A Novel Archive Maintenance for Adapting Weight Vectors in Decomposition-based Multi-objective Evolutionary Algorithms

Guang Peng
Institut für Informatik
Freie Universität Berlin
Berlin, Germany
Email: guang.peng@fu-berlin.de

Katinka Wolter
Institut für Informatik
Freie Universität Berlin
Berlin, Germany
Email: katinka.wolter@fu-berlin.de

***Abstract*—This paper proposes a novel archive maintenance for adapting weight vectors to improve the performance of the decomposition-based evolutionary algorithms for multi- and many-objective optimization problems with different Pareto front shapes (called AMAWV). AMAWV adopts a novel archive maintenance strategy for avoiding the dominance resistant solutions, as well as retaining the good diversity of non-dominated solution set. In addition, guided from the information of the archive, an adaptive weight vector method is designed to solve problems with various Pareto fronts. The proposed algorithm is compared with state-of-the-art algorithms on a number of test problems with different Pareto front shapes (the simplex-like, the inverted, the disconnected, the degenerated, the scaled, the mixed, the high dimensional). The experimental results have shown the superiority and versatility of the proposed algorithm.**

***Index Terms*—multi-objective optimization, many-objective optimization, evolutionary algorithm, decomposition-based, archive maintenance, weight vector adaptation**

I. Introduction

In many real-world applications, a decision-maker often needs to handle different conflicting objectives. Problems with more than one objective are called multi-objective optimization problems (MOPs). If an MOP has more than three objectives, it is often referred to as many-objective problem (MaOP). Multi-objective evolutionary algorithms (MOEAs) have been developed to solve these problems, which can be classified into three categories: Pareto dominance-based, decomposition-based and indicator-based MOEAs [1].

MOEA/D [2] is a general decomposition-based MOEA framework, where the decomposition approaches are used to decompose an MOP into a number of single-objective (or multi-objective) optimization subproblems. Recently, the MOEA/D framework has achieved great success and received much attention. However, since the diversity of the solutions is associated explicitly with the weight vectors, the quality of the non-dominated solution set generated by decomposition-based algorithms depends heavily on the Pareto front shapes. A set of uniform weight vectors distributed on a simplex often leads to a set of well-distributed solutions on a Pareto front with a simplex-like shape, but struggles to do so for various other shapes (e.g., inverted, degenerated, disconnected and scaled).

It is an open problem to find methods to adapt weight vectors in the evolutionary process for various different Pareto front shapes. Jiang et al. [3] proposed a Pareto-adaptive weight vector MOEA ($pa\lambda$-MOEA/D), which used the external archive information to sample the regression curve of the weight vectors. The weakness of $pa\lambda$-MOEA/D is the assumption of symmetry and continuity of the Pareto front, and $pa\lambda$-MOEA/D could not obtain solutions with good distribution for disconnected problems. Qi et al. [4] proposed an adaptive weight adjustment (MOEA/D-AWA) for irregular Pareto fronts. In MOEA/D-AWA, an external elite population is applied to help to add new weight vectors into sparse regions of the complex front. The weight vector adjustment is only operated during the last 20% of generations in MOEA/D-AWA, which may deteriorate the convergence of the algorithm. Lucas et al. [5] proposed the MOEA/D with uniformly randomly adaptive weights (MOEA/D-URAW) for different Pareto fronts. MOEA/D-URAW adopted the same weight vector adjustment strategy in MOEA/D-AWA but with different weight vector update frequency. Both MOEA/D-URAW and MOEA/D-AWA used the distance as criteria to select sparse solutions without normalizing the objectives, they could face difficulties in dealing with disparately scaled problems. Jain and Deb [6] presented an adaptive version of NSGA-III [7] (A-NSGA-III) for irregular fronts. In A-NSGA-III, the reference with an empty niche is deleted and the randomly new reference points are added into crowded reference points. Cheng et al. [8] introduced an adaptive version of RVEA (RVEA*) for irregular fronts. In RVEA*, two reference point sets have existed, one of which stores a set of uniformly distributed reference points and the other one is randomly adding new reference points based on the information of the current population. Li and Yao [9] put forward adaptive weights for any Pareto front shape in decomposition-based algorithms (AdaW). AdaW could achieve good performance for different shapes. However, there still exists a chance that the dominance resistant solutions in the archive may deteriorate the convergence, especially for MaOPs.

Following the above ideas, we present a novel archive maintenance for adapting weight vectors in the decomposition-based multi-objective evolutionary algorithms (called AMAWV). The main contributions of this work are summarized as follows: 1) A novel archive maintenance strategy is proposed for deleting the dominance resistant solutions and retaining good diversity. 2) An efficient weight vector adaptation method is presented for various Pareto front shapes. 3) The proposed AMAWV is competitive compared with other five state-of-the-art algorithms on test problems with a variety of Pareto front shapes.

The rest of this paper is organized as follows. Section II briefly introduces the background and the details of the proposed AMAWV are described in Section III. Section IV presents the experimental results of AMAWV compared with other state-of-the-art algorithms. Finally, Section V summarizes and presents the conclusion and future work.

II. Background

In this section some background concepts are provided.

A. Multi-objective Optimization

A multi-objective optimization problem (MOP) can be defined as follows:

$$\begin{array}{c} \min \ \ F(x) = (f_1(x), f_2(x), \cdots, f_m(x))^T \\ \text{subject to} \ \ x \in \Omega \subseteq R^n, \end{array} \tag{1}$$

where Ω is the decision space and x is a solution; $F : \Omega \to \Theta \subseteq R^m$ denotes the m-dimensional objective vector and Θ is the objective space.

A solution x^0 is said to dominate another solution x^1, denoted by $x^0 \prec x^1$, if

$$\begin{cases} f_i(x^0) \le f_i(x^1), & \forall i \in \{1, 2, \cdots, m\} \\ f_j(x^0) < f_j(x^1), & \exists j \in \{1, 2, \cdots, m\}. \end{cases} \tag{2}$$

A solution x^0 is called a Pareto optimal solution, if $\neg \exists x^1 : x^1 \prec x^0$.

The set of Pareto optimal solutions is defined as $PS = \{x^0 \,|\neg \exists x^1 \prec x^0\}$.

The Pareto optimal solution set in the objective space is called Pareto front (PF).

B. Modified Tchebycheff Approach

The modified Tchebycheff approach is defined as follows:

$$\min_{x\in\Omega} g(x|\lambda, z^*) = \min_{x\in\Omega} \max_{1\le i\le m} \left\{ \frac{1}{\lambda_i} |f_i(x) - z_i{}^*| \right\}, \tag{3}$$

where $\lambda = (\lambda_1, \lambda_2, \cdots, \lambda_m)^T$ is a weight vector and $\sum_{i=1}^m \lambda_i = 1, \lambda_i \ge 0, i = 1, 2, \cdots, m$. $z^* = (z_1^*, z_2^*, \cdots, z_m^*)^T$ is the reference point, for each objective, $z_i^* = \min\{f_i(x) | x \in \Omega\}, i = 1, 2, \cdots, m$. It has been proven that the modified Tchebycheff approach can produce more uniformly distributed solutions [4]. We adopt this approach in this paper.

III. The proposed AMAWV

This section presents the details of the proposed algorithm.

A. General Framework

The general framework of the proposed AMAWV is described in Algorithm 1. First, the population $P \leftarrow \{x^1, x^2, \cdots, x^N\}$ are randomly generated in the whole decision space, then the reference point $z^* = (z_1^*, z_2^*, \cdots, z_m^*)^T$ is initialized. After that we can initialize the archive A by adding the non-dominated solutions from the population P. A set of uniformly random weight vectors $\Lambda = \{\lambda^1, \lambda^2, \cdots, \lambda^N\}$ are generated as follows [5].

First, 5000 weight vectors are uniformly randomly generated for forming the set Λ_1. Λ is initialized as the set containing all the weight vectors $(1\ 0\ \ldots\ 0\ 0), (0\ 1\ \ldots\ 0\ 0), \ldots, (0\ 0\ \ldots\ 0\ 1)$. Second, the weight vector in Λ_1 with the largest distance to Λ is found, added to Λ, and removed from Λ_1. This process is repeated until the size of Λ is N.

The AMAWV adopts the uniformly random weight vectors method instead of Das and Dennis's systematic approach [10]. The advantage of this approach is that the population size is flexible, which is independent on the number of objectives. In the light of the generated weight vectors, the neighborhood set of subproblem i as $B(i) = \{i_1, \cdots, i_T\}$ can be obtained by computing the Euclidean distance, where T is the neighborhood size. Steps 8-28 in Algorithm 1 are iterated until the termination criterion is met. At each iteration, the mating pool is allowed to be selected from the whole population with a low probability $1 - \delta$. The widely used simulated binary crossover (SBX) and polynomial mutation are randomly selected mating solutions from E to generate offspring y. Then offspring y is used to update the reference point, the population and the archive. When the size of the archive exceeds the predefined limit size (N_A), a novel strategy is adopted to maintain the archive (Steps 20-22 in Algorithm 1), which will be introduced in Section III-B. When the frequency of updating the weight vectors is satisfied with the designed requirements, the weight vector adaptation method is conducted (Steps 23-26 in Algorithm 1), which will be presented in Section III-C.

B. Archive Maintenance

In AMAWV, we use the Pareto dominance to select the non-dominated solutions to be added to the archive, when the size of the archive exceeds the pre-set capacity (N_A). A novel archive maintenance strategy is applied to remove some dominance resistance solutions and some other solutions with poor distribution. The archive maintenance strategy is presented in Algorithm 2.

Aiming at dealing with the scaled problems with different objectives ranges, the adaptive normalization approach in NSGA-III [7] is adopted to normalize the solutions in the archive (Step 2 in Algorithm 2). The main idea of this approach is to find some extreme solutions to construct the hyperplane. The extreme solutions are determined by minimizing the achievement scalarizing function (ASF):

$$ASF(x, w) = \max_{i=1}^{m} \frac{f_i(x) - z_i^*}{w_i}, \quad for\ x \in S_t, \tag{4}$$

Algorithm 1: Framework of AMAWV

Input: A set of weight vectors $\Lambda \leftarrow \{\lambda^1, \lambda^2, \cdots, \lambda^N\}$, the maximum number of generations $t_{\max}$
Output: The final population P

Initialize the population $P \leftarrow \{x^1, x^2, \cdots, x^N\}$;
Initialize the reference point $z^* \leftarrow (z_1^*, z_2^*, \cdots, z_m^*)^T$;
Initialize the archive A;
for $i = 1 : N$ **do**
    $B(i) \leftarrow \{i_1, i_2, \cdots, i_T\}$, where $\lambda^{i_1}, \lambda^{i_2}, \cdots, \lambda^{i_T}$ are T closest weight vectors to λ^i;
end
$t \leftarrow 1$;
while $t < t_{\max}$ **do**
    for $i = 1 : N$ **do**
        if $uniform(0,1) < \delta$ **then**
            $E \leftarrow B(i)$;
        else
            $E \leftarrow \{1, 2, \ldots, N\}$;
        end
        $y = \text{offspring\_creation}(E)$;
        $z^* = \text{Update\_Ideal\_Point}(y, z^*)$;
        $P = \text{Update\_Population}(y, z^*, \Lambda, P)$;
        $A = \text{Update\_Archive}(y, A)$;
    end
    if $|A| > N_A$ **then**
        Maintain the archive A;
    end
    if $t > t_{\max} \times 10\% \wedge t == t_{\max} \times 5\% \wedge t < t_{\max} \times 90\%$ **then**
        $\Lambda = \text{Weight\_Vector\_Adaption}(t, P, A, \Lambda)$;
        Update the neighborhood set of each weight vector of Λ;
    end
    $t = t + 1$;
end

where S_t represents the current population. w is the axis direction, $w_i = 10^{-6}$ when it is zero. After m extreme solutions have been adopted, they are used to construct a hyperplane and the intercept a_j of the j-th objective axis on the hyperplane can be computed. Then the solutions are normalized as follows:

$$\bar{f}_j(x) = \frac{f_j(x) - z_j^*}{a_j}, \tag{5}$$

where $f_j(x)$ is the j-th objective value of solution x, $\bar{f}_j(x)$ is the normalized objective value.

According to the non-dominated solutions in the archive during the evolution process, we can estimate the shape of PF (Step 3 in Algorithm 2). The PF shape can guide the search direction [11]. First, the m solutions in the archive closest to the m-dimensional vector $V = (1, 1, \ldots, 1)$ are identified

Algorithm 2: Archive Maintenance

Input: The archive A ($|A| > N_A$)
Output: The new archive $newA$

1 $newA \leftarrow \phi$;
2 $A \leftarrow \text{normalization}(A)$;
3 $r \leftarrow estimateShape(A)$;
4 Set the reference point z^* to z^{nad} if $r < 1.1$, or itself otherwise;
5 $\{S+, S-\} \leftarrow classificationByHypercube(A)$;
6 **if** $|S+| > N_A$ **then**
7   /*——— $newA \leftarrow \text{selection}(S+, S-)$ ———*/
8   Add m extreme solutions into $newA$ and remove them from $S+$;
9   **repeat**
10   Add into $newA$ the solution in $S+$ that has the maximum angle to $newA$;
11   **until** $|A| = N_A$
12 **else**
13     $newA = S+$;
14 **end**

based on the angle between the non-dominated solutions and vector V. Then the ratio

$$r = \frac{\overline{d}}{d^{\perp}} \tag{6}$$

is used to estimate the PF shape, where $\overline{d}$ is the average Euclidean distance from m closest solutions to the coordinate origin, and $d^{\perp}$ is the Euclidean distance from coordinate origin to the hyperplane $\sum_{i=1}^{m} f_i = 1$. Since $d^{\perp} = \frac{|-1|}{\sqrt{m}} = \frac{|1|}{\sqrt{m}}$, we obtain

$$r = \overline{d} \times \sqrt{m}. \tag{7}$$

Therefore, the shape of PF can be estimated as convex (if $r < 0.9$), linear (if $r \in [0.9, 1.1]$), and concave (if $r > 1.1$).

According to the estimated PF shape, we set the reference point z^* to z^{nad} if $r < 1.1$, or itself otherwise (Step 4 in Algorithm 2). The nadir point z^{nad} is usually estimated to be $z^{nad} = z^{\max} = (z_1^{\max}, z_2^{\max}, \cdots, z_m^{\max})$. But in this paper, we set the $z^{nad} = I = (1, 1, \cdots, 1)$. Then the solutions in the archive can be divided into two repositories ($S+$ and $S-$) inside and outside the hypercube (Step 5 in Algorithm 2), which is bounded by z^* and z^{nad}.

The new archive ($newA$) will select the non-dominated solutions from the $S+$ and $S-$ (Steps 6-14 in Algorithm 2). When the size of $S+$ is beyond N_A, the m extreme solutions are added into $newA$ and removed from $S+$. Then we use the one by one adding solution procedure to choose the solutions from $S+$. At each stage, the solution in $S+$ that has the maximum angle to $newA$ will be placed into the $newA$ [12]. The above operation is repeated until the size of the new archive is equal to N_A. When the size of $S+$ is smaller than N_A, all the solutions in $S+$ will be added into $newA$. In the literature [11] the Pareto-adaptive reference points are used to calculate fitness values and the union population selects both the dominated and non-dominated solutions based on the fitness values. Also, the solutions outside the hypercube can be added to the population. In this paper, only the non-dominated solutions inside the hypercube are added to the archive to adjust the weight vectors for various PF shapes, which can avoid the dominance resistant solutions to improve the convergence.

The reference point set ($z^* \leftarrow z^{nad}$ if $r < 1.1$) plays an important role in the archive maintenance strategy, we will explain the process using the 2-dimensional example as shown in Fig. 1. Through the adaptive normalization procedure and classification by the hypercube, we can delete some dominance resistant solutions such as the solution C (or D), if C has an extremely poor value in the second objective (f_2) but has optimal value in the first objective (f_1). In this way, some extremely poor solutions in the archive can be deleted and will not guide the solutions to search in the wrong direction. This improves the convergence of the algorithm. On the other hand, in the one by one adding solution procedure, we use the maximum angle as the criteria to select the solution with good distribution. If the PF shape is very convex, there are many solutions along the coordinate (such as A and B), the angle between the vector $\overrightarrow{z^*A}$ and $\overrightarrow{z^*B}$ is close to zero. At this time, it is easy to ignore the solutions along the coordinate and finally obtain the non-dominated solution set with poor distribution. But the angle between the vector $\overrightarrow{z^{nad}A}$ and $\overrightarrow{z^{nad}B}$ is clear in this situation, we can differentiate these solutions by substituting the reference point z^* with z^{nad}. Therefore, we use different reference points (z^* or z^{nad}) based on the different PF shapes in order to obtain a good spread of non-dominated solutions in the archive.

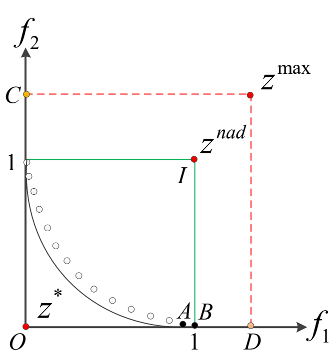

Fig. 1. Illustration of the reference points.

C. Weight Vector Adaptation

The non-dominated solutions in the archive can reflect the PF shape to some extent, hence it is necessary to use the information of the archive to guide the search direction of the decomposition-based algorithms for solving problems with irregular Pareto fronts. The weight vector addition and deletion method is an effective way to adapt the weight vectors for various PF shapes. We select an efficient strategy to add and delete some weight vectors.

Inspired by the literature [9], we first compare the population with the archive to find the undeveloped solutions. If a solution in the archive is located in a niche which has no solution in the population, the solution is considered as an undeveloped solution. The radius of the niche is set to the median of the distances from all the solutions to their closest solution in the archive. After finding all the undeveloped solutions we then compute the corresponding weight vectors of these solutions. Formally, let $z^* \leftarrow (z_1^*, z_2^*, \cdots, z_m^*)^T$ be the reference point and $\lambda = (\lambda_1, \lambda_2, \cdots, \lambda_m)^T$ be the optimal weight vector to a solution q in the modified Tchebycheff approach. Then it holds that

$$\frac{f_1(q) - z_1^*}{\lambda_1} = \frac{f_2(q) - z_2^*}{\lambda_2} = \cdots = \frac{f_m(q) - z_m^*}{\lambda_m}. \quad (8)$$

Since $\lambda_1 + \lambda_2 + \cdots + \lambda_m = 1$, we get

$$\lambda = \left(\frac{f_1(q) - z_1^*}{\sum_{i=1}^m f_i(q) - z_i^*}, \cdots, \frac{f_m(q) - z_m^*}{\sum_{i=1}^m f_i(q) - z_i^*} \right). \quad (9)$$

After obtaining the undeveloped solutions and their corresponding weight vectors, we need to determine whether the found undeveloped solutions are promising or not. For each of these undeveloped weight vectors, we find its neighboring weight vectors as well as their associated solutions in the population. Let q be an undeveloped solution in the archive and λ^q be its corresponding weight vector. Let λ^p be one of the neighboring weight vectors of λ^q and p be its associated solution in the population. The solution q outperforms p if

$$g(q, \lambda^q) < g(p, \lambda^q), \quad (10)$$

or

$$g(q, \lambda^q) = g(p, \lambda^q) \ \ and \ \sum_{i=1}^m f_i(q) < \sum_{i=1}^m f_i(p), \quad (11)$$

where $g()$ denotes the modified Tchebycheff function, $f_i()$ denotes the i-th objective function, and m is the number of objectives. If the undeveloped solution q outperforms all of its neighboring solutions in the population on the basis of λ^q, q will be considered the promising solution and added to the population.

After placing the undeveloped and promising solutions with their corresponding weight vectors into the population, the number of solutions and weight vectors in the new population may be beyond the predefined size N, hence the weight vector deletion operation needs to be applied. The one by one removing solution procedure is designed to delete the redundant population solutions and corresponding weight vectors. That means at each time, the solution (along with its weight vector) in the population which has minimum distance to another solution is chosen to be deleted. If there are several solutions with the same minimum distance the second smallest distance will be considered and so forth [13]. The deletion process will repeat until the number of weight vectors is equal to N.

The frequency of updating the weight vectors plays an important part of the performance of decomposition-based algorithms. The frequent change of weight vectors may deteriorate the convergence of these algorithms. In AMAWV the weight vectors do not change during the first 10% and last 10% generations for retaining the same search direction to improve convergence. The weight vector adaptation operation is conducted every 5% of the total generations in the middle evolution process.

The weight vector deletion process and the frequency of updating the weight vectors in the proposed algorithm is different from the literature [9]. The weight vector deletion process is more efficient and the frequency of updating the weight vectors can improve the whole convergence. In addition, the novel archive maintenance strategy for avoiding the dominance resistant solutions is used to maintain the archive, which can guide the weight vectors addition and deletion process to balance the convergence and diversity.

D. Computational Complexity

In the proposed AMAWV, the major computational costs are the iteration process in Algorithm 1. For one generation of AMAWV, Step 15 needs $O(N)$ operations to produce the offspring. Step 16 needs $O(mN)$ comparisons to update the reference point. Step 17 performs $O\left(mN^2\right)$ operations to update the population at the worst case. Step 18 needs $O\left(mNN_A\right)$ comparisons to update the archive. Step 21 requires $O\left(mN_A^2\right)$ operations to maintain the archive.

On average, the weight vector adaptation needs $O\left(N_A^2 \log N_A\right)$ operations and it needs $O\left(N^2\right)$ comparisons to update the neighborhood set. Taking into account all of the above computations, the overall computational complexity at one generation of AMAWV is $\max\left\{O\left(N_A^2 \log N_A\right), O\left(mN_A^2\right)\right\}$.

IV. The Experimental Studies

In this section empirical experiments are conducted on MOPs and MaOPs with different PF shapes to compare AMAWV with other five state-of-the-art algorithms.

A. Experimental Design

Five state-of-the-art algorithms, MOEA/D [2], MOEA/D-AWA [4], NSGA-III [7], RVEA [8] and VaEA [12] are chosen to evaluate the performance of AMAWV. In MOEA/D, the modification Tchebycheff approach is also applied.

The test problems are chosen from widely used benchmark test suites. They are categorized into eight groups according to different PF shape properties [9]: simplex-like (DTLZ1, DTLZ2, CDTLZ2 and MaF3), inverted (IDTLZ1 and IDTLZ2), highly nonlinear (SCH1 and FON), disconnected (ZDT3 and DTLZ7), degenerated (DTLZ5 and DTLZ6), scaled (SDTLZ1 and SDTLZ2), mixed (SCH2 and MaF4) and high-dimensional (DTLZ2-10 and IDTLZ1-10). The test problems in the first seven groups are 2- or 3-objective and the last group are 10-objective.

The inverted generational distance (IGD) [14] and hypervolume (HV) [14] are adopted to evaluate the performance of the compared algorithms. The smaller IGD and larger HV means better.

The neighborhood size T is set to $0.1N$, the probability δ of selecting from the mating pool is set to 0.9, the capacity N_A of the archive is $2N$. The crossover probability and distribution index of SBX are set to $p_c = 1$ and $\eta_c = 20$, respectively. For polynomial mutation, the mutation probability and distribution index are set to $p_m = 1/n$ and $\eta_m = 20$, where n is the number of decision variables.

The population size in decomposition-based algorithms cannot be arbitrary. For a fair comparison, we set the population size N to 100, 105 and 275 for the 2-, 3- and 10-objective problems, respectively. The maximal number of generations is set to 1000 for all the problems. Each algorithm is executed 30 times independently on each test instance, and the average and standard deviation of the metric values are recorded. The Wilcoxon rank sum test at a 5% significance level is used to compare the experimental results, where the symbol '$+$', '$-$' and '$\approx$' denotes that the result of another algorithm is significantly better, significantly worse and similar to that obtained by AMAWV, respectively.

B. Experimental Results

Table I presents the IGD metric values obtained by MOEA/D, MOEA/D-AWA, NSGA-III, RVEA, VaEA and AMAWV on the test problems with different PF shapes. The proposed algorithm has achieved the best performance on 13 of 18 test instances, while the number of best results obtained by MOEA/D, MOEA/D-AWA, NSGA-III, RVEA, VaEA are 1, 1, 1, 2 and 0, respectively. It can be seen that the proposed AMAWV has a clear advantage over other compared algorithms with regard to the IGD metric. Fig. 2 and Fig. 3 show the final non-dominated solution set with the median IGD value obtained by MOEA/D, MOEA/D-AWA, NSGA-III, RVEA, VaEA and AMAWV on 3-objective MaF3 and 3-objective DTLZ7. For 3-objective MaF3, the PF shape is convex, MOEA/D, NSGA-III, RVEA and VaEA focus on the center part of PF, MOEA/D-AWA could tune the diversity of the solutions a little, AMAWV can achieve the best balance between convergence and diversity. For 3-objective DTLZ7 the PF shape is disconnected, while VaEA and AMAWV can obtain the solution set with good distribution compared with the other four algorithms.

Table II presents the HV metric values obtained by MOEA/D, MOEA/D-AWA, NSGA-III, RVEA, VaEA and AMAWV on the test problems with different PF shapes. The proposed algorithm has achieved the best performance on 12 of 18 test instances, while the number of best results obtained by MOEA/D, MOEA/D-AWA, NSGA-III, RVEA, VaEA are 1, 1, 2, 2 and 0, respectively. It can be observed that propose AMAWV outperforms the other five compared algorithms a lot with respect to HV metric. Fig. 4 and Fig. 5 shows the final non-dominated solution set with the median HV value obtained by MOEA/D, MOEA/D-AWA, NSGA-III, RVEA, VaEA and AMAWV on 3-objective MaF4 and 10-objective IDTLZ1. For 3-objective MaF4, it has an inverted badly scaled

TABLE I
THE IGD VALUES OBTAINED BY MOEA/D, MOEA/D-AWA, NSGA-III, RVEA, VAEA AND AMAWV ON TEST PROBLEMS.

Problem	MOEA/D	MOEA/D-AWA	NSGA-III	RVEA	VaEA	AMAWV
DTLZ1	1.8983e-2 (9.28e-6) +	1.9741e-2 (4.76e-4) ≈	1.8989e-2 (2.69e-5) +	1.8981e-2 (9.20e-6) +	2.7894e-2 (8.25e-3) −	1.9675e-2 (1.69e-4)
DTLZ2	5.0315e-2 (6.41e-6) +	5.0674e-2 (1.91e-4) +	5.0301e-2 (1.23e-6) +	5.0301e-2 (1.35e-6) +	5.3517e-2 (6.94e-4) −	5.1491e-2 (4.15e-4)
CDTLZ2	4.3371e-2 (4.40e-6) −	3.3688e-2 (1.20e-3) −	4.3347e-2 (1.68e-4) −	3.8134e-2 (4.14e-4) −	5.8261e-2 (4.21e-3) −	3.1819e-2 (4.98e-4)
MaF3	4.3944e-2 (5.50e-4) −	3.6480e-2 (1.24e-3) −	4.4005e-2 (6.12e-4) −	3.9123e-2 (9.02e-4) −	1.0286e-1 (6.82e-2) −	3.2005e-2 (6.31e-4)
IDTLZ1	3.2323e-2 (7.86e-6) −	2.0001e-2 (1.47e-4) −	2.7614e-2 (5.36e-4) −	4.5021e-2 (1.16e-2) −	2.9573e-2 (1.21e-2) −	1.9765e-2 (1.41e-4)
IDTLZ2	9.7749e-2 (1.85e-5) −	5.2547e-2 (5.48e-4) −	6.8724e-2 (2.75e-3) −	7.6365e-2 (8.07e-4) −	6.9191e-2 (2.00e-3) −	5.2189e-2 (4.33e-4)
SCH1	4.7648e-2 (9.46e-5) −	1.7135e-2 (7.35e-5) ≈	4.7622e-2 (1.34e-4) −	4.4682e-2 (2.16e-4) −	5.5476e-2 (6.59e-3) −	1.7146e-2 (8.56e-5)
FON	3.5827e-3 (1.04e-5) +	3.6737e-3 (2.93e-5) +	3.5951e-3 (7.79e-6) +	4.6246e-3 (3.89e-4) −	4.6102e-3 (9.91e-5) −	3.9019e-3 (7.87e-5)
ZDT3	1.0967e-2 (4.34e-5) −	4.8912e-3 (6.16e-5) −	1.0013e-2 (9.94e-3) −	8.4826e-3 (9.45e-4) −	1.3966e-2 (1.29e-2) −	4.7013e-3 (1.12e-4)
DTLZ7	2.4607e-1 (1.85e-1) −	1.3131e-1 (9.16e-2) −	7.1013e-2 (3.20e-3) −	1.0369e-1 (2.33e-3) −	5.9058e-2 (1.32e-3) −	5.3663e-2 (7.62e-4)
DTLZ5	1.8610e-2 (2.32e-6) −	4.9479e-3 (8.68e-5) −	1.2185e-2 (1.64e-3) −	5.9830e-2 (2.74e-3) −	4.6950e-3 (1.33e-4) −	4.0480e-3 (6.67e-5)
DTLZ6	1.8612e-2 (1.83e-6) −	4.8461e-3 (1.26e-4) −	1.6910e-2 (2.32e-3) −	6.3984e-2 (1.12e-2) −	4.3482e-3 (1.50e-4) −	4.0338e-3 (4.62e-5)
SDTLZ1	2.7841e+0 (5.43e-3) −	2.1104e+0 (1.14e+0) −	9.6234e-1 (2.85e-2) −	8.9768e-1 (4.91e-2) −	8.1286e-1 (3.72e-1) −	6.0928e-1 (1.68e-2)
SDTLZ2	5.2277e+0 (5.55e-4) −	1.8208e+0 (3.04e-1) −	1.4903e+0 (3.61e-4) ≈	1.4973e+0 (8.44e-3) ≈	1.5179e+0 (8.60e-2) ≈	1.5013e+0 (6.69e-2)
SCH2	1.0517e-1 (8.89e-5) −	2.1807e-2 (5.20e-4) −	3.2901e-2 (3.76e-3) −	4.4899e-2 (2.58e-4) −	3.2417e-2 (3.11e-3) −	2.0903e-2 (1.98e-4)
MaF4	5.8395e-1 (1.55e-3) −	2.5536e-1 (5.95e-3) −	3.2311e-1 (1.63e-2) −	3.8401e-1 (1.06e-1) −	4.5213e-1 (2.27e-1) −	2.3830e-1 (3.08e-3)
DTLZ2-10	4.5187e-1 (2.73e-2) −	4.2634e-1 (1.46e-2) −	4.4224e-1 (4.33e-2) −	4.2101e-1 (3.62e-4) −	4.1345e-1 (2.02e-3) −	4.0212e-1 (7.49e-3)
IDTLZ1-10	2.3840e-1 (6.38e-3) −	1.7218e-1 (2.55e-2) −	1.4086e-1 (3.51e-3) −	2.6238e-1 (4.61e-2) −	1.1330e-1 (1.42e-2) −	1.1279e-1 (1.86e-3)
+/ − / ≈	3/15/0	2/14/2	3/14/1	2/15/1	0/17/1	

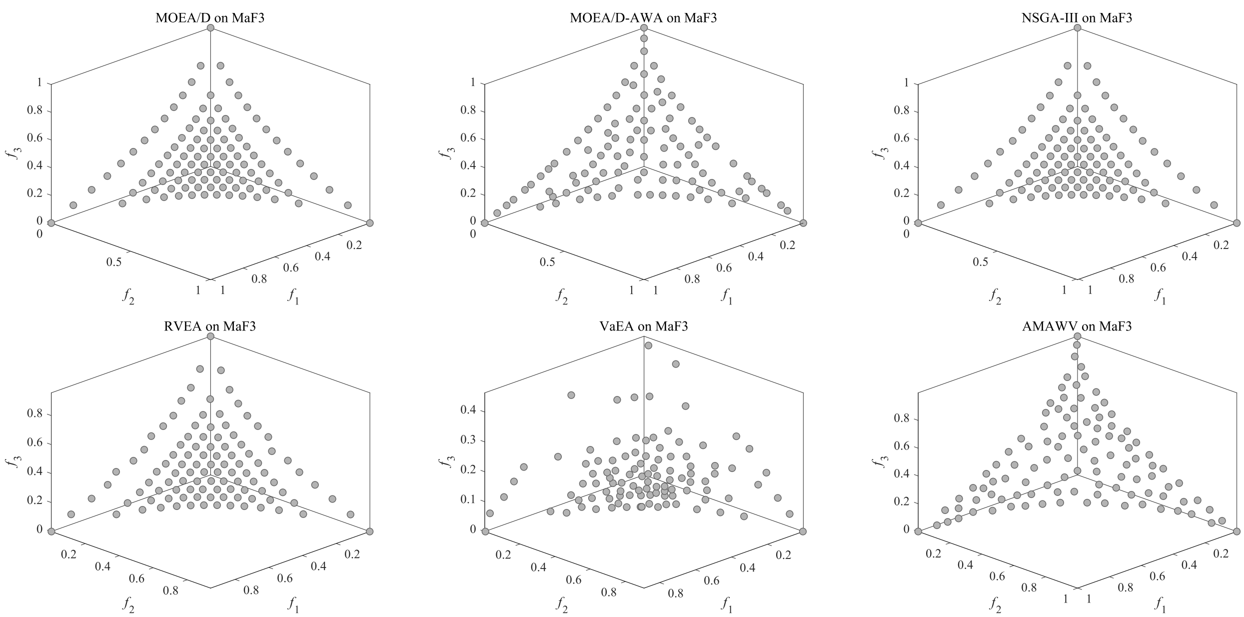

Fig. 2. The non-dominated solution set with the median IGD value obtained by MOEA/D, MOEA/D-AWA, NSGA-III, RVEA, VaEA and AMAWV on 3-objective MaF3.

PF shape, MOEA/D could only find a part of PF, the solution set of VERA seems to be sparse, AMAWV can obtain a good spread of the solutions compared with MOEA/D-AWA, NSGA-III and VaEA. For 10-objective IDTLZ1, it has a high-dimensional and inverted PF shape, RVEA cannot find the true PF, MOEA/D and MOEA/D-AWA could only find a small part of solutions, NSGA-III fails to cover the whole PF, VaEA has some solutions which fail to converge to the true PF. Only AMAWV can obtain a solution set to cover the whole PF with good convergence and diversity.

V. CONCLUSION AND FUTURE WORK

In this paper, we propose a novel archive maintenance for adapting weight vectors to make the decomposition-based multi-objective evolutionary algorithms solve MOPs and MaOPs with different PF shapes. The novel archive maintenance strategy can delete some dominance resistant solutions to improve the convergence and the one by one adding solution procedure can retain good diversity. The weight vector adaptation method helps the decomposition-based algorithms to be suitable for different problems with various PF properties. The experimental results have demonstrated the superiority and versatility of the proposed AMAWV.

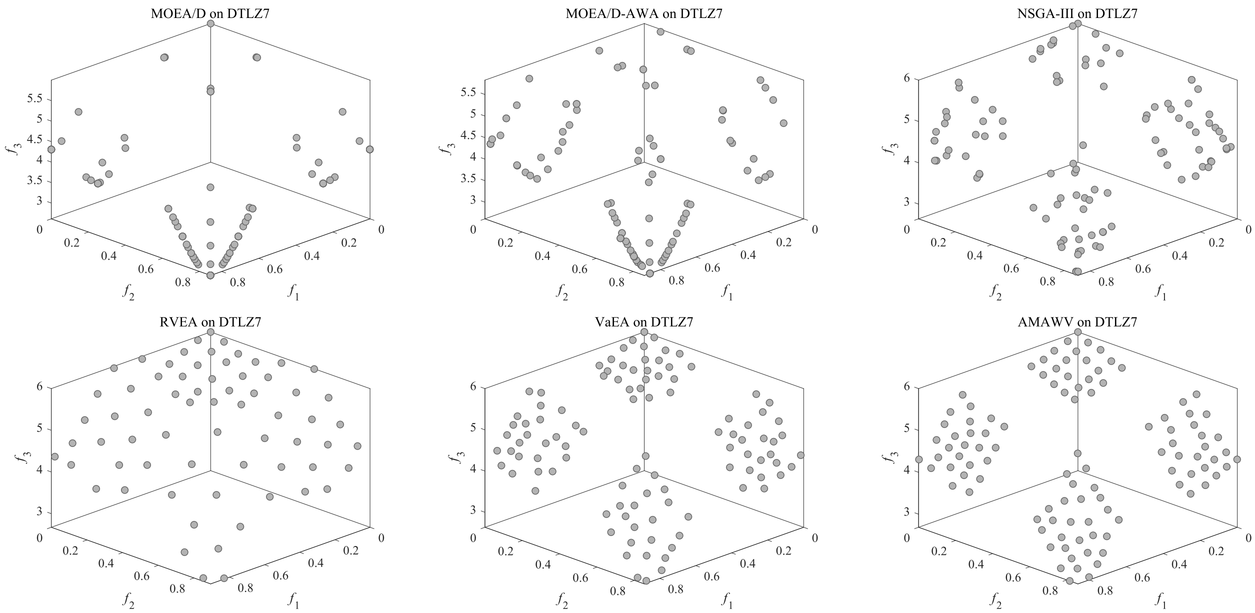

Fig. 3. The non-dominated solution set with the median IGD value obtained by MOEA/D, MOEA/D-AWA, NSGA-III, RVEA, VaEA and AMAWV on 3-objective DTLZ7.

TABLE II
THE HV VALUES OBTAINED BY MOEA/D, MOEA/D-AWA, NSGA-III, RVEA, VAEA AND AMAWV ON TEST PROBLEMS.

Problem	MOEA/D	MOEA/D-AWA	NSGA-III	RVEA	VaEA	AMAWV
DTLZ1	8.4428e-1 (1.30e-4) +	8.3738e-1 (7.26e-3) −	8.4420e-1 (3.31e-4) +	8.4431e-1 (1.33e-4) +	8.2025e-1 (1.92e-2) −	8.4286e-1 (4.05e-4)
DTLZ2	5.6302e-1 (4.11e-6) +	5.6491e-1 (3.92e-4) +	5.6302e-1 (6.40e-6) +	5.6301e-1 (1.99e-5) +	5.5825e-1 (1.37e-3) −	5.6162e-1 (9.39e-4)
CDTLZ2	9.5967e-1 (3.52e-6) −	9.6306e-1 (3.28e-4) −	9.5969e-1 (7.09e-5) −	9.6054e-1 (5.25e-4) −	9.5481e-1 (1.55e-3) −	9.6363e-1 (1.98e-4)
MaF3	9.5889e-1 (6.72e-4) −	9.6187e-1 (7.05e-4) −	9.5863e-1 (6.55e-4) −	9.6045e-1 (1.01e-3) −	9.1572e-1 (5.99e-2) −	9.6287e-1 (5.19e-4)
IDTLZ1	2.0345e-1 (1.01e-4) −	2.2342e-1 (3.22e-4) ≈	2.1113e-1 (8.86e-4) −	1.7968e-1 (1.62e-2) −	2.0726e-1 (1.74e-2) −	2.2353e-1 (3.10e-4)
IDTLZ2	5.0850e-1 (1.58e-5) −	5.3777e-1 (6.64e-4) −	5.2007e-1 (2.86e-3) −	5.1442e-1 (9.90e-4) −	5.3035e-1 (1.24e-3) −	5.3953e-1 (6.11e-4)
SCH1	8.5798e-1 (4.28e-6) −	8.5913e-1 (4.27e-5) ≈	8.5798e-1 (5.79e-6) −	8.5810e-1 (9.65e-6) −	8.5688e-1 (5.55e-4) −	8.5914e-1 (4.07e-5)
FON	4.3160e-1 (2.93e-5) +	4.3151e-1 (3.87e-5) ≈	4.3151e-1 (2.76e-5) ≈	4.2916e-1 (7.22e-4) −	4.2963e-1 (2.01e-4) −	4.3151e-1 (4.84e-5)
ZDT3	5.8139e-1 (1.29e-5) −	5.8320e-1 (2.61e-5) −	5.8217e-1 (3.33e-2) −	5.7843e-1 (1.11e-3) −	5.8321e-1 (4.34e-2) ≈	5.8330e-1 (5.12e-5)
DTLZ7	2.5390e-1 (1.82e-2) −	2.6177e-1 (1.12e-2) −	2.7184e-1 (1.63e-3) −	2.6020e-1 (2.47e-3) −	2.7779e-1 (7.25e-4) −	2.8080e-1 (2.75e-4)
DTLZ5	1.9268e-1 (2.21e-6) −	1.9963e-1 (9.10e-5) −	1.9466e-1 (1.05e-3) −	1.6334e-1 (2.50e-3) −	1.9967e-1 (1.24e-4) −	2.0016e-1 (1.87e-4)
DTLZ6	1.9268e-1 (1.49e-6) −	1.9966e-1 (8.29e-5) −	1.9231e-1 (1.66e-3) −	1.5706e-1 (7.70e-3) −	1.9993e-1 (8.08e-5) −	2.0025e-1 (3.85e-5)
SDTLZ1	6.8595e-1 (3.79e-4) −	7.5145e-1 (4.16e-2) −	8.4384e-1 (2.06e-3) +	8.4121e-1 (1.46e-2) −	8.2476e-1 (1.78e-2) −	8.4274e-1 (3.69e-4)
SDTLZ2	4.3510e-1 (1.29e-5) −	5.2712e-1 (1.06e-2) −	5.6302e-1 (6.68e-6) +	5.6256e-1 (2.49e-4) ≈	5.5817e-1 (1.43e-3) −	5.6239e-1 (5.72e-4)
SCH2	6.4666e-1 (5.20e-5) −	6.5528e-1 (4.61e-5) ≈	6.5458e-1 (2.12e-4) −	6.5214e-1 (9.42e-5) −	6.5499e-1 (3.46e-4) −	6.5529e-1 (1.15e-4)
MaF4	4.7320e-1 (1.08e-3) −	5.3039e-1 (1.87e-3) −	5.1973e-1 (4.55e-3) −	5.0494e-1 (2.65e-2) −	4.9773e-1 (5.77e-2) −	5.3759e-1 (8.18e-4)
DTLZ2-10	9.5643e-1 (1.30e-2) +	9.6452e-1 (6.90e-3) +	9.6119e-1 (1.73e-2) +	9.6979e-1 (2.08e-4) +	9.4757e-1 (3.02e-3) +	9.3416e-1 (1.67e-2)
IDTLZ1-10	6.4382-2 (2.88e-4) −	1.6567-1 (2.55e-2) −	9.5142e-2 (6.59e-3) −	3.6636e-2 (4.37e-3) −	3.3557e-1 (1.38e-2) ≈	3.3716e-1 (7.14e-3)
+/ − / ≈	4/14/0	2/12/4	5/12/1	3/14/1	1/15/2	

In the future, some other more efficient weight vector adaptation methods will be studied. The use of AMAWV in practice will also be investigated.

REFERENCES

[1] Y. Tian, R. Cheng, X. Zhang, F. Cheng, and Y. Jin, "An indicator-based multiobjective evolutionary algorithm with reference point adaptation for better versatility," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 4, pp. 609–622, 2017.

[2] Q. Zhang and H. Li, "Moea/d: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on evolutionary computation*, vol. 11, no. 6, pp. 712–731, 2007.

[3] J. Siwei, C. Zhihua, Z. Jie, and O. Yew-Soon, "Multiobjective optimization by decomposition with pareto-adaptive weight vectors," in *2011 Seventh International Conference on Natural Computation*, vol. 3. IEEE, 2011, pp. 1260–1264.

[4] Y. Qi, X. Ma, F. Liu, L. Jiao, J. Sun, and J. Wu, "Moea/d with adaptive weight adjustment," *Evolutionary computation*, vol. 22, no. 2, pp. 231–264, 2014.

[5] L. R. de Farias, P. H. Braga, H. F. Bassani, and A. F. Araújo, "Moea/d with uniformly randomly adaptive weights," in *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, 2018, pp. 641–648.

[6] H. Jain and K. Deb, "An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part ii: handling constraints and extending to an adaptive approach," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 602–622, 2013.

[7] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: solving problems with box constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2013.

[8] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A reference vector

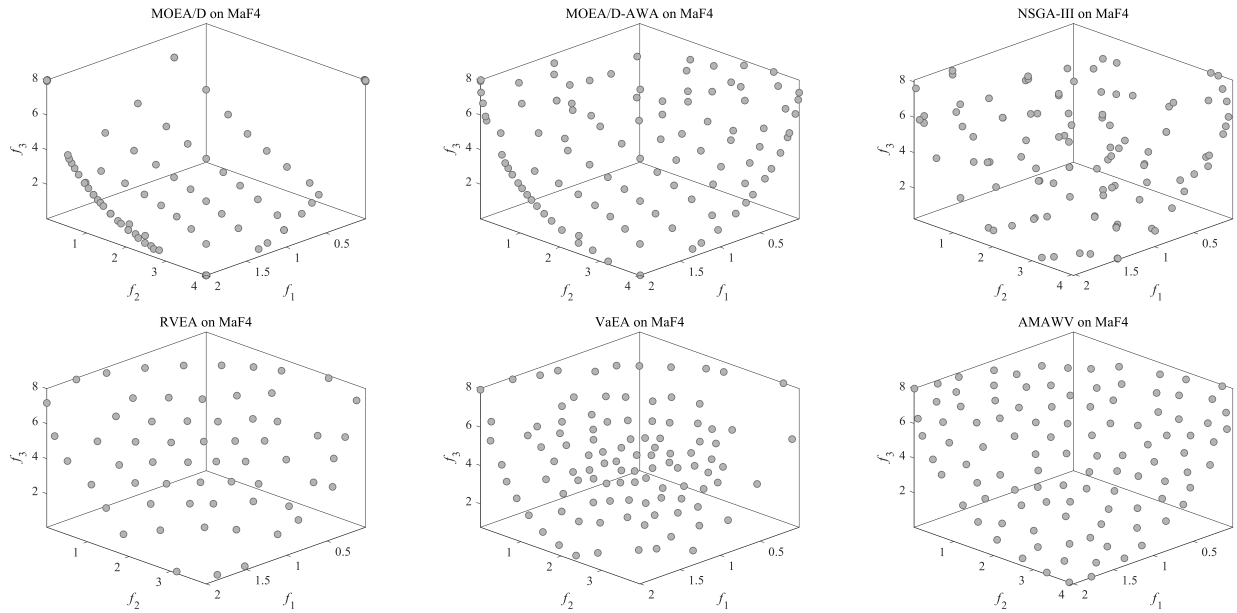

Fig. 4. The non-dominated solution set with the median HV value obtained by MOEA/D, MOEA/D-AWA, NSGA-III, RVEA, VaEA and AMAWV on 3-objective MaF4.

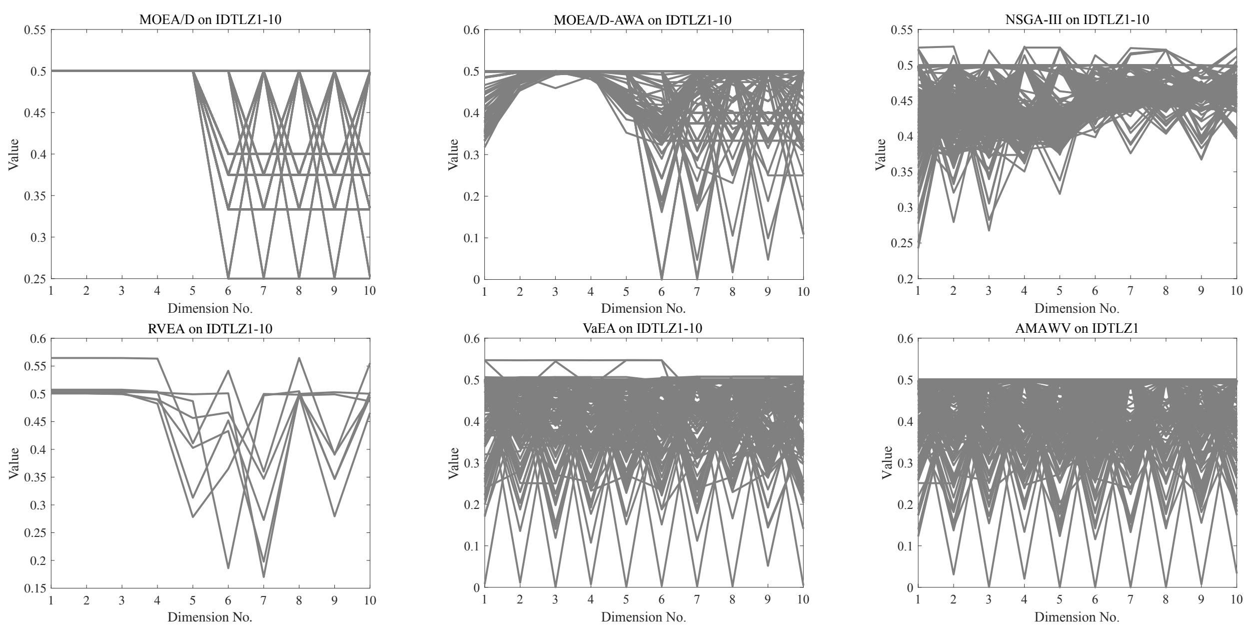

Fig. 5. The non-dominated solution set with the median HV value obtained by MOEA/D, MOEA/D-AWA, NSGA-III, RVEA, VaEA and AMAWV on 10-objective IDTLZ1.

guided evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 5, pp. 773–791, 2016.

[9] M. Li and X. Yao, "What weights work for you? adapting weights for any pareto front shape in decomposition-based evolutionary multi-objective optimisation," *arXiv preprint arXiv:1709.02679*, 2017.

[10] I. Das and J. E. Dennis, "Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems," *SIAM journal on optimization*, vol. 8, no. 3, pp. 631–657, 1998.

[11] Y. Xiang, Y. Zhou, X. Yang, and H. Huang, "A many-objective evolutionary algorithm with pareto-adaptive reference points," *IEEE Transactions on Evolutionary Computation*, 2019.

[12] Y. Xiang, Y. Zhou, M. Li, and Z. Chen, "A vector angle-based evolutionary algorithm for unconstrained many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 1, pp. 131–152, 2016.

[13] E. Zitzler, M. Laumanns, and L. Thiele, "Spea2: Improving the strength pareto evolutionary algorithm," *TIK-report*, vol. 103, 2001.

[14] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. Da Fonseca, "Performance assessment of multiobjective optimizers: An analysis and review," *IEEE Transactions on evolutionary computation*, vol. 7, no. 2, pp. 117–132, 2003.