

Another Difficulty of Inverted Triangular Pareto Fronts for Decomposition-Based Multi-Objective Algorithms

Linjun He

Southern University of Science and Technology
Shenzhen, China
this.hej@gmail.com

Auraham Camacho

CINVESTAV Unidad Tamaulipas
Mexico
acamacho@tamps.cinvestav.mx

Hisao Ishibuchi*

Southern University of Science and Technology
Shenzhen, China
hisao@sustech.edu.cn

ABSTRACT

A set of uniformly sampled weight vectors from a unit simplex has been frequently used in decomposition-based multi-objective algorithms. The number of the generated weight vectors is controlled by a user-defined parameter H . In the literature, good results are often reported on test problems with triangular Pareto fronts since the shape of the Pareto fronts is consistent with the distribution of the weight vectors. However, when a problem has an inverted triangular Pareto front, well-distributed solutions over the entire Pareto front are not obtained due to the inconsistency between the Pareto front shape and the weight vector distribution. In this paper, we demonstrate that the specification of H has an unexpected large effect on the performance of decomposition-based multi-objective algorithms when the test problems have inverted triangular Pareto fronts. We clearly explain why their performance is sensitive to the specification of H in an unexpected manner (e.g., $H = 3$ is bad but $H = 4$ is good for three-objective problems whereas $H = 3$ is good but $H = 4$ is bad for four-objective problems). After these discussions, we suggest a simple weight vector specification method for inverted triangular Pareto fronts.

CCS CONCEPTS

- Mathematics of computing → Evolutionary algorithms;

KEYWORDS

Decomposition-based multi-objective algorithm, weight vectors, Pareto front shape, multi-objective optimization, many-objective optimization

ACM Reference Format:

Linjun He, Auraham Camacho, and Hisao Ishibuchi. 2020. Another Difficulty of Inverted Triangular Pareto Fronts for Decomposition-Based Multi-Objective Algorithms. In *Genetic and Evolutionary Computation Conference (GECCO '20), July 8–12, 2020, Cancún, Mexico*. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3377930.3390196>

*Corresponding author: Hisao Ishibuchi

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

GECCO '20, July 8–12, 2020, Cancún, Mexico
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7128-5/20/07...\$15.00
<https://doi.org/10.1145/3377930.3390196>

1 INTRODUCTION

The decomposition-based strategy has attracted a lot of attention in the evolutionary multi-objective optimization (EMO) community. The prototype of decomposition-based algorithms, MOEA/D [22], provides a basic framework for algorithm design. Within this framework, a set of pre-defined weight vectors is used to decompose a multi-objective problem (MOP) into several subproblems. Since MOEA/D is an effective and efficient framework of EMO algorithms, a number of decomposition-based EMO algorithms have been proposed in recent years, including NSGA-III [4], θ -DEA [21], and MOEA/DD [16]. These algorithms work very well on the commonly-used continuous and scalable test suites DTLZ [5] and WFG [8] with up to 15 objectives. Thanks to the pre-defined weight vectors, uniformly distributed solutions are obtained for these test suites. However, when the weight vectors are not consistent with the shape of the Pareto front, the diversity of the obtained solutions can be deteriorated severely. Several studies [1, 6, 11, 14, 17] suggest that MOEA/D is not likely to work well on the minus-DTLZ [14], minus-WFG [14] and IDTLZ [15] test suites, which are known for their inverted triangular Pareto fronts. That is, a large number of solutions are obtained on the boundaries of the Pareto fronts.

A method proposed by Das and Dennis [3] is commonly employed to create a set of N evenly-distributed weight vectors. The value of N depends on the number of objectives m and a resolution parameter H which controls the number of divisions of each objective axis.

The relationship between the distribution of weight vectors and the shape of the Pareto front is important to obtain well-distributed solutions over the entire Pareto front. To illustrate this, let us apply MOEA/D with the PBI function to the 3- and 5-objective DTLZ1 and minus-DTLZ1 problems. We use enough computation load (i.e., 36,540 and 131,670 solution evaluations for the 3- and 5-objective problems, respectively). For details of our computational experiments, see Section 3. We set the reference point of the weight vectors in MOEA/D as the true ideal point (i.e., $(0, 0, \dots, 0)$ for DTLZ1 and $(-551.125, -551.125, \dots, -551.125)$ for minus-DTLZ1). Here, we employ a commonly-used specification of H for each problem, that is, $H = 12$ and $H = 6$ for 3- and 5-objective problems, respectively. Figures 1 and 2 show the experimental results in the normalized objective space with the ideal point $(0, 0, \dots, 0)$ and the nadir point $(1, 1, \dots, 1)$. The obtained solutions are shown in red and the true Pareto fronts are shown in gray.

We can see that uniformly distributed solutions are obtained by MOEA/D on the 3-objective DTLZ1 with $H = 12$ in Figure 1 (a) and the 5-objective DTLZ1 with $H = 6$ in Figure 2 (a). On the 3-objective minus-DTLZ1, as shown in Figure 1 (b), whereas the solutions

obtained by MOEA/D cover the entire Pareto front, they are not uniformly distributed. It should be noted that the vertices (i.e., the three extreme solutions) of the inverted triangular Pareto front are obtained. One may think that the extreme solutions of an inverted triangular Pareto front can be easily found since many solutions are obtained on the boundaries of the Pareto front. However, as shown in Figure 2 (b), even when the true ideal point is used as the reference point, the vertices of the Pareto front (i.e., extreme solutions with the minimum objective value for each objective) of the 5-objective minus-DTLZ1 are not obtained. No solutions close to the extreme solutions are obtained in Figure 2 (b). Similar results are reported in [11] by MOEA/D. These experimental results show that MOEA/D has two difficulties in its application to an MOP with an inverted triangular Pareto front: (i) many solutions are obtained on the boundary of the Pareto front, and (ii) the extreme solutions are not always obtained.

The first difficulty (i.e., concentration of solutions on the boundary of the inverted triangular Pareto front) is not observed in some other decomposition-based multi-objective algorithms (e.g., NSGA-III [4], θ -DEA [21], MOEA/DD [16]). However, all extreme solutions are not always obtained for inverted triangular Pareto fronts as reported in [1, 6, 11, 14, 17, 18, 20] (i.e., the second difficulty). Even for decomposition-based multi-objective algorithms with weight vector adaptation mechanisms (e.g., RVEA [2]), it is difficult to obtain well-distributed solutions over inverted triangular Pareto fronts [1, 6, 17, 20]. An interesting observation in the literature [6, 18, 20] is that similar results were also obtained by indicator-based algorithms (e.g., MOMBI-II [7]) with weight vectors for hypervolume approximation in its application to multi-objective problems with

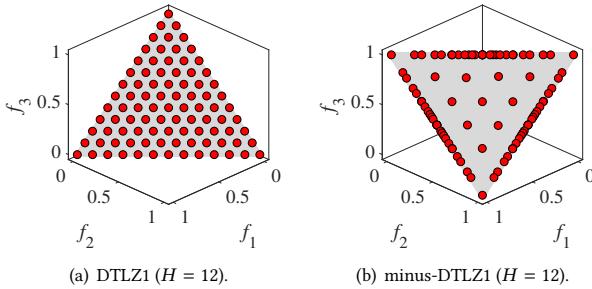


Figure 1: Pareto optimal solutions (gray) and solutions (red) obtained by MOEA/D on the 3-objective DTLZ1 and minus-DTLZ1.

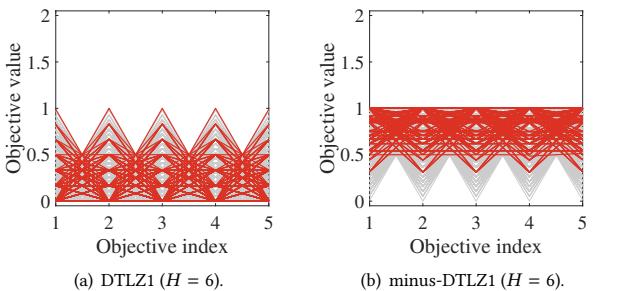


Figure 2: Pareto fronts (gray) and solutions (red) obtained by MOEA/D on the 5-objective DTLZ1 and minus-DTLZ1.

inverted triangular Pareto fronts. In this paper, we discuss the second difficulty, explain its reason, and suggest its remedy.

This paper is organized as follows. In Section 2, we briefly explain frequently-used weight vector specification methods. In Section 3, we explain why the extreme solutions of the inverted triangular Pareto fronts are not always obtained by MOEA/D. It is also demonstrated that the specification of H has a dominant effect on whether the extreme solutions are obtained or not. In Section 4, we propose a specification method of H . It is shown that the performance of MOEA/D is significantly improved by specifying the value of H according to the proposed method for MOPs with inverted triangular Pareto fronts. In Section 5, we demonstrate that the proposed method also improves the performance of NSGA-III, RVEA and MOMBI-II. Finally, we conclude this paper in Section 6.

2 WEIGHT VECTOR SPECIFICATION

2.1 Das and Dennis's Method

The method proposed by Das and Dennis [3] is commonly-used for generating a set of weight vectors uniformly. By dividing each objective axis with a uniform spacing $\Delta = 1/H$ (where $H(> 0)$ is the number of divisions along each objective axis), a set of weight vectors $\{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_N\}$ are uniformly sampled from a normalized hyperplane (i.e., a unit simplex). The weight vectors $\mathbf{w} = (w_1, w_2, \dots, w_m)$ generated by this method satisfy the following conditions:

$$\begin{aligned} w_1 + w_2 + \dots + w_m &= 1, \\ w_i &\in \left\{0, \frac{1}{H}, \dots, \frac{H}{H}\right\} \text{ for } i = 1, 2, \dots, m. \end{aligned} \quad (1)$$

The number of the generated weight vectors is calculated as

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

where m is the number of objectives.

In Figure 3, we show 15 weight vectors generated for a 3-objective optimization problem with four divisions for each axis ($H = 4$).

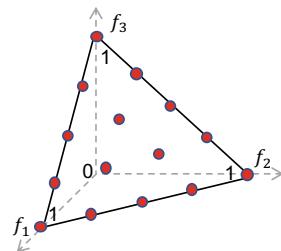


Figure 3: An example of the set of the generated weight vectors by Das and Dennis's method ($m = 3$ and $H = 4$).

2.2 Two-layered Weight Vector Generation

When H is specified as $H < m$, all the weight vectors generated by Das and Dennis's method are boundary weight vectors where at least one weight element is zero (i.e., $w_i = 0$ for at least one index i). When H is specified as $H = m$, we have only a single intermediate weight vector with $0 < w_i < 1$ for $i = 1, 2, \dots, m$. However, even when H is specified as $H = m$ (i.e., even when we have only a single intermediate weight vector), the total number of generated weight vectors is huge for many-objective problems. To generate more

intermediate weight vectors, we need to further increase the total number of weight vectors.

To address this issue, a two-layered weight vector generation method [4, 16] is usually used to generate an practically tractable number of intermediate and boundary weight vectors. Two sets of weight vectors (called a boundary layer and an inner layer, respectively) are generated using Das and Dennis's method. Two parameters, H_1 and H_2 , control the number of weight vectors on the boundary and inner layers, respectively. The set of weight vectors on the inner layer is then combined with the boundary layer after applying the following transformation:

$$w_i = \frac{1}{2}(w_i + \frac{1}{m}) \text{ for } i = 1, 2, \dots, m. \quad (3)$$

The total number of generated weight vectors is

$$N = \binom{H_1 + m - 1}{m - 1} + \binom{H_2 + m - 1}{m - 1}. \quad (4)$$

Figure 4 illustrates the two-layered weight vector generation method. A boundary layer with $H_1 = 2$ and an inner layer with $H_2 = 1$ are generated by Das and Dennis's method. Then, they are merged into a single set of weight vectors.

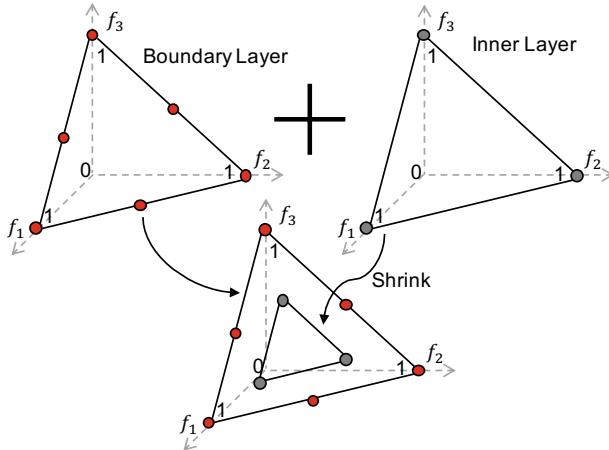


Figure 4: An example of the set of the weight vectors generated by the two-layered weight vector generation method with $m = 3$, $H_1 = 2$ (i.e., for the boundary layer) and $H_2 = 1$ (i.e., for the inner layer).

2.3 Typical Settings of Population Size

The population size (N) of decomposition-based evolutionary algorithms corresponds to the number of weight vectors, which depends on the value of H . Table 1 shows frequently-used specifications of H in the literature [4, 14, 16, 21, 22]. Das and Dennis's method is used for 3- and 5-objective problems, while the two-layered weight vector generation method is used for problems with 8 or more objectives. As shown in Table 1, one parameter H is needed in Das and Dennis's method, while two parameters (H_1, H_2) are needed in the two-layered approach.

3 EXPERIMENTAL STUDIES

In order to clearly show the relation between the setting of H and the coverage of Pareto fronts by obtained solutions, we examined

Table 1: Frequently-used specification of H and the corresponding number of weight vectors.

Number of objectives m	Divisions H (H_1, H_2)	Number of weight vectors N
3	12	91
5	6	210
8	(3, 2)	(120 + 36)
10	(3, 2)	(220 + 55)
15	(2, 1)	(120 + 15)

MOEA/D with various settings of the parameter H on the widely-used continuous and scalable test suite DTLZ [5] and its minus version (i.e., minus-DTLZ [14]) with 3, 4, and 5 objectives. Our experimental studies are performed in a similar manner to [21]. All the algorithms are implemented in the PlatEMO framework [19] with the following settings:

- Scalarizing function: PBI function (penalty parameter $\theta = 5$).
- Population size N : N is specified by H .
- Value of H : 2, 3, 4, 5, 6, 7.
- Crossover and mutation:
 - Simulated binary crossover (distribution index: 20) with crossover probability 1.0.
 - Polynomial mutation (distribution index: 20) with mutation probability $1/n$ where n is the number of variables.
- Termination condition (the number of solution evaluations): 36,540 (3-objective problems), 64,680 (4-objective problems), and 131,670 (5-objective problems).

We calculate the average hypervolume value over 31 runs of each algorithm on each problem. The objective space of each problem is normalized so that the ideal and nadir points are $(0, 0, \dots, 0)$ and $(1, 1, \dots, 1)$, respectively. The reference point $r = (r, r, \dots, r)$ for hypervolume calculation is specified for each problem as $r = 2$ in order to intentionally increase the HV contributions of the extreme points [13]. When we show experimental results, a single run with the median hypervolume value is selected from the 31 runs of each algorithm on each test problem.

3.1 Experimental Results on DTLZ1-4

DTLZ1-4 [5] have regular triangular Pareto fronts, which are consistent with the distribution of weight vectors. MOEA/D with various settings of H on DTLZ1-4 are examined. We found MOEA/D with any setting of $H > 0$ is able to obtain a well-distributed solution set on the entire Pareto front. That is, the extreme solutions of the Pareto front are obtained, regardless of the value of H . This observation is consistent with the results reported in [14]. Due to the page limit, we only show results on DTLZ1 in Figure 5.

In Figure 6, we show the average hypervolume value over 31 runs of MOEA/D for various specifications of $H = 2, 3, \dots, 8$ on the 3-objective DTLZ1 problem. We can see from Figure 6 that the average hypervolume values monotonically increase as the value of H increases (i.e., with the increase in the population size). This observation suggests that there is no strong dependency of the performance of MOEA/D on the specification of H when it is applied to the 3-objective DTLZ1 problem. Similar observations are obtained for all of DTLZ1-4 with 3-5 objectives.

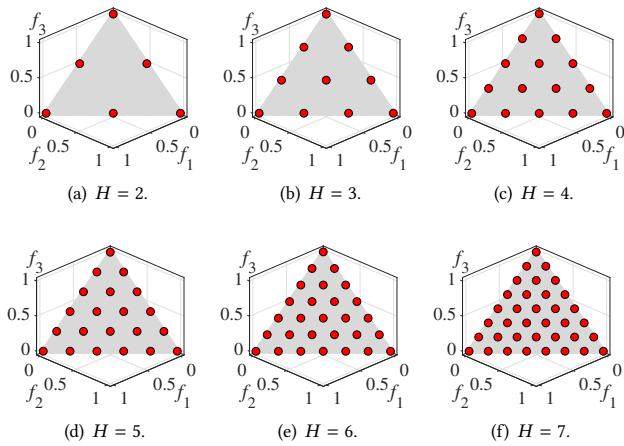


Figure 5: Obtained solutions by MOEA/D with various values of H on the 3-objective DTLZ1 problem.

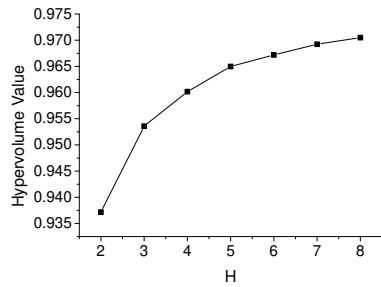


Figure 6: Average hypervolume value by each specification of H on the 3-objective DTLZ1 problem.

3.2 Experimental Results on minus-DTLZ1-4

3.2.1 Results on the 3-objective minus-DTLZ1. Figure 7 shows experimental results of MOEA/D with $H = 2, 3, \dots, 7$ on the 3-objective minus-DTLZ1 problem in the normalized objective space. From Figure 7, we can see that the three extreme solutions are not obtained when $H = 3, 5, 7$. That is, the three extreme solutions are obtained only when H is an even number. Let us explain the reason for this observation. In Figure 8 (a) with $H = 2$, each extreme solution has a weight vector at the same location. That is, each extreme solution is the best solution for the corresponding weight vector. As a result, it is likely that all extreme solutions are obtained by MOEA/D with $H = 2$ as shown in Figure 7 (a) and Figure 8 (a). However, in Figure 8 (b) with $H = 3$, each extreme solution has no corresponding weight vector. Each extreme solution is not the best solution for any weight vector. As a result, no extreme solution is obtained by MOEA/D with $H = 3$ as shown in Figure 7 (b) and Figure 8 (b). As explained in Figure 7 and Figure 8, an extreme solution is not obtained by MOEA/D when it is not the best solution for any weight vector. This is clear from the search strategy of MOEA/D (i.e., to search for the best solution for each weight vector). However, it has not been clearly explained in the literature why the extreme solutions of inverted triangular Pareto fronts cannot be obtained by MOEA/D when the specification of H is not appropriate. This is because test problems with inverted triangular Pareto fronts have not been used in many studies whereas they seem to be more

realistic than test problems with regular (i.e., triangular) Pareto fronts [12]. Basically similar (whereas somewhat different due to the difference in the curvature of the Pareto front) are obtained for minus-DTLZ2-4.

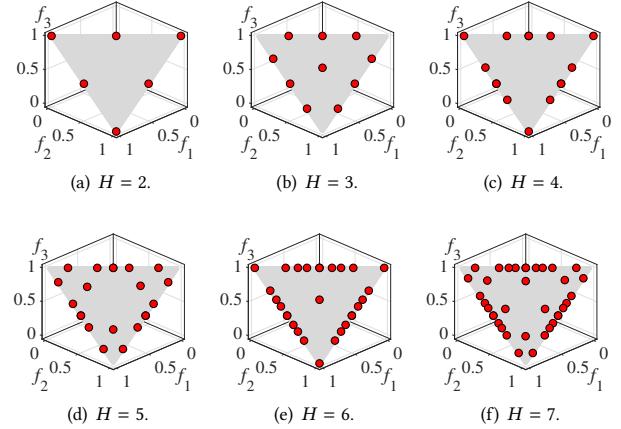


Figure 7: Obtained solutions by MOEA/D with various values of H on the 3-objective minus-DTLZ1 problem.

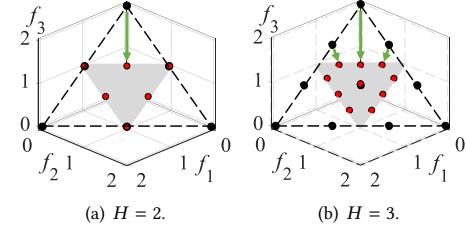
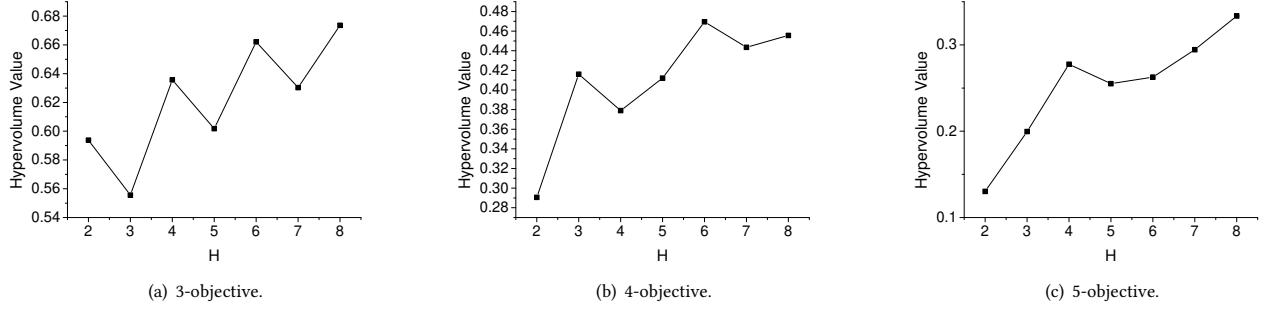
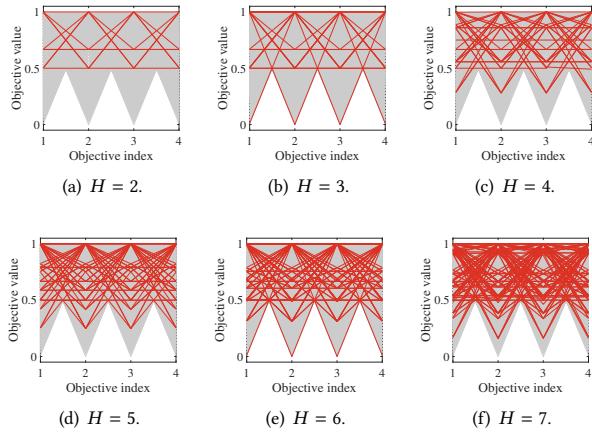
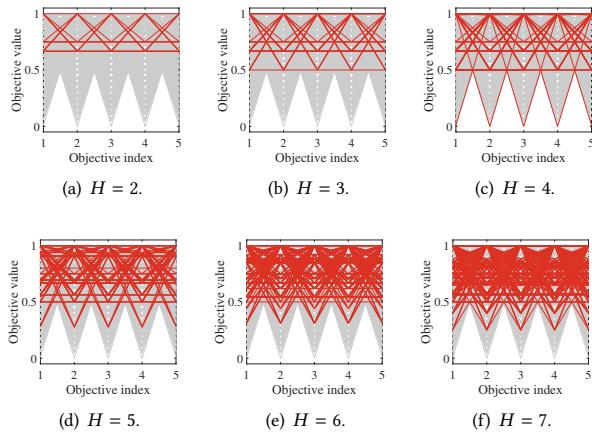


Figure 8: Relation between the weight vector distribution and the obtained solution set in the normalized objective space using the estimated ideal and nadir point.

In Figure 9 (a), we show the average hypervolume value over 31 runs of MOEA/D with each specification of H on the 3-objective minus-DTLZ1. Figure 9 (a) clearly shows that good results are obtained when $H = 2, 4, 6, 8$. This observation is consistent with Figure 7. Similar observations are obtained for the 3-objective minus-DTLZ2-4.

3.2.2 Results on the 4-objective minus-DTLZ1. Figure 10 shows experimental results of MOEA/D with $H = 2, 3, \dots, 7$ on the 4-objective minus-DTLZ1 problem in the normalized objective space using parallel coordinates graphs. Red lines show the obtained solutions from a single run with the median hypervolume value among 31 runs. In Figure 10, the Pareto front is approximated by 9,880 Pareto optimal solutions uniformly sampled from the Pareto front. From Figure 10, we can see that the four extreme solutions are obtained only when $H = 3$ and 6 . The reasons for these results are discussed in the next section in detail. Similar results are obtained for minus-DTLZ2-4.

In Figure 9 (b), we show the average hypervolume value over 31 runs of MOEA/D with each specification of H on the 4-objective minus-DTLZ1. Figure 9 (b) clearly shows that good results are obtained when $H = 3, 6$. This observation is consistent with Figure 10.

Figure 9: Average hypervolume value by each specification of H on the 3-, 4-, and 5-objective minus-DTLZ1 problem.Figure 10: Experimental results by MOEA/D with various values of H on the 4-objective minus-DTLZ1 problem.Figure 11: Experimental results by MOEA/D with various values of H on the 5-objective minus-DTLZ1 problem.

Similar observations are obtained for the 4-objective minus-DTLZ2-4.

3.2.3 Results on the 5-objective minus-DTLZ1. In the same manner as Figure 10, Figure 11 shows experimental results of MOEA/D with $H = 2, 3, \dots, 7$ on the 5-objective minus-DTLZ1 in the normalized objective space using parallel coordinates graphs. The Pareto

front is approximated by 9,880 points uniformly sampled from the Pareto front. From Figure 11, we can see that the five extreme solutions are obtained only when $H = 4$. The reasons for this result are discussed in the next section in detail. Similar results are obtained for minus-DTLZ2-4.

In Figure 9 (c), we show the average hypervolume value over 31 runs of MOEA/D with each specification of H on the 5-objective minus-DTLZ1 problem. Figure 9 (c) clearly shows that good results are obtained when $H = 4, 8$. This observation is consistent with Figure 11. Similar observations are obtained for other 5-objective minus-DTLZ problems.

4 PROPOSED METHOD

4.1 Analysis of Experimental Results

From our experimental results, we can see that good results were obtained for m -objective minus-DTLZ problems from $H = 2, 4, 6, 8$ ($m = 3$), $H = 3, 6$ ($m = 4$), and $H = 4, 8$ ($m = 5$). That is, the extreme solutions of the Pareto front of a minus-DTLZ problem can be obtained only when H is specified as $H = (m - 1) \times j$ where j is a positive integer (i.e., H should be a multiple of $m - 1$). However, for DTLZ with regular triangular Pareto fronts, their extreme solutions can be obtained for any specification of H .

As explained in [9, 10], each solution obtained by MOEA/D is close to its corresponding weight vector when the penalty parameter θ is large as in our computational experiments with $\theta = 5$. Thus, it is likely that there exist no weight vectors close to the extreme solutions when H is not a multiple of $(m - 1)$.

Let us further discuss the above explanation using the DTLZ1 and minus-DTLZ1 problems in the normalized objective space. The Pareto front of the normalized m -objective DTLZ1 can be represented as follows: $f_1 + f_2 + \dots + f_m = 1$. For the 3-objective DTLZ1, the Pareto front is defined as $f_1 + f_2 + f_3 = 1$ ($0 \leq f_i \leq 1$ for $i = 1, 2, 3$) where its three vertices are $(1, 0, 0)$, $(0, 1, 0)$, and $(0, 0, 1)$. The weight vectors corresponding to these three vertices are $(1, 0, 0)$, $(0, 1, 0)$, and $(0, 0, 1)$, respectively. Regardless of the specification of H , these three weight vectors are generated by the method of Das and Dennis. We can extend this discussion to the case of many-objective problems. The normalized m -objective DTLZ1 has m extreme solutions: $(1, 0, \dots, 0)$, $(0, 1, 0, \dots, 0)$, ..., $(0, \dots, 0, 1)$. Regardless of the value of H , the corresponding weight vectors are always generated. Since triangular Pareto fronts have these extreme solutions in the normalized objective space, we can use any value of H .

H depending on the required number of solutions (i.e., depending on the required population size).

The Pareto front of the normalized m -objective minus-DTLZ1 is represented as follows: $f_1 + f_2 + \dots + f_m = m - 1$. For the 3-objective minus-DTLZ1, the Pareto front is defined as $f_1 + f_2 + f_3 = 2$ ($0 \leq f_i \leq 1$ for $i = 1, 2, 3$) where its vertices are $(0, 1, 1)$, $(1, 0, 1)$, and $(1, 1, 0)$. In order to obtain these extreme solutions, the following three weight vectors are needed: $(0, 0.5, 0.5)$, $(0.5, 0, 0.5)$, and $(0.5, 0.5, 0)$. For 3-objective problems, these weight vectors are generated only when H is a multiple of 2 (see Eq. (1) where $w_i = 0.5$ can be generated only when H is a multiple of 2). Similarly, the extreme solutions of the Pareto front of the 5-objective minus-DTLZ1 are $(0, 1, 1, 1, 1)$, $(1, 0, 1, 1, 1)$, $(1, 1, 0, 1, 1)$, $(1, 1, 1, 0, 1)$, and $(1, 1, 1, 1, 0)$. In order to obtain the five extreme solutions, we need the following five weight vectors: $(0, 0.25, 0.25, 0.25, 0.25)$, $(0.25, 0, 0.25, 0.25, 0.25)$, $(0.25, 0.25, 0, 0.25, 0.25)$, $(0.25, 0.25, 0.25, 0, 0.25)$, and $(0.25, 0.25, 0.25, 0.25, 0)$. That is, H should be a multiple of 4 (see Eq. (1) where $w_i = 0.25$ can be generated only when H is a multiple of 4).

The normalized m -objective minus-DTLZ1 has m extreme solutions: $(0, \frac{1}{m-1}, \dots, \frac{1}{m-1})$, $(\frac{1}{m-1}, 0, \dots, \frac{1}{m-1})$, ..., $(\frac{1}{m-1}, \frac{1}{m-1}, \dots, 0)$. From Eq. (1), we can see that w_i can be $w_i = \frac{1}{m-1}$ only when H is a multiple of $(m-1)$. To obtain the extreme solutions, we need to choose a multiple of $(m-1)$ for the m -objective minus-DTLZ1. Since inverted triangular Pareto fronts have these extreme solutions in the normalized objective space, H should be a multiple of $(m-1)$ for MOPs with inverted triangular Pareto fronts (e.g., minus-DTLZ1-4, minus-WFG4-9, distance minimization problems [12]).

4.2 Specification of H

From the above discussion, an appropriate specification of H for an m -objective problem with an inverted triangular Pareto front is $(m-1) \times j$ where j is a positive integer. This specification can be also used for MOPs with triangular Pareto fronts since we can use any specification for those problems. Notice that if $j=1$, that is, $H=m-1 < m$, all the weight vectors are on the boundary. In this case, the two-layered approach mentioned in Section 2 may be needed.

Table 2 shows examples of H which satisfies the above-mentioned condition. For 3-objective and 5-objective problems, similar values of H to the frequently-used specifications in Table 1 are selected. No change is needed for 3-objective problems in Table 1. For 5-objective problems, the suggested H value is 4 or 8 in Table 2 instead of the frequently-used value 6 in Table 1. For problems with eight or more objectives, the suggested H values in Table 2 may be impractical since the population size is too large (especially for 10-objective and 15-objective problems).

Table 2: Recommend value of H derived from our analysis and the corresponding number of weight vectors.

Number of objectives m	Divisions H	Number of weight vectors N
3	12	91
5	4/8	70/495
8	7	3,432
10	9	48,620
15	14	40,116,600

4.3 Proposed Weight Vector Specification

Whereas it is important to find all extreme solutions, we cannot use the recommended specification of H in Table 2 for many-objective problems since the population size becomes too large. Instead of using the recommended specification, we propose an alternative method for weight vector specification.

From the discussions in Section 4.1, we can see that the weight vectors corresponding to the m extreme solutions of the m -objective minus-DTLZ1 and other MOPs with inverted triangular Pareto fronts are $(0, \frac{1}{m-1}, \dots, \frac{1}{m-1})$, $(\frac{1}{m-1}, 0, \frac{1}{m-1}, \dots, \frac{1}{m-1})$, ..., $(\frac{1}{m-1}, \dots, \frac{1}{m-1}, 0)$. Our proposal is to add these weight vectors to the set of weight vectors in Table 1. Table 3 shows the value of H and the number of the weight vectors in the proposed idea for each problem.

Table 3: Proposed weight vector specification.

Number of objectives m	Divisions H	Number of weight vectors N
3	12	91
5	6	210 + 5
8	(3, 2)	(120 + 36) + 8
10	(3, 2)	(220 + 55) + 10
15	(2, 1)	(120 + 15) + 15

4.4 Experimental Results

In this subsection, we visually compare MOEA/D with the frequently-used weight vector setting in Table 1 and with the proposed weight vector setting in Table 3. MOEA/D with each specification of weight vectors is applied to the minus-DTLZ1-4 problems with 5, 8, 10, 15 objectives. We use the same parameter settings as in Section 3. In order to examine the performance of MOEA/D after removing possible negative effects of an incorrect reference point specification, we use the following setting in this subsection: $\theta = 0$ in the first 100 generations and $\theta = 5$ in the subsequent generations.

Figure 12 shows obtained solutions by a single run with the median hypervolume value. We can see from these figures that all extreme solutions are obtained for the 10-objective minus-DTLZ2 when the ten weight vectors are added. Similar results to Figure 12 are obtained for other problems.

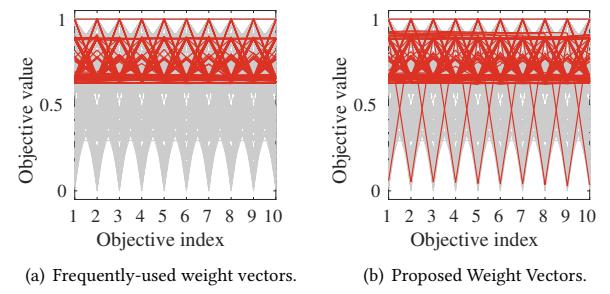


Figure 12: Solutions obtained by MOEA/D with different weight vectors given in Table 3 on the 10-objective minus-DTLZ2.

From careful comparison between the two results in Figure 12, we can observe that more solutions are obtained in Figure 12 (b)

Table 4: Average hypervolume values over 31 runs. Four solution sets are compared: (i) Solution sets obtained by MOEA/D with the added m weight vectors, (ii) Solution sets after removing the m extreme solutions from (i), (iii) Solution sets after randomly removing m non-extreme solutions, (iv) Solution sets obtained by MOEA/D without the added m weight vectors.

Problem	M	(i) With m weight vectors	(ii) Removal of extreme solutions	(iii) Random removal	(iv) Without m weight vectors
minus-DTLZ1	5	2.3776e-2 +	2.3644e-2 +	2.3610e-2 +	2.3204e-2
	8	9.6164e-4 +	9.5972e-4 +	9.3977e-4 +	7.1221e-4
	10	6.2391e-5 +	6.0905e-5 +	6.0251e-5 +	3.8799e-5
	15	2.4447e-6 +	2.4335e-6 +	2.1682e-6 +	0.6877e-6
minus-DTLZ2	5	1.4012e-1 +	1.3963e-1 +	1.3955e-1 +	1.3695e-1
	8	1.4351e-2 +	1.4333e-2 +	1.4219e-2 +	1.2664e-2
	10	1.9305e-3 +	1.9156e-3 +	1.9180e-3 +	1.7367e-3
	15	7.6218e-5 +	7.4324e-5 +	7.3279e-5 +	4.9121e-5
minus-DTLZ3	5	1.3943e-1 +	1.3896e-1 +	1.3880e-1 +	1.3695e-1
	8	1.4315e-2 +	1.4297e-2 +	1.4220e-2 +	1.2566e-2
	10	1.9437e-3 +	1.9274e-3 +	1.9275e-3 +	1.7220e-3
	15	7.2736e-5 +	7.0251e-5 +	7.0373e-5 +	4.7934e-5
minus-DTLZ4	5	1.2081e-1 +	1.2034e-1 +	1.2017e-1 ≈	1.2011e-1
	8	1.2111e-2 +	1.2103e-2 +	1.1973e-2 +	1.0481e-2
	10	1.7667e-3 +	1.7605e-3 +	1.7434e-3 +	1.5539e-3
	15	7.0126e-5 +	7.0023e-5 +	6.7236e-5 +	3.0773e-5
+/-≈		16/0/0	16/0/0	15/0/1	

than in Figure 12 (a) even if we ignore the ten extreme solutions. The same observation can be obtained from experimental results on minus-DTLZ3-4. These observations suggest that the addition of the weight vectors corresponding to the extreme solutions helps the search for other solutions (i.e., helps the diversity maintenance).

In order to further examine this issue (i.e., the effect of the added weight vectors on the search for the other solutions), we calculate the average hypervolume value over 31 runs of each algorithm on each problem. The reference point $r = (r, r, \dots, r)$ for hypervolume calculation is specified for each problem as follows: $r = 7/6$ (5-objective problems), $r = 4/3$ (8-objective problems), $r = 4/3$ (10-objective problems), $r = 3/2$ (15-objective problems). These specifications are based on Ishibuchi et al. [13]. The average hypervolume values are calculated for the following three solution sets:

(i) The solution set obtained by each run of MOEA/D with the added m weight vectors.

(ii) A subset of (i) where the m extreme solutions in (i) are removed. Our intention is to evaluate the performance of the obtained solution set for the original weight vectors.

(iii) A subset of (i) where m non-extreme solutions in (i) are randomly selected and removed. Our intention is to evaluate the performance of the obtained solution set without increasing the number of solutions.

We also examine the performance of MOEA/D with the original weight vectors in Table 1. That is, the average hypervolume value is also calculated for the following solution set:

(iv) The solution set obtained by each run of MOEA/D without the added m weight vectors.

Experimental results on minus-DTLZ1-4 with 5, 8, 10, and 15 objectives are summarized in Table 4. The worst result in each row

is shaded. The last row shows the statistical comparison results according to the Wilcoxon signed-rank test with 5% significance level between (iv) and one of (i)-(iii). Here, "+", "-" and "≈" mean that the compared solution sets are statistically better than, worse than, and have no statistically significant difference from the solution sets in (iv). Table 4 shows that the addition of the m weight vectors corresponding to the m extreme solutions clearly improves the obtained solution sets. This improvement is statistically significant for almost all problems even when the obtained extreme solutions are removed. That is, the added weight vectors help MOEA/D to search for not only the extreme solutions but also solutions corresponding to the other weight vectors.

5 EFFECTS ON OTHER ALGORITHMS

In this section, we demonstrate that the search for extreme solutions are not easy for the following algorithms when the Pareto front is inverted triangular:

- A recently proposed decomposition-based algorithm: NSGA-III.
- A decomposition-based algorithm with a weight vector adaptation method: RVEA.
- A indicator-based algorithm using a pre-defined weight vector set: MOMBI-II.

It should be noted that these algorithm use a pre-defined set of weight vectors.

We perform computational experiments using the above-mentioned algorithms. All algorithms coupled with the frequently-used weight vectors and the proposed weight vectors are run 31 times independently on minus-DTLZ1-4 with 10 objectives¹. Obtained

¹One exception is RVEA. We only show the results on problems with 5 objectives. This is because 8-objective and 10-objective minus problems are too difficult for the original RVEA.

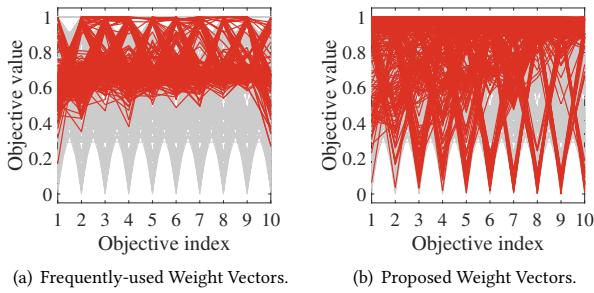


Figure 13: Solutions obtained by NSGA-III with different weight vectors on the 10-objective minus-DTLZ2.

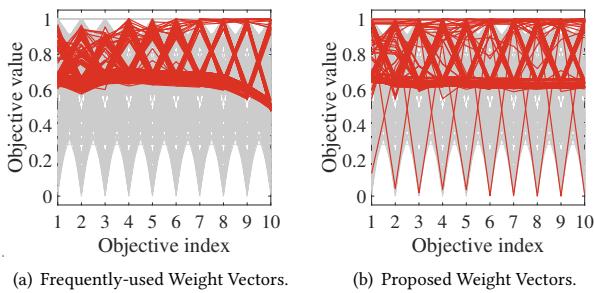


Figure 14: Solutions obtained by MOMBII-II with different weight vectors on the 10-objective minus-DTLZ2.

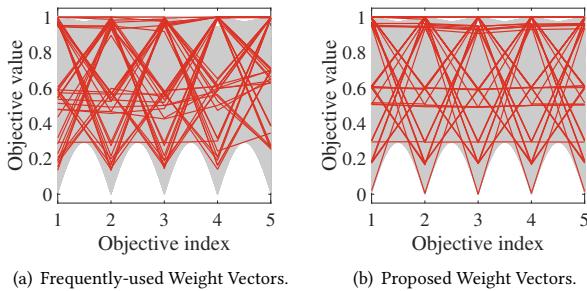


Figure 15: Solutions obtained by RVEA with different weight vectors on the 5-objective minus-DTLZ2.

solutions with the median HV among 31 runs are plotted. Due to the page limit, results only for minus-DTLZ2 are shown here (see Figures 13–15). Obtained solution sets for other problems are similar to those of minus-DTLZ2. We can see that NSGA-III with the proposed weight vector set can obtain good results in terms of both coverage (i.e., finding the extreme solutions) and diversity (i.e., finding many other solutions). MOMBII-II with the proposed weight vector set can obtain good results with respect to the coverage. Good results are also obtained by RVEA with the proposed weight vector set.

These results show that the proposed method helps the three algorithms to search for the extreme solutions of the inverted triangular Pareto fronts. This is because no weight vector in those algorithms corresponds to the extreme solutions as we have already explained in Section 4. Thus, the proposed method will improve the performance of other weight (reference) vector-based algorithms.

6 CONCLUSION

In this paper, we examined MOEA/D with different specifications of H for problems with triangular and inverted triangular Pareto fronts. When an MOP has a triangular Pareto front as DTLZ1–4 and WFG4–9, the value of H does not have a large effect on the ability of MOEA/D to search for the extreme solutions. However, the search ability of MOEA/D on MOPs with inverted triangular Pareto fronts strongly depends on the value of H . Through computational experiments, we obtained the following observations for MOPs with inverted triangular Pareto fronts:

- MOEA/D with the frequently-used specifications of H could not find the extreme solutions. That is, the obtained solution sets did not cover the entire Pareto fronts.
- The coverage of Pareto fronts by the obtained solutions was not always improved by simply increasing the value of H .
- Good results were obtained by carefully specifying the value of H , which depends on the number of objectives.

Based on these observations, we derived the following specification rule to search for the extreme solutions of MOPs with inverted triangular Pareto fronts: H should be a multiple of $(m - 1)$ where m is the number of objectives. The validity of this rule was confirmed through computational experiments and theoretical analysis. However, the specification based on this rule leads to an intractably large number of weight vectors for many-objective problems. As a simple remedy, we proposed an alternative way: to include the m weight vectors corresponding to the m extreme solutions of an m -objective problem. Through computational experiments, we demonstrated that our idea significantly improved the performance of MOEA/D on many-objective problems with inverted triangular Pareto fronts. We also demonstrated that our idea improved the performance of other EMO algorithms with fixed weight vectors (i.e., NSGA-III, MOMBII-II, and RVEA) with respect to the coverage of the Pareto fronts by the obtained solutions.

One future research direction is to examine the effect of the proposed idea for other decomposition-based EMO algorithms and other test problems with irregular Pareto fronts. It is also an interesting future research topic to consider test problems with triangular Pareto fronts that are rotated by a certain angle.

ACKNOWLEDGMENTS

This work was supported by National Natural Science Foundation of China (Grant No. 61876075), Guangdong Provincial Key Laboratory (Grant No. 2020B121201001), the Program for Guangdong Introducing Innovative and Entrepreneurial Teams (Grant No. 2017ZT07X386), Shenzhen Science and Technology Program (Grant No. KQTD2016112514355531), the Program for University Key Laboratory of Guangdong Province (Grant No. 2017KSYS008).

REFERENCES

- [1] Kalyan Shankar Bhattacharjee, Hemant Kumar Singh, Tapabrata Ray, and Qingfu Zhang. 2017. Decomposition based evolutionary algorithm with a dual set of reference vectors. In *Proceedings of the IEEE Congress on Evolutionary Computation*. San Sebastian, Spain, 105–112.
- [2] Ran Cheng, Yaochu Jin, Markus Olhofer, and Bernhard Sendhoff. 2016. A reference vector guided evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation* 20, 5 (Oct. 2016), 773–791.

- [3] Indranee Das and John E Dennis. 1998. Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. *SIAM Journal on Optimization* 8, 3 (July 1998), 631–657.
- [4] Kalyanmoy Deb and Himanshu Jain. 2014. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints. *IEEE Transactions on Evolutionary Computation* 18, 4 (Aug. 2014), 577–601.
- [5] Kalyanmoy Deb, Lothar Thiele, Marco Laumanns, and Eckart Zitzler. 2002. Scalable multi-objective optimization test problems. In *Proceedings of the IEEE Congress on Evolutionary Computation*, Vol. 1. Honolulu, USA, 825–830.
- [6] Xiaoyu He, Yuren Zhou, Zefeng Chen, and Qingfu Zhang. 2018. Evolutionary Many-objective Optimization based on Dynamical Decomposition. *IEEE Transactions on Evolutionary Computation* 23, 3 (June 2018), 361–375.
- [7] Raquel Hernández Gómez and Carlos A Coello Coello. 2015. Improved metaheuristic based on the R2 indicator for many-objective optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference*. Madrid, Spain, 679–686.
- [8] Simon Huband, Philip Hingston, Luiggi Barone, and Lyndon While. 2006. A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation* 10, 5 (Oct. 2006), 477–506.
- [9] Hisao Ishibuchi, Ken Doi, Hiroyuki Masuda, and Yusuke Nojima. 2015. Relation between weight vectors and solutions in MOEA/D. In *IEEE Symposium Series on Computational Intelligence*. Cape Town, South Africa, 861–868.
- [10] Hisao Ishibuchi, Ken Doi, and Yusuke Nojima. 2016. Use of piecewise linear and nonlinear scalarizing functions in MOEA/D. In *Proceedings of the International Conference on Parallel Problem Solving from Nature*. Edinburgh, Scotland, UK, 503–513.
- [11] Hisao Ishibuchi, Ken Doi, and Yusuke Nojima. 2017. On the effect of normalization in MOEA/D for multi-objective and many-objective optimization. *Complex & Intelligent Systems* 3, 4 (Dec. 2017), 279–294.
- [12] H. Ishibuchi, L. He, and K. Shang. 2019. Regular Pareto Front Shape is not Realistic. In *Proceedings of the IEEE Congress on Evolutionary Computation*. Wellington, New Zealand, 2034–2041.
- [13] Hisao Ishibuchi, Ryo Imada, Yu Setoguchi, and Yusuke Nojima. 2018. How to Specify a Reference Point in Hypervolume Calculation for Fair Performance Comparison. *Evolutionary Computation* 26, 3 (Sept. 2018), 411–440.
- [14] Hisao Ishibuchi, Yu Setoguchi, Hiroyuki Masuda, and Yusuke Nojima. 2017. Performance of decomposition-based many-objective algorithms strongly depends on Pareto front shapes. *IEEE Transactions on Evolutionary Computation* 21, 2 (April 2017), 169–190.
- [15] Himanshu Jain and Kalyanmoy Deb. 2014. An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point Based Nondominated Sorting Approach, Part II: Handling Constraints and Extending to an Adaptive Approach. *IEEE Transactions on Evolutionary Computation* 18, 4 (Aug. 2014), 602–622.
- [16] Ke Li, Kalyanmoy Deb, Qingfu Zhang, and Sam Kwong. 2014. An evolutionary many-objective optimization algorithm based on dominance and decomposition. *IEEE Transactions on Evolutionary Computation* 19, 5 (Oct. 2014), 694–716.
- [17] Miqing Li and Xin Yao. 2017. What Weights Work for You? Adapting Weights for Any Pareto Front Shape in Decomposition-based Evolutionary Multi-Objective Optimisation. *CoRR* abs/1709.02679 (2017). arXiv:1709.02679
- [18] Ye Tian, Ran Cheng, Xingyi Zhang, Fan Cheng, and Yaochu Jin. 2018. An Indicator-Based Multiobjective Evolutionary Algorithm With Reference Point Adaptation for Better Versatility. *IEEE Transactions on Evolutionary Computation* 22, 4 (Aug. 2018), 609–622.
- [19] Ye Tian, Ran Cheng, Xingyi Zhang, and Yaochu Jin. 2017. PlatEMO: A MATLAB platform for evolutionary multi-objective optimization. *IEEE Computational Intelligence Magazine* 12, 4 (Nov. 2017), 73–87.
- [20] Ye Tian, Ran Cheng, Xingyi Zhang, Yansen Su, and Yaochu Jin. 2019. A strengthened dominance relation considering convergence and diversity for evolutionary many-objective optimization. *IEEE Transactions on Evolutionary Computation* 23, 2 (April 2019), 331–345.
- [21] Yuan Yuan, Hua Xu, Bo Wang, and Xin Yao. 2016. A new dominance relation-based evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation* 20, 1 (Feb. 2016), 16–37.
- [22] Qingfu Zhang and Hui Li. 2007. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation* 11, 6 (Dec. 2007), 712–731.