A Similarity-Based Mating Scheme for Evolutionary 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. This paper proposes a new mating scheme for evolutionary multiobjective optimization (EMO), which simultaneously improves the convergence speed to the Pareto-front and the diversity of solutions. The proposed mating scheme is a two-stage selection mechanism. In the first stage, standard fitness-based selection is iterated for selecting a pre-specified number of candidate solutions from the current population. In the second stage, similarity-based tournament selection is used for choosing a pair of parents among the candidate solutions selected in the first stage. For maintaining the diversity of solutions, selection probabilities of parents are biased toward extreme solutions that are different from prototypical (i.e., average) solutions. At the same time, our mating scheme uses a mechanism where similar parents are more likely to be chosen for improving the convergence speed to the Pareto-front. Through computational experiments on multi-objective knapsack problems, it is shown that the performance of recently proposed well-known EMO algorithms (SPEA and NSGA-II) can be improved by our mating scheme.

1 Introduction

Evolutionary multi-objective optimization (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 [3]). In some studies, local search was combined with EMO algorithms for further improving the convergence speed to the Pareto-front [10, 12–14]. 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 [6, 17, 21]. The aim of this paper is to clearly demonstrate that the search ability of EMO algorithms can be improved by appropriately choosing parent solutions. For this aim, we propose a new mating scheme that is applicable to any EMO algorithms. For maintaining the diversity of solutions, the selection probabilities of parent solutions are biased toward extreme solutions that are different from prototypical (i.e., average) solutions in our mating scheme. At the same time,

our mating scheme uses a mechanism where similar parents are more likely to be chosen for improving the convergence speed to the Pareto-front.

Mating restriction was suggested by Goldberg [7] and used in EMO algorithms by Hajela & Lin [8] and Fonseca & Fleming [5]. The basic idea of mating restriction is to ban the crossover of dissimilar parents from which good offspring are not likely to be generated. In the implementation of mating restriction, a user-definable parameter σ_{mating} called the mating radius is usually used for banning the crossover of two parents whose distance is larger than σ_{mating}. The distance between two parents is measured in the decision space or the objective space. The necessity of mating restriction in EMO algorithms was also stressed by Jaszkiewicz [13] and Watanabe et al. [18]. On the other hand, Zitzler & Thiele [20] 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. [9] argued that information from very different types of tradeoffs could be combined to yield other kinds of good tradeoffs. Schaffer [16] examined the selection of dissimilar parents but observed no improvement.

In our previous study [11], we demonstrated positive and negative effects of mating restriction on the search ability of EMO algorithms through computational experiments on knapsack problems and flowshop scheduling problems. The positive effect of the recombination of similar parents is the improvement in the convergence speed to the Pareto-front while its negative effect is the decrease in the diversity of solutions. On the other hand, the positive effect of the recombination of dissimilar parents is the improvement in the diversity while its negative effect is the deterioration in the convergence speed. In this paper, we propose a new mating scheme for simultaneously improving the convergence speed and the diversity. The effect of the proposed mating scheme on the performance of the SPEA [21] and the NSGA-II [4] is examined through computational experiments on knapsack problems in Zitzler & Thiele [21]. Experimental results show that the search ability of those EMO algorithms on the two-objective and three-objective knapsack problems is significantly improved by the proposed mating scheme.

2   Proposed Mating Scheme

We describe our mating scheme using the following k-objective optimization problem:

$$\text{Optimize } \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_k(\mathbf{x})), \tag{1}$$
$$\text{subject to } \mathbf{x} \in \mathbf{X}, \tag{2}$$

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. Let us denote the distance between two solutions $\mathbf{x}$ and $\mathbf{y}$ as $|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y})|$ in the objective space. In this paper, the distance is measured by the Euclidean distance as

$$|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y})| = \sqrt{|f_1(\mathbf{x}) - f_1(\mathbf{y})|^2 + \cdots + |f_k(\mathbf{x}) - f_k(\mathbf{y})|^2} \ . \tag{3}$$

We propose a two-stage mating scheme illustrated in Fig. 1. The selection in the second stage (i.e., upper layer) is based on the similarity between solutions while the selection in the first stage (i.e., lower layer) uses the fitness value of each solution. Our mating scheme is applicable to any EMO algorithms because an arbitrary fitness definition can be directly used with no modification in its lower layer. For choosing the first parent (i.e., Parent A in Fig. 1), the standard fitness-based binary tournament selection with replacement is iterated α times for choosing α candidates (say $\mathbf{x}_1$, $\mathbf{x}_2$, ..., $\mathbf{x}_\alpha$). Next the center vector over the chosen α candidates is calculated in the objective space as

$$\bar{\mathbf{f}}(\mathbf{x}) = (\bar{f}_1(\mathbf{x}), \bar{f}_2(\mathbf{x}), ..., \bar{f}_k(\mathbf{x})) , \tag{4}$$

where

$$\bar{f}_i(\mathbf{x}) = \frac{1}{\alpha} \sum_{j=1}^{\alpha} f_i(\mathbf{x}_j) \ \text{ for } \ i = 1, 2, ..., k \ . \tag{5}$$

Then the most dissimilar solution to the center vector $\bar{\mathbf{f}}(\mathbf{x})$ is chosen as Parent A in Fig. 1 among the α candidates. That is, the most extreme solution with the largest distance from the center vector $\bar{\mathbf{f}}(\mathbf{x})$ in the objective space is chosen as the first parent A in Fig. 1. When multiple solutions have the same largest distance, one solution is randomly chosen among them (i.e., random tiebreak). The choice of the first parent is illustrated for the case of $\alpha = 3$ in Fig. 2 (a) where three solutions $\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{x}_3$ are selected as candidates of the first parent. The most dissimilar solution $\mathbf{x}_3$ to the center vector $\bar{\mathbf{f}}(\mathbf{x})$ is chosen as the first parent in Fig. 2 (a).
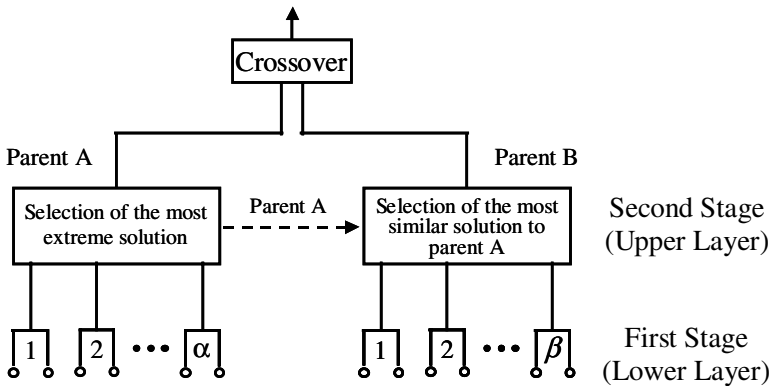

Fig. 1. The proposed mating scheme.

(a) Choice of the first parent ($\alpha = 3$)   (b) Choice of the second parent ($\beta = 5$)
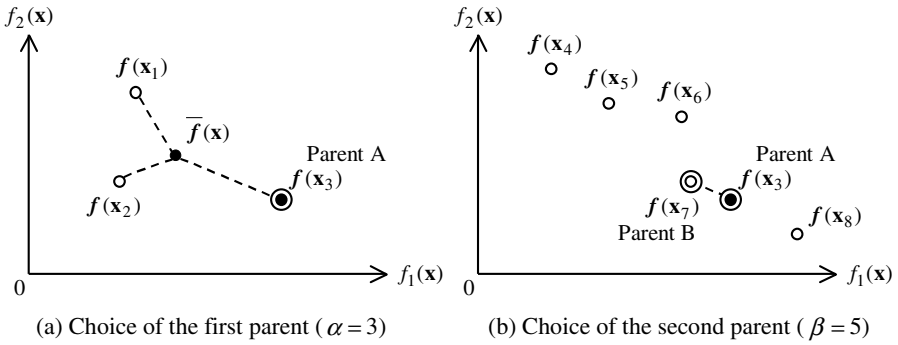
Fig. 2. Illustration of the proposed mating scheme.

When $\alpha = 1$, the choice of the first parent is the same as the standard binary tournament selection. The case of $\alpha = 2$ is actually the same as the standard binary tournament selection because two candidates always have the same distance from their center vector. Selection probabilities are biased toward extreme 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 in Fig. 1). Then the most similar solution to the first parent (i.e., Parent A in Fig. 1) is chosen as Parent B among the β candidates. That is, the solution with the smallest distance from Parent A is chosen. In this manner, similar parents are recombined in our mating scheme. The choice of the second parent is illustrated in Fig. 2 (b) for the case of $\beta = 5$. The most similar solution $\mathbf{x}_7$ to the first parent (i.e., $\mathbf{x}_3$) is selected as the second parent among the five candidates ($\mathbf{x}_4$, ..., $\mathbf{x}_8$) in Fig. 2 (b). A crossover operation is applied to $\mathbf{x}_3$ and $\mathbf{x}_7$ for generating new solutions.

The mating scheme in our former study [11] corresponds to the case of $\alpha = 1$. That is, the first parent was chosen by the standard fitness-based binary tournament selection. Not only the choice of the most similar solution as the second parent but also the choice of the most dissimilar solution was examined. Experimental results in our former study [11] suggested that the choice of similar parents improved the convergence speed to the Pareto-front while it had a negative effect on the diversity of solutions. On the other hand, the choice of dissimilar parents improved the diversity of solutions while it had a negative effect on the convergence speed to the Pareto-front. The main motivation to propose the new mating scheme in Fig. 1 is to simultaneously improve the diversity and the convergence speed by appropriately choosing parents for recombination.

Our mating scheme has high applicability and high flexibility. The positive aspects of our mating scheme are summarized as follows:

(1) Our mating scheme is applicable to any EMO algorithms because an arbitrary fitness definition can be directly used with no modification.
(2) The specification of the mating radius σ_{mating} is not necessary.

(3) The selection pressure toward extreme solutions is adjustable by the specification of the parameter α.

(4) The selection pressure toward similar solutions is adjustable by the specification of the parameter β.

(5) The proposed mating scheme has high flexibility. For example, not only the binary tournament selection but also other selection mechanisms can be used for choosing candidate solutions. The distance between solutions can be measured in the decision space as well as in the objective space. The choice of dissimilar parents can be also examined using our mating scheme.

On the other hand, the negative aspects of the proposed mating scheme are as follows:

(i) Appropriate specifications of the two parameters α and β seem to be problem-dependent. The sensitivity of the performance of EMO algorithms on the parameter specifications will be examined in the next section.

(ii) Additional computational load is required for performing our mating scheme in EMO algorithms. The increase in CPU time will be also examined in the next section. In general, the computational overhead caused by our mating scheme is negligible when the evaluation of each solution needs long CPU time.

3   Computational Experiments

In this section, we examine the effect of our mating scheme on the performance of EMO algorithms through computational experiments. For this purpose, we combined our mating scheme with recently developed well-known EMO algorithms: SPEA [21] and NSGA-II [4]. It should be noted that our mating scheme is the same as the standard binary tournament selection when the two parameters α and β are specified as $\alpha = 1$ and $\beta = 1$. In this case, the modified SPEA and NSGA-II algorithms with our mating scheme are the same as their original versions. Using 100 combinations of α and β (i.e., $\alpha = 1,2,...,10$ and $\beta = 1,2,...,10$), we examine the effect of our mating scheme on the performance of the EMO algorithms.

3.1   Test Problems and Parameter Specifications

In our computational experiments, we used four knapsack problems in Zitzler & Thiele [21]: two-objective 250-item, two-objective 500-item, three-objective 250-item, and three-objective 500-item test problems. Each solution in an m-item knapsack problem was coded as a binary string of the length m. Thus the search space size was 2^m. Each string was evaluated in the same manner as in Zitzler & Thiele [21].

The modified SPEA and NSGA-II algorithms with our mating scheme were applied to the four knapsack problems under the following parameter specifications:

Crossover probability: 0.8,
Mutation probability: $1/m$ where m is the string length,
Population size in NSGA-II: 200,
Population size in SPEA: 100,
Population size of the secondary population in SPEA: 100,
Stopping condition: 2000 generations.

3.2 Performance Measures

Various performance measures have been proposed in the literature for evaluating a set of non-dominated solutions. As explained in Knowles & Corne [15], no single performance measure can simultaneously evaluate various aspects of a solution set. Moreover, some performance measures are not designed for simultaneously comparing many solution sets but for comparing two solution sets with each other.

For comparing various combinations of α and β, we use the average distance from each Pareto-optimal solution to its nearest solution in a solution set. This performance measure was used in Czyzak & Jaszkiewicz [2] and referred to as $D1_R$ in Knowles & Corne [15]. The $D1_R$ measure needs all Pareto-optimal solutions of each test problem. For the two-objective 250-item and 500-item knapsack problems, the Pareto-optimal solutions are available from the homepage of the first author of [21]. For the three-objective 250-item and 500-item knapsack problems, we found near Pareto-optimal solutions using the SPEA and the NSGA-II. These algorithms were applied to each test problem 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 (see Subsection 3.1). We also used a single-objective genetic algorithm with a secondary population where all the non-dominated solutions were stored with no size limitation. Each of the three objectives was used in the single-objective genetic algorithm. This algorithm was applied to each three-objective test problem 30 times (10 times for each objective using the same stopping condition as the NSGA-II: 30000 generations with the population size 400). The SPEA and the NSGA-II were also applied to each test problem 10 times. Thus we obtained 50 solution sets for each test problem. Then we chose non-dominated solutions from the obtained 50 solution sets as near Pareto-optimal solutions. The number of the Pareto-optimal or near Pareto-optimal solutions of each test problem in our computational experiments is as follows:

567 solutions (2/250 test problem),
1427 solutions (2/500 test problem),
2158 solutions (3/250 test problem),
2142 solutions (3/500 test problem),

where the k/m test problem means the k-objective m-item test problem.

3.3 Experimental Results

The modified SPEA and NSGA-II algorithms with our mating scheme were applied to the four test problems using 100 combinations of α and β. For each combination, we performed ten runs from different initial populations for each test problem.

Average values of the $D1_R$ measure over ten runs are summarized in Figs. 3–6 where smaller values of the $D1_R$ measure (i.e., shorter bars) mean better results. In each figure, the result by the original EMO algorithm corresponds to the bar at the top-right corner where $\alpha = 1$ and $\beta = 1$. From these figures, we can see that our mating scheme improved the performance of the SPEA and the NSGA-II over a wide range of combinations of α and β. Especially the performance of the original EMO algorithms on the three-objective test problems (i.e., Fig. 5 and Fig. 6) was improved by our mating scheme in almost all combinations of α and β. This is also the case for the performance of the NSGA-II on the 2/500 test problem (i.e., Fig. 4 (b)). The significant deterioration in the performance was observed only when the value of α in the modified SPEA was too large in Fig. 3 (a) and Fig. 4 (a).
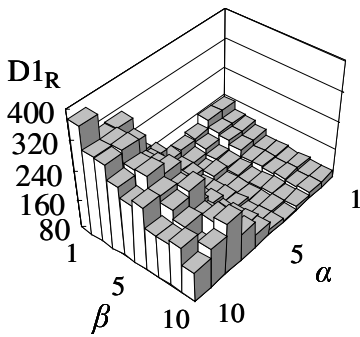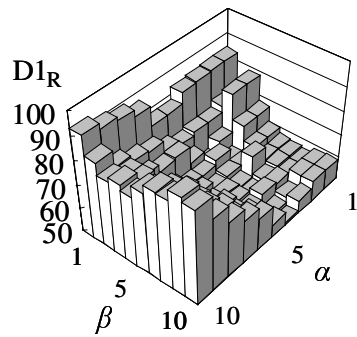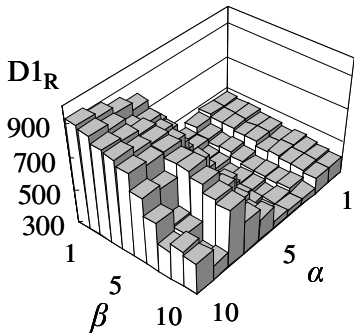

(a) Results by the modified SPEA   (b) Results by the modified NSGA-II

Fig. 3. Average values of the $D1_R$ measure for the two-objective 250-item problem.

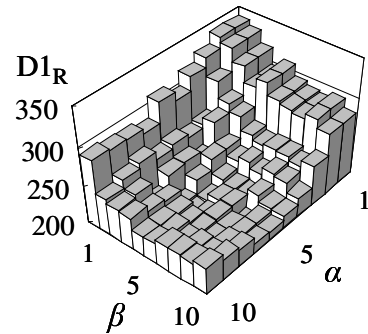(a) Results by the modified SPEA   (b) Results by the modified NSGA-II

Fig. 4. Average values of the $D1_R$ measure for the two-objective 500-item problem.
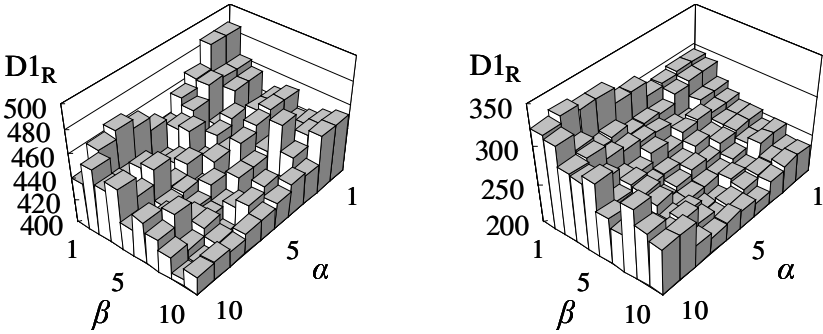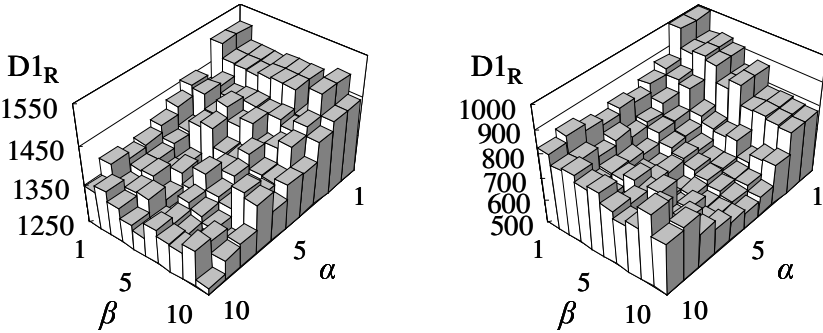
(a) Results by the modified SPEA   (b) Results by the modified NSGA-II

Fig. 5. Average values of the $D1_R$ measure for the three-objective 250-item problem.

(a) Results by the modified SPEA   (b) Results by the modified NSGA-II

Fig. 6. Average values of the $D1_R$ measure for the three-objective 500-item problem.

Using the Mann-Whitney U test, we examined the statistical significance of the improvement in the $D1_R$ measure by the proposed mating scheme. More specifically, the results by the original EMO algorithms (i.e., $\alpha = 1$ and $\beta = 1$) were compared with those by their modified versions (i.e., $\alpha \geq 2$ and/or $\beta \geq 2$) for examining the statistical significance of the improvement by the proposed mating scheme for three confidence levels (90%, 95% and 99%). Confidence levels of the improvement are summarized in Table 1 for the 2/250 test problem and Table 2 for the 3/500 test problem. From those tables, we can see that the performance of the SPEA and the NSGA-II was significantly improved by our mating scheme in many cases. As shown in our experimental results in Figs. 3-6 and Tables 1-2, the selection bias toward either extreme solutions (i.e., $\alpha \geq 3$ and $\beta = 1$) or similar parents (i.e., $\alpha = 1$ and $\beta \geq 2$) improved the performance of the EMO algorithms. It is, however, clearly shown by some experimental results (e.g., Fig. 4 (b), Fig. 5 (a) and Fig. 6 (a)) that the simultaneous use of them (i.e., $\alpha \geq 3$ and $\beta \geq 2$) improved their performance more significantly. For example, the best result was obtained in Fig. 6 (a) from the combination of $\alpha = 10$ and $\beta = 10$.

Table 1. Confidence levels of the improvement for the two-objective 250-item test problem. (* means that the corresponding confidence level is less than 90%)

(a) Results for the SPEA

α	The value of β									
	1	2	3	4	5	6	7	8	9	10
1	*	95	99	99	99	99	99	99	99	99
2	*	95	99	99	99	99	99	99	99	99
3	99	99	99	99	99	99	99	99	99	99
4	99	99	99	99	99	99	99	99	99	99
5	99	99	99	99	99	99	99	99	99	99
6	*	99	99	99	99	99	99	99	99	99
7	*	*	*	99	*	*	95	90	95	*
8	*	*	*	*	*	*	*	*	*	*
9	*	*	*	*	*	*	*	*	*	*
10	*	*	*	*	*	*	*	*	*	*

(b) Results for the NSGA-II

α	The value of β									
	1	2	3	4	5	6	7	8	9	10
1	*	99	90	99	99	99	99	99	99	99
2	*	99	90	99	99	99	99	99	99	99
3	*	99	99	99	99	99	99	99	99	99
4	*	99	99	99	99	99	99	99	99	99
5	95	99	99	99	99	99	99	99	99	99
6	90	99	99	99	99	99	99	99	99	99
7	*	99	99	99	99	99	99	99	99	99
8	*	95	99	99	99	99	99	99	99	99
9	*	95	95	95	99	99	99	95	90	*
10	*	*	90	*	90	*	*	*	*	*

**Table 2.** Confidence levels of the improvement for the three-objective 500-item test problem.

(a) Results for the SPEA

α	The value of β									
	1	2	3	4	5	6	7	8	9	10
1	*	*	*	90	*	*	*	99	*	95
2	*	*	*	90	*	*	*	99	*	95
3	99	95	99	95	95	99	99	99	95	99
4	90	95	99	99	99	99	99	99	99	99
5	99	99	99	99	99	99	99	99	99	99
6	99	99	99	99	99	99	99	99	99	99
7	99	99	99	99	99	99	99	99	99	99
8	99	99	99	99	99	99	99	99	99	99
9	99	99	99	99	99	99	99	99	99	99
10	99	99	99	99	99	99	99	99	99	99

(b) Results for the NSGA-II

α	The value of β									
	1	2	3	4	5	6	7	8	9	10
1	*	95	90	95	99	99	99	99	99	99
2	*	95	90	95	99	99	99	99	99	99
3	99	99	99	99	99	99	99	99	99	99
4	99	99	99	99	99	99	99	99	99	99
5	99	99	99	99	99	99	99	99	99	99
6	99	99	99	99	99	99	99	99	99	99
7	99	99	99	99	99	99	99	99	99	99
8	99	99	99	99	99	99	99	99	99	99
9	99	99	99	99	99	99	99	99	99	99
10	99	99	99	99	99	99	99	99	95	99

As mentioned in Section 2, additional computation load is required for executing our mating scheme. For evaluating such a computational overhead, we measured the average CPU time for each combination of α and β. Experimental results are summarized in Fig. 7 for the SPEA and Fig. 8 for the NSGA-II. Since the CPU time of the original and modified SPEA algorithms totally depends on the computation load for the clustering of non-dominated solutions in the secondary population, it is not easy to evaluate the pure effect of our mating scheme (see Fig. 7). On the other hand, the evaluation of the computational overhead caused by our mating scheme is easy for the NSGA-II as shown in Fig. 8 where we observe the linear increase in the average CPU time with the increase in the values of α and β. The increase in the average CPU time from the original NSGA-II with $\alpha = 1$ and $\beta = 1$ to the modified NSGA-II with $\alpha = 10$ and $\beta = 10$ was 15.7% in Fig. 8 (a) and 7.4% in Fig. 8 (b).
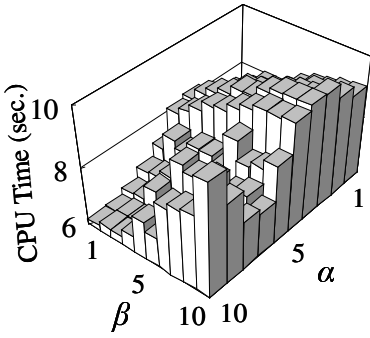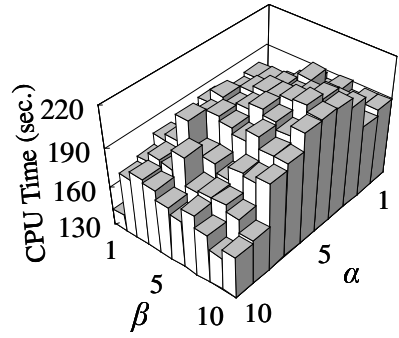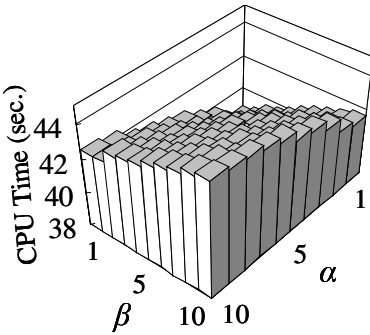
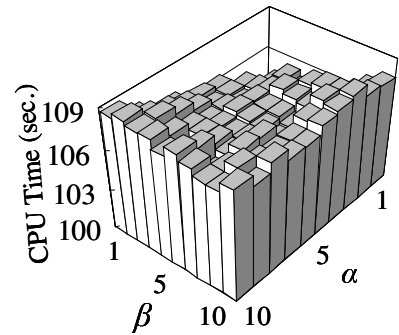(a) Two-objective 250-item test problem   (b) Three-objective 500-item test problem

Fig. 7. Average CPU time of the original and modified SPEA.
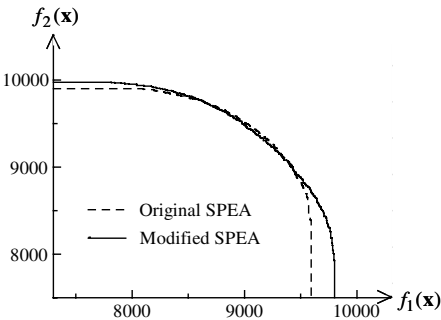

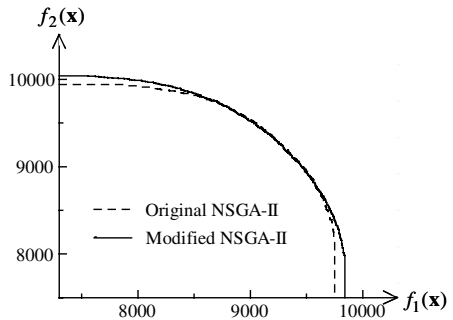(a) Two-objective 250-item test problem   (b) Three-objective 500-item test problem

Fig. 8. Average CPU time of the original and modified NSGA-II.

(a) SPEA and its modified version   (b) NSGA-II and its modified version

Fig. 9. 50% attainment surface for the two-objective 250-item test problem.

For visually demonstrating the improvement in the performance of the EMO algorithms by our mating scheme, we show the 50% attainment surface (e.g., see [3]) obtained by the original EMO algorithms and the modified EMO algorithms in Fig. 9

for the 2/250 problem. The best values of α and β with the smallest D1$_R$ measures in Fig. 3 were used in Fig. 9 for the modified SPEA and NSGA-II algorithms. We can see from Fig. 9 that better attainment surfaces were obtained by the modified algorithms. Similar improvement was also observed for the 2/500 problem.

4 Concluding Remarks

We proposed a two-stage mating scheme for simultaneously improving the diversity of solutions and the convergence speed to the Pareto-front. The basic idea is to bias selection probabilities toward extreme solutions for preserving the diversity and toward similar parents for improving the convergence speed. The effect of our mating scheme was examined through computational experiments on multiobjective knapsack problems where our mating scheme was combined with the two well-known EMO algorithms (i.e., SPEA and NSGA-II). It was shown that the performance of those EMO algorithms was improved by our mating scheme. It was also shown that the increase in the CPU time caused by our mating scheme was small if compared with the total CPU time (e.g., 7.4% increase). The simultaneous improvement in the diversity and the convergence speed is usually very difficult. This is also the case in our mating scheme. In our mating scheme, the two parameters (i.e., α and β) should be carefully adjusted to strike a balance between the diversity and the convergence speed. Further studies are needed for automatically specifying these parameter values appropriately. Further studies are also needed for examining the effectiveness of our mating scheme for recently developed other EMO algorithms such as SPEA2 [22].

Our mating scheme can be viewed as assigning a selection probability to each pair of solutions (not to each individual solution). Pairs of similar solutions tend to have higher selection probabilities than those of dissimilar solutions. At the same time, pairs of extreme solutions tend to have higher selection probabilities than those of prototypical solutions. While various sophisticated methods for assigning a fitness value to each individual solution have been proposed in the literature on EMO algorithms, the assignment of a fitness value (or a selection probability) to each pair of solutions has not been studied well. Experimental results in this paper clearly show that such an idea of fitness assignment has a possibility to improve EMO algorithms with sophisticated fitness assignment schemes to each individual solution.

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. Czyzak, P., and Jaszkiewicz, A.: Pareto-Simulated Annealing – A Metaheuristic Technique for Multi-Objective Combinatorial Optimization, *Journal of Multi-Criteria Decision Analysis* 7 (1998) 34–47.

3. Deb, K.: *Multi-Objective Optimization Using Evolutionary Algorithms*, John Wiley & Sons, Chichester (2001).
4. 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.
5. 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.
6. Fonseca, C. M., and Fleming, P. J.: An Overview of Evolutionary Algorithms in Multiobjective Optimization, *Evolutionary Computation* 3 (1995) 1–16.
7. Goldberg, D. E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading (1989).
8. Hajela, P., and Lin, C. Y.: Genetic Search Strategies in Multicriterion Optimal Design, *Structural Optimization* 4 (1992) 99–107.
9. 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.
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, *Proc. of Second International Conference on Evolutionary Multi-Criterion Optimization* (2003 in press).
12. 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* (2003 in press).
13. Jaszkiewicz, A.: Genetic Local Search for Multi-Objective Combinatorial Optimization, *European Journal of Operational Research* 137 (2002) 50–71.
14. Knowles, J. D., and Corne, D. W.: M-PAES: A Memetic Algorithm for Multiobjective Optimization, *Proc. of 2000 Congress on Evolutionary Computation* (2000) 325–332.
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. 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.
17. Van Veldhuizen, D. A., and Lamont, G. B.: Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-Art, *Evolutionary Computation* 8 (2000) 125–147.
18. Watanabe, S., Hiroyasu, T., and Miki, M.: LCGA: Local Cultivation Genetic Algorithm for Multi-Objective Optimization Problem, *Proc. of 2002 Genetic and Evolutionary Computation Conference* (2002) 702.
19. Zitzler, E., Deb, K., and Thiele, L.: Comparison of Multiobjective Evolutionary Algorithms: Empirical Results, *Evolutionary Computation* 8 (2000) 173–195.
20. 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.
21. Zitzler, E., and Thiele, L.: Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach, *IEEE Transactions on Evolutionary Computation* 3 (1999) 257–271.
22. Zitzler, E., Laumanns, M., and Thiele, L.: SPEA2: Improving the Performance of the Strength Pareto Evolutionary Algorithm, *Technical Report 103*, Computer Engineering and Communication Networks Lab, Swiss Federal Institute of Technology, Zurich (2001).