H. Ishibuchi, N. Akedo, and Y. Nojima, “A study on the specification of a scalarizing function in MOEA/D for many-objective knapsack problems,” *Lecture Notes in Computer Science Volume 7997: Learning and Intelligent Optimization - LION 7*, pp. 231-246, Springer, January 2013.

A Study on the Specification of a Scalarizing Function in MOEA/D for Many-Objective Knapsack Problems

Hisao Ishibuchi, Naoya Akedo and Yusuke Nojima

Department of Computer Science and Intelligent Systems, Graduate School of Engineering,
Osaka Prefecture University, 1-1 Gakuen-cho, Naka-ku, Sakai, Osaka 599-8531, Japan
{hisaoi@, naoya.akedo@ci., nojima@}cs.osakafu-u.ac.jp

Abstract. In recent studies on evolutionary multiobjective optimization, MOEA/D has been frequently used due to its simplicity, high computational efficiency, and high search ability. A multiobjective problem in MOEA/D is decomposed into a number of single-objective problems, which are defined by a single scalarizing function with evenly specified weight vectors. The number of the single-objective problems is the same as the number of weight vectors. The population size is also the same as the number of weight vectors. Multiobjective search for a variety of Pareto optimal solutions is realized by single-objective optimization of a scalarizing function in various directions. In this paper, we examine the dependency of the performance of MOEA/D on the specification of a scalarizing function. MOEA/D is applied to knapsack problems with 2-10 objectives. As a scalarizing function, we examine the weighted sum, the weighted Tchebycheff, and the PBI (penalty-based boundary intersection) function with a wide range of penalty parameter values. Experimental results show that the weighted Tchebycheff and the PBI function with an appropriate penalty parameter value outperformed the weighted sum and the PBI function with no penalty parameter in computational experiments on two-objective problems. However, better results were obtained from the weighted sum and the PBI function with no penalty parameter for many-objective problems with 6-10 objectives. We discuss the reason for these observations using the contour line of each scalarizing function. We also suggest potential usefulness of the PBI function with a negative penalty parameter value for many-objective problems.

Keywords: Evolutionary multiobjective optimization, many-objective problems, MOEA/D, scalarizing functions.

1 Introduction

Recently, high search ability of MOEA/D (multiobjective evolutionary algorithm based on decomposition [1]) has been reported in many studies [2]-[8]. MOEA/D is a simple and efficient scalarizing function-based EMO (evolutionary multiobjective optimization) algorithm. MOEA/D has clear advantages over Pareto dominance-based EMO algorithms such as NSGA-II [9] and SPEA2 [10]. For example, its scalarizing function-based fitness evaluation is very fast even for many-objective problems. Its hybridization with local search and other heuristics is often very easy.

One important issue in the implementation of MOEA/D is the choice of an appropriate scalarizing function. In our former study [11], we proposed an idea of automatically switching between the weighted sum and the weighted Tchebycheff during the execution of MOEA/D. We also proposed an idea of simultaneously using multiple scalarizing functions [12]. In the original study on MOEA/D [1], the weighted sum and the weighted Tchebycheff were used for multiobjective knapsack problems while the weighted Tchebycheff and the PBI (penalty-based boundary intersection) function were used for multiobjective continuous optimization. Good results were obtained from a different scalarizing function for a different test problem.

In this paper, we compare the weighted sum, the weighted Tchebycheff and the PBI function with each other, which were used in the original study on MOEA/D [1]. We examine a wide range of penalty parameter values in the PBI function. As test problems, we use multiobjective knapsack problems with 2-10 objectives. One of our test problems is the 2-500 (two-objective 500-item) knapsack problem in Zitzler & Thiele [13]. Many-objective test problems with 4-10 objectives are generated by adding new objectives to the 2-500 problem. Objectives in some test problems are strongly correlated [14] while objectives in other problems are not. MOEA/D with a different scalarizing function shows a different search behavior on each test problem.

This paper is organized as follows. In Section 2, we explain our implementation of MOEA/D, which is its basic version with no archive population [1]. In Section 2, we also explain the three scalarizing functions used in [1]: the weighted sum, the weighted Tchebycheff, and the PBI function. In Section 3, we compare the three scalarizing functions through computational experiments on the 2-500 problem. Experimental results by each scalarizing function are explained using its contour lines. It is also shown that the PBI function with different penalty parameter values has totally different contour lines. In Section 4, we examine the performance of each scalarizing function through computational experiments on many-objective problems. In Section 5, we suggest a potential usefulness of the PBI function with a negative penalty parameter value. Finally we conclude this paper in Section 6.

2 MOEA/D Algorithm

We explain the basic version of MOEA/D [1] with no archive population for the following m-objective maximization problem:

$$\text{Maximize} \quad \boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}),\ f_2(\boldsymbol{x}),\ ...,\ f_m(\boldsymbol{x}))\,, \tag{1}$$

where $\boldsymbol{f}(\boldsymbol{x})$ is an m-dimensional objective vector, $f_i(\boldsymbol{x})$ is the i-th objective to be maximized, and $\boldsymbol{x}$ is a decision vector. The use of the basic version is to concentrate on a choice of a scalarizing function. The maintenance of an archive population needs additional special care and large computation load in many-objective optimization.

The multiobjective problem in (1) is decomposed into a number of single-objective problems defined by a scalarizing function with different weight vectors. We examine the weighted sum, the weighted Tchebycheff and the PBI function used in [1]. The weighted sum is written using the weight vector $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, ..., \lambda_m)$ as

$$g^{WS}(\boldsymbol{x}\,|\,\boldsymbol{\lambda}) = \lambda_1 \cdot f_1(\boldsymbol{x}) + \lambda_2 \cdot f_2(\boldsymbol{x}) + \ \cdots \ + \lambda_m \cdot f_m(\boldsymbol{x})\,. \tag{2}$$

The weighted sum is to be maximized in its application to our multiobjective maximization problem. The weighted Tchebycheff in [1] is written using the weight vector $\boldsymbol{\lambda}$ and a reference point $\boldsymbol{z}^* = (z_1^*, z_2^*, ..., z_m^*)$ as

$$g^{TE}(\boldsymbol{x}\,|\,\boldsymbol{\lambda}, \boldsymbol{z}^*) = \max_{i=1,2,...,m} \{\lambda_i \cdot |z_i^* - f_i(\boldsymbol{x})|\}\,. \tag{3}$$

The weighted Tchebycheff is to be minimized. As in computational experiments in [1], we use the following specification of the reference point $\boldsymbol{z}^*$:

$$z_i^* = 1.1 \cdot \max\{f_i(\boldsymbol{x})\,|\,\boldsymbol{x} \in \Omega(t)\},\ i = 1, 2, ..., m\,, \tag{4}$$

where $\Omega(t)$ is the population at the t-th generation.

Zhang & Li [1] also used the PBI (penalty-based boundary intersection) function:

$$g^{PBI}(\boldsymbol{x}\,|\,\boldsymbol{\lambda}, \boldsymbol{z}^*) = d_1 + \theta d_2\,, \tag{5}$$

$$d_1 = \frac{\left\| (\boldsymbol{z}^* - \boldsymbol{f}(\boldsymbol{x}))^T \boldsymbol{\lambda} \right\|}{\left\| \boldsymbol{\lambda} \right\|}\,, \tag{6}$$

$$d_2 = \left\| \boldsymbol{f}(\boldsymbol{x}) - (\boldsymbol{z}^* - d_1 \boldsymbol{\lambda}) \right\|. \tag{7}$$

In this formulation, $\boldsymbol{z}^*$ is the reference point in (4) and θ is a user-definable penalty parameter. In [1], θ was specified as $\theta = 5$. The PBI function is to be minimized. For detailed explanations on scalarizing functions and their characteristic features, see textbooks on multiobjective optimization (e.g., see Miettinen [15])

In each of the three scalarizing functions, all weight vectors $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, ..., \lambda_m)$ satisfying the following two conditions are used for the m-objective problem:

$$\lambda_1 + \lambda_2 + \cdots + \lambda_m = 1\,, \tag{8}$$

$$\lambda_i \in \left\{0, \frac{1}{H}, \frac{2}{H}, ..., \frac{H}{H}\right\},\ i = 1, 2, ..., m\,, \tag{9}$$

where H is a user-definable positive integer. For example, we have a set of evenly specified two-dimensional 101 vectors for $H = 100$ and $m = 2$: $\boldsymbol{\lambda} = (0, 1), (0.01, 0.99), (0.02, 0.98), ..., (0.99, 0.01), (1, 0)$. The number of weight vectors can be calculated as $N = {}_{H+m-1}C_{m-1}$. The number of weight vectors is the same as the population size.

Let N be the number of weight vectors (i.e., N is the population size). We denote N weight vectors as $\boldsymbol{\lambda}^k$, $k = 1, 2, ..., N$. Each weight vector $\boldsymbol{\lambda}^k$ has the nearest T weight vectors as its neighbors where T is a user-definable positive integer. In MOEA/D, first N solutions are randomly generated as an initial population in the same manner as in other evolutionary algorithms. Each solution is assigned to a different weight vector.

Let $\boldsymbol{x}^k$ be the solution assigned to the k-th weight vector $\boldsymbol{\lambda}^k$. For each weight vector $\boldsymbol{\lambda}^k$ with the current solution $\boldsymbol{x}^k$, a pair of parent solutions is randomly selected from its neighbors to generate a new solution $\boldsymbol{y}^k$ by crossover and mutation. The newly generated solution $\boldsymbol{y}^k$ is compared with the current solution $\boldsymbol{x}^k$ using a scalarizing function with the k-th weight vector $\boldsymbol{\lambda}^k$. If the newly generated solution $\boldsymbol{y}^k$ is better, the current solution $\boldsymbol{x}^k$ is replaced with $\boldsymbol{y}^k$. The newly generated solution $\boldsymbol{y}^k$ is also compared with each of its neighbors. Let $\boldsymbol{\lambda}^h$ be a neighbor of the k-th weight vector $\boldsymbol{\lambda}^k$. The newly generated solution $\boldsymbol{y}^k$ for the k-th weight vector $\boldsymbol{\lambda}^k$ is compared with the current solution $\boldsymbol{x}^h$ of each neighbor $\boldsymbol{\lambda}^h$ using the scalarizing function with the neighbor's weight vector $\boldsymbol{\lambda}^h$. If $\boldsymbol{y}^k$ is better, the current solution $\boldsymbol{x}^h$ is replaced with $\boldsymbol{y}^k$. In this manner, the newly generated solution for each weight vector is locally compared with its own and its neighbors' current solutions.

In MOEA/D, local and global parent selection can be probabilistically used. In our implementation, we always use the above-mentioned local parent solution. The upper bound on the number of replaced current solutions with a newly generated one can be specified in MOEA/D. We do not use any upper bound on the number of replaced current solutions in our implementation. MOEA/D also has an option of using an archive population to store non-dominated solutions. We do not use any archive population. All of these settings are to clearly examine the effect of the choice of a scalarizing function on the performance of MOEA/D under a simple situation.

3 Experimental Results on a Two-Objective Knapsack Problem

In this section, we report experimental results on the 2-500 (two-objective 500-item) knapsack problem in Zitzler & Thiele [13]. This problem is written as follows:

$$\text{Maximize} \quad f_i(\boldsymbol{x}) = \sum_{j=1}^{n} p_{ij} x_j \,, \quad i = 1, 2, \tag{10}$$

$$\text{subject to} \quad \sum_{j=1}^{n} w_{ij} x_j \le c_i \,, \quad i = 1, 2, \tag{11}$$

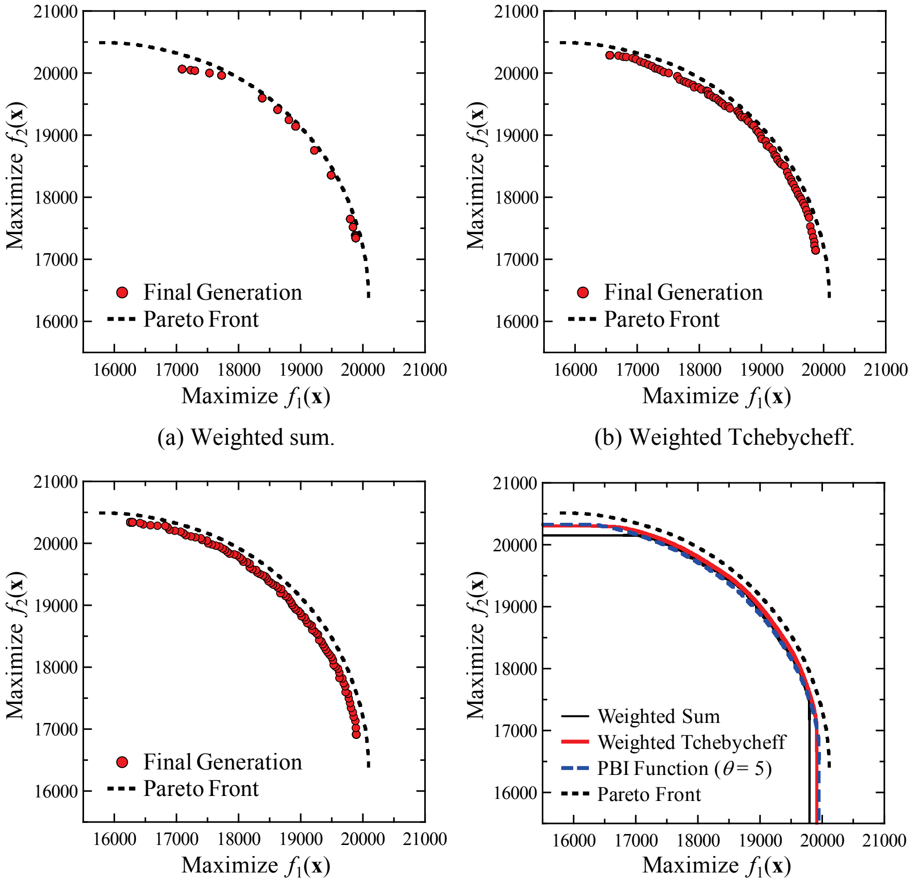
$$x_j = 0 \text{ or } 1, \quad j = 1, 2, ..., n, \tag{12}$$

where n is the number of items (i.e., n = 500), $\boldsymbol{x}$ is a 500-bit binary string, p_{ij} is the profit of item j according to knapsack i, w_{ij} is the weight of item j according to knapsack i, and c_i is the capacity of knapsack i [13]. The values of each profit p_{ij} and each weight w_{ij} were randomly specified integers in [10, 100], and the constant value c_i was set as a half of the total weight value (i.e., $c_i = (w_{i1} + w_{i2} + \cdots + w_{in})/2$) in [13]. In the next section, we generate many-objective knapsack problems by adding new objectives to the 2-500 problem.

We applied MOEA/D with the weighted sum, the weighted Tchebycheff, and the PBI function ($\theta = 5$) to the 2-500 problem using the following specifications:

Coding: 500-bit binary string, Population size: 200,
Termination condition: 200 × 2000 solution evaluations,
Parent selection: Random selection from the neighborhood,
Crossover: Uniform crossover (Probability: 0.8),
Mutation: Bit-flip mutation (Probability: 1/500 for each bit),
Constraint Handling: Maximum profit/weight ratio-based repair in [13],
Number of runs of MOEA/D with each scalarizing function: 100 runs.

Solutions in the final generation of a single run of MOEA/D with each scalarizing function are shown in Fig. 1 (a)-(c). Fig. 1 (d) shows the 50% attainment surface over 100 runs of MOEA/D with each scalarizing function. In Fig. 1, better distributions of solutions were obtained by the weighted Tchebycheff and the PBI function ($\theta = 5$).



(a) Weighted sum.

(b) Weighted Tchebycheff.

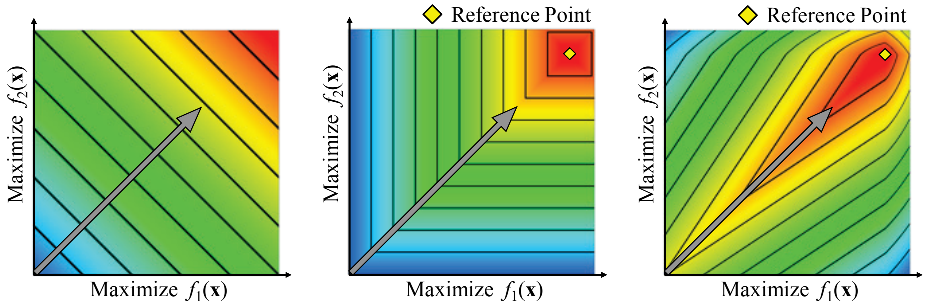
(c) PBI function ($\theta = 5$).

(d) 50% attainment surface.

Fig. 1. Experimental results on the 2-500 problem.

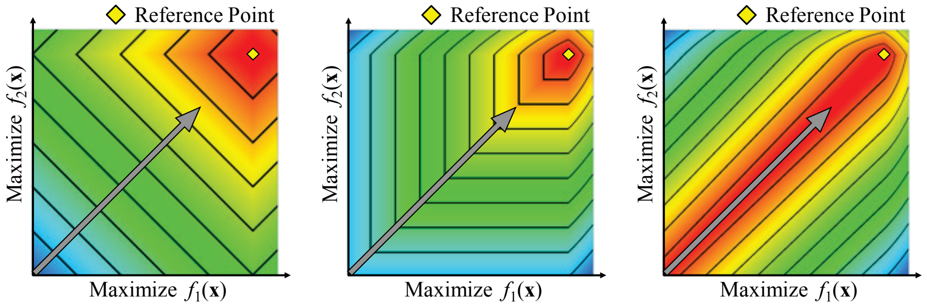
In Fig. 2, we show the contour lines of each scalarizing function for the weighted vector $\boldsymbol{\lambda} = (0.5, 0.5)$. Contour lines have been used to explain scalarizing functions in the literature [15]. When we used the weighted sum in Fig. 2 (a), the same solution was often obtained from different weight vectors. As a result, many solutions were not obtained in Fig. 1 (a). In the case of the weighted Tchebycheff in Fig. 2 (b), the same solution was not often obtained from different weight vectors. Thus more solutions were obtained in Fig. 1 (b) than Fig. 1 (a). The best results with respect to the number of solutions were obtained in Fig. 1 from the PBI function with $\theta = 5$ in Fig. 1 (c) due to the sharp edge of the contour lines in Fig. 2 (c). In Fig. 1 (d), the PBI function slightly outperformed the weighted Tchebycheff with respect to the width of the obtained solution sets along the Pareto front.

The shape of the contour lines of the PBI function totally depends on the value of θ. Fig. 3 shows the contour lines of the PBI function with some other values of θ. When $\theta = 0$, the contour lines in Fig. 3 (a) are similar to those of the weighted sum in Fig. 2 (a). The contour lines of the PBI function with $\theta = 1$ in Fig. 3 (b) look somewhat similar to those of the weighted Tchebycheff in Fig. 2 (b). The valley of the contour lines in Fig. 3 (c) with $\theta = 50$ is much longer than that in Fig. 2 (c) with $\theta = 5$. This means that the contour lines with $\theta = 50$ have a sharper edge, which leads to slower convergence speed towards the reference point (i.e., towards the Pareto front).


◇ Reference Point
◇ Reference Point
Maximize $f_2(\mathbf{x})$
Maximize $f_1(\mathbf{x})$
Maximize $f_2(\mathbf{x})$
Maximize $f_1(\mathbf{x})$
Maximize $f_2(\mathbf{x})$
Maximize $f_1(\mathbf{x})$

(a) Weighted sum. (b) Weighted Tchebycheff. (c) PBI function ($\theta = 5$).

Fig. 2. Contour lines of the three scalarizing functions.

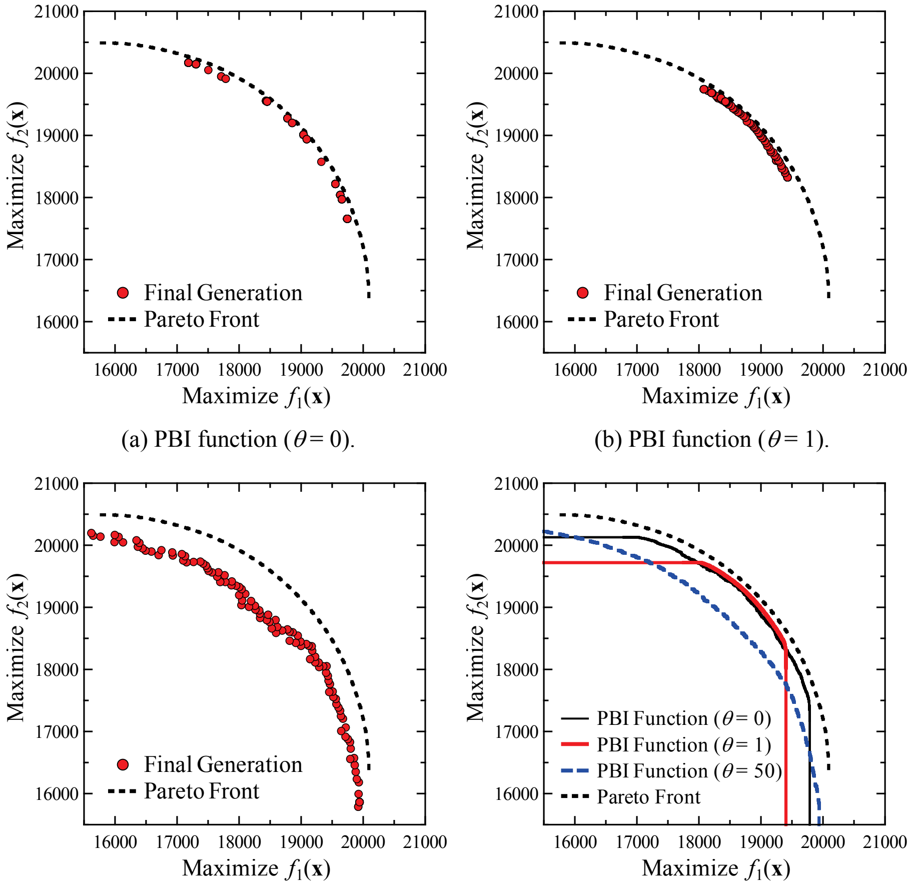

◇ Reference Point
◇ Reference Point
◇ Reference Point
Maximize $f_2(\mathbf{x})$
Maximize $f_1(\mathbf{x})$
Maximize $f_2(\mathbf{x})$
Maximize $f_1(\mathbf{x})$
Maximize $f_2(\mathbf{x})$
Maximize $f_1(\mathbf{x})$

(a) PBI function ($\theta = 0$). (b) PBI function ($\theta = 1$). (c) PBI function ($\theta = 50$).

Fig. 3. Contour lines of the PBI function with different values of the penalty parameter θ.

Fig. 4 shows experimental results using each value of the penalty parameter θ in Fig. 3. As expected, the use of $\theta = 50$ in Fig. 4 (c) deteriorated the convergence ability of MOEA/D from Fig. 1 (c) with $\theta = 5$. Fig. 4 (a) with $\theta = 0$ is similar to Fig. 1 (a) with the weighted sum. However, Fig. 4 (b) with $\theta = 1$ is not similar to Fig. 1 (b) with the weighted Tchebycheff. This is because the contour lines of the two scalarizing functions are not similar to each other for other weight vectors (see Fig. 5).

Fig. 5 shows the contour lines of each scalarizing function for the five weight vectors: $\boldsymbol{\lambda} = (1, 0), (0.75, 0.25), (0.5, 0.5), (0.25, 0.75), (0, 1)$. The contour lines of the weighted sum are similar to those of the PBI function with $\theta = 0$ in Fig. 5. Thus similar results were obtained from these two scalarizing functions in Fig. 1 and Fig. 4. Except for the PBI function with $\theta = 1$, the contour lines are (almost) vertical or (almost) horizontal when $\boldsymbol{\lambda} = (1, 0)$ and $\boldsymbol{\lambda} = (0, 1)$. Thus a wide variety of solutions were obtained in Fig. 1 and Fig. 4 except for the case of the PBI function with $\theta = 1$.



(a) PBI function (θ= 0).

(b) PBI function (θ= 1).

(c) PBI function (θ= 50).

(d) 50% attainment surface.

Fig. 4. Experimental results by the PBI function on the 2-500 problem.

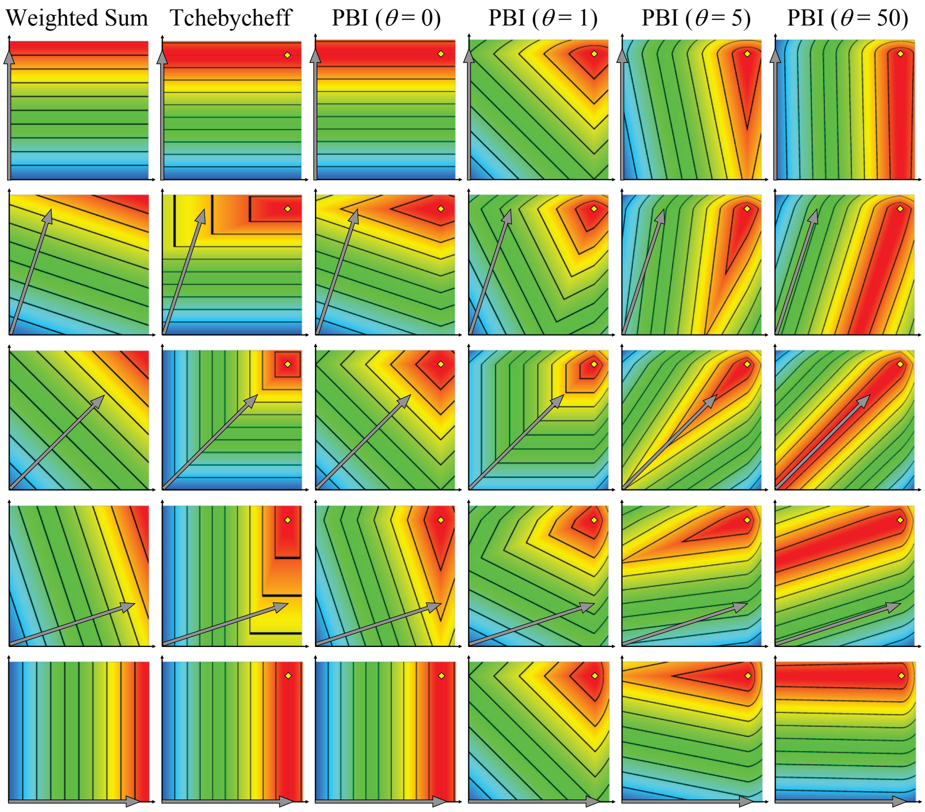


Fig. 5. Contour lines of the scalarizing functions for various weight vectors.

4 Experimental Results on Many-Objective Knapsack Problems

We generated many-objective test problems by adding new objectives to the 2-500 problem. More specifically, we generated eight additional objectives $f_i(\boldsymbol{x})$ in the form of Eq. (10) by randomly specifying the value of the profit p_{ij} as an integer in [10, 100] for $i = 3, 4, ..., 10$ and $j = 1, 2, ..., 500$. Then we specified ten new objectives $g_i(\boldsymbol{x})$ in the following manner using a real number parameter α ($0 \le \alpha \le 1$) where α can be viewed as the correlation strength among the generated objectives:

$$g_1(\boldsymbol{x}) = f_1(\boldsymbol{x}) \quad \text{and} \quad g_2(\boldsymbol{x}) = f_2(\boldsymbol{x}), \tag{13}$$

$$g_i(\boldsymbol{x}) = \alpha f_1(\boldsymbol{x}) + (1-\alpha) f_i(\boldsymbol{x}) \quad \text{for} \quad i = 3,5,7,9, \tag{14}$$

$$g_i(\boldsymbol{x}) = \alpha f_2(\boldsymbol{x}) + (1-\alpha) f_i(\boldsymbol{x}) \quad \text{for} \quad i = 4,6,8,10. \tag{15}$$

Using these ten objectives $g_i(\boldsymbol{x})$, $i = 1, 2, ..., 10$ and the constraint conditions of the original 2-500 knapsack problem in (10)-(12), we generated many-objective knapsack problems with four, six, eight and ten objectives (i.e., 4-500, 6-500, 8-500 and 10-500 problems). It should be noted that all of those test problems have the same constraint condition as in the 2-500 problem. This means that all of our test problems have the same feasible solution set as in the 2-500 problem.

In one extreme case with $\alpha = 1$ in (13)-(15), our many-objective test problems have only two different objectives $g_1(\boldsymbol{x})$ and $g_2(\boldsymbol{x})$ (i.e., $f_1(\boldsymbol{x})$ and $f_2(\boldsymbol{x})$ of the 2-500 problem). In the other extreme case with $\alpha = 0$, all objectives $g_i(\boldsymbol{x})$ are totally different from each other and have no strong correlation since profit values were randomly specified in each objective. We examined six values of α: $\alpha = 0, 0.2, 0.4, 0.6, 0.8, 1$.

We applied MOEA/D with the weighted sum, the weighed Tchebycheff and the PBI function ($\theta = 0, 0.1, 0.5, 1, 5, 10, 50$) to our many-objective 4-500, 6-500, 8-500 and 10-500 problems with $\alpha = 0, 0.2, 0.4, 0.6, 0.8, 1$. For comparison, we also applied NSGA-II and SPEA2 to the same test problems. Computational experiments were performed using the same parameter specifications as in Section 3 except for the population size in MOEA/D due to the combinatorial nature of the number of weight vectors. The population size in MOEA/D was specified as 220 (4-500), 252 (6-500), 120 (8-500), and 220 (10-500). In NSGA-II and SPEA2, the population size was always specified as 200. We evaluated the performance of each algorithm on each test problem using the average hypervolume over 100 runs. The origin of the objective space of each test problem (i.e., (0, 0, ..., 0)) was used as a reference point for hypervolume calculation.

Experimental results are summarized in Tables 1-4 where all experimental results are normalized using the results of MOEA/D with the weighted sum for each test problem. The average hypervolume value by MOEA/D with the weighted sum is normalized as 1.00. In Table 1, experimental results on the 2-500 problem are also included for comparison. The largest normalized average hypervolume value for each test problem is highlighted by bold face in Tables 1-4. Good results were obtained for all test problems by MOEA/D with the weighted sum and the PBI function with $\theta = 0$. The weighted Tchebycheff did not work well on many-objective problems with no or small correlation among objectives (e.g., 10-500 with $\alpha = 0.2$) while it worked well on many-objective problems with highly correlated objectives (e.g., 10-500 with $\alpha = 0.8$). It also worked well on the 2-500 and 4-500 problems. For the 2-500 problem, the best results were obtained by the PBI function with $\theta = 5$ and $\theta = 10$.

Table 1. Normalized average hypervolume on the 4-500 problems.

Problem (4-500)	NSGA-II	SPEA2	Weighted sum	Tcheby-cheff	PBI (θ)						
					0	0.1	0.5	1	5	10	50
$\alpha = 0.0$	0.86	0.85	**1.00**	**1.00**	**1.00**	0.96	0.82	0.78	0.92	0.94	0.95
$\alpha = 0.2$	0.90	0.89	1.00	**1.01**	1.00	0.97	0.85	0.83	0.94	0.95	0.96
$\alpha = 0.4$	0.95	0.92	1.00	**1.01**	1.00	0.97	0.88	0.87	0.96	0.97	0.98
$\alpha = 0.6$	0.96	0.94	1.00	**1.01**	1.00	0.97	0.89	0.88	0.97	0.99	1.00
$\alpha = 0.8$	0.95	0.94	1.00	**1.01**	1.00	0.97	0.89	0.88	0.97	0.99	1.00
$\alpha = 1.0$	0.94	0.94	1.00	**1.02**	1.00	0.98	0.88	0.87	0.97	0.99	1.00
2-500	0.96	0.96	1.00	1.01	1.00	0.99	0.93	0.97	**1.02**	**1.02**	1.01

Table 2. Normalized average hypervolume on the 6-500 problems.

Problem (6-500)	NSGA-II	SPEA2	Weighted sum	Tcheby-cheff	PBI (θ)						
					0	0.1	0.5	1	5	10	50
$\alpha = 0.0$	0.78	0.74	**1.00**	0.94	**1.00**	0.96	0.79	0.67	0.76	0.77	0.78
$\alpha = 0.2$	0.86	0.81	**1.00**	0.97	**1.00**	0.96	0.82	0.74	0.81	0.82	0.83
$\alpha = 0.4$	0.93	0.89	**1.00**	0.99	**1.00**	0.97	0.86	0.80	0.87	0.89	0.90
$\alpha = 0.6$	0.97	0.92	**1.00**	**1.00**	**1.00**	0.97	0.87	0.82	0.89	0.91	0.93
$\alpha = 0.8$	0.96	0.92	1.00	**1.02**	1.00	0.97	0.86	0.82	0.89	0.90	0.92
$\alpha = 1.0$	0.94	0.93	1.00	**1.01**	1.00	0.98	0.86	0.82	0.89	0.91	0.93

Table 3. Normalized average hypervolume on the 8-500 problems.

Problem (8-500)	NSGA-II	SPEA2	Weighted sum	Tcheby-cheff	PBI (θ)						
					0	0.1	0.5	1	5	10	50
$\alpha = 0.0$	0.73	0.68	**1.00**	0.90	**1.00**	0.96	0.78	0.66	0.68	0.70	0.71
$\alpha = 0.2$	0.83	0.77	**1.00**	0.91	**1.00**	0.96	0.82	0.72	0.74	0.76	0.77
$\alpha = 0.4$	0.92	0.86	**1.00**	0.94	**1.00**	0.97	0.85	0.78	0.81	0.82	0.83
$\alpha = 0.6$	0.97	0.91	**1.00**	0.99	**1.00**	0.97	0.86	0.80	0.83	0.85	0.86
$\alpha = 0.8$	0.99	0.94	1.00	**1.02**	1.00	0.98	0.87	0.81	0.84	0.86	0.87
$\alpha = 1.0$	0.98	0.97	1.00	**1.01**	1.00	1.00	0.88	0.82	0.85	0.87	0.89

Table 4. Normalized average hypervolume on the 10-500 problems.

Problem (10-500)	NSGA-II	SPEA2	Weighted sum	Tcheby-cheff	PBI (θ)						
					0	0.1	0.5	1	5	10	50
$\alpha = 0.0$	0.66	0.60	**1.00**	0.87	**1.00**	0.95	0.76	0.63	0.62	0.64	0.65
$\alpha = 0.2$	0.77	0.71	**1.00**	0.87	**1.00**	0.95	0.79	0.69	0.68	0.70	0.72
$\alpha = 0.4$	0.88	0.82	**1.00**	0.91	**1.00**	0.96	0.83	0.74	0.75	0.77	0.78
$\alpha = 0.6$	0.96	0.90	**1.00**	0.98	**1.00**	0.97	0.84	0.78	0.78	0.80	0.82
$\alpha = 0.8$	1.00	0.94	1.00	**1.02**	1.00	0.98	0.85	0.79	0.80	0.81	0.83
$\alpha = 1.0$	1.00	0.99	1.00	**1.01**	1.00	**1.01**	0.88	0.81	0.82	0.84	0.86

As pointed out by many studies [16]-[21], Pareto dominance-based EMO algorithms do not work well on many-objective problems. In Tables 1-4, we can observe the performance deterioration of NSGA-II and SPEA2 by the increase in the number of objectives. In the case of a two-objective maximization problem, a solution is dominated by other solutions in its upper right region of the objective space. The relative size of this region can be viewed as being 1/4 of the objective space. The relative size of this region exponentially decreases as the number of objectives increases (e.g., 1/1024 in the case of ten objectives). This is the reason why Pareto dominance-based EMO algorithms do not work well on many-objective problems.

The same reason can be used to explain why the performance of MOEA/D with the weighted Tchebycheff was deteriorated by the increase in the number of objectives. Let us assume a solution of the 2-500 problem at the corner of a contour line of the weighted Tchebycheff in Fig. 2 (b). This solution is outperformed by other solutions in its upper right region. The relative size of this region can be viewed as being 1/4 of the objective space. The relative size of this region exponentially decreases as the number of objective increases. As a result, it becomes very difficult for MOEA/D to find a better solution with respect to the weighted Tchebycheff. That is, the convergence of solutions towards the reference point using the weighted Tchebycheff is slowed down by the increase in the number of objectives. The performance deterioration of MOEA/D due to the increase in the number of objectives was more severe in Tables 1-4 for the PBI function than the weighted Tchebycheff except for the case of zero or small penalty values. This is because the contour lines of the PBI function have a sharper edge than those of the weighted Tchebycheff. It is more difficult for MOEA/D to find a better solution with respect to the PBI function with a large penalty parameter value than the case of the weighted Tchebycheff.

In Tables 1-4, we also observe that NSGA-II worked well on many-objective problems with highly correlated objectives (e.g., 10-500 with $\alpha = 0.8$) as pointed out in [14]. This is because the relative size of the above-mentioned dominating region in the objective space does not exponentially decrease with the number of highly correlated objectives (i.e., many solutions are not non-dominated with each other in the case of highly correlated objectives). As a result, the performance of NSGA-II and SPEA2 was not severely deteriorated by the increase in the number of objectives in Tables 1-4. For the same reason, MOEA/D with the weighted Tchebycheff worked well on many-objective problems with highly correlated objectives.

5 Use of a Negative Penalty Parameter Value

In Tables 1-4, the performance of MOEA/D with the PBI function on many-objective problems was improved by using a smaller value for the penalty parameter θ (e.g., see the results on the 10-500 problem with $\alpha = 0.0$). From this observation, one may think that better results would be obtained from a negative value of θ.

To examine this issue, we further performed computational experiments using three negative values of θ: $\theta = -1, -0.5, -0.1$. Experimental results are summarized in Tables 5-8 where the normalized average hypervolume is calculated in the same manner as in Tables 1-4. The origin of the objective space of each test problem was used in Tables 5-8 for hypervolume calculation as in Tables 1-4. We can see from Tables 5-8 that the best results were obtained from the PBI function with $\theta = -0.1$ for almost all test problems except for those with highly correlated objectives.

Table 5. Normalized average hypervolume on 4-500 (reference point: origin).

Problem (4-500)	Weighted sum	Tcheby-cheff	PBI (θ)									
			-1	-0.5	-0.1	0	0.1	0.5	1	5	10	50
$\alpha = 0.0$	1.00	1.00	0.78	0.95	**1.03**	1.00	0.96	0.82	0.78	0.92	0.94	0.95
$\alpha = 0.2$	1.00	1.01	0.80	0.98	**1.03**	1.00	0.97	0.85	0.83	0.94	0.95	0.96
$\alpha = 0.4$	1.00	1.01	0.80	0.99	**1.02**	1.00	0.97	0.88	0.87	0.96	0.97	0.98
$\alpha = 0.6$	1.00	1.01	0.79	0.97	**1.02**	1.00	0.97	0.89	0.88	0.97	0.99	1.00
$\alpha = 0.8$	1.00	1.01	0.75	0.94	**1.02**	1.00	0.97	0.89	0.88	0.97	0.99	1.00
$\alpha = 1.0$	1.00	**1.02**	0.71	0.92	1.01	1.00	0.98	0.88	0.87	0.97	0.99	1.00
2-500	1.00	1.01	0.93	0.99	1.01	1.00	0.99	0.93	0.97	**1.02**	**1.02**	1.01

Table 6. Normalized average hypervolume on 6-500 (reference point: origin).

Problem (6-500)	Weighted sum	Tcheby-cheff	PBI (θ)									
			-1	-0.5	-0.1	0	0.1	0.5	1	5	10	50
$\alpha = 0.0$	1.00	0.94	0.53	0.83	**1.03**	1.00	0.96	0.79	0.67	0.76	0.77	0.78
$\alpha = 0.2$	1.00	0.97	0.55	0.86	**1.04**	1.00	0.96	0.82	0.74	0.81	0.82	0.83
$\alpha = 0.4$	1.00	0.99	0.56	0.88	**1.03**	1.00	0.97	0.86	0.80	0.87	0.89	0.90
$\alpha = 0.6$	1.00	1.00	0.53	0.86	**1.02**	1.00	0.97	0.87	0.82	0.89	0.91	0.93
$\alpha = 0.8$	1.00	**1.02**	0.46	0.81	1.01	1.00	0.97	0.86	0.82	0.89	0.90	0.92
$\alpha = 1.0$	1.00	**1.01**	0.42	0.74	1.00	1.00	0.98	0.86	0.82	0.89	0.91	0.93

Table 7. Normalized average hypervolume on 8-500 (reference point: origin).

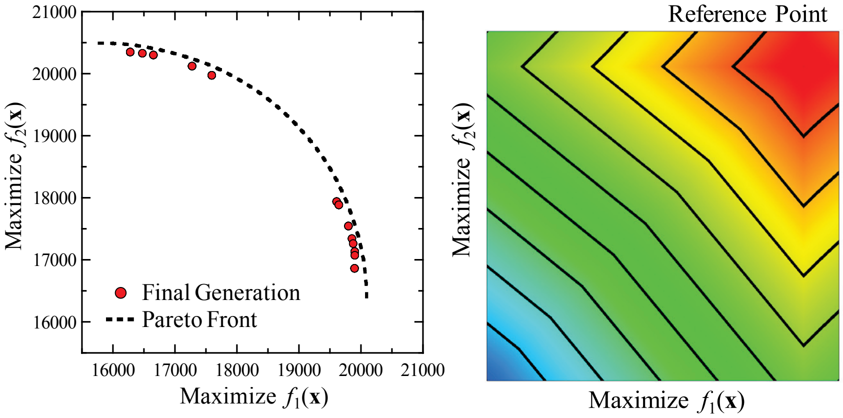
Problem (8-500)	Weighted sum	Tcheby-cheff	PBI (θ)									
			-1	-0.5	-0.1	0	0.1	0.5	1	5	10	50
$\alpha = 0.0$	1.00	0.90	0.00	0.56	**1.02**	1.00	0.96	0.78	0.66	0.68	0.70	0.71
$\alpha = 0.2$	1.00	0.91	0.00	0.57	**1.03**	1.00	0.96	0.82	0.72	0.74	0.76	0.77
$\alpha = 0.4$	1.00	0.94	0.00	0.58	**1.03**	1.00	0.97	0.85	0.78	0.81	0.82	0.83
$\alpha = 0.6$	**1.00**	0.99	0.00	0.59	**1.00**	**1.00**	0.97	0.86	0.80	0.83	0.85	0.86
$\alpha = 0.8$	1.00	**1.02**	0.00	0.48	0.99	1.00	0.98	0.87	0.81	0.84	0.86	0.87
$\alpha = 1.0$	1.00	**1.01**	0.00	0.41	0.97	1.00	1.00	0.88	0.82	0.85	0.87	0.89

Table 8. Normalized average hypervolume on 10-500 (reference point: origin).

Problem (10-500)	Weighted sum	Tcheby-cheff	PBI (θ)									
			-1	-0.5	-0.1	0	0.1	0.5	1	5	10	50
$\alpha = 0.0$	1.00	0.87	0.00	0.36	**1.01**	1.00	0.95	0.76	0.63	0.62	0.64	0.65
$\alpha = 0.2$	1.00	0.87	0.00	0.38	**1.04**	1.00	0.95	0.79	0.69	0.68	0.70	0.72
$\alpha = 0.4$	1.00	0.91	0.00	0.43	**1.03**	1.00	0.96	0.83	0.74	0.75	0.77	0.78
$\alpha = 0.6$	**1.00**	0.98	0.00	0.41	**1.00**	**1.00**	0.97	0.84	0.78	0.78	0.80	0.82
$\alpha = 0.8$	1.00	**1.02**	0.00	0.30	0.97	1.00	0.98	0.85	0.79	0.80	0.81	0.83
$\alpha = 1.0$	1.00	**1.01**	0.00	0.23	0.94	1.00	**1.01**	0.88	0.81	0.82	0.84	0.86

These experimental results in Table 5-8, however, are somewhat misleading. In Fig. 6 (a), we show all solutions in the final generation of a single run of MOEA/D on the 2-500 problem for the case of the PBI function with $\theta = -0.1$. We can see that no solutions were obtained around the center of the Pareto front. Such a distribution of the obtained solutions can be explained by the concave shape of the contour lines of the PBI function with $\theta = -0.1$, which are shown in Fig. 6 (b). Due to the concave contour lines, MOEA/D with the PBI function tends to search for solutions around the edges of the Pareto front even when the weight vector $\boldsymbol{\lambda}$ is specified as (0.5, 0.5). As a result, no solutions were obtained around the center of the Pareto front in Fig. 6 (a).

In Tables 1-8, we used the origin of the objective space as the reference point for hypervolume calculation. Since the origin is far from the Pareto front of each test problem, extreme solutions around the edge of the Pareto front have large effects on hypervolume calculation [22]. This is why the best results were obtained from the PBI function with $\theta = -0.1$ in Tables 5-8. Good results from $\theta = -0.1$ also suggest the importance of diversity improvement even for many-objective problems [23].



(a) Solutions in the final generation.

(b) Contour lines of the PBI function.

Fig. 6. Experimental results by the PBI function with $\theta = -0.1$ on the 2-500 problem.

In Tables 9-12, we recalculated the normalized average hypervolume using another reference point closer to the Pareto front for each test problem. For each test problem, we used (15000, 15000, ..., 15000) as the reference point for hypervolume calculation. This setting of hypervolume calculation increases the importance of the convergence of solutions around the center of the Pareto front. Thus the normalized average hypervolume value by the PBI function with $\theta = -0.1$ was deteriorated for almost all test problems in Tables 9-12 (e.g., see the last row in Table 9 on the 2-500 problem).

Table 9. Normalized average hypervolume on 4-500 (reference point: 15000).

Problem (4-500)	Weighted sum	Tcheby-cheff	PBI (θ)									
			−1	−0.5	−0.1	0	0.1	0.5	1	5	10	50
$\alpha = 0.0$	**1.00**	0.77	0.00	0.07	0.92	**1.00**	0.98	0.63	0.52	0.92	0.91	0.85
$\alpha = 0.2$	**1.00**	0.84	0.00	0.26	**1.00**	**1.00**	0.94	0.61	0.53	0.88	0.91	0.90
$\alpha = 0.4$	1.00	0.91	0.00	0.37	**1.02**	1.00	0.94	0.65	0.60	0.91	0.94	0.95
$\alpha = 0.6$	1.00	0.93	0.00	0.28	**1.02**	1.00	0.95	0.68	0.65	0.95	0.99	1.01
$\alpha = 0.8$	1.00	0.97	0.00	0.18	1.00	1.00	0.95	0.70	0.67	0.96	1.00	**1.03**
$\alpha = 1.0$	1.00	1.06	0.00	0.09	0.86	1.00	1.03	0.79	0.75	1.06	1.11	**1.14**
2-500	1.00	**1.04**	0.03	0.48	0.95	1.00	0.97	0.79	0.91	**1.04**	1.02	0.94

Table 10. Normalized average hypervolume on 6-500 (reference point: 15000).

Problem (6-500)	Weighted sum	Tcheby-cheff	PBI (θ)									
			−1	−0.5	−0.1	0	0.1	0.5	1	5	10	50
$\alpha = 0.0$	1.00	0.42	0.00	0.00	0.76	1.00	**1.10**	0.78	0.38	0.66	0.69	0.71
$\alpha = 0.2$	**1.00**	0.53	0.00	0.02	0.91	**1.00**	0.99	0.64	0.37	0.59	0.63	0.65
$\alpha = 0.4$	**1.00**	0.70	0.00	0.08	0.95	**1.00**	0.97	0.66	0.46	0.70	0.76	0.79
$\alpha = 0.6$	**1.00**	0.82	0.00	0.06	0.94	**1.00**	0.98	0.71	0.53	0.78	0.83	0.88
$\alpha = 0.8$	1.00	1.02	0.00	0.02	0.76	1.01	**1.09**	0.81	0.65	0.89	0.95	1.01
$\alpha = 1.0$	1.00	1.13	0.00	0.00	0.72	1.00	1.25	1.03	0.84	1.14	1.23	**1.31**

Table 11. Normalized average hypervolume on 8-500 (reference point: 15000).

Problem (8-500)	Weighted sum	Tcheby-cheff	PBI (θ)									
			-1	-0.5	-0.1	0	0.1	0.5	1	5	10	50
$\alpha = 0.0$	1.00	0.67	0.00	0.00	0.51	1.01	**1.35**	1.21	0.58	0.66	0.77	0.83
$\alpha = 0.2$	1.00	0.58	0.00	0.00	0.72	1.00	**1.11**	0.81	0.43	0.51	0.57	0.62
$\alpha = 0.4$	1.00	0.70	0.00	0.00	0.80	1.00	**1.05**	0.76	0.47	0.57	0.63	0.68
$\alpha = 0.6$	1.00	0.97	0.00	0.00	0.73	0.99	**1.17**	0.94	0.65	0.77	0.86	0.92
$\alpha = 0.8$	1.00	**1.37**	0.00	0.00	0.63	1.00	1.32	1.29	0.95	1.13	1.24	1.31
$\alpha = 1.0$	1.00	1.37	0.00	0.00	0.45	0.99	1.55	1.80	1.43	1.66	1.81	**1.90**

Table 12. Normalized average hypervolume on 10-500 (reference point: 15000).

Problem (10-500)	Weighted sum	Tcheby-cheff	PBI (θ)									
			-1	-0.5	-0.1	0	0.1	0.5	1	5	10	50
$\alpha = 0.0$	1.00	0.69	0.00	0.00	0.39	1.01	**1.47**	1.39	0.59	0.55	0.68	0.78
$\alpha = 0.2$	1.00	0.57	0.00	0.00	0.64	1.00	**1.15**	0.82	0.38	0.37	0.44	0.51
$\alpha = 0.4$	1.00	0.65	0.00	0.00	0.73	0.98	**1.06**	0.72	0.41	0.43	0.49	0.55
$\alpha = 0.6$	1.00	0.93	0.00	0.00	0.68	0.99	**1.19**	0.94	0.62	0.66	0.74	0.81
$\alpha = 0.8$	1.00	1.37	0.00	0.00	0.51	0.98	**1.38**	**1.38**	0.99	1.05	1.16	1.25
$\alpha = 1.0$	1.00	1.46	0.00	0.00	0.33	1.00	1.77	**2.31**	1.75	1.89	2.02	2.19

6 Conclusions

In this paper, we examined the choice of a scalarizing function in MOEA/D for many-objective knapsack problems. Good results were obtained from the weighted sum over various settings of test problems. With respect to the specification of the penalty parameter in the PBI function, we obtained the following interesting observations:

(i) The best hypervolume values were obtained from a small negative parameter value (i.e., $\theta = -0.1$) when a reference point was far from the Pareto front. In this case, the search of MOEA/D was biased towards the edges of the Pareto front.

(ii) When a reference point was close to the Pareto front, the best hypervolume values were obtained from a small positive parameter value (i.e., $\theta = 0.1$).

(iii) Almost the same results were obtained from $\theta = 0$ and the weighed sum.

References

1. Zhang, Q., Li, H.: MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. IEEE Trans. on Evolutionary Computation 11 (2007) 712-731
2. Chang, P. C., Chen, S. H., Zhang, Q., Lin, J. L.: MOEA/D for Flowshop Scheduling Problems. Proc. of IEEE Congress on Evolutionary Computation (2008) 1433-1438
3. Zhang, Q., Liu, W., Li, H.: The Performance of a New Version of MOEA/D on CEC09 Unconstrained MOP Test Instances. Proc. of IEEE Congress on Evolutionary Computation (2009) 203-208

4. Li, H., Zhang, Q.: Multiobjective Optimization Problems with Complicated Pareto Sets, MOEA/D and NSGA-II. IEEE Trans. on Evolutionary Computation 13 (2009) 284-302
5. Ishibuchi, H., Sakane, Y., Tsukamoto, N., Nojima, Y.: Evolutionary Many-Objective Optimization by NSGA-II and MOEA/D with Large Populations. Proc. of IEEE International Conference on Systems, Man, and Cybernetics (2009) 1820-1825
6. Zhang, Q., Liu, W., Tsang, E., Virginas, B.: Expensive Multiobjective Optimization by MOEA/D With Gaussian Process Model. IEEE Trans. on Evolutionary Computation 14 (2010) 456 - 474
7. Tan, Y. Y., Jiao, Y. C., Li, H., Wang, X. K.: A Modification to MOEA/D-DE for Multiobjective Optimization Problems with Complicated Pareto Sets. Information Sciences (2012) 14-38
8. Zhao, S. Z., Suganthan, P. N., Zhang, Q.: Decomposition-Based Multiobjective Evolutionary Algorithm with an Ensemble of Neighborhood Sizes. IEEE Trans. on Evolutionary Computation 16 (2012) 442-446
9. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. IEEE Trans. on Evolutionary Computation 6 (2002) 182-197
10. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the Strength Pareto Evolutionary Algorithm. TIK-Report 103, Department of Electrical Engineering, ETH, Zurich (2001)
11. Ishibuchi, H., Sakane, Y., Tsukamoto, N., Nojima, Y.: Adaptation of Scalarizing Functions in MOEA/D: An Adaptive Scalarizing Function-Based Multiobjective Evolutionary Algorithm. Lecture Notes in Computer Science, Vol. 5467: EMO 2009 (2009) 438-452
12. Ishibuchi, H., Sakane, Y., Tsukamoto, N., Nojima, Y.: Simultaneous Use of Different Scalarizing Functions in MOEA/D. Proc. of Genetic and Evolutionary Computation Conference (2010) 519-526
13. Zitzler, E., Thiele, L.: Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. IEEE Trans. on Evolutionary Computation 3 (1999) 257-271
14. Ishibuchi, H., Hitotsuyanagi, Y., Ohyanagi, H., Nojima, Y. Effects of the Existence of Highly Correlated Objectives on the Behavior of MOEA/D. Lecture Notes in Computer Science, Vol. 6576: EMO 2011, Springer, Berlin (2011) 166-181
15. Miettinen, K.: Nonlinear Multiobjective Optimization. Kluwer, Dordrecht (1999)
16. Hughes, E. J.: Evolutionary Many-Objective Optimisation: Many Once or One Many?. Proc. of IEEE Congress on Evolutionary Computation (2005) 222-227
17. Purshouse, R. C., Fleming, P. J.: On the Evolutionary Optimization of Many Conflicting Objectives. IEEE Trans. on Evolutionary Computation 11 (2007) 770-784
18. Sato, H., Aguirre, H. E., Tanaka, K.: Local Dominance and Local Recombination in MOEAs on 0/1 Multiobjective Knapsack Problems. European J. of Operational Research 181 (2007) 1708-1723
19. Ishibuchi, H., Tsukamoto, N., Hitotsuyanagi, Y., Nojima, Y.: Effectiveness of Scalability Improvement Attempts on the Performance of NSGA-II for Many-Objective Problems. Proc. of Genetic and Evolutionary Computation Conference (2008) 649-656
20. Ishibuchi, H., Tsukamoto, N., Nojima, Y.: Evolutionary Many-Objective Optimization: A Short Review. Proc. of IEEE Congress on Evolutionary Computation (2008) 2424-2431
21. Kowatari, N., Oyama, A., Aguirre, H., Tanaka, K.: Analysis on Population Size and Neighborhood Recombination on Many-Objective Optimization. Lecture Notes in Computer Science, Vol. 7492: PPSN XII, Part II, Springer, Berlin (2012) 22-31
22. Ishibuchi, H., Nojima, Y., Doi, T.: Comparison between Single-objective and Multi-objective Genetic Algorithms: Performance Comparison and Performance Measures. Proc. of IEEE Congress on Evolutionary Computation (2006) 3959-3966
23. Ishibuchi, H., Tsukamoto, N., Nojima, Y.: Diversity Improvement by Non-Geometric Binary Crossover in Evolutionary Multiobjective Optimization. IEEE Trans. on Evolutionary Computation 14 (2010) 985-998