23e+0$) =	$2.151e+0$ ($6.38e-1$) =	<i>$9.146e-1$ ($3.31e-2$)</i>
	10	$2.704e+0$ ($5.03e-1$) =	$2.252e+0$ ($4.09e-1$) =	$1.877e+1$ ($2.27e+0$) =	$1.007e+1$ ($1.48e+0$) =	<i>$1.452e+0$ ($6.76e-1$)</i>
	15	$3.181e+0$ ($6.21e-1$) =	$5.635e+0$ ($1.69e+0$) =	$2.844e+1$ ($4.67e+0$) =	$3.766e+1$ ($3.27e+0$) =	<i>$1.953e+0$ ($3.87e-1$)</i>
$w/l/t$		1/15/9	0/22/3	2/22/1	3/19/3	

TABLE 5: The statistical results (mean and standard deviation) of the IGD values obtained by MOEA/D, dMOPSO, MOBI2, ϵ -MOEA, and OBEA on WFG test suits. The best results are italicized.

Problem	M	MOEA/D	dMOPSO	MOMBI2	ϵ -MOEA	OBEA
WFG1	3	$1.299e+0$ ($8.45e-2$) =	$1.552e+0$ ($7.51e-3$) =	$1.094e+0$ ($7.15e-2$) =	$1.517e+0$ ($6.32e-2$) =	<i>$5.083e-1$ ($7.94e-2$)</i>
	5	$1.998e+0$ ($9.68e-2$) =	$2.210e+0$ ($3.04e-2$) =	$1.940e+0$ ($9.49e-2$) =	$2.038e+0$ ($1.74e-2$) =	<i>$9.814e-1$ ($1.56e-1$)</i>
	8	$2.840e+0$ ($1.72e-1$) =	$3.125e+0$ ($5.07e-2$) =	$2.635e+0$ ($2.10e-1$) =	$2.693e+0$ ($3.05e-2$) =	<i>$1.959e+0$ ($3.48e-1$)</i>
	10	$3.344e+0$ ($9.03e-2$) =	$3.566e+0$ ($5.15e-2$) =	$3.070e+0$ ($2.26e-2$) =	$3.051e+0$ ($1.86e-1$) =	<i>$2.088e+0$ ($4.77e-2$)</i>
	15	$4.294e+0$ ($8.37e-2$) =	$4.587e+0$ ($5.58e-2$) =	$3.136e+0$ ($3.54e-1$) =	$3.972e+0$ ($6.22e-2$) =	<i>$3.115e+0$ ($2.48e-1$)</i>
WFG2	3	$1.081e+0$ ($8.61e-2$) =	$9.277e-1$ ($1.79e-1$) =	$3.477e-1$ ($1.76e-2$) =	$2.916e-1$ ($5.33e-2$) =	<i>$2.747e-1$ ($6.59e-2$)</i>
	5	$5.835e+0$ ($2.62e-1$) =	$3.811e+0$ ($4.64e-1$) =	$1.348e+0$ ($1.68e-1$) =	$8.280e-1$ ($2.05e-1$) +	<i>$1.502e+0$ ($5.72e-1$)</i>
	8	$8.889e+0$ ($6.72e-1$) =	$6.697e+0$ ($4.23e-1$) =	$2.169e+0$ ($4.20e-1$) +	$1.482e+0$ ($2.97e-1$) +	<i>$3.321e+0$ ($8.42e-1$)</i>
	10	$1.696e+1$ ($2.00e-1$) =	$1.309e+1$ ($2.39e-1$) =	$3.028e+0$ ($4.88e-1$) +	$4.441e+0$ ($1.41e+0$) =	<i>$4.661e+0$ ($1.41e+0$)</i>
	15	$2.766e+1$ ($1.71e-1$) =	$2.274e+1$ ($1.33e+0$) =	$1.142e+1$ ($2.31e+0$) +	$3.401e+0$ ($6.14e-1$) +	<i>$1.632e+1$ ($3.19e+0$)</i>
WFG3	3	$4.820e-1$ ($1.29e-1$) =	$4.060e-1$ ($7.70e-2$) =	<i>$1.453e-1$ ($1.76e-2$) +</i>	$3.051e-1$ ($3.07e-2$) =	<i>$1.730e-1$ ($1.72e-2$)</i>
	5	$1.468e+0$ ($3.18e-1$) =	$7.413e-1$ ($4.80e-2$) =	$9.781e-1$ ($7.32e-2$) =	$7.482e-1$ ($3.16e-2$) =	<i>$5.355e-1$ ($7.74e-2$)</i>
	8	$4.279e+0$ ($2.87e-1$) =	$2.304e+0$ ($5.29e-1$) =	$3.362e+0$ ($4.40e-1$) =	<i>$1.107e+0$ ($3.13e-2$) +</i>	<i>$1.208e+0$ ($2.60e-1$)</i>
	10	$7.859e+0$ ($1.16e+0$) =	$3.300e+0$ ($8.09e-1$) =	<i>$1.263e+0$ ($5.16e-2$) +</i>	$3.607e+0$ ($5.57e-1$) =	<i>$1.464e+0$ ($2.81e-1$)</i>
	15	$1.337e+1$ ($1.89e+0$) =	$8.238e+0$ ($2.91e+0$) =	$1.035e+1$ ($1.37e+0$) =	<i>$1.493e+0$ ($8.05e-2$) +</i>	<i>$2.640e+0$ ($4.70e-1$)</i>
WFG4	3	$3.073e-1$ ($1.60e-2$) =	$4.085e-1$ ($2.38e-2$) =	$2.616e-1$ ($4.73e-3$) =	<i>$1.967e-1$ ($6.27e-3$) +</i>	<i>$2.407e-1$ ($3.55e-3$)</i>
	5	$2.095e+0$ ($1.43e-1$) =	$1.648e+0$ ($1.15e-1$) =	$1.851e+0$ ($1.09e-1$) =	<i>$8.292e-1$ ($5.25e-2$) +</i>	<i>$9.865e-1$ ($5.77e-3$)</i>
	8	$7.380e+0$ ($1.28e-1$) =	$7.561e+0$ ($3.05e-1$) =	$4.017e+0$ ($5.12e-1$) =	$3.549e+0$ ($2.22e-1$) =	<i>$2.985e+0$ ($1.58e-2$)</i>
	10	$9.817e+0$ ($2.22e-1$) =	$1.030e+1$ ($2.35e-1$) =	$5.938e+0$ ($3.00e-1$) =	$5.897e+0$ ($3.90e-1$) =	<i>$4.300e+0$ ($5.79e-2$)</i>
	15	$1.713e+1$ ($3.18e-1$) =	$1.672e+1$ ($1.74e-1$) =	$2.061e+1$ ($1.20e+0$) =	$1.316e+1$ ($4.78e-1$) =	<i>$8.888e+0$ ($1.17e-1$)</i>
WFG5	3	$3.099e-1$ ($1.66e-2$) =	$4.132e-1$ ($3.00e-2$) =	$2.748e-1$ ($3.00e-3$) =	<i>$2.138e-1$ ($9.41e-3$) +</i>	<i>$2.463e-1$ ($2.35e-3$)</i>
	5	$2.166e+0$ ($8.60e-2$) =	$1.314e+0$ ($2.07e-2$) =	$2.062e+0$ ($7.95e-2$) =	<i>$9.772e-1$ ($2.80e-2$) =</i>	<i>$9.842e-1$ ($8.25e-3$)</i>
	8	$7.144e+0$ ($7.49e-2$) =	$4.488e+0$ ($2.12e-1$) =	$3.767e+0$ ($3.57e-2$) =	$3.139e+0$ ($1.64e-1$) =	<i>$2.999e+0$ ($1.74e-2$)</i>
	10	$9.567e+0$ ($1.30e-1$) =	$5.826e+0$ ($2.32e-1$) =	$4.820e+0$ ($2.18e-1$) =	$5.511e+0$ ($1.82e-1$) =	<i>$4.266e+0$ ($4.17e-2$)</i>
	15	$1.631e+1$ ($1.30e-1$) =	$1.084e+1$ ($3.02e-1$) =	$2.156e+1$ ($1.93e+0$) =	$1.066e+1$ ($6.76e-1$) =	<i>$5.491e+0$ ($5.43e-2$)</i>

TABLE 5: Continued.

Problem	M	MOEAD	dMOPSO	MOMBI2	ε -MOEA	OBEA
WFG6	3	$3.453e-1$ ($2.07e-2$) –	$4.387e-1$ ($5.19e-2$) –	$2.842e-1$ ($7.16e-3$) –	$2.806e-1$ ($1.48e-2$) –	$2.729e-1$ ($8.43e-3$)
	5	$2.850e+0$ ($3.33e-1$) –	$2.372e+0$ ($1.64e-1$) –	$1.972e+0$ ($1.01e-1$) –	$1.125e+0$ ($2.99e-2$) –	$1.047e+0$ ($1.15e-2$)
	8	$7.849e+0$ ($1.55e-1$) –	$8.386e+0$ ($3.11e-1$) –	$3.760e+0$ ($8.83e-2$) –	$3.268e+0$ ($1.34e-1$) –	$3.044e+0$ ($1.51e-2$)
	10	$1.005e+1$ ($2.43e-1$) –	$1.049e+1$ ($2.93e-1$) –	$4.999e+0$ ($2.82e-1$) –	$5.465e+0$ ($9.81e-2$) –	$4.340e+0$ ($4.32e-2$)
	15	$1.672e+1$ ($1.88e-1$) –	$1.686e+1$ ($1.64e-1$) –	$1.929e+1$ ($1.73e+0$) –	$1.108e+1$ ($6.65e-1$) –	$5.619e+0$ ($4.12e-1$)
WFG7	3	$4.836e-1$ ($3.53e-2$) –	$4.494e-1$ ($1.34e-2$) –	$2.607e-1$ ($3.93e-3$) –	$2.035e-1$ ($1.26e-2$) –	$1.235e-1$ ($3.12e-4$)
	5	$2.716e+0$ ($3.13e-1$) –	$1.523e+0$ ($1.36e-1$) –	$2.024e+0$ ($1.06e-1$) –	$9.271e-1$ ($3.95e-2$) –	$9.132e-1$ ($7.13e-3$)
	8	$7.776e+0$ ($1.67e-1$) –	$6.017e+0$ ($1.26e+0$) –	$3.846e+0$ ($3.80e-1$) –	$3.218e+0$ ($2.54e-1$) –	$2.411e+0$ ($2.95e-2$)
	10	$1.005e+1$ ($1.88e-1$) –	$5.473e+0$ ($4.05e-1$) –	$5.523e-1$ ($1.38e-2$) =	$7.827e-1$ ($2.40e-2$) =	$5.117e-1$ ($1.60e-2$)
	15	$1.735e+1$ ($1.36e-1$) –	$1.394e+1$ ($1.98e+0$) –	$1.735e+1$ ($2.38e+0$) –	$1.203e+1$ ($7.03e-1$) –	$5.814e+0$ ($5.17e-1$)
WFG8	3	$3.797e-1$ ($2.78e-2$) –	$5.852e-1$ ($3.90e-2$) –	$3.184e-1$ ($7.85e-3$) –	$2.9223e-1$ ($7.87e-3$) =	$2.913e-1$ ($7.74e-3$)
	5	$2.131e+0$ ($2.08e-1$) –	$1.538e+0$ ($4.79e-2$) –	$2.246e+0$ ($6.38e-2$) –	$1.116e+0$ ($3.62e-2$) –	$1.049e+0$ ($6.13e-3$)
	8	$6.814e+0$ ($1.59e-1$) –	$6.756e+0$ ($4.51e-1$) –	$3.973e+0$ ($2.38e-1$) –	$3.523e+0$ ($1.35e-1$) –	$3.181e+0$ ($1.91e-2$)
	10	$9.115e+0$ ($2.44e-1$) –	$8.869e+0$ ($4.10e-1$) –	$5.228e+0$ ($2.71e-1$) –	$5.871e+0$ ($5.23e-1$) –	$4.762e+0$ ($8.61e-2$)
	15	$1.324e+1$ ($2.71e+0$) –	$1.545e+1$ ($3.51e-1$) –	$1.970e+1$ ($1.40e+0$) –	$1.138e+1$ ($5.51e-1$) –	$9.367e+0$ ($2.90e-1$)
WFG9	3	$4.680e-1$ ($6.68e-2$) –	$3.495e-1$ ($1.33e-2$) –	$2.710e-1$ ($1.06e-2$) =	$2.060e-1$ ($3.04e-2$) +	$1.321e-1$ ($5.13e-3$)
	5	$2.141e+0$ ($1.10e-1$) –	$2.030e+0$ ($8.89e-2$) –	$1.954e+0$ ($9.93e-2$) –	$1.015e+0$ ($4.68e-2$) –	$9.316e-1$ ($5.66e-3$)
	8	$7.046e+0$ ($2.36e-1$) –	$6.721e+0$ ($4.80e-1$) –	$3.715e+0$ ($4.70e-2$) –	$3.276e+0$ ($1.53e-1$) –	$3.104e+0$ ($4.92e-2$)
	10	$9.240e+0$ ($6.11e-1$) –	$8.124e+0$ ($7.77e-1$) –	$5.123e+0$ ($1.95e-1$) –	$5.399e+0$ ($7.48e-2$) –	$4.231e+0$ ($7.16e-2$)
	15	$1.436e+1$ ($2.01e+0$) –	$1.132e+1$ ($5.41e-1$) –	$2.143e+1$ ($1.63e+0$) –	$1.127e+1$ ($6.58e-1$) –	$5.793e+0$ ($6.84e-1$)
w/l/t		0/44/1	0/45/0	5/36/4	9/31/5	

TABLE 6: The statistical results (mean and standard deviation) of the HV values obtained by each algorithm on WFG test suits. The best results are italicized.

Problem	M	MOEAD	dMOPSO	MOMBI2	ε -MOEA	OBEA
WFG1	3	$3.380e-1$ ($3.00e-2$) –	$2.833e-1$ ($4.99e-3$) –	$4.312e-1$ ($2.66e-2$) –	$3.009e-1$ ($1.92e-2$) –	$7.613e-1$ ($3.46e-2$)
	5	$3.126e-1$ ($2.03e-2$) –	$2.247e-1$ ($1.19e-2$) –	$3.514e-1$ ($2.35e-2$) –	$2.940e-1$ ($2.49e-3$) –	$7.811e-1$ ($6.25e-1$)
	8	$3.256e-1$ ($4.42e-2$) =	$2.088e-1$ ($5.86e-3$) –	$3.482e-1$ ($6.54e-2$) =	$2.500e-1$ ($2.54e-3$) –	$4.913e-1$ ($1.18e-1$)
	10	$2.799e-1$ ($3.07e-2$) –	$1.957e-1$ ($8.73e-3$) –	$2.273e-1$ ($1.92e-3$) –	$2.774e-1$ ($2.62e-2$) –	$5.612e-1$ ($4.13e-2$)
	15	$2.222e-1$ ($1.94e-2$) –	$1.739e-1$ ($4.87e-3$) –	$5.879e-1$ ($9.09e-2$) –	$1.903e-1$ ($1.39e-3$) –	$9.913e-1$ ($3.14e-4$)
WFG2	3	$7.545e-1$ ($3.86e-2$) –	$7.539e-1$ ($1.35e-2$) –	$8.653e-1$ ($1.20e-2$) –	$8.672e-1$ ($2.67e-2$) –	$8.837e-1$ ($6.87e-3$)
	5	$7.459e-1$ ($4.20e-2$) –	$7.732e-1$ ($1.71e-2$) –	$9.332e-1$ ($1.13e-2$) +	$8.520e-1$ ($1.62e-2$) –	$8.870e-1$ ($1.17e-2$)
	8	$7.034e-1$ ($5.45e-2$) –	$6.770e-1$ ($3.34e-2$) –	$9.312e-1$ ($3.45e-2$) +	$7.917e-1$ ($2.16e-2$) –	$9.106e-1$ ($1.58e-2$)
	10	$7.050e-1$ ($5.43e-2$) –	$6.484e-1$ ($2.65e-2$) –	$7.580e-1$ ($1.78e-2$) –	$9.100e-1$ ($5.63e-2$) =	$9.086e-1$ ($1.56e-2$)
	15	$7.480e-1$ ($9.20e-2$) –	$5.836e-1$ ($5.33e-2$) –	$7.096e-1$ ($1.05e-1$) –	$7.998e-1$ ($2.76e-2$) –	$9.543e-1$ ($1.07e-2$)
WFG3	3	$2.189e-1$ ($3.28e-2$) –	$2.512e-1$ ($1.20e-2$) –	$3.648e-1$ ($8.42e-3$) +	$2.672e-1$ ($1.62e-2$) –	$3.475e-1$ ($8.44e-3$)
	5	$1.511e-2$ ($4.62e-2$) –	$3.024e-2$ ($1.68e-2$) –	$4.616e-2$ ($1.10e-2$) –	$2.112e-4$ ($8.31e-4$) –	$1.769e-1$ ($1.28e-3$)
	8	$1.201e-2$ ($6.64e-3$) –	$1.101e-1$ ($6.61e-2$) =	$2.821e-3$ ($8.30e-3$) –	$1.341e-2$ ($6.12e-3$) =	$1.131e-1$ ($2.43e-3$)
	10	$3.323e-1$ ($5.13e-2$) –	$1.012e-1$ ($2.71e-2$) –	$7.487e-1$ ($2.71e-2$) =	$5.451e-1$ ($6.35e-2$) –	$6.945e-1$ ($4.55e-2$)
	15	$2.013e-1$ ($5.61e-2$) =	$2.353e-1$ ($6.89e-2$) =	$3.267e-1$ ($3.82e-2$) =	$3.312e-1$ ($5.61e-2$) =	$3.455e-1$ ($4.69e-2$)
WFG4	3	$4.570e-1$ ($1.28e-2$) –	$4.055e-1$ ($1.01e-2$) –	$5.147e-1$ ($5.10e-3$) =	$5.103e-1$ ($3.38e-3$) =	$5.107e-1$ ($2.91e-3$)
	5	$4.955e-1$ ($3.67e-2$) –	$4.923e-1$ ($2.23e-2$) –	$5.887e-1$ ($3.28e-2$) –	$6.235e-1$ ($1.16e-2$) –	$6.785e-1$ ($5.15e-3$)
	8	$2.983e-1$ ($3.58e-2$) –	$2.195e-1$ ($3.11e-2$) –	$7.457e-1$ ($5.72e-2$) –	$6.187e-1$ ($1.72e-2$) –	$7.707e-1$ ($7.74e-3$)
	10	$2.696e-1$ ($4.86e-2$) –	$1.882e-1$ ($3.08e-2$) –	$6.134e-1$ ($1.68e-2$) –	$7.440e-1$ ($3.08e-2$) –	$7.690e-1$ ($1.19e-2$)
	15	$1.873e-1$ ($6.01e-2$) –	$6.377e-2$ ($4.26e-2$) –	$4.349e-1$ ($6.33e-2$) –	$6.256e-1$ ($1.15e-2$) –	$8.486e-1$ ($1.13e-2$)
WFG5	3	$4.399e-1$ ($1.37e-2$) –	$3.937e-1$ ($1.60e-2$) –	$4.785e-1$ ($5.36e-3$) –	$4.893e-1$ ($3.76e-3$) –	$4.939e-1$ ($4.50e-3$)
	5	$4.896e-1$ ($3.04e-2$) –	$4.669e-1$ ($1.35e-2$) –	$4.992e-1$ ($2.96e-2$) –	$5.023e-1$ ($9.26e-3$) –	$6.303e-1$ ($9.01e-3$)
	8	$3.250e-1$ ($4.38e-2$) –	$2.058e-1$ ($6.14e-2$) –	$7.237e-1$ ($1.47e-2$) =	$5.012e-1$ ($1.48e-2$) –	$7.098e-1$ ($1.44e-2$)
	10	$2.881e-1$ ($4.47e-2$) –	$1.902e-1$ ($7.04e-2$) –	$4.987e-1$ ($1.31e-2$) –	$7.131e-1$ ($2.16e-2$) +	$6.377e-1$ ($1.79e-2$)
	15	$1.560e-1$ ($5.02e-2$) –	$7.941e-2$ ($7.57e-2$) –	$3.269e-1$ ($6.71e-2$) –	$5.050e-1$ ($1.76e-2$) –	$7.498e-1$ ($2.03e-2$)
WFG6	3	$4.292e-1$ ($1.30e-2$) –	$4.035e-1$ ($3.33e-2$) –	$4.805e-1$ ($7.82e-3$) –	$4.625e-1$ ($7.54e-3$) –	$5.160e-1$ ($1.14e-2$)
	5	$3.586e-1$ ($6.06e-2$) –	$3.342e-1$ ($2.80e-2$) –	$5.172e-1$ ($3.11e-2$) –	$4.488e-1$ ($1.01e-2$) –	$6.927e-1$ ($1.50e-2$)
	8	$1.380e-1$ ($3.90e-2$) –	$1.802e-1$ ($4.98e-2$) –	$7.156e-1$ ($1.66e-2$) =	$4.533e-1$ ($1.18e-2$) –	$7.598e-1$ ($1.81e-2$)
	10	$1.572e-1$ ($7.08e-2$) –	$1.697e-1$ ($4.19e-2$) –	$4.502e-1$ ($1.34e-2$) –	$6.951e-1$ ($2.58e-2$) =	$7.557e-1$ ($2.25e-2$)
	15	$8.926e-2$ ($3.35e-2$) –	$3.924e-2$ ($3.06e-2$) –	$4.032e-1$ ($5.48e-2$) –	$4.611e-1$ ($2.03e-2$) –	$7.229e-1$ ($3.09e-2$)

TABLE 6: Continued.

Problem	M	MOEAD	dMOPSO	MOMBI2	ϵ -MOEA	OBEA
WFG7	3	$3.592e-1$ ($2.07e-2$) -	$3.550e-1$ ($8.31e-3$) -	$5.193e-1$ ($5.85e-3$) =	$5.171e-1$ ($6.69e-3$) =	$5.148e-1$ ($2.33e-3$)
	5	$4.289e-1$ ($5.14e-2$) -	$4.405e-1$ ($1.61e-2$) -	$5.614e-1$ ($2.86e-2$) -	$5.455e-1$ ($1.10e-2$) -	$7.182e-1$ ($5.40e-3$)
	8	$2.231e-1$ ($6.34e-2$) -	$2.951e-1$ ($7.75e-2$) -	$7.746e-1$ ($4.95e-2$) =	$5.460e-1$ ($1.95e-2$) -	$8.113e-1$ ($6.13e-3$)
	10	$1.791e-1$ ($5.47e-2$) -	$4.767e-1$ ($3.60e-2$) -	$5.523e-1$ ($1.38e-2$) -	$7.827e-1$ ($2.40e-2$) =	$8.130e-1$ ($4.13e-2$)
	15	$1.144e-1$ ($1.20e-2$) -	$1.424e-1$ ($1.37e-1$) -	$5.123e-1$ ($9.04e-2$) -	$5.605e-1$ ($2.20e-2$) -	$8.167e-1$ ($4.01e-2$)
WFG8	3	$3.974e-1$ ($1.82e-2$) -	$2.961e-1$ ($1.34e-2$) -	$4.507e-1$ ($8.06e-3$) -	$4.492e-1$ ($4.65e-3$) -	$5.122e-1$ ($5.15e-2$)
	5	$2.966e-1$ ($6.96e-2$) -	$3.578e-1$ ($1.01e-2$) -	$3.722e-1$ ($1.95e-2$) -	$4.931e-1$ ($1.04e-2$) -	$6.915e-1$ ($3.63e-3$)
	8	$1.558e-2$ ($1.85e-2$) -	$7.033e-2$ ($2.80e-2$) -	$5.993e-1$ ($1.84e-2$) -	$4.872e-1$ ($1.54e-2$) -	$6.124e-1$ ($6.71e-2$)
	10	$1.407e-2$ ($2.15e-2$) -	$5.938e-2$ ($1.78e-2$) -	$4.974e-1$ ($1.14e-2$) -	$6.578e-1$ ($1.43e-2$) =	$7.184e-1$ ($5.17e-2$)
	15	$2.826e-1$ ($3.26e-1$) -	$4.243e-2$ ($2.09e-2$) -	$3.664e-1$ ($6.96e-2$) -	$4.970e-1$ ($2.15e-2$) -	$7.841e-1$ ($5.17e-2$)
WFG9	3	$3.418e-1$ ($3.85e-2$) -	$4.255e-1$ ($7.30e-3$) -	$4.817e-1$ ($1.20e-2$) =	$4.907e-1$ ($1.88e-2$) =	$5.611e-1$ ($6.12e-3$)
	5	$4.198e-1$ ($5.69e-2$) -	$4.868e-1$ ($1.86e-2$) -	$4.522e-1$ ($3.79e-2$) -	$5.004e-1$ ($2.00e-2$) -	$6.655e-1$ ($4.17e-2$)
	8	$2.284e-1$ ($8.51e-2$) -	$2.534e-1$ ($3.38e-2$) -	$6.782e-1$ ($3.63e-2$) =	$4.898e-1$ ($1.31e-2$) -	$7.012e-1$ ($6.32e-2$)
	10	$1.441e-1$ ($7.10e-2$) -	$2.545e-1$ ($2.71e-2$) -	$4.898e-1$ ($1.66e-2$) -	$6.675e-1$ ($3.17e-2$) =	$6.791e-1$ ($4.17e-2$)
	15	$2.060e-1$ ($1.80e-1$) -	$2.494e-1$ ($8.73e-2$) -	$3.918e-1$ ($7.50e-2$) -	$5.196e-1$ ($2.05e-2$) -	$7.824e-1$ ($3.42e-2$)
$w/l/t$		0/43/2	0/43/2	3/32/10	1/34/10	

TABLE 7: The statistical results (mean and standard deviation) of the IGD values obtained by RVEA, NSGA-III, SPEAR, MOEA/DD, Two_Arch2, MOEA/DVA, and OBEA on DTLZ test suits. The best results are italicized.

Problem	M	RVEA	NSGA-III	SPEAR	MOEA/DD	Two_Arch2	MOEA/DVA	OBEA
DTLZ1	3	$4.997e-1$ ($3.35e-1$) -	$2.608e-1$ ($2.03e-1$) -	$1.100e-1$ ($1.30e-1$) -	$4.459e-1$ ($2.52e-1$) -	$4.650e-2$ ($2.52e-2$) -	$3.599e-1$ ($2.57e-1$) -	$1.121e-2$ ($5.13e-4$)
	5	$1.102e+0$ ($5.51e-1$) -	$1.511e+0$ ($5.80e-1$) -	$1.416e-1$ ($7.28e-2$) -	$1.030e+0$ ($4.11e-1$) -	$6.561e-2$ ($6.73e-3$) -	$5.957e-2$ ($2.36e-3$) -	$5.101e-2$ ($4.27e-4$)
	8	$6.312e-1$ ($3.18e-1$) -	$1.960e+0$ ($6.81e-1$) -	$2.331e-1$ ($6.85e-2$) -	$5.884e-1$ ($2.74e-1$) -	$1.190e-1$ ($1.61e-2$) -	$1.034e-1$ ($1.24e-3$) -	$8.202e-2$ ($2.51e-3$)
	10	$9.947e-1$ ($3.36e-1$) -	$3.182e+0$ ($1.57e+0$) -	$5.290e-1$ ($2.83e-1$) -	$1.057e+0$ ($6.06e-1$) -	$1.453e-1$ ($2.12e-2$) -	$1.119e-1$ ($7.27e-4$) =	$1.312e-1$ ($3.09e-3$)
	15	$3.456e-1$ ($1.19e-1$) =	$1.189e+0$ ($5.61e-1$) -	$7.744e+0$ ($4.56e+0$) -	$3.937e-1$ ($2.88e-1$) =	$2.518e-1$ ($1.01e-1$) -	$1.332e-1$ ($1.88e-3$) =	$1.214e-1$ ($3.01e-3$)
DTLZ2	3	$5.569e-2$ ($4.96e-4$) -	$5.495e-2$ ($1.42e-4$) -	$7.230e-2$ ($2.57e-3$) -	$5.541e-2$ ($4.55e-4$) =	$3.644e-2$ ($6.21e-4$) +	$5.164e-2$ ($9.58e-4$) =	$5.479e-2$ ($9.14e-5$)
	5	$1.759e-1$ ($2.14e-3$) -	$1.859e-1$ ($3.63e-3$) -	$2.154e-1$ ($1.13e-2$) -	$1.725e-1$ ($1.76e-3$) =	$2.106e-1$ ($1.61e-3$) -	$2.044e-1$ ($2.70e-3$) -	$1.696e-1$ ($9.83e-4$)
	8	$3.289e-1$ ($1.99e-3$) +	$4.330e-1$ ($1.00e-1$) -	$4.297e-1$ ($4.62e-2$) -	$3.285e-1$ ($2.60e-3$) +	$3.512e-1$ ($2.88e-3$) =	$3.167e-1$ ($1.98e-3$) =	$3.335e-1$ ($2.51e-3$)
	10	$4.239e-1$ ($4.74e-3$) =	$6.047e-1$ ($8.69e-2$) -	$8.016e-1$ ($4.78e-2$) -	$4.233e-1$ ($5.31e-3$) =	$4.788e-1$ ($6.39e-3$) -	$4.689e-1$ ($2.62e-3$) -	$4.215e-1$ ($3.95e-3$)
	15	$5.968e-1$ ($1.77e-2$) =	$7.367e-1$ ($5.21e-2$) -	$7.433e-1$ ($4.78e-2$) -	$6.068e-1$ ($2.31e-2$) -	$6.179e-1$ ($6.59e-3$) -	$5.086e-1$ ($2.58e-3$) +	$5.885e-1$ ($5.97e-3$)
DTLZ3	3	$1.546e+1$ ($4.20e+0$) -	$1.038e+1$ ($3.51e+0$) -	$8.519e-1$ ($7.73e-1$) =	$1.794e+1$ ($5.90e+0$) -	$2.827e+0$ ($1.13e+0$) -	$5.682e+1$ ($1.18e+1$) -	$5.494e-1$ ($3.21e-2$)
	5	$4.328e+1$ ($1.02e+1$) -	$5.736e+1$ ($1.20e+1$) -	$1.165e+1$ ($6.04e+0$) -	$5.004e+1$ ($1.05e+1$) -	$1.779e+0$ ($1.23e+0$) -	$4.175e-1$ ($2.50e-1$) -	$1.782e-1$ ($2.51e-3$)
	8	$2.744e+1$ ($9.59e+0$) -	$7.911e+1$ ($2.93e+1$) -	$4.192e+0$ ($2.77e+0$) -	$3.024e+1$ ($8.72e+0$) -	$7.705e+0$ ($2.79e+0$) -	$2.086e+0$ ($1.77e+0$) -	$8.911e-1$ ($8.01e-1$)
	10	$4.608e+1$ ($1.38e+1$) -	$1.403e+2$ ($2.52e+1$) -	$3.490e+1$ ($1.11e+1$) -	$5.597e+0$ ($1.09e+1$) -	$1.484e+1$ ($2.95e+0$) -	$2.567e+0$ ($9.06e-1$) =	$1.197e+0$ ($4.32e-1$)
	15	$1.996e+1$ ($7.94e+0$) -	$1.360e+2$ ($5.17e+1$) -	$1.968e+2$ ($6.42e+1$) -	$2.072e+0$ ($9.46e+0$) -	$4.698e+1$ ($9.88e+0$) -	$2.117e+0$ ($1.20e+0$) -	$1.569e+0$ ($4.91e-1$)
DTLZ4	3	$5.607e-2$ ($6.32e-4$) +	$1.686e-1$ ($2.09e-1$) +	$8.126e-2$ ($1.65e-2$) +	$5.548e-2$ ($3.79e-4$) +	$3.623e-2$ ($4.36e-4$) +	$5.949e-1$ ($1.81e-3$) -	$1.851e-1$ ($2.20e-1$)
	5	$1.794e-1$ ($3.97e-3$) -	$1.997e-1$ ($4.50e-2$) -	$4.009e-1$ ($2.69e-2$) -	$1.780e-1$ ($4.35e-3$) =	$1.663e-1$ ($2.42e-3$) =	$1.558e-1$ ($6.58e-4$) =	$1.770e-1$ ($1.44e-2$)
	8	$3.489e-1$ ($1.15e-2$) =	$4.614e-1$ ($9.89e-2$) -	$5.482e-1$ ($3.18e-2$) -	$3.539e-1$ ($2.72e-2$) =	$3.510e-1$ ($3.34e-3$) =	$3.192e-1$ ($3.70e-4$) =	$3.478e-1$ ($1.95e-3$)
	10	$4.640e-1$ ($4.80e-3$) =	$5.677e-1$ ($4.65e-2$) -	$6.921e-1$ ($2.76e-2$) -	$4.763e-1$ ($1.28e-2$) =	$4.285e-1$ ($3.09e-3$) =	$4.713e-1$ ($2.57e-2$) -	$4.631e-1$ ($3.80e-3$)
	15	$6.366e-1$ ($9.63e-3$) =	$8.185e-1$ ($6.48e-2$) -	$8.189e-1$ ($4.44e-2$) -	$6.413e-1$ ($1.21e-2$) -	$5.670e-1$ ($1.11e-2$) =	$5.285e-1$ ($1.30e-2$) +	$6.306e-1$ ($1.38e-3$)

TABLE 7: Continued.

Problem	M	RVEA	NSGA-III	SPEAR	MOEA/DD	Two_Arch2	MOEA/DVA	OBEA
DTLZ7	3	1.934e-1 (3.76e-2) -	1.204e-1 (7.52e-2) =	2.814e-1 (7.45e-2) -	5.516e-1 (2.48e-1) =	4.901e-2 (3.96e-3) +	4.045e-2 (1.08e-3) +	5.452e-2 (3.12e-3)
	5	8.014e-1 (1.00e-1) -	8.126e-1 (1.19e-1) -	1.460e+0 (4.28e-1) -	2.045e+0 (4.82e-1) +	2.046e-1 (9.47e-3) +	3.023e-1 (2.62e-2) +	4.141e-1 (5.13e-2)
	8	1.621e+0 (6.52e-1) =	4.793e+0 (7.44e-1) -	5.423e+0 (1.09e+0) -	1.574e+0 (4.65e-1) -	6.024e-1 (1.86e-2) +	1.282e+0 (2.53e-1) -	9.146e-1 (3.31e-2)
	10	3.657e+0 (1.87e+0) =	1.325e+1 (1.41e+0) -	1.656e+1 (2.73e+0) -	1.530e+0 (1.13e-1) =	1.614e+0 (4.31e-2) =	2.094e+0 (2.68e-1) =	1.452e+0 (6.76e-1)
	15	5.228e+0 (2.67e+0) =	1.779e+1 (2.38e+0) -	1.896e+1 (1.78e+0) -	3.201e+0 (1.22e-1) -	1.078e+1 (7.13e-1) -	3.241e+0 (3.75e-1) -	1.953e+0 (3.87e-1)
	w/l/t	2/14/9	1/23/1	2/22/1	3/13/9	5/14/6	4/15/6	

TABLE 8: The statistical results (mean and standard deviation) of the IGD values obtained by RVEA, NSGA-III, SPEAR, MOEA/DD, Two_Arch2, MOEA/DVA, and OBEA on WFG test suits. The best results are italicized.

Problem	M	RVEA	NSGA-III	SPEAR	MOEA/DD	Two_Arch2	MOEA/DVA	OBEA
WFG1	3	1.353e+0 (5.32e-2) -	1.351e+0 (3.94e-2) -	1.943e+0 (3.57e-2) -	1.505e+0 (5.41e-2) -	1.158e+0 (4.94e-2) -	1.358e+0 (2.63e-2) -	5.083e-1 (7.94e-2)
	5	1.924e+0 (3.36e-2) -	1.992e+0 (3.41e-2) -	2.387e+0 (4.86e-2) -	2.201e+0 (1.03e-1) -	1.627e+0 (7.58e-2) -	1.733e+0 (3.00e-2) -	9.814e-1 (1.56e-1)
	8	2.564e+0 (5.35e-2) -	2.686e+0 (4.65e-2) =	3.072e+0 (3.53e-1) -	2.790e+0 (3.35e-2) -	2.436e+0 (5.48e-2) -	2.780e+0 (6.81e-2) -	1.959e+0 (3.48e-1)
	10	2.924e+0 (5.26e-2) -	3.154e+0 (5.28e-2) -	3.327e+0 (8.51e-2) -	3.108e+0 (3.10e-2) =	2.799e+0 (3.64e-2) -	3.252e+0 (5.35e-2) -	2.088e+0 (4.77e-2)
	15	3.518e+0 (2.28e-1) =	4.022e+0 (1.91e-1) -	3.997e+0 (2.70e-1) =	4.272e+0 (7.06e-2) =	3.790e+0 (3.90e-2) -	4.230e+0 (7.38e-2) -	3.115e+0 (2.48e-1)
WFG2	3	3.055e-1 (2.62e-2) -	2.262e-1 (1.78e-2) -	7.566e-1 (2.46e-1) -	4.168e-1 (1.08e-1) -	1.210e-1 (5.90e-3) +	4.002e-1 (2.45e-2) -	2.747e-1 (6.59e-2)
	5	1.190e+0 (4.01e-1) =	6.904e-1 (6.05e-2) -	4.437e+0 (1.64e+0) -	2.753e+0 (4.53e-1) -	6.203e-1 (1.06e-1) +	1.372e+0 (1.77e-1) =	1.502e+0 (5.72e-1)
	8	4.110e+0 (9.56e-1) -	1.084e+0 (1.17e-1) +	5.338e+0 (1.98e+0) -	4.883e+0 (8.37e-1) -	1.331e+0 (4.07e-1) =	4.234e+0 (3.15e-1) -	3.321e+0 (8.42e-1)
	10	6.072e+0 (2.26e+0) =	1.379e+0 (1.47e-1) +	5.298e+0 (1.63e+0) =	8.855e+0 (1.82e+0) -	2.141e+0 (9.47e-2) +	8.325e+0 (3.81e+0) -	4.661e+0 (1.41e+0)
	15	1.213e+1 (3.43e+0) =	4.156e+0 (1.46e+0) -	2.602e+0 (6.92e-1) +	1.776e+1 (2.05e+0) =	2.201e+0 (7.24e-1) +	2.078e+1 (4.53e+0) -	1.632e+1 (3.19e+0)
WFG3	3	3.070e-1 (2.98e-2) -	2.471e-1 (3.06e-3) -	4.032e-1 (5.37e-2) -	3.944e-1 (9.74e-2) -	1.153e-1 (1.33e-2) =	8.484e-2 (7.19e-3) +	1.730e-1 (1.72e-2)
	5	7.938e-1 (1.39e-1) -	1.005e+0 (7.46e-3) -	8.296e-1 (1.26e-1) -	7.444e-1 (5.32e-2) -	3.525e-1 (1.92e-2) =	1.225e+0 (1.48e-1) -	5.355e-1 (7.74e-2)
	8	1.895e+0 (3.00e-1) -	3.039e+0 (8.46e-2) -	5.639e+0 (4.82e-1) -	2.037e+0 (1.45e-1) =	9.309e-1 (5.24e-2) =	3.879e+0 (1.91e-1) -	1.208e+0 (2.60e-1)
	10	2.323e+0 (2.08e-1) -	4.409e+0 (1.46e-1) -	5.276e+0 (1.03e+0) -	2.588e+0 (1.18e-1) =	1.228e+0 (1.65e-1) =	5.244e+0 (2.45e-1) -	1.464e+0 (2.81e-1)
	15	5.784e+0 (7.08e-1) -	9.537e+0 (4.60e-1) -	3.226e+0 (2.48e-1) -	5.694e+0 (2.75e-1) -	2.829e+0 (1.45e-1) -	1.229e+1 (2.95e+0) -	2.640e+0 (4.70e-1)
WFG4	3	2.816e-1 (8.35e-3) -	2.512e-1 (3.64e-3) -	3.248e-1 (1.22e-2) -	2.478e-1 (2.46e-3) =	2.626e-1 (3.78e-3) -	1.972e-1 (5.39e-3) +	2.407e-1 (3.55e-3)
	5	9.940e-1 (5.27e-3) -	1.033e+0 (9.27e-3) -	1.098e+0 (2.28e-2) -	1.057e+0 (5.30e-3) -	9.397e-1 (7.75e-3) =	1.391e+0 (9.82e-2) -	9.865e-1 (5.77e-3)
	8	3.151e+0 (8.52e-2) -	3.050e+0 (3.27e-2) -	3.219e+0 (6.05e-2) -	3.507e+0 (1.71e-1) -	2.938e+0 (9.92e-3) =	4.110e+0 (9.22e-2) -	2.985e+0 (1.58e-2)
	10	4.430e+0 (1.82e-1) -	4.368e+0 (5.06e-2) -	4.458e+0 (1.05e-1) -	4.424e+0 (7.57e-2) =	4.335e+0 (2.19e-2) =	5.886e+0 (2.28e-1) -	4.300e+0 (5.79e-2)
	15	9.522e+0 (7.92e-1) -	9.014e+0 (3.68e-1) -	9.325e+0 (2.04e-1) -	9.363e+0 (8.00e-1) =	8.384e+0 (1.43e-2) +	1.070e+1 (1.17e-1) -	8.888e+0 (1.17e-1)

TABLE 8: Continued.

Problem	M	RVEA	NSGA-III	SPEAR	MOEA/DD	Two_Arch2	MOEA/DVA	OBEA
WFG5	3	2.840e-1 (1.06e-2) -	2.837e-1 (1.27e-2) -	3.226e-1 (1.51e-2) -	2.523e-1 (3.85e-3) =	1.767e-1 (4.10e-3) -	2.857e-1 (5.13e-3) =	2.463e-1 (2.35e-3)
	5	1.015e+0 (8.88e-3) -	1.093e+0 (1.84e-2) -	1.041e+0 (1.75e-2) -	1.051e+0 (6.78e-3) -	9.864e-1 (5.11e-3) =	1.311e+0 (1.17e-1) -	9.842e-1 (8.25e-3)
	8	3.257e+0 (4.78e-2) -	3.102e+0 (3.54e-2) -	3.257e+0 (5.83e-2) -	3.443e+0 (1.15e-1) -	2.912e+0 (2.35e-2) =	3.902e+0 (1.61e-1) =	2.999e+0 (1.74e-2)
	10	4.169e+0 (2.21e-2) +	4.459e+0 (8.45e-2) -	4.540e+0 (1.23e-1) -	4.472e+0 (1.01e-1) =	4.324e+0 (1.67e-2) =	6.169e+0 (1.46e-1) -	4.266e+0 (4.17e-2)
	15	8.616e+0 (3.72e-1) =	9.206e+0 (2.96e-1) -	9.056e+0 (7.88e-2) -	9.132e+0 (6.34e-1) -	8.047e+0 (8.66e-2) -	1.104e+1 (2.19e-1) -	5.491e+0 (5.43e-2)
WFG6	3	3.475e-1 (1.15e-2) -	2.528e-1 (6.39e-3) -	3.671e-1 (3.40e-2) -	3.273e-1 (4.27e-2) =	2.101e-1 (1.10e-2) -	2.562e-1 (6.03e-3) =	2.729e-1 (8.43e-3)
	5	1.074e+0 (1.65e-2) -	1.056e+0 (1.08e-2) -	1.198e+0 (3.44e-2) -	1.095e+0 (1.09e-2) =	9.695e-1 (1.48e-2) =	1.780e+0 (1.40e-1) =	1.047e+0 (1.15e-2)
	8	3.378e+0 (7.79e-2) -	3.058e+0 (4.75e-2) -	3.274e+0 (6.16e-2) -	3.363e+0 (6.95e-2) =	2.989e+0 (2.55e-2) =	4.771e+0 (2.08e-1) =	3.044e+0 (1.51e-2)
	10	4.221e+0 (4.75e-2) +	6.636e-1 (2.32e-2) -	4.588e+0 (1.98e-1) -	4.487e+0 (7.94e-2) =	4.406e+0 (2.91e-2) -	6.915e+0 (5.56e-2) -	4.340e+0 (4.32e-2)
	15	8.478e+0 (4.55e-1) -	1.024e+1 (4.25e-1) -	9.376e+0 (9.01e-2) -	8.911e+0 (1.81e-1) -	8.161e+0 (9.22e-2) -	1.279e+1 (2.08e-1) -	5.619e+0 (4.12e-1)
WFG7	3	3.382e-1 (3.13e-2) -	2.984e-1 (8.05e-3) -	3.286e-1 (1.14e-2) -	2.677e-1 (1.59e-2) -	1.524e-1 (3.15e-3) -	2.018e-1 (7.59e-3) -	1.235e-1 (3.12e-4)
	5	1.037e+0 (1.43e-2) -	1.089e+0 (1.31e-2) -	1.133e+0 (2.00e-2) -	1.083e+0 (9.64e-3) =	9.253e-1 (3.99e-3) =	1.771e+0 (1.28e-1) -	9.132e-1 (7.13e-3)
	8	3.216e+0 (6.04e-2) -	3.404e+0 (2.32e-1) -	3.245e+0 (4.64e-2) -	3.279e+0 (5.50e-2) =	2.897e+0 (3.33e-2) -	4.807e+0 (1.21e-1) -	2.411e+0 (2.95e-2)
	10	6.658e-1 (2.40e-2) -	4.936e+0 (2.81e-1) =	6.574e-1 (1.76e-2) -	4.403e+0 (5.19e-2) -	4.299e+0 (1.62e-2) -	6.881e+0 (2.49e-1) -	5.117e-1 (1.60e-2)
	15	9.214e+0 (1.72e-1) -	1.090e+1 (7.17e-1) -	9.536e+0 (2.47e-1) -	9.318e+0 (3.92e-1) -	8.304e+0 (6.85e-2) -	1.062e+1 (1.90e-1) -	5.814e+0 (5.17e-1)
WFG8	3	3.484e-1 (1.09e-2) -	2.850e-1 (1.70e-2) -	4.027e-1 (1.65e-2) -	3.429e-1 (2.42e-2) -	2.3762e-1 (5.96e-3) +	2.510e-1 (6.15e-3) +	2.913e-1 (7.74e-3)
	5	1.082e+0 (8.50e-3) =	1.091e+0 (2.83e-2) =	1.175e+0 (1.30e-2) =	1.154e+0 (2.36e-2) =	1.120e+0 (9.66e-1) -	1.622e+0 (6.49e-2) =	1.049e+0 (6.13e-3)
	8	3.264e+0 (9.49e-2) =	3.205e+0 (7.42e-2) =	3.642e+0 (3.95e-1) -	3.397e+0 (3.91e-2) =	3.205e+0 (1.65e-2) =	4.547e+0 (2.61e-1) =	3.181e+0 (1.91e-2)
	10	4.756e+0 (1.55e-1) =	4.611e+0 (9.60e-2) -	6.357e+0 (3.26e-1) -	4.581e+0 (3.99e-2) =	4.781e+0 (4.29e-2) =	6.335e+0 (1.22e-1) -	4.762e+0 (8.61e-2)
	15	9.213e+0 (6.79e-1) =	9.433e+0 (3.07e-1) -	9.600e+0 (1.93e-1) -	9.308e+0 (3.98e-1) =	9.047e+0 (2.10e-1) =	9.756e+0 (2.89e-1) -	9.367e+0 (2.90e-1)
WFG9	3	3.481e-1 (3.38e-2) -	2.307e-1 (2.88e-2) =	3.063e-1 (1.02e-2) -	2.651e-1 (1.86e-2) -	1.682e-1 (2.26e-2) -	2.338e-1 (4.28e-3) -	1.321e-1 (5.13e-3)
	5	1.073e+0 (2.43e-2) -	9.430e-1 (2.99e-1) =	1.022e+0 (1.74e-2) -	1.091e+0 (1.99e-2) -	9.323e-1 (8.43e-3) =	1.281e+0 (3.44e-2) =	9.316e-1 (5.66e-3)
	8	3.161e+0 (5.55e-2) =	2.331e+0 (1.27e+0) +	3.263e+0 (5.85e-2) =	3.362e+0 (5.21e-2) -	3.055e+0 (8.28e-2) =	3.680e+0 (1.81e-1) =	3.104e+0 (4.92e-2)
	10	4.365e+0 (1.40e-1) =	3.608e+0 (1.06e+0) +	4.821e+0 (3.50e-1) -	4.479e+0 (9.37e-2) =	4.413e+0 (6.04e-2) =	5.541e+0 (2.92e-2) =	4.231e+0 (7.16e-2)
	15	8.560e+0 (2.38e-1) =	1.111e+1 (2.84e+0) -	9.011e+0 (6.74e-2) -	8.953e+0 (2.49e-1) =	8.493e+0 (1.18e-1) -	9.714e+0 (2.07e-1) -	5.793e+0 (6.84e-1)
w/l/t		2/31/12	4/32/6	1/40/4	0/23/22	6/19/20	3/31/11	

As for IGD results on WFG test problems shown in Table 8, OBEA still perform well on most test instances. To be more specific, OBEA covers all the best values on WFG1 and WFG7 test problems, which shows the best performance of the proposed algorithms. As for the WFG2 test problem, which covers the convex, disconnected, and nonseparable

PF, the performance of OBEA is a little worse than the NSGA-III and Two_Arch2. However, the proposed OBEA still tends to outperform the other algorithms on parts of instances in WFG2. Although the best values on 3-objective, 8-objective, and 10-objective instances in WFG3 are obtained by MOEA/DVA and Two_Arche2, respectively,

TABLE 9: The statistical results (mean and standard deviation) of the HV values obtained by RVEA, NSGA-III, SPEAR, MOEA/DD, Two_Arch2, MOEA/DVA, and OBEA on DTLZ test suits. The best results are italicized.

Problem	M	RVEA	NSGA-III	SPEAR	MOEA/DD	Two_Arch2	MOEA/DVA	OBEA
DTLZ1	3	1.381e-1 (2.16e-1) -	3.376e-1 (3.13e-1) -	6.201e-1 (2.52e-1) =	1.410e-1 (2.11e-1) -	7.960e-1 (5.40e-2) -	2.046e-1 (2.74e-1) -	8.471e-1 (6.13e-3)
	5	8.322e-3 (2.82e-2) -	3.785e-4 (2.07e-3) -	7.918e-1 (2.04e-1) -	6.365e-3 (2.85e-2) -	9.706e-1 (3.70e-3) -	9.327e-1 (1.45e-2) -	9.763e-1 (4.15e-4)
	8	1.372e-1 (2.09e-1) -	7.559e-1 (3.35e-1) -	7.544e-1 (1.72e-1) -	1.647e-1 (2.57e-1) -	9.770e-1 (1.16e-2) =	9.847e-1 (3.07e-4) =	9.524e-1 (6.84e-3)
	10	2.852e-3 (7.36e-3) -	3.791e-1 (4.86e-1) -	2.673e-1 (2.73e-1) -	1.034e-2 (3.15e-2) -	9.548e-1 (4.52e-2) =	9.899e-1 (6.59e-4) =	9.576e-1 (4.18e-3)
	15	3.745e-1 (3.30e-1) -	2.847e-2 (8.42e-2) -	6.304e-1 (4.22e-2) -	4.646e-1 (3.17e-1) -	6.639e-1 (3.42e-1) =	9.161e-1 (5.54e-3) =	9.456e-1 (4.46e-6)
DTLZ2	3	5.531e-1 (1.01e-3) -	5.554e-1 (6.39e-4) -	5.349e-1 (4.08e-3) -	5.546e-1 (9.79e-4) =	5.778e-1 (1.97e-4) =	5.373e-1 (7.82e-4) =	5.563e-1 (5.70e-4)
	5	7.682e-1 (6.07e-3) -	7.455e-1 (8.96e-3) -	7.619e-1 (7.89e-3) -	7.693e-1 (5.35e-3) =	7.232e-1 (7.10e-3) -	7.629e-1 (6.20e-3) =	7.829e-1 (3.48e-3)
	8	8.938e-1 (6.36e-3) -	7.789e-1 (7.29e-2) -	8.545e-1 (3.38e-2) -	8.708e-1 (1.08e-2) =	7.844e-1 (1.28e-2) -	9.204e-1 (3.92e-3) =	8.981e-1 (4.96e-3)
	10	9.011e-1 (1.16e-2) =	6.419e-1 (8.57e-2) -	5.809e-1 (7.73e-2) -	8.598e-1 (2.14e-2) -	6.907e-1 (2.68e-2) -	8.197e-1 (1.62e-3) -	9.071e-1 (1.22e-2)
	15	9.487e-1 (1.24e-2) =	6.697e-1 (9.14e-2) -	4.951e-1 (1.43e-1) -	9.425e-1 (1.25e-2) =	5.704e-1 (1.81e-2) -	9.060e-1 (3.08e-3) =	9.459e-1 (7.02e-3)
DTLZ3	3	2.2193e-1 (2.34e-1) -	2.819e-1 (1.32e-1) -	1.466e-1 (4.62e-1) -	6.061e-1 (1.64e-2) +	2.475e-1 (2.63e-1) -	2.032e-1 (6.63e-1) -	5.212e-1 (4.53e-2)
	5	4.536e-1 (4.22e-1) -	5.753e-1 (2.35e-1) -	3.244e-1 (2.25e-1) -	5.042e-1 (1.30e-2) -	6.200e-2 (1.29e-1) -	4.499e-1 (2.67e-1) -	7.688e-1 (5.15e-2)
	8	3.255e-1 (6.33e-2) -	4.822e-1 (3.56e-1) =	4.631e-1 (3.12e-2) =	4.152e-1 (1.80e-2) -	3.344e-1 (3.55e-1) =	2.025e-1 (4.05e-1) -	5.475e-1 (3.15e-1)
	10	3.451e-1 (5.42e-1) -	3.202e-1 (3.44e-1) -	3.441e-1 (2.73e-1) -	4.489e-1 (1.46e-2) =	3.126e-1 (6.77e-1) -	2.644e-1 (4.07e-1) -	4.912e-1 (4.38e-1)
	15	1.974e-1 (3.95e-1) -	3.644e-1 (4.57e-1) -	4.174e-1 (3.61e-1) -	4.465e-1 (3.15e-2) -	3.988e-1 (5.66e-1) =	8.995e-2 (1.80e-1) =	9.879e-1 (6.17e-2)
DTLZ4	3	5.523e-1 (1.70e-3) +	5.047e-1 (9.30e-2) +	5.262e-1 (1.40e-2) +	5.545e-1 (1.16e-3) +	5.774e-1 (5.25e-4) +	4.774e-1 (2.97e-4) -	4.954e-1 (1.03e-1)
	5	7.701e-1 (8.65e-3) -	7.411e-1 (2.30e-2) -	6.074e-1 (3.15e-2) -	7.697e-1 (1.12e-2) -	7.582e-1 (7.73e-3) =	8.139e-1 (1.11e-3) =	7.814e-1 (8.95e-3)
	8	9.073e-1 (8.10e-3) -	7.530e-1 (1.22e-1) -	7.371e-1 (4.85e-2) -	8.949e-1 (1.04e-2) -	7.861e-1 (8.53e-3) -	9.352e-1 (2.18e-3) +	9.113e-1 (6.95e-3)
	10	9.419e-1 (5.46e-3) =	7.469e-1 (8.64e-2) -	6.235e-1 (8.62e-2) -	9.273e-1 (1.46e-2) =	8.010e-1 (9.28e-3) -	8.615e-1 (5.74e-3) -	9.396e-1 (6.00e-3)
	15	9.857e-1 (3.51e-3) =	5.269e-1 (2.15e-1) -	5.219e-1 (1.44e-1) -	9.827e-1 (3.77e-3) =	7.948e-1 (2.26e-2) -	9.090e-1 (1.72e-3) -	9.875e-1 (8.11e-4)
DTLZ7	3	2.000e-1 (1.86e-2) -	2.431e-1 (8.61e-3) -	2.454e-1 (1.07e-2) -	2.025e-1 (1.83e-2) -	2.696e-1 (2.61e-3) -	2.029e-1 (4.40e-4) =	2.526e-1 (6.40e-3)
	5	8.128e-2 (2.62e-2) -	6.448e-2 (1.72e-2) -	5.921e-2 (2.54e-2) -	1.381e-1 (1.73e-2) =	2.464e-1 (1.26e-2) +	2.024e-1 (7.68e-3) +	1.320e-1 (2.08e-2)
	8	1.300e-2 (1.32e-2) -	2.252e-3 (3.69e-3) -	4.428e-2 (3.69e-2) -	2.967e-2 (3.68e-2) -	1.320e-1 (6.49e-3) +	5.777e-2 (2.30e-2) -	9.463e-2 (1.67e-2)
	10	7.479e-4 (1.59e-3) -	9.219e-2 (9.50e-3) =	5.648e-2 (1.25e-2) -	6.538e-4 (3.97e-4) -	6.737e-2 (1.45e-2) -	7.533e-2 (7.97e-3) =	8.100e-2 (1.87e-3)
	15	1.881e-4 (4.00e-4) -	4.649e-2 (6.18e-3) =	5.322e-3 (4.12e-5) -	5.973e-7 (3.38e-7) -	7.323e-3 (1.10e-5) -	4.484e-2 (7.19e-5) -	5.400e-2 (3.06e-4)
w/l/t		1/20/4	1/21/3	1/22/2	2/15/8	3/15/7	2/12/11	

OBEA covers all the other instances on the WFG3 test problem. For the WFG4 test problem, OBEA and MOEA/DVA, together with Two_Arch2, obtain the best value. As for the WFG5 and WFG6 test problems, although RVEA covers the best value on 10-objective instance, OBEA still

tends to outperform most of the algorithms on the remaining instances. Although best values on some instances of WFG8 have been covered by MOEA/DD and Two_Arch2, OBEA still obtained the best performance on 5-objective and 8-objective test instances. As for the WFG9,

TABLE 10: The statistical results (mean and standard deviation) of the HV values obtained by RVEA, NSGA-III, SPEAR, MOEA/DD, Two_Arch2, MOEA/DVA, and OBFA on WFG test suits. The best results are italicized.

Problem	M	RVEA	NSGA-III	SPEAR	MOEA/DD	Two_Arch2	MOEA/DVA	OBFA
WFG1	3	3.633e-1 (1.84e-2) -	3.558e-1 (1.13e-2) -	9.425e-2 (1.90e-2) -	2.823e-1 (2.02e-2) -	4.214e-1 (1.71e-2) -	3.291e-1 (9.59e-3) -	7.613e-1 (3.46e-2)
	5	3.118e-1 (1.09e-2) -	3.074e-1 (3.49e-3) -	2.180e-1 (1.37e-2) -	2.866e-1 (5.40e-3) -	3.935e-1 (2.47e-2) -	4.054e-1 (1.28e-2) -	7.811e-1 (6.25e-1)
	8	2.606e-1 (1.58e-2) =	2.596e-1 (6.14e-3) -	2.390e-1 (5.72e-3) -	2.433e-1 (7.23e-3) -	2.833e-1 (1.30e-2) -	3.131e-1 (1.20e-2) -	4.913e-1 (1.18e-1)
	10	2.351e-1 (1.24e-2) -	2.335e-1 (5.63e-3) -	2.219e-1 (1.25e-3) -	2.206e-1 (1.59e-3) -	2.649e-1 (5.49e-3) -	2.597e-1 (1.16e-2) -	5.612e-1 (4.13e-2)
	15	2.594e-1 (4.63e-2) -	2.615e-1 (3.43e-2) -	2.970e-1 (1.67e-2) -	1.884e-1 (3.51e-3) -	2.198e-1 (9.17e-3) -	2.240e-1 (6.26e-3) -	9.913e-1 (3.14e-4)
WFG2	3	8.442e-1 (1.27e-2) -	8.774e-1 (1.01e-2) -	7.574e-1 (1.86e-2) -	8.912e-1 (1.43e-2) =	9.075e-1 (5.74e-3) =	8.170e-1 (4.90e-3) -	8.837e-1 (6.87e-3)
	5	8.536e-1 (1.88e-2) -	8.712e-1 (1.00e-2) -	7.385e-1 (1.59e-2) -	8.983e-1 (1.79e-2) =	9.584e-1 (5.75e-3) +	9.556e-1 (5.00e-3) +	8.870e-1 (1.17e-2)
	8	8.454e-1 (3.38e-2) -	8.839e-1 (1.82e-2) -	8.297e-1 (2.12e-2) -	8.911e-1 (2.59e-2) -	9.564e-1 (2.12e-3) =	9.501e-1 (1.86e-2) =	9.106e-1 (1.58e-2)
	10	8.357e-1 (2.69e-2) -	8.666e-1 (1.34e-2) -	7.782e-1 (2.05e-2) -	8.544e-1 (2.99e-2) -	9.034e-1 (7.76e-3) =	9.083e-1 (1.15e-2) =	9.086e-1 (1.56e-2)
	15	8.429e-1 (5.49e-2) -	9.237e-1 (2.74e-2) -	9.075e-1 (2.17e-2) -	8.739e-1 (3.97e-2) -	9.390e-1 (9.77e-3) -	9.503e-1 (4.19e-3) =	9.543e-1 (1.07e-2)
WFG3	3	2.882e-1 (1.28e-2) -	3.246e-1 (7.96e-3) -	2.426e-1 (2.19e-2) -	3.246e-1 (3.25e-3) =	3.690e-1 (5.93e-3) =	3.813e-1 (3.53e-3) =	3.475e-1 (8.44e-3)
	5	9.344e-3 (1.16e-2) -	1.625e-2 (1.19e-2) -	2.808e-3 (6.30e-3) -	4.313e-1 (1.52e-3) +	1.522e-1 (1.06e-2) -	3.488e-2 (9.22e-3) -	1.769e-1 (1.28e-3)
	8	1.301e-1 (3.61e-2) =	1.372e-1 (6.35e-2) =	1.126e-1 (2.45e-3) =	1.087e-1 (3.20e-2) =	1.124e-1 (5.62e-3) =	1.012e-1 (5.32e-3) =	1.131e-1 (2.43e-3)
	10	5.172e-1 (4.77e-2) -	5.768e-1 (5.32e-2) -	5.616e-1 (2.77e-2) =	5.090e-1 (2.43e-2) -	4.361e-1 (6.33e-2) -	5.513e-1 (3.35e-2) -	6.945e-1 (4.55e-2)
	15	2.757e-1 (3.12e-2) =	2.313e-1 (4.66e-2) =	3.331e-1 (4.66e-2) =	3.152e-1 (8.52e-2) =	3.033e-1 (2.69e-2) =	3.133e-1 (2.66e-2) =	3.455e-1 (4.69e-2)
WFG4	3	4.907e-1 (4.55e-3) -	5.035e-1 (3.55e-3) -	4.660e-1 (6.87e-3) -	5.015e-1 (3.38e-3) -	4.472e-1 (2.18e-3) -	5.523e-1 (1.71e-3) =	5.107e-1 (2.91e-3)
	5	6.544e-1 (8.62e-3) -	6.438e-1 (8.01e-3) -	6.362e-1 (8.51e-3) -	6.722e-1 (1.28e-2) -	7.025e-1 (5.47e-3) =	6.561e-1 (1.07e-2) -	6.785e-1 (5.15e-3)
	8	7.043e-1 (2.18e-2) -	7.360e-1 (1.47e-2) -	7.165e-1 (1.38e-2) -	6.954e-1 (2.47e-2) -	6.833e-1 (7.37e-3) -	6.933e-1 (1.57e-2) -	7.707e-1 (7.74e-3)
	10	7.294e-1 (2.64e-2) -	7.236e-1 (1.56e-2) -	7.459e-1 (1.43e-2) -	7.074e-1 (2.30e-2) =	6.634e-1 (1.50e-2) -	7.249e-1 (2.00e-2) =	7.690e-1 (1.19e-2)
	15	7.800e-1 (3.65e-2) -	6.974e-1 (3.34e-2) -	7.657e-1 (3.12e-2) -	7.673e-1 (2.42e-2) -	6.192e-1 (6.80e-3) -	7.177e-1 (1.13e-2) -	8.486e-1 (1.13e-2)
WFG5	3	4.720e-1 (5.52e-3) -	4.888e-1 (5.08e-3) -	4.598e-1 (7.03e-3) -	4.955e-1 (4.10e-3) =	5.191e-1 (2.57e-3) =	5.314e-1 (3.57e-3) =	4.939e-1 (4.50e-3)
	5	5.954e-1 (9.61e-3) -	5.787e-1 (9.35e-3) -	6.466e-1 (9.90e-3) +	6.395e-1 (7.09e-3) =	6.730e-1 (5.44e-3) =	5.462e-1 (1.73e-2) -	6.303e-1 (9.01e-3)
	8	6.320e-1 (2.19e-2) -	6.478e-1 (1.57e-2) -	6.621e-1 (1.52e-2) -	6.307e-1 (2.10e-2) -	6.317e-1 (9.55e-3) -	7.035e-1 (1.41e-2) =	7.098e-1 (1.44e-2)
	10	5.892e-1 (2.82e-2) -	6.020e-1 (1.85e-2) -	6.577e-1 (9.98e-3) +	5.839e-1 (2.39e-2) -	5.674e-1 (1.14e-2) -	5.944e-1 (6.85e-3) -	6.377e-1 (1.79e-2)
	15	6.375e-1 (4.07e-2) -	6.663e-1 (4.52e-2) -	6.915e-1 (1.34e-2) -	6.406e-1 (5.40e-2) -	4.687e-1 (2.33e-2) -	6.781e-1 (5.97e-3) -	7.498e-1 (2.03e-2)
WFG6	3	4.414e-1 (8.75e-3) -	4.665e-1 (1.07e-2) -	4.412e-1 (2.98e-2) -	4.390e-1 (3.29e-2) -	5.040e-1 (6.53e-3) =	5.012e-1 (5.62e-4) -	5.160e-1 (1.14e-2)
	5	5.611e-1 (1.53e-2) -	5.403e-1 (1.56e-2) -	5.370e-1 (5.69e-2) -	5.935e-1 (1.83e-2) -	6.488e-1 (8.33e-3) -	5.667e-1 (1.95e-2) -	6.927e-1 (1.50e-2)
	8	5.571e-1 (4.80e-2) -	6.108e-1 (2.64e-2) -	6.491e-1 (7.53e-2) =	6.072e-1 (2.98e-2) -	5.989e-1 (1.95e-2) -	5.970e-1 (2.33e-2) -	7.598e-1 (1.81e-2)
	10	5.560e-1 (2.31e-2) -	5.458e-1 (2.06e-2) -	5.821e-1 (2.81e-2) -	5.751e-1 (2.66e-2) -	5.315e-1 (1.12e-2) -	5.909e-1 (2.63e-2) -	7.557e-1 (2.25e-2)
	15	6.181e-1 (4.76e-2) -	6.155e-1 (5.22e-2) -	6.656e-1 (2.79e-2) -	6.888e-1 (2.87e-2) -	4.290e-1 (1.47e-2) -	5.285e-1 (5.15e-2) -	7.229e-1 (3.09e-2)

TABLE 10: Continued.

Problem	M	RVEA	NSGA-III	SPEAR	MOEA/DD	Two_Arch2	MOEA/DVA	OBEA
WFG7	3	4.701e-1 (1.52e-2) -	5.005e-1 (6.55e-3) -	4.734e-1 (9.21e-3) -	5.111e-1 (1.29e-2) -	5.601e-1 (1.03e-3) +	5.648e-1 (9.74e-4) +	5.148e-1 (2.33e-3)
	5	6.106e-1 (1.89e-2) -	5.821e-1 (1.50e-2) -	5.966e-1 (1.46e-2) -	6.506e-1 (1.39e-2) -	7.271e-1 (2.98e-3) =	6.129e-1 (2.46e-2) -	7.182e-1 (5.40e-3)
	8	6.679e-1 (3.37e-2) -	6.735e-1 (2.44e-2) -	6.340e-1 (1.67e-2) -	6.250e-1 (2.70e-2) -	6.752e-1 (1.15e-2) -	6.735e-1 (2.38e-3) -	8.113e-1 (6.13e-3)
	10	6.658e-1 (2.40e-2) -	6.636e-1 (2.32e-2) -	6.574e-1 (1.76e-2) -	6.087e-1 (2.39e-2) -	6.128e-1 (1.43e-2) -	6.799e-1 (4.66e-3) -	8.130e-1 (4.13e-2)
	15	6.960e-1 (5.44e-2) -	6.274e-1 (5.04e-2) -	7.397e-1 (2.14e-2) -	6.825e-1 (3.93e-2) -	5.063e-1 (2.81e-2) -	6.691e-1 (2.88e-2) -	8.167e-1 (4.01e-2)
WFG8	3	4.329e-1 (6.27e-3) -	4.533e-1 (5.50e-3) -	3.919e-1 (9.86e-3) -	4.260e-1 (1.83e-2) -	4.883e-1 (3.72e-3) -	4.968e-1 (3.89e-3) -	5.122e-1 (5.15e-2)
	5	5.465e-1 (9.83e-3) -	5.398e-1 (1.22e-2) -	5.025e-1 (1.11e-2) -	5.238e-1 (2.02e-2) -	6.189e-1 (5.45e-3) -	5.277e-1 (1.40e-2) -	6.915e-1 (3.63e-3)
	8	5.081e-1 (7.05e-2) -	6.039e-1 (3.24e-2) -	5.586e-1 (4.46e-2) -	5.236e-1 (3.00e-2) -	5.257e-1 (6.61e-3) -	4.231e-1 (4.73e-2) -	6.124e-1 (6.71e-2)
	10	5.321e-1 (6.01e-2) -	5.568e-1 (2.18e-2) -	6.051e-1 (1.51e-2) =	5.069e-1 (2.76e-2) -	4.585e-1 (2.08e-2) -	4.168e-1 (2.98e-2) -	7.184e-1 (5.17e-2)
	15	5.909e-1 (1.26e-1) =	4.682e-1 (8.94e-2) -	6.588e-1 (2.44e-2) -	6.781e-1 (6.17e-2) -	3.421e-1 (1.90e-2) -	3.675e-1 (9.47e-2) -	7.841e-1 (5.17e-2)
WFG9	3	4.234e-1 (1.68e-2) -	4.534e-1 (1.39e-2) -	4.474e-1 (6.05e-3) -	4.873e-1 (1.62e-2) -	5.234e-1 (2.08e-2) -	4.940e-1 (2.47e-3) -	5.611e-1 (6.12e-3)
	5	5.346e-1 (2.17e-2) -	5.383e-1 (2.10e-2) -	6.423e-1 (6.23e-3) =	5.869e-1 (2.65e-2) -	6.733e-1 (1.54e-2) =	6.324e-1 (1.38e-2) =	6.655e-1 (4.17e-2)
	8	5.663e-1 (4.39e-2) -	5.807e-1 (4.29e-2) -	6.484e-1 (1.12e-2) =	5.386e-1 (3.35e-2) -	6.204e-1 (2.10e-2) -	6.512e-1 (1.32e-2) =	7.012e-1 (6.32e-2)
	10	5.373e-1 (3.41e-2) -	5.656e-1 (3.86e-2) -	6.499e-1 (1.96e-2) =	5.248e-1 (4.63e-2) -	5.793e-1 (3.06e-2) -	6.617e-1 (1.86e-2) =	6.791e-1 (4.17e-2)
	15	5.387e-1 (7.57e-2) -	6.343e-1 (5.56e-2) -	6.232e-1 (3.56e-2) -	5.315e-1 (6.46e-2) -	4.703e-1 (1.29e-2) -	5.670e-1 (5.39e-3) -	7.824e-1 (3.42e-2)
w/l/t		0/41/4	0/43/2	2/35/8	1/36/8	2/31/12	2/30/13	

NSGA-III shows better performance than that on the DTLZ test problem. However, OBEA still outperforms most of the algorithms, especially on 3-objective, 5-objective, and 15-objective test instances of WFG9.

The statistical results of the HV values obtained by the six algorithms are summarized in Tables 9 and 10. The values of HV presented in tables are all normalized to [0,1]. Clearly, a higher value of HV is recommended. OBEA obtains the majority of the best median values over the instances on DTLZ, which indicates the obvious advantages of the proposed algorithm. For the DTLZ1 test problem, although MOEA/DVA obtains the best values on 8-objective and 10-objective instances, OBEA shows good performance on 3-objective, 8-objective, and 15-objective instances. Since DTLZ2 test problems involve concave PFs, Two_arch2 and MOEA/DVA, together with RVEA, obtain the best performance on 3-objective, 8-objective, and 15-objective instances, respectively, while OBEA covers the best values on 5-objective as well as 10-objective instance and tends to be far more superior to the other algorithms. Furthermore, although DTLZ3 is a concave and multimodal test problem, OBEA obtains most of the best HV value on this test problem. As for the DTLZ4, the performance of OBEA is little worse than Two_Arch2 and MOEA/DVA. However, OBEA still outperform most of the algorithms, especially on 15-objective DTLZ4. In addition, due to the fact that disconnected and multimodal PFs have been employed in

the DTLZ7 test problem, the performance of OBEA is not that good on 5-objective and 8-objective instances; however, OBEA still outperforms other algorithms, especially on the other instances.

Statistical results of the HV values on WFG test sets are shown in Table 10. OBEA has shown the most competitive performance on WFG1, WFG6, and WFG8. By contrast, performance of RVEA on the WFG test functions is not as good as that on the DTLZ test functions. However, SPEAR shows good performance on WFG test problems, especially on the WFG5 test problem. Moreover, although the performance of OBEA is little worse than Two_Arch2 on some instances in WFG2, it still tends to be superior on most test instances like the 15-objective WFG5 instance, which is shown in Figure 4. As for the WFG2 test problem, the performance of NSGA-III is much better than that on other test instances. Finally, for the WFG7 and WFG9 test problems, although Two_Arch2 and MOEA/DVA cover some of the best values, OBEA still shows the best performance among most algorithms on these test instances.

In a word, OBEA can obtain the best performance on these test problems from multiobjective to many-objective. This may be due to the fact that the exploration ability and convergence of OBEA are emphasized due to the employed OBL and adaptive clustering mechanisms, which lead the

FIGURE 4: Parallel coordinates of the nondominated front obtained by each algorithm on 15-objective WG5 in the run associated with the median HV value. (a) OBEA on WFG5. (b) RVEA on WFG5. (c) NSGA-III on WFG5. (d) MOEA/DD on WFG5. (e) SPEAR on WFG5. (f) MOEA/DVA on WFG5. (g) Two Arch2 on WFG5.

results revealing high efficiency. Note that the performance of OBEA may become little worse while handling some test problems with irregular PF. This is because OBEA uses a set of uniformly distributed reference vectors to handle the problem. While using the uniformly distributed reference vectors is based on an assumption that the PF has a regular geometrical structure, this may have the influence on the performance of OBEA on the test problem with irregular PF. To sum up, the proposed OBEA can obtain the best performance on most test problems with the PF which has regular geometrical structure. As for the problem with quite irregular PF, the performance of OBEA is not as well as on the previous problem. However, the proposed OBEA still tends to provide the competitive results.

As to quantify how well each algorithm performs with the HV overall, the performance score [37] is introduced to

rank all the algorithms. Given a specific problem instance, suppose there are k algorithms $\text{Alg}_1, \text{Alg}_2, \dots, \text{Alg}_k$ involved in the comparison. $\delta_{i,j}$ is set to 1 if Alg_j is significantly better than Alg_i in terms of HV, and 0 otherwise. Thereafter, for each algorithm Alg_i , the performance score $P(\text{Alg}_i)$ is determined as follows:

$$P(\text{Alg}_i) = \sum_{j=1, j \neq i}^k \delta_{i,j}. \quad (17)$$

This value represents the number of algorithms which have significantly better performance than the corresponding algorithm on the entire tested instances, and zero means that no other algorithm tends to be significantly better in terms of the HV indicator. Clearly, the smaller the index, the better the algorithm; Figure 5

FIGURE 5: Ranking and score of average performance obtained by each compared algorithms in terms of HV.

shows the average performance score over all 70 test instances among the ten algorithms in terms of HV, and the rank of each algorithm according to the score is also given.

As the figure shows, OBEA is in the first place among the eleven algorithms, which indicates the better performance of the proposed algorithm on these test problems. Note that although MOEA/D has been ranked in the last place, it still shows good performance on some test problems. Moreover, in order to better visualize the performance, the average performance score summarized for different numbers of objectives and different test problems in terms of the HV are presented in Figure 6, respectively.

Figure 6(a) shows the average performance over all test problems for different number of objectives. The proposed OBEA works well on nearly all the considered problems of diverse objectives except on 3-objective problems. However, OBEA still outperforms most of the algorithms on 3-objective problems. Furthermore, the performance score for the individual test problems is shown in Figure 6(b), which covers all number of objectives. For DTLZ1-DTLZ3, OBEA shows good performance and nearly outperforms all other algorithms. As for DTLZ4 and DTLZ7, the performance of OBEA is not as good as that on DTLZ1-3 but still is generally superior among the corresponding algorithms. At last, for the WFG test problem, OBEA nearly covers the best values. In a word, OBEA is under a good performance among the ten algorithms in terms of the HV.

4.4.3. Analysis of Adaptive Clustering Mechanism. In the proposed OBEA, the adaptive clustering is adopted to assist the algorithm for selecting the individual from each sub-population. Nowadays, different approaches have been proposed to cluster the individuals. For example, RVEA and NSGA-III use the angle to partition the population and θ -DEA uses the perpendicular distance to cluster the individuals. Although most of these approaches simply focus on the acute angle or distance alone to cluster the individuals and may have superior performance in some situations. However, simplex use of the acute angle or distance may lead the crowding of individuals in cluster increase and lose some better situations and thus affects the performance of the algorithm in balancing diversity and convergence.

In the view of this, in our proposed OBEA, the acute angle and the distance are considered comprehensively with an adaptive mechanism to assist the algorithm to assign individuals efficiently. For example, in Figure 7(a), giving the individual x and its normalization $\tilde{f}(x)$, λ_1 and λ_2 are two different reference vectors. θ_1 and θ_2 indicate the acute angles between $\tilde{f}(x)$ and λ_1 and λ_2 , respectively. Moreover, d_{p1} and d_{p2} indicate the distances between the origin and the projection of $\tilde{f}(x)$ on λ_1 and λ_2 , respectively. Considering the distance-based approach to assign the individual x , the reference vector θ_2 will be the preferred one as its distance d_{p2} is smaller than the distance d_{p1} . However, to focus on the acute angle-based clustering, the individual x will be associated with λ_1 according to the fact that the acute angle of θ_1 is smaller than the angle of θ_2 , obviously. In our adaptive clustering strategy, the individuals will be associated with the reference vector according to the distance between the origin and the projection point at first to increase the diversity of the individuals in the cluster. Thereafter, with increase in the number of iterations, the individuals will be assigned according to the angle through our adaptive function.

As the function s is the key point in the adaptive scheme, here we consider the following three different functions to do the further analysis:

$$\text{linear: } s = \left\lceil \frac{t}{t_{\max}} \right\rceil,$$

$$\text{sigmoid: } s = \left\lceil \frac{1}{1 + \exp((\mu - t/t_{\max}) \times 15)} \right\rceil, \quad (18)$$

$$\text{exponential: } s = \left\lceil \frac{\exp(5t/t_{\max}) - 1}{\exp(5) - 1} \right\rceil.$$

Here, Figure 7(b) shows the values of S using three different functions. t is the current generation number, $t_{\max}$ is the maximal generation number, and $\lceil \cdot \rceil$ is the ceiling function. Moreover, μ is the phase control parameter for the sigmoid function. As Figure 8 shows, the IGD results on different test problems when using various μ values have indicated that 0.3, 0.4 is the best setting for the algorithm. Due to the fact that the function can obtain the standard phase in $[0, 1]$ with this value of μ equal to 0.3 or 0.4 as

FIGURE 6: (a) Average performance score obtained by eleven algorithms over all test problems of different numbers of objectives in terms of the HV and (b) average performance score obtained by ten algorithms on dimensions for different test problems in terms of the HV, Dx for DTLZ, and Wx for WFG. The values of the proposed OBEA are connected by a solid red line.

FIGURE 7: (a) An illustration of the special situation when solely using the acute angle or perpendicular distance to select individual. (b) An illustration of the three functions versus the iterative generations.

shown in Figure 8, here $\mu = 0.35$ is adopted in the following experiments. In order to validate the effectiveness of the proposed schemes, we compare OBEA (here we named our algorithm OBEA-sig) with the following two variants.

- (1) Variant I: in this variant, the adaptive scheme in OBEA is implemented based on adaptive clustering with the linear function (OBEA-lin).

- (2) Variant II: the difference of this variant and OBEA is that the adaptive scheme is constructed through the exponential function (OBEA-exp).

In this study, OBEA with two variants are tested on the DTLZ test suite and the WFG test suit with three to fifteen objectives for 30 runs and applied HV to quantify the performance. The parameter settings of the two variants

FIGURE 8: (a) An illustration of the IGD results on 15-objective DTLZ2 when using different μ . (b) An illustration of the IGD results on 10-objective WFG7 when using different μ . (c) An illustration of the IGD results on 8-objective DTLZ4 when using different μ . (d) An illustration of the IGD results on 5-objective WFG4 when using different μ .

TABLE 11: Performance comparison of OBEA and its two variants on DTLZ and WFG test suites in terms of HV.

Problem	M	OBEA-lin	OBEA-exp	OBEA-sig
DTLZ1	3	$7.2247e-1$ ($1.39e-1$) –	$6.2997e-1$ ($2.95e-1$) –	$8.471e-1$ ($6.13e-3$)
	5	$6.7003e-1$ ($3.78e-1$) –	$5.7003e-1$ ($3.78e-1$) –	$9.763e-1$ ($4.15e-4$)
	8	$2.7315e-1$ ($5.46e-2$) –	$2.1610e-1$ ($3.96e-1$) –	$9.524e-1$ ($6.84e-3$)
	10	$7.5060e-1$ ($1.06e-1$) –	$6.5552e-1$ ($3.17e-1$) –	$9.576e-1$ ($4.18e-3$)
	15	$2.3922e-1$ ($2.08e-1$) –	$1.6218e-1$ ($2.24e-1$) –	$9.456e-1$ ($4.46e-6$)
DTLZ2	3	$1.1140e-1$ ($2.84e-2$) –	$4.7777e-1$ ($3.06e-2$) –	$5.563e-1$ ($5.70e-4$)
	5	$1.5081e-1$ ($3.88e-2$) –	$6.1838e-1$ ($6.07e-2$) –	$7.829e-1$ ($3.48e-3$)
	8	$3.2693e-1$ ($1.91e-2$) –	$8.0224e-1$ ($2.08e-2$) =	$8.981e-1$ ($4.96e-3$)
	10	$1.9501e-1$ ($4.99e-2$) –	$5.0579e-1$ ($8.63e-2$) –	$9.071e-1$ ($1.22e-2$)
	15	$1.7267e-1$ ($4.04e-2$) –	$1.1953e-1$ ($3.14e-2$) –	$9.459e-1$ ($7.02e-3$)
DTLZ3	3	$1.7451e-3$ ($9.56e-3$) –	$1.5717e-1$ ($3.36e-4$) –	$5.212e-1$ ($4.53e-2$)
	5	$4.5189e-1$ ($1.68e-1$) –	$3.6816e-1$ ($1.38e-3$) –	$7.688e-1$ ($5.15e-2$)
	8	$7.0697e-1$ ($6.81e-2$) +	$4.4701e-1$ ($2.07e-2$) –	$5.475e-1$ ($3.15e-1$)
	10	$6.5542e-1$ ($5.27e-2$) +	$7.3079e-1$ ($8.45e-3$) +	$4.912e-1$ ($4.38e-1$)
	15	$2.5727e-1$ ($1.01e-2$) –	$5.5234e-1$ ($5.97e-2$) –	$9.879e-1$ ($6.17e-2$)

TABLE 11: Continued.

Problem	M	OBEA-lin	OBEA-exp	OBEA-sig
DTLZ4	3	1.6104e-1 (3.76e-2) -	4.6278e-1 (1.06e-1) -	4.954e-1 (1.03e-1)
	5	2.0649e-1 (4.92e-2) -	6.4816e-1 (5.07e-2) -	7.814e-1 (8.95e-3)
	8	3.4719e-1 (6.19e-3) -	8.2327e-1 (4.48e-2) =	9.113e-1 (6.95e-3)
	10	1.9669e-1 (6.25e-2) -	6.7233e-1 (6.52e-2) -	9.396e-1 (6.00e-3)
	15	1.7155e-1 (4.37e-2) -	1.1721e-1 (9.91e-2) -	9.875e-1 (8.11e-4)
DTLZ7	3	2.5727e-1 (1.01e-2) =	1.8185e-1 (3.85e-2) -	2.526e-1 (6.40e-3)
	5	1.8367e-1 (4.58e-2) +	1.1242e-1 (5.40e-2) -	1.320e-1 (2.08e-2)
	8	1.2285e-2 (2.86e-2) -	1.5925e-2 (1.36e-2) -	9.463e-2 (1.67e-2)
	10	1.2317e-2 (2.21e-4) -	3.9912e-2 (8.81e-3) -	8.100e-2 (1.87e-3)
	15	1.0056e-7 (1.82e-7) -	1.3300e-1 (4.90e-2) -	5.400e-2 (3.06e-4)
WFG1	3	5.0301e-1 (4.50e-2) -	2.4845e-1 (3.59e-2) -	7.613e-1 (3.46e-2)
	5	3.9091e-1 (4.40e-2) -	2.8672e-1 (3.36e-2) -	7.811e-1 (6.25e-1)
	8	8.5770e-1 (6.00e-3) +	2.4956e-1 (1.26e-3) -	4.913e-1 (1.18e-1)
	10	3.0816e-1 (3.93e-2) -	2.3200e-1 (2.02e-2) -	5.612e-1 (4.13e-2)
	15	9.0611e-1 (2.30e-4) =	8.6141e-2 (5.89e-2) -	9.913e-1 (3.14e-4)
WFG2	3	9.0441e-1 (9.92e-3) =	8.3414e-1 (5.40e-2) -	8.837e-1 (6.87e-3)
	5	9.5718e-1 (2.01e-2) =	8.9478e-1 (3.42e-2) =	8.870e-1 (1.17e-2)
	8	8.9191e-1 (2.13e-2) =	8.9445e-1 (3.75e-2) =	9.106e-1 (1.58e-2)
	10	9.6176e-1 (6.39e-2) =	8.9860e-1 (8.22e-2) =	9.086e-1 (1.56e-2)
	15	9.2253e-1 (6.05e-2) =	4.3771e-1 (1.93e-1) -	9.543e-1 (1.07e-2)
WFG3	3	3.7932e-1 (9.83e-3) =	3.4173e-1 (2.57e-2) =	3.475e-1 (8.44e-3)
	5	5.2190e-1 (3.31e-2) +	3.4134e-2 (3.78e-2) -	1.769e-1 (1.28e-3)
	8	1.0221e-1 (3.56e-3) =	1.4421e-1 (3.31e-3) =	1.131e-1 (2.43e-3)
	10	5.3674e-1 (5.52e-2) =	5.3927e-1 (3.50e-3) =	6.945e-1 (4.55e-2)
	15	3.2132e-1 (1.43e-1) =	3.2479e-1 (2.74e-2) =	3.455e-1 (4.69e-2)
<i>w/l/t</i>		5/29/16	1/30/9	

are the same as in OBEA. The mean and standard deviation of HV values of the two variants as well as OBEA are listed in Table 11, where the best mean value for each test instance is italicized. Also, the Wilcoxon rank sum test is performed to investigate whether there is a significant performance difference between OBEA and its two variants. As Table 11 shows, OBEA-sig obtains most of the best mean values among the DTLZ and WFG test instances. Although OBEA-lin and OBEA-exp have obtained parts of the best values, OBEA-sig still performs well on most test instances according to the Wilcoxon rank sum test. In a word, OBEA-sig is more stable and efficient than others. Therefore, the sigmoid function of adaptive strategy is adopted in this paper.

4.4.4. Analysis of Opposition-Based Selection. The key point in our opposition-based selection part is the proposed hybrid distance (HD). Here, we use the contrast experiment to further explore the effectiveness of the strategy. Note that the formulation of HD shares some similarity to angle-penalized distance (APD) [53] and penalty-based boundary intersection (PBI) [43], which are adopted in the RVEA and decomposition-based MOEAs, respectively. In this section, the comparative studies on the original OBEA, our proposed algorithm with the APD approach, and the proposed algorithm with the PBI approach are carried out. In particular, we replace the HD with APD or PBI. Thereafter, the opposition point is calculated according to the Section 3.4. Moreover, the experiment is

also conducted with the Wilcoxon rank sum test. For simplicity, the proposed algorithm with the APD and PBI approaches are denoted as OBEA-APD and OBEA-PBI, respectively.

The performance of OBEA, OBEA-APD, and OBEA-PBI has been verified on four benchmark test problems selected from different test suites named DTLZ1, DTLZ2, WFG2, and WFG4. As shown by the statistical results summarized in Table 12, OBEA shows the best performance on most problems, especially for the DTLZ2, on which the best values are all obtained by the proposed algorithm with HD. As for OBEA-APD and OBEA-PBI, these two algorithms obtain the best values on some instances. However, performance of OBEA on these instances is equal to these two algorithms according to the Wilcoxon rank sum test. To sum up, the results indicate that the proposed HD has better capability and effectiveness than APD and PBI, which further verify the effect of the proposed strategy on handling the many-objective problems.

4.4.5. Effectiveness Analysis of OBL and Adaptive Clustering in OBEA. As two main strategies have been designed in our OBEA, here three variational algorithms are generated to investigate the effectiveness of each strategy. In particular, they are OBEA1 (OBEA without OBL strategy), OBEA2 (OBEA without adaptive clustering strategy), and OBEA3 (OBEA without OBL strategy and adaptive clustering strategy). Moreover, the performance of these

TABLE 12: The statistical results of the HV obtained by OBEA, OBEA-APD, and OBEA-PBI. The best results are italicized.

Problem	M	DTLZ1	DTLZ2	WFG2	WFG4
OBEA	3	<i>8.471e-1 (6.13e-3)</i>	<i>5.563e-1 (5.70e-4)</i>	<i>8.837e-1 (6.87e-3)</i>	<i>5.107e-1 (2.91e-3)</i>
	5	<i>9.761e-1 (4.15e-4)</i>	<i>7.829e-1 (3.48e-3)</i>	<i>8.870e-1 (1.17e-2)</i>	<i>6.785e-1 (5.15e-3)</i>
	8	<i>9.524e-1 (6.84e-3)</i>	<i>8.981e-1 (4.96e-3)</i>	<i>9.106e-1 (1.58e-2)</i>	<i>7.707e-1 (7.74e-3)</i>
	10	<i>9.571e-1 (4.18e-3)</i>	<i>9.071e-1 (1.22e-2)</i>	<i>9.086e-1 (1.56e-2)</i>	<i>7.690e-1 (1.19e-2)</i>
	15	<i>9.456e-1 (4.46e-6)</i>	<i>9.459e-1 (7.02e-3)</i>	<i>9.543e-1 (1.07e-2)</i>	<i>8.486e-1 (1.13e-2)</i>
OBEA-APD	3	<i>6.299e-1 (2.95e-1) -</i>	<i>4.777e-1 (3.06e-2) -</i>	<i>8.341e-1 (5.40e-2) -</i>	<i>5.353e-1 (2.85e-3) -</i>
	5	<i>5.700e-1 (3.78e-1) -</i>	<i>6.183e-1 (6.07e-2) -</i>	<i>9.044e-1 (9.92e-3) =</i>	<i>7.116e-1 (7.46e-3) =</i>
	8	<i>9.161e-1 (3.96e-1) =</i>	<i>8.022e-1 (2.08e-2) -</i>	<i>8.571e-1 (2.01e-2) -</i>	<i>7.021e-1 (2.74e-2) =</i>
	10	<i>6.555e-1 (3.17e-1) -</i>	<i>9.057e-1 (8.63e-2) =</i>	<i>8.986e-1 (8.22e-2) =</i>	<i>6.675e-1 (8.16e-2) =</i>
	15	<i>9.521e-1 (2.24e-1) =</i>	<i>9.195e-1 (3.14e-2) -</i>	<i>4.377e-1 (1.93e-1) -</i>	<i>5.933e-1 (1.27e-2) -</i>
OBEA-PBI	3	<i>7.224e-1 (1.39e-1) -</i>	<i>4.114e-1 (2.84e-2) -</i>	<i>8.047e-1 (3.42e-2) -</i>	<i>4.479e-1 (2.57e-2) -</i>
	5	<i>6.700e-1 (3.78e-1) -</i>	<i>7.508e-1 (3.88e-2) -</i>	<i>8.014e-1 (3.75e-2) -</i>	<i>5.121e-1 (4.26e-2) -</i>
	8	<i>9.731e-1 (5.46e-2) =</i>	<i>7.269e-1 (1.91e-2) -</i>	<i>8.919e-1 (2.13e-2) =</i>	<i>6.299e-1 (4.47e-2) -</i>
	10	<i>7.506e-1 (1.06e-1) -</i>	<i>9.050e-1 (4.99e-2) =</i>	<i>9.176e-1 (6.39e-2) =</i>	<i>4.798e-1 (5.73e-2) -</i>
	15	<i>9.392e-1 (2.08e-1) =</i>	<i>9.026e-1 (4.04e-2) -</i>	<i>9.225e-1 (6.05e-2) =</i>	<i>6.059e-1 (1.90e-2) -</i>

TABLE 13: The statistical results of the HV obtained by various OBEAs. The best results are italicized.

Problem	M	OBEA1	OBEA2	OBEA3	OBEA
DTLZ1	3	<i>6.1214e-1 (3.24e-2) -</i>	<i>1.0640e-1 (5.12e-2) -</i>	<i>6.0619e-1 (4.74e-2) -</i>	<i>8.471e-1 (6.13e-3)</i>
	5	<i>5.9038e-1 (2.21e-2) -</i>	<i>6.8560e-1 (7.88e-3) -</i>	<i>7.4080e-1 (5.74e-4) -</i>	<i>9.763e-1 (4.15e-4)</i>
	8	<i>5.9182e-1 (4.12e-2) -</i>	<i>5.2621e-1 (4.82e-2) -</i>	<i>4.7633e-1 (6.07e-4) -</i>	<i>9.524e-1 (6.84e-3)</i>
	10	<i>8.9323e-1 (2.66e-2) -</i>	<i>6.9613e-1 (3.94e-1) -</i>	<i>7.8192e-1 (6.59e-4) -</i>	<i>9.576e-1 (4.18e-3)</i>
	15	<i>7.8893e-1 (5.45e-3) -</i>	<i>8.5401e-1 (6.84e-1) =</i>	<i>6.8614e-1 (5.54e-3) -</i>	<i>9.456e-1 (4.46e-6)</i>
DTLZ2	3	<i>5.7673e-1 (2.42e-4) =</i>	<i>5.6556e-1 (9.98e-4) =</i>	<i>5.1138e-1 (7.82e-4) -</i>	<i>5.563e-1 (5.70e-4)</i>
	5	<i>7.7404e-1 (3.05e-4) -</i>	<i>6.3986e-1 (4.48e-3) -</i>	<i>7.6299e-1 (6.20e-3) -</i>	<i>7.829e-1 (3.48e-3)</i>
	8	<i>8.8491e-1 (6.94e-3) =</i>	<i>7.1621e-1 (4.14e-2) -</i>	<i>6.2045e-1 (3.92e-3) -</i>	<i>8.981e-1 (4.96e-3)</i>
	10	<i>9.3184e-1 (3.75e-3) =</i>	<i>8.7840e-1 (7.51e-2) -</i>	<i>7.1978e-1 (1.62e-3) -</i>	<i>9.071e-1 (1.22e-2)</i>
	15	<i>8.2096e-1 (2.89e-2) -</i>	<i>8.2838e-1 (7.58e-2) -</i>	<i>6.7601e-1 (3.08e-3) -</i>	<i>9.459e-1 (7.02e-3)</i>
DTLZ3	3	<i>5.4667e-1 (4.62e-1) =</i>	<i>5.0214e-1 (6.47e-1) =</i>	<i>2.0327e-1 (6.63e-1) -</i>	<i>5.212e-1 (4.53e-2)</i>
	5	<i>6.2447e-1 (2.25e-1) -</i>	<i>6.7172e-1 (4.55e-1) -</i>	<i>4.4995e-1 (2.67e-1) -</i>	<i>7.688e-1 (5.15e-2)</i>
	8	<i>4.6312e-1 (3.12e-2) =</i>	<i>5.7638e-1 (3.66e-2) =</i>	<i>2.0254e-1 (4.05e-1) -</i>	<i>5.475e-1 (3.15e-1)</i>
	10	<i>3.4412e-1 (2.73e-1) -</i>	<i>4.5211e-1 (6.42e-1) -</i>	<i>2.6443e-1 (4.07e-1) -</i>	<i>4.912e-1 (4.38e-1)</i>
	15	<i>8.1743e-1 (3.61e-1) =</i>	<i>6.7614e-1 (5.42e-1) -</i>	<i>6.9950e-2 (1.80e-1) -</i>	<i>9.879e-1 (6.17e-2)</i>
WFG1	3	<i>6.1470e-1 (6.81e-2) -</i>	<i>5.0581e-1 (4.05e-2) -</i>	<i>3.6909e-1 (5.93e-3) -</i>	<i>7.613e-1 (3.46e-2)</i>
	5	<i>5.0201e-1 (2.28e-2) -</i>	<i>4.3295e-1 (2.37e-2) -</i>	<i>1.5220e-1 (1.06e-2) -</i>	<i>7.811e-1 (6.25e-1)</i>
	8	<i>4.1424e-1 (2.95e-2) =</i>	<i>2.8761e-1 (1.70e-2) -</i>	<i>1.1243e-1 (5.62e-3) -</i>	<i>4.913e-1 (1.18e-1)</i>
	10	<i>4.6201e-1 (2.57e-2) =</i>	<i>2.7123e-1 (5.51e-2) -</i>	<i>4.3613e-1 (6.33e-2) =</i>	<i>5.612e-1 (4.13e-2)</i>
	15	<i>8.2429e-1 (2.51e-2) -</i>	<i>8.0893e-1 (2.18e-2) -</i>	<i>6.0337e-1 (2.69e-2) -</i>	<i>9.913e-1 (3.14e-4)</i>
WFG2	3	<i>8.0484e-1 (3.64e-3) -</i>	<i>8.0997e-1 (2.71e-3) =</i>	<i>5.4720e-1 (2.18e-3) -</i>	<i>8.837e-1 (6.87e-3)</i>
	5	<i>9.0383e-1 (3.83e-3) =</i>	<i>8.3424e-1 (9.68e-3) =</i>	<i>7.0257e-1 (5.47e-3) -</i>	<i>8.870e-1 (1.17e-2)</i>
	8	<i>8.3108e-1 (6.92e-3) =</i>	<i>9.1790e-1 (1.42e-2) =</i>	<i>6.8330e-1 (7.37e-3) -</i>	<i>9.106e-1 (1.58e-2)</i>
	10	<i>8.3799e-1 (6.61e-3) =</i>	<i>8.9274e-1 (1.80e-2) -</i>	<i>6.6341e-1 (1.50e-2) -</i>	<i>9.086e-1 (1.56e-2)</i>
	15	<i>9.2971e-1 (1.54e-2) =</i>	<i>8.5648e-1 (4.82e-2) =</i>	<i>6.1924e-1 (6.80e-3) -</i>	<i>9.543e-1 (1.07e-2)</i>
<i>w/l/t</i>		0/13/12	0/17/8	0/24/1	

algorithms is shown in Table 13 and the better values are italicized.

The results show that the two strategies both have ability to improve the performance of the algorithm. In detail, although OBEA1 and OBEA2 both show the better capability of searching for the best function values for most cases, the performance of OBEA2 is slightly worse than OBEA1 according to the Wilcoxon rank sum test, which indicates that adaptive strategy has further influence on assisting algorithm than the OBL strategy. This can be attributed to the best structure of the adaptive clustering mechanism and

the effectiveness of the reference line. Moreover, although the performance of OBEA3 is little worse than the others, the values of the HV of this variation still tend to be good, which indicates the best potential of nondominated sorting in OBEA. To sum up, due to good performance of these strategies, our proposed OBEA gains best potential in solving different kinds of problems.

4.4.6. *Investigation of the Evolutionary Behavior.* As to observe the evolutionary behaviors of the algorithms, further

FIGURE 9: Trajectory of the mean IGD value on ten algorithms with fifteen objectives. (a) DTLZ1. (b) DTLZ4. (c) DTLZ7. (d) WFG4. (e) WFG7. (f) WFG9.

studies with these ten algorithms have been carried out to exhibit the evolutionary trajectories. Moreover, Figure 9 plots the performance trajectories of IGD versus the number of function evaluations for the ten algorithms on the 15-objective DTLZ1, DTLZ4, DTLZ7, WFG4, WFG7, and WFG9 test instances. Depending on the figures, OBEA shows an obvious advantage over most of its competitors. Although the performance of OBEA is little worse on DTLZ4, the figure clearly shows that the proposed OBEA still tends to be superior to the other algorithms on these two instances. By simultaneously considering the IGD, HV, and evolutionary behaviors, we can conclude that the overall performance of OBEA is better than or at least equal to the nine competitors.

5. Conclusion

Due to the loss of selection pressure and the ineffective design of balancing the diversity and convergence, general

MOEAs always encounter challenges in solving MaOPs. Although diversity of algorithms, especially the reference-based algorithms, have been designed to address this issue, most algorithms that belong to these categories simply use the angle or distance solely to measure the quality of the population members with the reference set, which may lose some good solutions due to their simplex selection mechanism. Therefore, we have presented a new many-objective evolutionary algorithm, called OBEA. Two strategies called OBL and adaptive clustering have been adopted in environmental selection with the purpose of improving the performance of the algorithm in balancing the convergence and the diversity.

To establish the strong competitiveness, we have employed extensive experimental comparison of OBEA with ten algorithms. A number of well-known benchmark problems such as DTLZ test suits and WFG test suits are chosen to explore the abilities of the algorithms. The

statistical results reveal that the proposed OBEA performs well on almost all the problem instances, and it obviously outperforms most state-of-the-art many-objective optimizers. However, the result also indicates that none of the algorithms is capable of preceding any of the other algorithms on all the instances which reflect the importance of the choice of algorithms carefully when solving the MaOPs.

In the future, we will extend OBEA to solve constrained many-objective problems by incorporating constraint handling techniques in order to further verify its effectiveness.

Data Availability

All data included in this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant 61873240.

References

- [1] K. Deb, K. Sindhya, and J. Hakanen, "Multi-objective optimization," in *Decision Sciences: Theory and Practice*, pp. 145–184, CRC Press, Boca Raton, FL, USA, 2016.
- [2] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*, MIT Press, Cambridge, MA, USA, 1992.
- [3] H.-G. Beyer and H.-P. Schwefel, "Evolution strategies—a comprehensive introduction," *Natural computing*, vol. 1, no. 1, pp. 3–52, 2002.
- [4] M. Dorigo and M. Birattari, "Ant colony optimization," in *Encyclopedia of Machine Learning*, pp. 36–39, Springer, Berlin, Germany, 2011.
- [5] K. Deb, "Multi-objective optimization," in *Search Methodologies*, pp. 403–449, Springer, Berlin, Germany, 2014.
- [6] C. A. C. Coello, G. B. Lamont, D. A. Van Veldhuizen et al., *Evolutionary Algorithms for Solving Multi-Objective Problems*, Springer, Vol. 5, Berlin, Germany, 2007.
- [7] E. Zitzler, M. Laumanns, and L. Thiele, "Spea2: improving the strength pareto evolutionary algorithm," *TIK-Report*, vol. 103, 2001.
- [8] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [9] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga, "Handling multiple objectives with particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 256–279, 2004.
- [10] W. K. Li, W. L. Wang, and L. Li, "Optimization of water resources utilization by multi-objective moth-flame algorithm," *Water Resources Management*, vol. 47, no. 10, pp. 3303–3316, 2018.
- [11] S. Mirjalili, P. Jangir, and S. Saremi, "Multi-objective ant lion optimizer: a multi-objective optimization algorithm for solving engineering problems," *Applied Intelligence*, vol. 46, no. 1, pp. 79–95, 2017.
- [12] B. Li, J. Li, K. Tang, and X. Yao, "Many-objective evolutionary algorithms: a survey," *ACM Computing Surveys (CSUR)*, vol. 48, no. 1, p. 13, 2015.
- [13] M. Farina and P. Amato, "On the optimal solution definition for many-criteria optimization problems," in *Proceedings of the Annual Meeting of the North American Fuzzy Information Processing Society, NAFIPS 2002*, pp. 233–238, IEEE, New Orleans LA, USA, June 2002.
- [14] P. J. Fleming, R. C. Purshouse, and R. J. Lygoe, "Many-objective optimization: an engineering design perspective," in *Proceedings of the International Conference on Evolutionary Multi-Criterion Optimization*, pp. 14–32, Springer, Guanajuato, Mexico, March 2005.
- [15] J. G. Herrero, A. Berlanga, and J. M. M. Lopez, "Effective evolutionary algorithms for many-specifications attainment: application to air traffic control tracking filters," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 1, pp. 151–168, 2009.
- [16] G. Fu, Z. Kapelan, J. R. Kasprzyk, and P. Reed, "Optimal design of water distribution systems using many-objective visual analytics," *Journal of Water Resources Planning and Management*, vol. 139, no. 6, pp. 624–633, 2012.
- [17] J. Knowles and D. Corne, "Quantifying the effects of objective space dimension in evolutionary multiobjective optimization," in *Proceedings of the Evolutionary Multi-Criterion Optimization*, pp. 757–771, Springer, Matsushima, Japan, March 2007.
- [18] P. Kata and X. Yao, "How well do multi-objective evolutionary algorithms scale to large problems," in *Proceedings of the IEEE Congress on Evolutionary Computation, CEC 2007*, pp. 3959–3966, IEEE, Singapore, September 2007.
- [19] H. Ishibuchi, N. Tsukamoto, and Y. Nojima, "Evolutionary many-objective optimization: a short review," in *Proceedings of the IEEE Congress on Evolutionary Computation, CEC 2008 (IEEE World Congress on Computational Intelligence)*, pp. 2419–2426, IEEE, Hong Kong, China, June 2008.
- [20] R. C. Purshouse and P. J. Fleming, "On the evolutionary optimization of many conflicting objectives," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 770–784, 2007.
- [21] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, "Combining convergence and diversity in evolutionary multiobjective optimization," *Evolutionary computation*, vol. 10, no. 3, pp. 263–282, 2002.
- [22] H. Aguirre and K. Tanaka, "Space partitioning with adaptive ϵ -ranking and substitute distance assignments: a comparative study on many-objective mnk-landscapes," in *Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation*, pp. 547–554, ACM, Montréal, Canada, July 2009.
- [23] H. Sato, H. Aguirre, and K. Tanaka, "Controlling dominance area of solutions and its impact on the performance of moeas," in *Proceedings of the Evolutionary Multi-Criterion Optimization*, pp. 5–20, Springer, Matsushima, Japan, March 2007.
- [24] H. Sato, H. Aguirre, and K. Tanaka, "Improved s-cdas using crossover controlling the number of crossed genes for many-objective optimization," in *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation*, pp. 753–760, ACM, Dublin, Ireland, July 2011.

- [25] K. Ikeda, H. Kita, and S. Kobayashi, "Failure of pareto-based moeas: does non-dominated really mean near to optimal?," in *Proceedings of the 2001 Congress on Evolutionary Computation*, vol. 2, pp. 957–962, IEEE, Seoul, Korea, May 2001.
- [26] Z. He, G. G. Yen, and J. Zhang, "Fuzzy-based pareto optimality for many-objective evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 2, pp. 269–285, 2014.
- [27] X. Zou, Y. Chen, M. Liu, and L. Kang, "A new evolutionary algorithm for solving many-objective optimization problems," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 38, no. 5, pp. 1402–1412, 2008.
- [28] J. Branke, T. Kaußler, and H. Schmeck, "Guidance in evolutionary multi-objective optimization," *Advances in Engineering Software*, vol. 32, no. 6, pp. 499–507, 2001.
- [29] G. Eichfelder, "Optimal elements in vector optimization with a variable ordering structure," *Journal of Optimization Theory and Applications*, vol. 151, no. 2, pp. 217–240, 2011.
- [30] S. Yang, M. Li, X. Liu, and J. Zheng, "A grid-based evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 5, pp. 721–736, 2013.
- [31] C. Zhu, L. Xu, and E. D. Goodman, "Generalization of pareto-optimality for many-objective evolutionary optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 2, pp. 299–315, 2016.
- [32] M. Li, S. Yang, and X. Liu, "Shift-based density estimation for pareto-based algorithms in many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 3, pp. 348–365, 2014.
- [33] Y. Yuan, H. Xu, B. Wang, and X. Yao, "A new dominance relation-based evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 1, pp. 16–37, 2016.
- [34] C. Zhou, G. Dai, and M. Wang, "Enhanced dominance and density selection based evolutionary algorithm for many-objective optimization problems," *Applied Intelligence*, vol. 1, pp. 1–21, 2017.
- [35] E. Zitzler and K. . Simon, "Indicator-based selection in multiobjective search," in *Proceedings of the International Conference on Parallel Problem Solving from Nature*, pp. 832–842, Springer, Birmingham, UK, September 2004.
- [36] N. Beume, B. Naujoks, and M. Emmerich, "Sms-emoa: multiobjective selection based on dominated hypervolume," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653–1669, 2007.
- [37] J. Bader and E. Zitzler, "HypE: an algorithm for fast hypervolume-based many-objective optimization," *Evolutionary computation*, vol. 19, no. 1, pp. 45–76, 2011.
- [38] K. Li, S. Kwong, J. Cao, M. Li, J. Zheng, and R. Shen, "Achieving balance between proximity and diversity in multi-objective evolutionary algorithm," *Information Sciences*, vol. 182, no. 1, pp. 220–242, 2012.
- [39] R. H. Gómez and C. A. C. Coello, "Mombi: a new meta-heuristic for many-objective optimization based on the r2 indicator," in *Proceedings of the 2013 IEEE Congress on Evolutionary Computation (CEC)*, pp. 2488–2495, IEEE, Cancun, Mexico, June 2013.
- [40] O. Schütze, X. Esquivel, A. Lara, and C. A. C. Coello, "Using the averaged hausdorff distance as a performance measure in evolutionary multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 4, pp. 504–522, 2012.
- [41] B. Li, K. Tang, J. Li, and X. Yao, "Stochastic ranking algorithm for many-objective optimization based on multiple indicators," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 6, pp. 924–938, 2016.
- [42] Y. Sun, G. G. Yen, and Y. Zhang, "IGD indicator-based evolutionary algorithm for many-objective optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 2, pp. 173–187, 2018.
- [43] Q. Zhang and H. Li, "MOEA/D: a multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on evolutionary computation*, vol. 11, no. 6, pp. 712–731, 2007.
- [44] M. Asafuddoula, T. Ray, and R. Sarker, "A decomposition-based evolutionary algorithm for many objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 3, pp. 445–460, 2015.
- [45] K. Li, K. Deb, Q. Zhang, and S. Kwong, "An evolutionary many-objective optimization algorithm based on dominance and decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 5, pp. 694–716, 2015.
- [46] Y. Yuan, H. Xu, B. Wang, B. Zhang, and X. Yao, "Balancing convergence and diversity in decomposition-based many-objective optimizers," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 2, pp. 180–198, 2016.
- [47] R. Wang, Z. Zhou, H. Ishibuchi, T. Liao, and T. Zhang, "Localized weighted sum method for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 1, pp. 3–18, 2018.
- [48] P. Kata and X. Yao, "A new multi-objective evolutionary optimisation algorithm: the two-archive algorithm," in *Proceedings of the 2006 International Conference on Computational Intelligence and Security*, vol. 1, pp. 286–291, IEEE, Guangzhou, China, November 2006.
- [49] H. Wang, L. Jiao, and X. Yao, "Two_arch2: an improved two-archive algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 4, pp. 524–541, 2015.
- [50] Y. Xiang, Y. Zhou, M. Li, and Z. Chen, "A vector angle-based evolutionary algorithm for unconstrained many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 1, pp. 131–152, 2017.
- [51] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based non-dominated sorting approach, Part I: solving problems with box constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2014.
- [52] X. Bi and C. Wang, "A niche-elimination operation based NSGA-III algorithm for many-objective optimization," *Applied Intelligence*, vol. 48, no. 1, pp. 118–141, 2018.
- [53] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A reference vector guided evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 5, pp. 773–791, 2016.
- [54] S. Jiang and S. Yang, "A strength pareto evolutionary algorithm based on reference direction for multiobjective and many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 3, pp. 329–346, 2017.
- [55] Y. Liu, D. Gong, X. Sun, and Y. Zhang, "Many-objective evolutionary optimization based on reference points," *Applied Soft Computing*, vol. 50, pp. 344–355, 2017.
- [56] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, 1997.
- [57] M. Li, S. Yang, X. Liu, and R. Shen, "A comparative study on evolutionary algorithms for many-objective optimization," in

- Proceedings of the International Conference on Evolutionary Multi-Criterion Optimization*, pp. 261–275, Springer, Sheffield, UK, March 2013.
- [58] I. Das and J. E. Dennis, “Normal-boundary intersection: a new method for generating the pareto surface in nonlinear multicriteria optimization problems,” *SIAM Journal on Optimization*, vol. 8, no. 3, pp. 631–657, 1998.
 - [59] H.-L. Liu, F. Gu, and Q. Zhang, “Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems,” *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 3, pp. 450–455, 2014.
 - [60] H. R. Tizhoosh, “Opposition-based learning: a new scheme for machine intelligence,” in *Proceedings of the International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC’06)*, vol. 1, pp. 695–701, IEEE, Vienna, Austria, November 2005.
 - [61] S. Mahdavi, S. Rahnamayan, and K. Deb, “Opposition based learning: a literature review,” *Swarm and evolutionary computation*, vol. 39, pp. 1–23, 2018.
 - [62] S. Rahnamayan, H. R. Tizhoosh, and M. M. A. Salama, “Opposition-based differential evolution,” in *Proceedings of the IEEE Symposium on Foundations of Computational Intelligence, FOCI 2007*, pp. 81–88, Honolulu, HI, USA, April 2007.
 - [63] W. L. Wang, W. K. Li, Z. Wang, and L. Li, “Opposition-based multi-objective whale optimization algorithm with global grid ranking,” *Neurocomputing*, vol. 341, pp. 41–59, 2019.
 - [64] K. Deb and R. B. Agrawal, “Simulated binary crossover for continuous search space,” *Complex Systems*, vol. 9, no. 3, pp. 115–148, 1994.
 - [65] W. Wang, S. Ying, L. Li, Z. Wang, and W. Li, “An improved decomposition-based multiobjective evolutionary algorithm with a better balance of convergence and diversity,” *Applied Soft Computing*, vol. 57, pp. 627–641, 2017.
 - [66] A. Carlos and C. Coello, “A multi-objective particle swarm optimizer based on decomposition,” in *Proceedings of the Conference on Genetic and Evolutionary Computation*, pp. 69–76, Dublin, Ireland, July 2011.
 - [67] G. Raquel Hernández and C. A. C. Coello, “Improved metaheuristic based on the r2 indicator for many-objective optimization,” in *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation*, pp. 679–686, ACM, Madrid, Spain, July 2015.
 - [68] K. Deb, M. Mohan, and S. Mishra, “Towards a quick computation of well-spread pareto-optimal solutions,” in *Proceedings of the International Conference on Evolutionary Multi-Criterion Optimization*, pp. 222–236, Faro, Portugal, April 2003.
 - [69] Z. He and G. G. Yen, “Many-objective evolutionary algorithm: objective space reduction and diversity improvement,” *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 1, pp. 145–160, 2016.
 - [70] D. Roman, L. Costa, and I. E. Santo, “Many-objective optimization using differential evolution with variable-wise mutation restriction,” in *Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation*, pp. 591–598, ACM, Amsterdam, Netherlands, July 2013.
 - [71] X. Ma, F. Liu, Y. Qi et al., “A multiobjective evolutionary algorithm based on decision variable analyses for multi-objective optimization problems with large-scale variables,” *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 2, pp. 275–298, 2016.
 - [72] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, “Scalable multi-objective optimization test problems,” in *Proceedings of the 2002 Congress on Evolutionary Computation, CEC’02*, pp. 825–830, Honolulu, HI, USA, May 2002.
 - [73] S. Huband, P. Hingston, L. Barone, and L. While, “A review of multiobjective test problems and a scalable test problem toolkit,” *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 5, pp. 477–506, 2006.
 - [74] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. Da Fonseca, “Performance assessment of multiobjective optimizers: an analysis and review,” *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 117–132, 2003.
 - [75] L. While, P. Hingston, L. Barone, and S. Huband, “A faster algorithm for calculating hypervolume,” *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 1, pp. 29–38, 2006.
 - [76] A. Auger, J. Bader, D. Brockhoff, and E. Zitzler, “Theory of the hypervolume indicator: optimal μ -distributions and the choice of the reference point,” in *Proceedings of the Tenth ACM SIGEVO Workshop on Foundations of Genetic Algorithms*, pp. 87–102, ACM, Orlando, Florida, USA, January 2009.
 - [77] H. Ishibuchi, Y. Hitotsuyanagi, N. Tsukamoto, and Y. Nojima, “Many-objective test problems to visually examine the behavior of multiobjective evolution in a decision space,” in *Proceedings of the International Conference on Parallel Problem Solving From Nature*, pp. 91–100, Kraków, Poland, September 2010.
 - [78] L. While, L. Bradstreet, and L. Barone, “A fast way of calculating exact hypervolumes,” *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 1, pp. 86–95, 2012.