A Hybridized Angle-encouragement-based Decomposition Approach for Many-objective Optimization Problems

Yuchao Su[1], Jia Wang*[1], Lijia Ma[1], Xiaozhou Wang[1], Qiuzhen Lin[1], Carlos A. Coello Coello[2], Jianyong Chen[1]

[1]*College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, PR.China,*

[2] *CINVESTAV-IPN, Department of Computer Science, Mexico, D.F., 07360, Mexico*

Abstract:

Due to the large objective space when handling many-objective optimization problems (MaOPs), it is a challenging work for multi-objective evolutionary algorithms (MOEAs) to balance convergence and diversity during the search process. Although a number of decomposition-based MOEAs have been designed for the above purpose, some difficulties are still encountered for tackling some difficult MaOPs. As inspired by the existing decomposition approaches, a new Hybridized Angle-Encouragement-based (HAE) decomposition approach is proposed in this paper, which is embedded into a general framework of decomposition-based MOEAs, called MOEA/D-HAE. Two classes of decomposition approaches, *i.e.*, the angle-based decomposition and the proposed encouragement-based boundary intersection decomposition, are sequentially used in HAE. The first one selects appropriate solutions for association in the feasible region of each subproblem, which is expected to well maintain the diversity of associated solutions. The second one acts as a supplement for the angle-based one under the case that no solution is located in the feasible region of subproblem, which aims to ensure the convergence and explore the boundaries. By this way, HAE can effectively combine their advantages, which helps to appropriately balance convergence and diversity in evolutionary search. To study the effectiveness of HAE, two series of well-known test MaOPs (***WFG*** and ***DTLZ***) are used. The experimental results validate the advantages of HAE when compared to other classic decomposition approaches and also confirm the superiority of MOEA/D-HAE over seven recently proposed many-objective evolutionary algorithms.

Keywords: Many-objective optimization problem; Angle-based decomposition; Encouragement-based decomposition

1. Introduction

A many-objective optimization problem (MaOP) often contains more than three objectives to be optimized simultaneously, which is extended from the definition of multi-objective optimization problem (MOP). Generally, a MaOP can be formulated as follows.

$$\min(F(x)) = \min(f_1(x), f_2(x), ..., f_m(x)), \text{ s.t. } x \in \Theta\,, \tag{1}$$

---

* Corresponding author
Email address: jia.wang@szu.edu.cn (J. Wang)

where x is a decision vector, m is the number of objectives, Θ indicates the decision space, and $F(x)$ includes a set of optimization objectives as represented by $f_i(x)$ (i =1, 2,…, m, and m>3). For MaOPs, there often exists a set of equally optimal solutions (termed Pareto-optimal set (***PS***)) when considering all the objectives, and the mapping of ***PS*** on the objective space is termed Pareto-optimal front (***PF***) [1].

During the last decades, multi-objective evolutionary algorithms (MOEAs) have become the popular and effective approaches when tackling MOPs [2-6]. According to the selection criteria for population update, most MOEAs can be categorized into three classes, *i.e.*, Pareto-based MOEAs [7-8], indicator-based MOEAs [9] and decomposition-based MOEAs [10-11]. These MOEAs are validated to show the very promising performance on tackling various kinds of MOPs. However, due to the curse of dimensionality in MaOPs, their performance deteriorates significantly when the number of optimization objectives is increased [12-13]. Pareto-based MOEAs cannot provide sufficient pressure toward the true ***PF*** due to the existence of a large number of non-dominated solutions for MaOPs [14-15]. Decomposition-based MOEAs have to specify a large set of weight vectors for the high dimensional objective space in MaOPs [16]. Moreover, the match on the shapes of weight vectors and the true ***PF*** strongly affects their performance [17-18]. Indicator-based MOEAs often suffer a high computational cost, which makes them not so popular and efficient for solving MaOPs in practical applications [19-20].

To solve the above challenges in MOEAs, some many-objective evolutionary algorithms (MaOEAs) are designed recently. Regarding Pareto-based MOEAs, the original Pareto-based dominance relation is modified to enhance the convergence pressure for solving MaOPs, such as fuzzy-dominance [21], corner sorting [22], reference point-based dominance [23], and generalized Pareto-optimality [24]. Another typical way is to strengthen the diversity management in Pareto-based MOEAs, like the shift-based density estimation in SPEA2-SDE [25] and the use of associated reference points for association in NSGA-III [26]. For indicator-based MOEAs, the hyper-volume (HV) indicator is re-calculated to be more efficient [27-29] and other performance indicators (*e.g.*, R2 [30-31], the additive approximation [32], and the combination indicators [33]) are also designed for MaOPs. With respect to decomposition-based MOEAs, two adaptive generation methods for weight vectors [34-35] are presented for MaOPs, the Pareto-based dominance is combined with the decomposition approach for solving MaOPs in [16], and two external archives are used to respectively ensure convergence and diversity on tackling MaOPs [36]. Moreover, there are several new MaOEAs using the vector angles for population update (*e.g.*, VaEA [37], MaOEA-DDFC [38], and MaOEA-CSS [39]) and other heuristic algorithms designed for MaOPs (*e.g.*, preference-inspired co-evolutionary algorithm with goals (PICEA-g) [40] and many-objective particle swarm optimization algorithms [41-42]), which have also shown the promising performance.

In this paper, we mainly focus on the decomposition approaches for MaOEAs. As pointed out in [17], the performance of decomposition-based MOEAs strongly depends on the shapes of weight vectors and the true ***PF***. In the case that decomposition-based MOEAs adopt the same set of weight vectors, the used decomposition approaches will also significantly affect the performance on tackling MaOPs, as revealed by [17]. Traditional decomposition methods, such as the weighted sum (WS) approach, the Tchebycheff (TCH) approach and the penalty-based boundary intersection (PBI) approach, have their own advantages in solving different kinds of MaOPs [17], which are further studied and enhanced in [43-45]. However, when tackling MaOPs, their performance is still not so promising, thus an invert PBI (iPBI) is designed in [46], a local weighted sum (LWS) decomposition approach is presented in MOEA/D-LWS [47] and an adaptive Pareto Front scalarizing (PaS) as well as an adaptive Pareto Front penalty-based boundary intersection (PaP) decomposition approach are proposed in [48] and [44], respectively, to match the true ***PF***s.

Following the work of these modified decomposition approaches [43, 47-48], this paper presents a hybridized angle-encouragement-based (HAE) decomposition method for MaOPs, which is embedded into a general decomposition-based MOEA, called MOEA/D-HAE. Two types of decomposition, *i.e.*, an angle-based decomposition and an encouragement-based boundary intersection (EBI) decomposition, are sequentially used in HAE. At first, the angle-based one is used to guarantee the population diversity, by only associating solution to each subproblem from its feasible region. When the subproblem has no associated solution, it indicates that the diversity on this subproblem is weak and thus the convergence should be emphasized in turn. In this case, the encouragement-based one is used as a supplement for associating solution, aiming to ensure the convergence. By combining their advantages, HAE is more effective to solve MaOPs. The well-known test problems (*i.e.*, ***WFG*** [49] and ***DTLZ*** [50] with 4, 6, 8, and 10 objectives) are used to verify the effectiveness of HAE and the experiments show the advantages of HAE over some traditional decomposition methods. Moreover, the experiments also confirm the superior performance of MOEA/D-HAE over seven competitive MaOEAs (*i.e.*, NSGA-III [26], MOEA/DD [16], SRA [33], MaOEA-R&D [51], VaEA [37], Two_Arch2 [52], and MaOEA-CSS [39]) when tackling most of the adopted test problems. Especially, four competitive decomposition approaches designed for MaOPs (iPBI, LWS, PaP and PaS) are also included to compare with HAE when they are all embedded into [53], which further confirms the superiority of HAE.

The rest of this paper is organized as follows. Section 2 introduces the related work about the existing decomposition approaches. The details of MOEA/D-HAE are given in Section 3, while the experimental results are provided in Section 4. At last, the conclusions and future work are presented in Section 5.

2. Traditional Decomposition Approaches

The commonly used traditional decomposition approaches include the WS [11], TCH [45] and PBI [11] methods, which are respectively introduced below. Please note that $\mathbf{w} = (w_1, w_2, ..., w_m)^T$ is a weight vector used in these methods with $w_i \geq 0$ and $\sum_{i=1}^{m} w_i = 1$, where m is the number of objectives.

The WS approach is defined by

$$\arg\min_{x\in\Omega}(g^{ws}(x \mid w)) = \arg\min_{x\in\Omega}(\textstyle\sum_{i=1}^{m} f_i(x) w_i) \,, \tag{2}$$

where Ω is a solution set. As pointed out in [43], this WS approach shows poor performance when solving the MOPs with concave true ***PF***s, mainly due to its poor ability to maintain diversity for each subproblem.

The TCH approach is formulated by

$$\arg\min_{x\in\Omega}(g^{tch}(x \mid w, z^*)) = \arg\min_{x\in\Omega}(\max(\mid f_i(x) - z_i^* \mid / w_i)) \,, \tag{3}$$

where $z^* = (z_1^*, z_2^*, ..., z_m^*)$ is an ideal point in objective space and $i = 1, ..., m$. This TCH method is mostly used in many MOEAs [53-58], since it can properly balance convergence and diversity.

The PBI approach is introduced by

$$\arg\min_{x\in\Omega}(g^{pbi}(x \mid w, z^*)) = \arg\min_{x\in\Omega}(d_1^x + \theta\, d_2^x)$$

$$\text{where } d_1^x = \frac{\|(F(x) - z^*)w\|}{\|w\|} \text{ and } d_2^x = \left\| F(x) - (z^* + d_1^x \frac{w}{\|w\|}) \right\|,\ \theta \geq 0 \quad , \tag{4}$$

where d_1^x is the distance of the ideal point z* and the foot point from $F(x)$ to the weight vector w, d_2^x is the distance of $F(x)$ and the weight vector w [11], and θ is a pre-set parameter to control the impact of d_2^x. Please note that d_1^x is used to reflect the convergence of x, while d_2^x is a kind of measurement to show the diversity of x for its subproblem. By summarizing the values of d_1^x and θd_2^x, this PBI method gives a composite measure for convergence and diversity [16], while the bias to convergence or diversity can be adjusted by the parameter θ.

To further show the difference of WS, TCH and PBI, Fig. 1 gives a schematic illustration to explain their preferences when solving a minimal optimization problem with two objectives, in which the solution marked with the red color is better than the one identified by the yellow color according to the preference. In fact, any solution in the gray-color region is better than the solution marked with yellow color.

As observed from Fig. 1 (a) and (b), the WS and TCH approaches may associate the far-away solution to the subproblem, in which a little promotion of the convergence may cause the relatively significant loss of the diversity. In Fig. 1 (c), the replacement of the PBI approach may cause the loss of the convergence and the improvement of the diversity, while another case plotted in Fig. 1(d) shows an opposed effect to that of Fig. 1(c). It can be summarized that it is always a difficult work to achieve the balance of convergence and diversity in MaOPs.

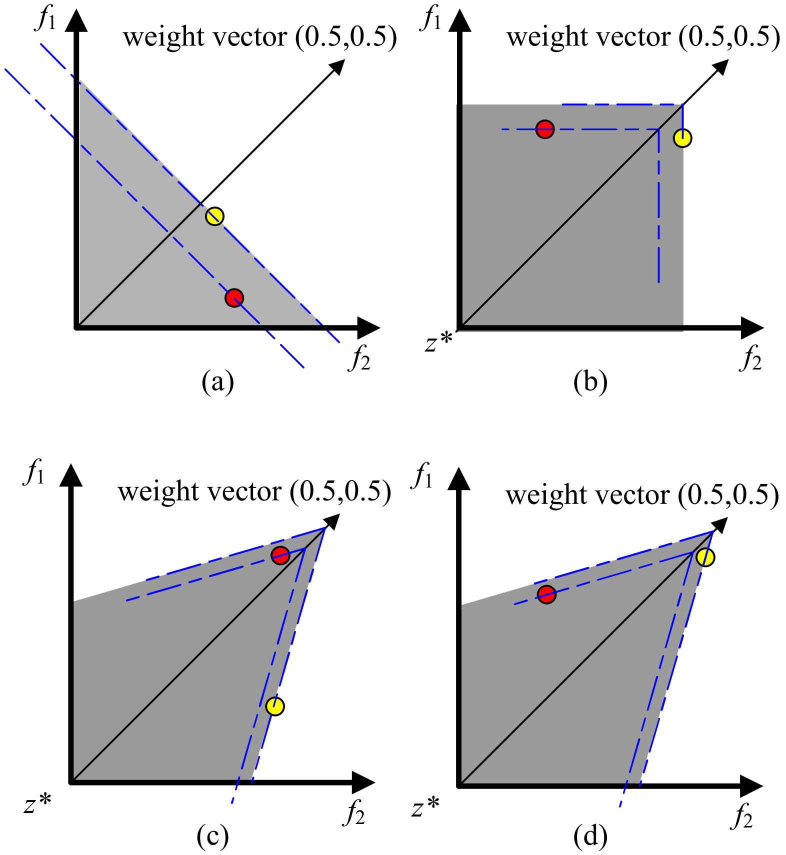
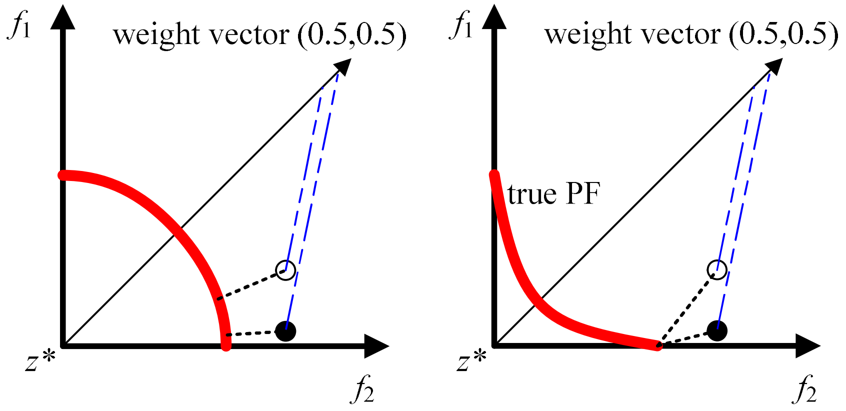


Fig. 1 An illustration to show the preferences of (a) WS, (b) TCH, (c) PBI, (d) PBI.

As studied in [17], the performance of decomposition-based MOEAs strongly depends on the shapes of weight vectors and the true ***PF***. When the same weight vectors are used to construct the subproblems, their performance will also highly rely on the used decomposition approaches. As defined in Eqs. (2)-(4), their candidate solutions associated to the subproblems are selected from the solution set Ω. However, in most decomposition-based MOEAs [53-58], Ω is generally composed by the neighboring solutions of one subproblem or the entire population. On some cases, the solution associated to the subproblem may be substituted by a far-away solution, as visually shown in Fig. 1(a) for WS and Fig. 1(b) for TCH. Regarding PBI, it may not always perform well when the population is crowded and some subproblems may not find any solution in their vicinities, especially in the early evolutionary stage. For example, in Fig. 2(a) with convex ***PF*** and Fig. 2(b) with concave ***PF***, the lengths of the dashed lines indicate the distances from the solutions to the true ***PF***s. In these two cases, PBI always prefers the far-away solutions, as the diversity estimations in these cases are ineffective, which result in a degenerate convergence. It is actually meaningless for PBI to consider diversity under these two cases as the associated solutions are already far away from the subproblem.



(a) (b)

Fig. 2 An illustration of solution association using PBI with $\theta > 1$

In order to avoid the above cases in Figs. 1-2, we design a hybridized angle-encouragement-based decomposition approach in this paper and embed it into a general framework of decomposition-based MOEA [54, 58] to validate the advantages. First, the current population and all the offspring are combined and the diversity for each subproblem is considered first when selecting the next population. For the cases that the subproblems can find the solutions around their vicinities in the objective space, the angle-based decomposition approach is used to select the associated solutions, so as to maintain the diversity, as a subproblem only links to one of its nearest solutions under this case. Second, regarding the case that the subproblem has no solution in its feasible region, the EBI decomposition approach is used to choose the solution from the union population, aiming to ensure the convergence without considering diversity and to search some extreme regions of ***PF***s.

3. The Proposed MOEA/D-HAE

3.1 Angle-based Decomposition Approach (AD)

It is difficult to balance convergence and diversity when solving MaOPs. When the convergence is always emphasized, the algorithm may easily fall into local optimal and induce a premature convergence. However, if the diversity is always preferred, the convergence speed may be significantly affected. In our approach, the angle-based decomposition approach is used to guarantee the diversity first, which selects one solution from the feasible region of each subproblem in the objective space. This angle-based decomposition model is defined by

$$\arg\min_{x \in \Omega^k}(\mathrm{g}(x \mid w^k, z^*)) \quad , \tag{5}$$

where z^* is the ideal point and Ω^k is the feasible region for each subproblem with w^k, as defined by

$$\Omega^k = \{x \mid \arg\min_{w \in W}(\mathrm{angle}(F(x), w, z^*)) = w^k\}, \tag{6}$$

where $\mathrm{angle}(F(x), w, z^*)$ returns the angle of F(x) and the weight vector w from the starting point z^*, as defined by

$$\mathrm{angle}(F(x), w, z^*) = \arccos\left|(\textstyle\sum_{i=1}^{m}\left(f_i(x) - z_i^*\right)\cdot w_i^k) / (\sqrt{\textstyle\sum_{i=1}^{m}\left(f_i(x) - z_i^*\right)^2} \cdot \sqrt{\textstyle\sum_{i=1}^{m}\left(w_i^k\right)^2})\right|. \tag{7}$$

With Eqs. (5-7), Ω^k will include all the solutions that are closest to *k*th subproblem when considering all the angles of the solution and the used weight vectors. Assume that the population size is *N* and the weight vectors are $W = \{w^1, w^2, ..., w^N\}$. Apparently, Eq. (5) can be considered as an extension of Eqs. (2)-(4), which defines the feasible region Ω^k for *k*th subproblem. To further clarify this approach, the pseudo-code of angle-based decomposition approach is provided in **Algorithm 1** with the input ***Q*** (*i.e.*, a solution set formed by the current population and their offspring), where ***R*** is used to preserve the weight vectors (*i.e.*, subproblems) that are not associated to any solution and ***S*** is adopted to store the solutions associated to the weight vectors. In line 1 of **Algorithm 1**, ***R*** and ***S*** are all set as an empty set. For each subproblem in line 2, if its feasible region can find any solution, the one with the best aggregated value using Eq. (5) will be added into ***S*** in line 4. Otherwise, the weight vectors will be added into ***R*** in line 6. At last, these two sets (***R*** and ***S***) are returned in line 9.

Algorithm 1: AD(*Q*)

```
1  $R = \varnothing$, $S = \varnothing$;
2  for $i$=1 to $N$
3      if $\Omega^i \neq \varnothing$   // these sets are obtained by Eq. (5)
4          $S = S \cup \arg\min(\mathrm{g}(x \mid w^i, z^*))$ ; // $x \in \Omega^i$
5      Else
6          $R = R \cup w^i$ ;
7      end if
8  end for
9  return [$R$, $S$];
```

To visually show the angle-based decomposition approach, Fig. 3 gives a schematic illustration to show its running to associate the solutions for the subproblems, in which the feasible space for each subproblem is encompassed by two closest dash lines, such as the region with blue background. According to Eq. (5), the solutions marked with the red color are selected for association. Obviously, the population ***S*** will include the solutions with good diversity and convergence for all the subproblems. It is also observed that the number of solutions in the population ***S*** may be less than the number of weight vectors, as no solution is located in the feasible regions of w_3 and w_4. Thus, to associate the solutions to w_3 and w_4 under this case, the EBI approach is used, as introduced in Section 3.2.

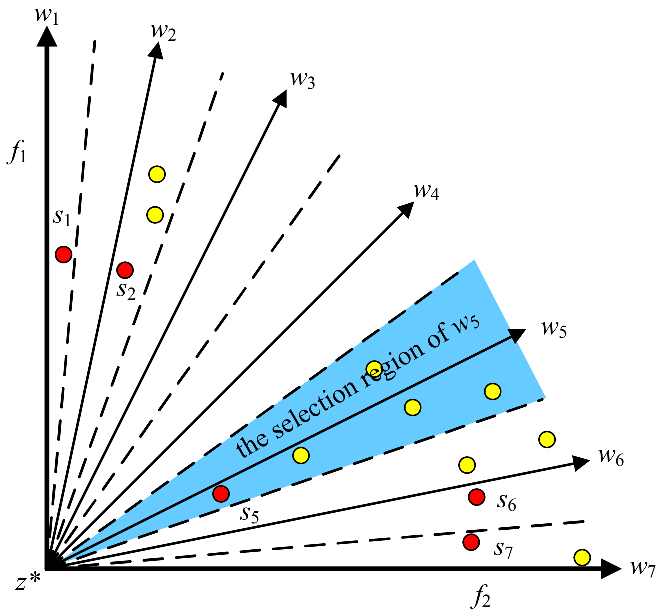


Fig. 3 An illustration to show the running of using the angle-based decomposition

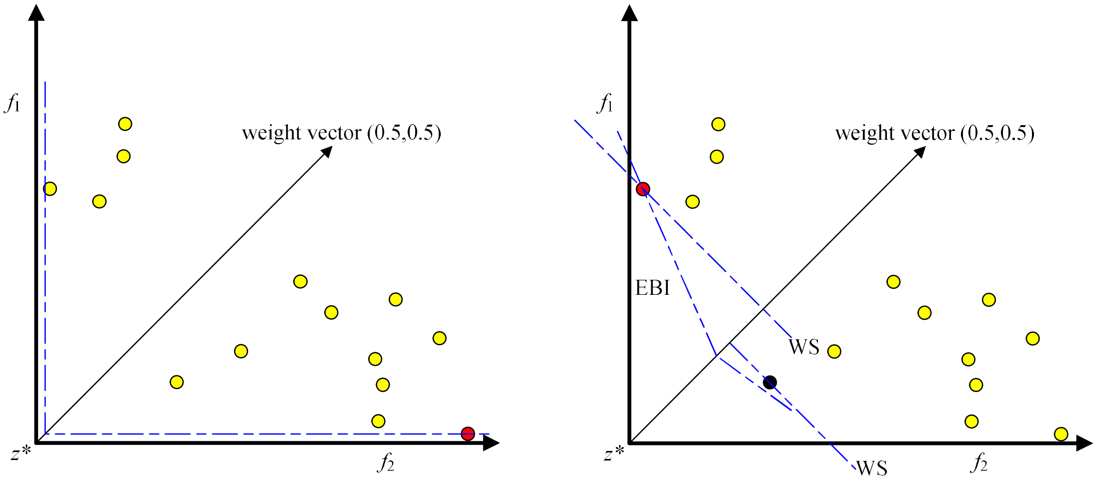
3.2 The EBI Decomposition Approach

When some subproblems are not associated with solutions using the angle-based decomposition, the EBI decomposition is further used to select the solutions from the current population and their offspring, only considering convergence and ignoring diversity. Using this approach, the solutions in boundary are preferred to extend the entire true ***PF***. This EBI approach is defined by

$$\arg\min_{x\in\Omega}(g^{ebi}(x\,|\,w,z^*)) = \arg\min_{x\in\Omega}(d_1^x - \theta d_2^x)$$

$$\text{where } d_1^x = \frac{\|(F(x)-z^*)w\|}{\|w\|} \text{ and } d_2^x = \left\|F(x)-(z^* + d_1^x\frac{w}{\|w\|})\right\|,\ \theta \ge 0 \cdot \qquad (8)$$

The definitions of d_1^x and d_2^x are the same to that in PBI (*i.e.*, Eq. (4)). However, different from the original PBI approach, the EBI approach in Eq. (8) changes the sign of θ and re-defines the solution set Ω which consists of the current population and their offspring. To show the selection preference of the EBI approach, Fig. 4 is provided by using the EBI approach with different θ values. As observed from Fig. 4, the EBI approach with $\theta = 1$ is able to find the solutions in the boundaries of current population. However, when $0 < \theta < 1$, the exploration for the boundaries is weakened, whereas the ability for speeding up convergence is enhanced. It can be observed from Fig. 4(b) that the WS approach selects the solution plotted by the black circle, while the EBI approach prefers the one marked by the red circle that is a suboptimal solution for the WS approach. Therefore, the HAE approach with $0 < \theta < 1$ is more effective to balance convergence and diversity in this case.



(a) (b)

Fig. 4 An illustration of the EBI approach with (a) $\theta = 1$ and (b) $0 < \theta < 1$

As introduced in Section 3.1, this EBI approach is employed for the subproblems that are not associated with any solution in their feasible regions. Under this case, the EBI approach aims to accelerate the convergence speed and to explore the boundary area of current population. To further clarify the running of EBI, its pseudo-code is given in **Algorithm 2**, with the input $\boldsymbol{Q}$ (a solution set) and $\boldsymbol{R}$ (a set of weight vectors that are not associated to solutions returned by **Algorithm 1**). **Algorithm 2** is only run as a supplement for **Algorithm 1**, as it is difficult for the angle-based decomposition approach to always work well in the whole evolutionary process (*i.e.*, some subproblems are often not associated in the early evolutionary stage due to the crowded population). In this case, the EBI approach can be used for solution association, so as to speed up the convergence and explore the boundaries of current population. In line 1 of **Algorithm 2**, $\boldsymbol{S}$ is initialized as an empty set. For each subproblem in line 2, the ith subproblem is selected from $\boldsymbol{R}$ in line 3 and then Eq. (8) is used to select a solution from $\boldsymbol{Q}$ to associate with this subproblem, which is added into $\boldsymbol{S}$ and removed from $\boldsymbol{Q}$ in line 4. At last, return the set $\boldsymbol{S}$ with the associated solutions for $\boldsymbol{R}$ in line 6.

Algorithm 2: EBI($\boldsymbol{Q}$, $\boldsymbol{R}$)

```
1  $\boldsymbol{S} = \varnothing$;
2  for $i$=1 to $|\boldsymbol{R}|$
3      Select the $i$th subproblem from $\boldsymbol{R}$;
4      Use Eq. (8) to select a solution from $\boldsymbol{Q}$ to associate with the
       subproblem, which is added into $\boldsymbol{S}$ and removed from $\boldsymbol{Q}$;
5  end for
6  return $\boldsymbol{S}$;
```

3.3 The Proposed HAE Approach and MOEA/D-HAE

The proposed HAE approach coordinates the running of the above angle-based decomposition approach in Section 3.1 and the EBI decomposition approach in Section 3.2. The angle-based one is first used to associate the solutions in the feasible region of each subproblem, which aims to maintain diversity. For the case that the subproblem has no solution in its feasible region, the EBI decomposition approach is further used to select the solutions from the union population, which ensures convergence and helps to search the boundary regions of current population. Here, the flow chart of MOEA/D-HAE is provided in Fig. 5 to have a clear understanding of its running. At the start of MOEA/D-HAE, the target problem and its parameters are inputted, and then MOEA/D-HAE is initialized by setting the related parameters. Then, the algorithm will go through the evolution to generate *N* offspring, where *N* denotes the population size. After that, all the offspring and parents are collected to do the selection using AD and EBI. At the end of each generation, the population will be outputted if the termination condition is satisfied. Otherwise, the above evolutionary and selection process will be run again.

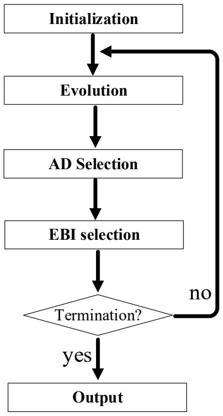


Fig. 5 The flow chart of MOEA/D-HAE

To show the actual running of HAE approach, Fig. 6 is given to show the solution association of HAE. The angle-based decomposition is first used to associate w_1, w_2, w_5, w_6, w_7, as shown in Fig. 3. Then, s_3 will be selected to associate w_3 according to the EBI method. Although s_3 and s_5 are the same in Fig. 6, they show the promising ability to accelerate the convergence speed. Similarly, s_4 is selected for the weight vector w_4 according to the EBI method. Even s_1 and s_4 are the same in Fig. 6, they are close to the boundaries of the population and help to extend the approximate ***PF***. Therefore, using the proposed HAE approach, the solutions marked by the red circle are associated to the corresponding subproblems.

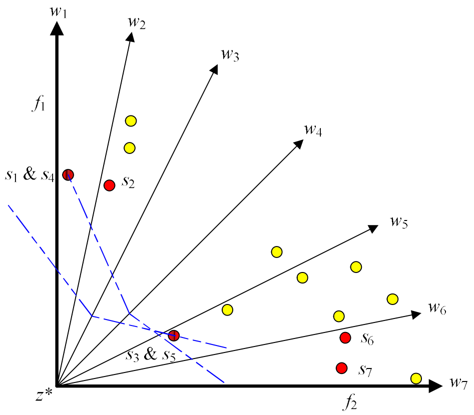


Fig. 6 An illustration of the HAE approach with $0<\theta<1$.

The proposed HAE approach is embedded into a general framework of MOEA/D [11], forming the so-called MOEA/D-HAE. To clarify the running of MOEA/D-HAE, its pseudo code is given in **Algorithm 3**, where *FEs* and *maxFEs* respectively indicate the counter for current function evaluation and the number of maximal function evaluation, *T* is the number of neighbors for subproblems, *G* and *maxG* are respectively the current generation and the maximal generation.

```
Algorithm 3: MOEA/D-HAE
1   Initialize the weight vectors and population $\boldsymbol{P}$, evaluate each individual and update $z^*$;
2   while $FEs<maxFEs$
3       $T$=max(|$\boldsymbol{P}$|×(1-$G/maxG$), 3) and $\boldsymbol{Q}=\varnothing$;
4       for $i$=1 to $N$
5           Select one solution from the $T$ neighboring solutions;
6           Generate one offspring by the solution associated to $i$th subproblem and its
            neighbors using SBX and PM;
7           Evaluate each offspring, add it into $\boldsymbol{Q}$, and update $z^*$;
8       end for
9       $\boldsymbol{Q}=\boldsymbol{P}\cup\boldsymbol{Q}$;
10      [$\boldsymbol{R}$, $\boldsymbol{P}$]=AD($\boldsymbol{Q}$); // Algorithm 1
11      if |$\boldsymbol{P}$|<$N$
12          $\boldsymbol{P}=\boldsymbol{P}\cup$EBI($\boldsymbol{Q}$, $\boldsymbol{R}$); // Algorithm 2
13      end if
14  end while
15  output $\boldsymbol{P}$;
```

In line 1 of **Algorithm 3**, all the used weight vectors are initialized, the evolutionary population $\boldsymbol{P}$ is randomly generated, and all the individuals are evaluated to update the ideal point z^*. In line 2, if *FEs* is smaller than *maxFEs*, the following evolutionary process is executed. In line 3, the T value is decreased with the increasing number of generations, which is set to the maximal value of $|\boldsymbol{P}|\times(1-G/maxG)$ and 3 (please note that this constraint is to ensure the minimal number of neighbors). The offspring set $\boldsymbol{Q}$ is initialized as an empty set. In lines 4-8, all the individuals are evolved using the simulated binary crossover (SBX) [59] and polynomial mutation (PM) [59]. Please note the mating pool in MOEA/D-HAE is only composed of the T neighboring solutions. The new offspring are preserved in $\boldsymbol{Q}$ and evaluated to update the ideal point z^*. In line 9, the parent population ***P*** and the offspring population ***Q*** are combined into ***Q***. Then, the angle-based decomposition approach is adopted in line 10 to associate solution for each subproblem and ***R*** is the set of weight vectors that have not any solution in their feasible regions. The running of AD (**Algorithm 1**) has been introduced in Section 3.1. When $|\boldsymbol{P}|<N$ in line 11, ***R*** is not empty as the angle-based decomposition approach only selects one solution for each subproblem. In line 12, the EBI approach (**Algorithm 2**) is further used to select appropriate solutions for the un-associated subproblems to ensure convergence. The above evolutionary process will be repeated until *maxFEs* is reached. At last, the solutions in ***P*** are outputted as the final approximation set.

4. Experimental Studies

In this section, the performance of MOEA/D-HAE is studied when tackling the ***WFG*** [49] and ***DTLZ1-DTLZ4*** [50] test problems. In the following subsections, some related information about the experiments is provided, including the used test problems, the performance indicators, and the parameters settings of the compared algorithms (NSGA-III [26], MOEA/DD [16], SRA [33], MaOEA-R&D [51], VaEA [37], Two_Arch2 [52], and MaOEA-CSS [39]). Then, the experimental results of the compared algorithms are given to validate the performance of MOEA/D-HAE. Especially, HAE as a decomposition approach is compared to some competitive decomposition approaches proposed recently (iPBI [46], LWS [47], PaP [44] and PaS [48]), which shows the advantages of HAE. Finally, the effects of the parameter θ in HAE are analyzed.

4.1 The Used Test Problems

The ***WFG*** and ***DTLZ*** test suites are widely used in many experimental studies for performance comparisons among different MaOEAs [16, 26, 33]. In this study, thirteen frequently-used test MaOPs without any constraint, including ***WFG1-WFG9*** and ***DTLZ1-DTLZ4***, are used to validate the performance of MOEA/D-HAE in solving different MaOPs with 4, 6, 8, and 10 objectives. Please note that the number of decision variables in ***DTLZ*** is $m+k-1$, where m is the number of objectives and k is set to 5. For the ***WFG*** test problems, their decision variables include k position-related variables and l distance-related variables, where k is set to $2(m-1)$ and l is set to 20 as recommended in [16, 33, 37, 64].

4.2 Performance Indicators

The main purposes of solving MaOPs include two aspects, *i.e.*, convergence and diversity. More specifically, a set of solutions is preferred to closely approximate the true ***PF***s and to evenly cover the true ***PF***s. Thus, the widely used performance metric (*i.e.*, HV [8]) is used in our experiments to evaluate the comprehensive performance of all the compared algorithms in terms of both convergence and diversity. This HV metric is not required to know the true ***PF***s in advance, which can reflect the convergence and the diversity by computing the high dimensional volumes between the non-dominated solutions and a reference point z^*, as defined by

$$\mathrm{HV}(PF, z^*) = volume(\bigcup_{x \in PF} v(x, z^*)), \tag{7}$$

where $v(x, z^*)$ is the hypercube created with x and z^*. Any individual that is dominated by z^* does not contribute to the computation of HV. The HV value is calculated by using a reference point as $1.1 \times (f_1^{max}(x), f_2^{max}(x), ..., f_m^{max}(x))$ ($f_i^{max}(x)$ is the maximal value of the ith problems in MaOPs and i = 1, 2,…, m), as suggested in [33]. Please note that, the computation consumption of HV is expensive for MaOPs. Thus, we use the Monte Carlo simulation [19] to compute the HV value with 10^5 sample points. A large value for HV indicates that the approximate solution set has good convergence and diversity.

Table 1

Parameters settings in test problems with various dimensions in objective space

The number of objectives	*H* values	Population size	Maximal function evaluations
4	8, 0	165	82500
6	5, 0	252	126000
8	4, 0	330	165000
10	3, 2	275	192500

4.3. Comparison Experiments with Different MaOPs

In this study, MOEA/D-HAE is compared with seven competitive MaOEAs, including NSGA-III [21], MOEA/DD [11], SRA [28], MaOEA-R&D [46], VaEA [32], Two_Arch2 [47], and MaOEA-CSS [34]. All the parameters sets of the compared algorithms are the same to that in their original references except for the weight vectors and the used evolutionary operators. All the compared algorithms use SBX [59] and PM [59] to generate the offspring for a fair comparison. The crossover probability of SBX and the mutation probability of PM are set as 1.0 and 1/n, respectively, where n is the number of decision variables. The distribution indexes of SBX and PM are respectively set to 30 and 20. MOEA/D-HAE, MOEA/DD and NSGA-III are run in jMetal [60] due to its fast running, while the rest compared algorithms are executed in PlatEMO [61]. MOEA/D-HAE and MOEA/DD use the same weight vectors in the source code of MOEA/D-ACD [10] from http://www.cs.cityu.edu.hk/~qzhang/publications.html. The *H* values to calculate the number of weight vectors [27] are shown in Table 1 for NSGA-III. Moreover, Table 1 also shows the population size [36, 46, 62-64] and the number of maximal function evaluations. Other population sizes can be used for all the compared algorithms, which won't significantly affect the final comparison results according to our preliminary experiments. The HAE approach with $\theta = 0.1$ and the angle-based TCH approach are respectively used for **Algorithm 2** and **Algorithm 1** in MOEA/D-HAE. The hardware configurations for running these experiments are Intel i5 8600K@4.7Ghz

Table 2

The HV comparison results of MOEA/D-HAE and seven competitors on all the ***WFG*** test problems

Problem	*m*		NSGA-III	MOEA/DD	Two_Arch2	SRA	VaEA	MaOEA-R&D	MaOEA-CSS	MOEA/D-HAE
WFG1	4	Mean	3.71E-1 —	4.45E-1 —	**7.52E-1 +**	5.82E-1 —	4.97E-1 —	5.18E-1 —	7.22E-1 =	7.22E-1
		Std	3.48E-2	3.10E-2	**2.84E-2**	4.39E-2	3.86E-2	6.36E-2	4.04E-2	2.89E-2
	6	Mean	3.98E-1 —	4.96E-1 —	**6.90E-1 +**	5.85E-1 —	3.80E-1 —	3.54E-1 —	6.38E-1 —	6.51E-1
		Std	3.68E-2	2.49E-2	**3.36E-2**	3.41E-2	3.84E-2	5.51E-2	4.07E-2	5.42E-2
	8	Mean	5.28E-1 —	**6.80E-1 +**	6.71E-1 +	5.43E-1 —	3.07E-1 —	2.94E-1 —	6.45E-1 +	5.69E-1
		Std	4.33E-2	**2.84E-2**	3.59E-2	4.09E-2	2.20E-2	5.98E-2	3.30E-2	8.40E-2
	10	Mean	6.56E-1 —	3.83E-1 —	7.83E-1 +	5.67E-1 —	2.92E-1 —	2.36E-1 —	**8.09E-1 +**	7.70E-1
		Std	3.10E-2	3.57E-2	5.61E-2	4.49E-2	2.05E-2	3.52E-2	**6.02E-2**	7.20E-2
WFG2	4	Mean	9.34E-1 —	9.22E-1 —	9.28E-1 —	9.36E-1 —	9.36E-1 —	9.02E-1 —	8.89E-1 —	**9.64E-1**
		Std	6.56E-2	6.40E-2	7.95E-2	5.58E-2	6.19E-2	7.26E-2	6.09E-2	**2.85E-2**
	6	Mean	9.52E-1 —	9.30E-1 —	9.57E-1 —	9.63E-1 —	9.70E-1 —	9.21E-1 —	9.41E-1 —	**9.89E-1**
		Std	5.95E-2	4.89E-2	6.66E-2	4.33E-2	5.33E-3	2.74E-2	4.14E-2	**3.31E-2**
	8	Mean	9.68E-1 —	9.41E-1 —	9.76E-1 —	9.79E-1 —	9.68E-1 —	9.31E-1 —	9.60E-1 —	**9.95E-1**
		Std	4.38E-2	9.50E-3	4.45E-2	3.13E-3	3.18E-2	3.19E-2	3.19E-2	**2.58E-3**
	10	Mean	9.66E-1 —	9.32E-1 —	9.85E-1 —	9.80E-1 —	9.68E-1 —	9.45E-1 —	9.62E-1 —	**9.88E-1**
		Std	5.27E-2	9.35E-3	3.22E-2	3.42E-3	4.36E-2	1.66E-2	3.16E-2	**4.19E-3**
WFG3	4	Mean	7.85E-1 —	7.71E-1 —	**8.06E-1 +**	7.68E-1 —	7.82E-1 —	1.03E-1 —	6.99E-1 —	8.05E-1
		Std	6.04E-3	6.02E-3	**1.95E-3**	7.37E-3	4.59E-3	4.50E-3	1.32E-2	3.64E-3
	6	Mean	7.77E-1 —	7.47E-1 —	7.97E-1 —	7.45E-1 —	7.73E-1 —	2.59E-1 —	6.47E-1 —	**8.01E-1**
		Std	5.93E-3	8.23E-3	2.47E-3	9.72E-3	5.92E-3	1.88E-1	7.90E-3	**4.51E-3**
	8	Mean	7.24E-1 —	6.85E-1 —	**7.73E-1 +**	7.17E-1 —	7.50E-1 —	5.48E-1 —	5.63E-1 —	7.61E-1
		Std	5.78E-3	8.38E-3	**4.73E-3**	1.19E-2	8.31E-3	1.18E-1	3.63E-3	4.75E-3
	10	Mean	7.69E-1 —	7.08E-1 —	**7.92E-1 +**	7.45E-1 —	7.84E-1 +	5.56E-1 —	6.18E-1 —	7.71E-1
		Std	1.30E-2	8.34E-3	**5.22E-3**	1.20E-2	5.92E-3	1.39E-1	3.07E-3	5.00E-3
WFG4	4	Mean	6.72E-1 —	6.71E-1 —	6.74E-1 —	6.49E-1 —	6.62E-1 —	5.15E-1 —	5.19E-1 —	**6.91E-1**
		Std	6.14E-3	4.78E-3	2.88E-3	4.26E-3	4.70E-3	1.39E-2	2.08E-2	**5.61E-3**
	6	Mean	7.93E-1 —	7.98E-1 —	7.57E-1 —	7.55E-1 —	7.69E-1 —	5.01E-1 —	5.14E-1 —	**8.40E-1**
		Std	9.09E-3	9.18E-3	5.45E-3	6.48E-3	6.65E-3	2.38E-2	2.31E-2	**8.22E-3**
	8	Mean	8.45E-1 —	8.29E-1 —	7.65E-1 —	7.97E-1 —	8.15E-1 —	5.02E-1 —	4.86E-1 —	**9.05E-1**
		Std	9.18E-3	1.13E-2	8.54E-3	1.13E-2	7.16E-3	4.40E-2	1.77E-2	**1.31E-2**
	10	Mean	**8.54E-1 +**	7.23E-1 —	7.46E-1 —	8.00E-1 —	8.42E-1 +	4.74E-1 —	4.14E-1 —	8.17E-1
		Std	**1.18E-2**	2.75E-2	9.19E-3	1.43E-2	7.92E-3	8.32E-2	2.83E-2	7.25E-2
WFG5	4	Mean	6.44E-1 —	6.31E-1 —	6.31E-1 —	6.14E-1 —	6.39E-1 —	5.04E-1 —	5.18E-1 —	**6.49E-1**
		Std	5.50E-3	5.54E-3	3.25E-3	3.75E-3	3.01E-3	1.05E-2	1.57E-2	**4.67E-3**
	6	Mean	7.78E-1 —	7.52E-1 —	7.21E-1 —	7.24E-1 —	7.51E-1 —	4.80E-1 —	5.31E-1 —	**7.96E-1**
		Std	7.91E-3	5.91E-3	4.09E-3	6.57E-3	5.32E-3	2.01E-2	2.65E-2	**8.77E-3**
	8	Mean	8.33E-1 —	7.68E-1 —	7.31E-1 —	7.65E-1 —	7.88E-1 —	4.36E-1 —	5.30E-1 —	**8.50E-1**
		Std	7.83E-3	1.12E-2	6.44E-3	7.35E-3	6.58E-3	1.88E-2	2.08E-2	**6.95E-3**
	10	Mean	8.44E-1 —	6.63E-1 —	7.03E-1 —	7.72E-1 —	8.14E-1 —	3.57E-1 —	4.71E-1 —	**8.52E-1**
		Std	4.04E-2	1.64E-2	6.65E-3	1.07E-2	6.07E-3	3.21E-2	2.58E-2	**7.47E-3**
WFG6	4	Mean	6.45E-1 —	6.41E-1 —	6.40E-1 —	6.16E-1 —	6.43E-1 —	5.27E-1 —	5.29E-1 —	**6.51E-1**
		Std	6.72E-3	8.97E-3	5.41E-3	9.02E-3	5.68E-3	2.21E-2	3.13E-2	**1.02E-2**
	6	Mean	7.85E-1 —	7.72E-1	7.25E-1 —	7.18E-1 —	7.58E-1 —	4.43E-1 —	5.42E-1 —	**7.95E-1**
		Std	7.60E-3	1.02E-2	7.50E-3	1.16E-2	8.61E-3	2.79E-2	4.13E-2	**1.08E-2**
	8	Mean	8.47E-1 —	8.18E-1 —	7.34E-1 —	7.54E-1 —	8.05E-1 —	3.64E-1 —	5.08E-1 —	**8.54E-1**
		Std	9.60E-3	1.30E-2	7.81E-3	1.07E-2	9.20E-3	2.38E-2	3.02E-2	**1.03E-2**
	10	Mean	8.77E-1 —	7.47E-1 —	7.03E-1 —	7.59E-1 —	8.54E-1 —	2.84E-1 —	4.30E-1 —	**8.87E-1**
		Std	1.18E-2	2.38E-2	6.65E-3	1.46E-2	6.90E-3	3.76E-2	4.02E-2	**1.38E-2**
WFG7	4	Mean	**6.92E-1 =**	6.84E-1 —	6.91E-1 —	6.64E-1 —	6.90E-1 —	5.20E-1 —	4.87E-1 —	**6.92E-1**
		Std	**4.09E-3**	6.80E-3	2.81E-3	5.92E-3	5.38E-3	1.60E-2	2.92E-2	**5.72E-3**
	6	Mean	8.33E-1 —	8.21E-1 —	7.93E-1 —	7.85E-1 —	8.19E-1 —	5.02E-1 —	4.63E-1 —	**8.45E-1**
		Std	9.01E-3	8.79E-3	4.61E-3	6.57E-3	4.96E-3	2.67E-2	3.08E-2	**6.28E-3**
	8	Mean	8.93E-1 —	8.75E-1 —	8.06E-1 —	8.40E-1 —	8.73E-1 —	4.10E-1 —	4.22E-1 —	**9.20E-1**
		Std	8.25E-3	1.02E-2	7.72E-3	1.44E-2	4.13E-3	3.86E-2	3.66E-2	**8.65E-3**
	10	Mean	9.13E-1 —	8.14E-1 —	7.83E-1 —	8.47E-1 —	9.14E-1 —	2.83E-1 —	3.47E-1 —	**9.42E-1**
		Std	9.12E-3	1.60E-2	8.06E-3	1.50E-2	4.82E-3	4.53E-2	3.59E-2	**1.09E-2**
WFG8	4	Mean	5.85E-1 —	5.79E-1 —	5.83E-1 —	5.52E-1 —	5.70E-1 —	4.70E-1 —	3.49E-1 —	**6.03E-1**
		Std	6.02E-3	6.27E-3	3.74E-3	8.16E-3	6.35E-3	2.14E-2	3.61E-2	**4.52E-3**
	6	Mean	6.92E-1 —	7.01E-1 —	6.25E-1 —	6.43E-1 —	6.47E-1 —	4.01E-1 —	2.70E-1 —	**7.37E-1**
		Std	9.87E-3	1.65E-2	9.09E-3	1.01E-2	1.16E-2	5.69E-2	2.85E-2	**6.72E-3**
	8	Mean	7.38E-1 —	7.53E-1 —	5.88E-1 —	6.90E-1 —	6.76E-1 —	2.91E-1 —	2.32E-1 —	**8.05E-1**
		Std	1.52E-2	2.76E-2	1.45E-2	1.05E-2	1.05E-2	7.95E-2	5.44E-2	**1.25E-2**
	10	Mean	7.65E-1 —	6.71E-1 —	5.28E-1 —	6.87E-1 —	7.30E-1 —	3.53E-1 —	1.07E-1 —	**8.13E-1**
		Std	1.46E-2	4.55E-2	1.78E-2	1.43E-2	1.57E-2	9.65E-2	3.99E-2	**2.00E-2**
WFG9	4	Mean	5.86E-1 +	5.80E-1 =	5.88E-1 +	**5.98E-1 +**	5.78E-1 —	4.89E-1 —	5.55E-1 —	5.80E-1
		Std	1.36E-2	1.70E-2	2.33E-2	**2.27E-2**	3.81E-3	1.93E-2	2.16E-2	4.62E-3
	6	Mean	6.84E-1 —	6.82E-1 —	6.62E-1 —	**7.00E-1 +**	6.74E-1 —	4.34E-1 —	6.29E-1 —	**6.92E-1**
		Std	2.86E-2	1.90E-2	2.26E-2	**2.28E-2**	8.25E-3	2.24E-2	1.27E-2	**6.32E-3**
	8	Mean	**7.32E-1 =**	6.92E-1 —	6.76E-1 —	**7.33E-1 =**	7.11E-1 —	3.98E-1 —	6.49E-1 —	**7.33E-1**
		Std	**2.18E-2**	1.90E-2	2.19E-2	**1.97E-2**	8.23E-3	3.50E-2	1.65E-2	**8.55E-3**
	10	Mean	7.26E-1 —	5.64E-1 —	6.28E-1 —	**7.32E-1** —	7.14E-1 —	3.45E-1 —	6.39E-1 —	**7.36E-1**
		Std	2.14E-2	3.78E-2	2.30E-2	**1.56E-2**	9.31E-3	4.89E-2	2.35E-2	**8.60E-3**

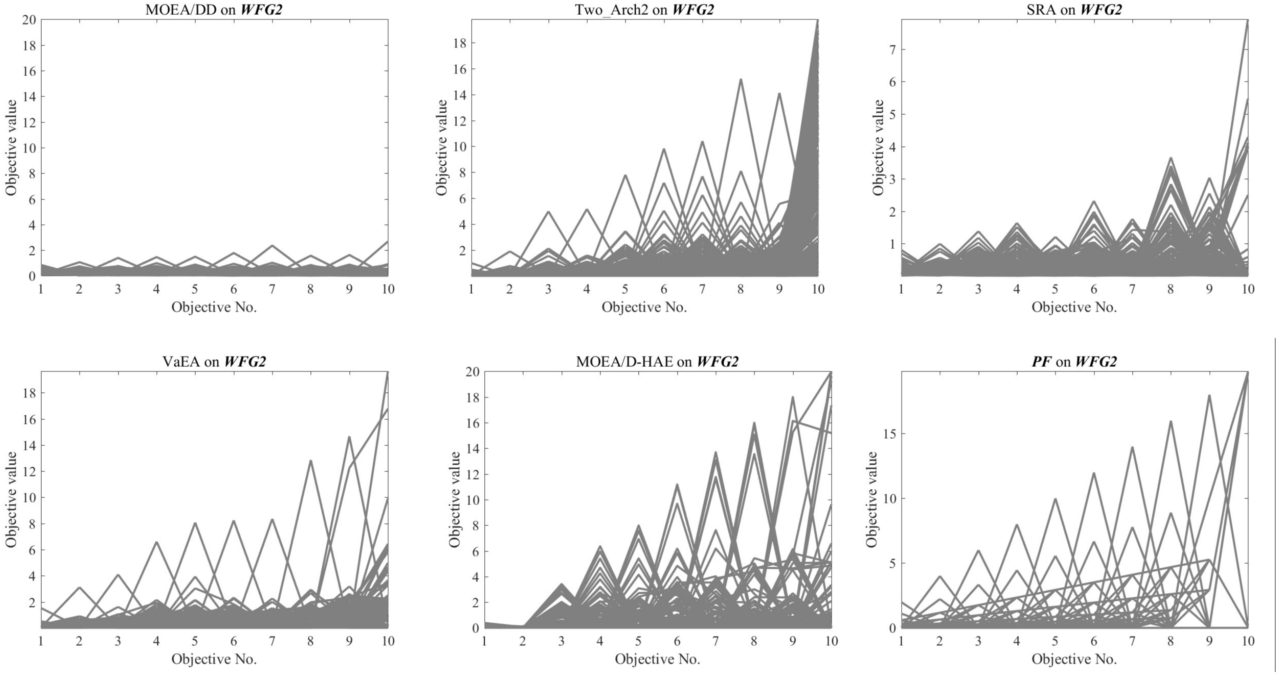
"—", "+", and "=" indicate that the results of the algorithm are worse than, better than, and similar to that of MOEA/D-HAE using Wilcoxon's rank sum test with $\alpha = 0.05$, respectively

(CPU) and 2*16GB DDR4 2666 (RAM).

For each test problem, all the compared algorithms are executed 30 times to get the mean value (Mean) and standard deviation (Std) regarding HV for performance comparisons. Please note that the best result in each test problem is marked with gray background and bold font in the comparison tables. Moreover, to make sure the statistical significance of the differences between the results obtained by MOEA/D-HAE and those obtained by the compared algorithms, Wilcoxon's rank sum test is run with a significance level α=0.05, as suggested in [33, 37, 52]. The symbols "−", "+", and "=" in each comparison table indicate that the result of the algorithm is worse than, better than, and similar to that of MOEA/D-HAE, respectively. Please note that when the best result is obtained by MOEA/D-HAE and another compared can get the statistically similar result, all their results are marked by gray background and bold font in the comparison table.

Table 2 gives all the HV comparison results on ***WFG*** test problems. As observed from Table 2, it is clear that MOEA/D-HAE outperforms other compared algorithms on most ***WFG*** test problems. On ***WFG1***, we observe that the obtained solutions in early evolutionary stage of MOEA/D-HAE are with a good diversity, but this also causes the poor convergence. As ***WFG1*** is a test problem that demands strong convergence, while MOEA/D-HAE prefers to maintain the diversity due to the use of AD (Algorithm 1), its superiority for solving ***WFG1*** is not clear. Even so, MOEA/D-HAE is still competitive on ***WFG1***. For ***WFG1*** with 4 objectives, MOEA/D-HAE performs worse than Two_Arch2, similarly to MaOEA-CSS, and better than the rest competitors. The result of MOEA/D-HAE is only worse than that of Two_Arch2 and better than those of the rest competitors on ***WFG1*** with 6 objectives. To ***WFG1*** with 8 objectives, MOEA/D-HAE performs worse than the three algorithms and is better than the other four competitors. When the number of objectives is increased to 10 objectives for ***WFG1***, Two-Arch2 and MaOEA-CSS have better performances than MOEA/D-HAE. Thus, it is reasonable to summarize that MOEA/D-HAE shows a median performance on ***WFG1*** as this test problem demands a strong convergence while the diversity is considered first in MOEA/D-HAE. Regarding ***WFG2*** with all the objectives, MOEA/D-HAE shows the best performance. On ***WFG3*** with 4, 8 and 10 objectives, MOEA/D-HAE is worse than Two_Arch2 and is better than Two_Arch2 on the 6 objectives, and outperforms other competitors. For the rest test problems (from ***WFG4*** to ***WFG9***, except for ***WFG4*** with 10 objectives and ***WFG9*** with 4 objectives), it is obvious that MOEA/D-HAE gets the best performance comprehensively. Those comparison results demonstrate that MOEA/D-HAE has the superior performance on solving most of the ***WFG*** test problems.

Moreover, some final solution sets of MOEA/D-HAE and the competitors (*i.e.*, MOEA/DD, Two_Arch2, SRA and VaEA) are provided to observe the final convergence. In Figs. 7-10, the final solution sets and the true ***PF***s on four representative test problems (***WFG2***, ***WFG4***, ***WFG6*** and ***WFG8*** with 10 objectives) are plotted to show the convergence. From these figures, it is observed that the final solutions of MOEA/D-HAE can well approximate to the true ***PF***s for all these test problems, while other four algorithms cannot well converge to the ***PF*** of ***WFG2***. Moreover, MOEA/DD and SRA also cannot well converge to the ***PF***s of ***WFG4***, ***WFG6*** and ***WFG8*** by observing their differences to the true ***PF***s.



MOEA/DD on ***WFG2***

Two_Arch2 on ***WFG2***

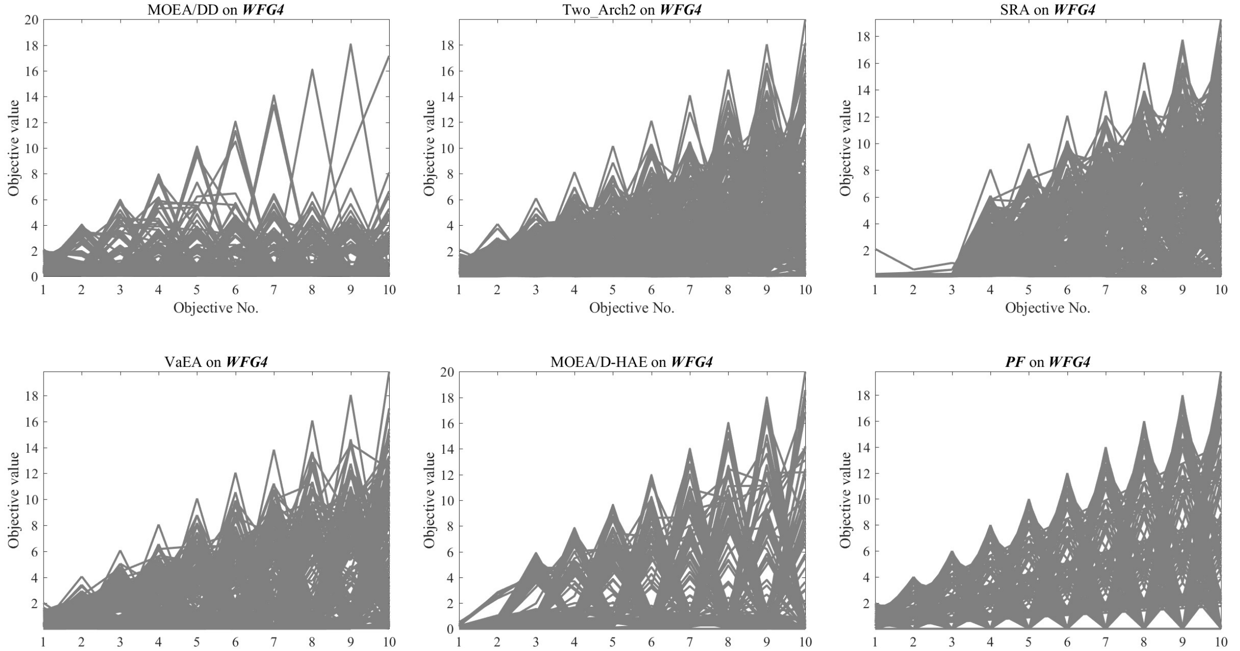
SRA on ***WFG2***

VaEA on ***WFG2***

MOEA/D-HAE on ***WFG2***

PF on ***WFG2***

Fig. 7 The final solution sets of all the tested algorithms and ***PF*** on ***WFG2*** with 10 objectives



MOEA/DD on ***WFG4***

Two_Arch2 on ***WFG4***

SRA on ***WFG4***

VaEA on ***WFG4***

MOEA/D-HAE on ***WFG4***

PF on ***WFG4***

Fig. 8 The final solution sets of all the tested algorithms and ***PF*** on ***WFG4*** with 10 objectives

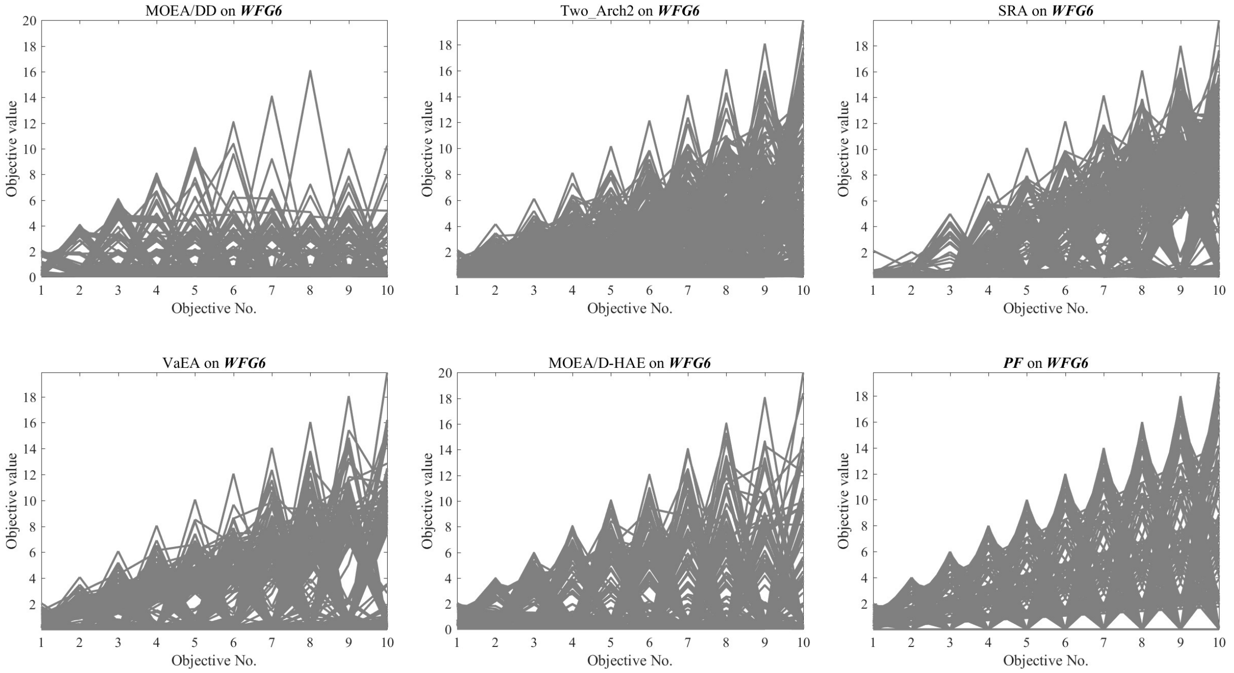


Fig. 9 The final solution sets of all the tested algorithms and ***PF*** on ***WFG6*** with 10 objectives

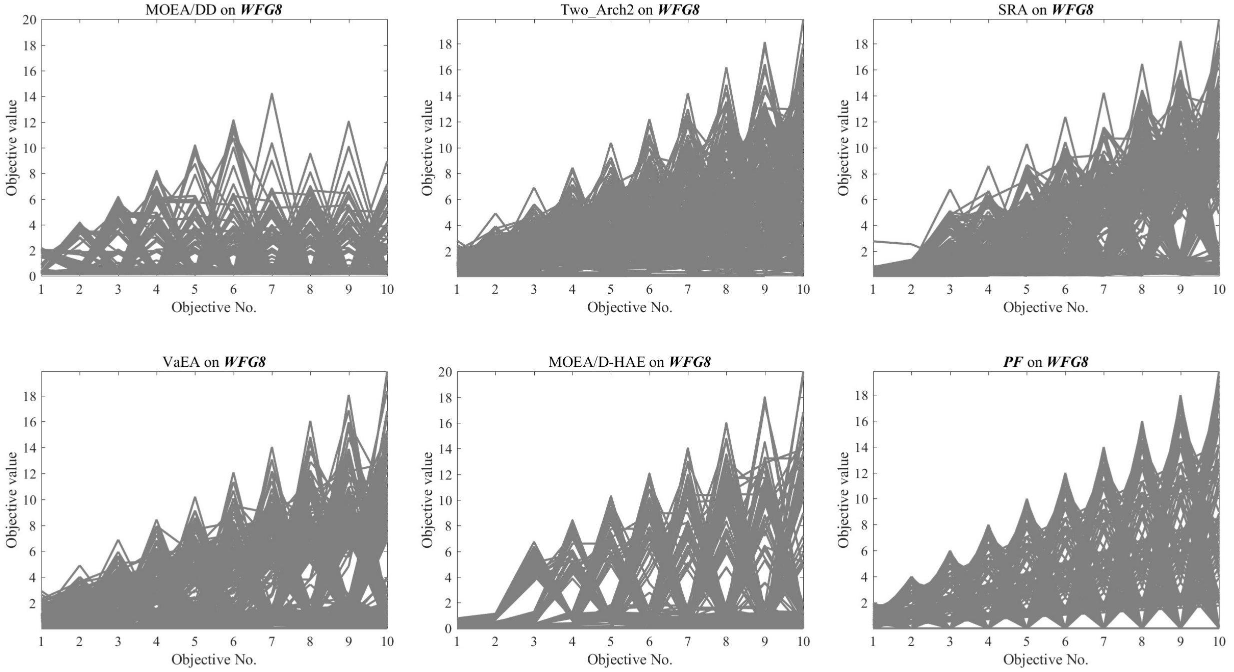


Fig. 10 The final solution sets of all the tested algorithms and ***PF*** on ***WFG8*** with 10 objectives

Table 3 gives all the HV comparison results on ***DTLZ*** test problems. As observed from the HV results in Table 3, MOEA/D-HAE performs very competitively on tackling ***DTLZ1-DTLZ4***, as it performs best on most cases. Only for ***DTZL3*** with 6 objectives, MOEA/D-HAE performs worse than MOEA/DD, NSGA-III, and SRA. Regarding the rest test problems of ***DTLZ1-DTLZ4***, MOEA/D-HAE always obtains the best results. For all the test problems, MOEA/DD and NSGA-III are main competitors to

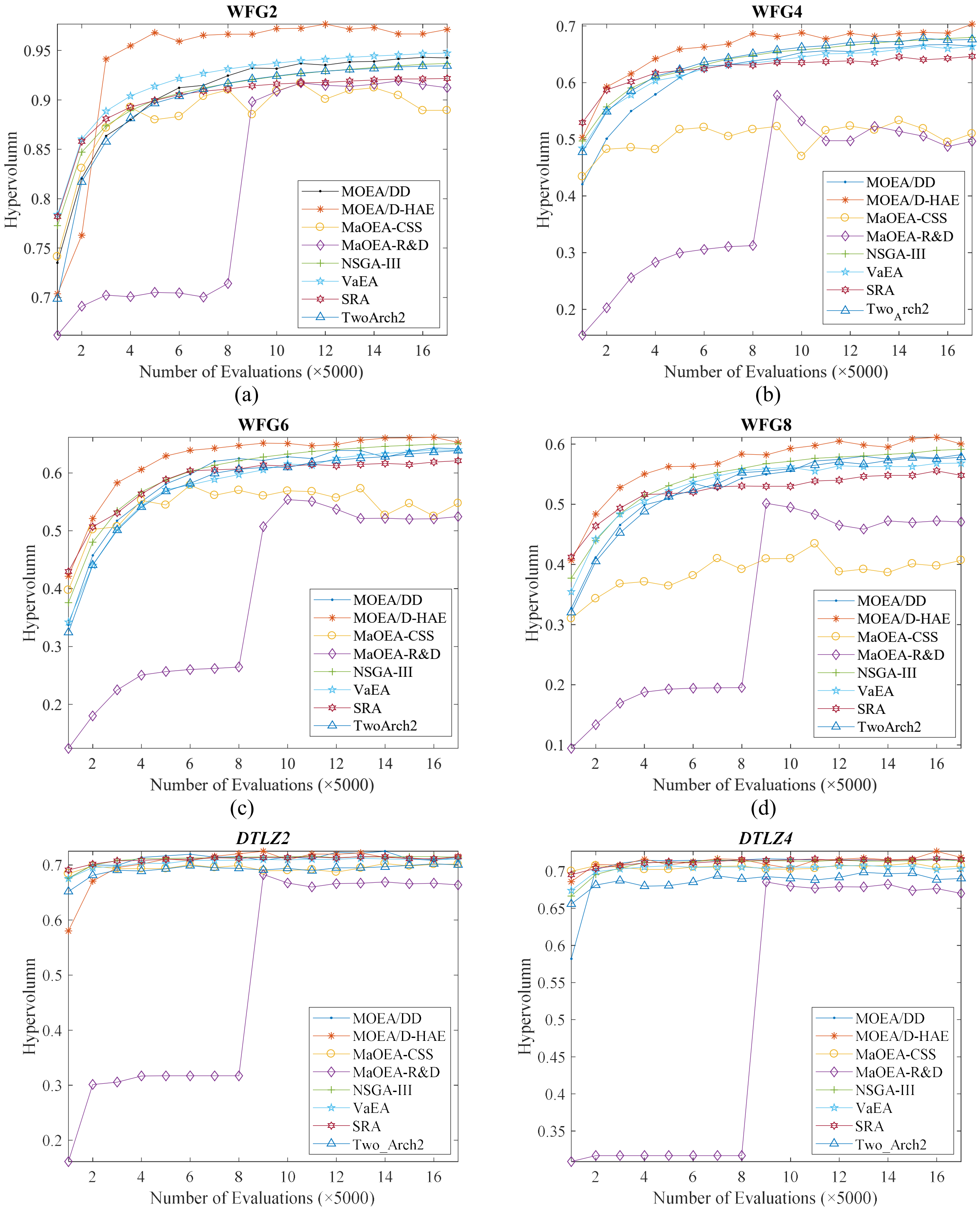
MOEA/D-HAE. Seeing ***DTLZ2-DTLZ4***, the performance of NSGA-III is worse than that of MOEA/DD and MOEA/D-HAE. On ***DLTZ1***, MOEA/D-HAE performs similarly to MOEA/DD and NSGA-III on 4, 6, 8 objectives and better than them on 10 objectives. Clearly, MOEA/D-HAE outperforms NSGA-III and MOEA/DD on ***DTLZ2*** and ***DTLZ4*** with 10 objectives. SRA only gets the best results on the three test problems. Two_Arch2, VaEA, MaOEA-R&D and MaOEA-CSS are not good at tackling ***DTLZ1-DTLZ4*** in our experiments. As pointed out in [17], the uniform weight vectors in these decomposition-based MOEAs (like MOEA/DD) can solve ***DTLZ1-DTLZ4*** quite well, as the used weight vectors perfectly match the shapes of ***PF***s for ***DTLZ1-DTLZ4***. Moreover, the Wilcoxon's rank sum test shows that MOEA/D-HAE and MOEA/DD perform similarly on most ***DTLZ1-DTLZ4*** with 4, 6, 8, and 10 objectives.

Table 3
The HV comparison results of MOEA/D-HAE and seven competitors on the ***DTLZ1- DTLZ4*** test problems

Problem	m		NSGA-III	MOEA/DD	Two_Arch2	SRA	VaEA	MaOEA-R&D	MaOEA-CSS	MOEA/D-HAE
DTLZ1	4	Mean	**9.44E-1 =**	**9.44E-1 =**	9.41E-1 —	9.38E-1 —	8.74E-1 —	9.35E-1 —	9.02E-1 —	**9.44E-1**
		Std	**5.26E-3**	**7.02E-3**	1.06E-3	1.35E-3	4.79E-2	1.76E-3	8.28E-3	**4.40E-3**
	6	Mean	**9.92E-1 =**	**9.93E-1 =**	9.89E-1 —	9.91E-1 —	9.01E-1 —	9.58E-1 —	9.57E-1 —	**9.93E-1**
		Std	**4.48E-3**	**3.68E-3**	5.77E-4	5.77E-4	5.18E-2	7.20E-3	5.84E-3	**4.29E-3**
	8	Mean	**9.99E-1 =**	**9.99E-1 =**	9.96E-1 —	9.98E-1 —	9.33E-1 —	8.89E-1 —	9.76E-1 —	**9.99E-1**
		Std	**3.44E-3**	**1.02E-3**	3.31E-4	1.79E-4	3.94E-2	5.62E-2	3.72E-3	**3.06E-3**
	10	Mean	9.99E-1 —	9.99E-1 —	9.96E-1 —	9.99E-1 —	9.48E-1 —	7.66E-1 —	9.79E-1 —	**1.00E+0**
		Std	2.59E-3	2.55E-3	3.48E-4	1.90E-4	7.52E-2	1.14E-1	3.43E-3	**2.63E-3**
DTLZ2	4	Mean	7.13E-1 —	**7.15E-1 =**	6.97E-1 —	**7.15E-1 =**	7.09E-1 —	6.64E-1 —	7.02E-1 —	**7.16E-1**
		Std	4.90E-3	**4.25E-3**	2.34E-3	**1.33E-3**	1.75E-3	6.10E-3	3.74E-3	**5.68E-3**
	6	Mean	**8.73E-1 =**	**8.73E-1 =**	8.11E-1 —	8.71E-1 —	8.58E-1 —	7.68E-1 —	8.55E-1 —	**8.74E-1**
		Std	**5.87E-3**	**5.35E-3**	8.57E-3	1.27E-3	2.78E-3	1.33E-2	3.87E-3	**7.22E-3**
	8	Mean	**9.44E-1 =**	**9.45E-1 =**	8.42E-1 —	9.35E-1 —	9.26E-1 —	7.56E-1 —	9.16E-1 —	**9.45E-1**
		Std	**1.27E-2**	**8.81E-3**	1.20E-2	1.48E-3	2.03E-3	1.88E-2	3.50E-3	**8.26E-3**
	10	Mean	9.51E-1 —	**9.72E-1** —	7.94E-1 —	9.51E-1 —	9.51E-1 —	6.53E-1 —	9.33E-1 —	**9.74E-1**
		Std	3.57E-2	**6.86E-3**	1.57E-2	1.82E-3	2.22E-3	4.34E-2	4.27E-3	**6.62E-3**
DTLZ3	4	Mean	7.08E-1 —	**7.14E-1 =**	6.90E-1 —	**7.12E-1** —	6.47E-1 —	6.84E-1 —	6.92E-1 —	**7.14E-1**
		Std	1.61E-2	**9.38E-4**	1.12E-2	**3.87E-3**	5.01E-2	6.90E-3	6.49E-3	**4.27E-3**
	6	Mean	8.70E-1 +	**8.75E-1 +**	8.09E-1 —	8.70E-1 +	6.67E-1 —	7.94E-1 —	8.38E-1 —	8.64E-1
		Std	5.18E-3	**5.93E-3**	1.36E-2	2.96E-3	7.54E-2	1.08E-2	9.34E-3	7.33E-3
	8	Mean	**9.42E-1** —	**9.44E-1 =**	8.70E-1 —	9.36E-1 —	7.58E-1 —	7.06E-1 —	9.10E-1 —	**9.44E-1**
		Std	**6.90E-3**	**8.02E-3**	1.05E-2	2.38E-3	4.62E-2	5.56E-2	7.92E-3	**1.38E-2**
	10	Mean	9.13E-1 —	**9.69E-1 =**	8.60E-1 —	9.53E-1 —	8.08E-1 —	4.49E-1 —	9.30E-1 —	**9.69E-1**
		Std	6.40E-2	**5.20E-3**	1.15E-2	2.58E-3	1.57E-1	1.55E-1	5.87E-3	**5.98E-3**
DTLZ4	4	Mean	7.13E-1 —	7.14E-1 —	6.94E-1 —	**7.17E-1 =**	7.07E-1 —	6.80E-1 —	7.09E-1 —	**7.17E-1**
		Std	3.42E-3	4.08E-3	3.00E-3	**1.05E-3**	2.12E-3	3.48E-3	2.35E-3	**3.94E-3**
	6	Mean	8.75E-1 —	8.75E-1 —	7.90E-1 —	**8.77E-1 =**	8.57E-1 —	8.15E-1 —	8.66E-1 —	**8.77E-1**
		Std	6.08E-3	5.96E-3	1.20E-2	**1.02E-3**	2.27E-3	1.31E-2	2.41E-3	**5.55E-3**
	8	Mean	**9.46E-1 =**	**9.46E-1 =**	8.09E-1 —	9.44E-1 —	9.25E-1 —	8.44E-1 —	9.33E-1 —	**9.46E-1**
		Std	**7.15E-3**	**7.05E-3**	6.47E-3	9.98E-4	2.51E-3	1.74E-2	2.14E-3	**8.00E-3**
	10	Mean	9.60E-1 —	9.68E-1 —	7.89E-1 —	9.64E-1 —	9.51E-1 —	7.58E-1 —	9.54E-1 —	**9.72E-1**
		Std	2.39E-2	7.94E-3	1.47E-2	1.09E-3	2.92E-3	4.00E-2	3.14E-3	**7.34E-3**

"—", "+", and "=" indicate that the results of the algorithm are worse than, better than, and similar to that of MOEA/D-HAE using Wilcoxon's rank sum test with $\alpha = 0.05$, respectively

In Fig. 11, the HV convergence processes of all the compared algorithms are plotted on solving each of ***WFG2***, ***WFG4***, ***WFG6***, ***WFG8***, ***DTLZ2*** and ***DTLZ4*** with 4 objectives. As observed from Fig. 11, when the number of evaluations surpasses 15000, MOEA/D-HAE has shown the superiority over other compared algorithms on the ***WFG*** problems. For the ***DTLZ*** cases, most of the compared algorithms perform very well and MOEA/D-HAE still gets the best results at the later stage of evolution. Moreover, all the comparison results based on the HV metric are summarized in Table 4 for all the ***DTLZ*** and ***WFG*** test



(a) (b) (c) (d) (e) (f)

Fig. 11 The HV convergence processes on (a) ***WFG2***, (b) ***WFG4***, (c) ***WFG6***, (d) ***WFG8***, (e) ***DTLZ2*** and (f) ***DTLZ4*** with 4 objectives

problems, in which the terms "*better/similar/worse*" indicate that MOEA/D-HAE respectively performs better than, similarly to, and worse than that of the compared algorithms using Wilcoxon's rank sum test with $\alpha = 0.05$ on the ***WFG*** and ***DTLZ1-DTLZ4***. The terms "*best/all*" in the last row indicates the proportion that the corresponding algorithm achieves the best results among all the compared algorithms when solving all the 52 cases (*i.e.*, 9 ***WFG*** and 4 ***DTLZ*** test problems with 4, 6, 8, and 10 objectives). The summarized results in Table 4 show that NSGA-III and MOEA/DD obtain the very promising performance when solving ***DTLZ1-DTLZ4***. However, they perform poorly on most of ***WFG*** test problems. Obviously, MOEA/D-HAE outperforms other seven recently proposed MaOEAs on most cases and can well solve the ***WFG*** and ***DTLZ1-DTLZ4*** test problems simultaneously. Please note that MOEA/D-HAE gets the best results on 41 out of 52 cases in these comparisons.

Table 4
The comparison summary of MOEA/D-HAE and seven competitors on the ***WFG*** and the ***DTLZ*** test problems

	NSGA-III	MOEA/DD	Two_Arch2	SRA	VaEA	MaOEA-R&D	MaOEA-CSS	MOEA/D-HAE
*better/similar/worse(**WFG**)*	32/2/2	34/1/1	28/0/8	33/1/2	34/0/2	36/0/0	33/1/2	
*better/similar/worse(**DTLZ**)*	9/6/1	5/10/1	16/0/0	14/3/1	16/0/0	16/0/0	16/0/0	
best/all	10/52	12/52	5/52	6/52	0/52	0/52	1/52	41/52

4.4. The Performance of HAE as Decomposition Approach

In this subsection, HAE as a decomposition approach is embedded into the MOEA/D framework [53], which is used to compare with iPBI, LWS, PaP and PaS in tackling the ***WFG*** and ***DTLZ1-DTLZ4*** test problems. It is noted that all the compared algorithms with different decomposition approaches are implemented under the same MOEA/D framework in order to have a fair comparison. The performance indicator in Section 4.2 and the setting for population sizes in Table 1 are used. Other parameters in the MOEA/D framework are clarified in Table 5, where *T* means the neighbors size, *delta* is the probability to select neighbors and *nr* is the number of the maximal updating. SBX and PM operators are used and their details are shown in Section 4.3. Due to the difference of information demanded for updating, HAE and LWS will do the population update procedure after combing the parents with their offspring, while iPBI, PaP and PaS will update the population once an offspring is generated. The nadir point of the iPBI approach is updated when an offspring replaces one parent, in which each element of the nadir point is always the maximal value of each objective of current population.

Table 5
Parameters settings in the MOEA/D framework

T	*delta*	*nr*
20	0.9	2

In Table 6, the HV experimental results of five decomposition approaches and HAE on ***WFG*** test problems are provided. It is obvious that HAE performs best in most of test problems except for ***WFG1***. Comparing LWS with HAE, it can be found that LWS is worse than HAE obviously, especially on ***WFG1***, and LWS hardly deals with ***WFG1*** in our experiment. LWS gets the HV values under an accuracy level of

Table 6

The HV comparison results of HAE and four decomposition methods on all the *WFG* test problems

Problem	m		LWS	PaS	PaP	iPBI	HAE
WFG1	4	Mean	1.21E-1 —	8.49E-1 +	**8.66E-1 +**	2.89E-1 —	7.01E-1
		Std	3.11E-2	2.77E-2	**3.17E-2**	2.87E-2	2.94E-2
	6	Mean	1.75E-2 —	8.74E-1 +	**9.05E-1 +**	1.43E-1 —	6.15E-1
		Std	3.43E-2	2.33E-2	**2.13E-2**	4.47E-2	5.28E-2
	8	Mean	0.00E+0 —	9.34E-1 +	**9.40E-1 +**	5.06E-2 —	5.33E-1
		Std	0.00E+0	9.65E-3	**5.61E-3**	3.89E-2	8.51E-2
	10	Mean	2.51E-3 —	9.15E-1 +	**9.44E-1 +**	3.68E-2 —	7.11E-1
		Std	4.28E-3	3.91E-2	**1.60E-2**	2.97E-2	8.28E-2
WFG2	4	Mean	8.27E-1 —	8.66E-1 —	8.70E-1 —	6.22E-1 —	**9.72E-1**
		Std	7.04E-2	7.35E-2	7.93E-2	4.04E-2	**2.35E-3**
	6	Mean	8.17E-1 —	8.80E-1 —	9.16E-1 —	5.67E-1 —	**9.96E-1**
		Std	8.53E-2	1.24E-1	8.83E-2	6.45E-2	**2.87E-3**
	8	Mean	7.91E-1 —	8.97E-1 —	9.19E-1 —	3.32E-1 —	**9.95E-1**
		Std	9.72E-2 —	1.15E-1	8.88E-2	3.92E-2	**1.77E-3**
	10	Mean	8.04E-1 —	8.53E-1 —	9.36E-1 —	3.56E-1 —	**9.91E-1**
		Std	9.94E-2	1.53E-1	8.43E-2	3.79E-2	**3.56E-3**
WFG3	4	Mean	7.22E-1 —	7.11E-1 —	7.71E-1 —	6.50E-1 —	**8.04E-1**
		Std	1.15E-2	7.21E-3	1.97E-3	1.86E-2	**2.89E-3**
	6	Mean	7.02E-1 —	7.44E-1 —	7.96E-1 —	5.97E-1 —	**8.02E-1**
		Std	8.99E-3	1.26E-2	4.43E-3	1.60E-2	**5.12E-3**
	8	Mean	6.21E-1 —	7.21E-1 —	7.86E-1 —	4.77E-1 —	**7.62E-1**
		Std	1.19E-2	9.10E-3	6.17E-3	4.81E-3	**4.91E-3**
	10	Mean	6.16E-1 —	7.57E-1 —	7.87E-1 —	4.84E-1 —	**7.66E-1**
		Std	9.94E-3	6.58E-3	7.92E-3	1.87E-2	**4.32E-3**
WFG4	4	Mean	5.41E-1 —	3.32E-1 —	3.20E-1 —	2.46E-1 —	**6.91E-1**
		Std	5.26E-3	3.27E-2	1.09E-2	2.06E-2	**2.01E-3**
	6	Mean	4.39E-1 —	4.64E-1 —	4.46E-1 —	1.48E-1 —	**8.41E-1**
		Std	5.85E-3	6.70E-2	2.10E-2	5.03E-3	**6.35E-3**
	8	Mean	3.70E-1 —	5.15E-1 —	4.87E-1 —	1.27E-1 —	**9.01E-1**
		Std	7.67E-3	6.58E-2	5.54E-2	2.25E-3	**1.81E-2**
	10	Mean	2.93E-1 —	5.57E-1 —	3.93E-1 —	1.05E-1 —	**8.04E-1**
		Std	5.58E-3	1.11E-1	7.31E-2	3.44E-3	**5.66E-2**
WFG5	4	Mean	5.23E-1 —	2.76E-1 —	2.70E-1 —	2.25E-1 —	**6.47E-1**
		Std	4.13E-3	1.88E-2	1.48E-5	1.61E-2	**2.10E-3**
	6	Mean	4.23E-1 —	4.00E-1 —	3.83E-1 —	1.43E-1 —	**7.91E-1**
		Std	6.71E-3	4.21E-2	7.58E-3	3.11E-3	**6.18E-3**
	8	Mean	3.51E-1 —	5.07E-1 —	3.67E-1 —	1.24E-1 —	**8.52E-1**
		Std	4.31E-3	5.72E-2	3.48E-2	1.99E-3	**6.16E-3**
	10	Mean	2.73E-1 —	5.77E-1 —	2.83E-1 —	1.03E-1 —	**8.54E-1**
		Std	2.83E-3	7.49E-2	2.75E-2	4.01E-3	**8.53E-3**
WFG6	4	Mean	5.40E-1 —	2.82E-1 —	2.74E-1 —	2.57E-1 —	**6.49E-1**
		Std	1.20E-2	1.47E-2	1.11E-2	2.85E-2	**7.60E-3**
	6	Mean	4.23E-1 —	3.89E-1 —	3.90E-1 —	1.44E-1 —	**7.89E-1**
		Std	1.02E-2	2.93E-2	1.05E-2	4.14E-3	**1.12E-2**
	8	Mean	3.52E-1 —	4.87E-1 —	3.34E-1 —	1.25E-1 —	**8.52E-1**
		Std	7.38E-3	5.63E-2	1.28E-2	3.03E-3	**1.11E-2**
	10	Mean	2.75E-1 —	5.37E-1 —	2.92E-1 —	1.01E-1 —	**8.85E-1**
		Std	9.53E-3	7.56E-2	3.81E-2	5.52E-3	**1.25E-2**
WFG7	4	Mean	5.44E-1 —	3.10E-1 —	3.17E-1 —	1.96E-1 —	**6.92E-1**
		Std	7.08E-3	2.06E-2	1.13E-4	1.70E-2	**2.05E-3**
	6	Mean	4.31E-1 —	4.23E-1 —	4.19E-1 —	1.43E-1 —	**8.44E-1**
		Std	1.00E-2	4.78E-2	2.50E-2	2.21E-3	**5.86E-3**
	8	Mean	3.68E-1 —	5.18E-1 —	3.81E-1 —	1.24E-1 —	**9.20E-1**
		Std	7.21E-3	9.59E-2	4.37E-3	2.65E-3	**7.41E-3**
	10	Mean	2.82E-1 —	5.58E-1 —	3.20E-1 —	1.01E-1 —	**9.38E-1**
		Std	1.27E-2	1.06E-1	1.46E-2	3.57E-3	**1.06E-2**
WFG8	4	Mean	4.97E-1 —	2.22E-1 —	2.69E-1 —	2.78E-1 —	**6.03E-1**
		Std	1.28E-2	8.54E-3	4.10E-2	1.23E-2	**2.20E-3**
	6	Mean	2.99E-1 —	3.04E-1 —	3.14E-1 —	1.74E-1 —	**7.41E-1**
		Std	3.29E-2	2.66E-2	2.68E-2	9.45E-3	**7.62E-3**
	8	Mean	2.23E-1 —	3.70E-1 —	3.39E-1 —	1.25E-1 —	**8.08E-1**
		Std	3.74E-2	3.30E-2	8.70E-3	2.97E-3	**1.23E-2**
	10	Mean	1.45E-1 —	4.24E-1 —	3.12E-1 —	1.04E-1 —	**8.14E-1**
		Std	6.76E-2	4.64E-2	1.63E-2	4.32E-3	**2.26E-2**
WFG9	4	Mean	5.14E-1 —	2.25E-1 —	2.61E-1 —	2.37E-1 —	**5.79E-1**
		Std	1.65E-2	7.12E-3	3.08E-2	2.75E-2	**1.27E-3**
	6	Mean	4.17E-1 —	3.27E-1 —	3.51E-1 —	1.61E-1 —	**6.94E-1**
		Std	1.11E-2	2.95E-2	4.09E-2	1.16E-2	**7.34E-3**
	8	Mean	3.55E-1 —	3.90E-1 —	4.29E-1 —	1.23E-1 —	**7.31E-1**
		Std	1.37E-2	4.30E-2	4.77E-2	6.16E-3	**9.00E-3**
	10	Mean	2.83E-1 —	4.07E-1 —	4.30E-1 —	1.03E-1 —	**7.31E-1**
		Std	2.21E-2	5.45E-2	6.39E-2	1.37E-2	**1.41E-2**

"—", "+", and "=" indicate that the results of the algorithm are worse than, better than, and similar to that of MOEA/D-HAE using Wilcoxon's rank sum test with $\alpha = 0.05$, respectively

10^{-2} on ***WFG1*** with 6 objectives, and cannot even find any solution that can dominate the reference point for HV on ***WFG1*** with 8 objectives. Considering PaS, its performance is better than HAE on ***WFG1***, but it is worse than HAE apparently on the other test problems. PaP adopts the similar method as PaS, which adjusts its parameters to find suitable solutions, thus PaS and PaP perform comparatively on all the test problems. As to iPBI, it is universally worse than HAE except ***WFG1***. The main issue of iPBI is how to set the nadir point that seriously influences its performance. In Table 2, the results of HAE on ***WFG1*** are not so promising, as ***WFG1*** requires strong convergence for decomposition approach and HAE executes the AD approach considering the diversity first, which makes the shortage of the convergence on ***WFG1***.

Table 7

The HV comparison results of HAE and four decomposition methods on the ***DTLZ1- DTLZ4*** test problems

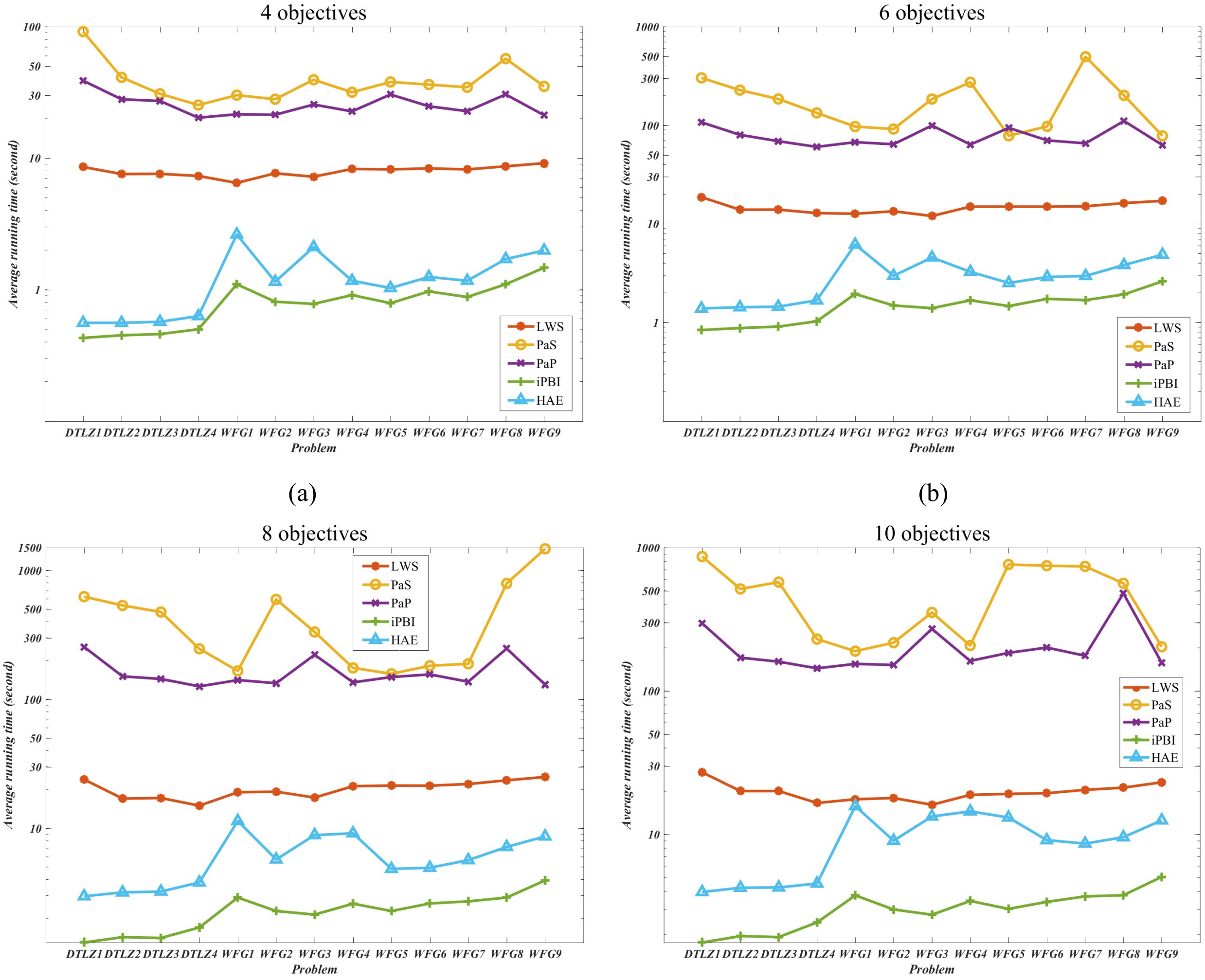
Problem	m		LWS	PaS	PaP	iPBI	HAE
DTLZ1	4	Mean	8.26E-1 —	4.08E-1 —	7.27E-1 —	0.00E+0 —	**9.45E-1**
		Std	6.09E-3	4.99E-2	5.43E-2	0.00E+0	**8.45E-5**
	6	Mean	9.43E-1 —	5.53E-1 —	5.65E-1 —	0.00E+0 —	**9.92E-1**
		Std	2.07E-2	5.21E-2	1.24E-1	0.00E+0	**2.03E-4**
	8	Mean	7.75E-1 —	6.25E-1 —	9.53E-1 —	0.00E+0 —	**9.98E-1**
		Std	1.31E-2	1.42E-1	2.61E-2	0.00E+0	**2.25E-3**
	10	Mean	7.30E-1 —	6.52E-1 —	9.35E-1 —	0.00E+0 —	**1.00E+0**
		Std	1.33E-2	1.91E-1	3.14E-2	0.00E+0	**2.77E-3**
DTLZ2	4	Mean	6.29E-1 —	3.21E-1 —	3.17E-1 —	6.47E-2 —	**7.15E-1**
		Std	7.71E-3	1.49E-2	3.56E-9	1.08E-2	**3.32E-5**
	6	Mean	4.35E-1 —	4.31E-1 —	4.39E-1 —	1.60E-5 —	**8.75E-1**
		Std	3.08E-4	2.10E-2	1.32E-2	7.85E-6	**1.52E-4**
	8	Mean	7.77E-1 —	5.19E-1 —	3.78E-1 —	9.68E-4 —	**9.47E-1**
		Std	1.30E-2	4.39E-2	4.54E-3	9.91E-5	**5.61E-3**
	10	Mean	7.92E-1 —	5.86E-1 —	3.16E-1 —	9.99E-4 —	**9.71E-1**
		Std	2.13E-2	7.61E-2	3.73E-3	3.84E-4	**6.75E-3**
DTLZ3	4	Mean	6.21E-1 —	3.12E-1 —	3.17E-1 —	0.00E+0 —	**7.13E-1**
		Std	7.10E-3	2.71E-2	6.79E-4	0.00E+0	**7.60E-4**
	6	Mean	4.36E-1 —	4.08E-1 —	4.21E-1 —	0.00E+0 —	**8.72E-1**
		Std	3.28E-3	7.42E-2	4.39E-2	0.00E+0	**9.63E-4**
	8	Mean	7.82E-1 —	4.69E-1 —	3.82E-1 —	0.00E+0 —	**9.47E-1**
		Std	1.81E-2	1.31E-1	2.35E-2	0.00E+0	**8.40E-3**
	10	Mean	7.99E-1 —	5.58E-1 —	3.14E-1 —	0.00E+0 —	**9.71E-1**
		Std	2.98E-2	1.34E-1	1.30E-2	0.00E+0	**6.41E-3**
DTLZ4	4	Mean	6.67E-1 —	2.57E-1 —	2.78E-1 —	3.79E-2 —	**7.15E-1**
		Std	4.80E-3	6.87E-2	4.71E-2	2.27E-2	**5.13E-5**
	6	Mean	3.89E-1 —	3.95E-1 —	3.90E-1 —	2.05E-4 —	**8.75E-1**
		Std	4.41E-2	5.94E-2	7.48E-2	7.09E-4	**9.14E-5**
	8	Mean	8.87E-1 —	5.00E-1 —	3.78E-1 —	8.69E-4 —	**9.47E-1**
		Std	1.02E-2	7.17E-2	1.17E-2	2.16E-4	**7.12E-3**
	10	Mean	9.08E-1 —	5.66E-1 —	3.17E-1 —	1.36E-3 —	**9.73E-1**
		Std	8.67E-3	4.94E-2	3.12E-3	5.42E-4	**6.25E-3**

"—", "+", and "=" indicate that the results of the algorithm are worse than, better than, and similar to that of MOEA/D-HAE using Wilcoxon's rank sum test with $\alpha = 0.05$, respectively

The HV comparison results of different decomposition approaches on ***DTLZ1-DTLZ4*** are shown in Table 7. It can be seen that HAE performs best on all the test problems. It is observed that LWS is better than PaP, PaS and iPBI on most of test problems and iPBI hardy finds good solutions to dominate the reference point. By observing the standard deviation values and the mean values of the results, it can be found that the mean values of HAE are large relatively, while its Std values are small with an accuracy level of 10^{-3} on all the test problems. However, LWS, PaS and PaP have relative small mean values and large Std values on some test problems (*e.g.*, , ***DTLZ1*** with 6, 8 and 10 objectives and ***DTLZ3*** with 8 and 10 objectives), this is because they may easily fall into local optimal ***PF***s and cannot cover the ***PF***s well.

Table 8
The comparison summary of HAE and four decomposition methods on the ***WFG*** and the ***DTLZ*** test problems

	LWS	PaS	PaP	iPBI	HAE
better/similar/worse(***WFG***)	36/0/0	32/0/4	32/0/4	36/0/0	
better/similar/worse(***DTLZ***)	16/0/0	16/0/0	16/0/0	16/0/0	
best/all	0/52	0/52	4/52	0/52	48/52



(a) (b)

(c) (d)

Fig. 12 the average running times of LWS, PaS, PaP, iPBI and HAE on ***DTLZ1-DTLZ4*** and ***WFG1-WFG9***

All the comparison results based on the HV metric are summarized in Table 8 for all the ***DTLZ*** and ***WFG*** test problems, in which the terms "*better/similar/worse*" indicate that HAE respectively performs better than, similarly to, and worse than that of the compared decomposition approaches using Wilcoxon's rank sum test with $\alpha = 0.05$ on the ***WFG*** and ***DTLZ1-DTLZ4*** test problems. The terms "*best/all*" in the last row indicates the proportion that the corresponding algorithm achieves the best results among all the compared algorithms when solving all the 52 cases (*i.e.*, 9 ***WFG*** and 4 ***DTLZ*** test problems with 4, 6, 8, and 10 objectives). The summarized results in Table 8 show that HAE obtains the very promising performance and performs better than other decomposition approaches on most cases of all the test problems adopted.

Moreover, the average running times for different decomposition approaches (*i.e.*, LWS, PaS, PaP, iPBI and HAE) are also recorded to analyze their efficiencies on solving all the used test problems. It is noted that all the compared approaches are implemented using Java in a personal computer with Intel i5 8600K@4.7Ghz (CPU) and 2*16GB DDR4 2666 (RAM). Figs. 12(a), (b), (c) and (d) show the average running times of 30 runs for LWS, PaS, PaP, iPBI and HAE, when solving each of ***DTLZ1-DTLZ4*** and ***WFG1-WFG9*** with 4, 6, 8 and 10 objectives, respectively. It can be found that iPBI gives the best results of the average running times on all the cases due to its simplicity, while HAE obtains the 2nd rank on all the cases, as it needs to sequentially run two decomposition approaches. LWS runs more slowly than HAE, as the solutions in LWS may need to calculate their aggregated functions for many times. At last, PaS and PaP show the slowest running speed, as their adopted greedy algorithms are more time consuming.

4.5. Parameter Sensitivity Analysis in HAE

In MOEA/D-HAE, only one parameter needs to be pre-set, *i.e.*, the θ value for the HAE approach. In order to study its effect, different θ values, *i.e.*, $\theta = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$ are examined on all the ***WFG*** test problems with 4 objectives, and the same settings for other parameters in Section 4.3 are used. Table 9 collects all the HV comparison results for MOEA/D-HAE with the different θ values, where "–", "+", and "=" indicate that the result of MOEA/D-HAE with $\theta = 0.1$ is better than, worse than, and similar to that of MOEA/D-HAE with other values of θ using Wilcoxon's rank sum test with $\alpha = 0.05$, respectively.

Table 9
The HV comparison results of HAE with $\theta = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$ on ***WFG*** test problems with 4 objectives

Problem	***PF*** Geometry		$\theta = 0.1$	$\theta = 0.2$	$\theta = 0.3$	$\theta = 0.4$	$\theta = 0.5$	$\theta = 0.6$	$\theta = 0.7$	$\theta = 0.8$	$\theta = 0.9$
WFG1	Convex Mixed	Mean	**7.22E-1**	7.04E-1 —	6.89E-1 —	6.69E-1 —	6.45E-1 —	6.17E-1 —	6.11E-1 —	6.03E-1 —	5.98E-1 —
		Std	**2.89E-2**	3.57E-2	3.30E-2	3.65E-2	4.84E-2	4.86E-2	5.84E-2	5.92E-2	5.88E-2
WFG2	Convex Disconnected	Mean	9.64E-1	**9.71E-1 +**	**9.70E-1 +**	**9.71E-1 +**	**9.71E-1 +**	**9.70E-1 +**	**9.70E-1 +**	**9.70E-1 +**	**9.71E-1 +**
		Std	2.85E-2	**3.62E-3**	**4.07E-3**	**4.13E-3**	**5.43E-3**	**5.23E-3**	**5.52E-3**	**5.18E-3**	**3.75E-3**
WFG3	Linear Degenerate	Mean	**8.05E-1**	8.03E-1 —	8.00E-1 —	8.01E-1 —	8.00E-1 —	8.00E-1 —	7.99E-1 —	8.00E-1 —	7.99E-1 —
		Std	**3.64E-3**	4.58E-3	5.45E-3	4.92E-3	5.11E-3	4.94E-3	6.28E-3	5.47E-3	5.52E-3
WFG4	Concave	Mean	**6.91E-1**	**6.90E-1 =**	**6.91E-1 =**	**6.90E-1 =**	**6.90E-1 =**	**6.91E-1 =**	**6.91E-1 =**	**6.90E-1 =**	**6.90E-1 =**
		Std	**5.61E-3**	**5.03E-3**	**4.62E-3**	**5.42E-3**	**5.99E-3**	**6.20E-3**	**6.12E-3**	**7.50E-3**	**6.71E-3**
WFG5	Concave	Mean	**6.49E-1**	**6.48E-1 =**	**6.48E-1 =**	**6.48E-1 =**	**6.48E-1 =**	**6.48E-1 =**	**6.48E-1 =**	**6.48E-1 =**	**6.49E-1 =**
		Std	**4.67E-3**	**6.29E-3**	**5.63E-3**	**4.19E-3**	**5.62E-3**	**4.67E-3**	**4.87E-3**	**7.27E-3**	**7.68E-3**
WFG6	Concave	Mean	**6.51E-1**	**6.52E-1 =**	**6.51E-1 =**	**6.50E-1 =**	**6.50E-1 =**	**6.52E-1 =**	**6.50E-1 =**	**6.50E-1 =**	**6.50E-1 =**
		Std	**1.02E-2**	**8.00E-3**	**9.77E-3**	**9.82E-3**	**8.95E-3**	**9.75E-3**	**1.00E-2**	**9.59E-3**	**8.50E-3**
WFG7	Concave	Mean	**6.91E-1**	**6.92E-1 =**	**6.92E-1 =**	**6.92E-1 =**	**6.91E-1 =**	**6.92E-1 =**	**6.92E-1 =**	**6.91E-1 =**	**6.92E-1 =**
		Std	**4.34E-3**	**5.03E-3**	**5.66E-3**	**4.14E-3**	**5.05E-3**	**4.51E-3**	**5.93E-3**	**6.39E-3**	**4.56E-3**
WFG8	Concave	Mean	**6.03E-1**	**6.03E-1 =**	**6.03E-1 =**	**6.04E-1 =**	**6.04E-1 =**	**6.04E-1 =**	**6.03E-1 =**	**6.02E-1 =**	**6.03E-1 =**
		Std	**4.52E-3**	**5.15E-3**	**4.54E-3**	**5.79E-3**	**5.63E-3**	**5.33E-3**	**4.50E-3**	**5.08E-3**	**5.46E-3**
WFG9	Concave	Mean	**5.80E-1**	**5.79E-1 =**	**5.79E-1 =**	**5.79E-1 =**	**5.79E-1 =**	**5.79E-1 =**	**5.79E-1 =**	**5.80E-1 =**	**5.80E-1 =**
		Std	**4.62E-3**	**4.25E-3**	**5.01E-3**	**5.08E-3**	**4.66E-3**	**4.58E-3**	**3.75E-3**	**5.19E-3**	**2.87E-3**

"—", "+", and "=" indicate that the results of the parameter are worse than, better than, and similar to that of HAE with $\theta = 0.1$ using Wilcoxon's rank sum test with $\alpha = 0.05$, respectively

In Table 9, the HV comparison results of HAE with different θ values are provided. On ***WFG1*** with convex and mixed ***PF***, the HV values decrease with the increase of the θ value, which shows that the solution of ***WFG1*** with 4 objectives needs strong convergence pressure. On ***WFG2***, the HV value for $\theta = 0.1$ is clearly worse than that with the other θ values. From $\theta = 0.2$ to $\theta = 0.9$, their HV results change slightly, and they are statistically similar to each other by Wilcoxon's rank sum test with $\alpha = 0.05$. MOEA/D-HAE is more effective to find the marginal solutions from the population for covering the whole ***PF*** when $\theta \geq 0.2$. For ***WFG3*** with linear and degenerate geometry, the HV value with $\theta = 0.1$ is better than that with $\theta \geq 0.2$. Observing $\theta = 0.2$ to $\theta = 0.9$ on ***WFG3***, it can be found that the performance is not significantly impacted by the θ value and it indicates that the convergence weights more than the marginal exploitation on ***WFG3***. For the rest cases (in ***WFG4-WFG9***), it can be observed that most of the HV results for MOEA/D-HAE with $\theta = 0.1$ are statistically similar to that of MOEA/D-HAE with $\theta = 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$. The reason may be that the true ***PF***s of ***WFG4-WFG9*** are concave and continuous. In this case, the HAE approach has few contributions to these test problems. The angle-based decomposition approach is able to find the suitable solutions associated to most of subproblems. Therefore, for most of test problems adopted in this paper, the setting of θ would not significantly affect the performance of MOEA/D-HAE.

5. Conclusions and Future Work

In this paper, we have presented the hybrid angle-encouragement decomposition approach, which is embedded into a general framework of MOEAs. The proposed algorithm includes the angle-based decomposition and the EBI decomposition approaches. The former one defines the feasible region for each subproblem and only selects the solutions in this area for association, which helps to maintain the diversity first and also consider the convergence when several solutions are under the feasible region. When the subproblems are not associated to any solution using the angle-based decomposition method, the latter one (EBI) is executed to only consider the convergence and to extend the boundaries of current population, as there has not any solution under the feasible region. Compared with several competitive MaOEAs (*i.e.*, NSGA-III, MOEA/DD, Two_Arch2, SRA, VaEA, MaOEAR&D, and MaOEA-CSS), the experiments confirm the superiority of MOEA/D-HAE when solving different types of MaOPs (*i.e.*, the ***DTLZ1-DTLZ4*** and ***WFG*** test problems with various objectives). Moreover, the advantages of HAE as a hybrid decomposition approach are also confirmed under the MOEA/D framework by comparing with other competitive decomposition approaches designed for MaOPs (i.e., iPBI, LWS, PaP, and PaS).

Our future work will further study the performance of the proposed hybrid decomposition approach in other decomposition-based MOEAs, such as MOEA/D-FRRMAB [65] with multiple differential evolution operators. Furthermore, the extension of MOEA/D-HAE to solve some real-world applications will also be studied.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grants 61806130, 61876110, 61836005 and 61672358, the Joint Funds of the National Natural Science Foundation of China under Key Program Grant U1713212, the Natural Science Foundation of Guangdong Province under grant 2017A030313338, Shenzhen Technology Plan under Grant JCYJ20170817102218122 and JCYJ20170302154032530, and CONACyT under Grant 221551.

Reference

[1] K. Miettinen, Nonlinear Multiobjective Optimization. Boston, Ma, USA: Kluwer Academic, 1999.

[2] O. Abedinia, N. Amjady and M. Naderi, “Multi-objective Environmental/Economic Dispatch using firefly technique,” in Proc. of the 2012 11th International Conference on Environment and Electrical Engineering, Venice, Italy, 2012, pp. 461-466.

[3] O. Abedinia, K. Kiani, H. Shayanfar, N Amjady and A. Ghasemi, “Multiobjective environmental and economic dispatch using imperialist competitive algorithm,” *International Journal for Physical and Engineering Sciences*, vol. 4, no. 2, pp. 63-70, 2012.

[4] A. Ghasemi, O. Abedinia, HA Shayanfar and M. Naderi, “PSO-TVAC Algorithm for Multi Objective PSS Design in Multi-Machine Power System,” in Proc. of the 2011 International Conference on Artificial Intelligence (ICAI’11), Las Vegas, Nevada, USA, 2011.

[5] O. Abedinia, D. Garmarodi, R. Rahbar, and F. Javidzadeh, “Multi-objective environmental/economic dispatch using interactive artificial bee colony algorithm,” *Journal of Basic and Applied Scientific Research*, vol. 2, no. 11, pp. 1272–11281, 2012.

[6] M. Bekravi and O. Abedinia, “A new multi-objective meta heuristic algorithm based on environmental/economic load dispatch with wind effect,” *Technical and Physical Problems of Engineering*, vol. 5, no. 2, pp. 81–89, 2013.

[7] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, 2002.

[8] E. Zitzler and L. Thiele, “Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach,” *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257–271, 1999.

[9] E. Zitzler and S. Künzli, “Indicator-based selection in multiobjective search,” Parallel Problem Solving from Nature - PPSN VIII, Lecture Notes in Computer Science, vol. 3242, pp. 832–842, 2004.

[10] L. Wang, Q. Zhang, A. Zhou, M. Gong, and L. Jiao, “Constrained subproblems in a

decomposition-based multiobjective evolutionary algorithm," *IEEE Trans. Evol. Comput.*, vol. 20, no. 3, pp. 475–480, 2016.

[11] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, 2007.

[12] E. Hughes, "Evolutionary many-objective optimization: many once or one many?" in Proc. of the 2005 IEEE Congress on Evolutionary Computation, Edinburgh, UK, 2005, pp. 222-227.

[13] R. C. Purshouse and P. J. Fleming, "Evolutionary many-objective optimization: an exploratory analysis," in Proc. of the 2003 IEEE Congress on Evolutionary Computation, Canberra, Australia, 2003, pp. 2066-2073.

[14] S. Bandyopadhyay and A. Mukherjee, "An algorithm for many-objective optimization with reduced objective computations: a study in differential evolution," *IEEE Trans. Evol. Comput.*, vol. 19, no. 3, pp. 400–413, 2015.

[15] H. Ishibuchi, N. Akedo, and Y. Nojima, "Behavior of multiobjective evolutionary algorithms on many-objective Knapsack problems," *IEEE Trans. Evol. Comput.*, vol. 19, no. 2, pp. 264–283, 2015.

[16] K. Li, K. Deb, Q.F. Zhang, and S. Kwong, "An evolutionary many-objective optimization algorithm based on dominance and decomposition," *IEEE Trans. Evol. Comput.*, vol. 19, no. 5, pp. 694–716, 2015.

[17] H. Ishibuchi, Y. Setoguchi, H. Masuda, and Y. Nojima, "Performance of Decomposition-Based Many-Objective Algorithms Strongly Depends on Pareto Front Shapes," *IEEE Trans. Evol. Comput.*, vol. 21, no. 2, pp. 169–190, 2017.

[18] B. Li, J. Li, K. Tang, and X. Yao, "Many-objective evolutionary algorithms: A survey," *ACM Computing Surveys*, vol. 48, no. 1, pp. 13:1–13:35, 2015.

[19] J. Bader and E. Zitzler, "HypE: an algorithm for fast hypervolume based many-objective optimization," *Evol. Compu.t*, vol.19, no. 1, pp. 45-76, 2011.

[20] S. Jiang, J. Zhang, "A simple and fast hypervolume indicator-based multiobjective evolutionary algorithm," *IEEE Trans. Cybern.,* vol. 45, no. 10, pp. 2202-2213, 2015.

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

[22] H. Wang, Y. Yao "Corner sort for Pareto-based many-objective optimization," *IEEE Trans. Evol. Cybern.*, vol. 44, no. 1, pp. 92-102, 2014.

[23] M. Elarbi, S. Bechikh, A. Gupta, L. Ben, and Y.S. Ong, "A new decomposition-based NSGA-II for many-objective optimization," *IEEE Trans. on Systems, Man, and Cybernetics: Systems*, vol. 48, no.

7, pp. 1191-1210, 2018.

[24] C. Zhu, L. Xu, and E. D. Goodman, "Generalization of Pareto-optimality for many-objective evolutionary optimization," *IEEE Trans. Evol. Comput.*, vol. 20, no. 2, pp. 299-315, 2016.

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

[26] 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 Trans. Evol. Comput.*, vol. 18, no. 4, pp. 577-601, 2014.

[27] K. Bringmann and T. Friedrich, "An efficient algorithm for computing hypervolume contributions," *Evol. Comput.*, vol. 18, no. 3, pp. 383–402, 2010.

[28] L. M. Russo and A. P. Francisco, "Quick hypervolume," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 481–502, 2014.

[29] L. While, L. Bradstreet, and L. Barone, "A fast way of calculating exact hypervolumes," *IEEE Trans. Evol. Comput.*, vol. 16, no. 1, pp. 86–95, Feb. 2012.

[30] D. Brockhoff, T. Wagner, and H. Trautmann, "On the properties of the R2 indicator," in Proc. 14th Annu. Conf. Genet. Evol. Comput., Philadelphia, PA, USA, 2012, pp. 465–472.

[31] R. H. Gomez and C. A. C. Coello, "MOMBI: A new metaheuristic for many-objective optimization based on the R2 indicator," in Proc. IEEE Congr. Evol. Comput., Cancún, Mexico, 2013, pp. 2488–2495.

[32] M. Wagner and F. Neumann, "A fast approximation-guided evolutionary multi-objective algorithm," in Proc. 15th Annu. Conf. Genet. Evol. Comput. Conf., Amsterdam, The Netherlands, pp. 687–694, 2013.

[33] B.D. Li, K. Tang, J.L. Li, and X. Yao, "Stochastic ranking algorithm for many-objective optimization based on multiple indicators", *IEEE Trans. Evol. Comput.*, vol. 20, no. 6, pp. 924-938, 2016.

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

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

[36] L. Cai, S. Qu and G. Cheng, "Two-archive method for aggregation-based many-objective optimization," *Inform. Sci.* 422 (2018) 305-317.

[37] Y. Xiang, Y. R. Zhou, M. Q. Li and Z. Chen, "A vector angle-based evolutionary algorithm for

unconstrained many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 21, no. 1, pp. 131-152, 2017.

[38] J. Cheng, G. Yen, G. Zhang, "A many-objective evolutionary algorithm with enhanced mating and environmental selections," *IEEE Trans. Evol. Comput.*, vol. 19, no. 4, pp. 592-605, 2015.

[39] Z. He and G. Yen, "Many-objective evolutionary algorithms based on coordinated selection strategy," *IEEE Trans. Evol. Comput.*, vol. 21, no. 2, pp. 220-233, 2017.

[40] R. Wang, R. Purshouse, and P. J. Fleming, "Preference-inspired coevolutionary algorithms for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 17, no. 4, pp. 474-494, 2013.

[41] E. Figueiredo, T. Lundermir and C. Bastos-Filho, "Many objective particle swarm optimization," *Inform. Sci.* 374 (2016) 115-134.

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

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

[44] M. Ming, R. Wang, Y. Zha and T. Zhang, "Pareto adaptive penalty-based boundary intersection method for multi-objective optimization," *Inform. Sci.* 414 (2017) 158-174.

[45] X. Ma, Q. Zhang, G. Tian, J. Yang, and Z. Zhu, "On Tchebycheff decomposition approaches for multi-objective evolutionary optimization," *IEEE Trans. Evol. Comput.*, vol. 22, no. 2, pp. 226-244, 2018.

[46] H. Sato, "Inverted PBI in MOEA/D and its impact on the search performance on multi and many-objective optimization," in Proc. Genet. Evol. Comput. Conf., Vancouver, BC, Canada, Jul. 2014, pp. 645–652.

[47] R. Wang, Z. B. Zhou, and H. ishibuchi, "Localized weighted sum method for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 22, no. 1, pp. 3–18, 2018.

[48] R. Wang, Q. Zhang and T. Zhang, "Decomposition-based algorithms using Pareto adaptive scalarizing methods," *IEEE Trans. Evol. Comput.*, vol. 20, no. 6, pp, 821-837, 2016.

[49] S. Huband, L. Barone, L. While, P. Hingston, A scalable multi-objective test problem toolkit, in: C.A.C. Coello, A.H. Aguirre, E. Zitzler (Eds.), Proceedings of Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science, 3410, Guanajuato, Mexico, 2005, pp. 280–295.

[50] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable test problems for evolutionary multi-objective optimization," in *Evolutionary Multiobjective Optimization. Theoretical Advances*

and Applications, A. Abraham, L. Jain, and R. Goldberg, Eds. Berlin, Germany: Springer, 2005, pp. 105–145.

[51] Z. He and G. Yen, “Many-objective evolutionary algorithm: object space reduction and diversity improvement,” *IEEE Trans. Evol. Comput.*, vol. 20, no. 1, pp. 145-160, 2016.

[52] H. Wang, L. Jiao and X. Yao, “Two_Arch2: an improved two-archive algorithm for many-objective optimization”, *IEEE Trans. Evol. Comput.*, vol. 19, no. 4,pp. 524–541, 2015.

[53] H. Li and Q. Zhang, “Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II,” *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, 2009.

[54] K. Li, Q. Zhang, S. Kwong, M. Li, and R. Wang, “Stable matching-based selection in evolutionary multiobjective optimization,” *IEEE Trans. Evol. Comput.*, vol. 18, no. 6, pp. 909–923, 2014.

[55] Q. Lin, G. Jin, Y. Ma, K. Wong, C.A. Coello Coello, J. Li, J. Chen and J. Zhang, “A diversity-enhanced resource allocation strategy for decomposition-based multiobjective evolutionary algorithm,” *IEEE Trans. Cybern.*, vol. 48, no. 8, pp. 2388-2401, 2018.

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

[57] Z. Wang, Q. Zhang, A. Zhou, M. Gong and L. Jiao, “Adaptive replacement strategies for MOEA/D," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp, 474-486, 2016.

[58] K. Li, S. Kwong, Q. Zhang, and K. Deb, “Interrelationship-based selection for decomposition multiobjective optimization,” *IEEE Trans. Cybern*., vol. 45, no. 10, pp. 2076–2088, 2015.

[59] K. Deb and R. B. Agrawal, “Simulated binary crossover for continuous search space,” *Complex Syst.*, vol. 9, pp. 115–148, 1995.

[60] J.J. Durillo, A.J. Nebro, E. Alba, “The jMetal framework for multi-objective optimization: Design and architecture,” in Proc. IEEE Congr. Evol. Comput., Barcelona, Spain, 2010, pp. 1–8.

[61] Ye Tian, Ran Cheng, Xingyi Zhang, and Yaochu Jin, PlatEMO: A MATLAB Platform for Evolutionary Multi-Objective Optimization [Educational Forum], *IEEE Comput Intell Mag*, 2017, 12(4): 73-87

[62] H. Ishibuchi, K. Doi and Y. Nojima, “Reference point specification in MOEA/D for multi-objective and many-objective problems,” in Proc. IEEE International Conference on SMC, Budapest, Hungary, 2016, pp. 4015–4020.

[63] H. Ishibuchi, H. Masuda and Y. Nojima, “Comparing solution sets of different size in evolutionary many-objective optimization,” in Proc. IEEE Congr. Evol. Comput., Sendai, Japan, 2015, pp.

2859–2866

[64] H. Ishibuchi, H. Masuda, and Y. Nojima, “Sensitivity of performance evaluation results by inverted generational distance to reference points,” in Proc. IEEE Congr. Evol. Comput., Vancouver, BC, Canada, 2016, pp. 1107-1114.

[65] K. Li, A. Fialho, S. Kwong, and Q. Zhang, “Adaptive operator selection with bandits for multiobjective evolutionary algorithm based decomposition,” *IEEE Trans. Evol. Comput.*, vol. 18, no. 1, pp. 114–130, 2014.