Behavior of EMO Algorithms on Many-Objective Optimization Problems with Correlated Objectives

Hisao Ishibuchi, Naoya Akedo, Hiroyuki Ohyanagi, and Yusuke Nojima
Department of Computer Science and Intelligent Systems
Graduate School of Engineering, Osaka Prefecture University
Sakai, Osaka 599-8531, Japan
{hisaoi@, naoya.akedo@ci., ohyanagi@ci., nojima@}cs.osakafu-u.ac.jp

***Abstract*—Recently it has been pointed out in many studies that evolutionary multi-objective optimization (EMO) algorithms with Pareto dominance-based fitness evaluation do not work well on many-objective problems with four or more objectives. In this paper, we examine the behavior of well-known and frequently-used EMO algorithms such as NSGA-II, SPEA2 and MOEA/D on many-objective problems with correlated or dependent objectives. First we show that good results on many-objective 0/1 knapsack problems with randomly generated objectives are not obtained by Pareto dominance-based EMO algorithms (i.e., NSGA-II and SPEA2). Next we show that the search ability of NSGA-II and SPEA2 is not degraded by the increase in the number of objectives when they are highly correlated or dependent. In this case, the performance of MOEA/D is deteriorated. As a result, NSGA-II and SPEA2 outperform MOEA/D with respect to the convergence of solutions toward the Pareto front for some many-objective problems. Finally we show that the addition of highly correlated or dependent objectives can improve the performance of EMO algorithms on two-objective problems in some cases.**

Keywords: Evolutionary multi-objective optimization (EMO); many-objective optimization problems; multi-objective 0/1 knapsack problems; correlated objectives; independent objectives.

I. INTRODUCTION

Evolutionary many-objective optimization is a hot issue in the field of evolutionary multi-objective optimization (EMO [4], [6], [23]). Whereas well-known and frequently-used Pareto dominance-based EMO algorithms (e.g., NSGA-II [7], SPEA [29] and SPEA2 [28]) work well on multi-objective problems with two or three objectives, they often do not work well on many-objective problems with four or more objectives. That is, their search ability is severely deteriorated by the increase in the number of objectives as pointed out in the literature [11], [17], [19], [30]. This is because almost all solutions in the current population become non-dominated with each other. As a result, strong selection pressure toward the Pareto front cannot be generated by Pareto dominance-based fitness evaluation mechanisms. This means that good solutions close to the Pareto front of a many-objective problem are not likely to be obtained by Pareto dominance-based EMO algorithms.

Various approaches have been proposed to the handling of many-objective problems by EMO algorithms in the literature [15]. Those approaches can be categorized as follows:

(1) Dimensionality reduction [2], [3], [8],
(2) Preference incorporation [9], [10], [24],
(3) Selection pressure enhancement [5], [16], [18], [20], [22],
(4) Different fitness evaluation schemes [1], [12], [25]-[27].

Approaches in the "dimensionality reduction" category try to decrease the number of objectives by removing unnecessary objectives. If we can decrease the number of objectives in a many-objective problem to two or three, EMO algorithms may work well on the reduced problem with two or three objectives. Approaches in the "preference incorporation" category use a decision maker's preference to realize efficient multi-objective search by concentrating on preferred regions of the Pareto front. The "selection pressure enhancement" category includes various proposals for increasing the selection pressure toward the Pareto front. Approaches in the last category do not use Pareto dominance for fitness evaluation. Hypervolume and scalarizing functions have been used for fitness evaluation.

A general process of dimensionality reduction approaches [2], [3], [8] can be written as the following three steps:

Step 1: Search for a large number of non-dominated solutions of a many-objective problem using an EMO algorithm.
Step 2: Decrease the number of objectives using the obtained non-dominated solutions in Step 1.
Step 3: Apply an EMO algorithm to the reduced problem.

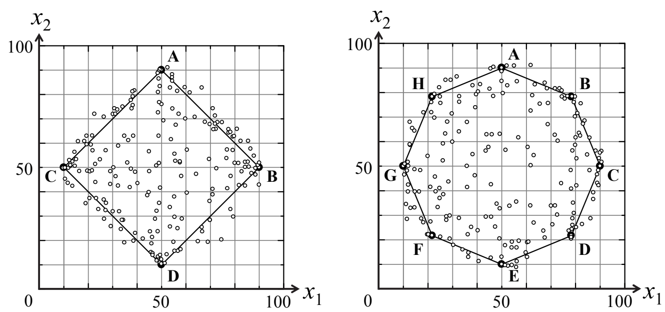
One question about this dimensionality reduction process is the quality of non-dominated solutions obtained by an EMO algorithm in Step 1. In general, EMO algorithms do not work well on many-objective problems. Thus it is not likely that the non-dominated solutions in Step 1 are close to the Pareto front. However, good results of dimensionality reduction have often been reported in the literature. There may be the following three possibilities:

Possibility 1: The quality of obtained non-dominated solutions in Step 1 is good when only a few objectives are important.
Possibility 2: Dimensionality reduction approaches work well even if the quality of obtained non-dominated solutions is not high (i.e., even if they are not close to the Pareto front).
Possibility 3: Both of the possibilities 1 and 2.

This study is motivated by the first possibility "The quality of obtained non-dominated solutions in Step 1 is good when

This work was supported in part by JSPS under Grant-in-Aid for Scientific Research (B) (20300084).

only a few objectives are important." This possibility has already been suggested in some studies on many-objective problems. For example, it is shown in [14] that distance minimization problems to multiple points in a two-dimensional decision space [13], [18], [22] can be handled by Pareto dominance-based EMO algorithms even when the number of objectives is four or more. In Fig. 1, we show experimental results of NSGA-II on distance minimization problems with four and eight objectives. Each plot in Fig. 1 shows all solutions in the final population of a single run of NSGA-II. NSGA-II was applied to each problem under the same conditions as in our former study [14] such as the population size 200 and the termination condition of 400,000 solution evaluations. Good results were obtained in both plots in Fig. 1 independent of the number of objectives.



(a) Four-objective problem. (b) Eight-objective problem.

Figure 1. All solutions at the final generation of a single run of NSGA-II on four-objective and eight-objective distance minimization problems. All points inside the square in (a) and the octagon in (b) are Pareto optimal.

Recently, Schutze *et al.* [21] suggested that the increase in the number of objectives does not necessarily make multi-objective problems more difficult when the dimensionality of the Pareto front is not increased. This feature was demonstrated through computational experiments on multi-objective distance minimization problems in a high-dimensional decision space. In this paper, we demonstrate that the increase in the number of objectives does not necessarily increase the difficulty of multi-objective 0/1 knapsack problems when highly correlated or dependent objectives are added. Moreover, we also show that the inclusion of those objectives can improve the performance of EMO algorithms on two-objective problems in some cases.

This paper is organized as follows. In Section II, we explain how to generate many-objective test problems with random, correlated and dependent objectives from the two-objective 500-item 0/1 knapsack problem of Zitzler and Thiele [29]. Our test problems are many-objective knapsack problems with up to ten objectives. In Section III, we report experimental results of NSGA-II [7], SPEA2 [28] and MOEA/D [26] on our test problems. Finally we conclude this paper in Section IV.

II. Test Problems

A. Two-Objective 0/1 Knapsack Problem

We generate our test problems from the two-objective 500-item 0/1 knapsack problem in Zitzler and Thiele [29]. Their two-objective n-item 0/1 knapsack problem is written as

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

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

$$x_j = 0 \text{ or } 1,\ \ j = 1, 2, ..., n\ , \tag{3}$$

$$\text{where} \qquad f_i(\mathbf{x}) = \sum_{j=1}^{n} p_{ij} x_j\ ,\quad i = 1, 2\ . \tag{4}$$

In this formulation, $\mathbf{x}$ is an n-dimensional binary vector, 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. Each solution $\mathbf{x}$ is handled as a binary string of length n in EMO algorithms. In our computational experiments, the number of items is always 500 (i.e., n = 500). The two-objective 500-item 0/1 knapsack problem of Zitzler and Thiele [29] is referred to as the 2-500 problem in this paper.

In the execution of EMO algorithms on the 2-500 problem, infeasible solutions are often generated by genetic operations. Infeasible solutions are also included in a randomly generated initial population. In order to transform an infeasible solution into a feasible one, we use the same greedy repair method based on the maximum profit/weight ratio as in Zitzler and Thiele [29]. That is, we remove items from an infeasible solution in an ascending order of the following value of each item j until the constraint conditions in (2) are satisfied:

$$q_j = \max\{p_{ij} / w_{ij} \mid i = 1, 2\},\ \ j = 1, 2, ..., 500\ . \tag{5}$$

We always use an ascending order of q_j in (5) in our computational experiments on all test problems in this paper.

In the 2-500 problem, the profit p_{ij} of each item j for each objective i was randomly specified as an integer in the interval [10, 100] in Zitzler and Thiele [29]. As a result, the 2-500 problem has a wide Pareto front with a large number of Pareto optimal solutions as shown in Fig. 2 where we also show randomly generated 200 solutions for comparison.

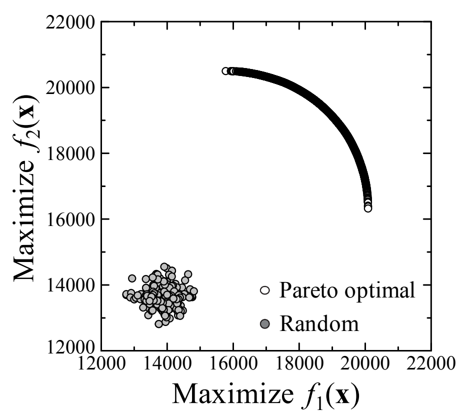


Figure 2. Pareto optimal solutions and randomly generated 200 solutions of the 2-500 problem.

B. Many-Objective Problems with Random Objectives

By randomly specifying the profit p_{ij} of each item j for each objective i as an integer in the interval [10, 100], we generated other eight objectives:

$$f_i(\mathbf{x}) = \sum_{j=1}^{n} p_{ij} x_j \,, \quad i = 3, 4, \ldots, 10 \,. \qquad (6)$$

In our computational experiments, we use the following many-objective problems with randomly generated objectives:

Random 4-500 problem: $(f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_4(\mathbf{x}))$,
Random 6-500 problem: $(f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_6(\mathbf{x}))$,
Random 8-500 problem: $(f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_8(\mathbf{x}))$,
Random 10-500 problem: $(f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_{10}(\mathbf{x}))$,

where all objectives are maximized. The constraint conditions in (2) of the 2-500 problem are used in all test problems.

C. Many-Objective Problems with Correlated Objectives

Using the randomly generated ten objectives in the previous subsection, we generated ten correlated objectives as follows:

$$g_i(\mathbf{x}) = f_i(\mathbf{x}) \,, \quad i = 1, 2 \,, \qquad (7)$$

$$g_i(\mathbf{x}) = \alpha \cdot f_i(\mathbf{x}) + (1-\alpha) \cdot f_1(\mathbf{x}) \,, \quad i = 3, 5, 7, 9 \,, \qquad (8)$$

$$g_i(\mathbf{x}) = \alpha \cdot f_i(\mathbf{x}) + (1-\alpha) \cdot f_2(\mathbf{x}) \,, \quad i = 4, 6, 8, 10 \,, \qquad (9)$$

where α is a small positive real number ($0 < \alpha < 1$).

These ten objectives can be divided into two groups: $\{g_1(\mathbf{x}), g_3(\mathbf{x}), g_5(\mathbf{x}), g_7(\mathbf{x}), g_9(\mathbf{x})\}$ and $\{g_2(\mathbf{x}), g_4(\mathbf{x}), g_6(\mathbf{x}), g_8(\mathbf{x}), g_{10}(\mathbf{x})\}$. When α is close to 1, $g_i(\mathbf{x})$ in (8)-(9) is almost the same as the randomly generated objective $f_i(\mathbf{x})$. When α is close to 0, all objectives in the first group are almost the same as $f_1(\mathbf{x})$ while all objectives in the second group are almost the same as $f_2(\mathbf{x})$. In our computational experiments, we examine the following problems for $\alpha = 0.1$:

Correlated 4-500 problem: $(g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_4(\mathbf{x}))$,
Correlated 6-500 problem: $(g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_6(\mathbf{x}))$,
Correlated 8-500 problem: $(g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_8(\mathbf{x}))$,
Correlated 10-500 problem: $(g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_{10}(\mathbf{x}))$.

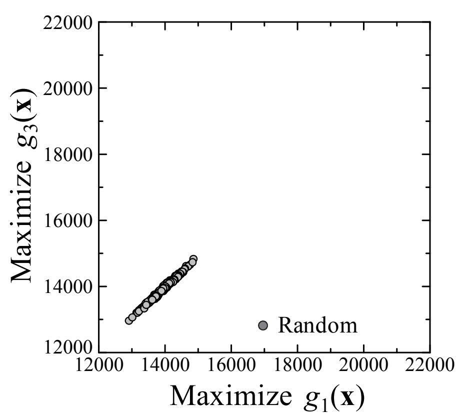


Figure 3. The randomly generated 200 solutions in Fig. 2, which are shown in a two-dimensional objective space with $g_1(\mathbf{x})$ and $g_3(\mathbf{x})$ for $\alpha = 0.1$ in this figure.

In Fig. 3, we show the projection onto the two-dimensional space with $g_1(\mathbf{x})$ and $g_3(\mathbf{x})$ for $\alpha = 0.1$ of the randomly generated 200 solutions in Fig. 2. That is, Fig. 3 is the projection of all the random solutions in Fig. 2 from the $f_1(\mathbf{x})$-$f_2(\mathbf{x})$ space to the $g_1(\mathbf{x})$-$g_3(\mathbf{x})$ space with $\alpha = 0.1$. We can see from Fig. 2 and Fig. 3 that $g_1(\mathbf{x})$ and $g_3(\mathbf{x})$ are highly correlated.

D. Many-Objective Problems with Dependent Objectives

We also generated dependent objectives from the original two objectives $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ of the 2-500 problem as follows:

$$h_i(\mathbf{x}) = f_i(\mathbf{x}) \,, \quad i = 1, 2 \,, \qquad (10)$$

$$h_3(\mathbf{x}) = f_1(\mathbf{x}) + \alpha \cdot f_2(\mathbf{x}) \,, \qquad (11)$$

$$h_4(\mathbf{x}) = f_2(\mathbf{x}) + \alpha \cdot f_1(\mathbf{x}) \,, \qquad (12)$$

$$h_5(\mathbf{x}) = f_1(\mathbf{x}) - \alpha \cdot f_2(\mathbf{x}) \,, \qquad (13)$$

$$h_6(\mathbf{x}) = f_2(\mathbf{x}) - \alpha \cdot f_1(\mathbf{x}) \,, \qquad (14)$$

$$h_7(\mathbf{x}) = f_1(\mathbf{x}) + \beta \cdot f_2(\mathbf{x}) \,, \qquad (15)$$

$$h_8(\mathbf{x}) = f_2(\mathbf{x}) + \beta \cdot f_1(\mathbf{x}) \,, \qquad (16)$$

$$h_9(\mathbf{x}) = f_1(\mathbf{x}) - \beta \cdot f_2(\mathbf{x}) \,, \qquad (17)$$

$$h_{10}(\mathbf{x}) = f_2(\mathbf{x}) - \beta \cdot f_1(\mathbf{x}) \,, \qquad (18)$$

where α and β are small positive real numbers ($0 < \alpha < \beta < 1$).

These ten objectives can be divided into two groups: $\{h_1(\mathbf{x}), h_3(\mathbf{x}), h_5(\mathbf{x}), h_7(\mathbf{x}), h_9(\mathbf{x})\}$ and $\{h_2(\mathbf{x}), h_4(\mathbf{x}), h_6(\mathbf{x}), h_8(\mathbf{x}), h_{10}(\mathbf{x})\}$. When α and β are very small (i.e., close to 0), all objectives in the first group are very similar to $f_1(\mathbf{x})$ while all objectives in the second group are very similar to $f_2(\mathbf{x})$. In our computational experiments, $\alpha = 0.1$ and $\beta = 0.2$ are used.

In the same manner as the correlated k-500 problems in the previous subsection, we specify dependent k-500 problems for $k = 4, 6, 8, 10$ using $h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_{10}(\mathbf{x})$.

In Fig. 4, we show the projections of Fig. 2 onto the two-dimensional space with $h_1(\mathbf{x})$ and $h_3(\mathbf{x})$ for $\alpha = 0.1$. All the randomly generated 200 solutions and the Pareto optimal solutions in Fig. 2 with $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ are mapped to Fig. 4 with $h_1(\mathbf{x})$ and $h_3(\mathbf{x})$. We can see from Fig. 4 that $h_1(\mathbf{x})$ and $h_3(\mathbf{x})$ are highly correlated as $g_1(\mathbf{x})$ and $g_3(\mathbf{x})$ in Fig. 3.

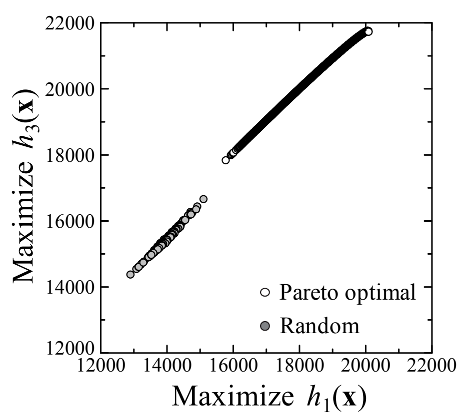


Figure 4. The Pareto optimal solutions of the 2-500 problem and the randomly generated 200 solutions in Fig. 2, which are shown in a two-dimensional objective space with $h_1(\mathbf{x})$ and $h_3(\mathbf{x})$ for $\alpha = 0.1$ in this figure.

III. Experimental Results

A. EMO Algorithms and Parameter Specifications

We examine the behavior of NSGA-II [7], SPEA2 [28] and MOEA/D [26] through computational experiments on our test problems with 2, 4, 6, 8 and 10 objectives. In MOEA/D, we do not use any external archive population for avoiding severe increase in computation load for many-objective problems.

MOEA/D uses all weight vectors $\boldsymbol{\lambda} = (\lambda_1, ..., \lambda_k)$ satisfying the following two conditions:

$$\lambda_1 + \lambda_2 + \cdots + \lambda_k = 1, \tag{19}$$

$$\lambda_i \in \left\{0, \frac{1}{H}, \frac{2}{H}, ..., \frac{H}{H}\right\}, \ i = 1, 2, ..., k, \tag{20}$$

where H is a user-definable positive integer. The number of weight vectors is calculated as $N = {}_{H+k-1}C_{k-1}$ [26] for a k-objective problem. It should be noted that the number of weight vectors is the same as the population size in MOEA/D.

NSGA-II, SPEA2 and MOEA/D are applied to each test problem using the following setting:

Coding: Binary string of length 500,
Termination condition: 400,000 solution evaluations,
Crossover probability: 0.8 (Uniform crossover),
Mutation probability: 1/500 (Bit-flip mutation),
Population size in NSGA-II and SPEA2: 200,
Integer H for the weight vector specification in MOEA/D:
  199 (2-objective), 9 (4-objective), 5 (6-objective),
  3 (8-objective), 3 (10-objective),
Population size in MOEA/D:
  200 (2-objective), 220 (4-objective), 252 (6-objective),
  120 (8-objective), 220 (10-objective),
Scalarizing function in MOEA/D: Weighted Tchebycheff,
Neighborhood size in MOEA/D: 10 solutions.

B. Performance Measures

Hypervolume has been used for performance evaluation of EMO algorithms. However, its use for many-objective cases is not easy due to its heavy computation load. In this paper, we use some simple performance measures. Of course, we also use hypervolume for multi-objective problems with 2-6 objectives.

At each generation, we calculate the maximum value of the sum of the objectives in the current population as follows:

$$\text{MaxSum}(\Psi) = \max_{\mathbf{x}\in\Psi} \sum_{i=1}^{k} f_i(\mathbf{x}), \tag{21}$$

where Ψ denotes the current population, and k is the number of the objectives in each test problem (k = 2, 4, 6, 8, 10). This measure evaluates the convergence of a population toward the Pareto front around its center region.

We also calculate the sum of the maximum value of each objective at each generation as follows:

$$\text{SumMax}(\Psi) = \sum_{i=1}^{k} \max_{\mathbf{x}\in\Psi} f_i(\mathbf{x}). \tag{22}$$

This measure evaluates the convergence of a population toward the optimal value of each objective in the Pareto front.

The sum of the range of objective values of each objective is also calculated in each generation as follows:

$$\text{Range}(\Psi) = \sum_{i=1}^{k} [\max_{\mathbf{x}\in\Psi}\{f_i(\mathbf{x})\} - \min_{\mathbf{x}\in\Psi}\{f_i(\mathbf{x})\}]. \tag{23}$$

This measure directly evaluates the diversity of a population in the objective space in each generation.

We use these three simple measures because the meaning of each measure is easy to understand even for many-objective problems. Computational efficiency in their calculation is also their merit. Of course, each measure is too simple to rigorously discuss the performance of EMO algorithms. In this paper, we use these simple measures mainly for monitoring the behavior of each EMO algorithm rather than for rigorously comparing different EMO algorithms.

C. Results on Randomly Generated Problems

First we show average results over 100 runs of NSGA-II on the random 2-500, 4-500, 6-500, 8-500 and 10-500 problems. In this paper, we always report average results over 100 runs. In Fig. 5, we show the average percentage of non-dominated solutions at each generation. We can see from Fig. 5 that the increase in the number of objectives leads to the increase in the percentage of non-dominated solutions. Except for the 2-500 problem with two objectives, almost all solutions in the current population became non-dominated within 100 generations.

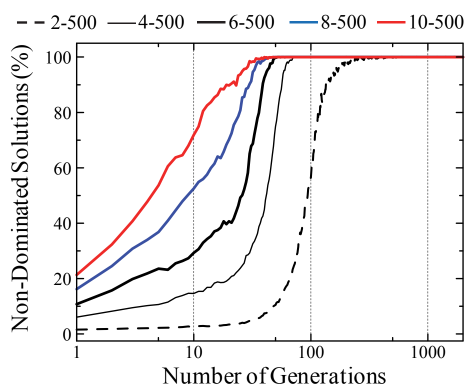
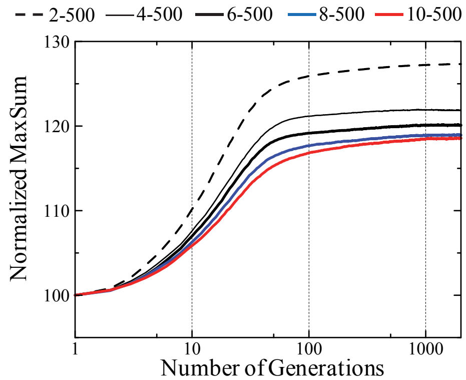


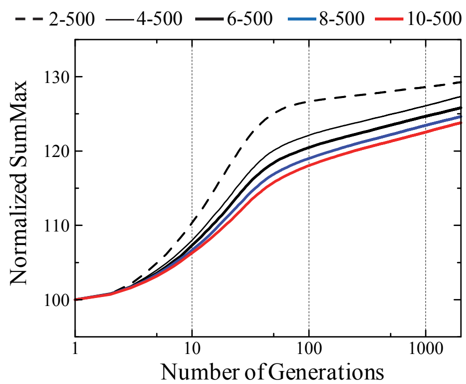
Figure 5. Average percentage of non-dominated solutions at each generation of NSGA-II on the random k-500 knapsack problems (k = 2, 4, 6, 8, 10).

Experimental results of NSGA-II on the random knapsack problems are summarized in Fig. 6. Fig. 6 (a) shows the average values of the normalized MaxSum. Normalization was performed so that the average value for the initial population of NSGA-II became 100. This normalization based on NSGA-II was also used for SPEA2 and MOEA/D. That is, all the performance measures on each test problem were normalized using the initial population of NSGA-II. In Fig. 6, we also show the normalized SumMax and the normalized Range.

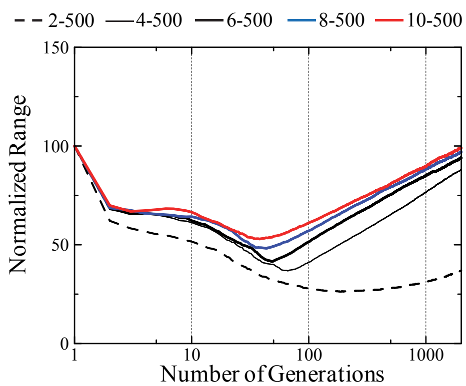
In the same manner as Fig. 6, we show experimental results of SPEA2 and MOEA/D in Fig. 7 and Fig. 8, respectively. The initial population of NSGA-II was used for normalization.



(a) Average normalized MaxSum measure at each generation.

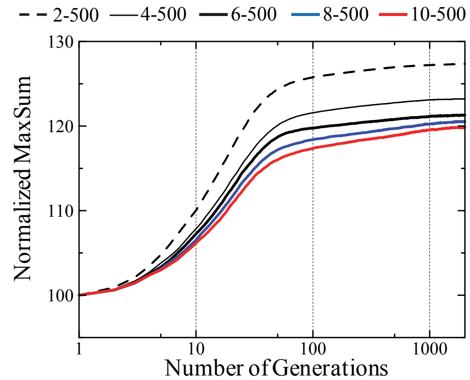


(b) Average normalized SumMax measure at each generation.

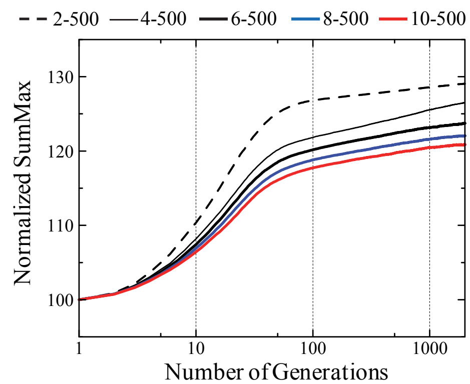


(c) Average normalized Range measure at each generation.

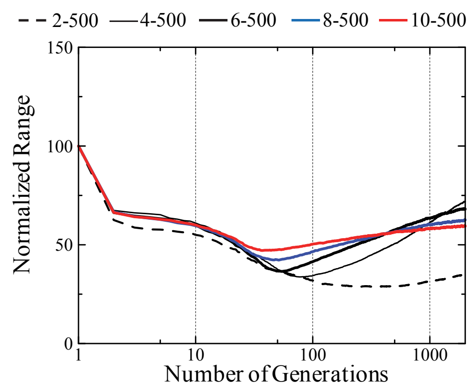
Figure 6. Experimental results of NSGA-II on the randomly generated 2-500, 4-500, 6-500, 8-500 and 10-500 knapsack problems.



(a) Average normalized MaxSum measure at each generation.



(b) Average normalized SumMax measure at each generation.



(c) Average normalized Range measure at each generation.

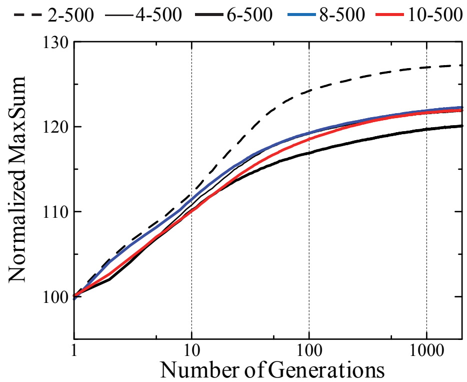
Figure 7. Experimental results of SPEA2 on the randomly generated 2-500, 4-500, 6-500, 8-500 and 10-500 knapsack problems.

It should be noted in Fig. 8 that the number of generations of MOEA/D on each test problem with different population size is converted to the equivalent one in the case of population size 200 in terms of the number of examined solutions.

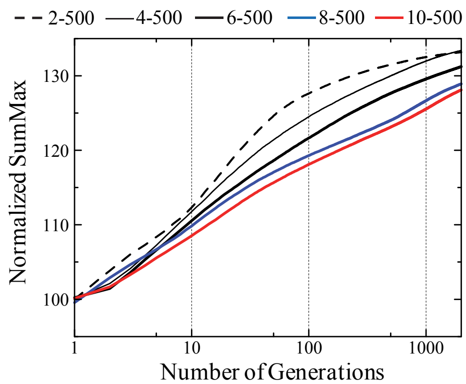
In Fig. 6 (a), we can observe that the increase in the number of objectives slowed down the convergence toward the center region of the Pareto front. Convergence improvement in Fig. 6 (a) was very slow after the 100th generation where almost all solutions in the current population were non-dominated except for the case of the 2-500 problem as shown in Fig. 5. From the comparison between Fig. 5 and Fig. 6 (c), we can see that the diversity of a population started to increase when almost all solutions in the current population became non-dominated.

In Fig. 6 (b), the convergence toward the optimal value of each objective continued to improve over 2000 generations thanks to the diversity improvements shown in Fig. 6 (c).

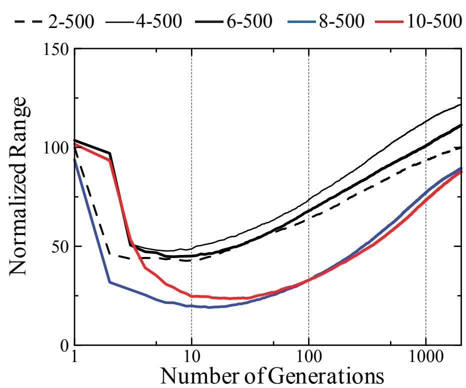
Similar results were obtained by NSGA-II and SPEA2 in Fig. 6 and Fig. 7. These two algorithms, however, showed different diversity improvement behavior (see Fig. 6 (c) and Fig. 7 (c)). The behavior of MOEA/D in Fig. 8 was different from NSGA-II and SPEA2. We can observe better results by MOEA/D in Fig. 8 (b) with respect to SumMax than NSGA-II in Fig. 6 (b) and SPEA2 in Fig. 7 (b). In Table I, these three algorithms are compared using the hypervolume measure. The origin of the objective space (e.g., (0, 0) for the 2-500 problem) was used as the reference point in hypervolume calculation.



(a) Average normalized MaxSum measure at each generation.



(b) Average normalized SumMax measure at each generation.



(c) Average normalized Range measure at each generation.

Figure 8. Experimental results of MOEA/D on the randomly generated 2-500, 4-500, 6-500, 8-500 and 10-500 knapsack problems.

TABLE I. AVERAGE HYPERVOLUME AND STANDARD DEVIATION ON RANDOM TEST PROBLEMS. STANDARD DEVIATION IS IN PARENTHESES.

Test Problem	NSGA-II	SPEA2	MOEA/D
Original 2-500	3.800E+08 (1.608E+06)	3.788E+08 (1.378E+06)	4.009E+08 (1.009E+06)
Random 4-500	1.231E+17 (8.996E+14)	1.218E+17 (7.245E+14)	1.436E+17 (7.101E+14)
Random 6-500	3.746E+25 (4.064E+23)	3.557E+25 (3.790E+23)	4.530E+25 (3.588E+23)

In Table I, the best results were obtained by MOEA/D. The increase in the number of objectives seems to increase the advantage of MOEA/D over NSGA-II and SPEA2 in Table I.

D. Results on the Correlated Problems

In the same manner as in the previous subsection, we report experimental results on the correlated 4-500, 6-500, 8-500 and 10-500 problems. In Fig. 9, we show the average percentage of non-dominated solutions at each generation of NSGA-II. From the comparison between Fig. 5 on the random *k*-500 problems and Fig. 9 on the correlated *k*-500 problems, we can see that the addition of the correlated objectives had almost no effects on the percentage of non-dominated solutions. As a result, the convergence of solutions toward the center region of the Pareto front was not slowed down by the increase in the number of objective as shown in Fig. 10. We can observe from the comparison between Fig. 6 (a) and Fig. 10 that the correlated objectives in Fig. 10 had almost no negative effects on the convergence of solutions in NSGA-II. Similar observations were obtained for SPEA2. However, the addition of the correlated objectives somewhat degraded the convergence property of MOEA/D toward the Pareto front as shown in Fig. 11 (compare Fig. 11 with Fig. 10).

The performance of the three algorithms on the correlated *k*-500 problems is examined using the hypervolume measure in Table II. For the correlated 4-500 and 6-500 test problems, the best results were obtained by MOEA/D as in the case of the random 4-500 and 6-500 problems in Table I. However, the difference in the average hypervolume between MOEA/D and the other EMO algorithms was much smaller in Table II on the correlated problems than Table I on the random problems.

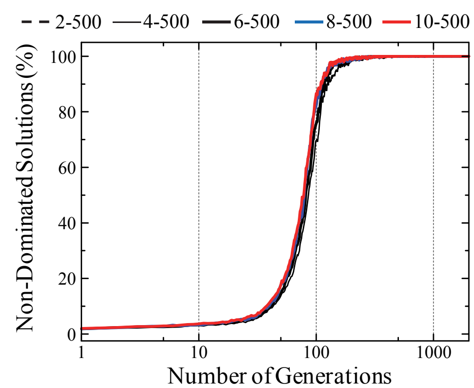


Figure 9. Average percentage of non-dominated solutions at each generation of NSGA-II on the correlated *k*-500 knapsack problems (k = 2, 4, 6, 8, 10).

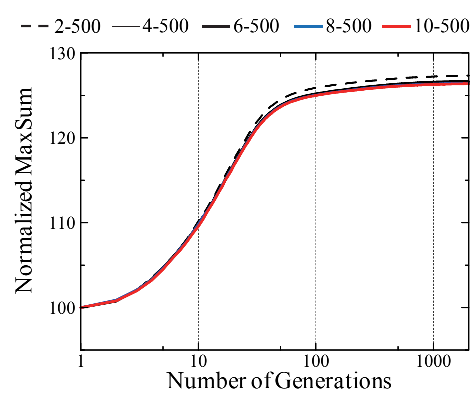


Figure 10. Average normalized MaxSum measure at each generation of NSGA-II on the correlated *k*-500 knapsack problems (k = 2, 4, 6, 8, 10).

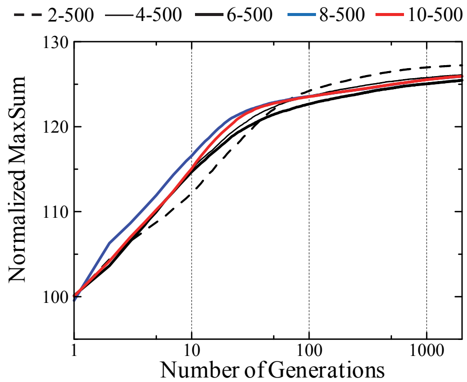


Figure 11. Average normalized MaxSum measure at each generation of MOEA/D on the correlated k-500 knapsack problems (k = 2, 4, 6, 8, 10).

TABLE II. AVERAGE HYPERVOLUME AND STANDARD DEVIATION ON CORRELATED TEST PROBLEMS. STANDARD DEVIATION IS IN PARENTHESES.

Test Problem	NSGA-II	SPEA2	MOEA/D
Correlated 4-500	1.400E+17 (9.361E+14)	1.377E+17 (8.953E+14)	1.501E+17 (7.505E+14)
Correlated 6-500	5.158E+25 (5.290E+23)	5.029E+25 (4.798E+23)	5.537E+25 (3.860E+23)

E. Results on the Dependent Problems

In this subsection, we report experimental results on the dependent 4-500, 6-500, 8-500 and 10-500 problems. In Fig. 12, we show the average percentage of non-dominated solutions at each generation of NSGA-II. As in Fig. 9 with the correlated objectives, the addition of the dependent objectives has almost no effect on the percentage of non-dominated solutions. As a result, the convergence of solutions toward the center region of the Pareto front in NSGA-II was not slowed down by the increase in the number of objectives. However, as in Fig. 11 with the correlated objectives, the addition of the dependent objectives degraded the convergence property of MOEA/D.

In Table III, we compare the performance of the three algorithms on the dependent k-500 problems using the hypervolume measure. As in Table I and Table II, the best results were obtained by MOEA/D on the dependent 4-500 and 6-500 problems. However, as in Table II, the difference in the average hypervolume between MOEA/D and the other EMO algorithms was much smaller in Table III on the dependent problems than Table I on the random problems.

If dimensionality reduction is applied to our correlated and dependent test problems, the number of objectives may be decreased to two. So we compare the performance of the three EMO algorithms on our test problems using the hypervolume measure on the two-dimensional objective space of the 2-500 problem. That is, the obtained solutions for each test problem by each algorithm were projected to the two-dimensional space with $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$. Then their hypervolume was calculated in the two-dimensional objective space. Experimental results are summarized in Table IV. In each column, the best result is highlighted by boldface. It is very interesting to observe that the best results for NSGA-II and SPEA2 were obtained by applying them to the dependent 10-500 problem rather than the original 2-500 problem. This is because the added objectives increased the diversity of solutions as shown in Fig. 13.

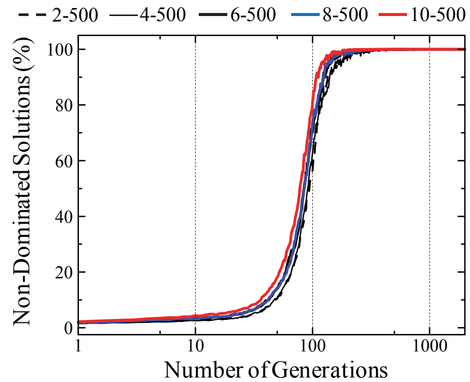


Figure 12. Average percentage of non-dominated solutions at each generation of NSGA-II on the dependent k-500 knapsack problems (k = 2, 4, 6, 8, 10).

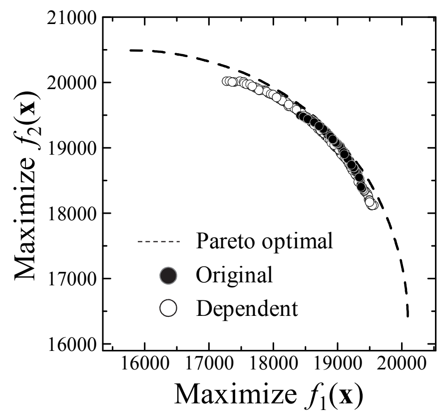


Figure 13. The obtained solutions at the 2000th generation by a single run of NSGA-II on the original 2-500 problem and the dependent 10-500 problem.

TABLE III. AVERAGE HYPERVOLUME AND STANDARD DEVIATION ON DEPENDENT TEST PROBLEMS. STANDARD DEVIATION IS IN PARENTHESES.

Test Problem	NSGA-II	SPEA2	MOEA/D
Dependent 4-500	1.728E+17 (1.092E+15)	1.716E+17 (1.078E+15)	1.842E+17 (8.562E+14)
Dependent 6-500	5.490E+25 (5.588E+23)	5.385E+25 (3.459E+23)	5.895E+25 (4.002E+23)

TABLE IV. AVERAGE HYPERVOLUME IN THE TWO-DIMENSIONAL OBJECTIVE SPACE OF OBTAINED SOLUTION SETS BY EACH ALGORITHM ON EACH TEST PROBLEM.

Test Problem	NSGA-II	SPEA2	MOEA/D
Original 2-500	3.800E+08	3.788E+08	**4.009E+08**
Random 4-500	3.744E+08	3.687E+08	3.927E+08
Random 6-500	3.676E+08	3.534E+08	3.819E+08
Random 8-500	3.629E+08	3.454E+08	3.640E+08
Random 10-500	3.590E+08	3.387E+08	3.565E+08
Correlated 4-500	3.814E+08	3.780E+08	3.977E+08
Correlated 6-500	3.819E+08	3.782E+08	3.958E+08
Correlated 8-500	3.828E+08	3.783E+08	3.917E+08
Correlated 10-500	3.831E+08	3.787E+08	3.916E+08
Dependent 4-500	3.803E+08	3.788E+08	3.971E+08
Dependent 6-500	3.837E+08	3.809E+08	3.970E+08
Dependent 8-500	3.835E+08	3.812E+08	3.920E+08
Dependent 10-500	**3.884E+08**	**3.838E+08**	3.918E+08

IV. CONCLUSIONS

In this paper, we demonstrated that the convergence ability of NSGA-II and SPEA2 was not severely degraded by the increase in the number of objectives when they were highly correlated or dependent. As a result, good solution sets were obtained by those Pareto dominance-based EMO algorithms for many-objective problems when their objectives were highly correlated or dependent. Moreover, in some cases, better results for the 2-500 problem were obtained through the application of EMO algorithms to many-objective problems with dependent objectives rather than their direct application to the 2-500 problem. This observation suggests the potential usefulness of multiobjectivization from two-objective problems to many-objective problems with dependent objectives.

The best results among NSGA-II, SPEA2 and MOEA/D were obtained by MOEA/D for the two-objective and many-objective knapsack problems when objectives were generated randomly. However, its search ability was degraded by the increase in the number of objectives even when they were highly correlated or dependent. As a result, MOEA/D did not always show the best performance for the correlated and dependent problems. These observations suggest that the difficulty of many-objective problems depends on features of EMO algorithms as well as those of many-objective problems.

REFERENCES

[1] N. Beume, B. Naujoks, and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *European Journal of Operational Research* 180 (3), pp. 1653-1669, 2007.

[2] D. Brockhoff and E. Zitzler, "Are all objectives necessary? On dimensionality reduction in evolutionary multiobjective optimization," *Lecture Notes in Computer Science, vol. 4193: Parallel Problem Solving from Nature - PPSN IX*, pp. 533-542, Springer, Berlin, September 2006.

[3] D. Brockhoff and E. Zitzler, "Improving hypervolume-based multiobjective evolutionary algorithms by using objective reduction methods," *Proc. of 2007 IEEE Congress on Evolutionary Computation*, pp. 2086-2093, Singapore, September 25-28, 2007.

[4] C. A. C. Coello and G. B. Lamont, *Applications of Multi-Objective Evolutionary Algorithms*, World Scientific, Singapore, 2004.

[5] D. Corne and J. Knowles, "Techniques for highly multiobjective optimization: Some non-dominated points are better than others," *Proc. of 2007 Genetic and Evolutionary Computation Conference*, pp. 773-780, London, July 7-11, 2007.

[6] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*, John Wiley & Sons, Chichester, 2001.

[7] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. on Evolutionary Computation* 6 (2), pp. 182-197, 2002.

[8] K. Deb and K. Saxena, "Searching for Pareto-optimal solutions through dimensionality reduction for certain large-dimensional multi-objective optimization problems," *Proc. of 2006 IEEE Congress on Evolutionary Computation*, pp. 3353-3360, Vancouver, July 16-21, 2006.

[9] K. Deb and J. Sundar, "Preference point based multi-objective optimization using evolutionary algorithms," *Proc. of 2006 Genetic and Evolutionary Computation Conference*, pp. 635-642, Seattle, July 8-12, 2006.

[10] P. J. Fleming, R. C. Purshouse, and R. J. Lygoe, "Many-objective optimization: An engineering design perspective," *Lecture Notes in Computer Science, vol. 3410: Evolutionary Multi-Criterion Optimization - EMO 2005*, pp. 14-32, Springer, Berlin, March 2005.

[11] E. J. Hughes, "Evolutionary many-objective optimization: Many once or one many?," *Proc. of 2005 IEEE Congress on Evolutionary Computation*, pp. 222-227, Edinburgh, September 2-5, 2005.

[12] E. J. Hughes, "MSOPS-II: A general-purpose many-objective optimiser," *Proc. of 2007 IEEE Congress on Evolutionary Computation*, pp. 3944-3951, Singapore, September 25-28, 2007.

[13] H. Ishibuchi, Y. Hitotsuyanagi, and Y. Nojima, "Many-objective test problems to visually examine the behavior of multi-objective evolution in a decision space," *Lecture Notes in Computer Science, Vol. 6239: PPSN XI*, Springer, Berlin, pp. 91-100, 2010.

[14] H. Ishibuchi, Y. Hitotsuyanagi, H. Ohyanagi, and Y. Nojima, "Effects of the Existence of Highly Correlated Objectives on the Behavior of MOEA/D," *Proc. of International Conference on Evolutionary Multi-Criterion Optimization - EMO 2011*, (in press) Springer, Berlin, April 2011.

[15] H. Ishibuchi, N. Tsukamoto, and Y. Nojima, "Evolutionary many-objective optimization: A short review," *Proc. of 2008 IEEE Congress on Evolutionary Computation*, pp. 2424-2431, 2008.

[16] S. Kukkonen and J. Lampinen, "Ranking-dominance and many-objective optimization," *Proc. of 2007 IEEE Congress on Evolutionary Computation*, pp. 3983-3990, Singapore, September 25-28, 2007.

[17] V. Khara, X. Yao, and K. Deb, "Performance scaling of multi-objective evolutionary algorithms," *Lecture Notes in Computer Science, Vol. 2632: EMO 2003*, Springer, Berlin, pp. 367-390, 2003.

[18] M. Köppen and K. Yoshida, "Substitute distance assignments in NSGA-II for handling many-objective optimization problems," *Lecture Notes in Computer Science, Vol. 4403: EMO 2007*, Springer, Berlin, pp. 727-741, 2007.

[19] R. C. Purshouse and P. J. Fleming, "Evolutionary many-objective optimization: An exploratory analysis," *Proc. of 2003 IEEE Congress on Evolutionary Computation*, pp. 2066-2073, Canberra, December 8-12, 2003.

[20] H. Sato, H. E. Aguirre, and K. Tanaka, "Controlling dominance area of solutions and its impact on the performance of MOEAs," *Lecture Notes in Computer Science, vol. 4403: Evolutionary Multi-Criterion Optimization - EMO 2007*, pp. 5-20, Springer, Berlin, March 2007.

[21] O. Schutze, A. Lara, and C. A. C. Coello, "On the influence of the number of objectives on the hardness of a multiobjective optimization problem," *IEEE Trans. on Fuzzy Systems*, to appear (Available online through IEEE Explore).

[22] H. Singh, A. Isaacs, T. Ray, and W. Smith, "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, pp. 411-420, 2008.

[23] K. C. Tan, E. F. Khor, and T. H. Lee, *Multiobjective Evolutionary Algorithms and Applications*, Springer, Berlin, 2005.

[24] L. Thiele, K. Miettinen, P. J. Korhonen, and J. Molina, "A preference-based interactive evolutionary algorithm for multiobjective optimization," *Helsinki School of Economics*, Working Paper, W-412, January 2007.

[25] T. Wagner, N. Beume, and B. Naujoks, "Pareto-, aggregation-, and indicator-based methods in many-objective optimization," *Lecture Notes in Computer Science, Vol. 4403: EMO 2007*, Springer, Berlin, pp. 742-756, 2007.

[26] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. on Evolutionary Computation* 11 (6), pp. 712-731, 2007.

[27] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," *Lecture Notes in Computer Science, Vol. 3242: PPSN VIII*, Springer, Berlin, pp. 832-842, 2004.

[28] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm," *TIK-Report 103*, Computer Engineering and Networks Laboratory (TIK), Department of Electrical Engineering, ETH, Zurich, 2001.

[29] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," *IEEE Trans. on Evolutionary Computation* 3 (4), pp. 257-271, 1999.

[30] X. Zou, Y. Chen, M. Liu, and L. Kang, "A new evolutionary algorithm for solving many-objective optimization problems," *IEEE Trans. on Systems, Man, and Cybernetics: Part B - Cybernetics* 38 (5), pp. 1402-1412, 2008.