H. Ishibuchi, Y. Hitotsuyanagi, H. Ohyanagi, and Y. Nojima, “Effects of the Existence of Highly Correlated Objectives on the Behavior of MOEA/D,” Lecture Notes in Computer Science 6576: Evolutionary Multi-Criterion Optimization - EMO 2011, pp. 166-181, Ouro Preto, Brazil, April 5-8, 2011 (Proceedings of EMO 2011).

Effects of the Existence of Highly Correlated Objectives on the Behavior of MOEA/D

Hisao Ishibuchi, Yasuhiro Hitotsuyanagi, Hiroyuki Ohyanagi, and Yusuke Nojima

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

Abstract. Recently MOEA/D (multi-objective evolutionary algorithm based on decomposition) was proposed as a high-performance EMO (evolutionary multi-objective optimization) algorithm. MOEA/D has high search ability as well as high computational efficiency. Whereas other EMO algorithms usually do not work well on many-objective problems with four or more objectives, MOEA/D can properly handle them. This is because its scalarizing function-based fitness evaluation scheme can generate an appropriate selection pressure toward the Pareto front without severely increasing the computation load. MOEA/D can also search for well-distributed solutions along the Pareto front using a number of weight vectors with different directions in scalarizing functions. Currently MOEA/D seems to be one of the best choices for multi-objective optimization in various application fields. In this paper, we examine its performance on multi-objective problems with highly correlated objectives. Similar objectives to existing ones are added to two-objective test problems in computational experiments. Experimental results on multi-objective knapsack problems show that the inclusion of similar objectives severely degrades the performance of MOEA/D while it has almost no negative effects on NSGA-II and SPEA2. We also visually examine such an undesirable behavior of MOEA/D using many-objective test problems with two decision variables.

Keywords: Evolutionary multi-objective optimization, evolutionary many-objective optimization, similar objectives, correlated objectives, MOEA/D.

1 Introduction

Since Goldberg’s suggestion in 1989 [6], Pareto dominance-based fitness evaluation has been the main stream in the evolutionary multi-objective optimization (EMO) community [3], [26]. Pareto dominance is used for fitness evaluation in almost all well-known and frequently-used EMO algorithms such as NSGA-II [4], SPEA [34] and SPEA2 [33]. Whereas Pareto dominance-based EMO algorithms usually work very well on multi-objective problems with two or three objectives, they often show difficulties in the handling of many-objective problems with four or more objectives as pointed out in several studies [7], [10], [16], [23], [24], [35]. This is because almost all individuals in the current population are non-dominated with each other when they are compared using many objectives. As a result, Pareto dominance-based fitness evaluation cannot generate strong selection pressure toward the Pareto front. This means that good solutions close to the Pareto front are not likely to be obtained.

Various approaches have been proposed to improve the search ability of Pareto dominance-based EMO algorithms for many-objective problems [13], [14]. The basic idea of those approaches is to increase the selection pressure toward the Pareto front. The increase in the selection pressure, however, usually leads to the decrease in the diversity of obtained solutions along the Pareto front. Thus simultaneous performance improvement in both the convergence and the diversity is not easy.

The use of other fitness evaluation schemes has also been examined for many-objective problems. One promising approach to many-objective optimization is the use of an indicator function that measures the quality of a solution set [1], [27], [28], [31], [32]. Hypervolume has been frequently used in such an indicator-based evolutionary algorithm (IBEA) where multi-objective problems are handled as single-objective hypervolume maximization problems. One difficulty of this approach is the exponential increase in the computation load for hypervolume calculation with the increase in the number of objectives. Thus some form of approximate hypervolume calculation may be needed when we have six or more objectives. Another promising approach to many-objective problems is the use of scalarizing functions [8], [15], [19], [29]. A number of scalarizing functions with different weight vectors are used to realize various search directions in the objective space. The main advantage of this approach is computational efficiency of scalarizing function calculation.

MOEA/D (multi-objective evolutionary algorithm based on decomposition) is a scalarizing function-based EMO algorithm proposed by Li and Zhang [19], [29]. High search ability of MOEA/D on various test problems including many-objective problems has already been demonstrated in the literature [2], [11], [12], [17], [20], [22], [30]. Its main feature is the decomposition of a multi-objective problem into a number of single-objective problems, which are defined by a scalarizing function with different weight vectors. MOEA/D can be viewed as a kind of cellular algorithm. Each cell has a different weight vector and a single elite solution with respect to its own weight vector. The task of each cell is to perform single-objective optimization of a scalarizing function with its own weight vector. To generate a new solution for each cell, parents are selected from its neighboring cells (i.e., local parent selection). If a better solution is generated by genetic operations, the current solution is replaced with the newly generated one. This solution replacement mechanism is applied to not only the current cell for which a new solution is generated but also its neighboring cells. That is, a good solution has a chance to survive at multiple cells. Such a local solution replacement mechanism together with local parent selection accelerates multi-objective search for better solutions (i.e., accelerates the convergence toward the Pareto front). At the same time, the diversity of solutions is maintained by the use of a number of weight vectors with various directions in MOEA/D.

In this paper, we report some interesting observations on the behavior of NSGA-II, SPEA2 and MOEA/D on multi-objective problems with highly correlated objectives. In computational experiments, we generate similar objectives to existing ones and add them to test problems with two objectives. Experimental results show that the inclusion of similar objectives severely deteriorates the search ability of MOEA/D while it has almost no negative effects on NSGA-II and SPEA2. As a result, MOEA/D does not always outperform NSGA-II and SPEA2 on many-objective problems with highly correlated objectives while it clearly shows better performance on many-objective problems with no strong correlations among objectives.

This paper is organized as follows. First we briefly explain MOEA/D in Section 2. Next we examine its behavior on two types of multi-objective knapsack problems in comparison with NSGA-II and SPEA2 in Section 3. One type has randomly generated objectives, and the other includes highly correlated objectives. Then we visually examine the behavior of MOEA/D using many-objective test problems in the two-dimensional decision space in Section 4. Finally we conclude this paper in Section 5.

2 MOEA/D

Let us consider the following *m*-objective maximization problem:

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

where $\mathbf{f}(\mathbf{x})$ is the *m*-dimensional objective vector, $f_i(\mathbf{x})$ is the *i*-th objective to be maximized, and $\mathbf{x}$ is the decision vector.

In MOEA/D [19], [29], a multi-objective problem is decomposed into a number of single-objective problems where each problem is to optimize a scalarizing function with a different weight vector. In this paper, we use the weighted Tchebycheff function since this function works very well on a wide range of multi-objective test problems [9], [11], [12]. Let us denote a weight vector as $\boldsymbol{\lambda} = (\lambda_1,\ \lambda_2,\ ...,\ \lambda_m)$. The weighted Tchebycheff function measures the distance from the reference point $\mathbf{z}^*$ to a solution $\mathbf{x}$ in the objective space as follows:

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

For multi-objective knapsack problems, we use the following specification of the reference point $\mathbf{z}^*$ in the same manner as in Zhang and Li [29]:

$$z_i^* = 1.1 \cdot \max\{f_i(\mathbf{x})\,|\,\mathbf{x} \in \Omega(t)\},\ \ i = 1,\ 2,\ ...,\ m\ , \tag{3}$$

where $\Omega(t)$ shows the population at the *t*-th generation. The reference point $\mathbf{z}^*$ is updated whenever the maximum value of each objective in (3) is updated.

For multi-objective function minimization problems, Zhang and Li [29] specified the reference point $\mathbf{z}^*$ as follows:

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

We use this specification for multi-objective continuous minimization problems.

MOEA/D uses a set of weight vectors placed on a uniform grid. More specifically, it uses all weight vectors satisfying the following two conditions:

$$\lambda_1 + \lambda_2 + \cdots + \lambda_m = 1, \tag{5}$$

$$\lambda_i \in \left\{0, \frac{1}{H}, \frac{2}{H}, ..., \frac{H}{H}\right\},\ i = 1, 2, ..., m, \tag{6}$$

where H is a positive integer parameter that specifies the granularity or resolution of weight vectors. The number of weight vectors is calculated from this parameter H and the number of objectives m as $N = {}_{H+m-1}C_{m-1}$ [29]. The same weight vector specification was used in a multi-objective cellular algorithm in Murata et al. [21].

Let N be the number of weight vectors. Then a multi-objective optimization problem is decomposed into N single-objective problems in MOEA/D. Each single-objective problem has the same scalarizing function (i.e., the weighted Tchebycheff function in this paper) with a different weight vector. Each weight vector can be viewed as a cell in a cellular algorithm with a grid of size N in the m-dimensional unit cube $[0, 1]^m$. A single individual is assigned to each cell. Thus the population size is the same as the number of weight vectors. In MOEA/D, genetic operations at each cell are locally performed within its neighboring cells as in cellular algorithms. For each cell, a pre-specified number of its nearest cells (e.g., ten cells including the cell itself in our computational experiments) are handled as its neighbors. Neighborhood structures in MOEA/D are defined by the Euclidean distance between weight vectors.

First MOEA/D generates an initial solution at each cell. In our computational experiments, initial solutions are randomly generated. Next an offspring is generated by local selection, crossover and mutation at each cell in an unsynchronized manner. In local selection, two parents are randomly chosen for the current cell from its neighboring cells (including the current cell itself). Local selection leads to the recombination of similar parents in the objective space. The generated offspring is compared with the solution at each of the neighboring cells. The comparison is performed based on the scalarizing function with the weight vector of the compared neighbor. All the inferior solutions are replaced with the newly generated offspring. That is, solution replacement is performed not only at the current cell for which the new offspring is generated but also at each of its neighboring cells. Since each cell has a different weight vector, the diversity of solutions can be maintained whereas a single offspring is compared with multiple neighbors for solution replacement. Local selection, crossover, mutation and local replacement are performed at each cell. These procedures are iterated over all cells until the termination condition is satisfied. In our computational experiments, we do not use any secondary population in MOEA/D.

3 Computational Experiments on Knapsack Problems

We used the same test problem as the two-objective 500-item knapsack problem of Zitzler & Thiele [34] with two constraint conditions. We denote this test problem as the 2-500 problem. The two objectives $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ of the 2-500 problem were generated by randomly assigning an integer in the closed interval [10, 100] to each item as its profit (see [34]). In the same manner, we generated other two objectives

$f_3(\mathbf{x})$ and $f_4(\mathbf{x})$. Since all the four objectives were randomly generated, they have no strong correlation with each other. Using these randomly-generated four objectives, we generated a four-objective 500-item knapsack problem with the same two constraint conditions as in the 2-500 problem in [34]. Exactly the same two constraint conditions as in [34] were also used in all the other test problems in this section.

We generated highly correlated objectives from the two objectives $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ of the 2-500 problem in the following manner:

$$f_5(\mathbf{x}) = f_1(\mathbf{x}) + 0.01 f_2(\mathbf{x}) , \tag{7}$$

$$f_6(\mathbf{x}) = f_1(\mathbf{x}) - 0.01 f_2(\mathbf{x}) , \tag{8}$$

$$f_7(\mathbf{x}) = f_2(\mathbf{x}) + 0.01 f_1(\mathbf{x}) , \tag{9}$$

$$f_8(\mathbf{x}) = f_2(\mathbf{x}) - 0.01 f_1(\mathbf{x}) . \tag{10}$$

It is clear that $f_5(\mathbf{x})$ and $f_6(\mathbf{x})$ are similar to $f_1(\mathbf{x})$ while $f_7(\mathbf{x})$ and $f_8(\mathbf{x})$ are similar to $f_2(\mathbf{x})$. In computational experiments, we used the following four test problems:

1. The 2-500 test problem with $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ of Zitzler & Thiele [34],
2. Random four-objective problem with $f_1(\mathbf{x})$, $f_2(\mathbf{x})$, $f_3(\mathbf{x})$ and $f_4(\mathbf{x})$,
3. Correlated four-objective problem with $f_1(\mathbf{x})$, $f_2(\mathbf{x})$, $f_5(\mathbf{x})$ and $f_6(\mathbf{x})$,
4. Correlated six-objective problem with $f_1(\mathbf{x})$, $f_2(\mathbf{x})$, $f_5(\mathbf{x})$, $f_6(\mathbf{x})$, $f_7(\mathbf{x})$ and $f_8(\mathbf{x})$.

In the correlated four-objective problem, $f_1(\mathbf{x})$, $f_5(\mathbf{x})$ and $f_6(\mathbf{x})$ are similar to each other while they are not similar to $f_2(\mathbf{x})$. In the correlated six-objective problem, $f_2(\mathbf{x})$, $f_7(\mathbf{x})$ and $f_8(\mathbf{x})$ are also similar to each other.

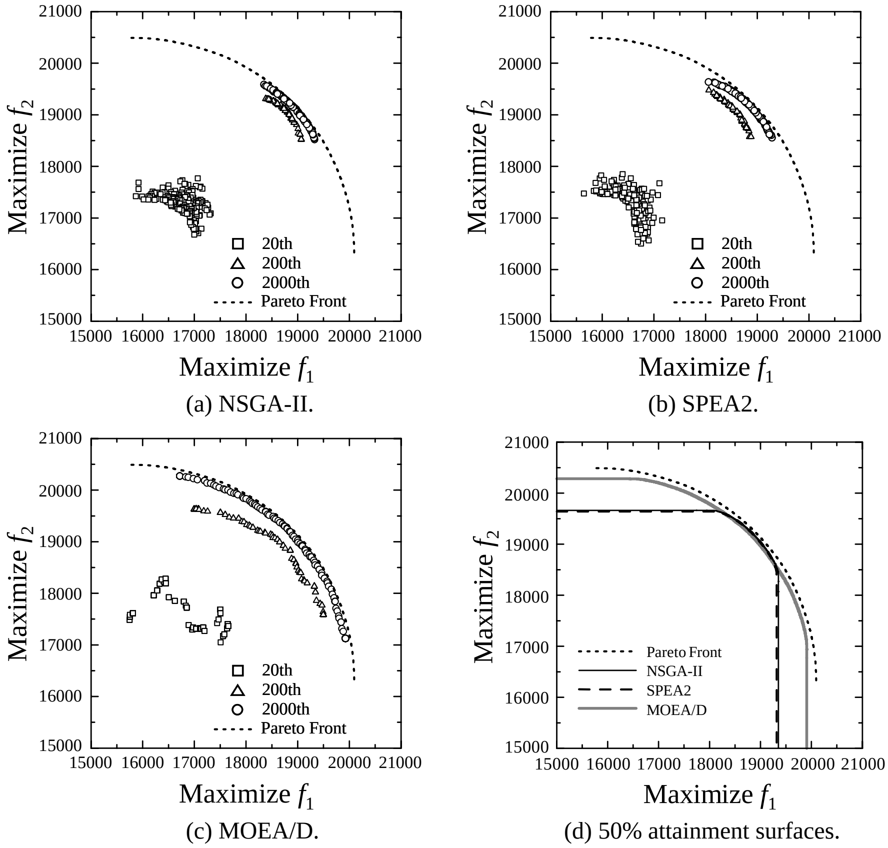
We applied MOEA/D to these four test problems using the following setting:

Population size (which is the same as the number of weight vectors):
200 (two-objective problem), 220 (four-objective), 252 (six-objective),
Parameter H for generating weight vectors:
199 (two-objective problem), 9 (four-objective), 5 (six-objective),
Coding: Binary string of length 500,
Stopping condition: 400,000 solution evaluations,
Crossover probability: 0.8 (Uniform crossover),
Mutation probability: 1/500 (Bit-flip mutation),
Constraint handling: Greedy repair used in Zitzler & Thiele [34],
Neighborhood size T (i.e., the number of neighbors): 10.

Since all the four test problems have the same constraint conditions, the same greedy repair as in the 2-500 problem in [34] was used in all test problems.

We also applied NSGA-II [4] and SPEA2 [33] to the four test problems using the same setting as in MOEA/D except that the population size was always specified as 200 in NSGA-II and SPEA2. Each EMO algorithm was applied to each test problem 100 times. In this section, we report experimental results by NSGA-II, SPEA2 and MOEA/D on each of the above-mentioned four test problems.

Results on the 2-500 Knapsack Problem: Experimental results of a single run of each algorithm are shown in Fig. 1 (a)-(c) where all solutions at the 20th, 200th and 2000th generations are depicted together with the true Pareto front. The 50% attainment surface [5] at the 2000th generation over 100 runs of each algorithm is depicted in Fig. 1 (d). As pointed out in the literature (e.g., see Jaszkiewicz [15]), it is not easy for EMO algorithms to find non-dominated solutions along the entire Pareto front of the 2-500 test problem. NSGA-II and SPEA2 found non-dominated solutions around the center of the Pareto front. Only MOEA/D found non-dominated solutions over almost the entire Pareto front. That is, MOEA/D found better solution sets than NSGA-II and SPEA2 with respect to the diversity of solutions along the Pareto front. With respect to the convergence property toward the Pareto front, the three algorithms have almost the same performance in Fig. 1 (d).



(a) NSGA-II. (b) SPEA2.

(c) MOEA/D. (d) 50% attainment surfaces.

Fig. 1. Experimental results on the original two-objective knapsack problem.

Results on Random Four-Objective Knapsack Problem: We calculated the average hypervolume and the standard deviation over 100 runs of each algorithm. The origin (0, 0, 0, 0) of the four-dimensional (4-D) objective space was used as the reference point for the hypervolume calculation. The following results were obtained:

NSGA-II: 1.23×10^{17} (Average), 9.67×10^{14} (Standard Deviation),
SPEA2: 1.19×10^{17} (Average), 9.47×10^{14} (Standard Deviation),
MOEA/D: 1.43×10^{17} (Average), 6.00×10^{14} (Standard Deviation).

The best results were obtained by MOEA/D for the random four-objective problem with respect to the hypervolume measure. In Fig. 2 (a)-(c), we show all solutions at the final generation in a single run of each algorithm in the two-dimensional (2-D) objective space with $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$. That is, each plot in Fig. 2 shows the projection of the final population of each algorithm in the 4-D objective space onto the 2-D objective space. The 50% attainment surface is depicted using those projections for 100 runs in Fig. 2 (d). For comparison, the Pareto front of the 2-500 problem is also shown in Fig. 2. In Fig. 2 (d), the convergence performance of NSGA-II and SPEA2 was severely degraded by the inclusion of the randomly generated objectives $f_3(\mathbf{x})$ and $f_4(\mathbf{x})$. The convergence performance of MOEA/D was also degraded but less severely than NSGA-II and SPEA2. It is interesting to observe that NSGA-II in Fig. 2 (a) and SPEA2 in Fig. 2 (b) did not have large diversity in the objective space even in the case of the four-objective test problem with the randomly generated objectives.

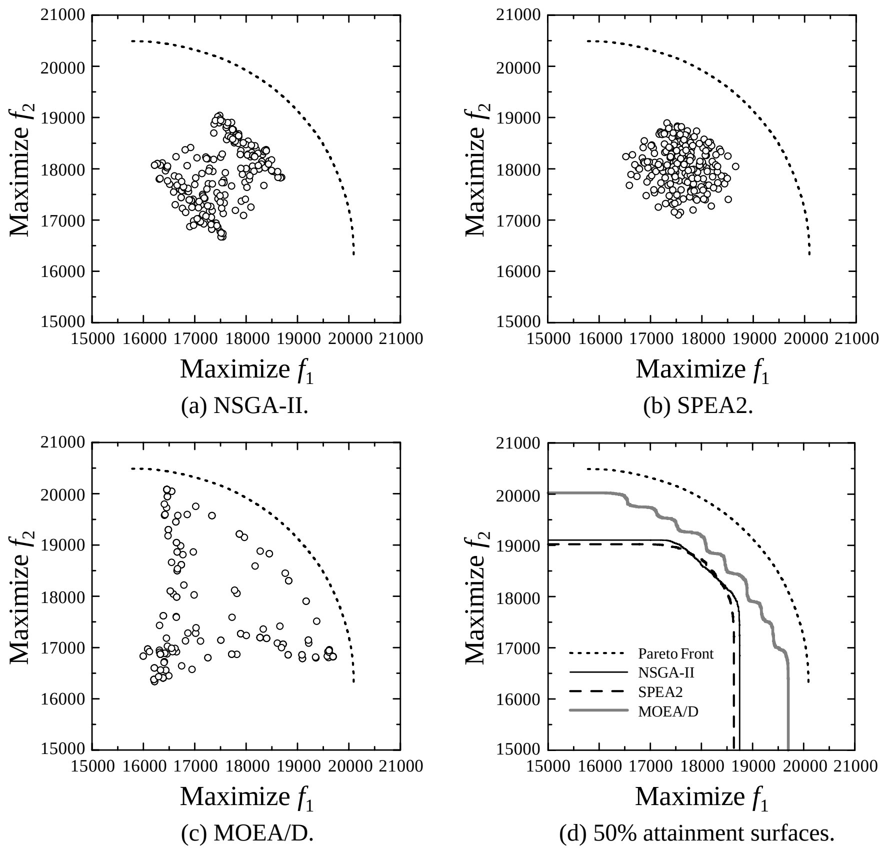


Fig. 2. Experimental results on the random four-objective knapsack problem (Projections of the final population in the 4-D objective space onto the 2-D one).

Results on Correlated Four-Objective Knapsack Problem: This problem has the three highly correlated objectives $f_1(\mathbf{x})$, $f_5(\mathbf{x})$ and $f_6(\mathbf{x})$. As in the previous computational experiments, we calculated the average hypervolume over 100 runs:

NSGA-II: 1.42×10^{17} (Average), 1.59×10^{15} (Standard Deviation),
SPEA2: 1.41×10^{17} (Average), 1.32×10^{15} (Standard Deviation),
MOEA/D: 1.55×10^{17} (Average), 7.46×10^{14} (Standard Deviation).

The best average result was obtained by MOEA/D. In the same manner as Fig. 2, we show experimental results on the correlated four-objective problem in Fig. 3. From the comparison between Fig. 1 and Fig. 3, we can see that the inclusion of $f_5(\mathbf{x})$ and $f_6(\mathbf{x})$ had almost no negative effects on the performance of NSGA-II and SPEA2. The performance of MOEA/D, however, was clearly degraded by their inclusion. We can also see that many solutions in Fig. 3 (c) are overlapping, which leads to the wavy 50% attainment surface by MOEA/D in Fig. 3 (d). In spite of the performance deterioration, the largest average hypervolume was still obtained by MOEA/D.

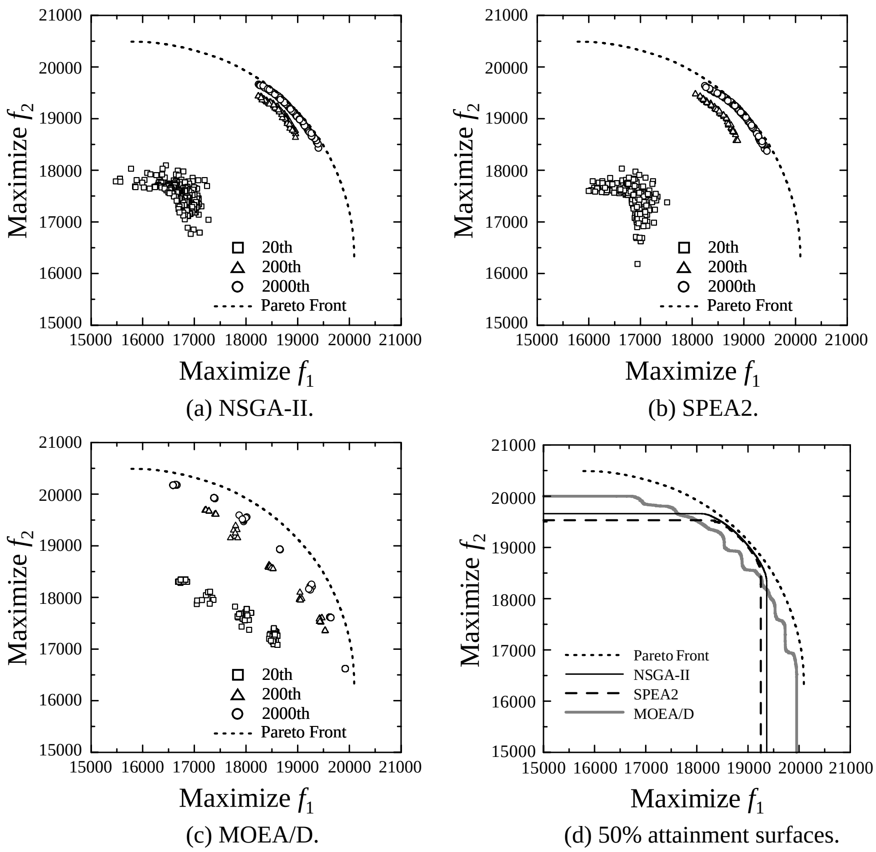
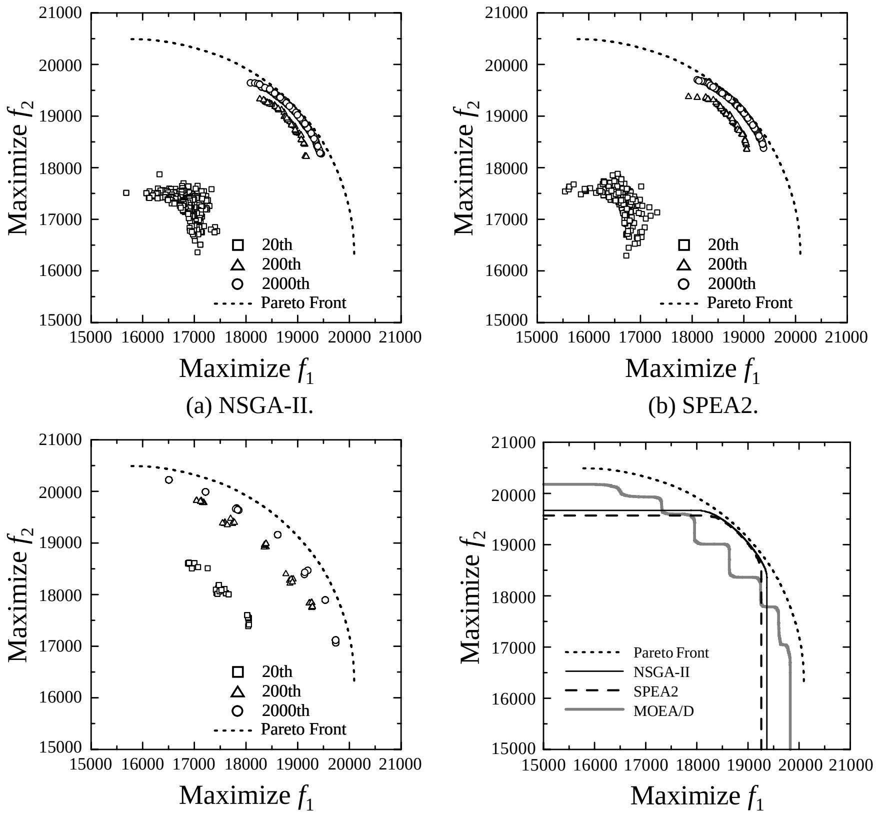


Fig. 3. Experimental results on the correlated four-objective knapsack problem (Projections from the 4-D objective space to the 2-D one). The number of generations of MOEA/D with population size 220 was converted to the equivalent one in the case of population size 200.

Results on Correlated Six-Objective Knapsack Problem: This problem has the two sets of the three highly correlated objectives: $\{f_1(\mathbf{x}), f_5(\mathbf{x}), f_6(\mathbf{x})\}$ and $\{f_2(\mathbf{x}), f_7(\mathbf{x}), f_8(\mathbf{x})\}$. We calculated the average hypervolume over 100 runs:

NSGA-II: 5.46×10^{25} (Average), 5.69×10^{23} (Standard Deviation),
SPEA2: 5.40×10^{25} (Average), 4.59×10^{23} (Standard Deviation),
MOEA/D: 5.87×10^{25} (Average), 4.72×10^{23} (Standard Deviation).

As in the other three test problems, the best results were obtained by MOEA/D with respect to the hypervolume measure. In the same manner as Fig. 2 and Fig. 3, we show experimental results in the two-dimensional (2-D) objective space in Fig. 4. From the comparison between Fig. 1 and Fig. 4, we can see that the inclusion of the four correlated objectives had almost no negative effects on the performance of NSGA-II and SPEA2. The performance of MOEA/D, however, was clearly degraded by their inclusion. That is, the convergence performance was degraded and the number of obtained solutions was decreased (see Fig. 1 (c), Fig. 3 (c) and Fig. 4 (c)).



(a) NSGA-II. (b) SPEA2. (c) MOEA/D. (d) 50% attainment surfaces.

Fig. 4. Experimental results on the correlated six-objective knapsack problem (Projections from the 6-D objective space to the 2-D one). The number of generations of MOEA/D with population size 252 was converted to the equivalent one in the case of population size 200.

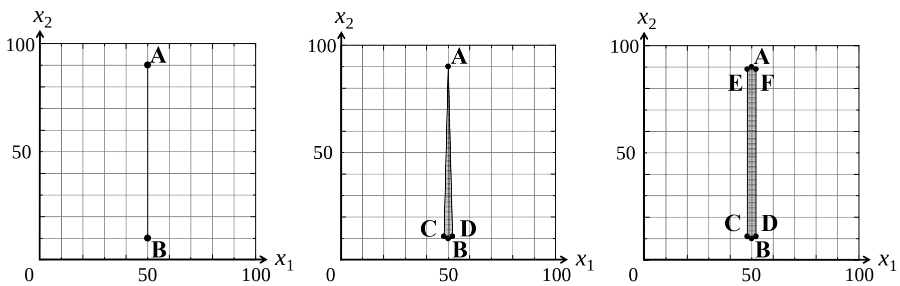
As in Figs. 2-4, we projected the solution set obtained by each run onto the 2-D objective space with $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$. Then we calculated its hypervolume in the 2-D objective space using the origin (0, 0) as the reference point. Table 1 summarizes the average result over 100 runs of each algorithm on each test problem. When we included the two randomly generated objectives into the 2-500 knapsack problem, the performance of NSGA-II and SPEA2 was severely degraded (see the row labeled as "Random 4-Obj." in Table 1). However, the inclusion of the two and four correlated objectives did not degrade their performance at all. MOEA/D shows a totally different behavior from the others. The performance of MOEA/D was clearly degraded by the inclusion of the correlated objectives as well as the randomly generated objectives. In spite of the performance deterioration, the best results were obtained by MOEA/D for all the four problems in Table 1.

Table 1. Average hypervolume and standard deviation in the original two-objective space.

Problem	NSGA-II		SPEA2		MOEA/D	
	Average	Stand. Dev.	Average	Stand. Dev.	Average	Stand. Dev.
Original 2-Obj.	3.800E+08	1.764E+06	3.790E+08	1.390E+06	4.005E+08	9.291E+05
Random 4-Obj.	3.577E+08	2.184E+06	3.537E+08	1.862E+06	3.902E+08	1.435E+06
Correlated 4-Obj.	3.800E+08	1.618E+06	3.790E+08	1.255E+06	3.949E+08	1.491E+06
Correlated 6-Obj.	3.804E+08	1.483E+06	3.782E+08	5.481E+06	3.947E+08	1.194E+06

4 Computational Experiments on Two-Dimensional Problems

In this section, we visually examine the behavior of EMO algorithms using test problems with only two decision variables. In our test problems, the distance to each of the given points in the two-dimensional decision space $[0, 100] \times [0, 100]$ is minimized. In Fig. 5, we show our three test problems used in this section.



(a) Two-objective problem. (b) Four-objective problem. (c) Six-objective problem.

Fig. 5. Three test problems used in Section 4.

For example, let us assume that four points A, B, C and D are given as in Fig. 5 (b). In this case, our four-objective test problem is written as follows:

$$\text{Minimize } \textit{distance}(\text{A}, \mathbf{x}),\ \textit{distance}(\text{B}, \mathbf{x}),\ \textit{distance}(\text{C}, \mathbf{x}) \text{ and } \textit{distance}(\text{D}, \mathbf{x}), \tag{11}$$

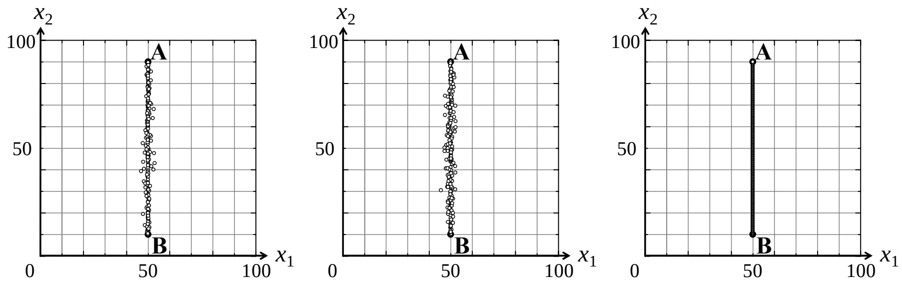
where *distance*(A, **x**) shows the Euclidean distance between the point A and a two-dimensional decision vector **x** in the decision space $[0, 100] \times [0, 100]$.

As shown in this formulation, the number of objectives is the same as the number of the given points. Thus we can generate various test problems with an arbitrary number of objectives. As in Fig. 5, we can also generate highly correlated objectives using closely located points. Regular polygons were used to generate this type of test problems in [18], [25]. Multiple polygons were used to generate test problems with multiple equivalent Pareto regions and/or disjoint Pareto regions in [9].

We applied NSGA-II, SPEA2 and MOEA/D to our three test problems in Fig. 5 using the following setting:

Population size in NSGA-II and SPEA2: 200
Population size in MOEA/D: 200 (2-objective), 220 (4-objective), 252 (6-objective),
Parameter H for generating weight vectors in MOEA/D:
199 (2-objective), 9 (4-objective), 5 (6-objective),
Coding: Real number string of length 2,
Stopping condition: 400,000 solution evaluations,
Crossover probability: 1.0 (SBX with $\eta_c = 15$),
Mutation probability: 0.5 (Polynomial mutation with $\eta_m = 20$),
Neighborhood size T (i.e., the number of neighbors): 10.

Experimental results of a single run of each algorithm on the two-objective test problem in Fig. 5 (a) are shown in Fig. 6. All solutions at the final generation are shown in each plot in Fig. 6 where Pareto optimal solutions are points on the line between the points A and B. In Fig. 6 (a) and Fig. 6 (b), the final populations included many sub-optimal solutions that are not on the line between the points A and B. Much better results with respect to the convergence to the Pareto front were obtained by MOEA/D in Fig. 6 (c) where all solutions are on the line between the points A and B.

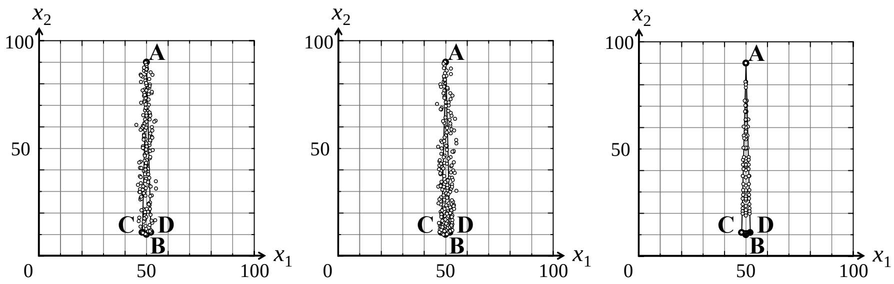


(a) NSGA-II. (b) SPEA2. (c) MOEA/D.

Fig. 6. Experimental results on the two-objective test problem in Fig. 5 (a).

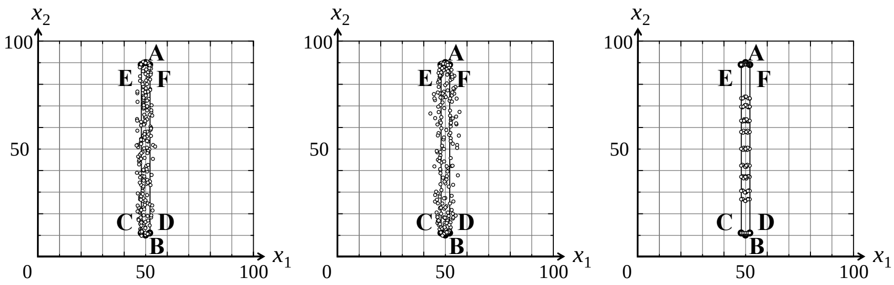
Fig. 7 shows experimental results on the four-objective test problem in Fig. 5 (b) where Pareto optimal solutions are points inside the four points A, B, C and D. As in Fig. 6, the best results with respect to the convergence were obtained by MOEA/D in Fig. 7. However, we can observe some regions with no solutions in Fig. 7 (c). That is, obtained solutions in Fig. 7 (c) are not uniformly distributed.

Fig. 8 shows experimental results on the six-objective test problem in Fig. 5 (c) where Pareto optimal solutions are points inside the six points A, B, C, D, E and F.



(a) NSGA-II. (b) SPEA2. (c) MOEA/D.

Fig. 7. Experimental results on the four-objective test problem in Fig. 5 (b).



(a) NSGA-II. (b) SPEA2. (c) MOEA/D.

Fig. 8. Experimental results on the six-objective test problem in Fig. 5 (c).

As in Fig. 6 and Fig. 7, the best results with respect to the convergence were obtained by MOEA/D in Fig. 8. However, we can observe that obtained solutions in Fig. 8 (c) by MOEA/D are not uniformly distributed.

We calculated the average hypervolume over 100 runs. The reference point for hypervolume calculation was specified as $1.1 \times$ (the maximum objective value for each objective among Pareto optimal solutions of each problem), which is $1.1 \times$ (the distance from each point to its farthest point). Experimental results are summarized in Table 2. For the two-objective problem, the best results were obtained by MOEA/D in Table 2, which is consistent with Fig. 6. The performance of MOEA/D, however, was the worst in Table 2 for the four-objective and six-objective problems. This is due to the existence of regions with no solutions as shown in Fig. 7 (c) and Fig. 8 (c).

Table 2. Average hypervolume and standard deviation for each test problem.

Problem	NSGA-II		SPEA2		MOEA/D	
	Average	Stand. Dev.	Average	Stand. Dev.	Average	Stand. Dev.
Two-Objective	4.521E+03	7.409E-01	4.518E+03	1.616E+00	4.528E+03	3.801E-03
Four-Objective	1.992E+07	1.720E+04	1.987E+07	2.694E+04	1.912E+07	6.626E+02
Six-Objective	3.761E+10	4.322E+07	3.736E+10	6.520E+07	3.227E+10	6.724E+06

We also calculated the average hypervolume in the 2-D objective space with the two objectives defined by the two points A and B. That is, solution sets obtained for the four-objective and six-objective problems were projected onto the two-dimensional (2-D) objective space as in Table 1 in the previous section. Then the average hypervolume were calculated. Experimental results are summarized in Table 3. When the three algorithms were applied to the two-objective problem, the best average value 4.528×10^3 was obtained by MOEA/D. However, this average value was decreased to 4.216×10^3 by 6.89% when MOEA/D was applied to the six-objective problem. In the case of NSGA-II, the decrease in the average hypervolume value was only 0.35% from 4.521×10^3 to 4.505×10^3. The decrease in the case of SPEA2 was also small (i.e., 0.58%). That is, the inclusion of the highly correlated objectives severely degraded the performance of MOEA/D whereas it had almost no negative effects on the other two algorithms.

In Fig. 7 (c) and Fig. 8 (c), solution sets with strange distributions were obtained by MOEA/D for the four-objective and six-objective problems. Such an undesirable behavior disappeared when we decreased the correlation among the similar objectives (i.e., when we increased the distance among the closely located points as shown in Fig. 9). From Fig. 9, we can see that the decrease in the correlation among the objectives leads to better distributions along the line between the two points A and B.

Table 3. Average hypervolume and standard deviation in the 2-D objective space.

Problem	NSGA-II		SPEA2		MOEA/D	
	Average	Stand. Dev.	Average	Stand. Dev.	Average	Stand. Dev.
Two-Objective	4.521E+03	7.409E-01	4.518E+03	1.616E+00	4.528E+03	3.801E-03
Four-Objective	4.504E+03	2.894E+00	4.485E+03	4.198E+00	4.412E+03	1.380E-01
Six-Objective	4.505E+03	2.013E+00	4.492E+03	3.283E+00	4.216E+03	1.216E-01

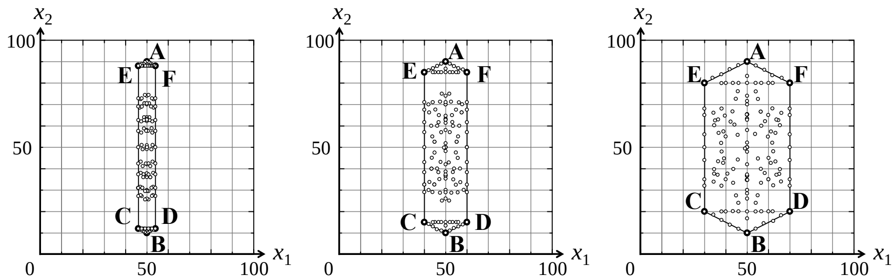


Fig. 9. Experimental results of MOEA/D on other six-objective test problems.

5 Conclusions

We demonstrated that highly correlated objectives severely degraded the performance of MOEA/D whereas they had almost no negative effects on the performance of NSGA-II and SPEA2. The reason for the performance deterioration of MOEA/D may be the use of a uniform grid of weight vectors independent of the correlation among objectives. When an *m*-objective problem has highly correlated objectives, its Pareto front in the objective space has a totally different shape from the uniform grid of weight vectors in the m-dimensional weight vector space $[0, 1]^m$. This leads to a strange distribution of obtained solutions by MOEA/D. Our experimental results clearly suggest the necessity of the adjustment of weight vectors according to the correlation among objectives, which is left as a future research issue. Further discussions on the behavior of each algorithm are also left as a future research issue.

References

1. Beume, N., Naujoks, B., Emmerich, M.: SMS-EMOA: Multiobjective Selection based on Dominated Hypervolume. European Journal of Operational Research 181 (2007) 1653-1669
2. Chang, P. C., Chen, S. H., Zhang, Q., Lin, J. L.: MOEA/D for Flowshop Scheduling Problems. Proc. of 2008 IEEE Congress on Evolutionary Computation (2008) 1433-1438
3. Deb, K.: Multi-Objective Optimization Using Evolutionary Algorithms. John Wiley & Sons, Chichester (2001)
4. 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
5. Fonseca, C. M., Fleming, P. J.: On the Performance Assessment and Comparison of Stochastic Multiobjective Optimizers. Lecture Notes in Computer Science, Vol. 114: PPSN IV. Springer, Berlin (1996) 584-593
6. Goldberg, D. E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, Reasing (1989)
7. Hughes, E. J.: Evolutionary Many-Objective Optimization: Many Once or One Many?. Proc. of 2005 IEEE Congress on Evolutionary Computation (2005) 222-227
8. Hughes, E. J.: MSOPS-II: A General-Purpose Many-Objective Optimizer. Proc. of 2007 IEEE Congress on Evolutionary Computation (2007) 3944-3951
9. Ishibuchi, H., Hitotsuyanagi, Y., Tsukamoto, N., Nojima, Y.: Many-Objective Test Problems to Visually Examine the Behavior of Multiobjective Evolution in a Decision Space. Lecture Notes in Computer Science, Vol. 6239: PPSN XI - Part II, Springer, Berlin (2010) 91-100
10. Ishibuchi, H., Nojima, Y., Doi, T.: Comparison between Single-objective and Multi-objective Genetic Algorithms: Performance Comparison and Performance Measures. Proc. of 2006 IEEE Congress on Evolutionary Computation (2006) 3959-3966
11. Ishibuchi, H., Sakane, Y., Tsukamoto, N., Nojima, Y.: Evolutionary Many-objective Optimization by NSGA-II and MOEA/D with Large Populations. Proc. of 2009 IEEE International Conference on Systems, Man, and Cybernetics (2009) 1820-1825
12. Ishibuchi, H., Sakane, Y., Tsukamoto, N., Nojima, Y.: Simultaneous Use of Different Scalarizing Functions in MOEA/D. Proc. of 2010 Genetic and Evolutionary Computation Conference (2010) 519-526
13. 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 2008 Genetic and Evolutionary Computation Conference (2008) 649-656
14. Ishibuchi, H., Tsukamoto, N., Nojima, Y.: Evolutionary Many-Objective Optimization: A Short Review. Proc. of 2008 IEEE Congress on Evolutionary Computation (2008) 2424-2431
15. Jaszkiewicz, A.: On the Computational Efficiency of Multiple Objective Metaheuristics: The Knapsack Problem Case Study. European Journal of Operational Research 158 (2004) 418-433
16. Khara, V., Yao, X., Deb, K.: Performance Scaling of Multi-Objective Evolutionary Algorithms. Lecture Notes in Computer Science, Vol. 2632: EMO 2003, Springer, Berlin (2004) 367-390

17. Konstantinidis, A., Yang, K., Zhang, Q. F., Zeinalipour-Yazti, D.: A Multi-objective Evolutionary Algorithm for the Deployment and Power Assignment Problem in Wireless Sensor Networks. Computer Networks 54 (2010) 960-976
18. Köppen, M., Yoshida, K.: Substitute Distance Assignments in NSGA-II for Handling Many-Objective Optimization Problems. Lecture Notes in Computer Science, Vol. 4403: EMO 2007, Springer, Berlin (2007) 727-741
19. Li, H., Zhang, Q. F.: A Multiobjective Differential Evolution based on Decomposition for Multiobjective Optimization with Variable Linkages. Lecture Notes in Computer Science, Vol. 4193: PPSN IX, Springer, Berlin (2006) 583-592
20. 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
21. Murata, T., Ishibuchi, H., Gen, M.: Specification of Genetic Search Directions in Cellular Multi-Objective Genetic Algorithm. Lecture Notes in Computer Science, Vol. 1993: EMO 2001, Springer, Berlin (2001) 82-95
22. Peng, W., Zhang, Q., Li, H.: Comparison between MOEA/D and NSGA-II on the Multi-objective Travelling Salesman Problem. In Goh, C. K., Ong, Y. S., Tan, K. C. (eds) Multi-Objective Memetic Algorithms. Springer, Berlin (2009) 309-324
23. Purshouse, R. C., Fleming, P. J.: Evolutionary Many-Objective Optimization: An Exploratory Analysis. Proc. of 2003 IEEE Congress on Evolutionary Computation (2003) 2066-2073
24. Purshouse, R. C., Fleming, P. J.: On the Evolutionary Optimization of Many Conflicting Objectives. IEEE Trans. on Evolutionary Computation 11 (2007) 770-784
25. Singh, H., Isaacs, A., Ray, T., Smith, W.: A Study on the Performance of Substitute Distance Based Approaches for Evolutionary Many Objective Optimization. Lecture Notes in Computer Science, Vol. 5361: SEAL 2008, Springer, Berlin (2008) 401-410
26. Tan, K. C., Khor, E. F., Lee, T. H.: Multiobjective Evolutionary Algorithms and Applications. Springer, Berlin (2005)
27. Ulrich, T., Bader, J., Zitzler, E.: Integrating Decision Space Diversity into Hypervolume-Based Multiobjective Search. Proc. of 2010 Genetic and Evolutionary Computation Conference (2010) 455-462
28. Wagner, T., Beume, N., Naujoks, B.: Pareto-, Aggregation-, and Indicator-Based Methods in Many-Objective Optimization. Lecture Notes in Computer Science, Vol. 4403: EMO 2007, Springer, Berlin (2007) 742-756
29. Zhang, Q., Li, H.: MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. IEEE Trans. on Evolutionary Computation 11 (2007) 712-731
30. Zhang, Q. F., Liu, W. D., Tsang, E., Virginas, B.: Expensive Multiobjective Optimization by MOEA/D With Gaussian Process Model. IEEE Trans. on Evolutionary Computation 14 (2010) 456-474
31. Zitzler, E., Brockhoff, D., Thiele, L.: The Hypervolume Indicator Revisited: On the Design of Pareto-Compliant Indicators via Weighted Integration. Lecture Notes in Computer Science, Vol. 4403: EMO 2007, Springer, Berlin (2007) 862-876
32. Zitzler, E., Künzli, S.: Indicator-Based Selection in Multiobjective Search. Lecture Notes in Computer Science, Vol. 3242: PPSN VIII, Springer, Berlin (2004) 832-842
33. Zitzler, E., Laumanns, M., Thiele L.: SPEA2: Improving the Strength Pareto Evolutionary Algorithm. TIK-Report 103, Computer Engineering and Networks Laboratory (TIK), Department of Electrical Engineering, ETH, Zurich (2001)
34. 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
35. Zou, X., Chen, Y., Liu, M., Kang, L.: A New Evolutionary Algorithm for Solving Many-Objective Optimization Problems. IEEE Trans. on Systems, Man, and Cybernetics: Part B - Cybernetics 38 (2008) 1402-1412