Mating Scheme for Controlling the Diversity-Convergence Balance for Multiobjective Optimization

Hisao Ishibuchi and Youhei Shibata

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

Abstract. The aim of this paper is to clearly demonstrate the potential ability of a similarity-based mating scheme to dynamically control the balance between the diversity of solutions and the convergence to the Pareto front in evolutionary multiobjective optimization. The similarity-based mating scheme chooses two parents in the following manner. For choosing one parent (say Parent A), first a pre-specified number of candidates (say α candidates) are selected by iterating the standard fitness-based binary tournament selection. Then the average solution of those candidates is calculated in the objective space. The most similar or dissimilar candidate to the average solution is chosen as Parent A. When we want to increase the diversity of solutions, the selection probability of Parent A is biased toward extreme solutions by choosing the most dissimilar candidate. The strength of this diversity-preserving effort is adjusted by the parameter α. We can also bias the selection probability toward center solutions by choosing the most similar candidate when we want to decrease the diversity. The selection probability of the other parent (i.e., the mate of Parent A) is biased toward similar solutions to Parent A for increasing the convergence speed to the Pareto front. This is implemented by choosing the most similar one to Parent A among a pre-specified number of candidates (say β candidates). The strength of this convergence speed-up effort is adjusted by the parameter β. When we want to increase the diversity of solutions, the most dissimilar candidate to Parent A is chosen as its mate. Our idea is to dynamically control the diversity-convergence balance by changing the values of two control parameters α and β during the execution of evolutionary multiobjective optimization algorithms. We examine the effectiveness of our idea through computational experiments on multiobjective knapsack problems.

1 Introduction

The goal of evolutionary multiobjective optimization (EMO) is to efficiently find Pareto-optimal (or near Pareto-optimal) solutions of multiobjective optimization problems as many as possible. There are two sub-goals for achieving this goal: maintaining the diversity of solutions in each population and increasing the

convergence speed of solutions to the Pareto front. In this paper, we propose an idea of using a similarity-based mating scheme for controlling the balance between these two sub-goals during the execution of EMO algorithms.

EMO algorithms have been applied to various problems for efficiently finding their Pareto-optimal or near Pareto-optimal solutions. Recent EMO algorithms usually share some common ideas such as elitism, fitness sharing and Pareto ranking for improving both the diversity of solutions and the convergence speed to the Pareto front (e.g., see Coello et al. [1] and Deb [2]). In some studies, local search was combined with EMO algorithms for further improving the convergence speed to the Pareto front [10], [13], [14]. While mating restriction has been often discussed in the literature, its effect has not been clearly demonstrated. As a result, it is not used in many EMO algorithms as pointed out in some reviews on EMO algorithms [5], [20], [22]. Mating restriction was suggested by Goldberg [6] and used in EMO algorithms by Hajela & Lin [7] and Fonseca & Fleming [4]. The basic idea of mating restriction is to ban the crossover of dissimilar parents from which good offspring are not likely to be generated. The necessity of mating restriction in EMO algorithms was stressed in Jaszkiewicz [14]. On the other hand, Zitzler & Thiele [21] reported that no improvement was achieved by mating restriction in their computational experiments. Moreover, there was also an argument for the selection of dissimilar parents. Horn et al. [8] argued that information from very different types of tradeoffs could be combined to yield other kinds of good tradeoffs. Schaffer [19] examined the selection of dissimilar parents but observed no improvement.

A similarity-based mating scheme was proposed in Ishibuchi & Shibata [11] for examining positive and negative effects of mating restriction on the search ability of EMO algorithms. In their mating scheme, one parent (say Parent A) was chosen by the standard fitness-based binary tournament scheme while its mate (say Parent B) was chosen among a pre-specified number of candidates (say β candidates) based on their similarity or dissimilarity to Parent A. Those candidates were selected by iterating the standard fitness-based binary tournament selection β times. Almost the same idea was independently proposed in Huang [9] where Parent B was chosen from two candidates (i.e., the value of β was fixed as $\beta = 2$). Ishibuchi & Shibata [12] extended their similarity-based mating scheme as shown in Fig. 1. That is, first a pre-specified number of candidates (say α candidates) were selected by iterating the standard fitness-based binary tournament selection α times. Next the average solution of those candidates was calculated in the objective space. The most dissimilar candidate to the average solution was chosen as Parent A. On the other hand, the most similar one to Parent A among β candidates was chosen as Parent B.

In this paper, we further extend this mating scheme in the following manner:

(1) Selection of Parent A: We examine the choice of the most similar candidate to the average solution as well as the choice of the most dissimilar one.

(2) Selection of Parent B: We examine the choice of the most dissimilar candidate to Parent A as well as the choice of the most similar one.

(3) Parameters α and β : We do not assume that these parameters are fixed during the execution of EMO algorithms. That is, they are handled as control parameters.
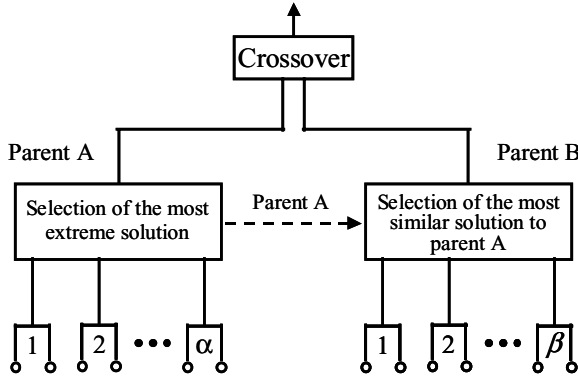

Fig. 1. Mating scheme in Ishibuchi & Shibata [12].

This paper is organized as follows. In Section 2, we describe the motivation behind the above-mentioned extensions of the mating scheme. The extended mating scheme is described in Section 3. In Section 4, we perform computational experiments on multiobjective knapsack problems by combining the extended mating scheme with the NSGA-II algorithm [3]. Finally we conclude this paper in Section 5.

2 Motivation

In the implementation of EMO algorithms, it is very important to find an appropriate balance between the diversity of solutions and the convergence to the Pareto front. Let us consider a two-objective maximization problem in Fig. 2. For increasing the diversity, the current population should be expanded in the direction of the two arrows labeled as D in Fig. 2 (a). On the other hand, the current population should be driven in the direction of the other arrow C in Fig. 2 (a) for the convergence to the Pareto front. When the diversity-convergence balance is appropriate, good solution sets may be obtained as shown in Fig. 2 (b). It is, however, not easy to find an appropriate balance because each problem and each EMO algorithm may require its own specification. Moreover, different specifications may be required in different stages of evolution. For example, it may be a good idea to emphasize the diversity in the early state of evolution and the convergence in the later stage of evolution.

This paper is motivated by Knowles & Corne [16], [17] where they proposed an idea of designing a memetic EMO algorithm based on the landscape analysis of multiobjective optimization problems. For example, if the search along the Pareto front is much easier than the convergence to the Pareto front, it seems to be a good strategy to first drive the population to the Pareto front as close as possible (see Fig. 3

(a)). On the other hand, if the search along the Pareto front is much more difficult than the convergence to the Pareto front, it seems to be a good strategy to first increase the diversity of solutions as large as possible (see Fig. 3 (b)). These discussions suggest the necessity to dynamically control the diversity-convergence balance during the execution of EMO algorithms. Thus we extend the similarity-based mating scheme in Fig. 1 in the above-mentioned manner in Section 1, which is explained in the next section. The aim of this paper is to clearly demonstrate the potential ability of the similarity-based mating scheme to dynamically control the diversity-convergence balance during the execution of EMO algorithms.
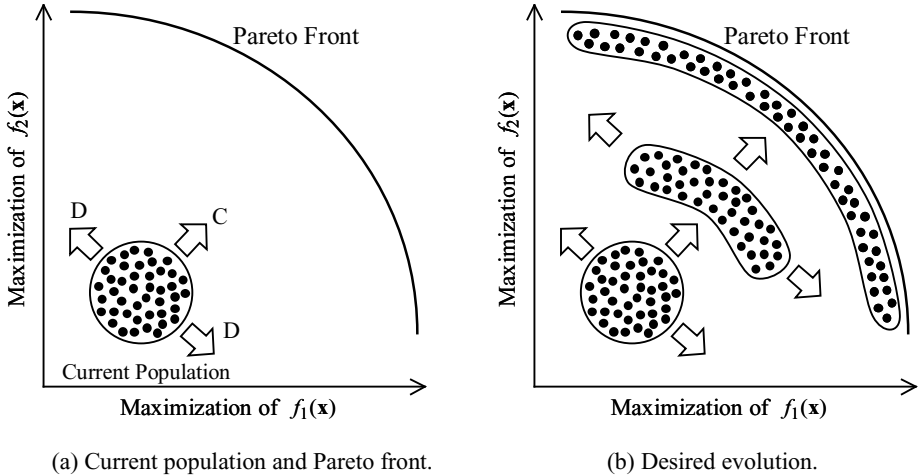

(a) Current population and Pareto front.   (b) Desired evolution.

Fig. 2. Search directions for increasing the diversity of solutions (i.e., Arrows D) and for improving the convergence speed to the Pareto front (i.e., Arrow C).
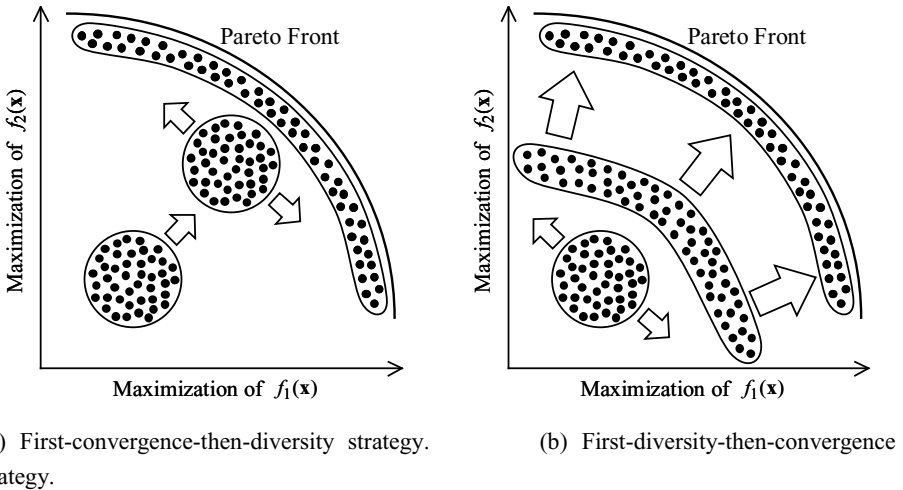

(a) First-convergence-then-diversity strategy.   (b) First-diversity-then-convergence strategy.

Fig. 3. Two search strategies. Emphasis is first placed on the convergence then on the diversity in (a) while emphasis is first placed on the diversity then on the convergence in (b).

3 Similarity-Based Mating Scheme

We explain the extended similarity-based mating scheme using the following n-objective optimization problem:

$$\text{Optimize } \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_n(\mathbf{x})) \text{ subject to } \mathbf{x} \in \mathbf{X}, \tag{1}$$

where $\mathbf{f}(\mathbf{x})$ is the objective vector, $f_i(\mathbf{x})$ is the i-th objective to be minimized or maximized, $\mathbf{x}$ is the decision vector, and $\mathbf{X}$ is the feasible region in the decision space. The distance between two solutions $\mathbf{x}$ and $\mathbf{y}$ is measured in the objective space by the Euclidean distance between $\mathbf{f}(\mathbf{x})$ and $\mathbf{f}(\mathbf{y})$.

As shown in the left-hand side of Fig. 1, the standard fitness-based binary tournament selection with replacement is iterated α times for choosing α candidates (say $\mathbf{x}_1$, $\mathbf{x}_2$, ..., $\mathbf{x}_\alpha$) among which the first parent (i.e., Parent A in Fig. 1) is selected. Then the average solution of the α candidates is calculated in the objective space as

$$\bar{\mathbf{f}}(\mathbf{x}) = (\bar{f}_1(\mathbf{x}), \bar{f}_2(\mathbf{x}), ..., \bar{f}_n(\mathbf{x})), \tag{2}$$

where

$$\bar{f}_i(\mathbf{x}) = \frac{1}{\alpha} \sum_{j=1}^{\alpha} f_i(\mathbf{x}_j) \text{ for } i = 1,2,...,n. \tag{3}$$

Finally the most similar or dissimilar candidate to the average solution is selected as Parent A. The similarity is measured by the Euclidean distance to the average solution $\bar{\mathbf{f}}(\mathbf{x})$ in the objective space. When the value of α is specified as $\alpha = 1$, the choice of the first parent is the same as the standard fitness-based binary tournament selection. The case of $\alpha = 2$ is also actually the same as the case of $\alpha = 1$ because two candidates always have the same distance from their average solution. In this case, one candidate is randomly chosen (i.e., random tiebreak). The selection probability of the first parent is biased toward extreme solutions or center solutions only when $\alpha \geq 3$.

On the other hand, the standard fitness-based binary tournament selection with replacement is iterated β times for choosing β candidates of the second parent (i.e., Parent B) as shown in the right-hand side of Fig. 1. Then the most similar or dissimilar candidate to the first parent (i.e., Parent A in Fig. 1) is chosen as Parent B. In this manner, similar or dissimilar parents are recombined in our mating scheme.

As shown in Table 1, our mating scheme is divided into nine (i.e., 3×3) operation modes depending on the strategies to choose Parent A and Parent B. The first operation mode with $\alpha = 1$ and $\beta = 1$ is the same as the standard fitness-based binary tournament selection. The first three operation modes were examined in [9] and [11] while the eighth operation mode was recommended in [12]. In those studies, the values of α and β were fixed throughout the execution of EMO algorithms. In

this paper, we examine all the nine operation modes where the values of α and β are not fixed but variable. Moreover our mating scheme can move from one operation mode to another during the execution of EMO algorithms.

Table 1. Nine operation modes of our mating scheme.

Operation mode	Selection of Parent A	Selection of Parent B
1	No bias: $\alpha = 1$	No bias: $\beta = 1$
2	No bias: $\alpha = 1$	Similar solution ($\beta \geq 2$)
3	No bias: $\alpha = 1$	Dissimilar solution ($\beta \geq 2$)
4	Center solution: Similar ($\alpha \geq 3$)	No bias: $\beta = 1$
5	Center solution: Similar ($\alpha \geq 3$)	Similar solution ($\beta \geq 2$)
6	Center solution: Similar ($\alpha \geq 3$)	Dissimilar solution ($\beta \geq 2$)
7	Extreme solution: Dissimilar ($\alpha \geq 3$)	No bias: $\beta = 1$
8	Extreme solution: Dissimilar ($\alpha \geq 3$)	Similar solution ($\beta \geq 2$)
9	Extreme solution: Dissimilar ($\alpha \geq 3$)	Dissimilar solution ($\beta \geq 2$)

4 Computational Experiments

4.1 Conditions of Computational Experiments

We combined our mating scheme with the NSGA-II [3], which was used for choosing candidate parents in our mating scheme. First we examined $7 \times 7 = 49$ combinations of constant values of α and β: $\alpha = 1$, $\alpha = 3, 5, 9$ (similar), $\alpha = 3, 5, 9$ (dissimilar) and $\beta = 1$, $\beta = 3, 5, 9$ (similar), $\beta = 3, 5, 9$ (dissimilar). Then we examined the effect of changing the values of α and β during the execution of the NSGA-II.

As test problems, we used four knapsack problems in Zitzler & Thiele [22]: 2-250 (i.e., two-objective 250-item), 2-500, 3-250 and 3-500 problems. Each solution for an m-item problem was coded as a binary string of the length m. Each string was evaluated in the same manner as in Zitzler & Thiele [22]. The NSGA-II algorithm with our mating scheme was applied to the four knapsack problems under the following parameter specifications: Crossover probability: 0.8, mutation probability: $1/m$, population size: 200, and stopping condition: 2000 generations.

Various performance measures have been proposed for evaluating a set of non-dominated solutions in the literature. As explained in [15], [18], [23], no performance measure can simultaneously evaluate various aspects of a solution set. In this paper, we use three performance measures: the generational distance (GD), the $D1_R$ measure, and a spread measure. The GD measure is the average distance from each solution in the solution set to its nearest Pareto-optimal solution. This measure has

been often used for evaluating the convergence to the Pareto front. On the other hand, the $D1_R$ measure is the average distance from each Pareto-optimal solution to its nearest solution in the solution set. This measure has been used in some studies for simultaneously evaluating both the convergence and the diversity. As a spread measure of the solution set S, we use the sum of the width for each objective:

$$Spread = \sum_{i=1}^{n} [\max_{\mathbf{x} \in S} \{f_i(\mathbf{x})\} - \min_{\mathbf{x} \in S} \{f_i(\mathbf{x})\}], \tag{4}$$

where n is the number of objectives.

The GD and $D1_R$ measures need all Pareto-optimal solutions of each test problem. For the 2-250 and 2-500 test problems, the Pareto-optimal solutions are available from the homepage of the first author of [22]. For the 3-250 and 3-500 test problems, we found near Pareto-optimal solutions using the SPEA [22], the NSGA-II [3] and a single objective genetic algorithm using much longer CPU time and larger memory storage (e.g., 30000 generations with the population size 400 for the NSGA-II) than the other computational experiments.

4.2   Experimental Results with Constant Parameters

The NSGA-II algorithm with our mating scheme was applied to the four test problems using the 7×7 combinations of α and β. For each combination, we performed ten runs from different initial populations for each test problem.

Average results over ten runs for the 2-500 and 3-500 test problems are summarized in Figs. 4-6 where smaller values (i.e., shorter bars) of the GD and $D1_R$ measures mean better results in Fig. 4 and Fig. 5 while larger values (i.e., longer bars) of the spread measure mean better results in Fig. 6. In each figure, the original NSGA-II algorithm corresponds to the bar at the center with $\alpha = 1$ and $\beta = 1$, which is depicted by a white bar. The effect of the selection of the first parent (i.e., the specification of α) is clear in Fig. 6. That is, the bias toward extreme solutions (i.e., the use of a large value of α (dissimilar)) increased the spread of solution sets. On the other hand, the recombination of similar parents (i.e., the use of a large value of β (similar)) improved the convergence to the Pareto front as shown in Fig. 4 with some exceptions in Fig. 4 (a). In the case of α (similar) and β (similar) in Fig. 4 (a), too small diversity of solutions had a bad effect on the evolution of solutions (see Fig. 6 (a)). Experimental results on the 2-250 and 3-250 test problems were very similar to those on the 2-500 and 3-500 test problems, respectively.

For visually demonstrating the effect of our mating scheme on the evolution of solutions, we show some intermediate and final solution sets in Fig. 7 where each solution set is a result of a single run on the 2-500 test problem using each combination of α and β. In Fig. 7, we also depict the true Pareto front for comparison. From Fig. 7, we can see that the search ability of the original NSGA-II

algorithm in Fig. 7 (a) was significantly improved by our mating scheme in terms of the diversity of solutions. More specifically, we can see that the recombination of dissimilar parents (i.e., the use of $\beta = 9$ (dissimilar)) increased the diversity of solutions in Fig. 7 (b) if compared with Fig. 7 (a) obtained by the original NSGA-II algorithm. The diversity of solutions was further increased in Fig. 7 (d) by the choice of extreme solutions (i.e., the use of $\alpha = 9$ (dissimilar)). It should be noted that the convergence was not seriously deteriorated in Fig. 7 (d). This is because the recombination of similar solutions (i.e., the use of $\beta = 9$ (similar)) improved the convergence as shown in Fig. 4 when the selection probability of the first parent was biased toward extreme solutions (i.e., when α was specified as $\alpha \geq 3$ (dissimilar)).
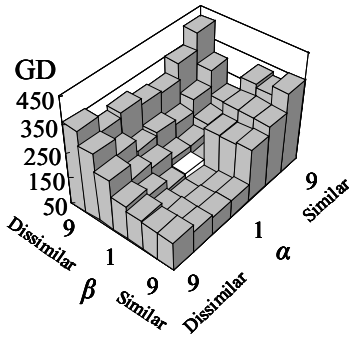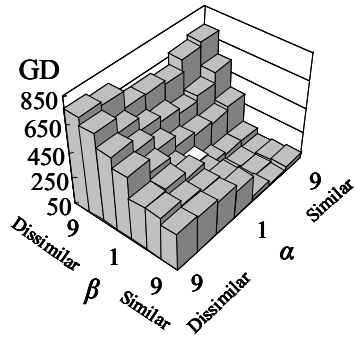
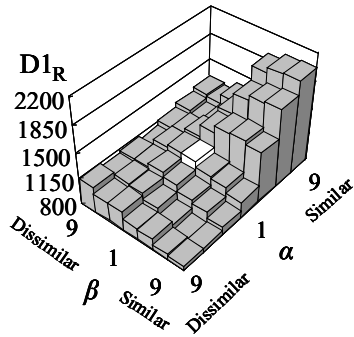(a) 2-500 test problem.   (b) 3-500 test problem.

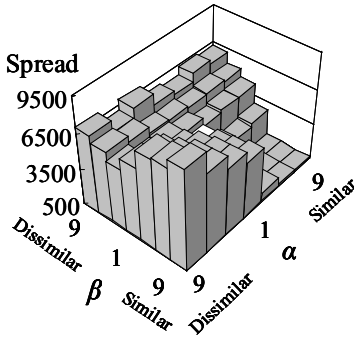**Fig. 4.** Average values of the GD measure.
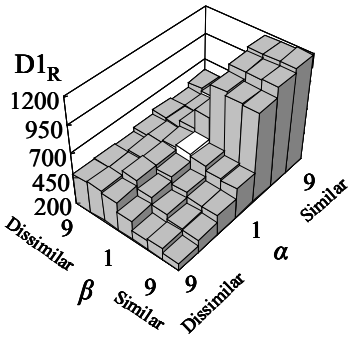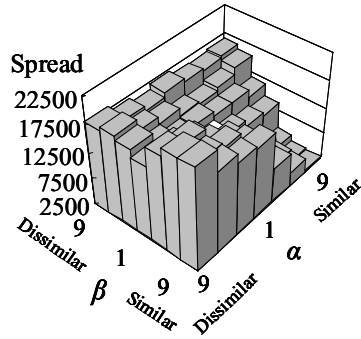

(a) 2-500 test problem.   (b) 3-500 test problem.
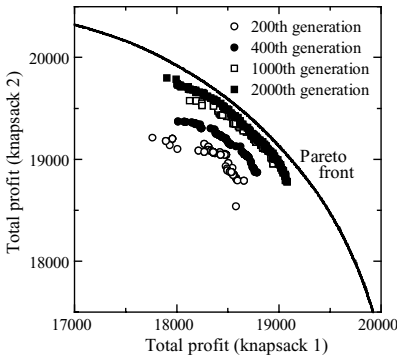
Fig. 5. Average values of the $D1_R$ measure.

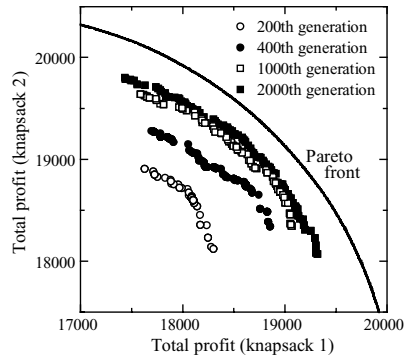(a) 2-500 test problem.   (b) 3-500 test problem.
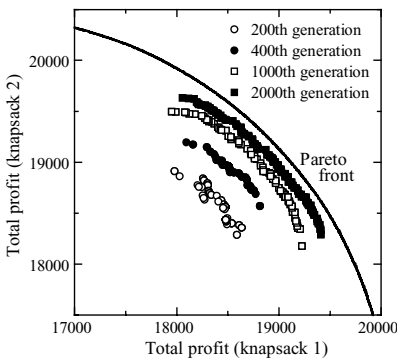
Fig. 6. Average values of the spread measure.
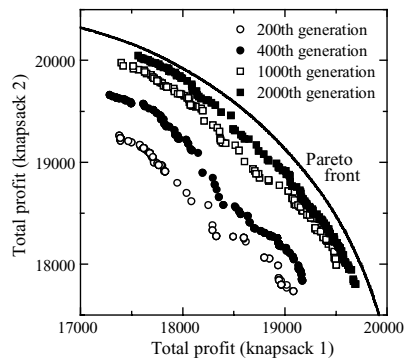

(a) Original NSGA-II: $\alpha = 1$ and $\beta = 1$.   (b) $\alpha = 1$ and $\beta = 9$ (dissimilar parents).

(c) $\alpha = 1$ and $\beta = 9$ (similar parents).   (d) $\alpha = 9$ (dissimilar) and $\beta = 9$ (similar).

Fig. 7. Solution set at each generation in a single run with each combination of α and β.

4.3 Experimental Results with Variable Parameters

We also performed the same computational experiments as in the previous subsection using variable parameters. More specifically, we changed the values of α and β during the execution of the NSGA-II algorithm with our mating scheme. In this subsection, we only report some illustrative cases.

In one case, the value of α was changed from $\alpha = 1$ to $\alpha = 9$ (dissimilar) after the 1000th generation while the value of β was fixed as $\beta = 9$ (similar). This change of α is for driving the population to the Pareto front in the first 1000 generations and for increasing the diversity in the last 1000 generations. Using this change of α, we intended to implement the first-convergence-then-diversity strategy in Fig. 3 (a). Experimental results are shown in Fig. 8 (a). From this figure, we can see that our intention was successfully implemented. That is, the diversity was increased in the last 1000 generations while the population with small diversity was driven to the Pareto front in the first 1000 generations. From the comparison between Fig. 7 (c) and Fig. 8 (a), we can observe the effect of changing the value of α. It should be noted that the first 1000 generations in Fig. 8 (a) are the same as Fig. 7 (c).

In another case, the value of α was changed from $\alpha = 9$ (dissimilar) to 1 after the 1000th generation while the value of β was fixed as $\beta = 9$ (similar). This change of α is for increasing the diversity in the first 1000 generations and for driving the population to the Pareto front in the last 1000 generations. Using this change of α, we intended to implement the first-diversity-then-convergence strategy in Fig. 3 (b). Experimental results are shown in Fig. 8 (b). From the comparison between Fig. 7 (d) and Fig. 8 (b), we can see that the convergence around the center of the Pareto front was improved by the change of the value of α from $\alpha = 9$ (dissimilar) to 1. That is, our intention to implement the first-diversity-then-convergence strategy was successfully achieved. It should be noted that the first 1000 generations in Fig. 8 (b) are the same as Fig. 7 (d).
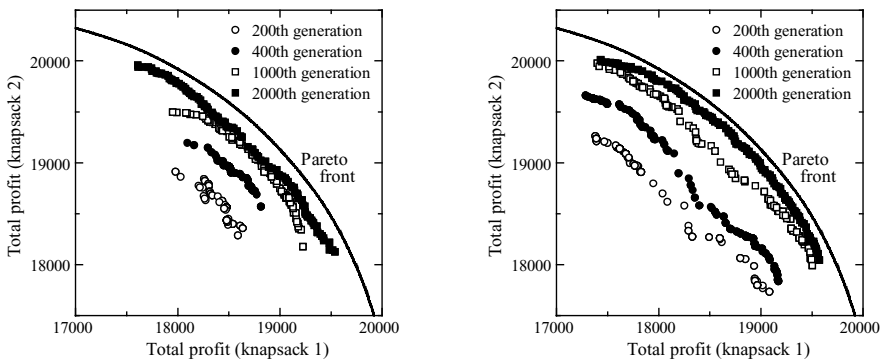
(a) α : 1 to 9 (dissimilar), $\beta = 9$ (similar).   (b) α : 9 (dissimilar) to 1, $\beta = 9$ (similar).

Fig. 8. Experimental results using variable parameters.
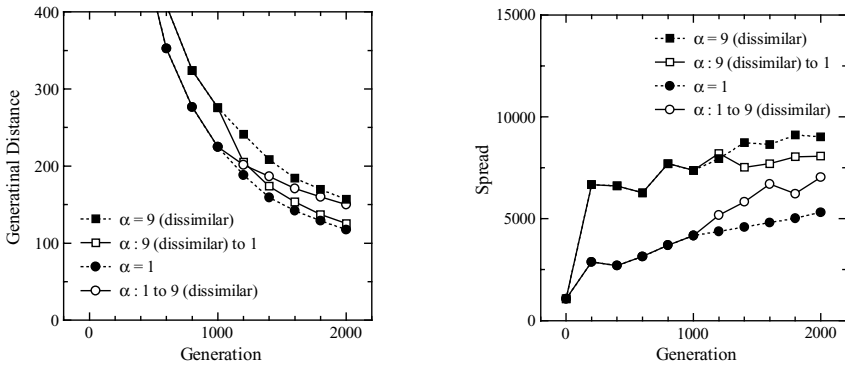
(a) Average values of the GD measure.   (b) Average values of the spread measure.

Fig. 9. Average results over ten runs for each specification of α and β ($\beta = 9$ (similar)).

We further examined the above-mentioned four cases of α and β: those in Fig. 7 (c), (d) and Fig. 8 (a), (b). Using each specification, we applied the NSGA-II with our mating scheme to the 2-500 test problem ten times. Then we calculated the average values of the GD and spread measures over ten runs. Experimental results are shown in Fig. 9. From the results depicted by white squares in Fig. 9 (a), we can see that the change of α from $\alpha = 9$ (dissimilar) to $\alpha = 1$ improved the convergence, which corresponds to the improvement from Fig. 7 (d) to Fig. 8 (b). This improvement was achieved at the cost of the decrease in the diversity as shown in Fig. 9 (b). We can also see from the results depicted by closed circles in Fig. 9 (b) that the change of α from $\alpha = 1$ to $\alpha = 9$ (dissimilar) improved the diversity.

5 Concluding Remarks

We proposed an idea of dynamically controlling the balance between the diversity of solutions and the convergence to the Pareto front using a similarity-based mating scheme. Through computational experiments on multiobjective knapsack problems, we demonstrated that our mating scheme has the potential ability to dynamically control the diversity-convergence balance. We also visually demonstrated that the performance of the NSGA-II algorithm was improved in terms of the diversity of solutions. Our mating scheme has two parameters α and β, which are the number of candidates from which two parents are selected. Since the interpretation of these parameters is very easy, we can intuitively control the values of these parameters.

As we described using several figures, different specifications of the diversity-convergence balance may be required in different stages of generation for each multiobjective optimization problem. Thus an automated adaptation method of α and β may be required in our mating scheme. This is left for future research.

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. Fonseca, C. M., and Fleming, P. J.: Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization, *Proc. of 5th International Conference on Genetic Algorithms* (1993) 416-423.
5. Fonseca, C. M., and Fleming, P. J.: An Overview of Evolutionary Algorithms in Multiobjective Optimization, *Evolutionary Computation* 3 (1995) 1-16.
6. Goldberg, D. E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading (1989).
7. Hajela, P., and Lin, C. Y.: Genetic Search Strategies in Multicriterion Optimal Design, *Structural Optimization* 4 (1992) 99-107.
8. Horn, J., Nafpliotis, N., and Goldberg, D. E.: A Niched Pareto Genetic Algorithm for Multi-Objective Optimization, *Proc. of 1st IEEE International Conference on Evolutionary Computation* (1994) 82-87.
9. Huang, C. F.: Using an Immune System Model to Explore Mate Selection in Genetic Algorithms, *Lecture Notes in Computer Sciences* 2723 (2003) 1041-1052. *Proc. of 2003 Genetic and Evolutionary Computation Conference.*
10. 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.
11. Ishibuchi, H., and Shibata, Y.: An Empirical Study on the Effect of Mating Restriction on the Search Ability of EMO Algorithms, *Lecture Notes in Computer Science* 2632 (2003) 433-447. *Proc. of Second International Conference on Evolutionary Multi-Criterion Optimization.*
12. Ishibuchi, H., and Shibata, Y.: A Similarity-Based Mating Scheme for Evolutionary Multiobjective Optimization, *Lecture Notes in Computer Sciences* 2723 (2003) 1065-1076. *Proc. of 2003 Genetic and Evolutionary Computation Conference.*
13. 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.
14. Jaszkiewicz, A.: Genetic Local Search for Multi-Objective Combinatorial Optimization, *European Journal of Operational Research* 137 (2002) 50-71.
15. Knowles, J. D., and Corne, D. W.: On Metrics for Comparing Non-Dominated Sets, *Proc. of 2002 Congress on Evolutionary Computation* (2002) 711-716.
16. Knowles, J. D., and Corne, D. W.: Towards Landscape Analyses to Inform the Design of a Hybrid Local Search for Multiobjective Quadratic Assignment Problem, In A. Abraham, J. Ruiz-del-Solar, and M. Koppen (eds.), *Soft Computing Systems*, IOS Press, Amsterdam (2002) 271-279.

17. Knowles, J. D., and Corne, D. W.: Instance Generators and Test Suites for the Multiobjective Quadratic Assignment Problem, *Lecture Notes in Computer Science* 1993 (2003) 295-310. *Proc. of Second International Conference on Evolutionary Multi-Criterion Optimization.*

18. 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.

19. Schaffer, J. D.: Multiple Objective Optimization with Vector Evaluated Genetic Algorithms, *Proc. of 1st International Conference on Genetic Algorithms and Their Applications* (1985) 93-100.

20. Van Veldhuizen, D. A., and Lamont, G. B.: Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-Art, *Evolutionary Computation* 8 (2000) 125-147.

21. Zitzler, E., and Thiele, L.: Multiobjective Optimization using Evolutionary Algorithms – A Comparative Case Study, *Proc. of 5th International Conference on Parallel Problem Solving from Nature* (1998) 292-301.

22. 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.

23. 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.