Effects of Removing Overlapping Solutions on the Performance of the NSGA-II Algorithm

Yusuke Nojima, Kaname Narukawa, Shiori Kaige, and Hisao Ishibuchi

Department of Industrial Engineering, Osaka Prefecture University
1-1 Gakuen-cho, Sakai, Osaka, 599-8531, Japan
{nojima, kaname, shiori, hisaoi}@ie.osakafu-u.ac.jp

Abstract. The focus of this paper is the handling of overlapping solutions in evolutionary multiobjective optimization (EMO) algorithms. In the application of EMO algorithms to some multiobjective combinatorial optimization problems, there exit a large number of overlapping solutions in each generation. We examine the effect of removing overlapping solutions on the performance of EMO algorithms. In this paper, overlapping solutions are removed from the current population except for a single solution. We implement two removal strategies of overlapping solutions. One is the removal of overlapping solutions in the objective space. In this strategy, one solution is randomly chosen among the overlapping solutions with the same objective vector and left in the current population. The other overlapping solutions with the same objective vector are removed from the current population. As a result, each solution in the current population has a different location in the objective space. It should be noted that the overlapping solutions in the objective space are not necessary the same solution in the decision space. Thus we also examine the other strategy where the overlapping solutions in the decision space are removed from the current population except for a single solution. As a result, each solution in the current population has a different location in the decision space. The effect of removing overlapping solutions is examined through computational experiments where each removal strategy is combined into the NSGA-II algorithm.

1 Introduction

The design of evolutionary multiobjective optimization (EMO) algorithms has been discussed in the literature to find well-distributed Pareto-optimal or near Pareto-optimal solutions as many as possible (e.g., see Coello et al. [1] and Deb [2]). The handling of overlapping solutions, however, has not been discussed explicitly in many studies. This is mainly because the performance evaluation of EMO algorithms has been performed through computational experiments on multiobjective optimization problems with a large number of Pareto-optimal solutions. Since EMO algorithms usually have diversity-preserving mechanisms, many overlapping solutions are not likely to exist in each generation when they are applied to multiobjective optimization problems with continuous decision variables and/or many objective functions. On the other hand, the handling of overlapping solutions becomes an important issue in the application of EMO algorithms to multiobjective combinatorial optimization problems with only a few objective functions. In such an application, there may exist a large number of overlapping solutions in each generation as we will show in this paper through computational experiments on some test problems.

In this paper, we examine the effect of removing overlapping solutions on the performance of EMO algorithms. Overlapping solutions are removed from the current population. We examine two removal strategies of overlapping solutions. One removal strategy is performed in the objective space. Only a single solution among the overlapping solutions with the same objective vector is left in the current population. That is, overlapping solutions are removed so that each solution in the current population has a different location in the objective space. It should be noted that the overlapping solutions with the same objective vector are not necessary the same solution in the decision space. Thus we also examine the other strategy where the removal of overlapping solutions is performed in the decision space. Only a single solution among the overlapping solutions with the same decision vector is left in the current population. That is, overlapping solutions are removed so that each solution in the current population has a different location in the decision space. In this strategy, multiple solutions with the same objective vector can exist in the current population if they are not the same solution in the decision space.

The effect of removing overlapping solutions on the performance of EMO algorithms is examined through computational experiments on multiobjective 0/1 knapsack problems where each removal strategy is combined into the NSGA-II algorithm of Deb et al. [3]. First we show that there actually exist a large number of overlapping solutions in each generation in the application of the NSGA-II algorithm to two-objective 0/1 knapsack problems. Next we show that the removal of overlapping solutions improves the performance of the NSGA-II algorithm on those test problems especially in terms of the diversity of obtained non-dominated solutions. No clear differences are observed in the performance between the two removal strategies. Finally we show that these two removal strategies have a large effect on the performance of the NSGA-II algorithm when they are used together with a weighted sum-based tournament selection scheme of parent solutions with a large tournament size. This may be because a good balance is realized between the diversity-preserving effect of the removal strategies and the high selection pressure toward the Pareto front by the weighted sum-based parent selection scheme.

2 Handling of Overlapping Solutions

The original NSGA-II algorithm of Deb et al. [3] has no explicit mechanism to remove overlapping solutions while smaller fitness values are likely to be assigned to overlapping solutions than non-overlapping ones with the same non-dominated rank due to its diversity-preserving mechanism. In this section, we first briefly explain the NSGA-II algorithm. Then we explain two strategies for removing overlapping solutions, each of which is combined into the NSGA-II algorithm in computational experiments on multiobjective 0/1 knapsack problems in the next section.

2.1 The NSGA-II Algorithm

As in many evolutionary algorithms, first an initial population P_0 of size N is randomly generated in the NSGA-II algorithm [3]. That is, N initial solutions are randomly generated. It is possible in this initialization phase that some initial solutions are the same in the objective space. Those overlapping solutions can be the same solutions or different ones in the decision space. Next an offspring population Q_0 of size N is generated from the initial population P_0 by genetic operations (i.e., selection, recombination and mutation operations). It is possible in this genetic search phase that some of newly generated offspring solutions are overlapping in the objective space. It is also possible that some parent and offspring solutions are overlapping in the objective space. Then the initial population P_0 and its offspring population Q_0 are combined to construct a merged population R_0. The next population P_1 is constructed by choosing the best N solutions from the merged population R_0. The genetic operations for generating an offspring population and the generation update are iterated until a pre-specified stopping condition is satisfied (see Fig. 1).

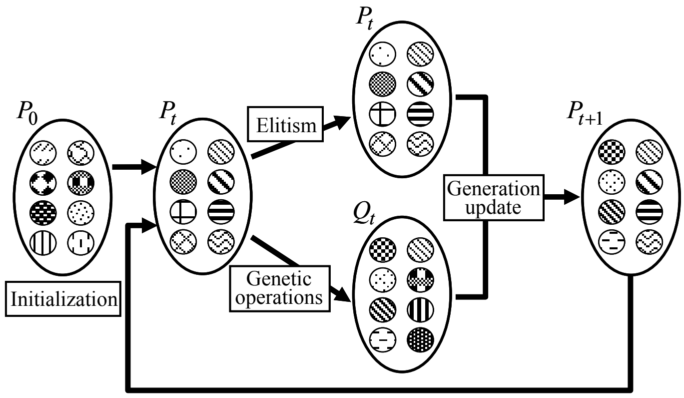


Fig. 1. Outline of the NSGA-II algorithm.

In the generation update phase (also in the parent selection phase), each solution is evaluated based on a non-dominated sorting scheme and a crowding measure. Lower rank solutions are viewed as being better than higher rank solutions in the NSGA-II algorithm according to the Pareto dominance relation. Among solutions with the same rank, solutions in less crowded regions are viewed as being better than those in more crowded regions according to the crowding measure in the NSGA-II algorithm. Thus overlapping solutions have lower fitness among solutions with the same rank. Low rank solutions in the merged population, however, are likely to be included in the next population even if they are overlapping solutions. When the number of non-dominated solutions in the merged population is smaller than N, all non-dominated solutions (i.e., all solutions with rank 1) are always included in the next population independent of overlapping or non-overlapping. Only when the number of non-dominated solutions in the merged population is larger than N, the diversity-preserving mechanism based on the crowding measure removes overlapping non-dominated solutions in the generation update phase of the NSGA-II algorithm.

2.2 Removal of Overlapping Solutions in the Objective Space

This removal strategy does not permit more than one solution with the same objective vector in each generation. Thus each solution in the current population has a different location in the objective space. In the generation phase of an initial population, we should generate N different solutions in the objective space. That is, we iterate the random generation procedure of initial solutions until N different solutions in the objective space are generated.

We do not have to modify the generation mechanism of an offspring population. That is, overlapping solutions can be generated as offspring solutions as in the original NSGA-II algorithm. This is because at least N different solutions are always included in the merged population in the case of this removal strategy (it should be noted that the current population always includes N different solutions in the objective space). In the generation update phase, overlapping solutions with the same objective vector are removed from the merged population except for a single solution. The single solution left in the merged population is randomly chosen among the overlapping solutions with the same objective vector. As a result, each solution in the merged population has a different location in the objective space. It should be noted that the merged population has at least N different solutions in the objective space because the current population includes N different solutions in the objective space. Each solution in the merged population is evaluated in the same manner as the original NSGA-II algorithm to choose the best N solutions from the merged population.

2.3 Removal of Overlapping Solutions in the Decision Space

The removal strategy in Subsection 2.2 can be performed in the decision space. That is, the existence of overlapping solutions in the objective space is permitted if they are not the same solutions in the decision space. Only when overlapping solutions have the same decision vector, those solutions are removed except for a single solution. As a result, each solution in the current population has a different location in the decision space while some solutions may have the same location in the objective space.

3 Computational Experiments

3.1 Conditions of Computational Experiments

As test problems, we use multiobjective 0/1 knapsack problems with k objectives, k constraints (i.e., k knapsacks) and n items in Zitzler & Thiele [13]:

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

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

where

$$f_i(\mathbf{x}) = \sum_{j=1}^{n} p_{ij} x_j , \quad i = 1,2, ..., k . \tag{3}$$

In this formulation, $\mathbf{x}$ is an n-dimensional binary vector (i.e., $(x_1, x_2, ..., x_n) \in \{0, 1\}^n$), 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. The k-objective n-item knapsack problem is referred to as a k-n knapsack problem in this paper. We examine 2-250, 2-500, 2-750, 3-500, and 4-500 knapsack problems. We employ the following parameter specifications:

Crossover probability (one-point crossover): 0.8,
Mutation probability (bit-flip mutation): $4/n$ (n: the number of items),
Population size: 150 (2-250), 200 (2-500), 250 (2-750), 250 (3-500), 300 (4-500),
Tournament size: 1, 2, 5, 10, 20,
Stopping condition: 500 generations.

The average performance is calculated for each test problem over 50 runs with different initial populations for each parameter specification (i.e., for each value of the tournament size).

Various performance measures have been proposed in the literature for evaluating a set of non-dominated solutions. As explained in Knowles & Corne [11], Okabe et al. [12], and Zitzler et al. [14], no performance measures can simultaneously evaluate various aspects of a solution set. Thus we visually show a solution set by a single run and the 50% attainment surface over 50 runs for each of the two-objective test problems. We also use four performance measures that are applicable to simultaneous comparison of many solution sets.

Let S be a solution set obtained by an EMO algorithm. The proximity of the solution set S to the Pareto front is evaluated by the generational distance (GD) defined as follows:

$$\mathrm{GD}(S) = \frac{1}{|S|} \sum_{\mathbf{x} \in S} \min\{d_{\mathbf{xy}} \mid \mathbf{y} \in S^*\} , \tag{4}$$

where S^* is a reference solution set (i.e., the set of all Pareto-optimal solutions) and $d_{\mathbf{xy}}$ is the distance between a solution $\mathbf{x}$ and a reference solution $\mathbf{y}$ in the k-dimensional objective space:

$$d_{\mathbf{xy}} = \sqrt{(f_1(\mathbf{x}) - f_1(\mathbf{y}))^2 + \cdots + (f_k(\mathbf{x}) - f_k(\mathbf{y}))^2} . \tag{5}$$

For evaluating both the diversity of solutions in the solution set S and their convergence to the Pareto front, we calculate the $\mathrm{D1_R}$ measure defined as follows:

$$\mathrm{D1_R}(S) = \frac{1}{|S^*|} \sum_{\mathbf{y} \in S^*} \min\{d_{\mathbf{xy}} \mid \mathbf{x} \in S\} \ . \tag{6}$$

It should be noted that $\mathrm{D1_R}(S)$ is the average distance from each reference solution $\mathbf{y}$ in S^* to its nearest solution in S while GD(S) in (4) is the average distance from each solution $\mathbf{x}$ in S to its nearest reference solution in S^* . The generational distance evaluates the proximity of the solution set S to the reference solution set S^* . On the other hand, the $\mathrm{D1_R}$ measure evaluates how well the solution set S approximates the reference solution set S^* . Since all the Pareto-optimal solutions are known for the 2-250 and 2-500 test problems, we can use them as the reference solution set S^* for each test problem. Since true Pareto-optimal solutions are not available for the other test problems, we use as S^* a set of near Pareto-optimal solutions obtained for each test problem using much longer CPU time and much larger memory storage than computational experiments reported in this paper.

The spread measure is calculated as follows:

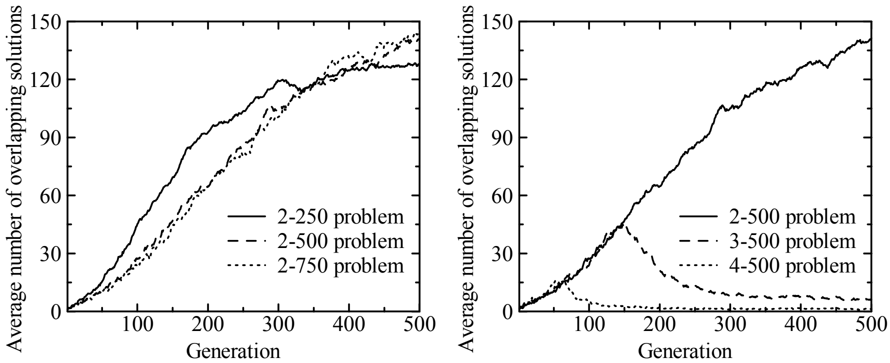
$$Spread = \sum_{i=1}^{k} [\max_{\mathbf{x} \in S_i} \{f_i(\mathbf{x})\} - \min_{\mathbf{x} \in S_i} \{f_i(\mathbf{x})\}] \, . \tag{7}$$

The hypervolume measure calculates the volume of the dominated region by the solution set S_i in the objective space. The ratio of non-dominated solutions is calculated for a solution set with respect to other solution sets. Let us assume that we have m solution sets S_1, S_2, ..., S_m. By merging these solution sets, we construct another solution set S as $S = S_1 \cup S_2 \cup \ ... \ \cup S_m$. Let S_{ND} be the set of non-dominated solutions in S . The ratio of non-dominated solutions is calculated for the solution set S_i as $|S_i \cap S_{ND}| / |S_i|$ where $|S_i|$ denotes the cardinality of S_i (i.e., the number of solutions in S_i).

3.2 Number of Overlapping Solutions in the Original NSGA-II Algorithm

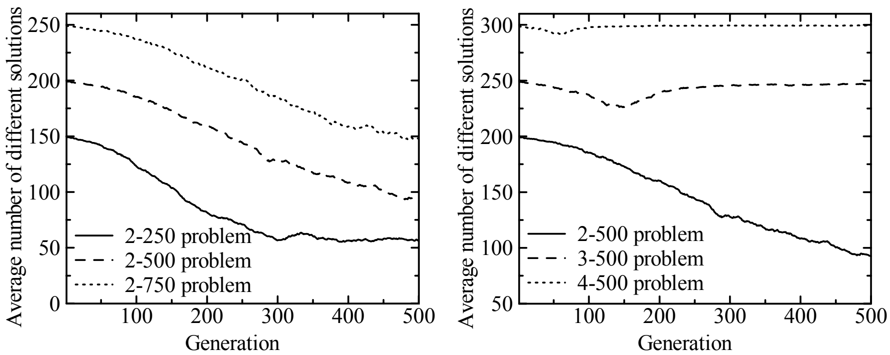
The average number of overlapping solutions in each generation of the original NSGA-II algorithm is shown in Fig. 2 (a) for the two-objective test problems and Fig. 2 (b) for the 500-item test problems. We can see from Fig. 2 (a) that the average number of overlapping solutions monotonically increases during the multiobjective evolution by the NSGA-II algorithm for all the two-objective test problems except for the final stage of the evolution for the 2-250 test problem. The final population includes about 135 overlapping solutions on the average. On the other hand, we can see from Fig. 2 (b) that the number of overlapping solutions is not large in the case of the three-objective and four-objective test problems.

In the same manner as Fig. 2, we show the average number of different solutions in the objective space in Fig. 3. From Fig. 3 (a), we can see that the number of different solutions monotonically decreases during the multiobjective evolution by the NSGA-II algorithm except for the final stage of the evolution for the 2-250 test problem. On the other hand, we can see from Fig. 3 (b) that the number of different solutions does not decrease during the computational experiments for the 3-500 and 4-500 test problems. Experimental results in Fig. 2 and Fig. 3 suggest that the issue of overlapping solutions is very important for multiobjective combinatorial optimization problems with two objective functions. When the number of objective functions is large, many overlapping solutions do not exist in each generation.



(a) Effect of the number of items. (b) Effect of the number of objectives.

Fig. 2. Average number of overlapping solutions in each generation.



(a) Effect of the number of items. (b) Effect of the number of objectives.

Fig. 3. Average number of different solutions in the objective space.

3.3 Performance Evaluation of Modified NSGA-II Algorithms

Experimental results by the original NSGA-II algorithm and its two variants are summarized in Figs. 4-6. In these figures, the two removal strategies are labeled as "Objective space" and "Decision space" depending on the space where the removal of overlapping solutions is performed. The original NSGA-II algorithm is labeled as "Standard". Each dashed line in these figures shows the average result by the original NSGA-II with the tournament size 2. From these figures, we can see that all of the four performance measures are improved by removing overlapping solutions when the tournament size is 2. On the other hand, the removal of overlapping solutions degrades some performance measures when the tournament size is 20. It should be noted in Fig. 4 that the $D1_R$ measure, the spread measure, and the hypervolume measure are improved by increasing the tournament size while the GD measure is degraded. That is, the increase in the tournament size improves the diversity of solutions and degrades the convergence to the Pareto front. As a result, the positive effect of removing overlapping solutions on the diversity of solutions is not clear when the tournament size is large.

We visually examine the validity of the above discussion by depicting a solution set obtained by a single run of the original NSGA-II algorithm and the 50% attainment surface over its 50 runs in Fig. 7 where two specifications of the tournament size are examined (i.e., 2 and 20) for the 2-250 test problem. From this figure, we can see that the increase in the tournament size increases the diversity of solutions and degrades the convergence to the Pareto front.

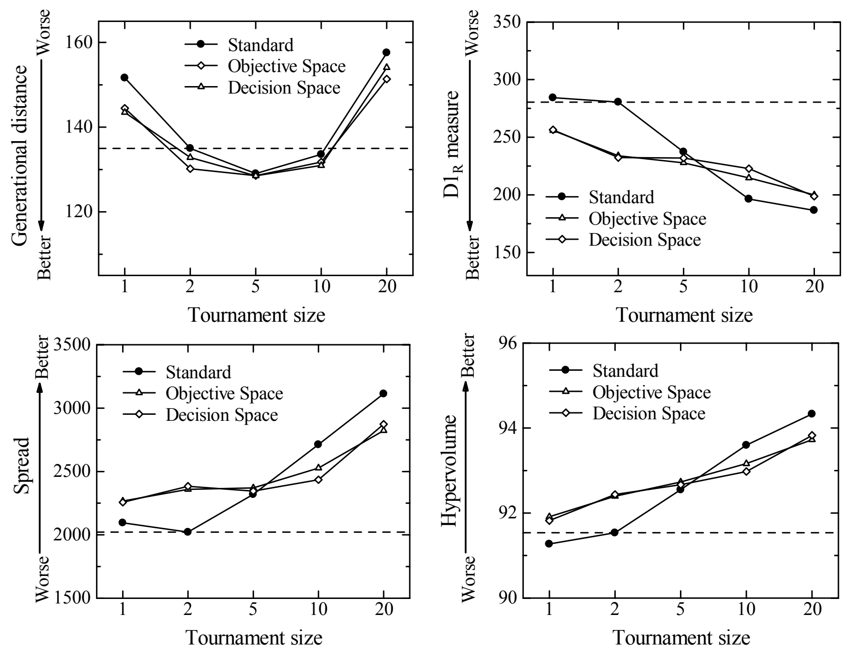


Fig. 4. Average results over 50 runs on 2-250 problem.

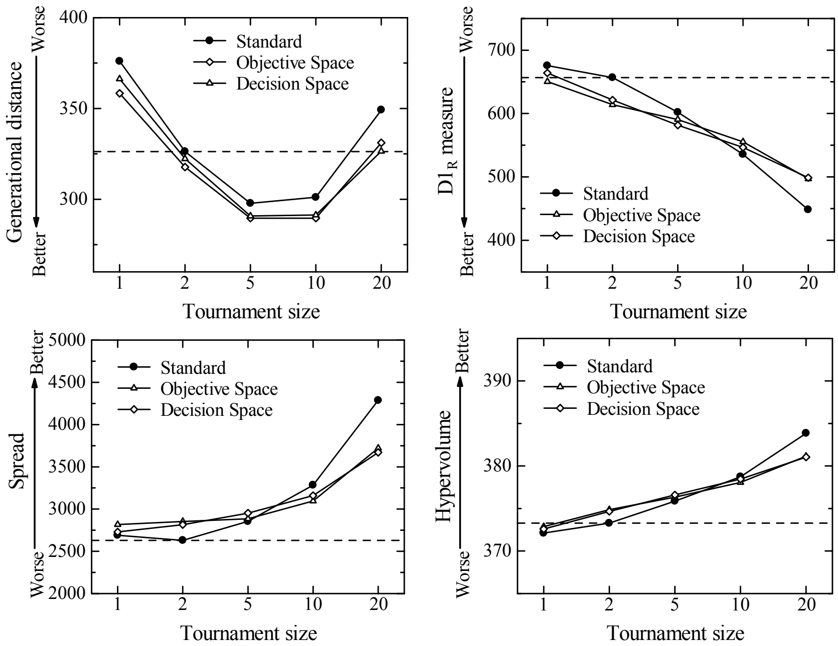


Fig. 5. Average results over 50 runs on 2-500 problem.

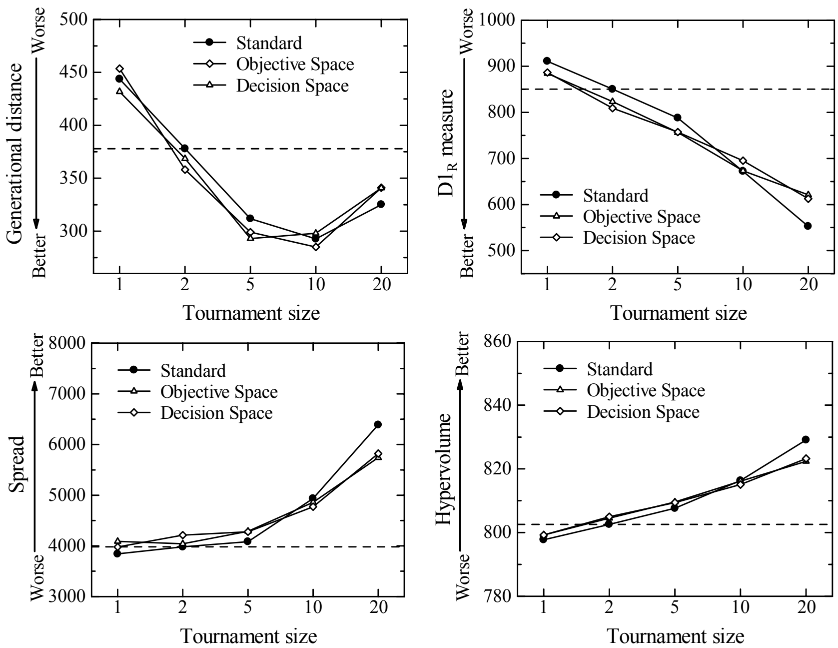
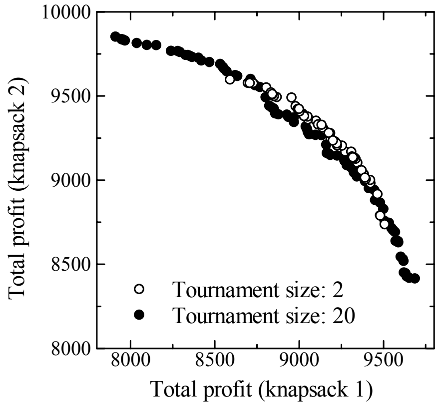
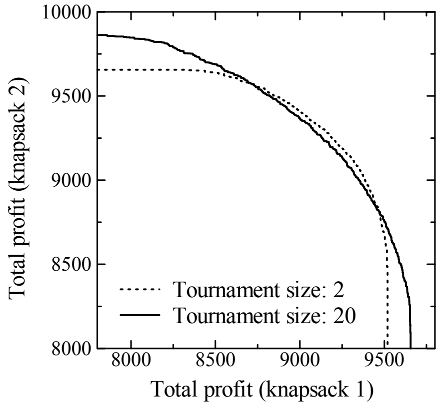


Fig. 6. Average results over 50 runs on 2-750 problem.



(a) Solution sets by single run.



(b) 50% attainment surface.

Fig. 7. Effect of the tournament size on obtained non-dominated solutions.

From the above-mentioned computational experiments, we can see that the removal of overlapping solutions does not work well together with a large tournament size. This is because both the removal of overlapping solutions and the increase in the tournament size have the same effect on the multiobjective evolution by the NSGA-II algorithm (i.e., they both increase the diversity of solutions). In order to achieve a good balance between the diversity of solutions and the convergence to the Pareto front, we examine the use of the following weighted sum of multiple objectives as a scalar fitness function in the selection phase of the NSGA-II algorithm:

$$f(\mathbf{x}, \boldsymbol{\lambda}) = \sum_{i=1}^{k} \lambda_i f_i(\mathbf{x}), \tag{8}$$

where $\lambda = (\lambda_1, \lambda_2, ..., \lambda_k)$ is a weight vector:

$$\forall i \;\; \lambda_i \geq 0 \quad \text{and} \quad \sum_{i=1}^{k} \lambda_i = 1. \tag{9}$$

The weight vector is randomly updated whenever a pair of parent solutions is to be chosen. This scalar fitness function was used in multiobjective genetic local search algorithms (MOGLS) of Ishibuchi et al. [5, 8] and Jaszkiewicz [9, 10].

Experimental results are shown in Figs. 8-10. From these figures, we can see that all the four performance measures are improved by increasing the tournament size and removing overlapping solutions. It should be noted that the effect of removing overlapping solutions is much larger in Figs. 8-10 with the weighted sum-based fitness function than in Figs. 4-6 with the fitness function of the original NSGA-II algorithm. This may be because the lack of a diversity-preserving mechanism in the weighted sum-based fitness function is compensated by the removal of overlapping solutions. This discussion is supported by obtained non-dominated solutions in Fig. 11 with the removal strategy in the objective space.

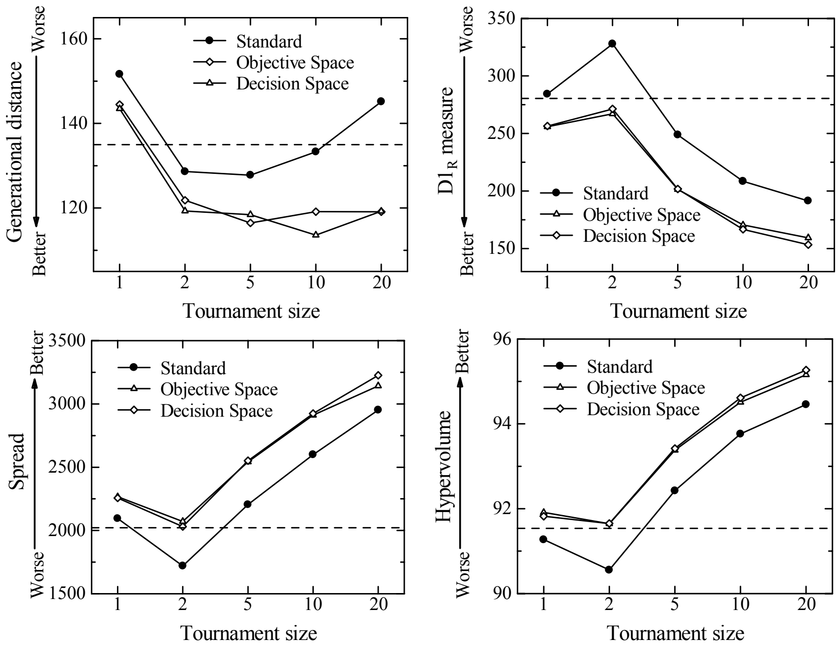


Fig. 8. Average results by the weighted sum-based fitness function for the 2-250 problem.

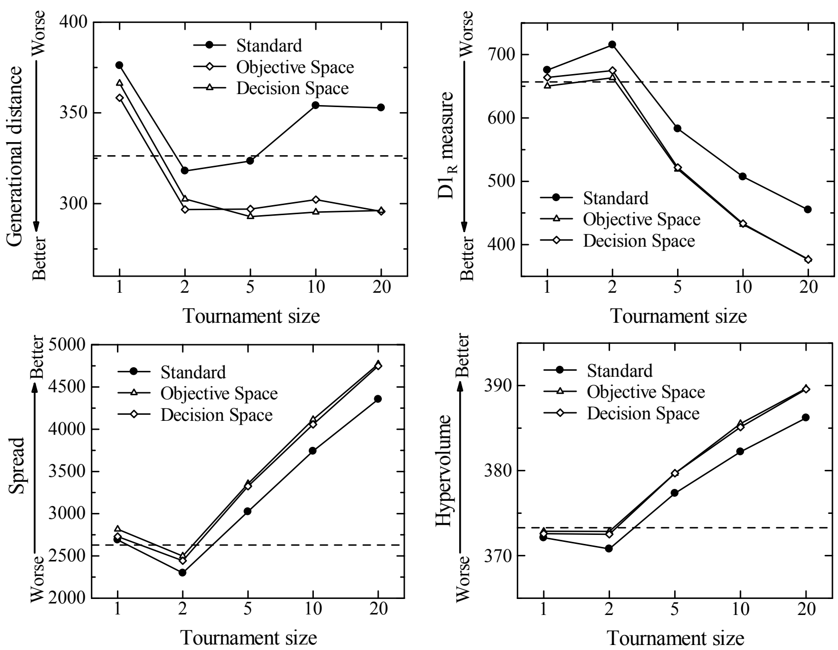


Fig. 9. Average results by the weighted sum-based fitness function for the 2-500 problem.

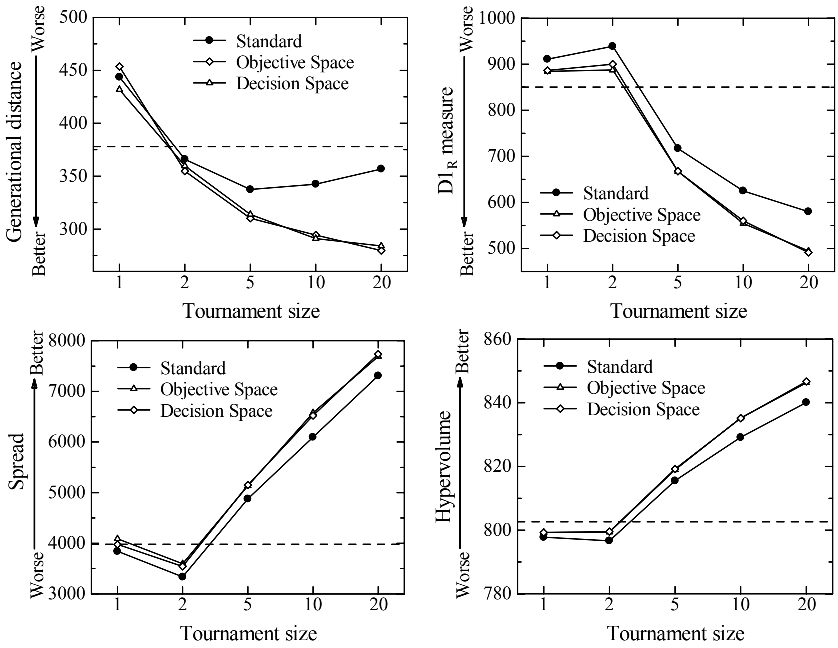
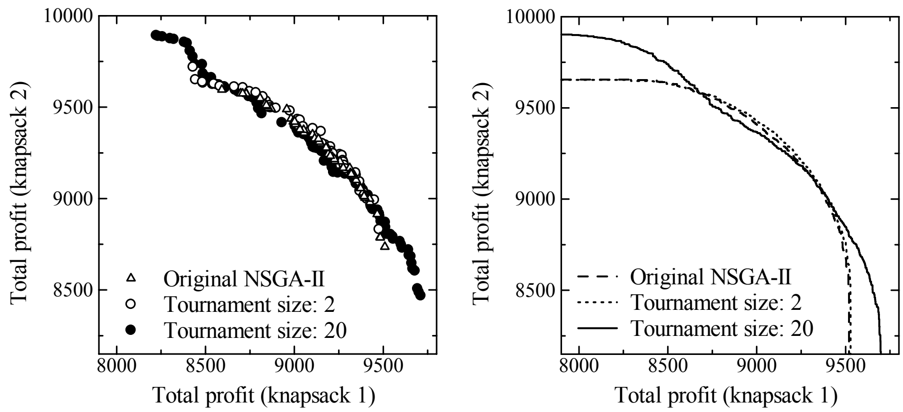


Fig. 10. Average results by the weighted sum-based fitness function for the 2-750 problem.



(a) Solution sets by single run.

(b) 50% attainment surface.

Fig. 11. Effect of the tournament size on obtained non-dominated solutions in the case of the weighted sum-based fitness function with the removal of overlapping solutions.

4 Concluding Remarks

In this paper, we examined the handling of overlapping solutions in EMO algorithms through computational experiments on multiobjective 0/1 knapsack problems using the NSGA-II algorithm. First we showed that the number of overlapping solutions increased during the multiobjective evolution by the NSGA-II algorithm for two-objective knapsack problems. We also showed that the number of overlapping solutions was very small in the case of three-objective and four-objective knapsack problems. These results suggest that the issue of overlapping solutions is very important only for multiobjective combinatorial optimization problems with two objective functions. Next we showed that the performance of the NSGA-II algorithm with binary tournament selection was improved by removing overlapping solutions for two-objective knapsack problems. When we increased the tournament size, the diversity of non-dominated solutions obtained by the NSGA-II algorithm increased and the positive effect of removing overlapping solutions disappeared. This may be because the increase in the tournament size and the removal of overlapping solutions have the same effect on the multiobjective evolution by the NSGA-II algorithm. That is, they both increase the diversity of solutions. Finally we suggested the use of the weighted sum-based fitness function together with the removal of overlapping solutions. In this case, the increase in the tournament size improved the convergence to the Pareto front while the removal of overlapping solutions improved the diversity of solutions. As a result, a good balance between the convergence and the diversity was achieved by increasing the tournament size. Experimental results on two-objective knapsack problems showed that the performance of the NSGA-II algorithm was improved by the weighted sum-based tournament selection with a large tournament size and the removal of overlapping solutions.

In computational experiments in this paper, we only used multiobjective 0/1 knapsack problems as test problems. As future studies, we are planning to examine the effect of removing overlapping solutions through computational experiments on other multiobjective combinatorial optimization problems (e.g., multiobjective rule selection problems [7, 6] and multiobjective flowshop scheduling problems [5, 8]) as well as multiobjective function optimization problems (e.g., standard multiobjective test problems in Deb et al. [4]). Such computational experiments will clarify how the handling of overlapping solutions is important in the application of EMO algorithms to different types of multiobjective optimization problems. Computational experiments were performed using only the NSGA-II algorithm in this paper. We will also examine the effect of removing overlapping solutions on the performance of other EMO algorithms in future studies.

References

1. Coello Coello, C. A., Van Veldhuizen, D. A., and Lamont, G. B.: Evolutionary Algorithms for Solving Multi-Objective Problems, Kluwer Academic Publishers, Boston (2002).
2. Deb, K.: Multi-Objective Optimization Using Evolutionary Algorithms, John Wiley & Sons, Chichester (2001).
3. Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T.: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. IEEE Trans. on Evolutionary Computation **6** (2002) 182-197.
4. Deb, K., Thiele, L., Laumanns, M., and Zitzler, E.: Scalable Multiobjective Optimization

Test Problems. Proc. of 2002 Congress on Evolutionary Computation (2002) 825-830.
5. Ishibuchi, H. and Murata, T.: A Multi-Objective Genetic Local Search Algorithm and Its Application to Flowshop Scheduling. IEEE Trans. on Systems, Man, and Cybernetics - Part C: Applications and Reviews **28** (1998) 392-403.
6. Ishibuchi, H., and Yamamoto, T.: Effects of Three-Objective Genetic Rule Selection on the Generalization Ability of Fuzzy Rule-Based Systems. Lecture Notes in Computer Science **2632**: Evolutionary Multi-Criterion Optimization, Springer, Berlin (2003) 608-622.
7. Ishibuchi, H., and Yamamoto, T.: Fuzzy Rule Selection by Multi-Objective Genetic Local Search Algorithms and Rule Evaluation Measures in Data Mining. Fuzzy Sets and Systems 141 (2004) 59-88.
8. Ishibuchi, H., Yoshida, T., and Murata, T.: Balance between Genetic Search and Local Search in Memetic Algorithms for Multiobjective Permutation Flowshop Scheduling. IEEE Trans. on Evolutionary Computation **7** (2003) 204-223.
9. Jaszkiewicz, A.: Comparison of Local Search-based Metaheuristics on the Multiple Objective Knapsack Problem. Foundations of Computing and Decision Sciences **26** (2001) 99-120.
10. Jaszkiewicz, A.: On the Performance of Multiple-Objective Genetic Local Search on the 0/1 Knapsack Problem - A Comparative Experiment. IEEE Trans. on Evolutionary Computation **6** (2002b) 402-412.
11. Knowles, J. D. and Corne, D. W.: On Metrics for Comparing Non-dominated Sets. Proc. of 2002 Congress on Evolutionary Computation (2002) 711-716.
12. Okabe, T., Jin, Y., and Sendhoff, B.: A Critical Survey of Performance Indices for Multi-Objective Optimization. Proc. of 2003 Congress on Evolutionary Computation (2003) 878-885.
13. Zitzler, E. and Thiele, L.: Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. IEEE Trans. on Evolutionary Computation 3 (1999) 257-271.
14. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., and da Fonseca, V. G.: Performance Assessment of Multiobjective Optimizers: An Analysis and Review. IEEE Trans. on Evolutionary Computation **7** (2003) 117- 132.