Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/mbe

MBE, 20(9): 17324–17355.
DOI: 10.3934/mbe.2023771
Received: 03 June 2023
Revised: 15 July 2023
Accepted: 24 July 2023
Published: 08 September 2023

Research article

A double association-based evolutionary algorithm for many-objective optimization

Junhua Liu[1,*], Wei Zhang[2], Mengnan Tian[1], Hong Ji[1] and Baobao Liu[1]

[1] The Shaanxi Key Laboratory of Clothing Intelligence, School of Computer Science, Xi'an Polytechinic University, Xi'an 710048, China
[2] College of Information Science and Engineering, Northeastern University, Shenyang, China

* **Correspondence:** Email: liujunhua@xpu.edu.cn.

Abstract: In this paper, a double association-based evolutionary algorithm (denoted as DAEA) is proposed to solve many-objective optimization problems. In the proposed DAEA, a double association strategy is designed to associate solutions with each subspace. Different from the existing association methods, the double association strategy takes the empty subspace into account and associates it with a promising solution, which can facilitate the exploration of unknown areas. Besides, a new quality evaluation scheme is developed to evaluate the quality of each solution in subspace, where the convergence and diversity of each solution is first measured, and in order to evaluate the diversity of solutions more finely, the global diversity and local diversity is designed to measure the diversity of each solution. Then, a dynamic penalty coefficient is designed to balance the convergence and diversity by penalizing the global diversity distribution of solutions. The performance of DAEA is validated by comparing with five state-of-the-art many-objective evolutionary algorithms on a number of well-known benchmark problems with up to 20 objectives. Experimental results show that our DAEA has high competitiveness in solving many-objective optimizatiopn problems compared with the other compared algorithms.

Keywords: many-objective optimization; double association; quality evaluation; convergence; diversity

1. Introduction

Multi-objective optimization problems (MOPs) exist in many fields of production practice [1]. Thus, lots of multi-objective evolutionary algorithms (MOEAs) have been designed to deal with MOPs with two or three objectives [2]. However, with the continuous improvement of people's requirements for sophistication, optimization models established in many complex industrial processes [3, 4] and industrial applications [5–7] often involve more than three objectives, which are called as many-objective optimization problems (MaOPs). Generally, an MaOP involving m conflicting objectives can be formulated as follows:

$$\begin{aligned} &Minimize \quad F(x) = (f_1(x), f_2(x), \ldots, f_m(x))^T \\ &s.t. \quad x \in \Omega \subseteq R^n \end{aligned} \tag{1}$$

where $x = (x_1, x_2, \ldots, x_n)^T$ is the decision vector in the decision space Ω. $F(x)$ denotes the objective vector and $f_i(x)$ is the ith objective ($i = 1, 2, \ldots, m$). Due to the conflicting property of different objectives, it is almost impossible to find a solution that optimizes all objectives simultaneously [8]. On the contrary, a set of trade-off solutions is needed, which is known as the Pareto front (PF) in the objective space and the Pareto set (PS) in the decision space [9].

Since the number of optimization objectives are increased, most of the solutions based on Pareto-dominance method become nondominated [10] and considerably slows down the evolutionary process, which leads to most Pareto dominance based MOEAs that fail to solve MaOPs. Besides, as the dimension of the objective space increases, the finite solutions become increasingly scarce in the high-dimensional objective space. As a result, many frequently used diversity maintenance strategy, such as crowding distance [11], are no longer effective on MaOPs.

To better solve MaOPs, researchers have proposed many different methods, which can be divided into the following five categories [12]:

The first category is modifying Pareto-dominance relation or developing new dominance relations to increase the selection pressure toward the PF for MaOPs. Examples of modifying dominance definitions include k-optimality relation [13], average and maximum ranking relations [14], ε-dominance [15], L-optimality [16], grid-dominance [17], preference order ranking [18], fuzzy dominance [19], θ-dominance [20] and RP-dominance [21]. The distinctive modifications have shown to be very promising for MaOPs. However, most of them can only be used within a specific algorithmic framework. Another typical idea in this category is to adjust the selection pressure through changing the dominated area of each solution, such as α-dominance [22]. Two representatives are CE-dominance [23] and CN-dominance [24]. The former can dynamically adjuste the solutions' dominant area by varying a parameter S, and thereby adjust the selection pressure. The latter transforms the original objectives by using the author designed nonlinear function to further adjust the area dominated by each solution. While the idea of dynamically adjusting the dominated area of each solution enables Pareto-based MOEAs to enhance selection pressure among solutions, ensuring the quality of the selected solutions remains a challenging endeavor.

The second category is the indicator-based approach, which uses an evaluation indicator instead of Pareto-dominance to evaluate the quality of solutions [25]. Among the current indicators available, the IGD indicator as a comprehensive indicator is used to select the potential solutions in each generation (MaOEA-IGD) [26]. The hypervolume (HV) indicator [27] possesses good theoretical properties and is often used as the indicator function in multiobjective search. However, the high computation cost of the hypervolume holds back its spread, especially for MaOPs with more than five objectives [28]. Bader and Zitzler [29] suggested a new HV-based algorithm (HypE), which adopts Monte Carlo simulation to replace the exact hypervolume calculations, causing the computational complexity been alleviated, but in turn, the inexactly calculation of HV deteriorates the performance of HypE on MaOPs [30].

The third category is the objective reduction-based methods. Since some many-objective optimization problems have redundancy between the objectives, this method aims to identify the most relevant objectives first and then expurgate redundant ones that do not contribute significantly to describing the Pareto front. Deb and Saxena [31] developed a novel algorithm called PCA-NSGA-II, which combines principal component analysis (PCA) with NSGA-II for effectively handling MaOPs that have redundant objectives. Moreover, Singh et al. [32] designed a novel approach that emphasizes a reduced set of objectives instead of dealing with the complete dimensionality of the MaOP. Meanwhile, they adopt corner solutions, which are located in the boundaries of PF, to estimate the dimensionality of the true PF. Recent research has confirmed that certain methods based on objective reduction are susceptible when dealing with optimization problems that involve a high-dimensional PF [33]. In other words, these techniques may be effective for addressing problems that involve a moderate number of conflicting objectives.

The fourth technique is to use preference incorporation-based approaches, where the timing of integrating preference information into the optimization process is a crucial factor. The preference information given by decision maker (DM) can make the search direction that is biased towards the area of the PF, which is focused by DM, the many-objective optimization algorithms combined with the preference information can reduce the consumption of computing resources during the search [33]. For example, Wang et al. [34] developed a novel algorithm named PICEA-g, which enables decision-makers to obtain a comprehensive and diverse representation of the Pareto fronts, prior to eliciting and applying their preferences. In [35], a reference vector guided evolutionary algorithm is proposed, where the reference vectors not only decompose a MaOP into single-objective subproblems, but also elucidate user preference to target a preferred subset of the whole PF. This shows promising performance on certain problem types, but they suffer from suboptimal results since the bias may lead to premature convergence [36].

The last technique is a decomposition-based method, which decomposes a complex MOP into a number of subproblems and simultaneously optimizes them in a collaborative manner, is often known as another promising way. MOEA/D [37] is the most typical representative of this method. By assigning a set of evenly distributed weight vectors, MOEA/D maintains population diversity better and has the ability to solve many kinds of optimization problems with varying degrees of success [38–44]. Based on the framework of MOEA/D, MaOACO-RP utilizes the designed penalty boundary intersection and adaptive reference points to pick out solutions for the next generation [38]. Zhang et al. [45] integrated the designed information feedback model into MOEA/D for solving MaOPs with many decision variables. MaOEA-IT [46] adopts two stages to balance the convergence and diversity. Recently, the reference point-based MOEAs are highlighted in many-objective optimizations. A little bit different from decomposition-based MOEAs, the reference points are used to facilitate the diversity of selected solutions by associating each solution in objective space with them, e.g., NSGA-III [47], which adopts a reference point-based niche-preservation operation to replace the crowding distance operator in NSGA-II. Recently, Yi et al. [48] combines the crossover operator with the NSGA-III for MaOPs with many decision variables. However, with the increasing number of objectives, Pareto-dominance gradually loses its ability to filter solutions, thus, the solutions obtained by NSGA-III stress diversity more than convergence. In literature [20], a new algorithm called θ-DEA is proposed to solve MaOPs, where a reference points-based dominance relation (θ-dominance) is designed to select the solutions. The experimental results show that θ-DEA can better improve the performance of NSGA-III for MaOPs, especially the convergence. In SPEA/R [30], it designs a new reference direction based fitness assignment scheme and an environmental selection strategy to improve the performance of SPEA2 in solving both multi-objective and many-objective problems, but the performance of SPEA/R remains constrained to Pareto-dominance with the number of objectives increasing.

In order to better maintain the diversity of solutions in objective space, and motivated by the idea of dividing objective space by well-distributed reference vectors to select the quality of obtained solutions, we propose a double association-based evolutionary algorithm for solving MaOPs. In comparison with existing approaches that rely on reference points/vectors, this paper offers the following key contributions:

1) A double association strategy is designed to associate each solution with a reference vector. When the objective space is divided by the well-distributed reference vectors, different from other existing association methods [20, 30], the double association strategy takes the empty subspace into account and associates it with the nearest solution. This new association method can facilitate the exploration of unknown areas.

2) A new quality evaluation scheme is proposed to quantify the quality of each solution in subspace, where the convergence and diversity of each solution is first computed, and then a dynamic penalty coefficient is designed to balance the convergence and diversity by penalizing the diversity distribution of solutions.

The remaining structure of this paper is as follows: Section 2 elaborates the 3DEA in detail. Section 3 presents the experimental design for solving many-objective optimization. In Section 4, the experimental results are conducted and analyzed. Finally, Section 5 presents the conclusions of the study.

2. Proposed algorithm: DAEA

2.1. General framework

The framework of DAEA is given in Algorithm 1 and Figure 1 shows the flowchart of DAEA, which mainly corresponds to Algorithm 1. First, we employ the Das and Dennis's [49] systematic approach to generate evenly reference vectors. Then, an initial population P_0 with N individuals are randomly produced. The ideal point z^* and nadir point z^{nad} are initialized in step 3 and step 4, respectively. They are computed by the minimum value and maximum value of obtained population P_0 for objective f_i, respectively, and both of them are updated in each iteration. Steps 6–23 denote that DAEA enters the iterative procedure. In step 7, on the basis of the preserved parent population P_t, the offspring population Q_t is produced by employing the widely used genetic operators, i.e., the simulated binary crossover (SBX) [50] and the polynomial mutation (PM) [51]. Meanwhile, a combined population R_t is formed by combining the parent population P_t and offspring population Q_t. The population $S_t = \bigcup_{i=1}^{\tau} F_i$, where F_i is the ith Pareto nondominated level of R_t and τ satisfies $\sum_{i=1}^{\tau-1} |F_i| < N$ and $\sum_{i=1}^{\tau} |F_i| \geq N$. In step 11, the normalization procedure [20] is implemented to normalize the objective population $\{F(x)|x \in S_t\}$ of S_t to $\tilde{F}(S_t) = \{\tilde{F}(x)|x \in S_t\}$, and then each member in $\tilde{F}(S_t)$ is associated with a reference vector by our proposed double association strategy in step 12. For each solution in

Algorithm 1 Main framework of the proposed DAEA

Input: the maximal number of fitness evaluations(MFEs)
Output: Final population P

1: $W \leftarrow$ Generate Reference vector(N);
2: $P_0 \leftarrow$ Initialize Population(N);
3: $z^* \leftarrow$ Initialize IdealPoint(P_0);
4: $z^{nad} \leftarrow$ Initialize NadirPoint(P_0);
5: $t \leftarrow 0$;
6: **while** the termination criterion is not met **do**
7: $Q_t \leftarrow$ Create Offspring Population(P_t);
8: $R_t \leftarrow$ Union Population $P_t \cup Q_t$;
9: $S_t \leftarrow$ Pareto Nondominated Levels(R_t);
10: Update Ideal Point(S_t);
11: $\tilde{F}(S_t) \leftarrow$ Normalize (S_t, z^*, z^{nad});
12: $[S(1), ..., S(N)] \leftarrow$ Double Association $(\tilde{F}(S_t), W)$;
13: $[T(1), T(2), ...] \leftarrow$ Quality Evaluation $[S(1), ..., S(N)]$;
14: $P_{t+1} \leftarrow \emptyset$;
15: $i \leftarrow 1$;
16: **while** $|P_{t+1}| + |T(i)| < N$ **do**
17: $P_{t+1} \leftarrow P_{t+1} \bigcup T(i)$;
18: $i \leftarrow i + 1$;
19: **end while**
20: Random Sort($T(i)$);
21: $P_{t+1} \leftarrow P_{t+1} \bigcup T(i)[1 : (N - |P_{t+1}|)]$;
22: $t \leftarrow t + 1$;
23: **end while**

its associated subspace, the proposed quality evaluation scheme is used to evaluate the quality of each solution, and then sort them according the evaluated values, the first-ranked solution in each subspace are put into the set $T(1)$, the second-ranked solution in each subspace are put into the set $T(2)$, and so on. Then steps 14–21 fill the population slots in P_{t+1} using one level at a time, starting from $T(1)$, for the solutions in last accepted level $T(i)$, we randomly select the number of needed solutions in $T(i)$, since our proposed quality evaluation scheme takes both convergence and diversity into account. In the following sections, the implementation details of important component of DAEA will be described step by step.

2.2. *Reference vectors generation*

In order to maintain the diversity of obtained solutions, evenly reference vectors $W = (\lambda_1, \lambda_2, \cdots, \lambda_N)$ are generated by Das and Dennis's [42] systematic approach in DAEA, which divide the objective space into N independent subspaces and reserve the most promising solutions of each subspace as much as possible. In this approach, reference vectors are sampled from a unit

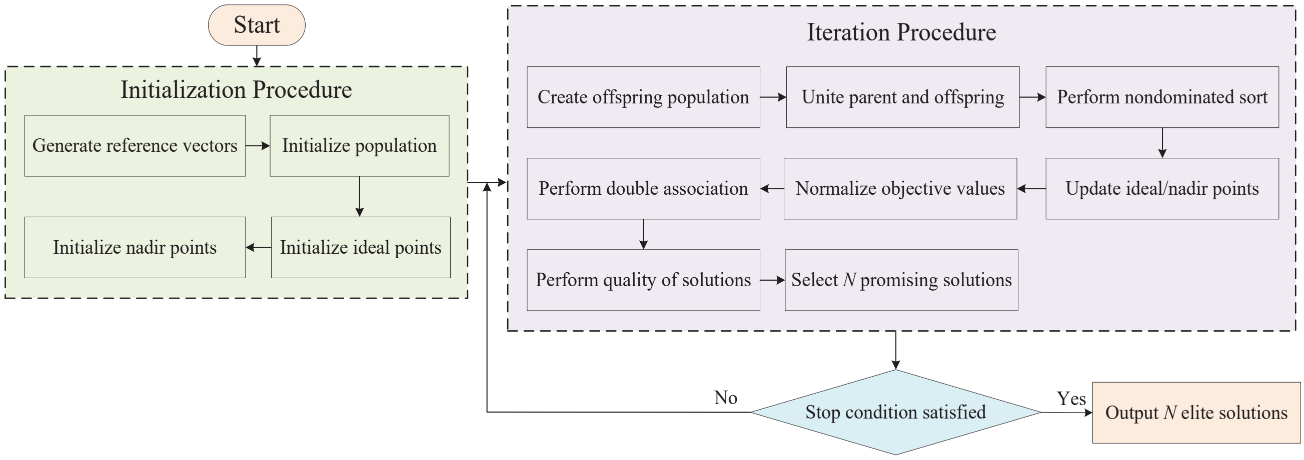


Figure 1. The flowchart of DAEA.

simplex and the number of reference vectors is equal to $N = \binom{H+m-1}{m-1}$, where m is the dimension of objective space and H is the number of divisions considered along each objective axis.

Suppose that $\lambda_i = (\lambda_i^1, \lambda_i^2, \cdots, \lambda_i^m)$ is the ith reference vector, let us consider

$$\sum_{j=1}^{m} u_i^j = H, u_i^j \in \mathrm{N} \tag{2}$$

Then, the elements of reference vector λ_i is obtained by

$$\lambda_i^k = \frac{u_i^k}{H}, k = 1, 2, \cdots, m \tag{3}$$

It can be clearly observed that when $H \geq m$, the above method will result in lots of reference vectors. Even if $H = m$, $N = \binom{8+8-1}{8-1} = 6435$. This will significantly increase the computational cost. However, if we adopt the method reducing H to relieve the computational burden, the generated reference vectors may unevenly distribute. To avoid such situation, the two-layered reference vectors with small values of H as suggested in [47] is adopted in DAEA. Supposing the number of divisions of boundary and inner layers is H_1 and H_2, respectively, then the reference vectors are generated by Das and Dennis's method. The number of reference vectors (i.e., population size) is computed as

$$N = \binom{H_1+m-1}{m-1} + \binom{H_2+m-1}{m-1} \tag{4}$$

2.3. *Double association strategy*

After the whole objective space is divided into N independent subspaces, the population distributed in the target space needs to be assigned to different subspaces. The idea of assigning the population by associating each member in the normalized objective set with its closest reference vector has been employed in many recent papers, such as [20, 21, 30, 43, 52], etc. However, they have different characteristics and motivations. In [52], when a subspace contains no solution, i.e., the subspace is empty, a solution is randomly chosen and associated with this subspace. This means a randomly selected solution is assigned to the empty subspace. This kind of association methods is the random association. Random association aims to increase the probability of the unknown region being explored, however, the operation of randomly selecting solutions may make the search even worse. In [34], each nondominated solution is associated with a subspace and then the density of each subspace is estimated by counting the number of solutions associated with the subspace. Since there may be some empty subspaces, it may lead to some incorrect estimation. Also, in [20, 21, 30], when empty subspace appears, these association methods have no further action and directly ignore them, which will destroy the diversity of the obtained solutions. This kind of association methods is the single association.

To better explore the unknown region and increase the diversity, we design a double association strategy, in which we do the first step of association as same as that in the existing one [53], i.e., each reference vector will generate a subspace, and each solution will be associated with its closest subspace, but like the discussion above, this can result in empty subspace(s) and can not keep the good diversity of the obtained solutions. To overcome this shortcoming, we do the second step of association. For each empty subspace, we associate a closest solution with it, in which the perpendicular Euclidean distance d_2 (the distance of each solution from each reference vector) [46] is used to measure the distance between the solution and the subspace (represented by the corresponding reference vector). In this way, the diversity of the obtained solutions will be greatly improved. The detail is as follows.

Currently, we have a normalized union population $\tilde{F}(R_t)$ (see Algorithm 1) in the objective space. Suppose that $\tilde{F}(x) = (\tilde{f}_1(x), ..., \tilde{f}_m(x))^T \in \tilde{F}(R_t)$, the ideal point is the origin, then the Euclidean distance between the origin and the foot of the normal drawn from the solution x to the reference vector λ_i is denoted as $d_1^i(x)$, and the length of the normal is denoted as $d_2^i(x)$. To be specific, their mathematical descriptions are [16]:

$$d_1^i(x) = \left\| \tilde{F}(x)^T \lambda_i \right\| / \|\lambda_i\| \tag{5}$$

$$d_2^i(x) = \left\| \tilde{F}(x) - d_1^i(x)(\lambda_i / \|\lambda_i\|) \right\| \tag{6}$$

For visually presenting them, Figure 2 illustrates the two distance measures $d_1^i(x)$ and $d_2^i(x)$ with respect to the reference vector λ_i in a two-objective minimization problem.

In the first step association, for each solution $x_i \in R_t$, we first compute its perpendicular Euclidean distance $d_2^j(x_i)$ between $\tilde{F}(x_i)$ and each $\lambda_j (j = 1, 2, ..., N)$, then the solution x_i is associated with its closest subspace (corresponding to the closest reference vector) based on the distance $d_2^j(x_i)$ for $j = 1, 2, ..., N$. After all solutions in R_t have been associated with their closest subspaces, we begin the second step association. We examine all subspaces to see whether there is an empty subspace. If yes, we associate the empty subspace with its closest solution in R_t using perpendicular Euclidean distance. Otherwise, we do not use the second step association. In this way, not only can the diversity of the obtained solutions be improved, but also the solution which is closest to the empty subspace will be preserved, which is very helpful to improve the diversity and can generate potential solutions in the subsequent evolution. The detailed description of the double association strategy is given in Algorithm 2.

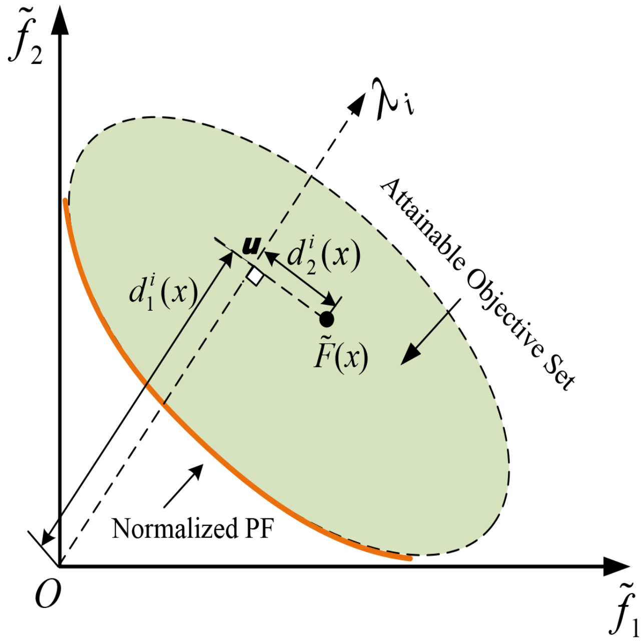


Figure 2. Illustration of distances $d_1^i(x)$ and $d_2^i(x)$.

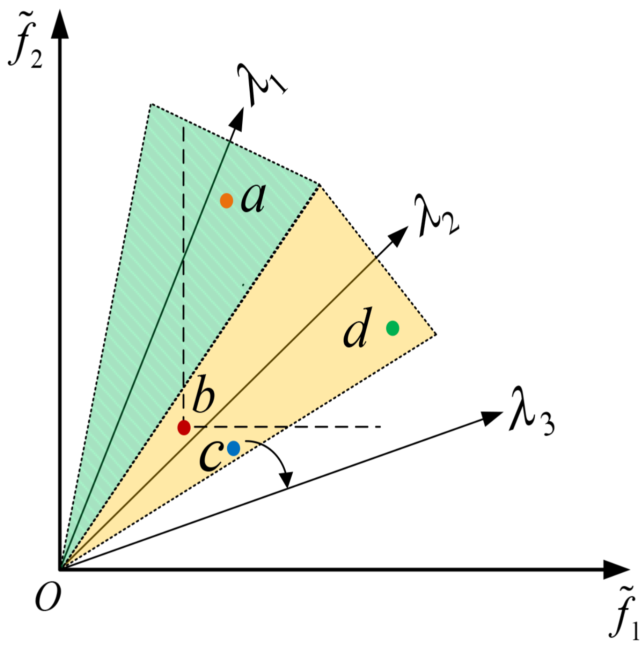


Figure 3. Influence of three different association schemes on the performance of the algorithm.

Algorithm 2 Double association

Input: $\tilde{F}(R_t)$(normalized union population),
W(reference vector set)
Output: $\{S(1), S(2), \ldots, S(N)\}$ ($S(i)$ stores all the individuals associated with the ith reference vector)

```
{S(1), S(2), ..., S(N)} ← {∅, ∅, ..., ∅};
for each x_i ∈ R̃_t do
    for each λ_j ∈ W do
        Compute the perpendicular Euclidean distance d_2^j(x_i) between F̃(x_i) and λ_j;
    end for
    k = argmin_{j∈{1,...,|w|}} d_2^j(x_i);
    S(k) = S(k) ∪ x_i;
end for
for each S(j) do
    if isempty S(j) then
        q = argmin_{i∈{1,...,|R̃t|}} d_2^j(x_i);
        S(j) = S(j) ∪ x_q
    end if
end for
```

To intuitively demonstrate the advantages of our proposed double association strategy, we compare it with two kinds of existing association schemes: Random association (randomly chooses a solution to associate it with an empty subspace in the second step association), e.g., [52], and single association (without second step association), e.g., [20, 21, 30]. Figure 3 gives a simple example to illustrate the influence of the random association, the single association and the proposed double association strategy on the performance of the algorithm. For these three association methods, the first step association operation of them are the same, we suppose that after the first step association operation, solution a is associated with sub-space $S(1)$ and b, c, d are all associated with sub-space $S(2)$. There is no solution associated with sub-space $S(3)$. If one uses the single association, such as ones in [20,21,30], $S(3)$ will be empty. This will result in bad diversity of solutions. If one uses the random association, a randomly chosen solution among $\{a, b, c, d\}$ (such as d) is assigned to $S(3)$, which also results in bad diversity of the solutions. If one uses our double association, since c is closest to λ_3, it will be directly assigned to associate with subspace $S(3)$, which helps to improve the diversity of the solutions. This is very necessary for exploring unknown areas. Besides, the survival of solution a in subspace $S(1)$ is very important for the unexploited area.

2.4. Quality evaluation scheme

For comprehensively assessing the quality of each solution, we design a novel quality evaluation function for the solutions in each subspace $S(i)$, where this evaluation function takes into account both the convergence measure and diversity measure of each solution. The mathematical description of the new quality value of solution x is as follows:

$$FV(x) = d_1^i(x) + \mu \cdot d_2^i(x) + \min_{y \in S(i), x \neq y} d(x,y) \tag{7}$$

where $d_1^i(x)$ means the shortest perpendicular Euclidean distance of solution x from the ideal point and is used to measure the convergence of solution x. The smaller the value of $d_1^i(x)$, the better convergence of solution x. The shortest perpendicular Euclidean distance $d_2^i(x)$ among the distances of each solution x from each reference vector λ_i is used to measure the contribution of solution x to the global diversity, the smaller the value of $d_2^i(x)$, the better the diversity of solution x. $\min_{y \in S(i), x \neq y} d(x,y)$ represents the minimum distance from the solution x to all other solutions associated to the subspace $S(i)$, which is used to measure the contribution of solution x to the local diversity. For diversity measure, the smaller the value of $d_2^i(x)$ and $\min_{y \in S(i), x \neq y} d(x,y)$, the better the diversity of solution x. The parameter μ is a penalty parameter, and is designed to punish the solutions in dense area from global perspective. Considering that in an ideal state, each subspace should have two solutions distributed, here we set $\mu = \frac{|S(i)|}{2}$. As the subspace $S(i)$ is distributed with more solutions, the parameter μ is going to get bigger, which means stronger punishment will be inflicted on local diversity. Therefore, when a solution obtains a smaller quality value FV, it indicates that the solution possesses superior convergence and diversity characteristics.

Therefore, the new quality evaluation simultanously consider convergence measure and diversity measure and can adaptively protect the solutions located in the sparse areas. The quality value of a solution is lower, which denotes the solution has better convergence and diversity. After the solutions in each subspace are assigned to the corresponding quality value, we ascending sort the solutions in subspace $S(i)$ as $Num_i(1), ..., Num_i(|S(i)|)$ based on the quality value FV. Then, the first-ranked solution in each subspace are put into the set $T(1)$, the second-ranked solution in each subspace are put into the set $T(2)$, and so on. The details are shown in Algorithm 3.

Algorithm 3 Quality evaluation scheme

Input:
d_1, d_2, $\{S(1), S(2), \ldots, S(N)\}$ ($S(i)$ stores the individuals associated with the ith reference vector)
Output: $[T(1), T(2), ...]$

for $i = 1 : N$ **do**
    for Each $x \in S(i)$ **do**
        Assign a quality value $FV_i(x)$ by Eq.(7);
        Sort FV_i as $[Num_i(1), Num_i(2), ..., Num_i(|S(i)|)]$;
    end for
end for
for $k = 1 : \max_{j=1:N} |S(j)|$ **do**
    for $i = 1 : N$ **do**
        $T(k) = [T(k), Num_i(k)]$;
    end for
end for

2.5. Computational complexity of DAEA

The computational complexity of DAEA in one generation is dominated by the double association operator that is described in Algorithm 2. For the procedure of double association, there are two steps: In the first step of association (lines 2–8 of Algorithm 2), for a solution x, calculating N perpendicular distances from x to N reference vectors needs $O(mN)$ computational cost, and calculating the shortest perpendicular distance from x to N reference vectors requires $O(N)$ computational cost. Thus, the association between $2N$ solutions with N reference vectors needs $O(mN^2)$ computational cost. In the second step association (lines 9–14 of Algorithm 2), finding the empty subspace from N subspaces and then associating it with a solution closest to it needs $O((N-1)N)$ operations in the worst situation. Thus, the overall worst complexity of one generation of DAEA is approximately $O(mN^2)$.

3. Experimental design

In this section, experimental design is given to verify the performance of the proposed DAEA on many-objective optimization problems. First, the test problems and the compared algorithms are presented. Then, we will give a brief rundown of the performance metrics utilized in our experimental study. Next, we give parameter settings in this work. All experiments are carried out in the open-source software PlatEMO [54] that has been widely used in many-objective optimization.

3.1. Test problems

In order to test the performance of six algorithms involved in this paper, DTLZ test suite (DTLZ1-DTLZ7) [55] and WFG test suite (WFG1-WFG9) [56] as two common used many-objective optimization benchmarks are adopted in our experiments. They contains many different features, such as degenerate, biased, large scale, non-separable and partially separable of the decision variable in decision space as well as linear, convex, concave, mixed geometric structures and multi-modal of the PFs in objective space. These different features pose great challenges to the comprehensive performance of an algorithm. Besides, the objectives and decision variables of all these test problems can be scaled to any number.

In our experiment, the number of objectives for all test problems are taken from 5 to 20, i.e., $m \in \{5, 8, 12, 16, 20\}$. For DTLZ test suite, the number of decision variables is given by $n = m + k - 1$. Different test problems may have different values of k. As suggested in [41, 48], the parameter k is set to 5 for DTLZ1, 10 for DTLZ2-DTLZ6 and 20 for DTLZ7. For WFG test suite, as suggested by the references [50] and [49], the number of decision variables for all test problems are set to 24 and the position related parameter is set to $m - 1$.

To test the performance of our proposed DAEA in solving MaOPs, five well-known MaOEAs are selected to compared with DAEA based on a series of experiments. To be specific, NSGA-III [47] is selected since it is typically used as a baseline for solving MaOPs. θ-DEA [20] adopts the single association method to decompose the population. SPEA/R [30] is mixed algorithm that combines decomposition and Pareto dominance. MaOEA-IGD [26] is typically indicator-based algorithm for solving MaOPs. MaOEA-IT [46] is a recently proposed MaOEAs which employs two-stage strategy to take the convergence and diversity of solutions into account.

3.2. Performance metrics

We select two widely used performance metrics to evaluate the performance of DAEA and other compared algorithms on MaOPs. More details are as follows:

(1) Inverted generational distance (IGD) [57]: The IGD metric can measure both the convergence and diversity of the obtained solution set, whose mathematical description is:

$$IGD(A,P^*) = \frac{1}{|P^*|}\sum_{i=1}^{|P^*|}\min_{f\in A} d(f_i{}^*, f) \tag{8}$$

where P^* is a set of solutions obtained by uniformly sampling from the true PF and A refers to non-dominated solutions obtained by an algorithm. $|P^*|$ is the size of P^*, and $d(f_i{}^*, f)$ represents the Euclidean distance from the point in true PF to their closest to the obtained population A. A smaller IGD value of a set A means a better performance of the set A.

(2) Hypervolume (HV) [27]: HV is a comprehensive indicator, which can assess both the convergence and diversity of a solution set as well. Its mathematical description is as follows:

$$HV(S) = \mathrm{VOL}\left(\cup\left[f_{r\in A}\left[f_1(x), z_1^r\right] \times \ldots \left[f_m(x), z_m^r\right]\right)\right. \tag{9}$$

where $\mathrm{VOL}(\cdot)$ represents the Lebesgue measure, A is the obtained non-dominated solutions, and $z = (z_1, \cdots, z_m)$ is the reference point and set to (1,1,·,1). In addition, When the number of objectives m exceeds 8, we employ the Monte Carlo method with 1,000,000 sampling points to approximate the HV values [25]. Obviously, the larger the HV value, the higher the quality of solution set A is.

3.3. Experimental settings

In this section, we outline the general experimental settings and provide specific parameter settings for each algorithm, as described below.

(1) Population size: The population size N of these algorithms using reference vectors (such as NSGA-III, θ-DEA) is determined by the parameter H and the number of objectives m [47]. To have a fair comparison, the population size of other compared algorithms keep identical to the above algorithms. Table 1 lists the detailed settings of population size N for problems with different number of objectives used in this paper.

Table 1. Settings of the population size.

No. of objectives (m)	Divisions (H)	Population size (N)	MFEs
5	6	210	99,960
8	(4,1)	338	99,990
12	(3,1)	376	100,100
16	(2,2)	272	100,386
20	(2,1)	230	99,960

(2) Number of runs and termination criterion: Each algorithm is executed independently for 20 runs on every test instance, and the maximum function evaluations (MFEs) is used as the terminal criterion of each run. More details are presented in Table 1.

(3) Significance test: To evaluate the statistical significance of the results obtained from DAEA and the other five compared algorithms, the Wilcoxon rank sum test is utilized at a significance level of 0.05. In this test, we use the symbols "+", "-" and "=" to signify that the results obtained by a compared algorithm are significantly better than, worse than, and similar to 3DEA, respectively.

(4) Parameters of genetic operators: SBX [50] and PM [51] are used in all algorithms. For SBX [50], the crossover probability $p_c = 1.0$ and the distribution index is set to $\eta_c = 20$ are used in all compared algorithms. For the polynomial mutation [51], the distribution index and the mutation probability are set to $\eta_m = 20$ and $p_m = 1$ for all compared algorithms.

(5) Parameter setting in each algorithm: The parameters used in all the compared algorithms remain unchanged from their original publications.

4. Experimental results and analysis

4.1. Performance comparisons on DTLZ test suite

Tables 2 and 3 present the statistical results of DAEA and five compared algorithms on the DTLZ test suite in terms of IGD and HV values, respectively, where the best results are highlighted in bold font.

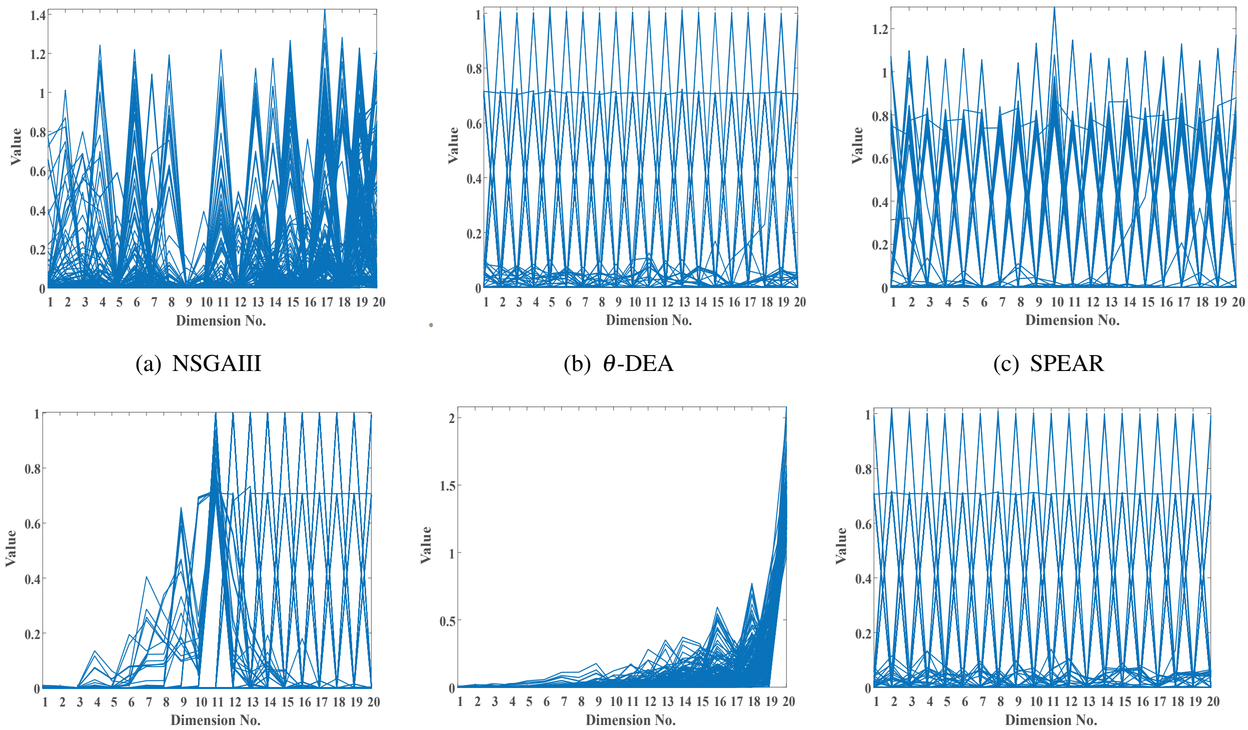
From the IGD results in Table 2, it can be observed that DAEA is significantly superior to NSGA-III, θ-DEA, SPEA/R, MaOEA-IGD and MaOEA-IT on 26, 18, 35, 26 and 35 test instances, respectively. Analogously, it can be observed from the HV results in Table 3 that DAEA defeats NSGA-III, θ-DEA, SPEA/R, MaOEA-IGD and MaOEA-IT on 28, 18, 32, 20 and 35 test instances, respectively. In summary, our proposed 3DEA is superior to the other state-of-the-art algorithms.

Specifically, DTLZ1 is a multi-modal problem with the linear PF, which mainly challenges the algorithm ability to jump out of local optima. DAEA obtains the best results in dealing with this problem, followed by θ-DEA.

For DTLZ2 test problem, DAEA performs best among the six compared algorithms. To more intuitively show the results obtained by six algorithms, we plot the parallel coordinates of final solutions obtained by each algorithm on 20-objective DTLZ2 instance in Figure 4. From this figure, we can see that θ-DEA and DAEA have better comprehensive performance in terms of convergence and diversity, while the other algorithms fail to converge to the true PF.

Since its multimodal property, DTLZ3 is difficult to solve. From the IGD and HV values highlighted in Tables 2 and 3, we can see that our DAEA performs best for most instances with different number of objective except 16-objective instance, which are won by θ-DEA. Other four algorithms are defeated by DAEA. It may because the double association in DAEA maintains a good diversity of solutions.

DTLZ4 mainly test the ability of algorithms to maintain population diversity. From Tables 2 and 3 we can see that DAEA performs best for most instances with different number of objective except 8-objective instance in Tables 3 and 8-, 12-, 16-objective instances in Tables 2, which are won by MaOEA-IGD. Other four algorithms are defeated by DAEA.



(a) NSGAIII (b) θ-DEA (c) SPEAR

(d) MaOEAIGD (e) MaOEAIT (f) DAEA

Figure 4. Parallel coordinates of final solutions achieved by six algorithms on 20-objective DTLZ2.

DTLZ5 and DTLZ6 have the degenerated PFs, which are used to test the ability of algorithms to deal with MaOPs with degenerate PFs. On these two problems, the IGD values obtained by DAEA are best for all instances. For HV metric, MaOEA-IGD is more suitable for solving this kind of problems than other compared algorithms. DAEA is the second and defeats the other four competitors on all test instances, which may be caused by the effective fitness assignment proposed by DAEA, since their tacit cooperation enables the algorithm to balance convergence and diversity.

DTLZ7, as a kind of mixed (concave and convex), disconnected and multi-modal problems, is difficult to better balance the convergence and diversity for existing evolutionary algorithms. From Tables 2, we can see that NSGA-III achieves the best IGD result on 5-objective instance, θ-DEA achieves the best IGD results on 8-, and 12-objective instances, MaOEA-IGD has the best IGD statistical results on 16-, and 20-objective instances. DAEA is statistically similar to that obtained by θ-DEA. From the HV results in Table 3, it can be observed that NSGA-III performs best on 5-objective instance, while θ-DEA performing best HV on 8-, and 20-objective instances. For our DAEA, it performs best on 12-, and 16-objective instances. As for MaOEA-IT, it is defeated by its peer competitors.

4.2. Performance comparisons on WFG test suite

As evidenced by statistical results of the IGD and HV values summarized in Tables 4 and 5, it is clearly observed that our DAEA performs best compared with other five algorithms in most cases.

Table 2. Statistical results (mean and standard deviation) of the IGD values obtained by six algorithms for DTLZ problems.

Problem	M	NSGA-III	θ-DEA	SPEA/R	MaOEA-IGD	MaOEA-IT	DAEA
DTLZ1	5	2.5939e-1 (2.72e-1) −	4.0697e-1 (5.29e-1) −	7.6791e-2 (1.95e-2) −	1.2614e-1 (1.04e-1) −	1.6647e+0 (3.64e+0) −	4.9172e-2 (5.12e-4)
	8	1.2208e-1 (5.38e-2) ≈	1.0748e-1 (3.04e-2) ≈	1.6194e-1 (5.49e-2) −	9.0395e-2 (5.56e-3) +	1.0092e+1 (8.63e+0) −	1.1280e-1 (5.42e-2)
	12	1.6554e-1 (6.55e-2) −	1.3493e-1 (2.43e-2) −	6.6817e-1 (3.80e-1) −	1.2406e-1 (7.78e-3) −	9.9151e+0 (7.77e+0) −	1.1378e-1 (1.58e-2)
	16	1.3893e-1 (1.54e-2) −	1.3895e-1 (3.17e-2) ≈	9.4617e-1 (5.12e-1) −	1.6317e-1 (6.29e-2) −	9.3445e+0 (9.35e+0) −	1.3227e-1 (1.76e-2)
	20	1.7432e+0 (1.99e+0) ≈	5.6653e-1 (4.54e-1) ≈	8.9357e-1 (5.35e-1) −	2.9398e-1 (1.33e-1) ≈	1.4090e+1 (1.34e+1) −	2.4431e-1 (3.80e-2)
DTLZ2	5	1.8127e-1 (1.86e-2) −	1.7969e-1 (1.68e-2) −	1.6831e-1 (5.96e-4) −	1.6874e-1 (5.94e-4) −	5.1210e-1 (7.07e-2) −	1.6511e-1 (5.12e-6)
	8	3.1519e-1 (1.01e-3) −	3.1259e-1 (9.18e-4) −	3.1929e-1 (1.07e-3) −	3.1158e-1 (1.51e-3) ≈	7.3607e-1 (5.95e-2) −	3.1163e-1 (3.69e-4)
	12	5.0400e-1 (1.01e-2) −	4.9742e-1 (2.56e-4) −	5.1908e-1 (3.80e-3) −	4.8895e-1 (1.64e-2) +	9.2191e-1 (3.78e-2) −	4.9614e-1 (5.06e-4)
	16	5.6841e-1 (3.19e-2) −	5.3807e-1 (6.89e-4) ≈	5.4596e-1 (1.73e-3) −	6.0157e-1 (4.86e-2) −	1.0911e+0 (8.87e-2) −	5.3813e-1 (1.32e-3)
	20	8.3085e-1 (1.07e-1) −	6.4059e-1 (2.21e-2) −	6.5141e-1 (9.30e-3) −	8.1877e-1 (1.52e-1) −	1.3304e+0 (3.81e-2) −	6.2279e-1 (1.04e-4)
DTLZ3	5	1.3262e+1 (1.55e+1) −	9.6356e+0 (1.16e+1) −	1.1833e+0 (1.23e+0) −	9.9392e+0 (3.28e+0) −	1.8904e+1 (1.84e+1) −	1.6502e-1 (2.89e-4)
	8	1.6894e+0 (1.40e+0) −	4.8179e-1 (9.37e-2) ≈	2.0831e+1 (7.57e+0) −	8.4596e+0 (3.66e+0) −	2.7048e+2 (6.44e+1) −	4.6603e-1 (9.06e-2)
	12	5.7464e+0 (4.34e+0) −	7.2829e-1 (3.12e-1) ≈	9.6636e+1 (2.73e+1) −	3.5768e+0 (1.91e+0) −	2.8098e+2 (8.68e+1) −	6.1707e-1 (7.00e-2)
	16	5.4120e+0 (2.55e+0) −	6.3162e-1 (6.52e-2) ≈	2.6765e+1 (1.23e+1) −	2.6480e+0 (1.48e+0) −	2.6151e+2 (8.01e+1) −	6.6507e-1 (1.98e-1)
	20	1.5099e+2 (7.61e+1) −	2.8344e+1 (3.24e+1) −	1.8933e+2 (7.79e+1) −	2.6276e+0 (1.88e+0) −	2.5575e+2 (1.19e+2) −	7.1357e-1 (1.38e-1)
DTLZ4	5	2.0617e-1 (8.56e-2) −	1.8208e-1 (2.00e-2) ≈	1.6851e-1 (8.61e-4) −	1.6815e-1 (1.64e-4) −	9.5214e-1 (9.19e-2) −	1.6507e-1 (2.02e-5)
	8	3.1512e-1 (6.45e-4) −	3.1405e-1 (6.26e-4) −	3.2456e-1 (1.54e-3) −	3.0971e-1 (3.98e-4) +	1.0085e+0 (7.19e-2) −	3.1308e-1 (3.76e-4)
	12	4.9863e-1 (3.71e-4) −	4.9799e-1 (1.46e-4) −	5.3197e-1 (4.58e-3) −	4.9499e-1 (9.77e-3) +	1.0705e+0 (5.15e-2) −	4.9762e-1 (2.08e-4)
	16	5.7109e-1 (4.60e-2) ≈	5.4595e-1 (2.22e-3) ≈	5.8503e-1 (1.08e-2) −	5.9435e-1 (9.45e-3) −	1.3666e+0 (4.99e-2) −	5.4685e-1 (2.58e-3)
	20	7.4141e-1 (1.20e-1) −	6.3718e-1 (1.68e-2) −	6.4541e-1 (5.53e-3) −	6.2891e-1 (5.94e-3) −	1.0677e+0 (9.18e-2) −	6.2286e-1 (1.27e-4)
DTLZ5	5	8.4498e-2 (2.44e-2) ≈	2.9769e-1 (1.02e-1) −	1.8032e-1 (2.86e-2) −	5.5991e-1 (1.45e-1) −	3.3280e-1 (4.75e-2) −	6.6476e-2 (2.31e-2)
	8	2.6381e-1 (4.85e-2) −	1.3932e-1 (1.69e-2) −	7.3947e-1 (1.00e-1) −	4.1669e-1 (1.76e-1) −	3.2769e-1 (1.81e-1) −	8.6543e-2 (1.33e-2)
	12	2.9711e-1 (6.86e-2) −	1.9821e-1 (2.33e-2) −	1.0233e+0 (1.60e-1) −	4.4597e-1 (1.91e-1) −	3.8060e-1 (2.03e-1) −	1.2661e-1 (1.30e-2)
	16	3.9936e-1 (1.24e-1) −	1.9319e-1 (4.10e-2) −	1.0488e+0 (2.28e-1) −	2.5775e-1 (1.06e-1) −	4.0334e-1 (5.69e-2) −	1.3472e-1 (2.64e-2)
	20	9.0180e-1 (4.28e-1) −	1.8029e-1 (4.05e-2) ≈	1.0145e+0 (3.20e-1) −	3.2690e-1 (9.62e-2) −	4.4991e-1 (7.68e-2) −	1.6519e-1 (2.38e-2)
DTLZ6	5	2.0024e-1 (4.67e-2) −	1.7203e-1 (4.57e-2) ≈	3.1990e-1 (1.34e-1) −	6.1062e-1 (5.22e-2) −	8.2900e+0 (1.97e-1) −	1.3102e-1 (5.06e-2)
	8	6.3033e-1 (3.38e-1) −	2.5260e-1 (3.30e-2) −	5.9415e+0 (8.79e-1) −	6.5391e-1 (1.13e-1) −	8.4818e+0 (1.57e-1) −	1.5568e-1 (4.30e-2)
	12	1.4339e+0 (4.92e-1) −	2.3895e-1 (4.76e-2) −	8.2463e+0 (6.18e-1) −	6.4578e-1 (1.43e-1) −	8.5341e+0 (1.17e-1) −	1.8940e-1 (3.64e-2)
	16	1.7275e+0 (1.12e+0) −	2.6499e-1 (5.65e-2) −	4.0255e+0 (8.32e-1) −	3.6858e-1 (1.23e-5) −	8.2822e+0 (1.87e-1) −	1.8479e-1 (4.05e-2)
	20	7.0043e+0 (2.11e+0) −	2.3670e-1 (5.46e-2) ≈	8.8242e+0 (5.64e-1) −	3.9494e-1 (9.58e-5) −	8.3734e+0 (3.38e-1) −	2.1291e-1 (4.95e-2)
DTLZ7	5	2.8518e-1 (7.36e-3) +	3.0960e-1 (2.99e-2) ≈	3.5710e-1 (5.41e-3) −	6.7529e-1 (1.33e-3) −	3.3571e+0 (2.40e+0) −	3.0883e-1 (1.43e-2)
	8	6.8878e-1 (2.40e-2) ≈	6.4153e-1 (4.85e-2) +	1.2164e+0 (1.28e-1) −	1.1487e+0 (3.51e-2) −	2.2976e+1 (1.62e+0) −	6.8070e-1 (1.73e-2)
	12	1.7824e+0 (3.65e-1) ≈	1.3153e+0 (1.75e-1) ≈	3.4219e+0 (3.90e-2) −	1.6631e+0 (3.01e-2) ≈	3.8619e+1 (1.95e+0) −	1.8297e+0 (6.65e-1)
	16	5.2080e+0 (1.19e+0) ≈	4.0932e+0 (6.50e-1) +	1.6952e+1 (8.84e+0) −	2.2296e+0 (7.73e-2) +	6.1326e+1 (4.11e+0) −	5.9838e+0 (1.15e+0)
	20	8.7893e+0 (1.12e+0) ≈	8.7351e+0 (6.28e-1) ≈	1.3015e+1 (1.89e-2) −	3.6207e+0 (1.78e-1) +	8.1247e+1 (2.32e+0) −	8.9963e+0 (1.76e+0)
+/−/≈		1/26/8	2/18/15	0/35/0	6/26/3	0/35/0	

Table 3. Statistical results (mean and standard deviation) of the HV values obtained by six algorithms for DTLZ problems.

Problem	M	NSGA-III	θ-DEA	SPEA/R	MaOEA-IGD	MaOEA-IT	DAEA
DTLZ1	5	5.9949e-1 (4.68e-1) ≈	5.6494e-1 (4.86e-1) ≈	9.6160e-1 (1.51e-2) ≈	8.1638e-1 (2.26e-1) ≈	3.5251e-1 (3.88e-1) −	9.5909e-1 (6.20e-3)
	8	9.5582e-1 (1.28e-1) +	9.8378e-1 (4.51e-2) +	8.6708e-1 (2.23e-1) ≈	9.9635e-1 (1.48e-3) +	3.7079e-3 (1.17e-2) −	9.2375e-1 (7.80e-2)
	12	9.3936e-1 (1.92e-1) −	9.9746e-1 (5.76e-3) +	1.6215e-1 (2.52e-1) −	9.9482e-1 (2.70e-3) +	0.0000e+0 (0.00e+0) −	9.6511e-1 (1.19e-2)
	16	9.9777e-1 (5.63e-3) +	9.9797e-1 (2.83e-3) ≈	3.1669e-2 (6.60e-2) −	9.1608e-1 (1.76e-1) −	0.0000e+0 (0.00e+0) −	9.9690e-1 (3.86e-3)
	20	4.7100e-1 (5.02e-1) ≈	5.3503e-1 (5.08e-1) ≈	9.1563e-2 (2.12e-1) −	7.4270e-1 (3.66e-1) ≈	3.3344e-2 (1.11e-1) −	9.3660e-1 (6.05e-2)
DTLZ2	5	7.9091e-1 (2.49e-2) −	7.9598e-1 (1.89e-2) −	8.1020e-1 (1.07e-3) −	8.1277e-1 (3.47e-4) ≈	1.8835e-1 (7.50e-2) −	8.1263e-1 (3.44e-4)
	8	9.4308e-1 (4.33e-4) −	9.4553e-1 (2.62e-4) −	9.3853e-1 (7.94e-4) −	9.4603e-1 (2.69e-4) ≈	2.1057e-1 (8.97e-2) −	9.4605e-1 (3.46e-4)
	12	9.8242e-1 (5.76e-3) −	9.8708e-1 (1.85e-4) −	9.7130e-1 (3.18e-3) −	9.8428e-1 (4.65e-3) −	1.8514e-1 (4.91e-2) −	9.8753e-1 (9.74e-5)
	16	9.8635e-1 (6.10e-3) −	9.9317e-1 (1.53e-4) −	9.6167e-1 (1.34e-2) −	9.8242e-1 (1.54e-2) −	1.1518e-1 (6.96e-2) −	9.9361e-1 (9.60e-5)
	20	6.8201e-1 (2.45e-1) −	9.3663e-1 (7.23e-2) −	9.9023e-1 (4.22e-3) −	8.4570e-1 (2.13e-1) −	5.2857e-2 (2.22e-2) −	9.9864e-1 (3.98e-5)
DTLZ3	5	4.3308e-1 (4.15e-1) −	4.3787e-1 (4.19e-1) −	1.9116e-1 (2.02e-1) −	0.0000e+0 (0.00e+0) −	0.0000e+0 (0.00e+0) −	8.0848e-1 (2.53e-3)
	8	1.9669e-1 (2.74e-1) −	6.7752e-1 (1.52e-1) ≈	0.0000e+0 (0.00e+0) −	0.0000e+0 (0.00e+0) −	0.0000e+0 (0.00e+0) −	7.4043e-1 (1.15e-1)
	12	1.0702e-3 (3.55e-3) −	6.3708e-1 (3.48e-1) ≈	0.0000e+0 (0.00e+0) −	7.1293e-3 (2.36e-2) −	0.0000e+0 (0.00e+0) −	7.3553e-1 (1.64e-1)
	16	0.0000e+0 (0.00e+0) −	9.0148e-1 (1.21e-1) ≈	0.0000e+0 (0.00e+0) −	8.0334e-2 (2.07e-1) −	0.0000e+0 (0.00e+0) −	8.5744e-1 (2.73e-1)
	20	0.0000e+0 (0.00e+0) −	3.6235e-1 (4.88e-1) −	0.0000e+0 (0.00e+0) −	2.1848e-2 (4.25e-2) −	0.0000e+0 (0.00e+0) −	8.6036e-1 (2.26e-1)
DTLZ4	5	7.7443e-1 (5.90e-2) −	7.9614e-1 (1.91e-2) ≈	8.0859e-1 (9.61e-4) −	8.1256e-1 (3.02e-4) ≈	1.1783e-1 (1.04e-1) −	8.1259e-1 (2.19e-4)
	8	9.4395e-1 (5.00e-4) −	9.4595e-1 (1.34e-4) −	9.3023e-1 (2.58e-3) −	9.4684e-1 (3.58e-4) +	2.8299e-2 (2.30e-2) −	9.4640e-1 (1.76e-4)
	12	9.8702e-1 (1.92e-4) −	9.8770e-1 (1.53e-4) −	9.6120e-1 (4.11e-3) −	9.8774e-1 (9.63e-4) −	6.1729e-2 (3.29e-2) −	9.8784e-1 (9.74e-5)
	16	9.8899e-1 (1.29e-2) −	9.9412e-1 (6.10e-5) −	9.5800e-1 (1.37e-2) −	9.9286e-1 (2.43e-3) −	1.7976e-2 (2.03e-2) −	9.9438e-1 (9.76e-5)
	20	7.8446e-1 (3.19e-1) −	9.9354e-1 (6.39e-3) ≈	9.9254e-1 (2.06e-3) −	9.9831e-1 (3.27e-4) −	1.1442e-1 (1.16e-1) −	9.9871e-1 (2.52e-5)
DTLZ5	5	1.1219e-1 (8.37e-3) −	9.8779e-2 (6.89e-3) −	3.8972e-2 (1.89e-2) −	8.8702e-2 (2.94e-2) −	1.8887e-3 (3.09e-3) −	1.1800e-1 (5.90e-3)
	8	5.4139e-2 (2.22e-2) −	9.0568e-2 (2.04e-3) −	0.0000e+0 (0.00e+0) −	9.1836e-2 (3.58e-4) −	1.9136e-3 (3.82e-3) −	9.6855e-2 (2.28e-3)
	12	6.3693e-2 (1.40e-2) −	8.7445e-2 (2.17e-3) −	0.0000e+0 (0.00e+0) −	9.1062e-2 (4.44e-4) ≈	1.0098e-3 (2.34e-3) −	9.1000e-2 (1.35e-3)
	16	5.2239e-2 (2.09e-2) −	8.9112e-2 (1.04e-3) −	0.0000e+0 (0.00e+0) −	9.3627e-2 (3.10e-4) +	3.1627e-3 (8.83e-3) −	9.0713e-2 (5.80e-4)
	20	7.6891e-3 (2.55e-2) −	9.0150e-2 (4.09e-4) ≈	1.2840e-9 (4.26e-9) −	9.1484e-2 (2.26e-4) +	1.2256e-3 (4.06e-3) −	9.0141e-2 (8.60e-4)
DTLZ6	5	9.7594e-2 (1.23e-2) −	9.6233e-2 (5.70e-3) −	1.0052e-2 (1.71e-2) −	8.8249e-2 (2.92e-2) −	0.0000e+0 (0.00e+0) −	1.0826e-1 (8.21e-3)
	8	1.6583e-2 (3.69e-2) −	9.0859e-2 (2.69e-4) −	0.0000e+0 (0.00e+0) −	9.2213e-2 (4.89e-5) ≈	0.0000e+0 (0.00e+0) −	9.1802e-2 (8.51e-4)
	12	0.0000e+0 (0.00e+0) −	9.0821e-2 (1.85e-4) ≈	0.0000e+0 (0.00e+0) −	8.3140e-2 (2.76e-2) ≈	0.0000e+0 (0.00e+0) −	9.1133e-2 (6.37e-4)
	16	8.2325e-3 (2.73e-2) −	9.0967e-2 (2.40e-4) ≈	0.0000e+0 (0.00e+0) −	9.3507e-2 (7.93e-5) +	0.0000e+0 (0.00e+0) −	9.1176e-2 (2.83e-4)
	20	0.0000e+0 (0.00e+0) −	9.0827e-2 (3.74e-4) ≈	0.0000e+0 (0.00e+0) −	9.1335e-2 (5.88e-5) +	0.0000e+0 (0.00e+0) −	9.1046e-2 (3.51e-4)
DTLZ7	5	2.5196e-1 (4.93e-3) +	2.1515e-1 (1.22e-2) ≈	2.4822e-1 (2.04e-3) +	1.8324e-1 (5.95e-4) −	1.9654e-2 (2.79e-2) −	2.1828e-1 (1.15e-2)
	8	1.8644e-1 (4.11e-3) ≈	1.8889e-1 (6.62e-3) ≈	1.5418e-1 (1.24e-2) −	3.5097e-2 (1.12e-2) −	0.0000e+0 (0.00e+0) −	1.8876e-1 (1.74e-2)
	12	1.3951e-1 (3.61e-3) −	1.6784e-1 (6.22e-3) −	1.1534e-1 (7.79e-3) −	1.5239e-3 (7.65e-4) −	0.0000e+0 (0.00e+0) −	1.8170e-1 (6.08e-3)
	16	5.3692e-2 (1.18e-2) −	8.6784e-2 (1.72e-2) −	2.5694e-2 (3.81e-2) −	1.8263e-4 (1.80e-4) −	0.0000e+0 (0.00e+0) −	1.3647e-1 (1.54e-2)
	20	1.2134e-1 (8.00e-3) ≈	1.4454e-1 (3.63e-3) +	1.1055e-1 (7.57e-3) −	4.5248e-5 (1.04e-4) −	0.0000e+0 (0.00e+0) −	1.2897e-1 (9.42e-3)
+/−/≈		3/28/4	3/18/14	1/32/2	7/20/8	0/35/0	

According to the statistic results of Wilcoxon rank sum test in the last line of Table 4, we can see that, among the 45 test instances, the IGD results obtained by DAEA are significantly better than those obtained by NSGA-III, θ-DEA, SPEA/R, MaOEA-IGD and MaOEA-IT in most test instances (in 25, 19, 34, 39 and 45 test instances, respectively), while the IGD results obtained by NSGA-III, θ-DEA, SPEA/R, MaOEA-IGD and MaOEA-IT are significantly better than those obtained by DAEA in only a few test instances (in 10, 9, 7, 3 and 0 test instances, respectively). According to the statistic results in Table 5, it can be concluded that DAEA defeats its competitors on most test instances. To be specific, DAEA outperforms NSGA-III, θ-DEA, SPEA/R, MaOEA-IGD and MaOEA-IT on 25, 21, 22, 39 and 42 test instances out of 45 test instances, respectively. Moreover, NSGA-III, θ-DEA, SPEA/R, MaOEA-IGD and MaOEA-IT outperform DAEA in terms of HV results on a limited number of test instances (8, 6, 17, 1 and 0 test instances, respectively). In what follows, we will delve into detailed discussions regarding the experimental results.

WFG1 has a flat bias and a combination of convex and concave structures in its PF. From Table 4, we can observe that DAEA obtains the best IGD values on 5-objective instance, θ-DEA achieves the best performance on 8-, 12- and 16-objective instances in terms of IGD values, while NSGA-III wins on 20-objective instance. In terms of the HV metric, SPEA/R exhibits the best performance on 5- and 8-objective instances, θ-DEA shows the best performance on 12- and 16-objective instances, while NSGA-III demonstrates the best performance on the 20-objective instance. As for MaOEA-IGD and MaOEA-IT, both of them perform worse for this problem.

WFG2 consists of multiple convex PF segments that are not connected, and its variables cannot be separated.For this problem, NSGA-III and SPEA/R are superior to other four algorithms, followed by DAEA and θ-DEA.

WFG3 is a variant of WFG2 where its PF is both linear and degenerate, which poses a big challenge to reference vector-based algorithms. Our proposed DAEA outperforms other methods on 8-, 12-, 16- and 20-objective test instances, while the best performance on 5-objective instance is achieved by θ-DEA. The results verifies the effectiveness of DAEA on degenerate problems.

The problems WFG4 to WFG9 share the same hyper-ellipse PF shape in the objective space, but their characteristics differ in the decision space. To be specific, WFG4 has the property of multimodality with "hill sizes", it can be used to test whether a algorithm has the ability to jump out of local optima.

DAEA shows distinct advantage in most instances except the 8-objective instance, which is defeated by SPEA/R. WFG5 is a deceptive problem, the IGD metric shows that DAEA performs best on most test instances, while NSGA-III and θ-DEA perform similar to DAEA on 8-, 12- and 16-objective instances. The HV metric shows that SPEA/R and DAEA are superior to the other four algorithms. WFG6 is a nonseparable and reduced problem, DAEA obtains the best IGD values on all test instances except 12-objective instance which is obtained by NSGA-III, which indicates that our proposed DAEA has promising performance of diversity and convergence, especially for the optimization problem with a larger number of objectives. WFG7-WFG9 all introduce some biases to challenge algorithms' diversity, DAEA performs best on these problems, followed by SPEA/R, θ-DEA and NSGA-III, MaOEA-IGD and MaOEA-IT have poor performance for these problems.

Table 4. Statistical results (mean and standard deviation) of the IGD values obtained by six algorithms for WFG problems.

Problem	M	NSGA-III	θ-DEA	SPEA/R	MaOEA-IGD	MaOEA-IT	DAEA
WFG1	5	9.0860e-1 (7.82e-2) −	7.0773e-1 (4.83e-2) −	7.4525e-1 (8.72e-2) −	4.6474e+0 (1.06e+0) −	2.8366e+0 (7.36e-1) −	6.7688e-1 (3.88e-2)
	8	1.7059e+0 (5.89e-2) −	1.3079e+0 (5.73e-2) ≈	1.4501e+0 (1.00e-1) −	6.4321e+0 (1.69e+0) −	3.0898e+0 (1.34e-1) −	1.3395e+0 (7.99e-2)
	12	2.1941e+0 (1.44e-1) −	1.3844e+0 (9.80e-2) +	1.9156e+0 (1.03e-1) −	7.0021e+0 (3.41e+0) −	3.6929e+0 (2.42e-2) −	1.4882e+0 (1.14e-1)
	16	2.0197e+0 (7.25e-2) −	1.6169e+0 (3.48e-2) +	2.3633e+0 (8.01e-2) −	7.1334e+0 (4.49e+0) −	4.5738e+0 (1.80e-2) −	1.6829e+0 (3.76e-2)
	20	3.4351e+0 (2.57e-1) +	4.2830e+0 (1.46e-1) +	4.6873e+0 (6.82e-2) ≈	6.7913e+0 (2.81e+0) −	5.7861e+0 (7.59e-2) −	4.6584e+0 (1.17e-1)
WFG2	5	3.8973e-1 (1.54e-3) +	3.8845e-1 (1.48e-3) +	3.9693e-1 (2.64e-3) +	1.9012e+0 (3.29e-1) −	1.4237e+0 (2.09e-1) −	4.2582e-1 (1.41e-2)
	8	9.3351e-1 (2.14e-1) ≈	9.6161e-1 (8.53e-2) −	8.9315e-1 (1.03e-2) +	2.2531e+0 (3.34e-1) −	2.9767e+0 (1.61e+0) −	9.9827e-1 (3.77e-2)
	12	1.2696e+0 (1.70e-1) +	2.1502e+0 (2.02e-1) ≈	1.2828e+0 (8.79e-2) +	2.3080e+0 (8.43e-1) ≈	5.5055e+0 (2.65e+0) −	2.1745e+0 (3.42e-1)
	16	1.6433e+0 (1.15e-1) +	4.4643e+0 (1.51e+0) −	1.5212e+0 (2.28e-2) +	2.4133e+0 (6.44e-1) +	6.3191e+0 (1.03e+0) −	3.7928e+0 (1.24e+0)
	20	3.9551e+0 (1.55e-1) +	7.7340e+0 (1.55e+0) −	3.4213e+0 (1.63e-1) +	5.2726e+0 (1.03e+0) +	2.4236e+1 (1.83e+0) −	6.8658e+0 (1.81e+0)
WFG3	5	5.1177e-1 (5.04e-2) +	4.4611e-1 (6.60e-2) +	4.5680e-1 (3.80e-2) +	5.4368e+0 (2.46e-2) −	1.7082e+0 (5.97e-1) −	5.8462e-1 (5.76e-2)
	8	1.0298e+0 (4.17e-1) ≈	8.7387e-1 (2.05e-1) −	1.2242e+0 (1.43e-1) −	4.2277e+0 (3.95e+0) −	1.5001e+0 (9.53e-1) −	7.8143e-1 (1.68e-1)
	12	1.0867e+0 (1.49e-1) −	1.0573e+0 (8.18e-2) −	2.2882e+0 (6.21e-2) −	2.5972e+0 (2.22e-1) −	2.9690e+0 (2.44e+0) −	5.9947e-1 (7.17e-2)
	16	2.7167e+0 (1.00e+0) −	2.1204e+0 (2.93e-1) −	4.4535e+0 (3.47e-1) −	6.7846e+0 (3.18e+0) −	4.3294e+0 (7.41e-1) −	1.6384e+0 (1.86e-1)
	20	8.5546e+0 (2.44e+0) −	2.8146e+0 (4.05e-1) −	5.6041e+0 (2.23e-1) −	7.5147e+0 (3.11e-1) −	9.7097e+0 (1.17e+0) −	2.2681e+0 (2.65e-1)
WFG4	5	9.6293e-1 (1.65e-3) +	9.6363e-1 (1.43e-3) ≈	9.7338e-1 (4.27e-3) −	6.1436e+0 (6.07e-1) −	2.1584e+0 (1.67e-1) −	9.6441e-1 (1.74e-3)
	8	2.7770e+0 (1.04e-2) ≈	2.7801e+0 (6.43e-3) ≈	2.7987e+0 (7.92e-3) −	9.7982e+0 (9.25e-2) −	7.2906e+0 (2.31e+0) −	2.7814e+0 (9.52e-3)
	12	5.7568e+0 (2.27e-2) ≈	5.7552e+0 (1.59e-2) ≈	5.8076e+0 (1.45e-2) −	1.2124e+1 (3.63e+0) −	1.3653e+1 (2.55e+0) −	5.7653e+0 (1.99e-2)
	16	8.8737e+0 (1.16e-1) ≈	8.7811e+0 (3.38e-2) +	8.9581e+0 (1.04e-1) −	2.1252e+1 (7.15e+0) −	2.2709e+1 (3.91e+0) −	8.8464e+0 (5.08e-2)
	20	1.2862e+1 (8.25e-1) −	1.1460e+1 (2.17e-2) ≈	1.1525e+1 (3.00e-2) −	3.0573e+1 (1.20e+1) −	3.9563e+1 (3.36e-1) −	1.1455e+1 (2.06e-2)
WFG5	5	9.5729e-1 (8.40e-4) −	9.5710e-1 (6.07e-4) −	9.6656e-1 (3.80e-3) −	6.8193e+0 (1.15e+0) −	1.9890e+0 (3.46e-1) −	9.5630e-1 (7.22e-4)
	8	2.7514e+0 (7.69e-3) ≈	2.7538e+0 (9.49e-3) ≈	2.7817e+0 (6.58e-3) −	9.0835e+0 (5.64e+0) −	4.8627e+0 (2.42e-1) −	2.7535e+0 (1.12e-2)
	12	5.5840e+0 (1.54e-2) ≈	5.5914e+0 (1.94e-2) ≈	5.7317e+0 (9.39e-3) −	9.3416e+0 (7.30e+0) −	9.3790e+0 (3.12e-1) −	5.5925e+0 (2.57e-2)
	16	8.5294e+0 (1.45e-1) −	8.3110e+0 (1.08e-1) ≈	9.2334e+0 (4.49e-2) −	2.8622e+1 (8.40e+0) −	1.7289e+1 (1.72e+0) −	8.3364e+0 (1.31e-1)
	20	1.1787e+1 (4.94e-1) −	1.1308e+1 (4.41e-2) −	1.1526e+1 (9.34e-3) −	3.0995e+1 (1.59e+1) ≈	3.4108e+1 (2.63e+0) −	1.1278e+1 (4.28e-2)
WFG6	5	9.6048e-1 (1.06e-3) ≈	9.5995e-1 (9.14e-4) ≈	9.7057e-1 (5.39e-3) −	5.8291e+0 (6.97e-1) −	2.1748e+0 (2.99e-1) −	9.5985e-1 (7.31e-4)
	8	2.8035e+0 (1.12e-2) −	2.8044e+0 (1.19e-2) −	2.8124e+0 (1.06e-2) −	8.6983e+0 (3.27e+0) −	5.8344e+0 (1.17e+0) −	2.7950e+0 (6.35e-3)
	12	5.7527e+0 (1.11e-2) ≈	5.7657e+0 (1.03e-2) −	5.8085e+0 (1.05e-2) −	1.4878e+1 (5.71e+0) −	1.0701e+1 (1.90e+0) −	5.7616e+0 (1.15e-2)
	16	8.8778e+0 (5.95e-2) −	8.8463e+0 (3.46e-2) −	9.2482e+0 (5.78e-2) −	2.1590e+1 (8.60e+0) −	1.8797e+1 (1.52e+0) −	8.8127e+0 (3.19e-2)
	20	1.4135e+1 (9.32e-1) −	1.1518e+1 (8.94e-3) ≈	1.1591e+1 (4.10e-2) −	2.2328e+1 (1.43e+1) ≈	3.4559e+1 (2.65e+0) −	1.1518e+1 (1.65e-2)
WFG7	5	9.6389e-1 (8.51e-4) +	9.6456e-1 (5.36e-4) +	9.7069e-1 (2.64e-3) −	5.7817e+0 (5.93e-1) −	1.8108e+0 (2.32e-1) −	9.6527e-1 (2.92e-4)
	8	2.8038e+0 (1.26e-2) −	2.8010e+0 (9.66e-3) −	2.8009e+0 (9.35e-3) ≈	8.2638e+0 (2.11e+0) −	5.2291e+0 (4.79e-1) −	2.7935e+0 (9.16e-3)
	12	5.7146e+0 (2.99e-2) +	5.7563e+0 (1.14e-2) −	5.7814e+0 (1.05e-2) −	9.3477e+0 (5.43e+0) −	1.0496e+1 (8.48e-1) −	5.7550e+0 (1.73e-2)
	16	8.7667e+0 (9.37e-2) −	8.7622e+0 (4.30e-2) −	9.2655e+0 (4.24e-2) −	2.0412e+1 (9.01e+0) −	1.9445e+1 (3.14e+0) −	8.3971e+0 (2.05e-1)
	20	1.4669e+1 (4.87e-1) −	1.1558e+1 (2.41e-2) ≈	1.1768e+1 (4.43e-2) −	2.7680e+1 (1.21e+1) −	3.9596e+1 (1.65e+0) −	1.1580e+1 (3.44e-2)
WFG8	5	9.5004e-1 (8.57e-4) −	9.4920e-1 (9.63e-4) ≈	9.6462e-1 (4.62e-3) −	4.6258e+0 (3.62e-1) −	2.1709e+0 (3.13e-1) −	9.4853e-1 (1.11e-3)
	8	2.7141e+0 (8.39e-3) −	2.7130e+0 (8.80e-3) ≈	2.8204e+0 (6.40e-3) −	1.0217e+1 (2.52e+0) −	5.5551e+0 (7.91e-1) −	2.7054e+0 (1.11e-2)
	12	5.5299e+0 (3.51e-1) ≈	5.3176e+0 (7.94e-2) ≈	5.7568e+0 (2.54e-2) −	1.5460e+1 (6.91e-1) −	1.1226e+1 (1.70e+0) −	5.3351e+0 (7.01e-2)
	16	9.0385e+0 (6.20e-1) +	9.0242e+0 (4.04e-1) +	9.4247e+0 (4.47e-2) ≈	2.5601e+1 (2.93e+0) −	1.9715e+1 (2.24e+0) −	9.5431e+0 (5.18e-1)
	20	1.4608e+1 (2.35e+0) −	1.1648e+1 (3.49e-1) +	1.1616e+1 (6.93e-2) +	3.3699e+1 (2.31e+0) −	3.5216e+1 (2.19e+0) −	1.2903e+1 (8.71e-1)
WFG9	5	9.4127e-1 (6.97e-3) −	9.3471e-1 (3.38e-3) ≈	9.5208e-1 (7.89e-3) −	4.0779e+0 (9.91e-1) −	1.9805e+0 (1.15e-1) −	9.3386e-1 (3.24e-3)
	8	2.6844e+0 (1.67e-2) −	2.6853e+0 (1.29e-2) −	2.7563e+0 (6.57e-3) −	3.9176e+0 (1.65e+0) −	5.2374e+0 (2.35e-1) −	2.6671e+0 (2.26e-2)
	12	5.2795e+0 (4.72e-2) −	5.2667e+0 (4.37e-2) −	5.5806e+0 (2.60e-2) −	6.0522e+0 (3.64e-1) −	1.0139e+1 (4.50e-1) −	5.2529e+0 (3.20e-2)
	16	8.6590e+0 (1.14e-1) −	8.1217e+0 (1.81e-1) −	9.2093e+0 (1.07e-1) −	1.1587e+1 (6.25e+0) −	1.8142e+1 (8.80e-1) −	8.1031e+0 (2.17e-1)
	20	1.3567e+1 (9.40e-1) −	1.1640e+1 (2.18e-1) ≈	1.1821e+1 (5.08e-2) ≈	1.0694e+1 (4.74e-1) +	3.7661e+1 (1.87e+0) −	1.1712e+1 (2.21e-1)
+/−/≈		10/25/10	9/19/17	7/34/4	3/39/3	0/45/0	

Table 5. Statistical results (mean and standard deviation) of the HV values obtained by six algorithms for WFG problems.

Problem	M	NSGA-III	θ-DEA	SPEA/R	MaOEA-IGD	MaOEA-IT	DAEA
WFG1	5	7.0107e-1 (1.02e-1) ≈	7.7825e-1 (9.35e-2) ≈	8.1028e-1 (8.60e-2) +	1.2226e-1 (3.42e-2) −	8.7614e-2 (4.27e-2) −	7.1947e-1 (1.80e-2)
	8	5.1893e-1 (8.91e-2) ≈	6.9441e-1 (1.21e-1) +	7.3242e-1 (9.99e-2) +	3.0458e-1 (1.28e-1) −	1.6711e-2 (1.87e-2) −	5.9573e-1 (2.84e-2)
	12	5.3257e-1 (1.45e-1) −	8.6074e-1 (1.06e-1) +	8.3271e-1 (1.04e-1) +	4.4712e-1 (1.18e-1) −	3.5816e-2 (9.90e-3) −	7.5827e-1 (5.28e-2)
	16	9.5696e-1 (7.48e-2) ≈	9.9766e-1 (8.83e-4) +	9.8813e-1 (3.59e-2) −	6.5300e-1 (1.11e-1) −	7.1142e-2 (2.14e-2) −	9.9629e-1 (1.15e-3)
	20	9.9984e-1 (1.86e-4) +	9.9767e-1 (7.70e-4) ≈	9.9278e-1 (4.17e-3) −	7.6164e-1 (2.52e-1) −	9.8757e-2 (2.30e-2) −	9.9752e-1 (6.92e-4)
WFG2	5	9.8629e-1 (2.09e-3) +	9.8701e-1 (1.88e-3) +	9.8773e-1 (1.10e-3) +	8.6568e-1 (4.97e-2) −	6.0691e-1 (2.53e-2) −	9.6857e-1 (7.54e-3)
	8	9.8333e-1 (8.70e-3) +	9.7477e-1 (5.89e-3) ≈	9.7941e-1 (2.47e-3) +	9.0472e-1 (4.14e-2) −	5.6365e-1 (7.70e-2) −	9.6927e-1 (8.13e-3)
	12	9.8939e-1 (3.90e-3) +	9.5744e-1 (1.21e-2) ≈	9.8212e-1 (2.86e-3) +	9.4132e-1 (5.78e-2) ≈	5.4702e-1 (7.77e-2) −	9.5439e-1 (1.38e-2)
	16	9.9351e-1 (2.04e-3) +	9.1166e-1 (7.18e-2) −	9.8544e-1 (4.81e-3) +	9.5793e-1 (1.62e-2) ≈	5.5495e-1 (4.88e-2) −	9.3612e-1 (5.75e-2)
	20	9.9682e-1 (1.24e-3) +	8.4618e-1 (6.44e-2) −	9.9464e-1 (2.57e-3) +	9.6847e-1 (3.18e-2) +	3.1055e-1 (2.94e-2) −	8.9782e-1 (8.31e-2)
WFG3	5	1.0963e-1 (1.58e-2) +	1.7278e-1 (1.83e-2) +	1.6477e-1 (2.50e-2) +	5.7829e-2 (5.63e-3) −	0.0000e+0 (0.00e+0) −	8.0869e-2 (2.69e-2)
	8	0.0000e+0 (0.00e+0) −	3.8263e-2 (1.91e-2) −	7.6590e-4 (2.97e-3) −	0.0000e+0 (0.00e+0) −	0.0000e+0 (0.00e+0) −	1.2427e-1 (1.29e-2)
	12	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0)
	16	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0)
	20	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0)
WFG4	5	7.8210e-1 (2.15e-3) −	7.8568e-1 (2.68e-3) −	7.8774e-1 (2.36e-3) −	1.1655e-1 (4.75e-2) −	3.4332e-1 (1.16e-2) −	7.8849e-1 (2.00e-3)
	8	8.7806e-1 (4.08e-3) −	8.8580e-1 (3.08e-3) −	8.9565e-1 (3.07e-3) +	9.5410e-2 (2.13e-2) −	3.3561e-1 (9.94e-2) −	8.9203e-1 (3.05e-3)
	12	9.1894e-1 (9.94e-3) −	9.2273e-1 (6.00e-3) −	9.2695e-1 (5.90e-3) −	2.8894e-1 (3.07e-1) −	3.4707e-1 (7.63e-2) −	9.2702e-1 (3.94e-3)
	16	9.3912e-1 (1.16e-2) −	9.4100e-1 (5.13e-3) −	9.0144e-1 (2.55e-2) −	2.4671e-1 (2.43e-1) −	2.6542e-1 (1.05e-1) −	9.4820e-1 (4.44e-3)
	20	8.8115e-1 (4.33e-2) −	9.8748e-1 (2.64e-3) ≈	9.8824e-1 (3.29e-3) −	2.8234e-1 (3.06e-1) −	1.1271e-1 (5.73e-3) −	9.8897e-1 (2.26e-3)
WFG5	5	7.5682e-1 (6.31e-4) −	7.5825e-1 (7.13e-4) −	7.5836e-1 (7.79e-4) −	1.0942e-1 (4.26e-2) −	2.4200e-1 (3.43e-2) −	7.5903e-1 (6.06e-4)
	8	8.6374e-1 (1.64e-3) −	8.6897e-1 (1.94e-3) −	8.7533e-1 (1.18e-3) +	3.7913e-1 (3.08e-1) −	3.1396e-1 (7.66e-3) −	8.7098e-1 (1.58e-3)
	12	8.9339e-1 (2.34e-3) −	8.9878e-1 (1.29e-3) −	9.0654e-1 (6.93e-4) +	6.1226e-1 (2.74e-1) −	2.9971e-1 (1.78e-2) −	9.0075e-1 (2.14e-3)
	16	8.7848e-1 (8.92e-3) −	8.6480e-1 (1.01e-2) −	7.1205e-1 (3.12e-2) −	1.8666e-1 (2.45e-1) −	2.1659e-1 (3.40e-2) −	8.9520e-1 (2.30e-3)
	20	8.7832e-1 (2.77e-2) −	9.1237e-1 (1.21e-3) ≈	9.1462e-1 (7.26e-4) +	3.2685e-1 (3.57e-1) −	1.1497e-1 (2.24e-2) −	9.1267e-1 (9.78e-4)
WFG6	5	7.4793e-1 (4.70e-3) ≈	7.5140e-1 (4.54e-3) ≈	7.5403e-1 (6.26e-3) ≈	1.5827e-1 (5.22e-2) −	2.1243e-1 (3.11e-2) −	7.5175e-1 (7.43e-3)
	8	8.4591e-1 (7.38e-3) −	8.5419e-1 (5.53e-3) ≈	8.5782e-1 (6.09e-3) −	2.5431e-1 (1.58e-1) −	2.6716e-1 (5.07e-2) −	8.5787e-1 (5.08e-3)
	12	8.6859e-1 (7.28e-3) −	8.7732e-1 (9.90e-3) ≈	8.9306e-1 (7.60e-3) +	3.0384e-1 (1.96e-1) −	2.8629e-1 (6.65e-2) −	8.8046e-1 (9.18e-3)
	16	8.7111e-1 (1.99e-2) ≈	8.6644e-1 (1.48e-2) ≈	7.1903e-1 (4.11e-2) −	2.9858e-1 (2.14e-1) −	2.1473e-1 (3.77e-2) −	8.7303e-1 (1.71e-2)
	20	8.3650e-1 (4.71e-2) −	8.9443e-1 (1.94e-2) ≈	9.1817e-1 (2.78e-2) ≈	4.3662e-1 (2.73e-1) −	1.6361e-1 (2.30e-2) −	9.0482e-1 (2.80e-2)
WFG7	5	7.9481e-1 (1.99e-3) −	8.0011e-1 (9.78e-4) −	7.9900e-1 (1.19e-3) −	1.8601e-1 (6.73e-2) −	3.3682e-1 (3.24e-2) −	8.0507e-1 (5.88e-4)
	8	9.1000e-1 (4.09e-3) −	9.2042e-1 (1.80e-3) −	9.2153e-1 (1.98e-3) −	3.1720e-1 (1.43e-1) −	3.7262e-1 (2.17e-2) −	9.2633e-1 (1.32e-3)
	12	9.5209e-1 (9.41e-3) −	9.5152e-1 (4.03e-3) −	9.4709e-1 (3.97e-3) −	6.3470e-1 (2.33e-1) −	3.7542e-1 (3.27e-2) −	9.5870e-1 (2.07e-3)
	16	9.5962e-1 (9.14e-3) ≈	9.4904e-1 (7.33e-3) −	8.2391e-1 (2.93e-2) −	3.8346e-1 (3.26e-1) −	2.8677e-1 (8.08e-2) −	9.6561e-1 (4.36e-3)
	20	8.4469e-1 (3.75e-2) −	9.9272e-1 (7.08e-4) ≈	9.7872e-1 (2.62e-3) −	4.3018e-1 (3.50e-1) −	1.0107e-1 (3.32e-2) −	9.9240e-1 (1.05e-3)
WFG8	5	7.2675e-1 (2.01e-3) −	7.2860e-1 (1.74e-3) ≈	7.3612e-1 (1.98e-3) +	6.7585e-2 (6.07e-2) −	2.2896e-1 (1.72e-2) −	7.3022e-1 (1.81e-3)
	8	8.1010e-1 (5.87e-3) ≈	8.1032e-1 (4.92e-3) ≈	8.3586e-1 (2.71e-3) +	2.0683e-1 (1.30e-1) −	3.1276e-1 (3.58e-2) −	8.1284e-1 (3.91e-3)
	12	8.5811e-1 (3.03e-2) +	8.2792e-1 (1.17e-2) ≈	8.7595e-1 (1.05e-2) +	2.8140e-1 (4.31e-2) −	2.9974e-1 (7.46e-2) −	8.3203e-1 (8.97e-3)
	16	8.8869e-1 (3.41e-2) ≈	8.7462e-1 (3.38e-2) −	7.6026e-1 (4.91e-2) −	1.9837e-1 (6.21e-2) −	2.1323e-1 (4.07e-2) −	8.8987e-1 (9.55e-3)
	20	5.9911e-1 (1.78e-1) −	9.6045e-1 (9.55e-3) +	9.6081e-1 (1.62e-2) ≈	2.2075e-1 (3.54e-2) −	1.5093e-1 (2.15e-2) −	9.5280e-1 (4.90e-3)
WFG9	5	7.2182e-1 (2.56e-2) −	7.4765e-1 (4.65e-3) −	7.1184e-1 (2.13e-2) −	2.7325e-1 (1.10e-1) −	2.4290e-1 (1.62e-2) −	7.5753e-1 (4.91e-3)
	8	8.1627e-1 (8.71e-3) −	8.4388e-1 (8.86e-3) −	8.2523e-1 (1.79e-2) −	6.3098e-1 (1.39e-1) −	2.9556e-1 (1.44e-2) −	8.5943e-1 (7.02e-3)
	12	8.4770e-1 (1.74e-2) −	8.6476e-1 (1.40e-2) −	8.5304e-1 (1.38e-2) −	7.3167e-1 (1.61e-2) −	3.2330e-1 (1.77e-2) −	8.8320e-1 (1.19e-2)
	16	8.7519e-1 (3.21e-2) ≈	8.4067e-1 (1.26e-2) −	7.7586e-1 (2.90e-2) −	6.6414e-1 (1.94e-1) −	2.9630e-1 (2.98e-2) −	8.7871e-1 (1.09e-2)
	20	8.2567e-1 (3.92e-2) −	9.1908e-1 (8.74e-3) −	9.0566e-1 (1.50e-2) −	8.4398e-1 (4.21e-2) −	1.4200e-1 (4.10e-2) −	9.2399e-1 (6.99e-3)
+/−/≈		8/25/12	6/21/18	17/22/6	1/39/5	0/42/3	

4.3. The effectiveness of double association

To verify the advantage of our double association, we use it to replace the association method of other decomposition-based algorithms. Here, θ-DEA [16] and RPD-NSGA-II [17] as two classical decomposition-based MaOEAs are selected to make experiments. Both of them use single association method to associate solutions.

Tables 6 and 7 show the statistical results of θ-DEA with different association methods in terms of IGD and HV values on DTLZ test suite, respectively. θ-DEA_rand and θ-DEA_DA represent using the random association method and double association method to replace the association method used in θ-DEA. The best results are highlighted in bold. We can clearly observe from Tables 6 and 7 that after using our proposed double association method to replace the single association method used in θ-DEA, the performance of θ-DEA is significantly improved, while the random association makes the performance of θ-DEA worse.

Tables 8 and 9 show the statistical results of RPD-NSGA-II with different association methods in terms of IGD and HV values on DTLZ test suite, respectively. RPD-NSGA-II_rand and RPD-NSGA-II_DA represent that using random association method and double association method to replace the association method used in RPD-NSGA-II. The best results are highlighted in bold. It can be seen clearly from Tables 8 and 9 that after using our proposed double association method to replace the single association method used in RPD-NSGA-II, the performance of RPD-NSGA-II is significantly improved, while the random association makes the performance of RPD-NSGA-II worse.

These results indicate that double association, as a new method to associate the solutions in objective space, can better maintain the diversity of the obtained solutions and give some potential solutions an opportunity for further evolution, and thus can improve the performance of reference point-based algorithms.

4.4. Effect of different penalty parameter μ

In our proposed quality evaluation scheme, a penalty parameter μ is designed to punish the solutions in dense area from global perspective. Considering that in an ideal state, each subspace should have two solutions distributed, so we set $\mu = \frac{|S(i)|}{2}$ to balance the convergence and diversity. To investigate the effect of different value of parameter μ on the proposed quality evaluation scheme, DAEA is tested on DTLZ1-7 with three different μ values: 1)$\mu = 1$; 2)$\mu = 5$; 3)$\mu = \frac{|S(i)|}{2}$.

Tables 10 and 11 show the statistical results of DAEA with three different μ values in terms of IGD and HV values on DTLZ test suite, respectively. The best results are highlighted in bold. As can be observed, DAEA_$\mu = \frac{|S(i)|}{2}$ has achieved most of the best results out of 35 test instances. From the statistic results of Wilcoxon rank sum test in the last line of Tables 10 and 11, we can also see that, among the 35 test instances, the IGD results obtained by DAEA_$\mu = \frac{|S(i)|}{2}$ are significantly better than those obtained by DAEA_$\mu = 1$ and DAEA_$\mu = 5$ in most test instances. These experimental results indicate that setting the penalty parameter μ as a dynamic parameter is much better than taking constants 1 and 5.

4.5. Further discussion on DAEA

According to the experimental results (the effectiveness of double association) and comparative results, we can clearly see that these algorithms adopting the existing association operator for

Table 6. Statistical results (mean and standard deviation) of the igd values obtained by three association methods based on θ-DEA.

Problem	M	θ-DEA	θ-DEA_rand	θ-DEA_DA
DTLZ1	5	4.4240e-1 (5.44e-1) −	4.7501e-1 (1.41e-2) −	5.2688e-2 (5.13e-5)
	8	1.0901e-1 (3.16e-2) ≈	4.9120e-1 (1.45e-2) −	9.4331e-2 (8.02e-3)
	12	1.3530e-1 (2.56e-2) ≈	4.9541e-1 (1.47e-2) −	1.5249e-1 (4.44e-2)
	16	1.4073e-1 (3.28e-2) ≈	5.0929e-1 (1.30e-2) −	1.2878e-1 (1.97e-2)
	20	5.9928e-1 (4.64e-1) ≈	5.4043e-1 (4.90e-3) −	2.2740e-1 (3.66e-2)
DTLZ2	5	1.8115e-1 (1.69e-2) ≈	1.1021e+0 (1.90e-2) −	1.6512e-1 (9.16e-6)
	8	3.1244e-1 (8.12e-4) ≈	1.2139e+0 (3.47e-9) −	3.1234e-1 (4.48e-4)
	12	4.9740e-1 (2.62e-4) ≈	1.2588e+0 (1.73e-9) −	4.9728e-1 (3.08e-4)
	16	5.3798e-1 (6.55e-4) ≈	1.2962e+0 (1.97e-9) −	5.3850e-1 (1.14e-3)
	20	6.4238e-1 (2.25e-2) −	1.3357e+0 (1.45e-9) −	6.2316e-1 (3.11e-4)
DTLZ3	5	1.0583e+1 (1.17e+1) ≈	1.0880e+0 (4.26e-2) −	1.6598e-1 (7.41e-4)
	8	4.8590e-1 (9.77e-2) ≈	1.2148e+0 (5.78e-4) −	4.5528e-1 (1.03e-1)
	12	7.3979e-1 (3.27e-1) ≈	1.2595e+0 (7.66e-4) −	6.9328e-1 (2.03e-1)
	16	6.1792e-1 (4.92e-2) ≈	1.2969e+0 (7.23e-4) −	5.9654e-1 (5.79e-2)
	20	3.1061e+1 (3.28e+1) −	1.3369e+0 (1.33e-3) −	8.0789e-1 (3.49e-1)
DTLZ4	5	1.8379e-1 (2.02e-2) ≈	1.1081e+0 (1.37e-9) −	1.6508e-1 (2.23e-5)
	8	3.1412e-1 (6.19e-4) −	1.2139e+0 (1.19e-9) −	3.1352e-1 (4.85e-4)
	12	4.9795e-1 (8.95e-5) ≈	1.1944e+0 (1.03e-2) −	4.9789e-1 (1.63e-4)
	16	5.4567e-1 (2.12e-3) ≈	1.2793e+0 (2.86e-3) −	5.4516e-1 (1.59e-3)
	20	6.3861e-1 (1.70e-2) −	1.3350e+0 (1.56e-3) −	6.2295e-1 (1.10e-4)
DTLZ5	5	3.1261e-1 (9.46e-2) −	7.3813e-1 (1.25e-2) −	5.6616e-2 (7.99e-3)
	8	1.4105e-1 (1.68e-2) −	2.2039e+0 (5.22e-1) −	7.4929e-2 (8.66e-3)
	12	1.9590e-1 (2.32e-2) −	2.9270e+0 (6.77e-2) −	1.3027e-1 (2.12e-2)
	16	1.9380e-1 (4.31e-2) −	2.9439e+0 (3.82e-2) −	1.2846e-1 (2.30e-2)
	20	1.8249e-1 (4.20e-2) ≈	2.9560e+0 (4.15e-11) −	1.4674e-1 (3.82e-2)
DTLZ6	5	1.7190e-1 (4.81e-2) −	7.4209e-1 (2.89e-8) −	1.1853e-1 (2.66e-2)
	8	2.5362e-1 (3.46e-2) −	3.3118e+0 (6.01e-1) −	1.8213e-1 (6.63e-2)
	12	2.3504e-1 (4.83e-2) −	6.5446e+0 (4.33e-1) −	1.7638e-1 (3.31e-2)
	16	2.7272e-1 (5.30e-2) −	8.4206e+0 (3.43e-1) −	2.0287e-1 (1.76e-2)
	20	2.3690e-1 (5.75e-2) ≈	8.9562e+0 (2.96e-1) −	2.1557e-1 (5.88e-2)
DTLZ7	5	3.1000e-1 (3.15e-2) ≈	1.7463e+0 (2.72e-2) −	3.1070e-1 (1.92e-2)
	8	6.4153e-1 (4.85e-2) +	3.6844e+0 (2.71e-2) −	6.8944e-1 (4.59e-2)
	12	1.3153e+0 (1.75e-1) −	6.2480e+0 (1.23e-2) −	1.0904e+0 (6.31e-2)
	16	4.0932e+0 (6.50e-1) ≈	1.1464e+1 (5.41e-1) −	4.0745e+0 (8.51e-1)
	20	8.7351e+0 (6.28e-1) −	1.6108e+1 (5.23e-1) −	7.4542e+0 (1.03e+0)
+/−/≈		1/15/19	0/35/0	

Table 7. Statistical results (mean and standard deviation) of the HV values obtained by three association methods based on θ-DEA.

Problem	M	θ-DEA	θ-DEA_rand	θ-DEA_DA
DTLZ1	5	5.2349e-1 (4.91e-1) −	9.8304e-2 (2.37e-2) −	9.7972e-1 (1.64e-4)
	8	9.8228e-1 (4.73e-2) ≈	1.2296e-1 (2.67e-2) −	9.9821e-1 (1.07e-3)
	12	9.9723e-1 (6.02e-3) ≈	1.2111e-1 (2.70e-2) −	9.8239e-1 (4.24e-2)
	16	9.9783e-1 (2.95e-3) ≈	1.1653e-1 (2.44e-2) −	9.9850e-1 (2.03e-3)
	20	4.9419e-1 (5.16e-1) ≈	1.0567e-1 (9.64e-3) −	9.8832e-1 (1.24e-2)
DTLZ2	5	7.9430e-1 (1.91e-2) −	9.1516e-2 (1.92e-3) −	8.1248e-1 (4.54e-4)
	8	9.4555e-1 (2.68e-4) ≈	9.0909e-2 (5.20e-9) −	9.4545e-1 (3.42e-4)
	12	9.8708e-1 (1.95e-4) ≈	9.0909e-2 (2.50e-9) −	9.8724e-1 (1.42e-4)
	16	9.9317e-1 (1.61e-4) ≈	9.0909e-2 (2.76e-9) −	9.9317e-1 (1.79e-4)
	20	9.3043e-1 (7.30e-2) −	9.0909e-2 (1.97e-9) −	9.9860e-1 (4.86e-5)
DTLZ3	5	4.0104e-1 (4.23e-1) ≈	9.2788e-2 (6.34e-3) −	8.0228e-1 (4.50e-3)
	8	6.6856e-1 (1.57e-1) ≈	8.9667e-2 (8.65e-4) −	7.2520e-1 (1.67e-1)
	12	6.1467e-1 (3.59e-1) ≈	8.9888e-2 (1.10e-3) −	6.4951e-1 (3.03e-1)
	16	9.3001e-1 (7.92e-2) ≈	8.9941e-2 (1.01e-3) −	9.3233e-1 (8.00e-2)
	20	3.9097e-1 (5.05e-1) ≈	8.9297e-2 (1.81e-3) −	7.8047e-1 (4.13e-1)
DTLZ4	5	7.9448e-1 (1.93e-2) ≈	9.0909e-2 (2.24e-9) −	8.1251e-1 (5.44e-4)
	8	9.4594e-1 (1.36e-4) ≈	9.0909e-2 (1.78e-9) −	9.4601e-1 (2.59e-4)
	12	9.8769e-1 (1.57e-4) ≈	8.3199e-2 (6.78e-3) −	9.8776e-1 (5.68e-5)
	16	9.9411e-1 (4.69e-5) ≈	8.9620e-2 (1.23e-3) −	9.9413e-1 (1.04e-4)
	20	9.9303e-1 (6.49e-3) ≈	9.1036e-2 (4.98e-4) −	9.9871e-1 (4.57e-5)
DTLZ5	5	9.9274e-2 (7.05e-3) −	8.1827e-2 (2.87e-2) −	1.1949e-1 (2.28e-3)
	8	9.0492e-2 (2.13e-3) −	9.0909e-3 (2.87e-2) −	9.7509e-2 (1.23e-3)
	12	8.7500e-2 (2.28e-3) −	0.0000e+0 (0.00e+0) −	9.1881e-2 (9.81e-4)
	16	8.8949e-2 (9.40e-4) −	0.0000e+0 (0.00e+0) −	9.0883e-2 (5.67e-4)
	20	9.0111e-2 (4.10e-4) ≈	0.0000e+0 (0.00e+0) −	9.0417e-2 (4.99e-4)
DTLZ6	5	9.6538e-2 (5.91e-3) −	9.1524e-3 (2.87e-2) −	1.0990e-1 (6.90e-3)
	8	9.0875e-2 (2.79e-4) ≈	0.0000e+0 (0.00e+0) −	9.1523e-2 (1.27e-3)
	12	9.0811e-2 (1.93e-4) −	0.0000e+0 (0.00e+0) −	9.1005e-2 (1.78e-4)
	16	9.0980e-2 (2.48e-4) ≈	0.0000e+0 (0.00e+0) −	9.1029e-2 (4.83e-4)
	20	9.0830e-2 (3.94e-4) −	0.0000e+0 (0.00e+0) −	9.1258e-2 (3.80e-4)
DTLZ7	5	2.1587e-1 (1.26e-2) ≈	2.2390e-2 (3.14e-4) −	2.2123e-1 (1.21e-2)
	8	1.8889e-1 (6.62e-3) ≈	1.7539e-2 (6.49e-4) −	1.8702e-1 (1.29e-2)
	12	1.6784e-1 (6.22e-3) −	1.5163e-2 (8.39e-4) −	1.8436e-1 (5.02e-3)
	16	8.6784e-2 (1.72e-2) −	2.9130e-2 (3.13e-2) −	1.2226e-1 (1.39e-2)
	20	1.4454e-1 (3.63e-3) +	8.3927e-2 (2.39e-2) −	1.1883e-1 (1.08e-2)
+/ − / ≈		1/12/22	0/35/0	

Table 8. Sstatistical results (mean and standard deviation) of the IGD alues obtained by three association methods based on RPD-NSGA-II.

Problem	*M*	RPD-NSGA-II	RPD-NSGA-II_rand	RPD-NSGAII_DA
DTLZ1	5	6.1512e-2 (1.06e-3) −	1.8369e-1 (3.06e-2) −	5.9109e-2 (1.81e-3)
	8	1.3007e-1 (2.48e-2) ≈	2.4255e-1 (1.05e-1) −	1.1911e-1 (2.11e-2)
	12	1.7769e-1 (1.69e-2) ≈	4.3168e-1 (1.33e-1) −	1.8011e-1 (9.89e-3)
	16	1.6846e-1 (6.48e-3) +	3.8623e-1 (1.69e-1) −	1.8820e-1 (6.97e-3)
	20	2.0861e-1 (8.84e-3) ≈	7.3910e-1 (3.52e-1) −	2.0357e-1 (9.60e-3)
DTLZ2	5	1.6639e-1 (1.27e-3) ≈	2.3446e-1 (1.21e-2) −	1.6645e-1 (9.28e-4)
	8	3.1130e-1 (5.90e-4) −	3.7418e-1 (9.87e-3) −	3.1060e-1 (1.01e-3)
	12	4.8922e-1 (1.33e-3) ≈	5.3838e-1 (1.08e-2) −	4.8979e-1 (1.23e-3)
	16	5.3964e-1 (1.94e-3) +	7.1640e-1 (2.58e-2) −	5.4299e-1 (2.17e-3)
	20	6.2435e-1 (1.52e-3) −	1.1012e+0 (8.95e-2) −	6.2168e-1 (1.40e-3)
DTLZ3	5	2.0070e-1 (4.08e-3) −	1.1385e+1 (4.69e+0) −	1.9386e-1 (6.09e-3)
	8	3.4970e-1 (1.22e-2) ≈	2.7850e+1 (9.21e+0) −	3.4634e-1 (1.52e-2)
	12	5.2323e-1 (1.84e-2) ≈	5.8508e+1 (2.07e+1) −	5.1967e-1 (3.05e-2)
	16	6.0176e-1 (5.33e-2) ≈	4.5266e+1 (1.67e+1) −	5.7453e-1 (1.15e-2)
	20	7.3898e-1 (2.50e-1) ≈	8.6181e+1 (2.82e+1) −	7.0508e-1 (2.26e-1)
DTLZ4	5	1.6858e-1 (9.28e-4) ≈	2.5456e-1 (8.63e-3) −	1.6806e-1 (1.28e-3)
	8	3.1271e-1 (1.22e-3) ≈	4.0707e-1 (1.21e-2) −	3.1185e-1 (9.01e-4)
	12	4.9246e-1 (1.08e-3) ≈	5.9980e-1 (1.81e-2) −	4.9232e-1 (1.07e-3)
	16	5.5665e-1 (1.40e-3) +	8.0970e-1 (3.02e-2) −	5.5930e-1 (2.60e-3)
	20	6.2742e-1 (8.97e-4) −	1.0987e+0 (8.13e-2) −	6.2560e-1 (5.83e-4)
DTLZ5	5	9.4819e-2 (2.57e-2) −	2.1829e-1 (2.94e-2) −	5.6864e-2 (7.99e-3)
	8	1.1654e-1 (1.29e-2) −	1.7542e-1 (2.66e-2) −	6.9108e-2 (1.71e-2)
	12	1.6217e-1 (1.20e-2) −	1.8434e-1 (4.13e-2) −	1.0235e-1 (9.57e-3)
	16	1.6523e-1 (1.38e-2) −	3.4193e-1 (5.51e-2) −	3.6315e-2 (9.69e-3)
	20	1.7108e-1 (1.11e-2) −	3.1531e-1 (5.30e-2) −	1.4936e-1 (1.07e-2)
DTLZ6	5	2.0796e-1 (1.49e-1) −	4.3726e-1 (2.34e-1) −	8.5257e-2 (2.35e-2)
	8	1.7850e-1 (4.90e-2) −	3.0353e+0 (4.59e-1) −	7.1994e-2 (1.98e-2)
	12	1.6568e-1 (2.57e-2) −	4.4509e+0 (8.63e-1) −	8.8925e-2 (2.53e-2)
	16	2.1528e-1 (8.83e-2) −	3.1505e+0 (7.36e-1) −	3.2185e-2 (1.25e-2)
	20	2.3634e-1 (7.17e-2) −	8.3142e+0 (5.41e-1) −	1.2443e-1 (4.05e-2)
DTLZ7	5	3.2590e-1 (5.21e-3) −	3.4387e-1 (2.75e-2) −	3.1308e-1 (7.57e-3)
	8	1.0637e+0 (2.69e-2) −	1.2022e+0 (1.50e-1) −	6.7709e-1 (1.70e-2)
	12	2.7331e+0 (2.84e-1) ≈	3.2798e+0 (5.15e-1) ≈	3.0976e+0 (9.37e-1)
	16	7.9525e+0 (4.41e-1) ≈	1.1818e+1 (2.09e+0) −	8.2591e+0 (8.15e-1)
	20	1.3533e+1 (1.17e+0) ≈	2.9395e+1 (6.70e+0) −	1.3529e+1 (1.04e+0)
+/ − / ≈		3/17/15	0/34/1	

Table 9. statistical results (mean and standard deviation) of the HV values obtained by three association methods based on RPD-NSGA-II.

Problem	*M*	RPD-NSGA-II	RPD-NSGA-II_rand	RPD-NSGAII_DA
DTLZ1	5	9.7349e-1 (1.46e-3) ≈	7.9516e-1 (6.23e-2) −	9.7465e-1 (1.48e-3)
	8	9.7408e-1 (2.62e-2) ≈	7.2094e-1 (3.24e-1) −	9.8582e-1 (1.72e-2)
	12	9.7139e-1 (2.10e-2) +	2.9413e-1 (2.98e-1) −	9.5428e-1 (2.02e-2)
	16	9.6681e-1 (1.25e-2) ≈	4.7984e-1 (3.91e-1) −	9.4465e-1 (2.73e-2)
	20	9.6983e-1 (9.62e-3) +	1.7406e-1 (3.36e-1) −	9.5259e-1 (1.56e-2)
DTLZ2	5	8.1076e-1 (7.25e-4) +	7.2760e-1 (1.00e-2) −	8.0998e-1 (5.87e-4)
	8	9.4239e-1 (9.10e-4) ≈	8.4481e-1 (1.91e-2) −	9.4244e-1 (8.86e-4)
	12	9.8681e-1 (4.31e-4) ≈	9.0111e-1 (1.14e-2) −	9.8691e-1 (5.45e-4)
	16	9.9094e-1 (4.67e-4) ≈	7.3201e-1 (6.47e-2) −	9.9129e-1 (8.41e-4)
	20	9.9813e-1 (1.18e-4) ≈	1.7005e-1 (1.01e-1) −	9.9811e-1 (1.23e-4)
DTLZ3	5	7.7893e-1 (8.46e-3) ≈	0.0000e+0 (0.00e+0) −	7.8549e-1 (6.37e-3)
	8	9.0970e-1 (1.49e-2) ≈	0.0000e+0 (0.00e+0) −	9.1290e-1 (1.31e-2)
	12	9.6468e-1 (1.12e-2) ≈	0.0000e+0 (0.00e+0) −	9.6693e-1 (1.22e-2)
	16	9.4265e-1 (9.14e-2) ≈	0.0000e+0 (0.00e+0) −	9.7792e-1 (6.32e-3)
	20	8.7129e-1 (3.10e-1) ≈	0.0000e+0 (0.00e+0) −	8.9527e-1 (3.11e-1)
DTLZ4	5	8.0802e-1 (1.67e-3) ≈	7.1195e-1 (1.53e-2) −	8.0919e-1 (1.96e-3)
	8	9.4474e-1 (6.90e-4) ≈	8.5856e-1 (1.38e-2) −	9.4428e-1 (7.15e-4)
	12	9.8815e-1 (1.74e-4) ≈	8.9848e-1 (2.37e-2) −	9.8821e-1 (1.83e-4)
	16	9.9408e-1 (1.80e-4) ≈	6.0449e-1 (9.02e-2) −	9.9413e-1 (1.01e-4)
	20	9.9846e-1 (6.31e-5) ≈	1.9163e-1 (9.25e-2) −	9.9848e-1 (5.15e-5)
DTLZ5	5	1.1327e-1 (3.81e-3) −	9.1168e-2 (3.72e-4) −	1.2648e-1 (6.93e-4)
	8	9.4723e-2 (1.38e-3) −	9.0816e-2 (3.30e-4) −	1.0502e-1 (4.99e-4)
	12	7.9339e-2 (8.49e-3) −	9.0761e-2 (4.34e-4) −	9.6513e-2 (4.52e-4)
	16	8.6975e-2 (1.87e-3) −	9.0105e-2 (4.22e-4) −	9.4089e-2 (2.02e-4)
	20	8.9082e-2 (2.26e-3) −	9.0967e-2 (4.28e-4) −	9.1917e-2 (3.43e-4)
DTLZ6	5	9.6125e-2 (3.40e-2) −	8.7136e-2 (3.12e-2) −	1.2306e-1 (1.71e-3)
	8	8.1241e-2 (3.04e-2) −	0.0000e+0 (0.00e+0) −	1.0448e-1 (9.52e-4)
	12	8.6737e-2 (1.76e-2) −	0.0000e+0 (0.00e+0) −	9.4160e-2 (8.74e-3)
	16	8.4571e-2 (2.22e-2) −	0.0000e+0 (0.00e+0) −	9.3860e-2 (3.41e-4)
	20	7.3580e-2 (3.78e-2) −	0.0000e+0 (0.00e+0) −	9.2582e-2 (4.07e-4)
DTLZ7	5	2.5918e-1 (2.40e-3) −	2.3233e-1 (4.64e-3) −	2.6482e-1 (1.50e-3)
	8	1.8968e-1 (2.37e-3) −	1.4120e-1 (9.86e-3) −	2.1010e-1 (2.92e-3)
	12	1.5914e-1 (1.14e-2) −	8.2322e-2 (8.94e-3) −	1.7686e-1 (6.80e-3)
	16	1.6631e-1 (3.21e-3) +	1.5403e-2 (1.84e-2) −	1.5894e-1 (9.13e-3)
	20	1.1577e-1 (2.01e-2) ≈	0.0000e+0 (0.00e+0) −	1.1545e-1 (1.99e-2)
+/−/≈		4/13/18	0/35/0	

Table 10. statistical results (mean and standard deviation) of the IGD values obtained by three different μ values based on DAEA.

Problem	M	D	DAEA_$\mu = 1$	DAEA_$\mu = 5$	DAEA_$\mu = \frac{\lvert S(i)\rvert}{2}$
DTLZ1	5	9	4.9697e-2 (4.65e-4) −	5.2658e-2 (8.34e-5) −	4.9172e-2 (4.99e-4)
	8	12	1.0800e-1 (2.17e-2) ≈	9.4784e-2 (1.30e-2) +	1.1271e-1 (4.69e-2)
	12	16	1.2593e-1 (4.07e-2) −	1.2532e-1 (7.93e-3) −	1.2253e-1 (2.75e-2)
	16	20	1.6764e-1 (4.83e-2) −	1.4620e-1 (6.11e-2) ≈	1.3974e-1 (3.34e-2)
	20	24	2.5023e-1 (2.96e-2) −	2.3939e-1 (5.29e-2) ≈	2.3781e-1 (3.49e-2)
DTLZ2	5	14	1.6519e-1 (1.00e-4) −	1.6513e-1 (6.48e-6) −	1.6511e-1 (8.48e-6)
	8	17	3.1197e-1 (6.22e-4) ≈	3.1208e-1 (4.02e-4) −	3.1155e-1 (3.06e-4)
	12	21	4.9518e-1 (6.84e-4) ≈	4.9708e-1 (3.41e-4) −	4.9610e-1 (4.46e-4)
	16	25	5.3545e-1 (1.32e-3) ≈	5.3816e-1 (1.04e-3) ≈	5.3785e-1 (9.95e-4)
	20	29	6.2299e-1 (3.86e-4) ≈	6.2323e-1 (1.51e-4) −	6.2279e-1 (1.04e-4)
DTLZ3	5	14	1.6527e-1 (5.87e-4) −	1.6657e-1 (1.69e-3) −	1.6493e-1 (2.46e-4)
	8	17	4.2297e-1 (1.30e-1) ≈	4.3958e-1 (1.16e-1) ≈	4.5962e-1 (8.58e-2)
	12	21	6.9888e-1 (2.92e-1) ≈	7.9676e-1 (5.55e-1) ≈	5.9533e-1 (8.68e-2)
	16	25	7.5356e-1 (5.51e-1) ≈	5.9583e-1 (5.39e-2) ≈	6.5141e-1 (1.69e-1)
	20	29	1.1294e+0 (8.34e-1) −	1.0717e+0 (9.85e-1) ≈	8.0177e-1 (3.09e-1)
DTLZ4	5	14	1.6519e-1 (7.78e-5) −	1.6508e-1 (2.26e-5) ≈	1.6506e-1 (3.72e-5)
	8	17	3.1299e-1 (4.51e-4) ≈	3.1334e-1 (3.06e-4) ≈	3.1317e-1 (4.13e-4)
	12	21	4.9720e-1 (2.05e-4) +	4.9797e-1 (1.92e-4) −	4.9756e-1 (1.89e-4)
	16	25	5.4822e-1 (2.27e-3) ≈	5.4586e-1 (2.33e-3) ≈	5.4687e-1 (2.54e-3)
	20	29	6.2268e-1 (2.54e-4) +	6.2299e-1 (6.42e-5) −	6.2287e-1 (1.04e-4)
DTLZ5	5	14	6.1556e-2 (2.39e-2) ≈	5.9657e-2 (1.87e-2) ≈	7.5732e-2 (4.55e-2)
	8	17	9.4172e-2 (2.46e-2) ≈	8.5318e-2 (9.13e-3) ≈	8.3139e-2 (1.31e-2)
	12	21	1.1370e-1 (1.41e-2) ≈	1.3713e-1 (2.24e-2) ≈	1.2358e-1 (1.32e-2)
	16	25	1.0961e-1 (1.88e-2) +	1.3878e-1 (2.88e-2) ≈	1.3315e-1 (2.38e-2)
	20	29	1.5568e-1 (3.70e-2) ≈	1.5593e-1 (2.73e-2) ≈	1.6225e-1 (2.22e-2)
DTLZ6	5	14	1.0620e-1 (4.59e-2) ≈	1.3771e-1 (5.28e-2) ≈	1.2169e-1 (4.62e-2)
	8	17	1.7569e-1 (3.56e-2) ≈	1.7248e-1 (4.43e-2) ≈	1.5675e-1 (3.95e-2)
	12	21	2.3007e-1 (1.79e-2) −	2.0198e-1 (3.02e-2) ≈	1.8632e-1 (3.82e-2)
	16	25	2.0145e-1 (1.55e-2) ≈	1.9360e-1 (2.73e-2) ≈	1.9294e-1 (3.85e-2)
	20	29	2.5995e-1 (5.03e-2) ≈	2.2883e-1 (5.04e-2) ≈	2.3362e-1 (4.92e-2)
DTLZ7	5	24	3.5144e-1 (3.24e-2) −	3.0785e-1 (2.28e-2) ≈	3.0784e-1 (1.56e-2)
	8	27	7.1936e-1 (4.02e-2) −	6.8736e-1 (3.95e-2) ≈	6.8731e-1 (2.60e-2)
	12	31	2.3150e+0 (8.40e-1) ≈	1.1127e+0 (6.74e-2) +	1.6662e+0 (5.86e-1)
	16	35	6.9230e+0 (9.21e-1) −	3.6936e+0 (5.46e-1) +	5.8983e+0 (9.77e-1)
	20	39	9.2385e+0 (2.15e+0) ≈	7.3038e+0 (9.80e-1) +	8.9384e+0 (1.60e+0)
	+/−/≈		3/12/20	4/9/22	

Table 11. statistical results (mean and standard deviation) of the HV values obtained by three different μ values based on DAEA.

Problem	M	D	DAEA_$\mu = 1$	DAEA_$\mu = 5$	DAEA_$\mu = \frac{\lvert S(i)\rvert}{2}$
DTLZ1	5	9	9.3911e-1 (9.57e-3) −	9.7973e-1 (1.39e-4) +	9.5827e-1 (6.78e-3)
	8	12	9.5089e-1 (2.03e-2) ≈	9.9662e-1 (6.88e-3) +	9.3008e-1 (6.70e-2)
	12	16	9.7224e-1 (1.50e-2) +	9.9981e-1 (1.28e-4) +	9.6234e-1 (1.20e-2)
	16	20	9.8496e-1 (2.61e-2) −	9.7749e-1 (7.94e-2) −	9.9375e-1 (1.36e-2)
	20	24	9.5040e-1 (6.60e-2) ≈	9.5968e-1 (6.79e-2) ≈	9.5982e-1 (5.59e-2)
DTLZ2	5	14	8.1243e-1 (6.13e-4) ≈	8.1239e-1 (3.86e-4) −	8.1271e-1 (3.04e-4)
	8	17	9.4597e-1 (4.31e-4) −	9.4535e-1 (2.67e-4) −	9.4607e-1 (3.23e-4)
	12	21	9.8858e-1 (1.56e-4) +	9.8719e-1 (1.15e-4) −	9.8755e-1 (1.16e-4)
	16	25	9.9380e-1 (1.58e-4) +	9.9321e-1 (1.42e-4) −	9.9359e-1 (1.05e-4)
	20	29	9.9862e-1 (3.61e-5) −	9.9860e-1 (3.39e-5) −	9.9865e-1 (3.85e-5)
DTLZ3	5	14	8.0731e-1 (3.89e-3) −	8.0014e-1 (7.94e-3) −	8.0889e-1 (1.89e-3)
	8	17	7.5373e-1 (2.04e-1) ≈	7.5112e-1 (1.70e-1) ≈	7.5303e-1 (1.08e-1)
	12	21	6.6094e-1 (3.59e-1) −	6.9215e-1 (3.61e-1) −	7.7273e-1 (1.81e-1)
	16	25	8.6723e-1 (2.54e-1) ≈	9.4778e-1 (5.00e-2) ≈	8.7748e-1 (2.34e-1)
	20	29	6.0436e-1 (4.33e-1) ≈	7.6096e-1 (3.91e-1) ≈	7.6537e-1 (3.68e-1)
DTLZ4	5	14	8.1244e-1 (4.59e-4) −	8.1249e-1 (5.28e-4) −	8.1258e-1 (2.52e-4)
	8	17	9.4610e-1 (3.47e-4) −	9.4593e-1 (2.82e-4) −	9.4635e-1 (1.90e-4)
	12	21	9.8873e-1 (1.43e-4) +	9.8765e-1 (8.90e-5) −	9.8784e-1 (1.04e-4)
	16	25	9.9452e-1 (1.29e-4) +	9.9416e-1 (1.00e-4) −	9.9438e-1 (9.37e-5)
	20	29	9.9870e-1 (2.76e-5) −	9.9869e-1 (4.72e-5) −	9.9870e-1 (3.36e-5)
DTLZ5	5	14	1.1888e-1 (5.92e-3) ≈	1.1826e-1 (5.07e-3) ≈	1.1520e-1 (9.26e-3)
	8	17	9.6929e-2 (2.27e-3) ≈	9.7049e-2 (1.30e-3) ≈	9.6742e-2 (1.94e-3)
	12	21	9.2916e-2 (8.14e-4) +	9.0740e-2 (1.40e-3) ≈	9.1081e-2 (1.33e-3)
	16	25	9.1644e-2 (9.53e-4) +	9.0598e-2 (5.88e-4) ≈	9.0677e-2 (4.98e-4)
	20	29	9.1574e-2 (8.51e-4) +	9.0758e-2 (4.82e-4) +	9.0213e-2 (7.62e-4)
DTLZ6	5	14	1.0883e-1 (9.03e-3) ≈	1.1242e-1 (6.16e-3) ≈	1.0967e-1 (7.56e-3)
	8	17	9.1229e-2 (2.59e-3) −	9.1399e-2 (5.67e-4) −	9.1761e-2 (8.03e-4)
	12	21	9.0925e-2 (2.97e-4) −	9.0838e-2 (2.71e-4) −	9.1090e-2 (5.76e-4)
	16	25	9.0881e-2 (2.45e-4) −	9.1074e-2 (4.39e-4) ≈	9.1075e-2 (2.99e-4)
	20	29	9.1198e-2 (4.27e-4) −	9.1217e-2 (6.55e-4) ≈	9.1224e-2 (5.21e-4)
DTLZ7	5	24	2.2246e-1 (2.13e-2) ≈	2.2598e-1 (1.29e-2) ≈	2.2040e-1 (1.36e-2)
	8	27	1.8110e-1 (2.25e-2) ≈	1.9663e-1 (1.35e-2) ≈	1.8476e-1 (1.96e-2)
	12	31	1.6599e-1 (7.70e-3) −	1.8279e-1 (1.17e-2) ≈	1.8045e-1 (5.84e-3)
	16	35	1.4163e-1 (7.58e-3) ≈	1.1315e-1 (9.01e-3) −	1.3506e-1 (1.37e-2)
	20	39	8.9536e-2 (1.46e-2) −	1.1070e-1 (9.32e-3) −	1.2885e-1 (7.62e-3)
	+/ − / ≈		8/15/12	4/17/14	

many-objective optimization may not work effectively and efficiently, especially in solving these problems with the multi-modal property or the complicated PF. One possible reason is that these algorithms ignore the importance of the subspace without associated, and further deteriorate their ability to explore unknown regions. In DAEA, we design the double association strategy that considers the empty subspace and associates it with a promising solution. This is helpful for DAEA to explore more unknown regions, and further improve the performance of DAEA. The effectiveness of the double association strategy has been validated by the corresponding experiment (i.e., the effectiveness of double association). In addition, most existing algorithms only consider the convergence and global diversity to measure the quality of solutions, which is not conductive to trade off the convergence and diversity of population. On the contrary, DAEA takes the convergence, global diversity, and local diversity into account, while using the designed dynamic penalty coefficient to adjust these three factors, and further balance the convergence and diversity of population. The effectiveness of the dynamic penalty coefficient has been confirmed by the corresponding experiment (i.e., parameter analysis).

However, our experimental results on the DTLZ7 indicate that the proposed DAEA does not present a clear advantage over θ-DEA and MaOEA-IGD. Even so, the proposed DAEA still outperforms the other competitors. In addition, when DAEA deals with WFG1 and WFG2 with irregular PFs, DAEA is slightly worse than NSGA-III and SPEA/R. This is because of this fact that, for many-objective problems with the irregular Pareto fronts, the even distribution of reference vectors combing with our proposed double association strategy may result in partial waste of computing resources and affect the performance of algorithm.

5. Conclusions and future work

In this paper, we have presented a double association-based evolutionary algorithm (denoted as DAEA) for many-objective optimization problems. The proposed double association strategy takes the empty subspace into account and associates it with a solution that is closest to this subspace, which can increase the probability of an unknown area being explored. In addition, a new quality evaluation scheme that takes the convergence and diversity of solutions into account is developed to measure the quality of each solution in subspace, where the diversity of each solution is subdivided into global diversity and local diversity. Then, a dynamic penalty coefficient is designed to protect these solutions located in sparse areas by penalizing the worse global diversity of solutions.

To demonstrate the high competitiveness of DAEA, we compare it with five state-of-the-art many-objective evolutionary algorithms on two test suites, DTLZ and WFG, with the number of objectives varying from 5 to 20. The corresponding experimental results demonstrate that our proposed DAEA has higher competitiveness in terms of both convergence enhancement and diversity maintenance compared with the other state-of-the-art MaOEAs.

Although DAEA outperforms the compared algorithms on most of the test problems, it struggles in dealing with some problems with the irregular Pareto fronts. For future research, we would like to add the idea of reference vector dynamic adjustment mechanism on DAEA to improve the algorithm's adjustability on the irregular Pareto fronts. In addition, we also consider verifying the performance of DAEA on MaOPs in the real world.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work was supported by National Natural Science Foundation of China (NO. 62202366) and Natural Science Basic Research Program of Shaanxi Province of China (2022JQ-624 and 2021JQ-674).

Conflicts of interest

The authors declare that they have no conflicts of interest.

References

1. J. Sun, Z. Miao, D. Gong, X. Zeng, J. Li, G. Wang, Interval multiobjective optimization with memetic algorithms, *IEEE Trans. Cybern.*, **50** (2022), 3444–3457. https://doi.org/10.1109/TCYB.2019.2908485
2. Y. Hua, Q. Liu, K. Hao, Y. Jin, A survey of evolutionary algorithms for multi-objective optimization problems with irregular pareto fronts, *IEEE-CAA J. Automatica Sin.*, **8** (2021), 303–318. https://doi.org/10.1109/JAS.2021.1003817
3. L. Ma, N. Li, Y. Guo, X. Wang, S. Yang, M. Huang, et al., Learning to optimize: Reference vector reinforcement learning adaption to constrained many-objective optimization of industrial copper burdening system, *IEEE Trans. Cybern.*, **52** (2022), 12698–12711. https://doi.org/10.1109/TCYB.2021.3086501
4. Z. Zhang, M. Zhao, H. Wang, Z. Cui, W. Zhang, An efficient interval many-objective evolutionary algorithm for cloud task scheduling problem under uncertainty, *Inf. Sci.*, **583** (2022), 56–72. https://doi.org/10.1016/j.ins.2021.11.027
5. M. Gao, B. Ai, Y. Niu, W. Wu, P. Yang, F. Lyu, et al., Efficient hybrid beamforming with anti-blockage design for high-speed railway communications, *IEEE Trans. Veh. Technol.*, **69** (2020), 9643–9655. https://doi.org/10.1109/TVT.2020.3000757
6. C. Tan, J. Yao, K. Tang, J. Sun, Cycle-based queue length estimation for signalized intersections using sparse vehicle trajectory data, *IEEE Trans. Intell. Transp. Syst.*, **22** (2021), 91–106. https://doi.org/10.1109/TITS.2019.2954937
7. Y. Guo, X. Tian, G. Fang, Y. Xu, Many-objective optimization with improved shuffled frog leaping algorithm for inter-basin water transfers, *Adv. Water Resour.*, **138** (2020), 103531. https://doi.org/10.1016/j.advwatres.2020.103531
8. F. Li, L. Gao, A. Garg, W. Shen, S. Huang, Two infill criteria driven surrogate-assisted multi-objective evolutionary algorithms for computationally expensive problems with medium dimensions, *Swarm Evol. Comput.*, **60** (2021), 100774. https://doi.org/10.1016/j.swevo.2020.100774

9. M. Wu, L. Wang, J. Xu, P. Hu, P. Xu, Adaptive surrogate-assisted multi-objective evolutionary algorithm using an efficient infill technique, *Swarm Evol. Comput.*, **75** (2022), 101170. https://doi.org/10.1016/j.swevo.2022.101170

10. B. Li, J. Li, K. Tang, X. Yao, Many-objective evolutionary algorithms: A survey, *ACM Comput. Surv.*, **48** (2015), 1–35. https://doi.org/10.1145/2792984

11. K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Trans. Evol. Comput.*, **6** (2002), 182–197. https://doi.org/10.1109/4235.996017

12. W. Zhang, J. Liu, S. Tan, H. Wang, A decomposition-rotation dominance based evolutionary algorithm with reference point adaption for many-objective optimization, *Expert Syst. Appl.*, **215** (2023), 119424. https://doi.org/10.1016/j.eswa.2022.119424

13. M. Laumanns, L. Thiele, K. Deb, E. Zitzler, Combining convergence and diversity in evolutionary multiobjective optimization, *Evol. Comput.*, **10** (2002), 263–282. https://doi.org/10.1162/106365602760234108

14. P. J. Bentley, J. P. Wakefield, Finding acceptable solutions in the Pareto-optimal range using multiobjective genetic algorithms, in *Soft Computing in Engineering Design and Manufacturing*, (1998), 231–240. https://doi.org/10.1007/978-1-4471-0427-8

15. D. Hadka, P. Reed, Borg: An auto-adaptive many-objective evolutionary computing framework, *Evol. Comput.*, **21** (2013), 231–259. https://doi.org/10.1162/EVCO_a_00075

16. X. Zou, Y. Chen, M. Liu, L. Kang, A new evolutionary algorithm for solving many-objective optimization problems, *IEEE Trans. Syst. Man Cybern. Part B Cybern.*, **38** (2008), 1402–1412. https://doi.org/10.1109/TSMCB.2008.926329

17. L. Li, G. Li, L. Chang, A many-objective particle swarm optimization with grid dominance ranking and clustering, *Appl. Soft. Comput.*, **96** (2020), 106661. https://doi.org/10.1016/j.asoc.2020.106661

18. F. D. Pierro, S. T. Khu, D. A. Savic, An investigation on preference order ranking scheme for multiobjective evolutionary optimization, *IEEE Trans. Evol. Comput.*, **11** (2007), 17–45. https://doi.org/10.1109/TEVC.2006.876362

19. G. Wang, H. Jiang, Fuzzy-dominance and its application in evolutionary many objective optimization, in *2007 International Conference on Computational Intelligence and Security Workshops*, 2007. https://doi.org/10.1109/CISW.2007.4425478

20. Y. Yuan, H. Xu, B. Wang, X. Yao, A new dominance relation based evolutionary algorithm for many-objective optimization, *IEEE Trans. Evol. Comput.*, **20** (2015), 16–37. https://doi.org/10.1109/TEVC.2015.2420112

21. M. Elarbi, S. Bechikh, A. Gupta, L. B. Said, Y. S. Ong, A new decomposition-based NSGA-II for many-objective optimization, *IEEE Trans. Syst. Man Cybern. Syst.*, **48** (2018), 1191–1210. https://doi.org/10.1109/TSMC.2017.2654301

22. K. Ikeda, H. Kita, S. Kobayashi, Failure of pareto-based MOEAs: Does non-dominated really mean near to optimal?, in *Proceedings of the 2001 Congress on Evolutionary Computation*, 2001. https://doi.org/10.1109/CEC.2001.934293

23. H. Sato, H. E. Aguirre, K. Tanaka, Controlling dominance area of solutions and its impact on the performance of moeas, in *International Conference on Evolutionary Multi-Criterion Optimization*, 2007. https://doi.org/10.1007/978-3-540-70928-2_5

24. C. Dai, Y. Wang, M. Ye, A new evolutionary algorithm based on contraction method for many-objective optimization problems, *Appl. Math. Comput.*, **245** (2014), 191–205. https://doi.org/10.1016/j.amc.2014.07.069

25. H. Ishibuchi, N. Tsukamoto, Y. Sakane, Y. Nojima, Indicator-based evolutionary algorithm with hypervolume approximation by achievement scalarizing functions, in *Proceedings of the Conference on Genetic and Evolutionary Computation*, 2010. https://doi.org/10.1145/1830483.1830578

26. Y. Sun, G. G. Yen, Z. Yi, IGD indicator-based evolutionary algorithm for many-objective optimization problems, *IEEE Trans. Evol. Comput.*, **23** (2019), 173–187. https://doi.org/10.1109/TEVC.2018.2791283

27. E. Zitzler, L. Thiele, Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach, *IEEE Trans. Evol. Comput.*, **3** (1999), 257–271. https://doi.org/10.1109/4235.797969

28. C. A. Rodríguez-Villalobos, C. A. C. Coello, A new multi-objective evolutionary algorithm based on a performance assessment indicator, in *Conference on Genetic and Evolutionary Computation*, 2012. https://doi.org/10.1145/2330163.2330235

29. J. Bader, E. Zitzler, HypE: An algorithm for fast hypervolume-based many-objective optimization, *Evol. Comput.*, **19** (2014), 45–76. https://doi.org/10.1162/EVCO_a_00009

30. S. Jiang, S. Yang, A strength pareto evolutionary algorithm based on reference direction for multiobjective and many-objective optimization, *IEEE Trans. Evol. Comput.*, **21** (2017), 329–346. https://doi.org/10.1109/TEVC.2016.2592479

31. K. Deb, D. Saxena, On finding pareto-optimal solutions through dimensionality reduction for certain large-dimensional multi-objective optimization problems, *Kangal Rep.*, **2005** (2005), 1–19.

32. H. K. Singh, A. Isaacs, T. Ray, A pareto corner search evolutionary algorithm and dimensionality reduction in many-objective optimization problems, *IEEE Trans. Evol. Comput.*, **15** (2011), 539–556. https://doi.org/10.1109/TEVC.2010.2093579

33. L. Thiele, K. Miettinen, P. J. Korhonen, J. Molina, A preference based evolutionary algorithm for multi-objective optimization, *Evol. Comput.*, **17** (2014), 411–436. https://doi.org/10.1162/evco.2009.17.3.411

34. R. Wang, R. C. Purshouse, P. J. Fleming, Preference-inspired coevolutionary algorithms for many-objective optimization, *IEEE Trans. Evol. Comput.*, **17** (2013), 474–494. https://doi.org/10.1109/TEVC.2012.2204264

35. R. Cheng, Y. Jin, M. Olhofer, B. Sendhoff, A reference vector guided evolutionary algorithm for many-objective optimization, *IEEE Trans. Evol. Comput.*, **20** (2016), 773–791. https://doi.org/10.1109/TEVC.2016.2519378

36. Y. Zhao, J. Zeng, Y. Tan, Neighborhood samples and surrogate assisted multi-objective evolutionary algorithm for expensive many-objective optimization problems, *Appl. Soft. Comput.*, **17** (2013), 474–494. https://doi.org/10.1016/j.asoc.2021.107268

37. Q. Zhang, H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, *IEEE Trans. Evol. Comput.*, **11** (2007), 712–731. https://doi.org/10.1109/TEVC.2007.892759

38. H. Zhao, C. Zhang, X. Zheng, C. Zhang, B. Zhang, A decomposition-based many-objective ant colony optimization algorithm with adaptive solution construction and selection approaches, *Swarm Evol. Comput.*, **68** (2022), 100977. https://doi.org/10.1016/j.swevo.2021.100977

39. Y. Zhou, Y. Xiang, Z. Chen, J. He, J. Wang, An adaptive convergence enhanced evolutionary algorithm for many-objective optimization problems, *Swarm Evol. Comput.*, **75** (2022), 101180. https://doi.org/10.1016/j.swevo.2022.101180

40. F. Gu, Y. M. Cheung, Self-organizing map-based weight design for decomposition-based many-objective evolutionary algorithm, *IEEE Trans. Evol. Comput.*, **22** (2018), 211–225. https://doi.org/10.1109/TEVC.2017.2695579

41. D. Han, W. Du, W. Du, Y. Jin, C. Wu, An adaptive decomposition based evolutionary algorithm for many-objective optimization, *Inf. Sci.*, **491** (2019), 204–222. https://doi.org/10.1016/j.ins.2019.03.062

42. R. Liu, J. Liu, R. Zhou, C. Lian, R. Bian, A region division based decomposition approach for evolutionary many-objective optimization, *Knowl. Based Syst.*, **194** (2020), 105518. https://doi.org/10.1016/j.knosys.2020.105518

43. H. Zhao, C. Zhang, B. Zhang, A decomposition-based many-objective ant colony optimization algorithm with adaptive reference points, *Inf. Sci.*, **540** (2020), 435–448. https://doi.org/10.1016/j.ins.2020.06.028

44. L. Ma, M. Huang, S. Yang, R. Wang, X. Wang, An adaptive localized decision variable analysis approach to large-scale multiobjective and many-objective optimization, *IEEE Trans. Cybern.*, **52** (2022), 6684–6696. https://doi.org/10.1109/TCYB.2020.3041212

45. Y. Zhang, G. Wang, K. Li, W. Yeh, M. Jian, J. Dong, Enhancing MOEA/D with information feedback models for large-scale many-objective optimization, *Inf. Sci.*, **522** (2020), 1–16. https://doi.org/10.1016/j.ins.2020.02.066

46. Y. Sun, B. Xue, M. Zhang, G. G. Yen, A new two-stage evolutionary algorithm for many-objective optimization, *IEEE Trans. Evol. Comput.*, **23** (2019), 748–761. https://doi.org/10.1109/TEVC.2018.2882166

47. K. Deb, H. Jain, An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints, *IEEE Trans. Evol. Comput.*, **18** (2014), 577–601. https://doi.org/10.1109/TEVC.2013.2281535

48. J. Yi, L. Xing, G. Wang, J. Dong, A. V. Vasilakos, A. H. Alavi, et al., Behavior of crossover operators in NSGA-III for large-scale optimization problems, *Inf. Sci.*, **509** (2020), 470–487. https://doi.org/10.1016/j.ins.2018.10.005

49. I. Das, J. E. Dennis, Normal-boundary intersection: A new method for generating the pareto

surface in nonlinear multicriteria optimization problems, *SIAM J. Optim.*, **8** (2006), 631–657. https://doi.org/10.1137/S1052623496307510

50. K. Deb, R. B. Agrawal, Simulated binary crossover for continuous search space, *Complex Syst.*, **9** (1994), 115–148.
51. K. Deb, M. Goyal, A combined genetic adaptive search (GeneAS) for engineering design, *Comput. Sci. Inf.*, **26** (1996), 30–45.
52. C. Dai, Y. Wang, A new multiobjective evolutionary algorithm based on decomposition of the objective space for multiobjective optimization, *J. Appl. Math.*, **2014** (2014), 1–9. https://doi.org/10.1155/2014/906147
53. Y. Yuan, H. Xu, B. Wang, An improved NSGA-III procedure for evolutionary many-objective optimization, in *16th Genetic and Evolutionary Computation Conference*, 2014. https://doi.org/10.1145/2576768.2598342
54. Y. Tian, R. Cheng, X. Zhang, Y. Jin, PlatEMO: A matlab platform for evolutionary multi-objective optimization [educational forum], *IEEE Comput. Intell. Mag.*, **12** (2017), 73–87. https://doi.org/10.1109/MCI.2017.2742868
55. K. Deb, L. Thiele, M. Laumanns, E. Zitzler, Scalable multi-objective optimization test problems, in *Proceedings of the 2002 Congress on Evolutionary Computation*, 2002. https://doi.org/10.1109/CEC.2002.1007032
56. S. Huband, P. Hingston, L. Barone, L. While, A review of multiobjective test problems and a scalable test problem toolkit, *IEEE Trans. Evol. Comput.*, **10** (2006), 477–506. https://doi.org/10.1109/TEVC.2005.861417
57. E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, V. G. D. Fonseca, Performance assessment of multiobjective optimizers: An analysis and review, *IEEE Trans. Evol. Comput.*, **7** (2003), 117–132. https://doi.org/10.1109/TEVC.2003.810758