## ORIGINAL ARTICLE



# On the effect of normalization in MOEA/D for multi-objective and many-objective optimization

Received: 3 August 2017 / Accepted: 5 October 2017 / Published online: 19 October 2017 © The Author(s) 2017. This article is an open access publication

**Abstract** The frequently used basic version of MOEA/D (multi-objective evolutionary algorithm based on decomposition) has no normalization mechanism of the objective space, whereas the normalization was discussed in the original MOEA/D paper. As a result, MOEA/D shows difficulties in finding a set of uniformly distributed solutions over the entire Pareto front when each objective has a totally different range of objective values. Recent variants of MOEA/D have normalization mechanisms for handling such a scaling issue. In this paper, we examine the effect of the normalization of the objective space on the performance of MOEA/D through computational experiments. A simple normalization mechanism is used to examine the performance of MOEA/D with and without normalization. These two types of MOEA/D are also compared with recently proposed many-objective algorithms: NSGA-III, MOEA/DD, and  $\theta$ -DEA. In addition to the frequently used many-objective test problems DTLZ and WFG, we use their minus versions. We also propose two variants of the DTLZ test problems for examining the effect of the normalization in MOEA/D. Test problems in one variant have objective functions with totally different ranges. The other variant has a kind of deceptive nature,

where the range of each objective is the same on the Pareto front but totally different over the entire feasible region. Computational experiments on those test problems clearly show the necessity of the normalization. It is also shown that the normalization has both positive and negative effects on the performance of MOEA/D. These observations suggest that the influence of the normalization is strongly problem dependent.

**Keywords** Evolutionary multi-objective optimization (EMO) · Many-objective optimization · Objective space normalization · MOEA/D · Decomposition-based algorithms

#### Introduction

Recently, many-objective optimization has received a lot of attention in the evolutionary multi-objective optimization (EMO) community, where optimization of four or more objectives is called many-objective optimization [14, 15, 23]. Many-objective problems present a number of challenges [10,19] to the EMO community such as the deterioration in search ability of Pareto dominance-based algorithms [6,29] and the increase in computation time of hypervolume-based algorithms [2,3]. For many-objective problems, it has been demonstrated in the literature [10,12] that MOEA/D [27] works well in comparison with Pareto dominance-based and hypervolume-based algorithms in terms of their search ability and computation time. As a result, a number of EMO algorithms have been proposed for many-objective problems based on the same or similar framework as MOEA/D (e.g., NSGA-III [5], MOEA/DD [16], I-DBEA [1], and  $\theta$ -DEA

Various approaches to the improvement of MOEA/D have also been proposed in the literature (for detail, see a review

> Ken Doi ken.doi@ci.cs.osakafu-u.ac.jp

Yusuke Nojima nojima@cs.osakafu-u.ac.jp

- Shenzhen Key Laboratory of Computational Intelligence, Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, China
- Department of Computer Science and Intelligent Systems, Graduate School of Engineering, Osaka Prefecture University, Sakai, Japan



[22]). One important research issue is the specification of a scalarizing function [11,21,24]. The specification of weight vectors [8] and their adjustments [20] is also important. Other important issues include solution—subproblem matching [17, 18] and resource allocation to subproblems [28]. Whereas the normalization of the objective space is also important, this issue has not been actively studied in the literature [4]. One exception is Bhattacharjee et al. [4], where a six-sigma-based method was proposed for removing the influence of dominance-resistant solutions on nadir point calculation.

In this paper, we examine the effect of the normalization on the performance of MOEA/D. The frequently used basic version of MOEA/D [27] has no normalization mechanism, whereas the normalization was discussed in the original MOEA/D paper [27]. As a result, MOEA/D is often outperformed by other EMO algorithms with normalization mechanisms in their applications to test problems, where each objective has a totally different range of objective values. For example, in the WFG4-9 test problems [9], the range of the Pareto front on the *i*th objective is [0, 2i]. This means that the tenth objective  $f_{10}$  has a ten times wider range than the first objective  $f_1$ . It is difficult for MOEA/D without normalization to find a set of uniformly distributed solutions over the entire Pareto front for such a many-objective test problem. In this paper, we combine a simple normalization mechanism to MOEA/D in order to compare the performance between MOEA/D with and without normalization.

This paper is organized as follows. First, we briefly explain MOEA/D and a simple normalization mechanism. Next, we demonstrate that the combined normalization mechanism has positive and negative effects on the performance of MOEA/D through computational experiments on the DTLZ1-4 [7] and WFG4-9 [9] test problems with three to ten objectives. We also discuss why the normalization deteriorates the performance of MOEA/D on some test problems. Then, we create two variants of the DTLZ1-4 test problems to further examine the effect of the normalization in MOEA/D. Each test problem in one variant has objective functions with totally different ranges. More specifically, the range of the Pareto front on the *i*th objective is  $[0, \alpha^{i-1}]$ , where  $\alpha$  is a nonnegative parameter (e.g., 2 and 10). The other variant has a kind of deceptive nature. Whereas the range of each objective is totally different over the entire feasible region, the Pareto front has the same range for each objective. In other words, each objective has totally different values in early generations and similar values after enough generations. Experimental results on the new test problems clearly show the necessity of the normalization as well as its negative effects. Finally, we report experimental results on minus versions [13] of the DTLZ1-4 and WFG4-9 test problems. The best results on almost all minus test problems are obtained by MOEA/D with normalization among MOEA/D, NSGA-III, MOEA/DD, and  $\theta$ -DEA.



#### MOEA/D and normalization

We explain MOEA/D and the normalization of the objective space for the following m-objective minimization problem:

Minimize 
$$f(x) = (f_1(x), f_2(x), \dots, f_m(x))$$
 subject to  $x \in X$ ,

where  $f_i(x)$  is the *i*th objective to be minimized (i = 1, 2, ..., m), x is a decision vector, and X is a feasible region of x in the decision space.

In MOEA/D, a multi-objective problem is handled as a set of single-objective problems. Each single-objective problem is defined by a scalarizing function with a different weight vector  $\mathbf{w} = (w_1, w_2, \dots, w_m)$ . A set of uniformly distributed weight vectors is generated based on the following relations [27]:

$$\sum_{i=1}^{m} w_i = 1 \text{ and } 0 \le w_i \le 1 \text{ for } i = 1, 2, ..., m$$

$$w_i \in \left\{0, \frac{1}{H}, \frac{2}{H}, ..., \frac{H}{H}\right\} \text{ for } i = 1, 2, ..., m,$$

where H is a positive integer for determining the resolutions of the generated weight vectors. MOEA/D generates all weight vectors satisfying these relations. The population size is the same as the total number of the generated weight vectors. This is because each single-objective problem with a different weight vector has a single solution.

The basic idea of MOEA/D is to search for the best solution along each weight vector, as shown in Fig. 1. The starting point of all weight vectors, which is called a reference point, is specified by the best value of each objective. When the best value of each objective is unknown, the best value among all the examined solutions is used for each objective.

A scalarizing function is used in MOEA/D for combining multiple objectives into a single objective. The original paper on MOEA/D by Zhang and Li [27] examined the weighted sum, Tchebycheff and PBI (penalty-based boundary intersection) functions as a scalarizing function. In this paper, we use the PBI function, since it has been frequently used in the literature. The idea of the PBI function has also been

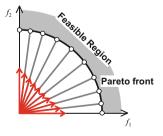


Fig. 1 Basic idea of MOEA/D. The best solution for each weight vector is searched for approximating the entire Pareto front

used in recently proposed many-objective algorithms such as I-DBEA [1] and  $\theta$ -DEA [26]. The PBI function is defined as

Minimize 
$$f^{PBI}(\mathbf{x}|\mathbf{w}, \mathbf{z}^*) = d_1 + \theta d_2$$
,

where  $\theta$  is a penalty parameter, and  $d_1$  and  $d_2$  are defined using the reference point  $z^*$  (which is the starting point of the weight vectors in Figs. 1 and 2):

$$d_1 = \left| (f(x) - z^*)^T w \right| / \|w\|$$

$$d_2 = \left\| f(x) - z^* - d_1 \frac{w}{||w||} \right\|.$$

These two distances  $d_1$  and  $d_2$  are explained in Fig. 2a. The idea behind the use of the PBI function is to search for a solution on the Pareto front along each weight vector, as shown in Fig. 1, by minimizing both  $d_1$  and  $d_2$ . The value of  $\theta$  specifies the importance of each distance. Its effect is explained in Fig. 2b, c using a contour line for the current solution (open circle). When  $\theta$  is large, the search along the weight vector is emphasized, as shown in Fig. 2b. When  $\theta$  is small, the search toward the Pareto front is emphasized, as shown in Fig. 2c. In this paper,  $\theta = 5.0$  is used as in the original MOEA/D paper.

Whereas MOEA/D without normalization is usually used in the literature, its necessity is clear. Figure 3a illustrates the uniformly distributed weight vectors and the obtained solutions. Whereas the weight vectors are uniformly distributed, the obtained solutions are biased towards the region with small  $f_1$  values. This is because the range of each objective on the Pareto front is different. Figure 3b shows the normalization of Fig. 3a for clear illustration of the obtained solutions. If MOEA/D has a normalization mechanism, the search is performed in the normalized objective space using the uniformly distributed weight vectors, as shown in Fig. 4b. In this case, we can obtain the uniformly distributed solutions. The obtained solutions in Fig. 4b are also uniformly distributed in the original objective space, as shown in Fig. 4a.

In this paper, we use a simple normalization mechanism based on non-dominated solutions in the current population. Let  $z_{iL}$  and  $z_{iU}$  be the minimum and maximum values of the

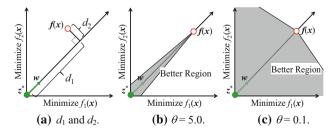


Fig. 2 Explanations of the PBI function

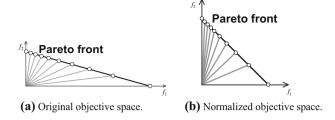


Fig. 3 Explanations of the obtained solutions without normalization. The search by MOEA/D is performed in the original space in a

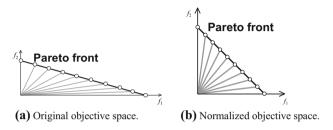


Fig. 4 Explanations of the obtained solutions with normalization. The search by MOEA/D is performed in the normalized space in b

*i*th objective  $f_i(x)$  among non-dominated solutions in the current population (i = 1, 2, ..., m), respectively. In this setting,  $z_L = (z_{1L}, ..., z_{mL})$  and  $z_U = (z_{1U}, ..., z_{mU})$  can be viewed as estimated ideal and nadir points from the non-dominated solutions in the current population, respectively. Using  $z_{iL}$  and  $z_{iU}$ , the objective value  $z_i$  of the *i*th objective  $f_i(x)$  is normalized as

$$z_i := \frac{z_i - z_{iL}}{z_{iU} - z_{iL} + \varepsilon}$$
 for  $i = 1, 2, ..., m$ ,

where  $\varepsilon$  is a positive value for preventing the denominator from becoming zero in the case of  $z_{iL}=z_{iU}$ . We examine two specifications of  $\varepsilon$ :  $10^{-6}$  and 1. By this normalization, all non-dominated solutions are normalized to points in  $[0, 1]^m$  in the normalized objective space. In the original MOEA/D paper [27], the use of the above-mentioned normalization mechanism with  $\varepsilon=0$  was examined for the two-objective ZDT3 problem and a modified version of the two-objective ZDT1 problem.

## **Experimental results on DTLZ and WFG**

Let us examine the performance of MOEA/D with and without the simple normalization mechanism through computation experiments on frequently used many-objective test problems in the literature: DTLZ1-4 [7] and WFG4-9 [9] with 3, 5, 8, and 10 objectives. The range of each objective on the Pareto front of each test problem is as follows: [0, 0.5] in DTLZ1, [0, 1] in DTLZ2-4, and [0, 2i] for i = 1, 2, ..., m



in WFG4-9. Thus, the normalization of the objective space is needed in WFG4-9, whereas it is not needed in DTLZ1-4.

For comparison, we also apply NSGA-III [5], MOEA/DD [16], and  $\theta$ -DEA [26] to DTLZ1-4 and WFG4-9. As NSGA-III, we use Yuan's implementation [25]. All of these three EMO algorithms and MOEA/D use the same set of uniformly distributed weight vectors. Their basic search strategies are also the same. They search for the best solution for each weight vector. Similar solution sets are obtained when they are applied to a simple test problem with a normalized triangular Pareto front such as the three-objective DTLZ1. Their differences can be summarized as follows:

- 1. Normalization: NSGA-III and  $\theta$ -DEA have a sophisticated normalization mechanism while MOEA/DD and MOEA/D in this paper do not have a normalization mechanism.
- 2. Generation update: NSGA-III and  $\theta$ -DEA use a  $(\mu + \mu)$  ES model, while MOEA/D and MOEA/DD use a  $(\mu + 1)$  ES model.
- Assignment of a new solution to weight vectors: a new solution is assigned to the single nearest weight vector in NSGA-III, MOEA/DD and θ-DEA, while a new solution is compared to all solutions in its neighborhood in MOEA/D.
- 4. Replacement of solutions: only a single solution can be replaced with a new solution in NSGA-III, MOEA/DD, and  $\theta$ -DEA, while multiple solutions in the neighborhood can be replaced with a new solution in MOEA/D.
- 5. Handling of weight vectors outside the Pareto front: no new solutions are assigned to those weight vectors in NSGA-III, MOEA/DD, and θ-DEA (i.e., some weight vectors may have no solutions and some others may have multiple solutions), while each weight vector always has a single solution and continues to search for its best solution in MOEA/D.
- 6. Basic mechanisms of fitness evaluation: the PBI function (MOEA/D), Pareto dominance, the number of assigned solutions to each weight vector and the distance to the closest weight vector (NSGA-III), and Pareto dominance, the number of assigned solutions to each weight vector and PBI (MOEA/DD and  $\theta$ -DEA).

Our computational experiments are performed in the same manner as in Yuan et al. [11]. The neighborhood size is 20. The polynomial mutation with the distribution index 20 is used with the mutation probability 1/n, where n is the number of decision variables. The simulated binary crossover with the distribution index 20 is used with the crossover probability 1.0. The number of weight vectors is specified as 91 (three objectives), 210 (five objectives), 156 (eight objectives), and 275 (ten objectives).



The performance of each approach is evaluated by calculating the average hypervolume value over 51 runs. Before calculating the hypervolume, the objective space is normalized using the true Pareto front of each test problem, so that the ideal and nadir points are  $(0,0,\ldots,0)$  and  $(1,1,\ldots,1)$ . This information is used only for the hypervolume calculation after the execution of each algorithm. The reference point for the hypervolume calculation is specified as  $(1.1,1.1,\ldots,1.1)$  in the normalized objective space.

Experimental results are shown in Table 1. For easy reading of the experimental results, the average hypervolume value of each algorithm on each test problem is normalized using the best result in each row, as shown in Table 1. Thus, the best value for each test problem is always 1.0000, as shown in Table 1. The best and worst results are highlighted by the bold font and the underline, respectively.

The performance of MOEA/DD in Table 1 clearly shows the necessity of normalization. Whereas the best results are obtained from MOEA/DD for most test problems in DTLZ1-4, NSGA-III, and  $\theta$ -DEA outperform MOEA/DD in their applications to WFG4-9. This is because MOEA/DD has no normalization mechanism, whereas the search in NSGA-III and  $\theta$ -DEA is performed in the normalized objective space. However, when they were applied to the normalized WFG4-9 in [13], the best results were obtained from MOEA/DD.

MOEA/D with no normalization mechanism also shows clear performance deterioration for WFG4-9, as shown in Table 1. By combining the simple normalization mechanism with  $\varepsilon=10^{-6}$  into MOEA/D, its performance is improved for the three-objective and five-objective WFG4-9 test problems, as shown in Table 1. However, the simple normalization mechanism with  $\varepsilon=10^{-6}$  deteriorates the performance of MOEA/D for almost all of the other test problems. Especially, negative effects of the simple normalization mechanism on the performance of MOEA/D are clearly observed for DTLZ1-4, as shown in Table 1.

By increasing the value of  $\varepsilon$  from  $\varepsilon=10^{-6}$  to  $\varepsilon=1$  in Table 1, the severe negative effects of the simple normalization mechanism with  $\varepsilon=10^{-6}$  are remedied for most test problems except for DTLZ3 and DTLZ4. In the next section, we discuss why the performance of MOEA/D on some test problems is severely degraded by the simple normalization mechanism.

## **Negative effect of normalization**

Let us examine the search behavior of MOEA/D with the simple normalization mechanism ( $\varepsilon = 10^{-6}$ ) on the ten-objective DTLZ1 and WFG9 test problems, where the negative effects of the normalization are severe, as shown

Table 1 Normalized average hypervolume values

Problem	Objectives	MOEA/D	Normalized MOEA/D $(\varepsilon = 10^{-6})$	Normalized MOEA/D $(\varepsilon = 1)$	MOEA/DD	NSGA-III	θ-DEA
DTLZ1	3	0.9976	0.9963	0.9976	1.0000	0.9954	0.9978
	5	0.9999	0.9992	0.9997	1.0000	0.9996	0.9998
	8	0.9993	0.3827	0.9985	0.9998	0.9999	1.0000
	10	0.9999	0.3829	0.9998	0.9999	1.0000	1.0000
DTLZ2	3	0.9996	0.9994	0.9995	1.0000	0.9985	0.9993
	5	0.9997	0.9996	0.9996	1.0000	0.9965	0.9993
	8	0.9998	0.9383	<u>0.9271</u>	1.0000	0.9953	0.9995
	10	1.0000	0.9767	0.8650	1.0000	0.9975	0.9996
DTLZ3	3	0.9961	0.9902	0.9624	1.0000	0.9955	0.9967
	5	0.9984	0.9882	0.5262	1.0000	0.9955	0.9988
	8	0.9006	0.7676	0.4812	1.0000	0.9907	0.9971
	10	0.9757	0.8241	0.6730	1.0000	0.9972	0.9994
DTLZ4	3	0.7425	0.4122	0.3442	1.0000	0.9339	0.9422
	5	0.9299	0.5606	0.4284	1.0000	0.9997	1.0000
	8	0.9585	0.3519	0.3273	1.0000	0.9997	1.0000
	10	0.9912	0.5542	0.3577	1.0000	0.9999	1.0000
WFG4	3	0.9393	0.9469	0.9463	0.9875	0.9987	1.0000
	5	0.8946	0.9854	0.9850	0.9790	0.9984	1.0000
	8	0.5974	0.9277	0.9922	0.9346	0.9992	1.0000
	10	0.5809	1.0000	0.9985	0.8926	0.9973	0.9977
WFG5	3	0.9547	0.9656	0.9631	0.9851	0.9995	1.0000
	5	0.9060	0.9957	0.9894	0.9727	0.9994	1.0000
	8	0.6812	0.5417	0.9966	0.9272	0.9998	1.0000
	10	0.6442	0.5739	1.0000	0.8892	0.9994	0.9994
WFG6	3	0.9524	0.9597	0.9609	0.9881	0.9993	1.0000
	5	0.8360	0.9946	0.9943	0.9770	0.9992	1.0000
	8	0.3859	0.2252	0.9973	0.9104	0.9985	1.0000
	10	0.3512	0.3274	1.0000	0.8713	0.9939	1.0000
WFG7	3	0.8472	0.8782	0.8771	0.9863	0.9976	1.0000
	5	0.8248	0.9914	0.9901	0.9721	0.9974	1.0000
	8	0.4159	<u>0.1617</u>	0.9971	0.9213	0.9989	1.0000
	10	0.3866	0.3970	1.0000	0.9072	0.9985	0.9990
WFG8	3	0.9440	0.9547	0.9597	0.9857	0.9964	1.0000
	5	0.7928	1.0000	0.9945	0.9703	0.9938	0.9944
	8	0.1684	0.1207	1.0000	0.9572	0.9958	0.9985
	10	0.1362	0.1121	1.0000	0.9620	0.9870	0.9877
WFG9	3	0.8613	0.8652	0.8717	0.9890	0.9997	1.0000
	5	0.8439	0.9617	0.9539	0.9402	0.9902	1.0000
	8	0.5225	0.2536	0.9218	0.8735	0.9866	1.0000
	10	0.4897	0.1216	0.8855	0.8234	0.9889	1.0000

The best value is normalized to 1.0000 in each row

in Table 1. In Fig. 5a, we show the obtained solution set by a single run of MOEA/D without normalization on the tenobjective DTLZ1 in the normalized objective space  $[0, 1]^{10}$ . The single run with the median average hypervolume value is selected from the 51 runs as a representative run. The corresponding result by MOEA/D with normalization is shown in Fig. 5b, where the diversity of solutions is severely degraded by normalization.

In the same manner as in Fig. 5, we show the obtained solution sets on the ten-objective WFG9, as shown in Fig. 6.



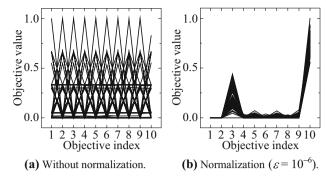


Fig. 5 Obtained solution sets by a single run of MOEA/D with and without normalization on the ten-objective DTLZ1 test problem

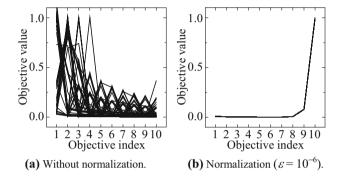


Fig. 6 Obtained solution sets by a single run of MOEA/D with and without normalization on the ten-objective WFG9 test problem

Figure 6 also shows that the diversity of solutions is severely degraded by normalization. For comparison, we show the results by MOEA/D with normalization in the case of  $\varepsilon=1$ , as shown in Fig. 7. Comparison between Fig. 6a without normalization and Fig. 7b with normalization shows clear positive effects of the normalization when  $\varepsilon$  is specified as  $\varepsilon=1$ .

From Figs. 5, 6, and 7, we can see that the reason for poor results by MOEA/D with normalization ( $\varepsilon=10^{-6}$ ) is the deterioration of solution diversity. Now, the question is why the diversity is severely degraded by the normalization. To address this issue, we monitor the width of the range of non-dominated solutions for each objective in each generation. That is, we monitor the value of ( $z_{iU}-z_{iL}$ ), which is used in the denominator of the simple normalization mechanism in each generation. This monitoring is performed for the selected runs in Figs. 5 and 6. The monitoring results in the first 100 generations are shown in Fig. 8 on DTLZ1 and Fig. 9 on WFG9.

From Fig. 8a on the ten-objective DTLZ1 problem, we can see that the value of the width  $(z_{iU} - z_{iL})$  is totally different for each objective in the first ten generations. Some objectives have very small values of the width (e.g., about 0.2) and others have large values of the width (e.g., more than 5). This is more clearly observed in Fig. 9a, where the width of some

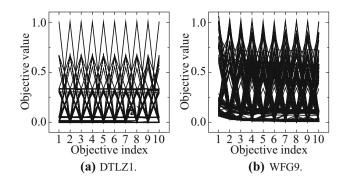
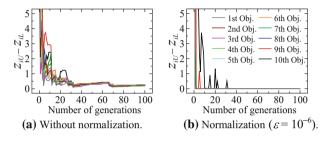
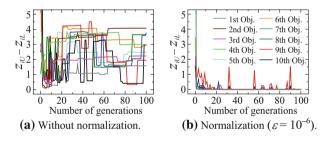


Fig. 7 Obtained solution sets by MOEA/D with normalization ( $\varepsilon = 1$ ) on the 10-objective DTLZ1 and WFG9 test problems



**Fig. 8** Width of the range of non-dominated solutions for each objective in each of the first 100 generations of the runs in Fig. 5 on the ten-objective DTLZ1 test problem. Results for the ten objectives are simultaneously depicted



**Fig. 9** Width of the range of non-dominated solutions for each objective in each of the first 100 generations of the runs in Fig. 6 on the ten-objective WFG9 test problem. Results for the ten objectives are simultaneously depicted

objectives is close to zero, whereas other objectives have much larger width (e.g., about 3). These observations suggest that the normalization based on non-dominated solutions in the current population may severely modify the scale of each objective.

Since the denominator in the simple normalization mechanism is  $(z_{iU} - z_{iL}) + \varepsilon$ , the objective space is severely rescaled by the normalization when  $\varepsilon$  is very small. As a result, the diversity of solutions is severely deteriorated in Figs. 8b and 9b within the first ten generations. When  $\varepsilon$  is not very small (e.g.,  $\varepsilon = 1$ , as shown in Fig. 7), the problem of very small values of  $(z_{iU} - z_{iL})$  is remedied.



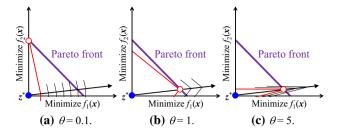


Fig. 10 Best solution for the PBI function with the weight vector (0.9, 0.1) for different specifications of  $\theta$ 

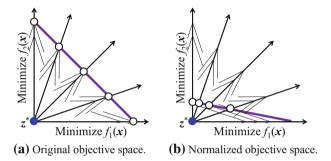


Fig. 11 Explanations of the obtained solutions by the PBI function using the contour lines for each subproblem

Let us further discuss why unnecessary normalization decreases the diversity of solutions. Figure 10 shows the relation between the best solution for the PBI function with the weight vector (0.9, 0.1) for three specifications of  $\theta$ when the Pareto front is linear. When  $\theta$  is large, the optimal solution is obtained on the search line specified by the weight vector, as shown in Fig. 10b, c. When  $\theta$  is small, the best solution is not on the search line specified by the weight vector, as shown in Fig. 10a. Thus, a large value of  $\theta$  such as  $\theta = 5$  is frequently used in the literature. However, a large value of  $\theta$  makes the better region than the current solution very small (i.e., inside the red contour line in Fig. 10, see also Fig. 2). This leads to slow convergence especially in the case of many-objective optimization [10]. Thus, an appropriate specification of  $\theta$  is very important and difficult in MOEA/D with the PBI func-

When the search of the best solution is performed in the original objective space, we can obtain a set of well-distributed solutions for the DTLZ1 test problem by the PBI function with  $\theta=5$ , as shown in Fig. 11a. However, when the search is performed in a heavily rescaled objective space, as shown in Fig. 11b, a wide variety of solutions are not likely to be obtained. For example, in Fig. 11b, the best solution for the weight vector (1,0) is on the  $f_2$  axis (not on the  $f_1$  axis). As a result, the obtained solutions may be heavily biased toward the region with small values of  $f_1$ . This may explain severe deterioration of the diversity of solutions.

Another potential difficulty of using a small value of  $\varepsilon$  is that objective values of dominated solutions may become very large when the width  $(z_{iU} - z_{iL})$  is small. For example, when  $z_{iL} = 0.10$ ,  $z_{iU} = 0.11$ , and  $z_i = 1.00$ , these values are normalized to 0.00, 1.00, and 89.99 by the formulation  $(z_i - 0.10)/(0.01 + \varepsilon)$  with  $\varepsilon = 10^{-6}$ . This difficulty is remedied by increasing the value of  $\varepsilon$ . For example,  $z_{iL} = 0.10$ ,  $z_{iU} = 0.11$  and  $z_i = 1.00$  are normalized to 0.00, 0.01, and 0.89, respectively, when  $\varepsilon = 1$ .

However, in this case (i.e., when  $\varepsilon$  is not very small), we have a different type of difficulty: the width of the Pareto front on each objective in the normalized objective space is different [i.e., the nadir point in the normalized objective space is not (1, 1, ..., 1)]. For example, let us assume that the Pareto front is a line between (0, 3) and (1, 0) in the original two-dimensional objective space. The Pareto front is in a subspace  $[0, 1] \times [0, 3]$ . This Pareto front is normalized to a line between (0, 1) and (1, 0) when  $\varepsilon$  is very small. That is, the subspace  $[0, 1] \times [0, 3]$  including the Pareto front is normalized to  $[0, 1] \times [0, 1]$ . However, the same Pareto front is normalized to a line between (0.0, 0.75) and (0.5, 0.0) when  $\varepsilon = 1$ . That is, the subspace  $[0, 1] \times [0, 3]$  is normalized to  $[0.0, 0.5] \times [0.0, 0.75]$ . As shown by this simple example, a hyper-rectangle including the Pareto front of an m-objective problem is not normalized to a unit hypercube  $[0,1]^m$  when  $\varepsilon$  cannot be negligible. This negative effect of using a large value of  $\varepsilon$  is illustrated in Fig. 12 using experimental results on the three-objective WFG4 test problem by the three variants of MOEA/D: no normalization, normalization with  $\varepsilon = 10^{-6}$  and  $\varepsilon = 1$ . When no normalization mechanism is used in Fig. 12a, many solutions are obtained around the bottom-left corner of the Pareto front. When  $\varepsilon$  is very small in Fig. 12b, well-distributed solutions are obtained. However, when  $\varepsilon$  is not very small (i.e.,  $\varepsilon = 1$ ) in Fig. 12c, we can see that more solutions are obtained around the bottom-left corner of the Pareto front than the other two corners (e.g., compare the  $3 \times 3$  solutions around each corner).

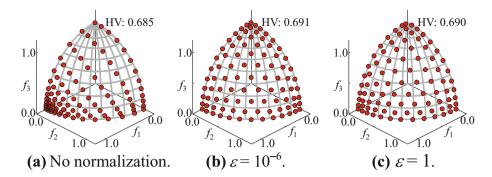
As we have explained, small and large values of  $\varepsilon$  have their own negative effects. As a result, better results are obtained from  $\varepsilon=10^{-6}$  for some test problems and  $\varepsilon=1$  for other test problems in Table 1 (and other tables in this paper).

# New many-objective test problems

We have already explained that the simple normalization mechanism has both positive and negative effects on the performance of MOEA/D. In this section, we propose new many-objective test problems to further examine the effect of the normalization. Our new test problems are generated by slightly modifying the DTLZ1-4 test problems as follows.



Fig. 12 Solutions obtained by a single run with the median hypervolume value among the 51 runs of each variants of MOEA/D on the three-objective WFG4 problem in Table 1



Each objective in the DTLZ1-4 test problems is written as

$$f_i(\mathbf{x}) = (1 + g(\mathbf{x}_M))h_i(\mathbf{x}_{pos})$$
 for  $i = 1, 2, ..., m$ .

This function can be easily modified to create new rescaled test problems as follows:

$$f_i(\mathbf{x}) = \alpha^{i-1} (1 + g(\mathbf{x}_M)) h_i(\mathbf{x}_{pos})$$
 for  $i = 1, 2, ..., m$ ,

where  $\alpha$  is a positive value for rescaling. The ith objective has an  $\alpha^{i-1}$  times larger range than the first objective. The newly created test problems from DTLZ1-4 using this objective function are referred to as the rescaled DTLZ1-4 test problems. In our computational experiments,  $\alpha$  is specified as  $\alpha=10$ .

The objective function in DTLZ1-4 can also be easily modified to create a kind of deceptive rescaled test problems as

$$f_i(\mathbf{x}) = (1 + \alpha^{i-1}g(\mathbf{x}_M))h_i(\mathbf{x}_{pos})$$
 for  $i = 1, 2, ..., m$ .

Since  $g(x_M) = 0$  on the Pareto front, this modification does not change the Pareto fronts of DTLZ1-4. However, the feasible region in the objective space is heavily rescaled. In this paper, the newly created test problems from DTLZ1-4 using this objective function are referred to as the deceptive rescaled DTLZ1-4 test problems. In our computational experiments,  $\alpha$  is specified as  $\alpha = 10$  as in the rescaled DTLZ1-4. The Pareto fronts and the feasible regions of DTLZ2 and its two variants are illustrated in Fig. 13. As shown in Fig. 13b, the whole objective space is uniformly rescaled in the rescaled DTLZ1-4. However, in the deceptive rescaled DTLZ1-4, the feasible region is rescaled without changing the shape of the Pareto front, as shown in Fig. 13c.

We perform computational experiments on the rescaled and deceptive rescaled DTLZ1-4 test problems in the same manner as in the above-mentioned computational experiments on DTLZ1-4 and WFG4-9. Experimental results are summarized in Table 2. The average hypervolume value by each algorithm on each test problem is normalized by the best

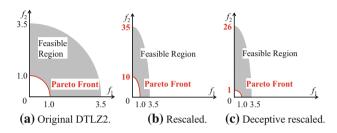


Fig. 13 Pareto fronts and the feasible regions of the original DTLZ2, the rescaled DTLZ2, and the deceptive rescaled DTLZ2

result. The best result for each test problem is highlighted by the bold font in Table 2, while the worst result is underlined.

The experimental results on the rescaled DTLZ1-4 in Table 2 clearly show the necessity of normalization. Good results are not obtained by MOEA/D without normalization and MOEA/DD. On the contrary, good results are obtained from MOEA/D without normalization for the deceptive rescaled DTLZ1-4 test problems. In Table 2, we can also observe severe performance deterioration of NSGA-III and  $\theta$ -DEA in their application to the deceptive rescaled DTLZ1 and DTLZ3 with eight and ten objectives. For visual examination, obtained solutions by a single run of MOEA/D and  $\theta$ -DEA on the ten-objective deceptive rescaled DTLZ1 are shown in Fig. 14. In Fig. 14b, most solutions are not converged to the Pareto front in  $[0, 1]^{10}$ . This observation suggests a future research topic: improvement of the normalization mechanisms in the state-of-the-art many-objective algorithms such as NSGA-III and  $\theta$ -DEA.

## **Experimental results on minus test problems**

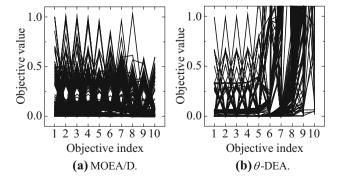
The minus version of DTLZ and WFG was proposed as many-objective test problems with inverted triangular Pareto fronts in [13]. The minus version can be easily formulated by changing each objective in DTLZ and WFG from  $f_i(x)$  to  $-f_i(x)$ . This modification is the same as changing from "minimization of  $f_i(x)$ " to "maximization of  $f_i(x)$ " in DTLZ and WFG. The main feature of the minus DTLZ and WFG test problems is that their Pareto fronts are inverted triangular,



Table 2 Normalized average hypervolume values on the new test problems

Problem	Objectives	MOEA/D	Normalized MOEA/D $(\varepsilon = 10^{-6})$	Normalized MOEA/D $(\varepsilon = 1)$	MOEA/DD	NSGA-III	θ-DEA
Rescaled DTLZ1	3	0.3276	0.9973	0.1575	0.6492	0.9992	1.0000
	5	0.3531	0.9997	0.1613	0.4632	0.9990	1.0000
	8	0.2291	0.7780	<u>0.1416</u>	0.3502	0.9947	1.0000
	10	0.2287	0.7287	0.2496	0.3482	0.9975	1.0000
Rescaled DTLZ2	3	0.2832	1.0000	0.9778	0.5318	0.9991	0.9998
	5	0.1940	1.0000	0.9965	0.3075	0.9870	0.9998
	8	0.1148	0.8259	0.8049	0.1621	0.9953	1.0000
	10	0.1092	0.9475	0.7955	0.1558	0.9976	1.0000
Rescaled DTLZ3	3	<u>0.1834</u>	0.9953	0.9099	0.5071	0.9976	1.0000
	5	0.1855	0.9893	0.7738	0.3252	0.9824	1.0000
	8	0.1148	0.8656	0.2277	0.1467	0.7214	1.0000
	10	0.1093	0.6570	0.1505	0.1508	0.9294	1.0000
Rescaled DTLZ4	3	0.2418	0.3561	0.2491	0.5257	1.0000	0.9920
	5	0.1911	0.3808	<u>0.1613</u>	0.3310	0.9997	1.0000
	8	0.1141	0.3220	0.1336	0.1511	0.9972	1.0000
	10	0.1091	0.3415	0.1115	0.1463	0.9999	1.0000
Deceptive rescaled DTLZ1	3	0.8579	0.9512	0.9136	1.0000	0.9960	0.9974
	5	0.9080	0.9996	0.9200	0.9714	0.9617	1.0000
	8	1.0000	0.0000	0.9959	0.7548	0.2640	0.5702
	10	1.0000	0.0000	0.9827	0.7875	0.1620	0.2732
Deceptive rescaled DTLZ2	3	0.8936	0.9367	<u>0.8576</u>	1.0000	0.9889	0.9948
	5	0.7898	0.9069	0.3273	0.8518	0.9473	1.0000
	8	0.7913	0.3393	0.6748	0.6794	0.8234	1.0000
	10	0.9243	0.3229	1.0000	0.6925	0.6250	0.9975
Deceptive rescaled DTLZ3	3	0.6596	0.7294	0.6243	1.0000	0.9461	0.9428
	5	0.5444	0.9514	0.7551	0.9023	0.8624	1.0000
	8	1.0000	0.0362	0.4996	0.2603	0.0012	0.1405
	10	1.0000	0.0297	0.3635	0.3415	0.0040	0.0339
Deceptive rescaled DTLZ4	3	0.6529	0.3731	0.2346	1.0000	0.9588	0.9745
	5	0.9010	0.2834	0.2409	0.9936	0.9707	1.0000
	8	0.9554	0.1694	0.1369	0.7511	0.9194	1.0000
	10	1.0000	0.1617	0.1483	0.7817	0.9720	0.9272

The best value is normalized to 1.0000 in each row



**Fig. 14** Obtained solution sets by a single run of MOEA/D without normalization and  $\theta$ -DEA on the ten-objective deceptive rescaled DTLZ1 test problem

whereas the Pareto fronts of DTLZ and WFG are triangular. We perform computational experiments on the minus DTLZ1-4 and WFG4-9 test problems in the same manner as in the previous computational experiments on DTLZ1-4 and WFG4-9.

Experimental results on the minus DTLZ1-4 and WFG4-9 are summarized in Table 3. The results on the minus WFG4-9 show the necessity of normalization. Good results are not obtained for those test problems by MOEA/D without normalization and MOEA/DD. One unexpected observation is that the best results are obtained from MOEA/D with normalization ( $\varepsilon=10^{-6}$ ) for most test problems in Table 3 except for the minus DTLZ1. That is, the original MOEA/D with



Table 3 Normalized average hypervolume values on the minus test problems

Problem	Objectives	MOEA/D	Normalized MOEA/D $(\varepsilon = 10^{-6})$	Normalized MOEA/D $(\varepsilon = 1)$	MOEA/DD	NSGA-III	θ-DEA
Minus DTLZ1	3	0.9594	0.9584	0.9606	0.9129	1.0000	0.9191
	5	1.0000	0.9607	0.9616	0.5667	0.7184	0.5097
	8	0.1402	0.8952	0.8957	0.1704	1.0000	0.8332
	10	0.0744	0.6903	0.6892	0.0921	1.0000	0.3926
Minus DTLZ2	3	0.9972	1.0000	0.9979	0.9889	0.9904	0.9940
	5	0.8972	1.0000	0.9998	0.4772	0.7963	0.7514
	8	0.9133	0.9656	1.0000	0.3958	0.6710	0.5354
	10	0.7771	0.9346	1.0000	0.2424	0.9217	0.7531
Minus DTLZ3	3	0.9994	0.9995	1.0000	0.9910	0.9948	0.9987
	5	0.9256	1.0000	0.9995	0.4637	0.7496	0.7580
	8	0.9469	1.0000	0.9969	0.3936	0.6517	0.6118
	10	0.8422	0.9986	1.0000	0.2696	0.8382	0.9285
Minus DTLZ4	3	0.8627	0.9816	0.8474	0.9910	0.9972	1.0000
	5	0.8146	1.0000	0.8342	0.4764	0.8174	0.7732
	8	0.4505	1.0000	0.6369	0.4501	0.9927	0.9147
	10	0.3698	0.8860	0.9417	0.4217	0.9801	1.0000
Minus WFG4	3	0.9907	0.9993	1.0000	0.9596	0.9593	0.9948
	5	0.7978	1.0000	0.9858	0.6427	0.7371	0.8224
	8	0.1020	1.0000	0.8518	0.0390	0.8236	0.8396
	10	0.0385	1.0000	0.7760	0.0117	0.9266	0.4467
Minus WFG5	3	0.9886	1.0000	0.9998	0.9730	0.9658	0.9943
	5	0.7971	1.0000	0.9807	0.7064	0.7391	0.6840
	8	0.1399	1.0000	0.8606	0.0982	0.7293	0.7465
	10	0.0472	1.0000	0.7664	0.0395	0.8073	0.4733
Minus WFG6	3	0.9863	1.0000	0.9998	0.9722	0.9854	0.9955
	5	0.7868	1.0000	0.9830	0.7017	0.7582	0.7317
	8	0.1328	1.0000	0.8514	0.1209	0.7477	0.6654
	10	0.0480	1.0000	0.7818	0.0428	0.9258	0.3733
Minus WFG7	3	0.9866	1.0000	0.9997	0.9517	0.9490	0.9945
	5	0.7925	1.0000	0.9912	0.6557	0.6815	0.6902
	8	0.0975	1.0000	0.7460	0.0773	0.7888	0.7182
	10	0.0365	1.0000	0.6784	0.0343	0.9392	0.4894
Minus WFG8	3	0.9849	1.0000	0.9983	0.9752	0.9900	0.9946
	5	0.7955	1.0000	0.9866	0.7515	0.7823	0.7081
	8	0.1499	1.0000	0.8669	0.2144	0.7837	0.7046
	10	0.0514	1.0000	0.7610	0.0868	0.9441	0.4050
Minus WFG9	3	0.9692	0.9851	0.9854	0.9406	0.9789	1.0000
	5	0.7911	1.0000	0.9827	0.6965	0.8083	0.7161
	8	0.1344	1.0000	0.8334	0.1414	0.8336	0.7614
	10	0.0453	1.0000	0.7317	0.0564	0.8693	0.4880

The best value is normalized to 1.0000 in each row

normalization clearly outperforms all the other algorithms, as shown in Table 3. This observation provides us an interesting research question: why the simple normalization mechanism can improve the performance of MOEA/D even on the minus DTLZ1-4 test problems that do not need any normalization

of the objective space. In Figs. 15, 16, 17, and 18, we show obtained solution sets by a single run (with the median average hypervolume value among the 51 runs) of MOEA/D with and without normalization ( $\varepsilon=10^{-6}$ ) on the 10-objective minus DTLZ1-4 problems. It seems from Figs. 15, 16, 17, and



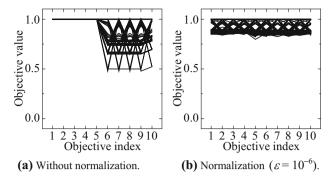


Fig. 15 Obtained solution sets by a single run of MOEA/D with and without normalization on the 10-objective minus DTLZ1

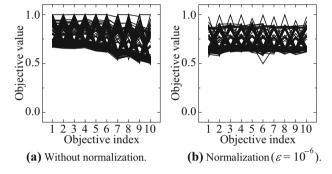


Fig. 16 Obtained solution sets by a single run of MOEA/D with and without normalization on the 10-objective minus DTLZ2

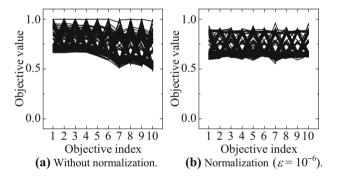


Fig. 17 Obtained solution sets by a single run of MOEA/D with and without normalization on the 10-objective minus DTLZ3

18 that the simple normalization mechanism helps MOEA/D to find well-distributed solutions around the center of the Pareto front. Further examinations are needed to explain why the simple normalization mechanism can improve the performance of MOEA/D for not only the minus WFG4-9 but also the minus DTLZ1-4.

From careful examinations of Table 3, we can observe that MOEA/D without normalization outperforms MOEA/DD on most minus DTLZ1-4 test problems, while MOEA/DD was the best on DTLZ1-4 in Table 1. We can also see that MOEA/D without normalization also outperforms NSGA-III and  $\theta$ -DEA on some minus DTLZ1-4 test problems. Similar observations were reported in [13]. The reason for the infe-

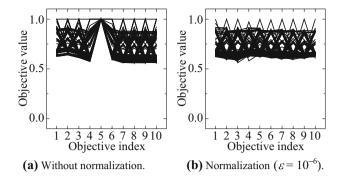


Fig. 18 Obtained solution sets by a single run of MOEA/D with and without normalization on the 10-objective minus DTLZ4

rior performance of those state-of-the-art algorithms (i.e., NSGA-III, MOEA/DD and  $\theta$ -DEA) is the inverted triangular shape of the Pareto fronts of minus DTLZ1-4. As pointed out in [13], since the simplex lattice structure of weight vectors is triangular, a large number of weight vectors exist outside the inverted triangular Pareto front of each of the minus DTLZ1-4 test problems. No solutions are assigned to those weight vectors in NSGA-III, MOEA/DD, and  $\theta$ -DEA, whereas each weight vector always has a single solution and continues to search for its best solution in MOEA/D. This difference in the handling of weight vectors outside the Pareto front has a significant influence on the performance comparison results, as shown in Table 3 (for details, see [4,13]). In [4], the simultaneous use of triangular and inverted triangular lattice structures of weight vectors in MOEA/D was proposed to remedy the high dependency of the performance of MOEA/D on the shape of the Pareto front.

The inferior performance of NSGA-III, MOEA/DD, and  $\theta$ -DEA in Table 3 can be explained in this manner using the shape of the Pareto front of each minus test problem. However, it is still totally unclear why the simple normalization mechanism improves the performance of MOEA/D on the minus DTLZ1-4 test problems, where the normalization of the object space is not needed.

## **Further computational experiments**

Our experimental results in this paper suggest that a very small range of objective values of non-dominated solutions in the current population is the main reason for the severe negative effects of the normalization on the performance of MOEA/D. One may think that the use of all solutions in the current population for the normalization may remedy its severe negative effects. That is, it may be a good idea to calculate the maximum value  $z_{iU}$  and the minimum value  $z_{iL}$  from all solutions in the current population (i.e., not only from non-dominated solutions but also from all the other solutions) in the following formulation of the normalization:



**Table 4** Comparison between the two normalization strategies on DTLZ1-4 and WFG4-9

Problem	Objectives	Based on no	n-dominated	solutions	Based on all	solutions	
		$\varepsilon = 10^{-12}$	$\varepsilon = 10^{-6}$	$\varepsilon = 1$	$\varepsilon = 10^{-12}$	$\varepsilon = 10^{-6}$	$\varepsilon = 1$
DTLZ1	3	0.9968	0.9963	0.9976	0.9094	0.9212	0.9976
	5	0.5014	0.9992	0.9997	0.1392	0.2431	0.9998
	8	0.2815	0.3827	0.9985	0.1000	0.1329	0.9992
	10	0.2523	0.3829	0.9998	0.1066	0.1095	0.9999
DTLZ2	3	0.9994	0.9994	0.9995	0.9993	0.9993	0.9995
	5	0.9995	0.9996	0.9996	0.9474	0.9821	0.9996
	8	0.3857	0.9383	0.9271	0.1191	0.1941	0.9999
	10	0.5454	0.9767	0.8650	0.1065	0.3532	1.0000
DTLZ3	3	0.9899	0.9902	0.9624	0.9111	0.9111	0.9947
	5	0.9856	0.9882	0.5262	0.1119	0.1119	0.9986
	8	0.3544	0.7676	0.4812	0.0984	0.0984	0.8968
	10	0.3288	0.8241	0.6730	0.0937	0.0937	1.0002
DTLZ4	3	0.3213	0.4122	0.3442	0.7130	0.6517	0.7324
	5	0.3010	0.5606	0.4284	0.9606	0.9538	0.9368
	8	0.2556	0.3519	0.3273	0.9866	0.9636	0.9404
	10	0.3068	0.5542	0.3577	0.9988	0.9964	0.9960
WFG4	3	0.9470	0.9469	0.9463	0.9457	0.9457	0.9483
	5	0.9854	0.9854	0.9850	0.9866	0.9865	0.9871
	8	0.3519	0.9277	0.9922	0.0992	0.0992	0.9933
	10	0.4854	1.0000	0.9985	0.0939	0.0986	0.9989
WFG5	3	0.9656	0.9656	0.9631	0.9638	0.9645	0.9619
	5	0.9957	0.9957	0.9894	0.9960	0.9960	0.9903
	8	0.1979	0.5417	0.9966	0.9343	0.9767	0.9966
	10	0.0874	0.5739	1.0000	0.6801	0.7632	1.0000
WFG6	3	0.9597	0.9597	0.9609	0.9617	0.9618	0.9586
	5	0.9946	0.9946	0.9943	0.9935	0.9934	0.9945
	8	0.2038	0.2252	0.9973	0.0967	0.0967	1.0006
	10	0.2286	0.3274	1.0000	0.0915	0.0915	1.0002
WFG7	3	0.8783	0.8782	0.8771	0.8741	0.8740	0.8793
	5	0.9914	0.9914	0.9901	0.9930	0.9930	0.9917
	8	0.1571	0.1617	0.9971	0.0987	0.0987	0.9980
	10	0.2408	0.3970	1.0000	0.0939	0.0977	1.0002
WFG8	3	0.9547	0.9547	0.9597	0.9560	0.9560	0.9592
	5	1.0000	1.0000	0.9945	0.9991	0.9991	0.9950
	8	0.1204	0.1207	1.0000	0.1118	0.1114	1.0001
	10	0.1126	0.1121	1.0000	0.1032	0.1030	1.0001
WFG9	3	0.8652	0.8652	0.8717	0.8704	0.8704	0.8599
	5	0.9616	0.9617	0.9539	0.9665	0.9665	0.9611
	8	0.1434	0.2536	0.9218	0.2261	0.1900	0.9288
	10	0.0833	0.1216	0.8855	0.1469	0.1464	0.8946

$$z_i := \frac{z_i - z_{iL}}{z_{iU} - z_{iL} + \varepsilon}$$
 for  $i = 1, 2, ..., m$ .

Let us examine this setting for the DTLZ1-4 and WFG4-9 test problems for three specifications of  $\varepsilon$ :  $10^{-12}$ ,  $10^{-6}$  and

1. Experimental results are summarized in Table 4. For comparison, we also show the results of the normalization based on non-dominated solutions in the current population. In Table 4, the best results are obtained for many test problems from the normalization based on all solutions with  $\varepsilon=1$ .



Table 5 Comparison between the two normalization strategies on the rescaled and deceptive rescaled DTLZ1-4

Problem	Objectives	Based on non-	-dominated solu	tions	Based on all s	solutions	
		$\varepsilon = 10^{-12}$	$\varepsilon = 10^{-6}$	$\varepsilon = 1$	$\varepsilon = 10^{-12}$	$\varepsilon = 10^{-6}$	$\varepsilon = 1$
Rescaled DTLZ1	3	0.9988	0.9973	0.1575	0.9109	0.9217	0.8928
	5	0.8424	0.9997	0.1613	0.1382	0.2202	0.7555
	8	0.3062	0.7780	0.1416	0.1000	0.1090	0.4345
	10	0.2747	0.7287	0.2496	0.0998	0.0909	0.4095
Rescaled DTLZ2	3	1.0000	1.0000	0.9778	1.0000	1.0000	0.9940
	5	0.9999	1.0000	0.9965	0.9651	0.9999	0.9965
	8	0.4432	0.8259	0.8049	0.1043	0.1264	0.4963
	10	0.8817	0.9475	0.7955	0.0938	0.0982	0.4312
Rescaled DTLZ3	3	0.9966	0.9953	0.9099	0.9161	0.9160	0.9950
	5	0.9923	0.9893	0.7738	0.1122	0.1122	0.9978
	8	0.0367	0.8656	0.2277	0.0986	0.0986	0.3895
	10	0.0641	0.6570	0.1505	0.0938	0.0938	0.3785
Rescaled DTLZ4	3	0.3573	0.3561	0.2491	0.6600	0.6868	0.7670
	5	0.3184	0.3808	0.1613	0.9607	0.9467	0.9422
	8	0.2207	0.3220	0.1336	0.9890	0.9753	0.3159
	10	0.1469	0.3415	0.1115	0.9987	0.9976	0.3267
Deceptive rescaled DTLZ1	3	0.9536	0.9512	0.9136	0.7535	0.7399	0.8987
	5	0.3207	0.9996	0.9200	0.0480	0.0259	0.9077
	8	0.0000	0.0000	0.9959	0.0000	0.0000	0.9908
	10	0.0000	0.0000	0.9827	0.0000	0.0000	0.9883
Deceptive rescaled DTLZ2	3	0.9300	0.9367	0.8576	0.6952	0.9976 0.7399 0.0259 0.0000 0.0000 0.6814	0.9503
	5	0.9594	0.9069	0.3273	0.1157	0.1392	1.0211
	8	0.3038	0.3393	0.6748	0.0561	0.0494	0.7976
	10	0.3171	0.3229	1.0000	0.0031	0.0015	1.0321
Deceptive rescaled DTLZ3	3	0.7399	0.7294	0.6243	0.6289	0.6095	0.6073
	5	0.9637	0.9514	0.7551	0.0012	0.0010	0.7758
	8	0.0000	0.0362	0.4996	0.0000	0.0000	0.6386
	10	0.0000	0.0297	0.3635	0.0000	0.0000	0.5582
Deceptive rescaled DTLZ4	3	0.3534	0.3731	0.2346	0.6371	0.6415	0.6341
	5	0.1947	0.2834	0.2409	0.8687	0.8230	0.8988
	8	0.1259	0.1694	0.1369	0.9262	0.9757	0.9620
	10	0.1306	0.1617	0.1483	0.8767	0.9593	1.0042

The same computational experiments are performed on the rescaled and deceptive rescaled DTLZ1-4 test problems. Experimental results are shown in Table 5. In this table, it is difficult to say which is better between the two normalization strategies. We also perform the same computational experiments on the minus DTLZ1-4 and WFG4-9 test problems. Experimental results are shown in Table 6. High performance of the normalized MOEA/D with  $\varepsilon=10^{-12}$  and  $\varepsilon=10^{-6}$  in Table 6 is clearly deteriorated by the use of all solutions for the normalization. Our experimental results in Tables 4 and 5 and 6 show that the choice between the two normalization strategies is problem dependent.

### **Conclusions**

In this paper, we clearly demonstrated the necessity of the objective space normalization in MOEA/D. Good results were not obtained from MOEA/D without normalization for the WFG4-9, rescaled DTLZ1-4, and minus WFG4-9 test problems. We examined the effect of the normalization by combining a simple normalization mechanism into MOEA/D. It was demonstrated through computational experiments that the normalization has both positive and negative effects on the performance of MOEA/D. That is, the performance of MOEA/D was improved by the incorporation of the simple normalization mechanism for some



Table 6 Comparison between the two normalization strategies on the minus DTLZ1-4 and WFG4-9

Problem	Objectives	Based on non-	dominated solutio	ns	Based on all so	olutions	
		$\varepsilon = 10^{-12}$	$\varepsilon = 10^{-6}$	$\varepsilon = 1$	$\varepsilon = 10^{-12}$	$\varepsilon = 10^{-6}$	$\varepsilon = 1$
Minus DTLZ1	3	0.9584	0.9584	0.9606	0.9596	0.9596	0.9604
	5	0.9607	0.9607	0.9616	1.0380	1.0380	1.0379
	8	0.8952	0.8952	0.8957	0.8480	0.8480	0.8460
	10	0.6903	0.6903	0.6892	0.5502	0.5502	0.5507
Minus DTLZ2	3	1.0000	1.0000	0.9979	0.9982	0.9982	0.9993
	5	0.9996	1.0000	0.9998	0.8943	0.8943	0.8951
	8	0.9656	0.9656	1.0000	0.8979	0.8979	0.9024
	10	0.9346	0.9346	1.0000	0.7807	0.7807	0.8039
Minus DTLZ3	3	0.9995	0.9995	1.0000	0.9998	0.9998	0.9996
	5	1.0000	1.0000	0.9995	0.9227	0.9227	0.9221
	8	1.0000	1.0000	0.9969	0.9371	0.9371	0.9387
	10	0.9986	0.9986	1.0000	0.8282	0.8282	0.8281
Minus DTLZ4	3	0.9818	0.9816	0.8474	0.9452	0.9302	0.8298
	5	1.0312	1.0000	0.8342	0.9960	0.9958	0.9815
	8	1.1847	1.0000	0.6369	0.6954	0.6876	0.4797
	10	0.9731	0.8860	0.9417	0.5223	0.5155	0.4368
Minus WFG4	3	0.9993	0.9993	1.0000	0.9994	0.9994	1.0007
	5	1.0000	1.0000	0.9858	0.9353	0.9353	0.9288
	8	1.0008	1.0000	0.8518	0.9365	0.9365	0.8143
	10	1.0002	1.0000	0.7760	0.8822	0.8824	0.7160
Minus WFG5	3	1.0000	1.0000	0.9998	1.0010	1.0010	1.0021
	5	1.0000	1.0000	0.9807	0.9301	0.9301	0.9184
	8	0.9995	1.0000	0.8606	0.9622	0.9622	0.8307
	10	1.0000	1.0000	0.7664	0.8929	0.8929	0.7207
Minus WFG6	3	1.0000	1.0000	0.9998	0.9999	0.9999	0.9997
	5	0.9998	1.0000	0.9830	0.9123	0.9123	0.9042
	8	1.0005	1.0000	0.8514	0.9433	0.9433	0.8185
	10	0.9997	1.0000	0.7818	0.8888	0.8888	0.7116
Minus WFG7	3	1.0002	1.0000	0.9997	0.9957	0.9957	0.9977
	5	1.0000	1.0000	0.9912	0.9240	0.9241	0.9176
	8	0.9995	1.0000	0.7460	0.9806	0.9803	0.7863
	10	1.0009	1.0000	0.6784	0.9606	0.9607	0.6692
Minus WFG8	3	1.0000	1.0000	0.9983	0.9997	0.9998	0.9989
	5	0.9998	1.0000	0.9866	0.9209	0.9209	0.9115
	8	1.0007	1.0000	0.8669	0.9424	0.9424	0.8181
	10	0.9986	1.0000	0.7610	0.8677	0.8677	0.6984
Minus WFG9	3	0.9851	0.9851	0.9854	0.9840	0.9841	0.9836
	5	1.0000	1.0000	0.9827	0.9529	0.9529	0.9411
	8	1.0000	1.0000	0.8334	0.9940	0.9941	0.8526
	10	1.0013	1.0000	0.7317	0.9769	0.9765	0.7757

test problems but degraded for other test problems. Our experimental results on the newly created deceptive rescaled DTLZ1-4 test problems suggest the existence of negative effects of the normalization on the performance of not only MOEA/D but also NSGA-III and  $\theta$ -DEA. Further examina-

tions are needed for evaluating positive and negative effects of the normalization on the performance of those recently proposed many-objective algorithms. In our computational experiments on the minus DTLZ1-4 and WFG4-9 test problems, the best results were obtained for most test problems



by MOEA/D with normalization. This observation suggests the existence of positive effects of the normalization even for test problems that do not need any normalization. Further examination of positive effects of the normalization for such a test problem is also an interesting future research topic.

Our computational experiments in this paper suggests the following issues as interesting future research topics:

- Development of an effective and robust normalization mechanism for MOEA/D: the observed negative effects of the simple normalization mechanism on the performance of MOEA/D throughout this paper clearly suggest the necessity of such a normalization mechanism in MOEA/D.
- 2. Improvement of the normalization mechanisms in NSGA-III and  $\theta$ -DEA: our experimental results on the deceptive rescaled DTLZ1-4 test problems suggest that the normalization mechanisms in NSGA-III and  $\theta$ -DEA can be further improved.
- 3. Examination on positive effects of the simple normalization mechanism on the performance of MOEA/D: Our experimental results on the minus DTLZ1-4 and WFG4-9 test problems pose the following question: Why MOEA/D with normalization can outperform MOEA/D without normalization, MOEA/DD, NSGA-III and  $\theta$ -DEA for both the minus DTLZ1-4 and WFG4-9 test problems whereas the minus DTLZ1-4 test problems need no normalization of the objective space.

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

#### References

- Asafuddoula M, Ray T, Sarker R (2015) A decomposition-based evolutionary algorithm for many objective optimization. IEEE Trans Evolut Comput 19(3):445–460
- Bader J, Zitzler E (2011) HypE: an algorithm for fast hypervolumebased many-objective optimization. Evolut Comput 19(1):45–76
- Beume N, Naujoks B, Emmerich M (2007) SMS-EMOA: multiobjective selection based on dominated hypervolume. Eur J Oper Res 181(3):1653–1669
- Bhattacharjee KS, Singh HK, Ray T, Zhang Q (2017) Decomposition based evolutionary algorithm with a dual set of reference vectors. In: Proc. of IEEE congress on evolutionary computation (CEC 2017), pp 105–112
- Deb K, Jain H (2014) An evolutionary many-objective optimization algorithm using reference-point-based non-dominated sorting approach, part I: solving problems with box constraints. IEEE Trans Evolut Comput 18(4):577–601

- Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evolut Comput 6(2):182–197
- Deb K, Thiele L, Laumanns M, Zitzler E (2002) Scalable multiobjective optimization test problems. In: Proc. of 2002 IEEE congress on evolutionary computation (CEC 2002), pp 825–830
- Giagkiozis I, Purshouse RC, Fleming PJ (2014) Generalized decomposition and cross entropy methods for many-objective optimization. Inf Sci 282:363–387
- Huband S, Hingston P, Barone L, While L (2006) A review of multiobjective test problems and a scalable test problem toolkit. IEEE Trans Evolut Comput 10(5):477–506
- Ishibuchi H, Akedo N, Nojima Y (2015) Behavior of multiobjective evolutionary algorithms on many-objective knapsack problems. IEEE Trans Evolut Comput 19(2):264–283
- Ishibuchi H, Doi K, Nojima Y (2016) Use of piecewise linear and nonlinear scalarizing functions in MOEA/D. In: Proc. of 14th international conference on parallel problem solving from nature. Lecture notes in computer science, vol 9921. Parallel problem solving from nature—PPSN XIV, Edinburgh, Scotland, UK, 17–21 September 2016, pp 503–523
- Ishibuchi H, Sakane Y, Tsukamoto N, Nojima Y (2009) Evolutionary many-objective optimization by NSGA-II and MOEA/D with large populations. In: Proc. of 2009 IEEE international conference on systems, man, and cybernetics (SMC 2009), pp 1758–1763
- Ishibuchi H, Setoguchi Y, Masuda H, Nojima Y (2017) Performance of decomposition-based many-objective algorithms strongly depends on Pareto front shapes. IEEE Trans Evolut Comput 21(2):169–190
- Ishibuchi H, Tsukamoto N, Nojima Y (2008) Evolutionary manyobjective optimization: a short review. Proc. of 2008 IEEE congress on evolutionary computation (CEC 2008), pp 2419–2426
- 15. Li B, Li J, Tang K, Yao X (2015) Many-objective evolutionary algorithms: a survey. ACM Comput Surv 48(1):1–35 ( Article 13)
- Li K, Deb K, Zhang Q, Kwong S (2015) An evolutionary many-objective optimization algorithm based on dominance and decomposition. IEEE Trans Evolut Comput 19(5):694

  –716
- Li K, Kwong S, Zhang Q, Deb K (2015) Interrelationship-based selection for decomposition multiobjective optimization. IEEE Trans Cybern 45(10):2076–2088
- Li K, Zhang Q, Kwong S, Li M, Wang R (2014) Stable matching based selection in evolutionary multiobjective optimization. IEEE Trans Evolut Comput 18(6):909–923
- Purshouse RC, Fleming PJ (2007) On the evolutionary optimization of many conflicting objectives. IEEE Trans Evolut Comput 11(6):770–784
- 20. Qi Y, Ma X, Liu F, Jiao L, Sun J, Wu J (2014) MOEA/D with adaptive weight adjustment. Evolut Comput 22(2):231–264
- Sato H (2014) Inverted PBI in MOEA/D and its impact on the search performance on multi- and many-objective optimization.
   In: Proc. of 2014 genetic and evolutionary computation conference (GECCO 2014) pp 645–652
- Trivedi A, Srinivasan D, Sanyal K, Ghosh A (2017) A survey of multiobjective evolutionary algorithms based on decomposition. IEEE Trans Evolut Comput 21(3):440–462
- Von Lücken C, Barán B, Brizuela C (2014) A survey on multiobjective evolutionary algorithms for many-objective problems. Comput Optim Appl 58(3):707–756
- Wang R, Zhou Z, Ishibuchi H, Liao T, Zhang T (2016) Localized weighted sum method for many-objective optimization. IEEE Trans Evol Comput. doi:10.1109/TEVC.2016.2611642
- Yuan's many-objective EA codes. [Online]. https://github.com/ yyxhdy/ManyEAs. Accessed 15 July 2016
- Yuan Y, Xu H, Wang B, Yao X (2016) A new dominance relation based evolutionary algorithm for many-objective optimization. IEEE Trans Evolut Comput 20(1):16–37



- Zhang Q, Li H (2007) MOEA/D: a multiobjective evolutionary algorithm based on decomposition. IEEE Trans Evolut Comput 11(6):712–731
- 28. Zhou A, Zhang Q (2016) Are all the subproblems equally important? Resource allocation in decomposition-based multiobjective evolutionary algorithms. IEEE Trans Evolut Comput 20(1):52–64
- Zitzler E, Laumanns M, Thiele L (2001) SPEA2: improving the strength Pareto evolutionary algorithm. TIK-Report 103, Computer Engineering and Networks Laboratory (TIK), Department of Electrical Engineering, ETH, Zurich

## **Publisher's Note**

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

