# MOEA/D with Adaptive Weight Adjustment

**Yutao Qi**                                                    qi_yutao@163.com
School of Computer Science and Technology, Xidian University,
Xi'an, 710071, China; Key Laboratory of Intelligent Perception
and Image Understanding of Ministry of Education of China,
Xidian University, Xi'an, 710071, China

**Xiaoliang Ma**                                          maxiaoliang@yeah.net
School of Computer Science and Technology, Xidian University,
Xi'an, 710071, China; Key Laboratory of Intelligent Perception
and Image Understanding of Ministry of Education of China,
Xidian University, Xi'an, 710071, China

**Fang Liu**                                                      f63liu@163.com
School of Computer Science and Technology, Xidian University,
Xi'an, 710071, China; Key Laboratory of Intelligent Perception
and Image Understanding of Ministry of Education of China,
Xidian University, Xi'an, 710071, China

**Licheng Jiao**                                   lchjiao@mail.xidian.edu.cn
Key Laboratory of Intelligent Perception and Image Understanding of
Ministry of Education of China, Xidian University, Xi'an, 710071, China

**Jianyong Sun**                                           j.sun@abertay.ac.uk
School of Engineering, Computing and Applied Mathematics,
University of Abertay Dundee, Dundee, DD1 1HG, UK

**Jianshe Wu**                                      jshwu@mail.xidian.edu.cn
Key Laboratory of Intelligent Perception and Image Understanding of
Ministry of Education of China, Xidian University, Xi'an, 710071, China

**Abstract**

Recently, MOEA/D (multi-objective evolutionary algorithm based on decomposition) has achieved great success in the field of evolutionary multi-objective optimization and has attracted a lot of attention. It decomposes a multi-objective optimization problem (MOP) into a set of scalar subproblems using uniformly distributed aggregation weight vectors and provides an excellent general algorithmic framework of evolutionary multi-objective optimization. Generally, the uniformity of weight vectors in MOEA/D can ensure the diversity of the Pareto optimal solutions, however, it cannot work as well when the target MOP has a complex Pareto front (PF; i.e., discontinuous PF or PF with sharp peak or low tail). To remedy this, we propose an improved MOEA/D with adaptive weight vector adjustment (MOEA/D-AWA). According to the analysis of the geometric relationship between the weight vectors and the optimal solutions under the Chebyshev decomposition scheme, a new weight vector initialization method and an adaptive weight vector adjustment strategy are introduced in MOEA/D-AWA. The weights are adjusted periodically so that the weights of subproblems can be redistributed adaptively to obtain better uniformity of solutions. Meanwhile, computing

efforts devoted to subproblems with duplicate optimal solution can be saved. Moreover, an external elite population is introduced to help adding new subproblems into real sparse regions rather than pseudo sparse regions of the complex PF, that is, discontinuous regions of the PF. MOEA/D-AWA has been compared with four state of the art MOEAs, namely the original MOEA/D, Adaptive-MOEA/D, $pa\lambda$-MOEA/D, and NSGA-II on 10 widely used test problems, two newly constructed complex problems, and two many-objective problems. Experimental results indicate that MOEA/D-AWA outperforms the benchmark algorithms in terms of the IGD metric, particularly when the PF of the MOP is complex.

## 1 Introduction

Many real-world applications have more than one conflicting objective to be optimized simultaneously. These problems are the so-called multi-objective optimization problems (MOPs). A MOP can be defined as follows:

$$
\begin{cases}
\min \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x}))^T \\
\text{subject to: } \mathbf{x} \in \Omega
\end{cases}
\tag{1}
$$

where $\Omega \subset \mathbb{R}^n$ is the decision space and $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \Omega$ is a decision variable which represents a solution to the target MOP. $\mathbf{F}(\mathbf{x})\colon \Omega \to \mathbb{R}^m$ denotes the $m$-dimensional objective vector of the solution $\mathbf{x}$.

Suppose that $\mathbf{x}_A, \mathbf{x}_B \in \Omega$ are two solutions of a MOP, we say that $\mathbf{x}_A$ dominates $\mathbf{x}_B$ (written as $\mathbf{x}_A \prec \mathbf{x}_B$), if and only if $f_i(\mathbf{x}_A) \leq f_i(\mathbf{x}_B), \forall i \in \{1, \ldots, m\}$, and there exists a $j \in \{1, \ldots, m\}$ which makes $f_j(\mathbf{x}_A) < f_j(\mathbf{x}_B)$. A solution $\mathbf{x}^* \in \Omega$ is called Pareto optimal if there is no other solution that dominates $\mathbf{x}^*$. The collection of all Pareto optimal solutions is called the Pareto optimal set (PS), that is $PS = \{\mathbf{x}^* \mid \neg \exists \mathbf{x} \in \Omega, \mathbf{x} \prec \mathbf{x}^*\}$. The Pareto optimal front (PF) is thus defined as the corresponding objective vectors of the solutions in Pareto optimal set, that is, $PF = \{\mathbf{F}(\mathbf{x}) \mid \mathbf{x} \in PS\}$. Solving a MOP is to find its $PS$.

In traditional mathematical programming approaches, the decomposition technique which transfers the MOP into a set of scalar optimization problems has been widely applied (Miettinen, 1999). However, these decomposition-based approaches can only find one Pareto-optimal solution in a single run. By taking advantage of evolutionary algorithms (EAs), Schaffer developed a vector evaluated genetic algorithm (VEGA) by combining the decomposition technique with the genetic algorithms to obtain a set of optimal solutions in a single run (Schaffer, 1985). VEGA is considered to be the first multi-objective evolutionary algorithm (MOEA). After that, the concept of Pareto optimality was introduced into MOEA, where it contributed greatly to this research field. MOEAs with Pareto-ranking-based selection became popular. According to the classification proposed by Coello Coello (2006), the first generation of MOEAs is characterized by the usage of selection mechanisms based on the Pareto ranking and the fitness sharing. Typical representatives include the multi-objective GA (MOGA; Fonseca and Fleming, 1993), the non-dominated sorting GA (NSGA; Srinivas and Deb, 1994) and the niched Pareto GA (NPGA; Horn et al., 1994). The second generation of MOEAs is characterized by the use of elitism strategy. Typical examples include the strength Pareto evolutionary algorithm (SPEA; Zitzler and Thiele, 1999) and its improved version SPEA2 (Zitzler et al., 2001), the Pareto-envelope-based selection algorithm (PESA; Knowles and Corne, 2000) and its enhanced version PESA-II (Corne et al., 2001), the

Pareto archived evolution strategy (PAES; Corne et al., 2000), and the improved version of NSGA (NSGA-II; Deb et al., 2002).

Recently, Zhang and colleagues (Zhang and Li, 2007; Li and Zhang, 2009; Zhang et al., 2009) introduced the decomposition approaches into MOEA and developed the outstanding MOEA/D, which provides an excellent algorithmic framework of evolutionary multi-objective optimization. MOEA/D decomposes the target MOP into a number of scalar optimization problems and then applies the EA to optimize these subproblems simultaneously. In recent years, MOEA/D has attracted increasing research interest and many follow-up studies have been published. Existing research studies may be divided into the following five aspects:

1. Combinations of MOEA/D with other nature inspired meta-heuristics, such as simulated annealing (Li and Landa-Silva, 2011), ant colony optimization (Ke et al., 2010), particle swarm optimization (Moubayed et al., 2010; Martinez and Coello Coello, 2011), and estimation of distribution algorithm (Shim et al., 2012).

2. Changes of the offspring reproducing mechanisms in MOEA/D. Newly developed reproducing operators include the guided mutation operator (Chen et al., 2009), nonlinear crossover and mutation operator (Sindhya et al., 2011), differential evolution schemes (Huang and Li, 2010), and a new mating parent selection mechanism (Lai, 2009).

3. Research on the decomposition approaches. In Zhang, Li, et al. (2010), an NBI-style Chebyshev decomposition approach is proposed to solve MOPs with disparately scaled objectives. In Ishibuchi et al. (2009, 2010), different decomposition approaches are used simultaneously. In Deb and Jain (2012a, 2012b), an enhanced Chebyshev decomposition approach is developed.

4. Refinements of the weight vectors for scalar subproblems. The uniformly distributed weight vectors used in MOEA/D are predetermined. Recent studies have shown that the fixed weight vectors used in MOEA/D might not be able to cover the whole PF very well (Li and Landa-Silva, 2011). Therefore, some studies have been done to refine the weight vectors in MOEA/D. In Gu and Liu (2010), new weight vectors are periodically created according to the distribution of the current weight set. In Li and Landa-Silva (2011), each weight vector is periodically adjusted to make its solution of the subproblem far from the corresponding nearest neighbor. In Jiang et al. (2011), another weight adjustment method is developed by sampling the regression curve of objective vectors of the solutions in an external population.

5. Applications of MOEA/D on benchmark and real-world problems. Most of the applications have been dedicated to multi-objective combinatorial optimization problems, such as the knapsack problem (Li and Landa-Silva, 2011; Ke et al., 2010; Ishibuchi et al., 2010), the traveling salesman problem (Li and Landa-Silva, 2011; Zhang, Li, et al., 2010), the flow-shop scheduling problem (Chang et al., 2008), and the capacitated arc routing problem (Mei et al., 2011). Some practical engineering problems like antenna array synthesis (Pal, Das, et al., 2010; Pal, Qu, et al., 2010), wireless sensor networks (Konstantinidis et al., 2010), robot path planning (Waldocka and Corne, 2011), missile control (Zhang, Tang, et al., 2010), portfolio management (Zhang, Liu, et al., 2010) and rule mining in machine learning (Chan et al., 2010) have also been investigated.

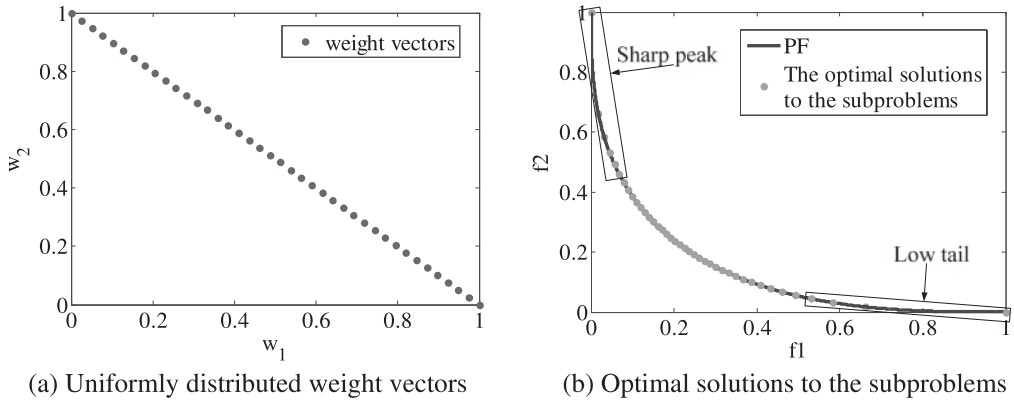(a) Uniformly distributed weight vectors          (b) Optimal solutions to the subproblems

Figure 1: (a) Uniformly distributed weight vectors of the subproblems. (b) The corresponding optimal solutions when the PF has a sharp peak and a low tail.

In this work, we aim at the refinement of weight vectors in MOEA/D. As claimed by Zhang and Li (2007), to make the optimal solutions evenly distributed over the target PF, the weight vectors need to be selected properly under a chosen decomposition scheme. The basic assumption of MOEA/D is that the uniformity of weight vectors will naturally lead to the diversity of the Pareto optimal solutions. In order to ensure the diversity of scalar subproblems, MOEA/D employs a predefined set of uniformly distributed weight vectors. In general, as claimed in Liu et al. (2010), when the PF is close to the hyper-plane $\sum_{i=1}^{m} f_i = 1$ in the objective space, MOEA/D can obtain the uniformly distributed Pareto optimal solutions. However, the basic assumption of MOEA/D might be violated in the cases where the PF is complex, that is, when the PF is discontinuous or has the shape of sharp peak and low tail (see an illustration in Figure 1(b)).

When the target MOP has a discontinuous PF, several subproblems will have the same optimal solution (see the analysis in Section 2). Dealing with these subproblems simultaneously will be a waste of computing effort, as it contributes nothing to the performance of the algorithm. When the target MOP has a PF with a sharp peak or a low tail, the shape of the PF will be far from the hyper-plane $\sum_{i=1}^{m} f_i = 1$ in the objective space. As shown in Figure 1(b), at the sharp peak or low tail areas, there are many non-dominated solutions distributed within a narrow gap in one of the objectives. In this case, MOEA/D cannot obtain a set of uniformly distributed optimal solutions on the PF.

In this paper, we develop an enhanced MOEA/D with adaptive weight vector adjustment (MOEA/D-AWA) to address the MOPs with complex PFs. The major contributions of the paper are the developments of a novel weight vector initialization method and an elite population-based adaptive weight vector adjustment (AWA) strategy. The development of the weight vector initialization method is based on our analysis of the geometric relationship between the weight vectors and their corresponding optimal solutions under the Chebyshev decomposition scheme. The AWA strategy is designed to regulate the distribution of the weight vectors of subproblems periodically according to the current optimal solution set. In the AWA strategy, the elite population is introduced to help in adding new subproblems into the real sparse regions of the complex PF rather than the discontinuous parts which are pseudo-sparse regions.

The rest of this paper is organized as follows. Section 2 analyzes the characteristics of the Chebyshev decomposition approach. Section 3 presents the suggested MOEA/D-AWA algorithm. Section 4 shows the comparison results between the newly developed algorithm and other state of the art algorithms. Some further studies on the effectiveness of MOEA/D-AWA are also made in this part. Section 5 concludes the paper.

## 2    The Analysis

In this section, we briefly introduce the Chebyshev decomposition approach. A geometric analysis on the relationship between the weight vectors and the optimal solutions under this decomposition scheme is carried out.

### 2.1    Analysis on the Chebyshev Decomposition Approach for MOPs

Many approaches have been developed to decompose a MOP into a set of scalar optimization problems (Das and Dennis, 1998; Miettinen, 1999; Messac et al., 2003; Zhang and Li, 2007; Zhang, Li, et al., 2010). Among these decomposition approaches, the Chebyshev approach (Miettinen, 1999) is the most widely used (Zhang and Li, 2007; Zhang, Li, et al., 2010) due to its ability of solving MOPs with non-convex Pareto-optimal fronts. Here we carry out a further analysis on the Chebyshev decomposition approach, which provides a theoretical basis for this work.

Under the Chebyshev scheme, a scalar optimization subproblem can be stated as follows:

$$\min_{\mathbf{x} \in \Omega} g^{tc}(\mathbf{x}|\lambda, z^*) = \min_{\mathbf{x} \in \Omega} \max_{1 \le i \le m} \{\lambda_i |f_i(\mathbf{x}) - z_i^*|\} \tag{2}$$

where $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m)(\sum_{i=1}^m \lambda_i = 1, \lambda_i \ge 0, i = 1, \ldots, m)$ is the weight vector of the scalar optimization subproblem, and $z^* = (z_1^*, z_2^*, \ldots, z_m^*)$ is the reference point (i.e., $z_i^* < \min\{f_i(\mathbf{x})|\mathbf{x} \in \Omega\}, i = 1, \ldots, m$).

It has been proved in Miettinen (1999) that under mild conditions, for each Pareto optimal solution $\mathbf{x}^*$ there exists a weight vector $\lambda$ such that $\mathbf{x}^*$ is the optimal solution of Equation (2). On the other hand, each optimal solution of Equation (2) is a Pareto optimal solution to Equation (1). This property allows obtaining different Pareto optimal solutions by varying the weight vectors.

Due to $z_i^* < \min\{f_i(\mathbf{x})|\mathbf{x} \in \Omega\}, i = 1, \ldots, m$, we have:

$$\min_{\mathbf{x} \in \Omega} g^{tc}(\mathbf{x}|\lambda, z^*) = \min_{\mathbf{x} \in \Omega} \max_{1 \le i \le m} \{\lambda_i \times |f_i(\mathbf{x}) - z_i^*|\} = \min_{\mathbf{x} \in \Omega} \max_{1 \le i \le m} \{\lambda_i \times (f_i(\mathbf{x}) - z_i^*)\}.$$

Based on the above definition, we have the following Theorem 1.

THEOREM 1: *Assume that the target PF of a MOP is piecewise continuous. If the straight line* $\frac{f_1 - z_1^*}{\frac{1}{\lambda_1}} = \frac{f_2 - z_2^*}{\frac{1}{\lambda_2}} = \cdots = \frac{f_m - z_m^*}{\frac{1}{\lambda_m}}(\lambda_i \ne 0, i = 1, 2, \ldots, m)$, *taking* $f_1, f_2, \ldots, f_m$ *as variables, has an intersection with the PF, then the intersection point is the optimal solution to the scalar subproblem with weight vector* $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m)(\sum_{i=1}^m \lambda_i = 1, \lambda_i > 0, i = 1, 2, \ldots, m)$, *where* $z^* = (z_1^*, \ldots, z_m^*)$ *is the reference point.*

The proof can be found in Appendix A.

Moreover, if we let $\lambda_i \ne 0, i = 1, 2, \ldots, m$, then $\frac{f_1 - z_1^*}{\frac{1}{\lambda_1}} = \frac{f_2 - z_2^*}{\frac{1}{\lambda_2}} = \cdots = \frac{f_m - z_m^*}{\frac{1}{\lambda_m}}$ is the straight line that passes through the reference point $z^* = (z_1^*, \ldots, z_m^*)$ with the direction vector $\lambda' = (\frac{\frac{1}{\lambda_1}}{\sum_{i=1}^m \frac{1}{\lambda_i}}, \frac{\frac{1}{\lambda_2}}{\sum_{i=1}^m \frac{1}{\lambda_i}}, \ldots, \frac{\frac{1}{\lambda_m}}{\sum_{i=1}^m \frac{1}{\lambda_i}})$. In the following, we define the direction vector $\lambda'$ as the solution mapping vector of the scalar subproblem with weight vector $\lambda$.
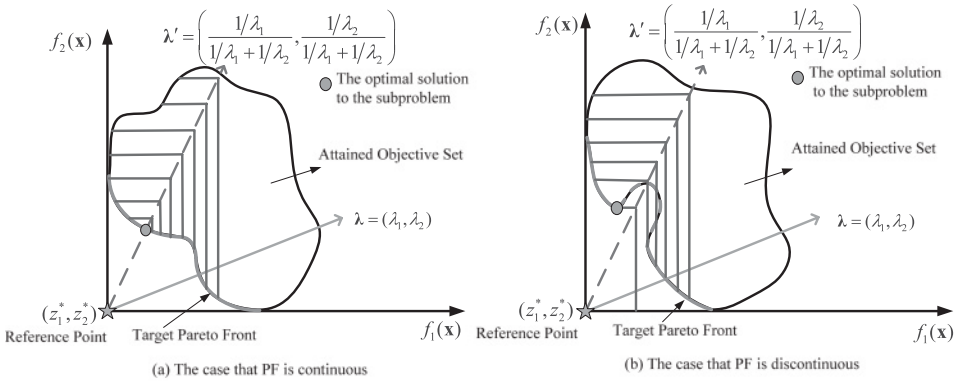
Figure 2: Contour lines of a scalar subproblem and the geometric relationship between its weight vector and optimal solution under the Chebyshev decomposition scheme.
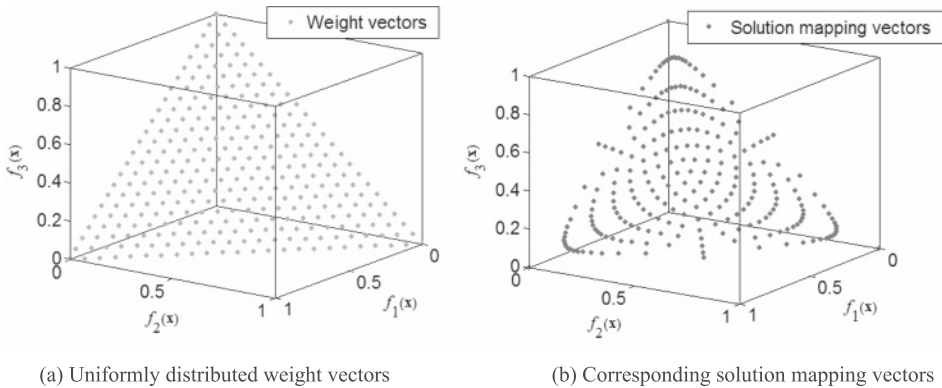


Figure 3: The geometric relationship between the weight vectors and their corresponding solution mapping vectors for tri-objective optimization problems.

For clarity, Figure 2 illustrates the contour lines of a scalar subproblem taking a bi-objective minimization problem as an example. In Figure 2, we combine the objective space and the weight space. The weight vector $\lambda$ is in the weight space and its solution mapping vector $\lambda'$ is in the objective space. It can be seen from Figure 2(a) that the optimal solution of the scalar subproblem with weight vector $\lambda$ is located on the lowest contour line, which is also the intersection point between $\lambda'$ and the target PF. However, when the target PF is discontinuous, as shown in Figure 2(b), there might be no intersection point between $\lambda'$ and the target PF. In this situation, the optimal solution is the point on the target PF that is located on the lowest contour line of the subproblem with weight vector $\lambda$.

For bi-objective optimization problems, the uniformity of the weight vectors and that of the corresponding solution mapping vectors are consistent with each other. However, for tri-objective optimization problems, the situation becomes different. Figure 3 illustrates the geometric relationship between the weight vectors and their corresponding solution mapping vectors in three-dimensional objective space. It can be seen that a
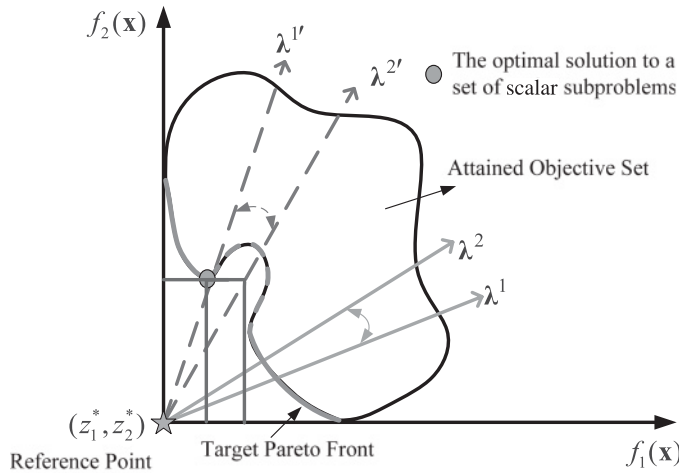
Figure 4: Analysis of the Chebyshev decomposition approach for the MOPs with discontinuous target PF.

set of uniformly distributed weight vectors lead to a set of regularly but nonuniformly distributed solution mapping vectors. The reason is that the relationship between a weight vector and its solution mapping vector is nonlinear.

Theorem 1 indicates that if the solution mapping vector has an intersection point with the target PF, then the intersection point must be the optimal solution of the corresponding scalar subproblem. However, if the PF is discontinuous, the solution mapping vector may have no intersection with the target PF. Figure 4 shows the scalar subproblems whose solution mapping vectors pass through the discontinuous part of the target PF. It can be seen from Figure 3 that at the discontinuous part of the target PF, more than one scalar subproblem has the same optimal solution.

## 2.2 Motivation of the Proposed Work

According to the above analysis, we may draw conclusions as follows. First, uniformly distributed weight vectors cannot guarantee the uniformity of the optimal solutions on the PF. Second, for the MOPs with discontinuous target PFs, some scalar subproblems have the same optimal solution which reduces the diversity of the Pareto optimal solutions. Therefore, in order to maintain a set of uniformly distributed Pareto optimal solutions on the PF, decomposition-based MOEAs should pursue a uniform distribution of the solution mapping vectors. This motivates us to develop a new weight vector initialization method with which a set of uniformly distributed solution mapping vectors are created.

Moreover, we can see that the uniformity of the solution mapping vectors can lead to a set of uniformly distributed Pareto optimal solutions over the PF when the shape of the PF is close to the hyperplane $\sum_{i=1}^{m} f_i = 1$ in the objective space. Unfortunately, the shape of the PF is usually not known in advance. When the target MOP has a complex PF, the new weight vector initialization method cannot solve all the problems. The first challenge is that the PF is discontinuous. In this case, many scalar subproblems obtain the same optimal solution on the breakpoints, which reduces the diversity of subproblems and wastes computing effort. The second challenge is that the PF has a
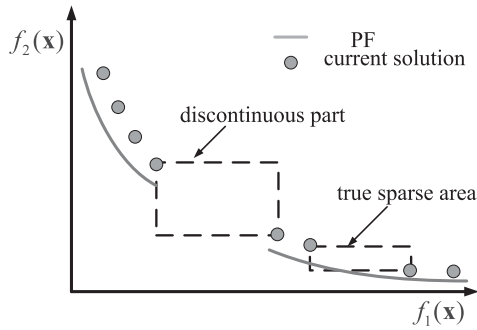
Figure 5: Illustration of the discontinuous part and the real sparse area for the PF which is discontinuous.

complex shape, that is, it has a sharp peak and a low tail, as shown in Figure 1. At the sharp peak or the low tail of the PF, there are many non-dominated solutions distributed within a narrow gap in one of the objectives. In this case, MOEA/D is unable to obtain uniformly distributed optimal solutions on the target PF.

It is natural to pursue uniformly distributed Pareto optimal solutions by removing subproblems from the crowded regions and adding new ones into the sparse regions. However, if the target PF is discontinuous, it is a big challenge to distinguish between the discontinuous regions, which need fewer subproblems to save computing resources, and the sparse regions, which need more subproblems to enhance diversity, as shown in Figure 5. In order to obtain evenly distributed Pareto optimal solutions, more computing resources need to be spent on searching the real sparse regions rather than the pseudo-sparse regions, that is, discontinuous regions. To achieve this goal, we propose to introduce an elite population that is used as a guide for adding and removing subproblems. If an elite individual is located in a sparse region of the evolving population, it will be recalled into the evolving population and a new weight vector will be generated and added to the subproblem set. This strategy helps to introduce new subproblems into the real sparse regions. When the evolving population converges to some extent, individuals in the elite population can be considered as non-dominated solutions. As there is no non-dominated solution that exists at the discontinuous regions of the target PF, the enhanced algorithm will not introduce too many new subproblems into the discontinuous parts.

To handle the MOPs with unknown PF shapes, we propose to deploy an elite population-based adaptive weight vector adjustment strategy to obtain uniformly distributed Pareto optimal solutions. The proposed adjustment strategy can remove redundant scalar subproblems whose solution mapping vectors pass through the discontinuous part of the target PF. This could help to improve the computational efficiency of the algorithm.

For convenience of discussion, we introduce the WS-transformation. It maps the weight vector of a scalar subproblem to its solution mapping vector. If $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m) \in \mathbb{R}^m$, satisfying $\sum_{i=1}^{m} \lambda_i = 1, \lambda_i \geq 0, i = 1, 2, \ldots, m$, is a weight vector, then the WS-transformation, giving rise to $\lambda'$, on $\lambda$ can be defined as:

$$\lambda' = WS(\lambda) = \left( \frac{\frac{1}{\lambda_1}}{\sum_{i=1}^{m} \frac{1}{\lambda_i}}, \frac{\frac{1}{\lambda_2}}{\sum_{i=1}^{m} \frac{1}{\lambda_i}}, \ldots, \frac{\frac{1}{\lambda_m}}{\sum_{i=1}^{m} \frac{1}{\lambda_i}} \right). \tag{3}$$

It can be proved that the WS-transformation is self-inverse. That is, $\lambda = WS(\lambda') = WS(WS(\lambda))$. Thus, if we apply the WS-transformation on the solution mapping vector $\lambda'$, we can obtain the weight vector $\lambda$. For example, if the points in Figure 3(b) are used as the weight vectors, then their solution mapping vectors will be the points in Figure 3(a). In Section 3, this mathematical property will be applied to facilitate the development of our algorithm.

## 3 The Algorithm—MOEA/D-AWA

In this section, the basic idea of the proposed MOEA/D-AWA and its main framework will be described. Then, its computational complexity will be theoretically analyzed.

### 3.1 Basic Idea

Our main goal is to obtain a uniformly distributed optimal solution set on the PF of the target MOPs using MOEA/D by assigning appropriate weight vectors to the scalar subproblems. At first, we develop a novel weight vector initialization method to generate a set of weight vectors by applying the WS-transformation on the original weight set used in MOEA/D. As the WS-transformation is self-inverse, the weight vectors generated by this initialization strategy will lead to a set of solution mapping vectors that are uniformly distributed on the hyperplane $f_1 + f_2 + \cdots + f_m = 1$, as shown in Figure 3.

The new weight vector initialization method can significantly improve the performance of MOEA/D in general. However, it cannot solve all the problems. For the MOPs with complex PF, the uniformity of solution mapping vectors of the subproblems still cannot guarantee the uniformity of the optimal solutions on complex PFs. In this case, the suggested algorithm is expected to obtain a set of uniformly distributed optimal solutions by using the adaptive weight vector adjustment strategy.

In summary, our basic idea is to apply a two-stage strategy to deal with the generation of the weight vectors. In the first stage, a set of predetermined weight vectors are used until the population is considered converged to some extent. Then, a portion of the weight vectors are adjusted according to the current Pareto optimal solutions based on our geometric analysis. Specifically, some subproblems will be removed from the crowded parts of the PF, and some new subproblems will be created into other parts of the PF.

### 3.2 Novel Weight Vector Initialization Method

In this section we present the new weight vector initialization method. The method is summarized in Algorithm 1.

Taking a tri-objective optimization problem as an example, we can see from Figure 3 the difference between the weight vectors generated by MOEA/D (Figure 3(a)) and the weight vectors constructed by our method (Figure 3(b)), here the subproblem number is set as $N = 300$.

### 3.3 Recreating Subproblems

Recently, Kukkonen and Deb (2006) proposed a fast and effective method for pruning of non-dominated solutions. A crowding estimation approach using $k$ nearest neighbors of each solution, which is termed as vicinity distance, was adopted in their work. The vicinity distance of the $j$th solution among a population of solutions can be defined as

---

**Algorithm 1** New weight vector initialization

---

**Require:** $N$ : the number of the subproblems used in MOEA/D-AWA; $m$ : the number of the objective functions in a MOP (1);

**Ensure:** a set of initial weight vectors $\lambda^1, \lambda^2, \cdots, \lambda^N$;

   **Step 1**: Generate a set of evenly scattered $N$ weight vectors $\{\lambda^{1'}, \lambda^{2'}, \cdots, \lambda^{N'}\}$ using the method from Zhang and Li (2007) in its feasible space $\{(w_1, w_2, \cdots, w_m) | \sum_{i=1}^{m} w_i = 1, w_i \geq 0, i = 1, 2, \cdots, m\}$.

   **Step 2**: Apply the WS-transformation on the generated weight vectors:

   **for** $j = 1, \cdots, N$, **do**

$$\lambda^j = \begin{cases} WS(\lambda^{j'}), & \text{if } \prod_{i=1}^{m} \lambda_i^{j'} \neq 0 \\ WS(\lambda^{j'} + \varepsilon * [1, 1, \cdots, 1]), & otherwise \end{cases}$$

   where $\varepsilon > 0$ and $\varepsilon \to 0$.

---

**Algorithm 2** Deleting overcrowded subproblems

---

**Require:** $evol\_pop$ : The current population;

   $nus$ : the maximal number of subproblems adjusted in AWA;

**Ensure:** The adjusted population $evol\_pop'$;

   **Step 1**: Update the current evolutional population $EP$:

   **if** $g^{tc}(\mathbf{x}^i | \lambda^j, z) < g^{tc}(\mathbf{x}^j | \lambda^j, z)$, $\mathbf{x}^i, \mathbf{x}^j \in evol\_pop$, $i, j = 1, \cdots, |evol\_pop|$,

   **then** set $\mathbf{x}^j = \mathbf{x}^i$, $FV^j = FV^i$, where $FV^i$ and $FV^j$ are the objective function values of $\mathbf{x}^i$ and $\mathbf{x}^j$;

   **Step 2**: Calculate the sparsity level for each individual $ind^i (i = 1, 2, \ldots, |evol\_pop|)$ in the population $evol\_pop$ among $evol\_pop$ by Equation (4);

   **Step 3**: Delete overcrowded subproblem:

   **if** the number of deleted subproblems does not reach the required number $nus$,

   **then** remove the individual with the minimum sparsity level, goto Step 2;

   **else** output the remaining individual as the resulting $evol\_pop'$

---

$V^j = \prod_{i=1}^{k} L_2^{NN_i^j}$, where $L_2^{NN_i^j}$ is the Euclidean distance from the $j$th solution to its $i$th nearest neighbor.

   In MOEA/D-AWA, the vicinity distance is adopted to evaluate the sparsity level of a solution among current non-dominated set. The sparsity level of the $j$th individual $ind^j$ among the population $pop$ can be defined as:

$$SL(ind^j, pop) = \prod_{i=1}^{m} L_2^{NN_i^j}. \tag{4}$$

   Algorithms 2 and 3 summarize the procedures for deleting overcrowded subproblems and creating new subproblems, respectively.

   In Step 3.1 of Algorithm 3, Equation (5) can result in a good weight vector according to the individual with largest sparsity level. To justify, we introduce the definition of the optimal weight vector of a MOP solution.

---

**Algorithm 3** Adding new subproblems into the sparse regions

---

**Require:** $evol\_pop'$ : the resultant population after subproblems deletion;

$z^*$ : the reference point; $EP$ : an external archive population;

$nus$ : the maximal number of subproblems adjusted in AWA;

**Ensure:** The adjusted population $evol\_pop''$.

**Step 1**: Remove the individuals in $EP$ which are dominated by the individuals in $evol\_pop'$;

**Step 2**: Calculate sparsity levels of the individuals in $EP$ among the population $evol\_pop'$ using Equation (4);

**Step 3**: Add a new subproblem to the sparse region:

3.1 Generate a new subproblem using the individual $ind^{sp} = (x^{sp}, FV^{sp})$ which has the largest sparsity level. The weight vector $\lambda^{sp}$ of the new constructed subproblem can be calculated as follows, in which $FV^{sp} = (f_1^{sp}, \cdots, f_m^{sp})$,

$$\lambda^{sp} = \left( \frac{\frac{1}{f_1^{sp}-z_1^*}}{\sum_{k=1}^m \frac{1}{f_1^{sp}-z_k^*}}, \cdots, \frac{\frac{1}{f_m^{sp}-z_m^*}}{\sum_{k=1}^m \frac{1}{f_1^{sp}-z_k^*}} \right), \quad \prod_{j=1}^m (f_j^{sp} - z_j^*) \neq 0; \quad (5)$$

3.2 Set the solution of the new constructed subproblem as $ind^{sp}$ and add it to the current population $evol\_pop'$;

**Step 4**: Stopping criteria: If the number of added subproblems reach the required number $nus$, then output the current individual set as the resulting $evol\_pop''$. Otherwise goto Step 2.

---

DEFINITION: *Given the objective values $F = (f_1, \ldots, f_m)$ of a solution and the reference point $z^* = (z_1^*, \ldots, z_m^*)$, denote $h(\lambda|F, z^*) = max_{1 \leq i \leq m}\{\lambda_i|f_i - z_i^*|\}$ and $W_m = \{(\lambda_1, \lambda_2, \ldots, \lambda_m)| \sum_{i=1}^m \lambda_i = 1, \lambda_i \geq 0, i = 1, 2, \ldots, m\}$, we say that $\lambda^{opt}$ is the optimal weight vector to the solution F, if*

$$h(\lambda^{opt}|F, z^*) = \min_{\lambda \in W_m} h(\lambda|F, z^*) = \min_{\lambda \in W_m} \max_{1 \leq i \leq m}\{\lambda_i|f_i - z_i^*|\} = \min_{\lambda \in W_m} \max_{1 \leq i \leq m}\{\lambda_i * (f_i - z_i^*)\}.$$

From the definition, we can see that for given $F$ and $z^*$, the optimal weight vector is the one that makes $h(\lambda|F, z^*)$ reach the minimum value. Thus, it seems sensible to allocate the optimal weight vector to the individual with the largest sparsity level. Theorem 2 provides a theoretical basis for constructing the optimal weight vector of a given solution.

THEOREM 2: *Given $F = (f_1, \ldots, f_m)$ and $z^* = (z_1^*, \ldots, z_m^*)$, if $\prod_{j=1}^m (f_j - z_j^*) \neq 0$, then $\lambda^{opt} = WS(F - z^*) = (\frac{\frac{1}{f_1-z_1^*}}{\sum_{k=1}^m \frac{1}{f_1-z_k^*}}, \ldots, \frac{\frac{1}{f_m-z_m^*}}{\sum_{k=1}^m \frac{1}{f_1-z_k^*}})$ is the optimal weight vector to F based on $z^*$.*

The proof can be found in Appendix B.

Figure 6 takes a bi-objective problem as an example to illustrate how to construct the optimal weight vector for a given solution $F$ and a reference point $z^*$. In case $\prod_{j=1}^m (f_j^{sp} - z_j^*) = 0$, we replace Equation (5) by Equation (6).

$$\lambda^{sp} = \left( \frac{\frac{1}{f_1^{sp}-z_1^*+\varepsilon}}{\sum_{k=1}^m \frac{1}{f_1^{sp}-z_k^*+\varepsilon}}, \ldots, \frac{\frac{1}{f_m^{sp}-z_m^*+\varepsilon}}{\sum_{k=1}^m \frac{1}{f_1^{sp}-z_k^*+\varepsilon}} \right), \quad \varepsilon > 0, \varepsilon \to 0. \quad (6)$$
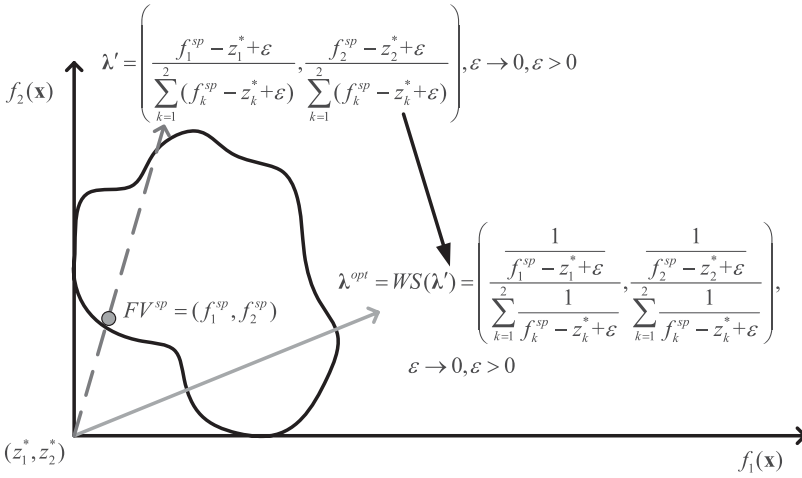
$$\lambda' = \left( \frac{f_1^{sp} - z_1^* + \varepsilon}{\sum\limits_{k=1}^{2}(f_k^{sp} - z_k^* + \varepsilon)}, \frac{f_2^{sp} - z_2^* + \varepsilon}{\sum\limits_{k=1}^{2}(f_k^{sp} - z_k^* + \varepsilon)} \right), \varepsilon \to 0, \varepsilon > 0$$

$$\lambda^{opt} = WS(\lambda') = \left( \frac{\frac{1}{f_1^{sp} - z_1^* + \varepsilon}}{\sum\limits_{k=1}^{2} \frac{1}{f_k^{sp} - z_k^* + \varepsilon}}, \frac{\frac{1}{f_2^{sp} - z_2^* + \varepsilon}}{\sum\limits_{k=1}^{2} \frac{1}{f_k^{sp} - z_k^* + \varepsilon}} \right),$$

$$\varepsilon \to 0, \varepsilon > 0$$

$FV^{sp} = (f_1^{sp}, f_2^{sp})$

$f_2(\mathbf{x})$

$(z_1^*, z_2^*)$

$f_1(\mathbf{x})$

Figure 6: Plot of constructing the optimal weight vector for a given solution and the reference point in bi-objective problems.

## 3.4 MOEA/D-AWA

MOEA/D-AWA has the same framework as MOEA/D. Moreover, it employs the strategy of allocating different computational resources to different subproblems as proposed in Zhang et al. (2009). In this strategy, a utility function is defined and computed for each subproblem. Computational efforts are distributed to each of the subproblems based on their utility function values. The major differences between MOEA/D and the suggested MOEA/D-AWA focus on two aspects. One is the weight vector itialization method, the other is the update of the weight vectors during the search procedure.

During the search procedure, MOEA/D-AWA maintains the following items:

- An evolving population $evol\_pop = \{ind^1, \cdots, ind^N\}$, and $ind^i = (\mathbf{x}^i, FV^i)$, $i = 1, 2, \ldots, N$ where $\mathbf{x}^i$ is the current solution to the $i$th subproblem and $FV^i = F(\mathbf{x}^i)$;

- A set of weight vectors $\lambda^1, \lambda^2, \ldots, \lambda^N$;

- A reference point $z^* = (z_1^*, \ldots, z_m^*)^T$, where $z_i^*$ is less than the best value obtained so far for the $i$th objective;

- The utility estimations of the subproblems $\pi^1, \pi^2, \ldots, \pi^N$;

- An external population $EP$ for the storage of non-dominated solutions during the search.

MOEA/D-AWA requires a set of parameters as input, including the neighborhood size $T$, the probability of selecting mate subproblem from its neighborhood $\delta$, the iteration interval of utilizing the adaptive weight vector adjustment strategy $wag$, the maximal number of subproblems needed to be adjusted $nus$, the maximum iteration times $G_{\max}$, the ratio of iteration times to evolve with only MOEA/D, $rate\_evol$ and

the max size of $EP$. Given these items and the parameters, MOEA/D-AWA can be summarized in Algorithm 4.

In MOEA/D-AWA, the adaptive weight adjustment strategy in Step 4 is applied periodically. It comes into play only after the population has converged to some extent. The purpose of introducing the external population is to store the visited non-dominated solutions and provide a guidance of adding and removing subproblems in the current evolving population to obtain a better diversity.

For a MOP whose PF is close to the hyperplane $\sum_{i=1}^{m} f_i = 1$, the enhanced MOEA/D with the new weight vector initialization method can perform well. However, when the assumption is violated, the adaptive weight adjustment strategy based on the elite population could be of great help. It introduces new subproblems into sparse regions of the evolving population without adding too many subproblems into the discontinuous parts (cf. Section 2.2).

### 3.5 Computational Complexity of MOEA/D-AWA

The time complexity for allocating the computing resources in Step 2 is $O(N)$, while for the operation of updating the solutions of evolution population in Step 3 is $O(m \times N^2)$. The time complexity for removing the overcrowded subproblems in Step 4.1 is $O(nus \times m \times N^2)$, while adding a new subproblem in Step 4.2 is $O(nus \times m \times N^2)$. The time complexity for updating the neighborhood in Step 4.3 is $O(m \times T \times N^2)$ and the updating of the external elite population is $O(m \times N^2)$. In summary, the time complexity of the adaptive strategy in Step 4 at each iteration is $O(nus \times m \times N^2 + T \times N^2 + 2m \times T \times N^2)$.

Taking the time complexity of initialization, which is $\mathcal{O}(m \times T \times N)$, into account, the total time complexity of MOEA/D-AWA is $O(m \times N^2 \times (T + nus) \times G_{\max})$. Comparing with the computational complexity of NSGA-II (Deb et al., 2002) and MOEA/D (Zhang et al., 2009), MOEA/D-AWA allocates additional computational resources for the strategy of adaptive weight vector adjustment.

## 4 Experimental Study

In this section, we first compare MOEA/D-AWA with four other algorithms: the original MOEA/D (Zhang et al., 2009), NSGA-II (Deb et al., 2002), Adaptive-MOEA/D (Li and Landa-Silva, 2011), and $pa\lambda$-MOEA/D (Jiang et al., 2011). Adaptive-MOEA/D and $pa\lambda$-MOEA/D are newly developed MOEA/D with adaptive weight vector design. Adaptive-MOEA/D employs the weight vectors adjustment approach that was designed in EMOSA (Li and Landa-Silva, 2011). As EMOSA is proposed for handling combinatorial problems, we replace the proposed weight vectors adjustment strategy by the one in EMOSA to form the Adaptive-MOEA/D. $pa\lambda$-MOEA/D (Jiang et al., 2011) is another improved MOEA/D with weight vector adjustment. Secondly, we study the effectiveness of the developed weight vector initialization method and the elite population-based AWA strategy. Thirdly, the effectiveness of the proposed method on many-objective test problems is studied.

### 4.1 Test Instances

In the experimental study, we select five widely used bi-objective ZDT test instances and five tri-objective DTLZ problems to compare the proposed MOEA/D-AWA with MOEA/D, Adaptive-MOEA/D, $pa\lambda$-MOEA/D, and NSGA-II. In order to investigate the capability of MOEA/D-AWA to solve MOPs whose PF has a sharp peak and low tail, we construct two new test instances F1 and F2 whose ideal PFs are of complex shapes.

---

**Algorithm 4** MOEA/D-AWA

---

**Require:** A stopping criterion, the parameter set
**Ensure:** $\mathbf{x}^1, \cdots, \mathbf{x}^N$ and $FV^1, \cdots, FV^N$

**Step 1: Initialization**

1.1 Initialize the weight vectors $\lambda^1, \lambda^2, \cdots, \lambda^N$ by applying the WS-transformation Equation (3) on the original evenly scattered weight vectors in MOEA/D (Zhang and Li, 2007; Li and Zhang, 2009). Compute the Euclidean distances between any two weight vectors and find the closest weight vectors to each weight vector. For each $i = 1, \cdots, N$, set the neighborhood list of the $i$th subproblem as $B(i) = \{i_1, \cdots, i_T\}$ where $\lambda_1^i, \cdots, \lambda_T^i$ are the $T$ closest weight vectors to $\lambda_i$;

1.2 Initialize $evol\_pop$ by generating $\mathbf{x}^1, \cdots, \mathbf{x}^N$ randomly, set $FV^i = F(\mathbf{x}^i)$; initialize $z^* = (z_1^*, \cdots, z_m^*)^T$ by setting $z_i^* = \min\{f_i(\mathbf{x}^1), \cdots, f_i(\mathbf{x}^N)\} - 10^{-7}$;

1.3 Set $\pi^i = 1$ for all $i = 1, \cdots, N$, and $gen = 0, EP = \emptyset$.

**Step 2: Allocation of Computing Resources**

2.1 Update the utility function $\pi^1, \pi^2, \cdots, \pi^N$. If $gen$ is a multiple of 50, calculate:
$\Delta^i = \dfrac{\text{old function value} - \text{new function value}}{\text{old function value}}$.

2.2 Update the utility function

$$\pi^i = \begin{cases} 1 & : \quad \Delta^i > 0.001 \\ (0.95 + 0.05 * \frac{\Delta^i}{0.001})\pi^i & : \quad \text{otherwise} \end{cases}.$$

2.3 Select the subproblems according to the utility function. Set $I = \emptyset$ and select the indices of the $m$ subproblems whose weight vectors are permutations of $(1, 0, \ldots, 0)$. Choose other $\lfloor \frac{N}{5} \rfloor - m$ indices using 10-tournament selection (Miller and Goldberg, 1995) according to $\pi^i$, and add them to $I$.

**Step 3: Evolution**

For each $i \in I$, do

3.1 Select mating scope: Uniformly create a random number $rand$ from $(0, 1)$, set

$$P = \begin{cases} B(i) & : \quad rand < \delta \\ \{1, 2, \cdots, N\} & : \quad otherwise \end{cases}.$$

3.2 Reproduce: Set $r_1 = i$ and randomly choose two indices $r_2$ and $r_3$ from $P$, construct a solution $\bar{y}$ from $\mathbf{x}^{r_1}$, $\mathbf{x}^{r_2}$ and $\mathbf{x}^{r_3}$ by the SBX operator and apply the polynomial mutation operator (Deb and Beyer, 2001) on $\bar{y}$ with probability $p_m$ to generate a new solution $\mathbf{y}$.

3.3 Repair: If an element of $\mathbf{y}$ is out of $\Omega$, its value will be reset as a randomly selected value within $\Omega$.

3.4 Update the reference $z$: for all $j \in \{1, \cdots, m\}$, if $f_j(y) < z_j^*$, set $z_j^* = f_j(y) - 10^{-7}$.

3.5 Update the solutions: Set $c = 0$ and do the following:

(1) **if** $c = n_r$ or $P = \emptyset$, then jump out of Step 3.5; **else** randomly select an index $j$ from $P$.

(2) **if** $g(\mathbf{y}|\lambda^j, z^*) \leq g(\mathbf{x}^j|\lambda^j, z^*)$, set $\mathbf{x}^j = \mathbf{y}$, $FV^j = F(\mathbf{y})$ and $c := c + 1$.

(3) Delete index $j$ from $P$ and go to (1).

---

---

**Step 4: Adaptive Weight Adjustment**

    **if** $gen \geq raat\_evol * G_{\max}$ and $gen \bmod wag = 0$, adaptively adjust the weight vectors as follows:

        4.1 Update the external population $EP$ by the new generation of offspring according to the vicinity distance based non-dominated sorting (Yang et al., 2010);

        4.2 Delete the overcrowded subproblems by Algorithm 2;

        4.3 Add new subproblems into the sparse regions by Algorithm 3;

        4.4 for $i = 1, \cdots, N$, find the $T$ closest weights to $\lambda^i$ and build the new $B(i)$.

    **else** go to Step 5.

**Step 5: Stopping Criteria**

    if the stopping criterion is met, stop; else set $gen := gen + 1$, go to Step 2.

---

To investigate the ability of MOEA/D-AWA for many-objective problems, DTLZ5(I,m) (Deb and Saxena, 2005; Saxena and Deb, 2007; Singh et al., 2011) and its variation DTLZ4(I,m) are selected as test problems. Some detailed descriptions of these problems can be found in Table 6 in Appendix C.

    The simulation codes of the compared approaches are developed by Visual Studio 2005 and run on a workstation with Inter Core6 2.8 GHz CPU and 32GB RAM.

## 4.2    Performance Metric

In the experimental study, we use the inverted generational distance (IGD) metric which is a comprehensive index of convergence and diversity (Zitzler et al., 2003) to evaluate the performance of all compared algorithms. Let $P^*$ be a set of evenly distributed points over the PF (in the objective space). Suppose that $P$ is an approximate set of the PF, the average distance from $P^*$ to $P$ is defined as:

$$IGD(P^*, P) = \frac{\sum_{v \in P^*} d(v, P)}{|P^*|} \tag{7}$$

where $d(v, P)$ is the minimum Euclidean distance between $v$ and the solutions in $P$. When $|P^*|$ is large enough, $IGD(P^*, P)$ can measure both the uniformity and the convergence of $P$. A low value of $IGD(P^*, P)$ indicates that $P$ is close to the PF and covers most of the whole PF.

    For all the benchmark algorithms, we use the evolutionary population as the population $P$ to calculate the IGD metric.

## 4.3    Parameter Settings

In our experimental study, NSGA-II follows the implementation of Deb et al. (2002). The parameters of all compared algorithms are set as follows.

    The simulated binary crossover (SBX) operator and polynomial mutation (Deb and Beyer, 2001) are employed in MOEA/D-AWA, MOEA/D, Adaptive-MOEA/D, $pa\lambda$-MOEA/D, and NSGA-II for the 14 test problems. The parameter settings are listed in Table 1, where $n$ is the number of variables and *rand* is a uniform random number in [0,1].

    The population size is set to $N = 100$ for the five bi-objective ZDT problems, $N = 300$ for the given tri-objective DTLZ problems, $N = 100$ for the bi-objective $F_1$ problem,

Table 1: Parameter settings for SBX and PM.

| Parameter | MOEA/D | MOEA/D-AWA | NSGA-II |
|---|---|---|---|
| Crossover probability $p_c$ | 1 | 1 | 1 |
| Distribution index for crossover | 20 | 20 | 20 |
| Mutation probability $p_m$ | $1/n$ | $1/n$ | $1/n$ |
| Distribution index for mutation | 20 | 20 | 20 |

$N = 300$ for the tri-objective $F_2$ problem, and $N = 252$ for the 6-objective DTLZ4(3,6) and DTLZ5(3,6). The size of external elite is set as $1.5N$.

All the compared algorithms stop when the number of function evaluation reaches the maximum number. The maximum number is set to 50,000 for the five bi-objective ZDT problems, 75,000 for the five tri-objective DTLZ problems, 50,000 for the bi-objective $F_1$ problem, 75,000 for the tri-objective $F_2$ problem, and 200,000 for the two 6-objective DTLZ4(3,6) and DTLZ5(3,6) problems.

For MOEA/D-AWA, the maximal number of subproblem adjusted *nus* is set to rate_update_weight $\times N$ and the parameter of rate_update_weight is set to 0.05. The parameter rate_evol is set to 0.8, that is, 80% of computing resources are devoted to MOEA/D while the remaining 20% are assigned to the adaptive weight vector adjustment.

MOEA/D uses weight vectors as shown in the literature (Zhang and Li, 2007; Li and Zhang, 2009; Zhang et al., 2009). MOEA/D-AWA and Adaptive-MOEA/D apply the new constructed initial weight vectors which have been defined in Step 1.1 of Algorithm 2, $pa\lambda$-MOEA/D uses the weight vectors from Jiang et al. (2011). For MOEA/D-AWA, MOEA/D, Adaptive-MOEA/D, and $pa\lambda$-MOEA/D, the size of neighborhood list $T$ is set to $0.1N$, and the probability of choosing the mate subproblem from its neighborhood $\delta$ is set as 0.9.

In Adaptive-MOEA/D, the iteration intervals of utilizing the adaptive weight vector adjustment strategy (the parameter of $wag$) is set to different values for different problems. For ZDT problems, DTLZ2, DTLZ4, and DTLZ6, $wag$ is set to 100. For P_F2, P_F3, and P_F4, $wag$ is set to 120. For DTLZ1, DTLZ3, and P_F1, $wag$ is set to 125. For F2, $wag$ is set to 160. For F1, $wag$ is set to 200. For DTLZ4(3,6) and DTLZ5(3,6), $wag$ is set to 250.

## 4.4    Experimental Studies on MOEA/D-AWA and Comparisons

This part of the experiments is designed to study the effectiveness of MOEA/D-AWA on different types of MOPs. Firstly, the classical ZDT and DTLZ problems are investigated. Then the performance of MOEA/D-AWA on MOPs with complex PFs is studied.

### 4.4.1    Experimental Results for ZDT and DTLZ Problems

In this part of the experiments, the ZDT and the DTLZ problems are bi-objective and tri-objective problems, respectively. Their mathematical descriptions and the ideal PFs of the ZDT and DTLZ problems can be found in Deb et al. (2002).

Table 2 presents the mean and standard deviation of the IGD metric values of the final solutions obtained by each algorithm for five 30-dimensional ZDT problems and five 10-dimensional DTLZ problems over 30 independent runs. This table reveals that in terms of the IGD metric, the final solutions obtained by MOEA/D-AWA are better than

Table 2: Statistic IGD metric values of the Pareto-optimal solutions founded by the five compared algorithms on the ZDT and DTLZ problems. The numbers in parentheses are the standard deviations.*

| Instance | MOEA/D-AWA | MOEA/D | Adaptive-MOEA/D | $pa\lambda$-MOEA/D | NSGA-II |
|---|---|---|---|---|---|
| ZDT1 | 4.470e–3 | 4.739e–3 | 4.750e–3 | **3.674e–3** | 4.696e–3 |
| | (2.239e–4) | (3.973e–5) | (1.801e–4) | **(5.923e–5)** | (1.435e–4) |
| ZDT2 | 4.482e–3 | 4.461e–3 | 4.727e–3 | **3.900e–3** | 4.724e–3 |
| | (1.837e–4) | (9.849e–5) | (1.277e–4) | **(2.735e–4)** | (1.390e–4) |
| ZDT3 | 6.703e–3 | 1.362e–2 | 1.248e–2 | 9.737e–3 | **5.281e–3** |
| | (4.538e–4) | (1.574e–4) | (1.876e–4) | (6.636e–4) | **(1.788e–4)** |
| ZDT4 | **4.238e–3** | 4.692e–3 | 4.835e–3 | 5.174e–3 | 4.880e–3 |
| | **(3.102e–4)** | (1.339e–4) | (3.917e–4) | (7.339e–4) | (3.713e–4) |
| ZDT6 | 4.323e–3 | 4.474e–3 | 5.331e–3 | **3.601e–3** | 4.261e–3 |
| | (2.819e–4) | (3.666e–4) | (5.899e–4) | **(4.250e–4)** | (2.255e–4) |
| DTLZ1 | **1.237e–2** | 1.607e–2 | 1.605e–2 | 1.632e–2 | 3.982e–2 |
| | **(1.617e–3)** | (9.458e–4) | (3.266e–3) | (2.106e–3) | (1.121e–3) |
| DTLZ2 | **3.065e–2** | 3.878e–2 | 3.189e–2 | 3.232e–2 | 4.696e–2 |
| | **(1.183e–4)** | (2.974e–4) | (1.045e–3) | (9.275e–4) | (1.435e–3) |
| DTLZ3 | **3.196e–2** | 3.921e–2 | 4.487e–2 | 5.723e–2 | 8.741e–2 |
| | **(8.036e–4)** | (5.883e–4) | (1.080e–2) | (9.761e–2) | (5.430e–2) |
| DTLZ4 | **3.068e–2** | 3.889e–2 | 3.277e–2 | 3.443e–2 | 3.951e–2 |
| | **(1.351e–4)** | (3.202e–4) | (1.135e–3) | (9.476e–3) | (1.065e–3) |
| DTLZ6 | **3.610e–2** | 8.778e–2 | 7.787e–2 | 6.980e–2 | 4.156e–2 |
| | **(5.054e–3)** | (2.603e–3) | (2.830e–3) | (2.539e–3) | (1.483e–3) |

*Note: The values in bold indicate the best performing algorithm for the particular instance of a test problem.

MOEA/D for nine out of 10 problems, except for the simple problem ZDT2. Especially for the tri-objective DTLZ problems, MOEA/D-AWA performs much better than all of the compared algorithms.

It can be seen from Table 2 that the proposed MOEA/D-AWA performs best on tri-objective DTLZ problems and much better than MOEA/D and two other MOEA/D based algorithms on ZDT3. The five DTLZ problems investigated here are tri-objective MOPs with simple PFs. For these problems, the developed weight vector initialization method in MOEA/D-AWA plays an important role in maintaining good uniformity. As has been analyzed in Section 2, the initial weight vector set in MOEA/D-AWA corresponds to a set of uniformly distributed solution mapping vectors which can lead to a number of uniformly distributed Pareto optimal solutions over the PF when the curve shape of the PF is close to the hyper-plane $\sum_{i=1}^{m} f_i = 1$.

Figure 7 and Figure 8 show in the objective space, the distribution of the final solutions with the lowest IGD value found by each algorithm for tri-objective DTLZ problems. The ideal PF of DTLZ1 is a hyperplane in the first quadrant. The ideal PFs of DTLZ2, DTLZ3, and DTLZ4 are unit spheres in the first quadrant. The ideal PF of DTLZ6 is composed of four curve surfaces. It is visually evident that MOEA/D-AWA is significantly better than the original MOEA/D in terms of the uniformity of final solutions. Among the benchmark algorithms, the final non-dominated solutions found by MOEA/D distribute with some regularity and concentrate at the center of the target PF. The points obtained by Adaptive-MOEA/D distribute with no regularity, but their uniformity is not as good as those obtained by MOEA/D-AWA. The points obtained by
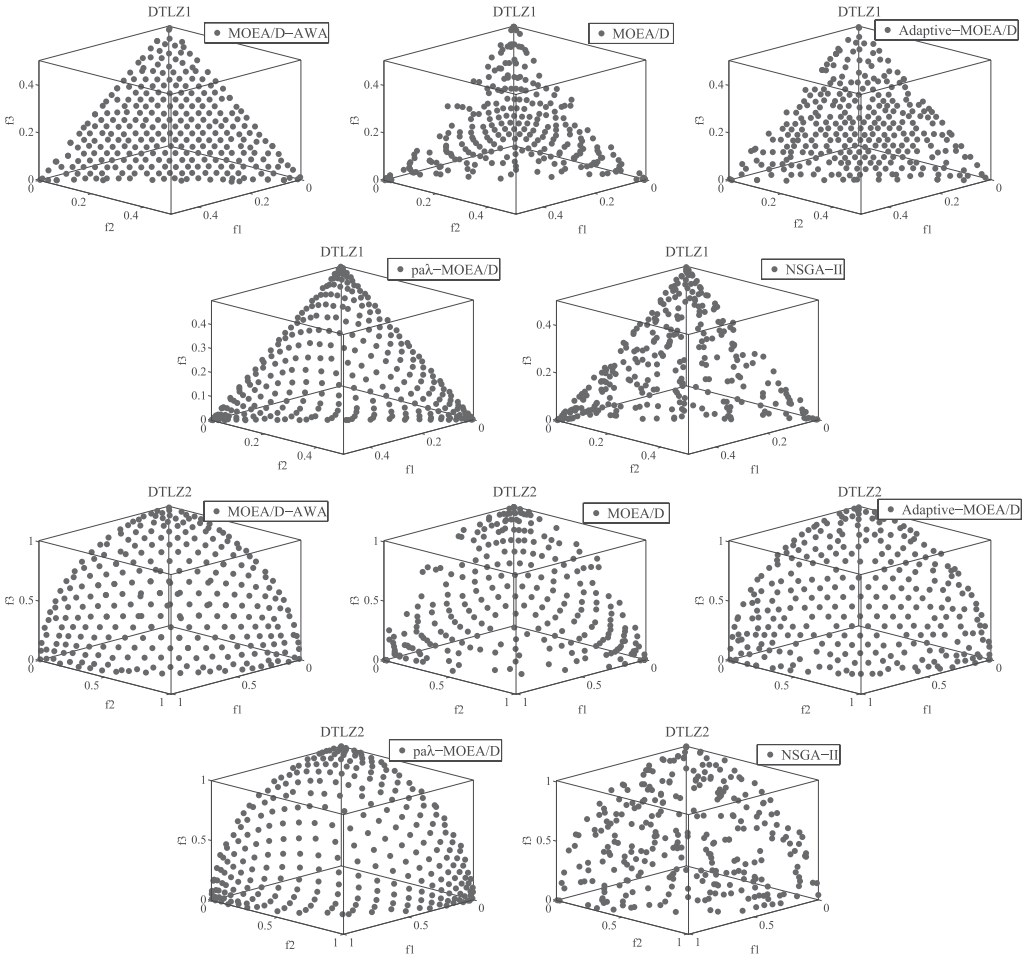
Figure 7: The distribution of the final non-dominated solutions with the lowest IGD values found by MOEA/D-AWA, MOEA/D, Adaptive-MOEA/D, $pa\lambda$-MOEAD, and NSGA-II on the 10-dimensional DTLZ problems (part I).

$pa\lambda$-MOEA/D are uniformly distributed but with some regularity. NSGA-II obtains a good diversity, but its uniformity is not very good.

Moreover, it should be pointed out that MOEA/D-AWA obtains good performance on DTLZ6, whose PF is discontinuous and is not close to the hyperplane $\sum_{i=1}^{m} f_i = 1$. When dealing with this problem, the AWA strategy helps to maintain better diversity and saves computing effort that will then be devoted to the discontinuous parts of the PF in the original MOEA/D. In Section 4.5.1, further discussions will be made on the computing effort assignment among the discontinuous parts of the PF.

As for the bi-objective ZDT problems, the weight vector initialization method in MOEA/D-AWA does not work well, because the new weight vector set is exactly the same as the original one. According to the definition of the WS-transformation in Section 2, in the two-dimensional objective space, the weight vector set is basically the same after the WS-transformation except that the order is reversed. Therefore, when
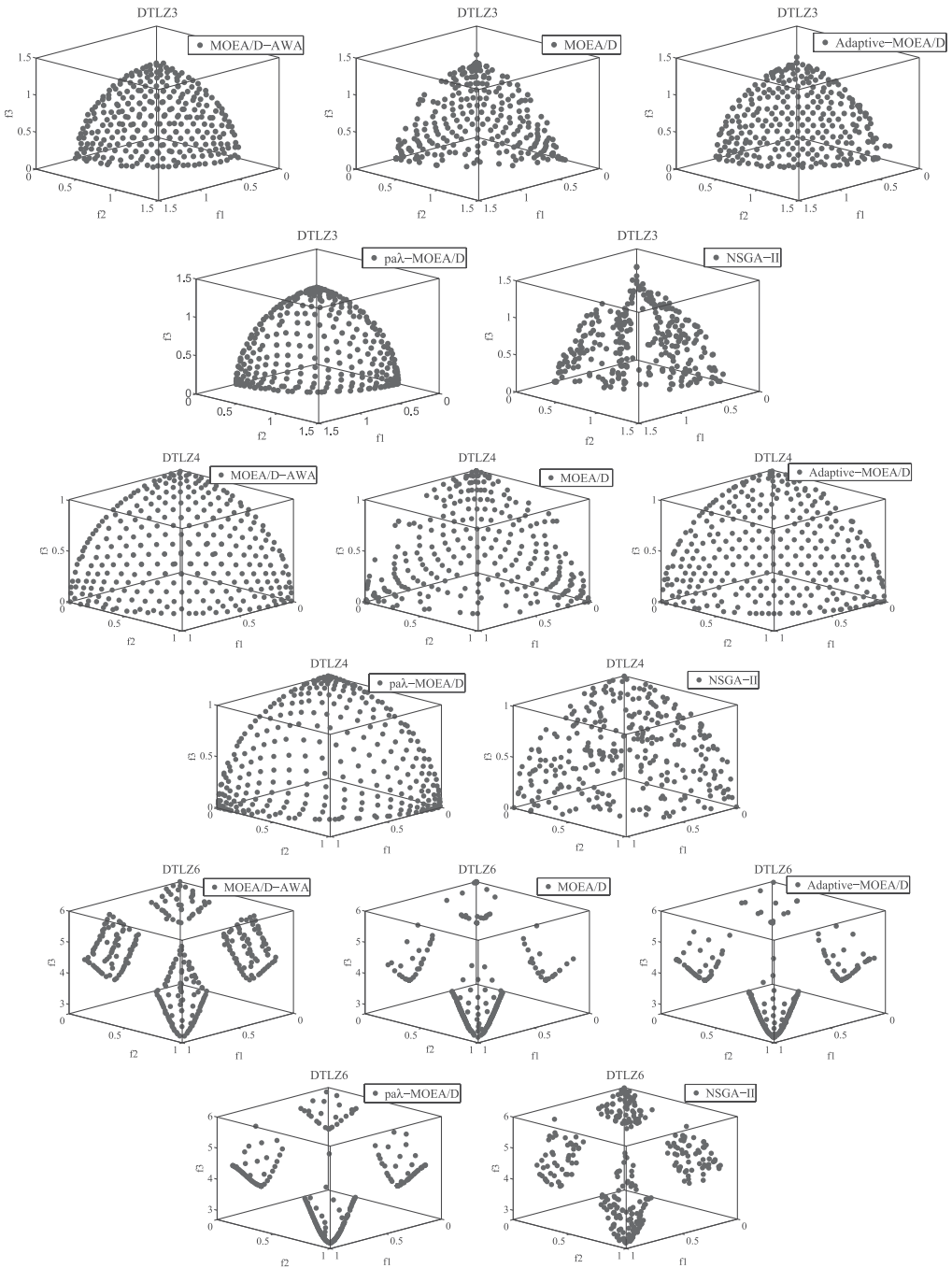
Figure 8: The distribution of the final non-dominated solutions with the lowest IGD values found by MOEA/D-AWA, MOEA/D, Adaptive-MOEA/D, $pa\lambda$-MOEAD, and NSGA-II on the 10-dimensional DTLZ problems (part II).
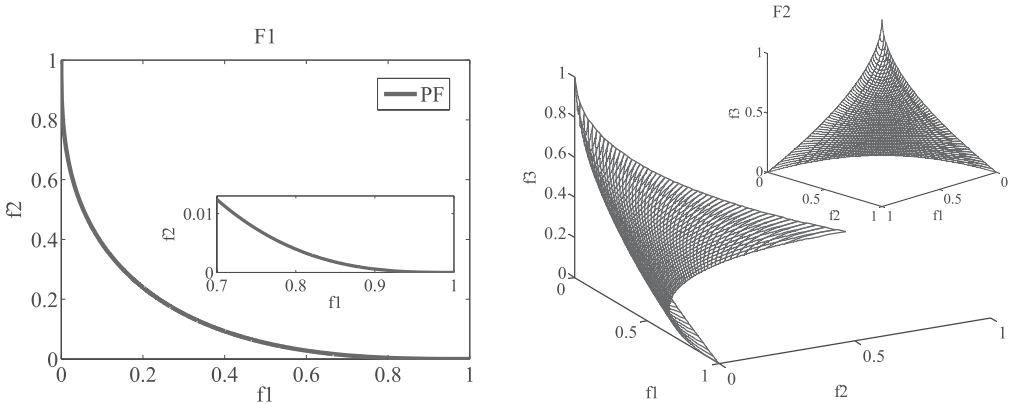
Figure 9: The ideal Pareto-optimal fronts of F1 and F2.

dealing with bi-objective MOPs, the AWA strategy plays an important role in MOEA/D-AWA. As shown in Table 2, MOEA/D-AWA performs as well as or slightly better than MOEA/D on ZDT1, ZDT2, ZDT4, and ZDT6, which have simple PFs. As for ZDT3, where PF is discontinuous, MOEA/D-AWA performs significantly better than MOEA/D and two other MOEA/D based algorithms. This implies that the developed AWA strategy is successful as it is designed particularly for MOPs with complex PFs.

We can conclude from the above results that the weight vector initialization method in MOEA/D-AWA significantly improves the performance of MOEA/D on tri-objective MOPs with simple PF. On the other hand, the AWA strategy works well when the target MOP has a complex PF.

### 4.4.2 Performances on Newly Constructed MOPs with Complex PFs

The above experimental results on ZDT3 and DTLZ6 problems indicate that the proposed MOEA/D-AWA can obtain good performance on the MOPs with discontinuous PF. In this section, we intend to study the performance of MOEA/D-AWA on problems with sharp peak or low tail PFs. The test problem $F_1$ is a newly constructed bi-objective problem which is a variant of ZDT1. And $F_2$ is a tri-objective problem which is a variant of DTLZ2 and described in Deb and Jain (2012a). Their mathematical descriptions can be found in Table 6 in Appendix C. The ideal PSs of F1 and F2 are $F1\_PS = \{(x_1, x_2, \ldots, x_n)|x_1 \in [0, 1], x_2 = x_3 = \cdots = x_n = 0\}$ and $F2\_PS = \{(x_1, x_2, \ldots, x_n)|x_1, x_2 \in [0, 1], x_3 = x_4 = \cdots = x_n = 0.5\}$, respectively. The ideal PFs of $F_1$ and $F_2$ are $F1\_PF = \{(f_1, f_2)|(1 - f_1)^{2.8} + (1 - f_2)^{2.8} = 1, f_1, f_2 \in [0, 1]\}$ and $F2\_PF = \{(f_1, f_2, f_3)|\sqrt{f_1} + \sqrt{f_2} + f_3 = 1, f_1, f_2, f_3 \in [0, 1]\}$, respectively. Figure 9 gives the ideal Pareto-optimal fronts of F1 and F2.

Table 3 lists the IGD metric values obtained by the compared algorithms on $F_1$ and $F_2$. It can be seen from Table 3 that MOEA/D-AWA performs best among the five compared algorithms. The second best algorithm is NSGA-II. NSGA-II performs better than MOEA/D and its two variants Adaptive-MOEA/D and $pa\lambda$-MOEA/D. Adaptive-MOEA/D and $pa\lambda$-MOEA/D perform as well as or better than the original MOEA/D, especially for tri-objective problems. This suggests that weight adjustment does improve MOEA/D significantly in terms of uniformity for the MOPs with complex PFs.

Table 3: Statistic IGD metric values of the solutions founded by the five compared algorithms on the two MOPs with complex PFs. The numbers in parentheses present the standard deviation.*

| Instance | MOEA/D-AWA | MOEA/D | Adaptive-MOEA/D | $pa\lambda$-MOEA/D | NSGA-II |
|---|---|---|---|---|---|
| F1 | **5.204e–3** | 2.531e–2 | 2.331e–2 | 6.111e–3 | 5.404e–3 |
| | **(7.975e–5)** | (7.264e–4) | (2.060e–4) | (3.608e–4) | (2.137e–4) |
| F2 | **1.637e–2** | 4.023e–2 | 3.647e–2 | 1.663e–2 | 1.956e–2 |
| | **(3.104e–4)** | (4.162e–4) | (1.795e–3) | (3.233e–4) | (5.286e–4) |

*Note: The values in bold indicate the best performing algorithm for the particular instance of a test problem.

Figure 10 shows the distribution of the final non-dominated fronts with the lowest IGD values found by MOEA/D-AWA, MOEA/D, Adaptive-MOEA/D, $pa\lambda$-MOEA/D, and NSGA-II for $F_1$ and $F_2$. As shown in Figure 10, MOEA/D-AWA performs better than four other compared algorithms in term of uniformity. For the bi-objective problem $F_1$, MOEA/D-AWA performs as well as NSGA-II and is superior to MOEA/D, Adaptive-MOEA/D, and $pa\lambda$-MOEA/D. It obtains better uniformity among the sharp peak and low tail parts of the target PF. As for the tri-objective problem $F_2$, MOEA/D-AWA performs obviously better than all the other algorithms. It is visually evident that as to the distribution of final solutions, MOEA/D obtains a set of regularly but non-uniformly distributed non-dominated solutions whose distribution is similar to that of the solution mapping vectors of the original MOEA/D with uniformly distributed weight vectors. Adaptive-MOEA/D changes the regular distribution of the non-dominated fronts in MOEA/D, and it performs slightly better than MOEA/D. $pa\lambda$-MOEA/D performs as well as NSGA-II and better than Adaptive-MOEA/D. It obtains a set of uniformly distributed non-dominated solutions, but the points are distributed with some regularity. NSGA-II devotes more effort to the remote and boundary non-dominated solutions. A possible reason for the success of MOEA/D-AWA is that it treats all the subproblems equally.

## 4.5 Effectiveness of the Adaptive Weight Adjustment Strategy

The AWA strategy is designed to enhance the performance of MOEA/D on the MOPs with complex PFs. When solving MOPs with discontinuous PFs, the AWA strategy can remove redundant scalar subproblems whose solution mapping vectors pass through the discontinuous part of the target PF. As for the MOPs whose PFs have a sharp peak and a low tail, the AWA strategy can obtain good uniformity by removing subproblems from the crowded parts and adding new ones to the sparse regions. In this section, experiments are designed to study the effectiveness of the AWA strategy in both cases.

### 4.5.1 The Capability of Recognizing Discontinuous PFs

In this section, ZDT3 and DTLZ6, which are two MOPs with discontinuous PFs, are used to study the capability of the AWA strategy to recognize discontinuous PFs. Moreover, the aim of these experiment studies is to see whether the AWA strategy could help reduce the number of subproblems.

The sets of breakpoints on the PF of ZDT3 and DTLZ6 are plotted in seven solid circles and eight segments, respectively, on the left side of Figure 11, while the right side shows the average number of subproblems over 30 runs converging to the breakpoints in the evolution process. If the Euclidean distance in the objective space between the current solution of a subproblem in the evolutionary population and the number of
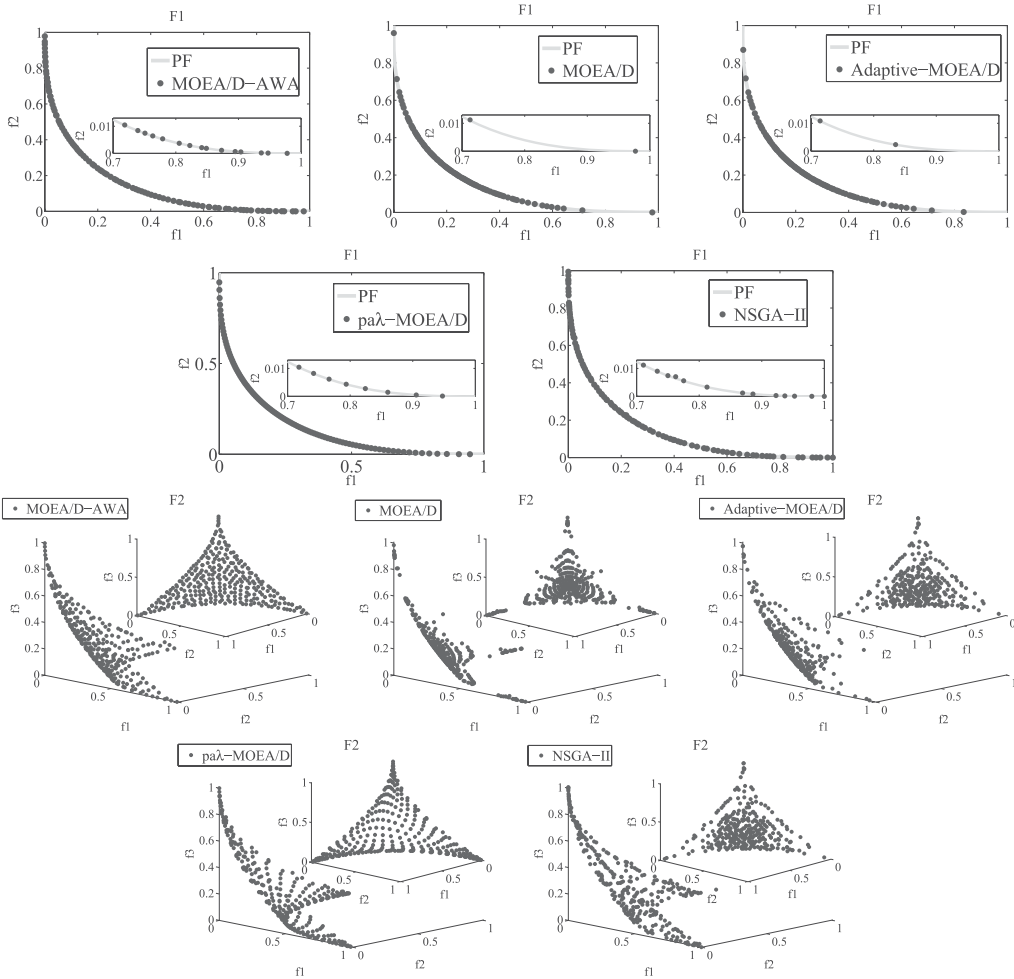
Figure 10: The distribution of the final non-dominated solutions with the lowest IGD values found by MOEA/D-AWA, MOEA/D, Adaptive-MOEA/D, *pa*λ-MOEA/D, and NSGA-II, on the two newly constructed bi-objective and tri-objective MOPs with a sharp peak and a low tail.

breakpoints is less than a small number (which is set $1 \times 10^{-2}$ for ZDT3 and $4 \times 10^{-2}$ for DTLZ6), the subproblem will be considered as a subproblem assembling in the breakpoints. On the other hand, the larger the number of subproblems is assembled in the breakpoints, the more computing effort is wasted.

From Figure 11, it can be seen clearly that AWA does decrease the number of subproblems around the breakpoints of the PF on ZDT3 and DTLZ6. It may be concluded from the above observations that MOEA/D-AWA can recognize the discontinuous parts of the complex PF and reduce the computing effort on searching the discontinuous parts.

However, it is desirable to know which of the two modifications makes a larger contribution to the capability of MOEA/D-AWA. Figure 12 shows the number of subproblems devoted to the discontinuous fields by MOEA/D-AWA and its two variants. Among the two variants of MOEA/D-AWA, OIW-MOEA/D-AWA represents
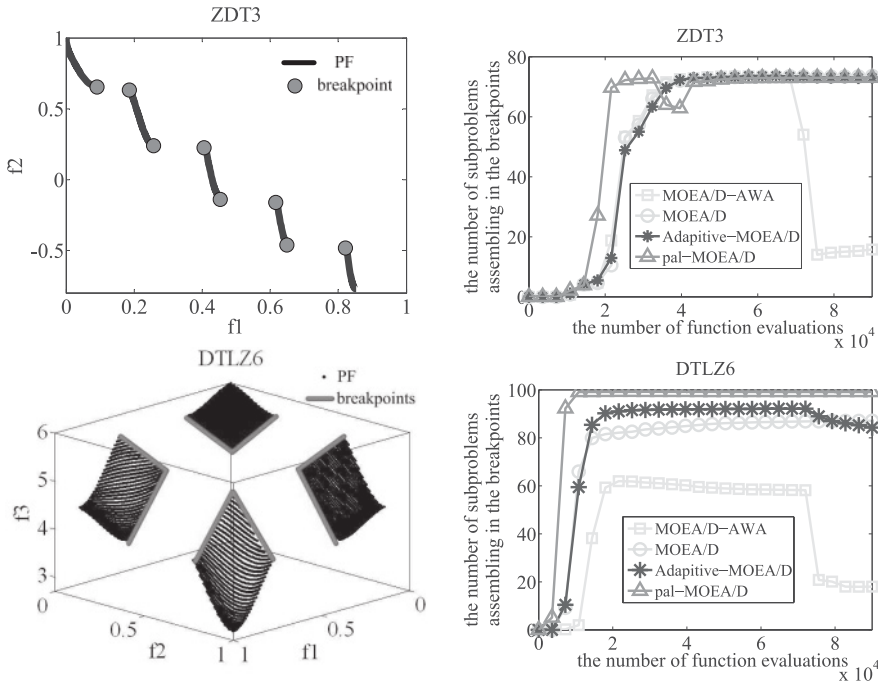
Figure 11: Left plots display the distribution to breakpoints of ZDT3 and DTLZ6; right plots are the number of the current solutions to the subproblems converging to the breakpoints, using MOEA/D-AWA, MOEA/D, Adaptive-MOEA/D, and $pa\lambda$-MOEA/D. Here the population size of MOEA/D is 300, the maximum number of function evaluations is 90,000, and the ratio of iteration times to evolve with MOEA/D only is set as 0.8 in MOEA/D-AWA and Adaptive-MOEA/D.
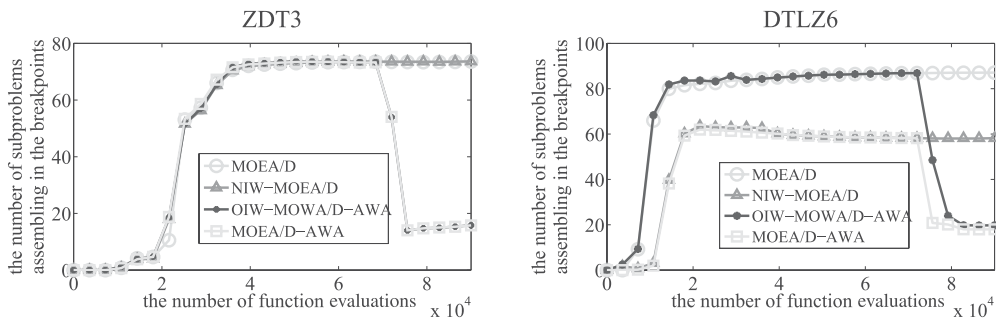


Figure 12: Plot of the number of the current solutions to the subproblems converging to the breakpoints, using MOEA/D, NIW-MOEA/D, OIW-MOEA/D-AWA, and MOEA/D-AWA. Here the population size of MOEA/D is 300, the maximum number of function evaluations is 90,000, and the ratio of iteration times to evolve with MOEA/D only is set as 0.8 in MOEA/D-AWA and OIW-MOEA/D-AWA.

Table 4: Statistic IGD metric values of the solutions founded by MOEA/D-AWA, MOEA/D, OIW-MOEA/D-AWA, and NIW-MOEA/D on the two MOPs with complex PFs. The numbers in parentheses present the standard deviation.*

| Instance | MOEA/D-AWA | MOEA/D | OIW-MOEA/D-AWA | NIW-MOEA/D |
|----------|-----------|--------|----------------|------------|
| F1 | **5.204e–3** | 2.531e–2 | 5.204e–3 | 2.531e–2 |
|  | **(7.975e–5)** | (7.264e–4) | (2.060e–4) | (7.264e–4) |
| F2 | **1.637e–2** | 4.023e–2 | 1.758e–2 | 3.516e–2 |
|  | **(3.104e–4)** | (4.162e–4) | (2.549e–4) | (8.048e–4) |

*Note: The values in bold indicate the best performing algorithm for the particular instance of a test problem.

MOEA/D-AWA with the AWA strategy only. That is, the new weight vector initialization is ignored in OIW-MOEA/D-AWA. While in NIW-MOEA/D, only the new weight vector initialization is employed without the adaptive strategy.

As shown in Figure 12, only the curves of MOEA/D-AWA and OIW-MOEA/D-AWA which are common in employing the AWA strategy, reduce to a low level. These results lead us to the conclusion that the AWA strategy contributes to MOEA/D-AWA with the capability of recognizing discontinuous PFs and saving unnecessary computing effort from searching among discontinuous fields.

### 4.5.2    The Capability of Pursuing Uniformity on PFs with Complex Shape

Experimental results on $F_1$ and $F_2$ in Section 4.4.2 indicate that MOEA/D-AWA can obtain good performance on the MOPs that have PFs with a sharp peak and a low tail. In this section, MOEA/D-AWA is compared with its two variants OIW-MOEA/D-AWA and NIW-MOEA/D on $F_1$ and $F_2$. The aim of the experiment is to see which one of the two modifications really works on enhancing the performance of MOEA/D on the MOPs with complex PFs, or how the two modifications collaborate with each other.

Table 4 shows the mean and standard deviations of the IGD metric values of the final solutions obtained by MOEA/D-AWA, MOEA/D, OIW-MOEA/D-AWA, and NIW-MOEA/D on $F_1$ and $F_2$ with complex PFs over 30 independent runs. As analyzed before, the new weight vector initialization does not work when dealing with bi-objective problems. Hence, for $F_1$, which is bi-objective, the performances of MOEA/D-AWA and OIW-MOEA/D-AWA will be the same, and MOEA/D and NIW-MOEA/D are exactly the same. As shown in Table 4 , MOEA/D-AWA and NIW-MOEA/D perform much better than MOEA/D and OIW-MOEA/D-AWA on $F_1$, which indicates that the AWA does work on pursuing uniformity on the PFs with complex shape. As for the tri-objective problem $F_2$, MOEA/D-AWA performs the best, OIW-MOEA/D-AWA performs better than NIW-MOEA/D, and MOEA/D is the worst. These results suggest that both modifications enhance the performance of MOEA/D, and the AWA strategy is more important.

For a clearer view, we plot in Figure 13 the distribution of the final non-dominated solutions with the lowest IGD values found by MOEA/D-AWA, MOEA/D, OIW-MOEA/D-AWA, and NIW-MOEA/D on $F_1$ and $F_2$. It can be seen from Figure 13 that for $F_1$, MOEA/D-AWA, and OIW-MOEA/D-AWA obtained better uniformity than MOEA/D and NIW-MOEA/D, especially at the sharp peak and low tail parts of the target PF. As for $F_2$, the original MOEA/D obtains a set of non-dominated solutions that are distributed with some regularity and concentrated at the center of the target PF.
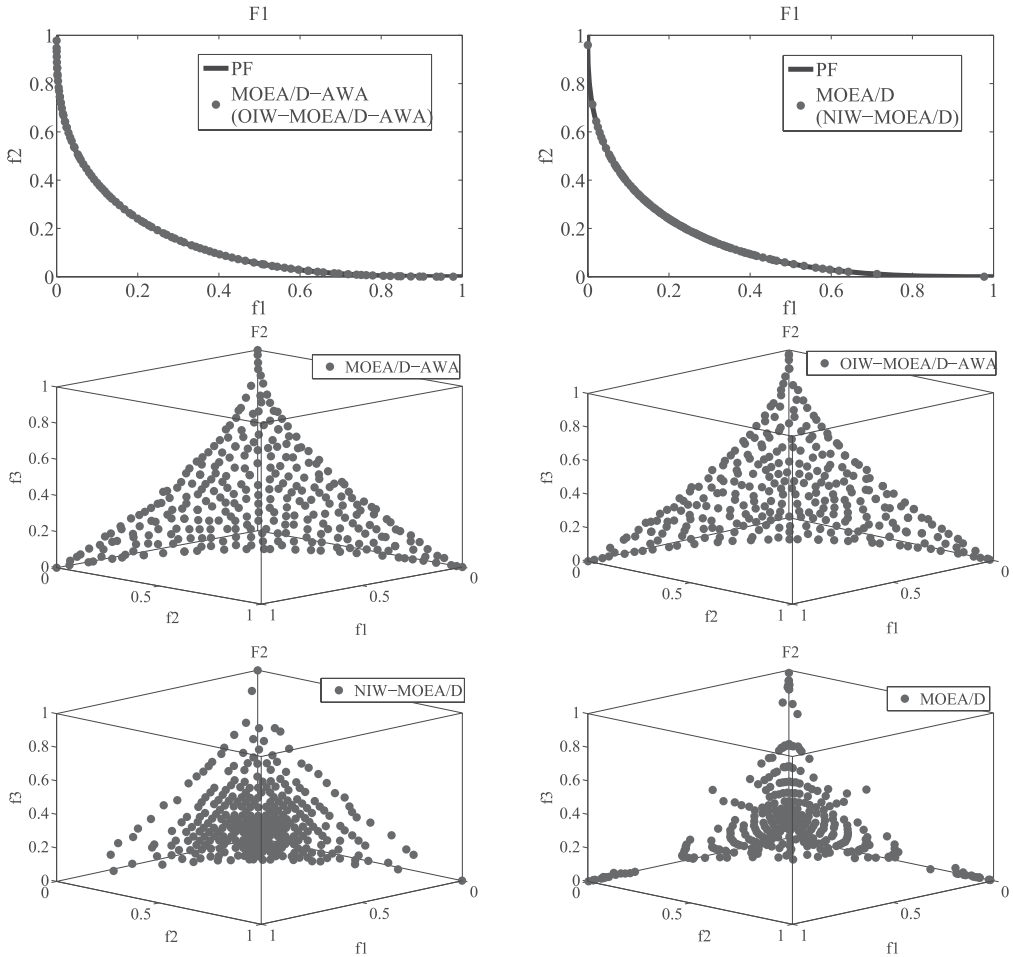
Figure 13: The distribution of the final non-dominated solutions with the lowest IGD values found by MOEA/D-AWA, MOEA/D, OIW-MOEA/D-AWA, and NIW-MOEA/D in solving the two newly constructed MOPs with complex PFs.

By adding the new weight vector initialization method into MOEA/D, the enhanced algorithm NIW-MOEA/D can obtain a set of non-dominated solutions that are uniformly distributed but concentrated at the center of the target PF. This result suggests that the new weight vector initialization method can change the distribution type of the Pareto optimal set, but it fails to obtain good coverage on the sharp peak and low tail parts of the target PF. With the help of the AWA strategy, the enhanced algorithm OIW-MOEA/D-AWA obtains good coverage, but the solutions are distributed under similar regularity to those of MOEA/D. This result suggests that the AWA strategy helps MOEA/D-AWA to achieve better coverage on the sharp peak and low tail parts of the target PF, but it changes the distribution of non-dominated solutions slowly and cannot obtain satisfactory uniformity. MOEA/D-AWA combines the new weight vector initialization method and the AWA strategy together, so the best performance in terms of both coverage and uniformity is achieved.
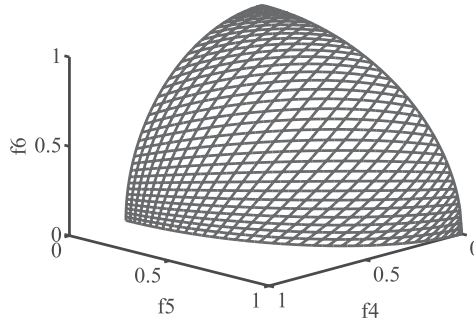
Figure 14: The projection of the ideal PFs of DTLZ5(3,6) and DTLZ4(3,6) on the subspace spanned by $f_4, f_5$, and $f_6$.

Table 5: Statistic IGD metric values of the solutions founded by MOEA/D-AWA, MOEA/D, Adaptive-MOEA/D, and NSGA-II on DTLZ4(3,6) and DTLZ5(3,6) which are many objective problems with low-dimensional PF. The numbers in parentheses present the standard deviation.*

| Instance | MOEA/D-AWA | MOEA/D | Adaptive-MOEA/D | NSGA-II |
|---|---|---|---|---|
| DTLZ4(3,6) | **0.0379** | 0.1778 | 0.0901 | 1.7974 |
| | **(0.0005)** | (0.0090) | (0.0064) | (0.6792) |
| DTLZ5(3,6) | **0.0382** | 0.1946 | 0.0882 | 2.7806 |
| | **(0.0006)** | (0.0160) | (0.0015) | (0.6480) |

*Note: The values in bold indicate the best performing algorithm for the particular instance of a test problem.

### 4.6    Study on Many-Objective Problems

In this section, two test problems DTLZ5($I, m$) and its variation DTLZ4($I, m$) are used to study the ability of the MOEA/D-AWA on many-objective problems. DTLZ5($I, m$) (Deb and Saxena, 2005; Saxena and Deb, 2007; Singh et al., 2011) and DTLZ4($I, m$) are $m$-objective problems with $I$-dimensional Pareto-optimal surface in the $m$-dimensional objective space, where $I < m$. DTLZ5($I, m$) and DTLZ4($I, m$) are many-objective problems with low-dimensional PF in the objective space; thus, their PFs are convenient for the visual display of the distribution of solutions. The definition of DTLZ5(3,6) and DTLZ4(3,6) can be found in Table 6 of Appendix C.

The PFs of DTLZ5(3,6) and DTLZ4(3,6) are surfaces in a three-dimensional subspace of the objective space. The three prime conflict objectives are $f_4, f_5$, and $f_6$. The projection of the obtained solutions on $f_4$, $f_5$, and $f_6$ will reflect the distribution of the whole obtained solutions to a large extent. The ideal PF of DTLZ5(3,6) and DTLZ4(3,6) can be mathematically described as: $\{(f_1, f_2, f_3, f_4, f_5, f_6) | f_4 = \sqrt{2} f_3 = 2 f_2 = 2 f_1, 2 f_4^2 + f_5^2 + f_6^2 = 1, f_4, f_5, f_6 \geq 0\}$. The projection of the ideal PFs of DTLZ5(3,6) and DTLZ4(3,6) on the prime objectives is shown in Figure 14.

Table 5 demonstrates the mean and standard deviations of the IGD metric values found by the benchmark algorithms on DTLZ4(3,6) and DTLZ5(3,6) over 30 independent runs. Figure 15 illustrates the distribution of the final non-dominated solutions
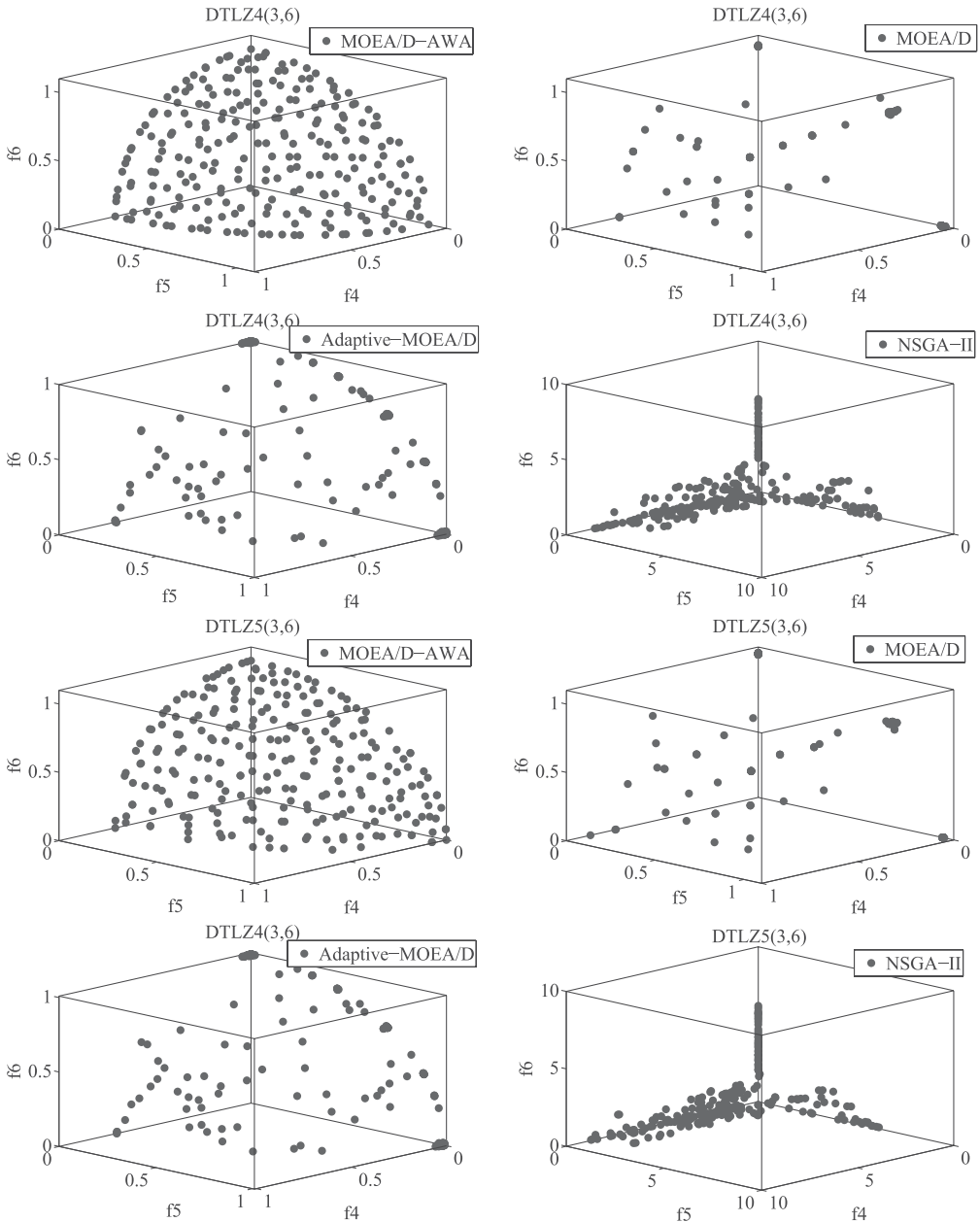
Figure 15: Plot of the non-dominated solutions in a three-dimensional subspace of the objective space with the lowest IGD metric values found by MOEA/D-AWA, MOEA/D, and Adaptive-MOEA/D with the Chebyshev approach and NSGA-II for DTLZ4(3,6) and DTLZ5(3,6) problems.

with the lowest IGD values found by MOEA/D-AWA, MOEA/D, Adaptive-MOEA/D, and NSGA-II on DTLZ4(3,6) and DTLZ5(3,6) problems.

As can be seen in Figure 15, the proposed MOEA/D-AWA performs significantly better than the benchmark algorithms in terms of both uniformity and convergence.

These experimental results suggest that the AWA strategy in MOEA/D-AWA can help MOEA/D to reallocate its computing resource and obtain better diversity, especially for many-objective problems. Adaptive-MOEA/D fails to obtain good diversity on many-objective problems. NSGA-II can obtain good diversity, but it has poor convergence. This is because the non-dominance-based schemes have insufficient selection pressure on the Pareto front (Ishibuchi et al., 2008; Singh et al., 2011). With an increase of the number of objectives, almost all solutions in the current population are on the same rank of non-domination. NSGA-II deteriorates the situation by preferring the remote and boundary non-dominated solutions (Saxena et al., 2010); therefore, it obtains good coverage and diversity but converges slowly.

We can conclude from the above results that, with the help of the AWA strategy, MOEA/D-AWA can obtain good diversity when solving many-objective problems. The computing efforts in MOEA/D-AWA are evenly distributed and have no preference to the boundary solutions.

## 5    Concluding Remarks

In this paper, we have proposed an improved MOEA/D for the MOPs with complex PFs. The proposed MOEA/D-AWA is an enhanced MOEA/D with two major modifications. One is a new weight vector initialization method based on the geometric analysis of the Tchbeycheff decomposition approach, and the other is an adaptive weight vector adjustment (AWA) strategy for dealing with the MOPs with complex PFs. We compare MOEA/D-AWA with four other state of the art algorithms: the original MOEA/D, Adaptive-MOEA/D, $pa\lambda$-MOEAD (which are two enhanced MOEA/D with weight adjustment), and NSGA-II. Experimental studies have been carried out on 10 well-known ZDT and DTLZ instances and two newly constructed instances with specially designed complex PFs having a sharp peak and a low tail. The proposed MOEA/D-AWA approach was also applied to a pair of many-objective problems.

Experimental results indicated that the proposed MOEA/D-AWA approach is able to successfully obtain a well-converged and well diversified set of non-dominated solutions. A detailed study has also indicated that the new weight vector initialization method can significantly improve the performance of MOEA/D on tri-objective MOPs with simple PFs. As for solving the MOPs with discontinuous PFs, the AWA strategy can help MOEA/D to recognize the discontinuous parts of the complex PF and reduce the computing effort on the search among discontinuous fields. We have also shown that when the PF of the target MOP has a shape with a sharp peak and a low tail, the AWA strategy can help MOEA/D to obtain better uniformity, especially at the sharp peak and low tail parts. MOEA/D-AWA has also been applied to many-objective problems with low-dimensional PFs which can be visually displayed. Experimental results have shown that MOEA/D-AWA performs significantly better than the compared algorithms in terms of both uniformity and convergence.

In the field of evolutionary multi-objective optimization (EMO), many research efforts have been performed to develop effective selection mechanisms in the objective space, but few of them have been devoted to the design of evolving strategies in the decision space according to the feature of the MOPs. In our future work, we may introduce machine learning methods into EMO and predict the evolving tendency of the population by learning from the evolutionary history and the distribution of the visited solutions.

## Acknowledgments

## References

Chan, Y., Chiang, T., and Fu, L. (2010). A two-phase evolutionary algorithm for multiobjective mining of classification rules. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1–7.

Chang, P., Chen, S., and Zhang, Q. (2008). MOEA/D for flowshop scheduling problem. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1433–1438.

Chen, C., Chen, C., and Zhang, Q. (2009). Enhancing MOEA/D with guided mutation and priority update for multi-objective optimization. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 209–216.

Coello Coello, C. A. (2006). Evolutionary multi-objective optimization: A historical view of the field. *IEEE Transactions on Computational Intelligence Magazine*, 1(1):28–36.

Corne, D., Jerram, N., Knowles, J., and Oates, M. (2001). PESA-II: Region-based selection in evolutionary multi-objective optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 283–290.

Corne, D. W., Knowles, J. D., and Oates, M. J. (2000). The Pareto-envelope based selection algorithm for multiobjective optimization. In *Parallel problem solving from nature PPSN VI. Lecture notes in computer science*, vol. 1917 (pp. 839–848). Berlin: Springer-Verlag.

Das, I., and Dennis, J. (1998). Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. *SIAM Journal on Optimization*, 8(3):631–657.

Deb, K., Agrawal, S., Pratap, A., and Meyarivan, T. (2002). A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197.

Deb, K., and Beyer, H. G. (2001). Self-adaptive genetic algorithms with simulated binary crossover. *Evolutionary Computation*, 9(2):197–221.

Deb, K., and Jain, H. (2012a). An improved NSGA-II procedure for many-objective optimization, Part I: Solving problems with box constraints. Technical Report 2012009, KanGAL.

Deb, K., and Jain, H. (2012b). An improved NSGA-II procedure for many-objective optimization, Part II: Handling constraints and extending to an adaptive approach. Technical Report 2012010, KanGAL.

Deb, K., and Saxena, D. K. (2005). On finding Pareto-optimal solutions through dimensionality reduction for certain large-dimensional multi-objective optimization problems. Technical Report 2005011, KanGAL.

Fonseca, C., and Fleming, P. (1993). Genetic algorithm for multiobjective optimization: Formulation, discussion and generation. In *Proceedings of the International Conference on Genetic Algorithms*, pp. 416–423.

Gu, F., and Liu, H. (2010). A novel weight design in multi-objective evolutionary algorithm. In *Proceedings of the International Conference on Computational Intelligence and Security*, pp. 137–141.

Horn, J., Nafpliotis, N., and Goldberg, D. (1994). A niched Pareto genetic algorithm for multi-objective optimization. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 82–87.

Huang, W., and Li, H. (2010). On the differential evolution schemes in MOEA/D. In *Proceedings of the International Conference on Natural Computation*, pp. 2788–2792.

Ishibuchi, H., Sakane, Y., Tsukamoto, N., and Nojima, Y. (2009). Adaptation of scalarizing functions in MOEA/D: An adaptive scalarizing function-based multiobjective evolutionary algorithm. In *Proceedings of the International Conference on Evolutionary Multi-Criterion Optimization*, pp. 438–452.

Ishibuchi, H., Sakane, Y., Tsukamoto, N., and Nojima, Y. (2010). Simultaneous use of different scalarizing functions in MOEA/D. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 519–526.

Ishibuchi, H., Tsukamoto, N., and Nojima, Y. (2008). Evolutionary many-objective optimization: A short review. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 2424–2431.

Jiang, S., Cai, Z., Zhang, J., and Ong, Y. S. (2011). Multiobjective optimization by decomposition with Pareto-adaptive weight vectors. In *Proceedings of the International Conference on Natural Computation*, pp. 1260–1264.

Ke, L., Zhang, Q., and Battiti, R. (2010). Multiobjective combinatorial optimization by using decomposition and ant colony. Technical report CES-526, University of Essex.

Knowles, J., and Corne, D. (2000). Approximating the nondominated front using the Pareto archived evolution strategy. *Evolutionary Computation*, 8(2):149–172.

Konstantinidis, A., Charalambous, C., Zhou, A., and Zhang, Q. (2010). Multi-objective mobile agent-based sensor network routing using MOEA/D. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1–8.

Kukkonen, S., and Deb, K. (2006). A fast and effective method for pruning of non-dominated solutions in many-objective problem. In *Proceedings of the International Conference on Parallel Problem Solving from Nature*, pp. 553–562.

Lai, Y. (2009). Multiobjective optimization using MOEA/D with a new mating selection mechanism. Master's thesis, Taiwan Normal University.

Li, H., and Landa-Silva, D. (2011). An adaptive evolutionary multi-objective approach based on simulated annealing. *Evolutionary Computation*, 19(4):561–595.

Li, H., and Zhang, Q. (2009). Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II. *IEEE Transactions on Evolutionary Computation*, 12(2):284–302.

Liu, H., Gu, F., and Cheung, Y. (2010). T-MOEA/D: MOEA/D with objective transform in multi-objective problems. In *Proceedings of the International Conference of Information Science and Management Engineering*, pp. 282–285.

Martinez, Z. S., and Coello Coello, C. A. (2011). A multi-objective particle swarm optimizer based on decomposition. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 69–76.

Mei, Y., Tang, K., and Yao, X. (2011). Decomposition-based memetic algorithm for multi-objective capacitated arc routing problem. *IEEE Transactions on Evolutionary Computation*, 15(2):151–165.

Messac, A., Ismail-Yahaya, A., and Mattson, C. (2003). The normalized normal constraint method for generating the Pareto frontier. *Structural and Multidisciplinary Optimization*, 25(2):86–98.

Miettinen, K. (1999). *Nonlinear multiobjective optimization*. Dordrecht, The Netherlands: Kluwer Academic Publishers.

Miller, B. L., and Goldberg, D. E. (1995). Genetic algorithms, tournament selection, and the effects of noise. *Complex Systems*, 9:193–212.

Moubayed, N., Petrovski, A., and McCall, J. (2010). A novel multi-objective particle swarm optimisation based on decomposition. In *Proceedings of the International Conference on Parallel Problem Solving from Nature*, pp. 1–10.

Pal, S., Das, S., Basak, A., and Suganthan, P. N. (2010). Synthesis of difference patterns for monopulse antennas with optimal combination of array-size and number of subarrays—A multi-objective optimization approach. In *Progress in Electromagnetics Research B*, 21:257–280.

Pal, S., Qu, B., Das, S., and Suganthan, P. N. (2010). Linear antenna array synthesis with constrained multi-objective differential evolution. In *Progress in Electromagnetics Research B*, 21:87–111.

Saxena, D., and Deb, K. (2007). Non-linear dimensionality reduction procedures for certain large-dimensional multi-objective optimization problems: Employing correntropy and a novel maximum variance unfolding. In *Proceedings of the International Conference on Evolutionary Multi-Criterion Optimization*, pp. 772–787.

Saxena, D. K., Duro, J. A., Tiwari, A., Deb, K., and Zhang, Q. (2010). Objective reduction in many-objective optimization: Linear and nonlinear algorithms. Technical Report 2010008, KanGAL.

Schaffer, J. (1985). Multiple objective optimization with vector evaluated genetic algorithms. In *Proceedings of the International Conference on Genetic Algorithms*, pp. 93–100.

Shim, V. A., Tan, K. C., and Tan, K. K. (2012). A hybrid estimation of distribution algorithm for solving the multi-objective multiple traveling salesman problem. In *Proceedings of the WCCI 2012*, pp. 771–778.

Sindhya, K., Ruuska, S., Haanpaa, T., and Miettinen, K. (2011). A new hybrid mutation operator for multiobjective optimization with differential evolution. *Soft Computing—A Fusion of Foundations, Methodologies and Applications*, 15(10):2041–2055.

Singh, H., Isaacs, A., and Ray, T. (2011). A Pareto corner search evolutionary algorithm and dimensionality reduction in many-objective optimization problems. *IEEE Transactions on Evolutionary Computation*, 15(4):539–556.

Srinivas, N., and Deb, K. (1994). Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation*, 2(3):221–248.

Waldocka, A., and Corne, D. (2011). Multiple objective optimisation applied to route planning. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 1827–1834.

Yang, D., Jiao, L., Gong, M., and Feng, J. (2010). Adaptive ranks clone and *k*-nearest neighbour list-based immune multi-objective optimization. *Computational Intelligence*, 26(4):359–380.

Zhang, J., Tang, Q., Zhang, Q., and Feng, Z. (2010). The research on multiple-impulse correction submunition multi-objective optimization based on MOEA/D. *Journal of Projectiles, Rockets, Missiles and Guidance*, 30(2):195–197.

Zhang, Q., and Li, H. (2007). MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6):712–731.

Zhang, Q., Li, H., Maringer, D., and Tsang, E. (2010). MOEA/D with NBI-style Chebyshev approach for portfolio management. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1–8.

Zhang, Q., Liu, W., and Li, H. (2009). The performance of a new version of MOEA/D on CEC09 unconstrained MOP test instances. Technical report, School of CS EE, University of Essex.

Zhang, Q., Liu, W., Tsang, E., and Virginas, B. (2010). Expensive multiobjective optimization by MOEA/D with Gaussian process model. *IEEE Transactions on Evolutionary Computation*, 14(3):456–474.

Zitzler, E., Laumanns, M., and Thiele, L. (2001). SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. In *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems, Proceedings of the Eurogen 2001 Conference*, pp. 95–100.

Zitzler, E., and Thiele, L. (1999). Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Transactions on Evolutionary Computation*, 3(4):257–271.

Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., and Fonseca, V. G. (2003). Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation*, 7(2):117–132.

## Appendix A: Proof of Theorem 1

PROOF: We carry out the proof by contradiction. Suppose that the intersection of the line $\frac{f_1 - z_1^*}{\frac{1}{\lambda_1}} = \frac{f_2 - z_2^*}{\frac{1}{\lambda_2}} = \cdots = \frac{f_m - z_m^*}{\frac{1}{\lambda_m}}(\lambda_i \neq 0, i = 1, 2, \ldots, m)$ and the PF is the optimal solution to the subproblem with weight vector $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m)(\sum_{i=1}^m \lambda_i = 1, \lambda_i > 0, i = 1, 2, \ldots, m)$. In Miettinen (1999), the authors proved that the optimal solution to a Chebyshev subproblem is a Pareto optimal solution. Thus we assume that the optimal solution to the resultant subproblem with weight vector $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m)$ is $\bar{f} = (\bar{f}_1, \ldots, \bar{f}_m)$, $\bar{f} = (\bar{f}_1, \ldots, \bar{f}_m)$ is a Pareto optimal solution but not the point in the line of $\frac{f_1 - z_1^*}{\frac{1}{\lambda_1}} = \frac{f_2 - z_2^*}{\frac{1}{\lambda_2}} = \cdots = \frac{f_m - z_m^*}{\frac{1}{\lambda_m}}$. Then it must have the following non-empty sets:

$$\bar{J} = \{j | \lambda_j \times |\bar{f}_j - z_j^*| < \max_{1 \leq i \leq m}\{\lambda_i \times |\bar{f}_i - z_i^*|\}, j = 1, \ldots, m\} \neq \emptyset \quad (8)$$

$$\bar{K} = \{k | \lambda_k \times |\bar{f}_k - z_k^*| = \max_{1 \leq i \leq m}\{\lambda_i \times |\bar{f}_i - z_i^*|\}, j = 1, \ldots, m\} \neq \emptyset. \quad (9)$$

Therefore,

$$\lambda_j \times |\bar{f}_j - z_j^*| < \lambda_k \times |\bar{f}_k - z_k^*| = \max_{1 \leq i \leq m}\{\lambda_i \times |\bar{f}_i - z_i^*|\}, j \in \bar{J}, k \in \bar{K}.$$

Without loss of generality, we assume that $\bar{f} = (\bar{f}_1, \ldots, \bar{f}_m)$ is an internal point of the PF. Since the ideal PF is piecewise continuous, suppose that $\hat{f} = (\hat{f}_1, \ldots, \hat{f}_m)$ is in the neighborhood of $\bar{f}$, then $\hat{f}$ satisfies the following conditions:

$\hat{f} = (\hat{f}_1, \ldots, \hat{f}_m)$ is a Pareto optimal solution.

$\hat{f}_j > \bar{f}_j, j \in \bar{J}; \hat{f}_k > \bar{f}_k, k \in \bar{K}.$

Since $z_i^* = \min\{f_i(\mathbf{x})|\mathbf{x} \in \Omega\}, i = 1, 2, \ldots, m \Rightarrow z_i^* \leq f_i(\mathbf{x}), i = 1, \ldots, m$, then $|f_i(\mathbf{x}) - z_i^*| = f_i(\mathbf{x}) - z_i^* \geq 0$. Since $\hat{f} = (\hat{f}_1, \ldots, \hat{f}_m)$ is in the $\delta$-neighborhood of $\bar{f} = (\bar{f}_1, \ldots, \bar{f}_m)$, we have

$$\lambda_j \times (\bar{f}_j - z_j^*) < \lambda_j \times (\hat{f}_j - z_j^*) < \lambda_k \times (\hat{f}_k - z_k^*) < \lambda_k \times (\bar{f}_k - z_k^*), j \in \bar{J}, k \in \bar{K}.$$

This implies that

$$g^{tc}(\hat{f}|\lambda, z^*) = \max_{1 \leq i \leq m} \{\lambda_i \times (\hat{f}_i - z_i^*)\} = \max_{k \in \bar{K}} \{\lambda_k \times (\hat{f}_k - z_k^*)\} \tag{10}$$

$$< \max_{k \in \bar{K}} \{\lambda_k \times (\bar{f}_k - z_k^*)\} = g^{tc}(\bar{f}|\lambda, z^*). \tag{11}$$

This contradicts the assumption that $\bar{f}$ is a Pareto optimal solution. ∎

## Appendix B: Proof of Theorem 2

PROOF: Suppose that $\bar{\lambda} = (\bar{\lambda}_1, \dots, \bar{\lambda}_m)$, rather than $\lambda^{\mathrm{opt}}$, is the optimal weight vector to the individual $F = (f_1, \dots, f_m)$ with the reference point $z^* = (z_1^*, \dots, z_m^*)$. That is, $\lambda^{\mathrm{opt}} \neq \bar{\lambda}$ and $h(\bar{\lambda}|F, z^*) < h(\lambda^{\mathrm{opt}}|F, z^*)$. Since $\bar{\lambda}, \lambda^{\mathrm{opt}} \in W_m = \{(w_1, \dots, w_m)|, \sum_{i=1}^m w_i = 1, w_i \geq 0, i = 1, \dots, m\}$ and $\bar{\lambda} \neq \lambda^{\mathrm{opt}}$, if we note that

$$B = \{j|\bar{\lambda}_j > \lambda_j^{\mathrm{opt}}\}, S = \{k|\bar{\lambda}_k < \lambda_k^{\mathrm{opt}}\}, E = \{l|\bar{\lambda}_l = \lambda_l^{\mathrm{opt}}\}$$

we have $B \neq \emptyset, S \neq \emptyset$. Recall that $\prod_{i=1}^m (f_i - z_i^*) \neq 0$, we obtain $0 < \lambda_i^{\mathrm{opt}} < 1, i = 1, \dots, m$.

$$\begin{cases} \forall j \in B, \quad \bar{\lambda}_j > \lambda_j^{\mathrm{opt}} \\ \prod_{i=1}^m (f_i - z_i^*) \neq 0 \Longrightarrow f_i - z_i^* > 0, i = 1, \dots, m \end{cases} \tag{12}$$

$$\Longrightarrow \bar{\lambda}_j \times (f_j - z_j^*) > \lambda_j^{\mathrm{opt}} \times (f_j - z_j^*) = \frac{1}{\sum_{k=1}^m \frac{1}{f_k - z_k^*}}, j \in B$$

$$\begin{cases} \forall k \in S, \quad \bar{\lambda}_k < \lambda_k^{\mathrm{opt}} \\ \prod_{i=1}^m (f_i - z_i^*) \neq 0 \Longrightarrow f_i - z_i^* > 0, i = 1, \dots, m \end{cases} \tag{13}$$

$$\Longrightarrow \bar{\lambda}_k \times (f_k - z_k^*) < \lambda_k^{\mathrm{opt}} \times (f_k - z_k^*) = \frac{1}{\sum_{i=1}^m \frac{1}{f_i - z_i^*}}, k \in S$$

$$\begin{cases} \forall l \in E, \quad \bar{\lambda}_l = \lambda_l^{\mathrm{opt}} \\ \prod_{i=1}^m (f_i - z_i^*) \neq 0 \Longrightarrow f_i - z_i^* > 0, i = 1, \dots, m \end{cases} \tag{14}$$

$$\Longrightarrow \bar{\lambda}_l \times (f_l - z_l^*) = \lambda_l^{\mathrm{opt}} \times (f_l - z_l^*) = \frac{1}{\sum_{i=1}^m \frac{1}{f_i - z_i^*}}, l \in E.$$

So $\forall k \in S, l \in E, j \in B$, we have $\bar{\lambda}_k \times (f_k - z_k^*) < \frac{1}{\sum_{i=1}^m \frac{1}{f_i - z_i^*}} = \bar{\lambda}_l \times (f_l - z_l^*) < \bar{\lambda}_j \times (f_j - z_j^*)$. Because $B \neq \emptyset$, we can get

$$h(\bar{\lambda}|F, z^*) = \max_{1 \leq i \leq m} \{\bar{\lambda}_i \times (f_i - z_i^*)\} > \frac{1}{\sum_{i=1}^m \frac{1}{f_i - z_i^*}}$$

$$= \max_{1 \leq i \leq m} \{\lambda_i^{\mathrm{opt}} \times (f_i - z_i^*)\} = h(\lambda^{\mathrm{opt}}|F, z^*). \tag{15}$$

This conclusion conflicts with the assumption. ∎

## Appendix C: The Test Instances

Table 6 shows the mathematical definition of the test instances.

Table 6: Test problems used in this study.

| Instance | Variable bound | Objective functions |
|---|---|---|
| F1 | $[0,1]^n$ <br> $n = 30$ | $f_1(x) = x_1$ <br> $f_2(x) = g(x)\{1 - [1 - (1 - x_1)^M]^{\frac{1}{M}}\}$ <br> where $g(x) = 1 + 9(\sum_{i=2}^n x_i)/(n-1); M = 2.8;$ |
| F2 | $[0,1]^n$ <br> $n = 10$ | $f_1(x) = \{[1 + g(x)]\cos(0.5\pi x_1)\cos(0.5\pi x_2)\}^4$ <br> $f_2(x) = \{[1 + g(x)]\cos(0.5\pi x_1)\sin(0.5\pi x_2)\}^4$ <br> $f_3(x) = \{[1 + g(x)]\sin(0.5\pi x_1)\}^2$ <br> where $g(x) = \sum_{i=3}^n (x_i - 0.5)^2$ |
| DTLZ4(3,6) | $[0,1]^n$ <br> $n = 26$ | $f_1(x) = [1 + g(x)]\cos(\theta_1)\cos(\theta_2)\cos(\theta_3)\cos(\theta_4)\cos(\theta_5)$ <br> $f_2(x) = [1 + g(x)]\cos(\theta_1)\cos(\theta_2)\cos(\theta_3)\cos(\theta_4)\sin(\theta_5)$ <br> $f_3(x) = [1 + g(x)]\cos(\theta_1)\cos(\theta_2)\cos(\theta_3)\sin(\theta_4)$ <br> $f_4(x) = [1 + g(x)]\cos(\theta_1)\cos(\theta_2)\sin(\theta_3)$ <br> $f_5(x) = [1 + g(x)]\cos(\theta_1)\sin(\theta_2)$ <br> $f_6(x) = [1 + g(x)]\sin(\theta_1)$ <br> $\theta_i = \frac{\pi x_i^\alpha}{2}$ $i = 1, 2;$ $\theta_i = \frac{\pi}{4[1+g(x)]}[1 + 2g(x)x_i^\alpha]$ $i = 3, 4, 5$ <br> where $g(x) = \sum_{i=6}^n (x_i - 0.5)^2, \alpha = 100$ |
| DTLZ5(3,6) | $[0,1]^n$ <br> $n = 26$ | $f_1(x) = [1 + g(x)]\cos(\theta_1)\cos(\theta_2)\cos(\theta_3)\cos(\theta_4)\cos(\theta_5)$ <br> $f_2(x) = [1 + g(x)]\cos(\theta_1)\cos(\theta_2)\cos(\theta_3)\cos(\theta_4)\sin(\theta_5)$ <br> $f_3(x) = [1 + g(x)]\cos(\theta_1)\cos(\theta_2)\cos(\theta_3)\sin(\theta_4)$ <br> $f_4(x) = [1 + g(x)]\cos(\theta_1)\cos(\theta_2)\sin(\theta_3)$ <br> $f_5(x) = [1 + g(x)]\cos(\theta_1)\sin(\theta_2)$ <br> $f_6(x) = [1 + g(x)]\sin(\theta_1)$ <br> $\theta_i = \frac{\pi x_i}{2}$ $i = 1, 2;$ $\theta_i = \frac{\pi}{4[1+g(x)]}[1 + 2g(x)x_i]$ $i = 3, 4, 5$ <br> where $g(x) = \sum_{i=6}^n (x_i - 0.5)^2$ |