

Effects of the Use of Non-Geometric Binary Crossover on Evolutionary Multiobjective Optimization

Hisao Ishibuchi, Yusuke Nojima, Noritaka Tsukamoto, and Ken Ohara
Graduate School of Engineering, Osaka Prefecture University
1-1 Gakuen-cho, Naka-ku, Sakai, Osaka 599-8531, Japan
Phone: +81-72-254-9350

{hisaoi, nojima}@cs.osakafu-u.ac.jp, {nori, ohara}@ci.cs.osakafu-u.ac.jp

ABSTRACT

In the design of evolutionary multiobjective optimization (EMO) algorithms, it is important to strike a balance between diversity and convergence. Traditional mask-based crossover operators for binary strings (e.g., one-point and uniform) tend to decrease the diversity of solutions in EMO algorithms while they improve the convergence to the Pareto front. This is because such a crossover operator, which is called geometric crossover, always generates an offspring in the segment between its two parents under the Hamming distance in the genotype space. That is, the sum of the distances from the generated offspring to its two parents is always equal to the distance between the parents. In this paper, first we propose a non-geometric binary crossover operator to generate an offspring outside the segment between its parents. Next we examine the effect of the use of non-geometric binary crossover on single-objective genetic algorithms. Experimental results show that non-geometric binary crossover improves their search ability. Then we examine its effect on EMO algorithms. Experimental results show that non-geometric binary crossover drastically increases the diversity of solutions while it slightly degrades their convergence to the Pareto front. As a result, some performance measures such as hypervolume are clearly improved.

Categories and Subject Descriptors

I.2.8 [Artificial Intelligence]: Problem Solving, Control Methods, and Search – *Heuristic Methods*.

General Terms

Algorithms.

Keywords

Non-geometric crossover operators, evolutionary multiobjective optimization (EMO) algorithms, multiobjective combinatorial optimization, diversity preserving, balance between diversity and convergence.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GECCO'07, July 7-11, 2007, London, England, United Kingdom.
Copyright 2007 ACM 978-1-59593-697-4/07/0007...\$5.00.

1. INTRODUCTION

Evolutionary multiobjective optimization (EMO) algorithms have been successfully applied to various application areas [3]. EMO algorithms are designed to find a set of well-distributed Pareto-optimal or near Pareto-optimal solutions with a wide range of objective values, which approximates the entire Pareto front of a multiobjective optimization problem. It is important in the design of EMO algorithms to strike a balance between diversity and convergence [1], [13]. Usually no *a priori* information about the decision maker's preference is used when EMO algorithms search for Pareto-optimal solutions. A set of non-dominated solutions is presented to the decision maker as a result of the search by an EMO algorithm. The decision maker is supposed to choose a final solution from the presented solutions based on his/her preference. The EMO approach, which consists of the search for a number of non-dominated solutions and the choice of a final solution from the obtained non-dominated solutions, is referred to as an ideal multiobjective optimization procedure in Deb [3]. It is implicitly assumed in the EMO approach that the choice of a final solution from the obtained non-dominated solutions is much easier for the decision maker than the elicitation of his/her preference before the search for non-dominated solutions.

It is essential for the success of the EMO approach to find a set of non-dominated solutions that well approximates the entire Pareto front. It is, however, not easy (usually very difficult) for EMO algorithms to find such a good non-dominated solution set of a large-scale combinatorial multiobjective optimization problem as pointed out in some studies (e.g., Jaszkievicz [14]). This is the case even when multiobjective optimization problems have only two objectives. It was visually demonstrated in Ishibuchi et al. [10] that crossover had a negative effect on the diversity of solutions (while it improved their convergence to the Pareto front) when EMO algorithms were applied to large-scale two-objective 0/1 knapsack problems. Several ideas of recombining similar parents were proposed to improve the performance of EMO algorithms by decreasing such a negative effect of crossover and increasing its positive effect [12], [15], [21], [23].

Recently the concept of geometric crossover was proposed by Moraglio and Poli [16]-[18] to analyze crossover operators in terms of the distances between an offspring and its parents. Roughly speaking, a crossover operator is referred to as being geometric crossover when the following relation always holds between an offspring C and its two parents P1 and P2:

$$Distance(C, P1) + Distance(C, P2) = Distance(P1, P2), \quad (1)$$

where $Distance(A, B)$ denotes the distance between A and B in the genotype space (for details, see [16]-[18]). Traditional mask-based crossover operators for binary strings (e.g., one-point, two-point and uniform) are geometric crossover [16], [17] because the relation in (1) always holds for the Hamming distance. That is, such a crossover operator always generates an offspring in the segment between its two parents under the Hamming distance in the genotype space. On the other hand, many crossover operators for real number strings such as simulated binary crossover (SBX) [4], [5], blend crossover (BLX- α [7]), extended line crossover [19], unimodal normal distribution crossover (UNDX [20]), and linear crossover [24] are non-geometric crossover [18]. That is, such a crossover operator can generate an offspring C satisfying the following relation for its two parents $P1$ and $P2$:

$$Distance(C, P1) + Distance(C, P2) > Distance(P1, P2), \quad (2)$$

where the Euclidean distance is used to measure the distance between real number strings.

In almost all genetic algorithms for real number strings, non-geometric crossover operators have been used to maintain the diversity of solutions. On the other hand, geometric crossover operators have been used in almost all genetic algorithms for binary strings.

In this paper, we propose the use of non-geometric crossover in EMO algorithms for binary strings to increase the diversity of solutions. First we propose a non-geometric crossover operator for binary strings in Section 2. Next we examine its effect on the performance of single-objective genetic algorithms (SOGA) in Section 3. Then we examine its effect on the performance of EMO algorithms in Section 4 using NSGA-II [6]. Finally we conclude this paper in Section 5.

2. NON-GEOMETRIC CROSSOVER

Let \mathbf{x} and \mathbf{y} be two real number vectors (they can also be viewed as real number strings). A simple line crossover operator for generating an offspring vector \mathbf{z} from the two parent vectors \mathbf{x} and \mathbf{y} can be written as follows:

$$\mathbf{z} = \alpha \mathbf{x} + (1 - \alpha) \mathbf{y}, \quad (3)$$

where α is a randomly specified real number. When α is always in the unit interval $[0, 1]$, this is geometric crossover. On the other hand, this is non-geometric crossover when α may assume a real number outside the unit interval $[0, 1]$. In the latter case, the crossover operator in (3) is referred to as extended line crossover [19], by which an offspring C satisfying the inequality relation in (2) can be generated.

Non-geometric crossover operators have been used in almost all genetic algorithms for real number strings as in [4], [5], [7], [19], [20], [24]. On the contrary, traditional mask-based binary crossover operators (e.g., one-point, two-point, and uniform) are geometric crossover. We illustrate uniform crossover in Fig. 1 where an offspring C is generated from two parents $P1$ and $P2$. In Fig. 1, the Hamming distance between the two parents (i.e., 10) is equal to the sum of the Hamming distances from the offspring C to its two parents $P1$ and $P2$. The same uniform crossover operator is illustrated in Fig. 2 in the Hamming distance space.

The horizontal axis of Fig. 2 is the Hamming distance from Parent 1 ($P1$) to the offspring C while its vertical axis is the Hamming distance from Parent 2 ($P2$). The offspring C in Fig. 1 is located at the point (5, 5) in Fig. 2. Each open circle in Fig. 2 shows a possible location of an offspring that can be generated from $P1$ and $P2$ in Fig. 1 by geometric crossover. As shown in Fig. 2, an offspring is always generated in the segment between its parents $P1$ and $P2$ in the Hamming distance space. The three arrows from C in Fig. 2 show possible moves by a bit-flip mutation for a single bit of the offspring C in Fig. 1.

	* * * * *	* * * * *
Parent 1 (P1)	0000000000	1111111111
Offspring (C)	0000000000	0110010101
Parent 2 (P2)	0000000000	0000000000

Figure 1. Illustration of uniform crossover.

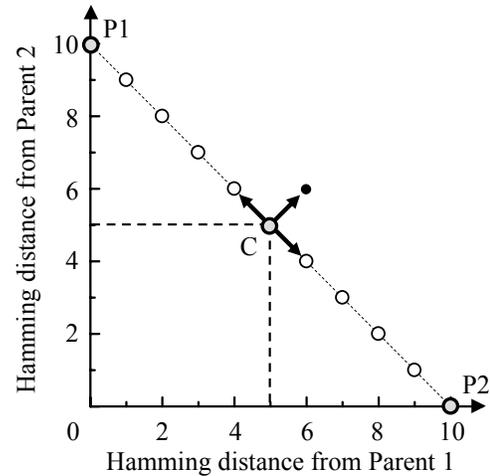


Figure 2. Relation between an offspring C and its two parents $P1$ and $P2$ in the Hamming distance space. The three arrows from C show possible moves by a bit-flip mutation for a single bit of the offspring C in Fig. 1.

As shown in Fig. 2, traditional mask-based binary crossover operators always generate an offspring in the segment between its two parents in the Hamming distance space. This corresponds to the situation where α is always in the unit interval $[0, 1]$ in line crossover in (3). In order to generate an offspring outside the segment between its two parents, we propose a non-geometric binary crossover operator which corresponds to the situation where α is outside the unit interval $[0, 1]$ in line crossover in (3).

Let \mathbf{x} and \mathbf{y} be two binary strings of length n . We denote them as $\mathbf{x} = x_1 x_2 \dots x_n$ and $\mathbf{y} = y_1 y_2 \dots y_n$. The proposed non-geometric binary crossover operator generates an offspring $\mathbf{z} = z_1 z_2 \dots z_n$ satisfying the following relation:

$$Distance(\mathbf{x}, \mathbf{z}) + Distance(\mathbf{y}, \mathbf{z}) > Distance(\mathbf{x}, \mathbf{y}), \quad (4)$$

where the distance between two binary strings is measured by the Hamming distance. The basic idea is to generate an offspring from one parent in the opposite side of the other parent.

First one parent is chosen as a primary parent (say \mathbf{x}). The choice of a primary parent is discussed later. The other parent is used as a secondary parent (say \mathbf{y}). Then an offspring \mathbf{z} is generated from the primary parent \mathbf{x} and the secondary parent \mathbf{y} as follows:

(1) When $x_i = y_i$:

$$z_i = x_i \text{ with a probability } (1 - P_M^*),$$

$$z_i = (1 - x_i) \text{ with a probability } P_M^*.$$

(2) When $x_i \neq y_i$:

$$z_i = x_i.$$

In the proposed non-geometric binary crossover operator, the standard bit-flip mutation operator is applied to x_i of the primary parent with a prespecified probability when $x_i = y_i$ (i.e., when the values are the same between the two parents). On the other hand, when $x_i \neq y_i$ (i.e., when the values are different between the two parents), x_i is always inherited to the offspring.

The proposed crossover operator is illustrated in Fig. 3 where Parent 1 is used as a primary parent (i.e., as $\mathbf{x} = x_1 x_2 \dots x_n$). Since the values are the same between the two parents in the first 10 loci in Fig. 3, the standard bit-flip mutation operator is applied to each of the first 10 values of Parent 1 with a prespecified probability (0.3 in Fig. 3). On the other hand, the last 10 values of Parent 1 are inherited to the offspring with no modification since the values in the last 10 loci are different between the two parents.

	* * *
Parent 1 (P1)	0000000000 1111111111
Offspring (C)	0010010010 1111111111
Parent 2 (P2)	0000000000 0000000000

Figure 3. Illustration of the proposed non-geometric binary crossover operator. In this figure, Parent 1 and Parent 2 are used as primary and secondary parents, respectively.

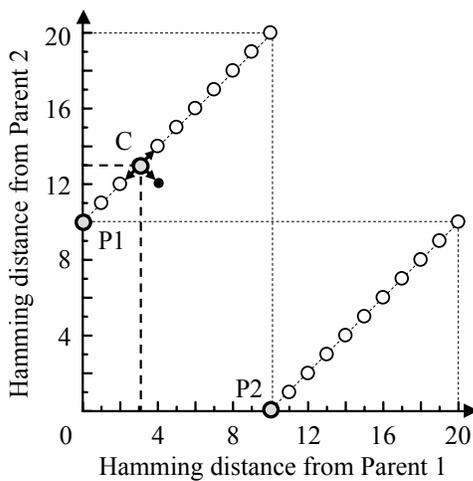


Figure 4. Relation between an offspring C and its two parents P1 and P2 in the case of non-geometric crossover. The three arrows from C show possible moves by a bit-flip mutation.

From Fig. 3, we can see that the Hamming distance between the offspring C and the secondary parent P2 is 13 while the Hamming distance between the two parents is 10. The generated offspring C in Fig. 3 is depicted in the Hamming distance space in Fig. 4. Its location shows the Hamming distances from Parent 1 and Parent 2. When Parent 1 is a primary parent, our crossover operator can generate an offspring at one of the open circles in the left-upper part of Fig. 4. On the other hand, the open circles in the right-lower part of Fig. 4 show possible offspring by our crossover operator when Parent 2 is a primary parent. The three arrows from C in Fig. 4 show possible moves by a bit-flip mutation for a single bit of the offspring C in Fig. 3.

The choice of a primary parent can be performed randomly from two parents. We can also choose the better one from two parents as a primary parent. These two strategies (i.e., random choice and better parent choice) are compared with each other through computational experiments in this paper.

We use the proposed non-geometric binary crossover operator with a prespecified probability P and a standard geometric binary crossover operator with a probability $(1 - P)$. When the crossover probability is P_X , the non-geometric and geometric crossover operators are used with the probabilities $P \cdot P_X$ and $(1 - P) \cdot P_X$, respectively.

3. EFFECTS ON SOGA ALGORITHMS

First we examine the effect of the proposed non-geometric binary crossover operator on the performance of single-objective genetic algorithms (SOGAs). Its effect on EMO algorithms is examined in the next section. In our computational experiments, we used as a test problem a single-objective 500-item 0/1 knapsack problem with two constraint conditions. This test problem was generated from the two-objective 500-item 0/1 knapsack problem of Zitzler and Thiele [27] by using the sum of the two objectives as a scalarizing fitness function. That is, the following fitness function was maximized:

$$fitness(\mathbf{x}) = f_1(\mathbf{x}) + f_2(\mathbf{x}), \quad (5)$$

where \mathbf{x} is a 500-dimensional binary vector, $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ are the two objectives of the original 2-500 test problem in [27]. Each solution is represented by a binary string of length 500. Thus the size of the search space is 2^{500} .

As an SOGA, we used a single-objective version of NSGA-II (i.e., SOGA with the $(\mu + \lambda)$ -ES generation update mechanism where $\mu = \lambda$). In our SOGA (and also in NSGA-II in the next section), we used the maximum ratio scheme [27] to repair infeasible solutions into feasible ones. This repair scheme was implemented in the Lamarckian manner [9]. Our SOGA was applied to the test problem 50 times using the following parameter specifications:

Population size: 200 (i.e., $\mu = \lambda = 200$),

Crossover probability P_X : 0.8,

Mutation probability P_M : 0.002 (i.e., $1/500$),

Termination condition: 2000 generations.

As a traditional geometric binary crossover operator, we used the uniform crossover operator. The proposed non-geometric binary and uniform crossover operators were used with the probabilities $P \cdot P_X$ and $(1 - P) \cdot P_X$, respectively. We examined six values of

P : $P = 0.0, 0.1, 0.2, 0.3, 0.4, 0.5$. It should be noted that only the uniform crossover operator was used in the case of $P = 0.0$. On the other hand, we examined eight values of P_M^* : $P_M^* = 0.002, 0.004, 0.008, 0.012, 0.016, 0.020, 0.040, 0.080$. As a result, we used 48 (i.e., 6×8) combinations of these parameter values in our computational experiments.

Using the optimal solution \mathbf{x}^* of our test problem, we calculated the relative error of the solution \mathbf{x} as

$$\text{Relative error}(\mathbf{x}) = \frac{\text{fitness}(\mathbf{x}^*) - \text{fitness}(\mathbf{x})}{\text{fitness}(\mathbf{x}^*)} \times 100 \text{ (\%)} \quad (6)$$

The average relative error was calculated over 50 runs of our SOGA after the 2000th generation for each combination of the above-mentioned parameter values.

Experimental results are summarized in Fig. 5 where we used the better parent choice to choose a primary parent in our non-geometric binary crossover operator. The base plane of Fig. 5 shows the combination of the parameter values while the height of each bar is the average relative error. The performance of SOGA with only uniform crossover (i.e., $P = 0.0$ in Fig. 5) was improved by the use of the proposed non-geometric binary crossover operator around the right-bottom corner in Fig. 5. For example, the average relative error was improved from 0.424 (with $P = 0.0$) to 0.363 (with $P = 0.5$ and $P_M^* = 0.008$). This improvement is statistically significant (the p -value was 0.002 in Student's t -test).

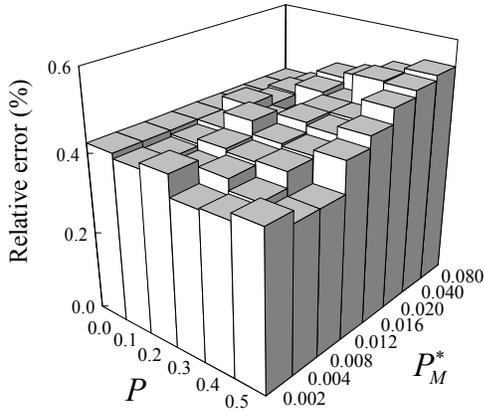


Figure 5. Average relative errors by SOGA with the proposed non-geometric binary crossover operator where the better parent choice was used to choose a primary parent.

For comparison, we examined the random choice of a primary parent. Experimental results are shown in Fig. 6. We can see that similar results were obtained in Fig. 5 and Fig. 6. That is, the choice of a primary parent did not have a large effect on the performance of SOGA with our non-geometric crossover operator.

In order to examine the effect of our non-geometric crossover operator on the diversity of solutions, we calculated the entropy of each population. Let p_{0k} and p_{1k} be the ratios of 0 and 1 at the k -th locus over all binary strings of length n in a population,

respectively (thus $p_{0k} + p_{1k} = 1$). Then the entropy of this population n is calculated as follows (n : string length):

$$H = - \sum_{k=1}^n (p_{0k} \log p_{0k} + p_{1k} \log p_{1k}). \quad (7)$$

Average entropy values over 50 runs with 2000 generations in Fig. 5 are summarized in Fig. 7. From Fig. 7, we can see that our non-geometric crossover operator increased the diversity of solutions in many cases. The diversity was decreased by our non-geometric crossover operator in only a few cases with small values of P_M^* . In those cases, the consistent use of the better parent as a primary parent has a negative effect on the diversity of solutions. Actually, this negative effect can be removed by randomly choosing one of two parents as a primary parent (see Fig. 8 where average entropy values are summarized for computational experiments in Fig. 6). In Fig. 7 and Fig. 8, we can also see that P_M^* has a positive effect on the diversity of solutions.

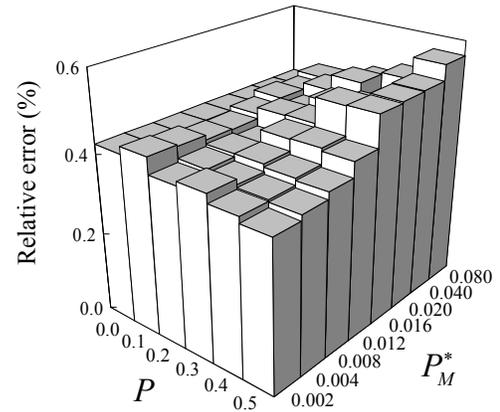


Figure 6. Average relative errors by SOGA with the proposed non-geometric crossover operator where the random parent choice was used to choose a primary parent.

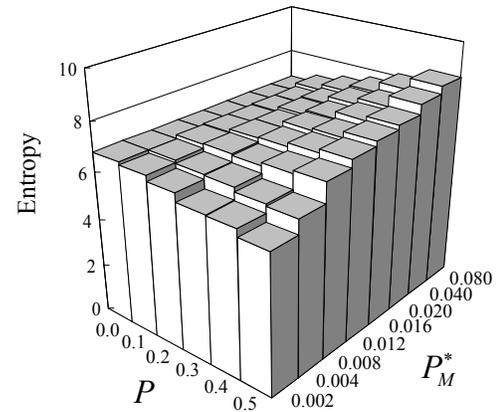


Figure 7. Average values of the entropy over 50 runs in Fig. 5 where the better parent choice was used to choose a primary parent.

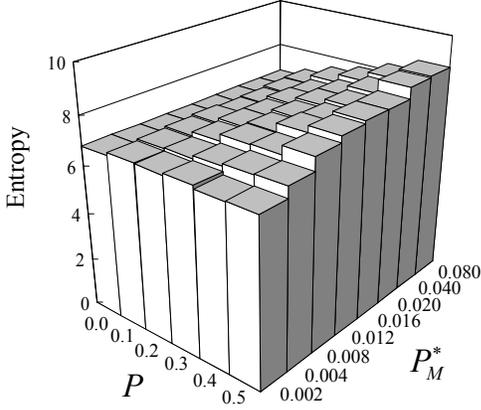


Figure 8. Average values of the entropy over 50 runs in Fig. 6 where the random parent choice was used to choose a primary parent.

4. EFFECTS ON EMO ALGORITHMS

We have already demonstrated that the proposed non-geometric binary crossover operator improved the performance of SOGA and increased the diversity of solutions. Whereas the performance improvement was statistically significant, it was not so impressive. In this section, we demonstrate that our non-geometric crossover operator has a large effect on the performance of EMO algorithms through computational experiments on the two-objective 500-item 0/1 knapsack problem [27] using NSGA-II [6].

Using the same parameter specifications as in the previous section, we applied NSGA-II to the 2-500 knapsack problem. The effect of our non-geometric crossover operator was evaluated using four performance measures (i.e., generational distance, $D1_R$ measure, hypervolume measure, and range measure).

Let S and S^* be a non-dominated solution set and the Pareto-optimal solution set, respectively. The convergence of the non-dominated solution set S to the Pareto-optimal solution set S^* has been often measured by the following performance index called the generational distance [22]:

$$GD(S) = \frac{1}{|S|} \sum_{\mathbf{x} \in S} \min \{ \|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y})\| : \mathbf{y} \in S^* \}, \quad (8)$$

where $\|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y})\|$ is the Euclidean distance between the two solutions \mathbf{x} and \mathbf{y} in the objective space, and $|S|$ is the number of solutions in S (i.e., $|S|$ is the cardinality of S). The generational distance is the average distance from each solution in S to its nearest Pareto-optimal solution in S^* .

Whereas the generational distance measures the proximity of the non-dominated solution set S to the Pareto-optimal solution set S^* , it can not measure the diversity of solutions. In order to measure not only the convergence but also the diversity, we can use the following performance index called the $D1_R$ measure:

$$D1_R(S) = \frac{1}{|S^*|} \sum_{\mathbf{y} \in S^*} \min \{ \|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y})\| : \mathbf{x} \in S \}. \quad (9)$$

The $D1_R$ measure is the average distance from each Pareto-optimal solution \mathbf{y} in S^* to its nearest solution in S . This measure was used in [2], [13].

The diversity of the non-dominated solution set S can be more directly measured by the sum of the range of objective values for each objective function:

$$\text{Range}(S) = \sum_{i=1}^K [\max_{\mathbf{x} \in S} \{f_i(\mathbf{x})\} - \min_{\mathbf{x} \in S} \{f_i(\mathbf{x})\}], \quad (10)$$

where $f_i(\mathbf{x})$ is the i -th objective and K is the total number of objectives. This measure is similar to the maximum spread of Zitzler [25].

In order to measure both the diversity and the convergence, we can also use the hypervolume measure [26] that calculates the volume of the dominated region by the non-dominated solution set S in the objective space. The boundary of the dominated region in the objective space is called the attainment surface [8]. From multiple attainment surfaces obtained by multiple runs of an EMO algorithm for a multiobjective optimization problem, we can calculate the 50% attainment surface as a kind of their average result. For the calculation of the 50% attainment surface, see [3], [8].

As in the previous section, we examined two strategies for the choice of a primary parent: Better parent choice and random parent choice. In the better parent choice, the better one between two parents was always chosen as a primary parent. Two parents were compared using Pareto ranking and a crowding measure in the same manner as the fitness evaluation in NSGA-II. In the random parent choice, one of two parents is randomly chosen.

Average values of the generational distance are summarized in Fig. 9 with the better parent choice and Fig. 10 with the random parent choice. In these figures, the generational distance was monotonically degraded by the increase in the probability P of our non-geometric crossover operator. This may suggest that the convergence of solutions to the Pareto front was degraded by the use of our non-geometric crossover operator.

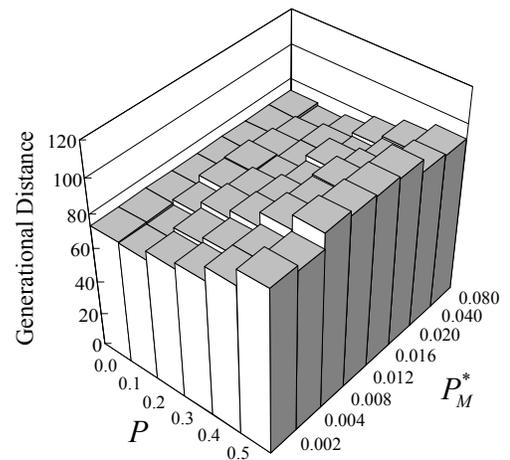


Figure 9. Average values of the generational distance by NSGA-II. The better parent choice was used to choose a primary parent in our non-geometric crossover operator.

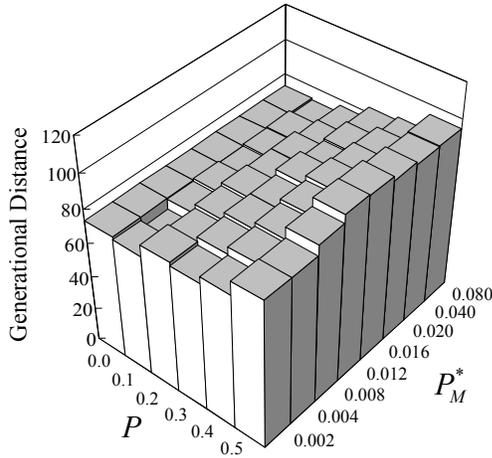


Figure 10. Average values of the generational distance by NSGA-II. The random parent choice was used to choose a primary parent in our non-geometric crossover operator.

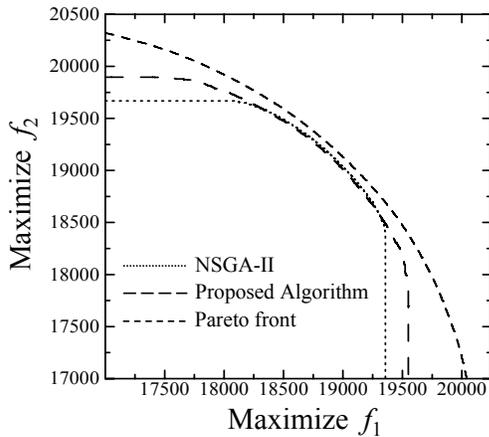


Figure 11. 50% attainment surfaces obtained by the original NSGA-II and the proposed algorithm (i.e., NSGA-II with our non-geometric crossover operator).

The above-mentioned observations are somewhat misleading. Whereas the generational distance was increased, the convergence of solutions to the Pareto front was not severely degraded. This is visually demonstrated by depicting the 50% attainment surface by each algorithm in Fig. 11 where our non-geometric crossover operator was implemented using the better parent choice, $P = 0.5$ and $P_M^* = 0.008$ (i.e., the best combination in the previous section: Good results are also obtained from this combination in this section as shown in Figs. 14-18 later). From Fig. 11, we can see that the convergence of solutions to the Pareto front was not severely degraded by our non-geometric crossover operator.

The effect of our non-geometric crossover operator on NSGA-II can be more clearly demonstrated by depicting all individuals in some generations in a single run of the original NSGA-II and the NSGA-II with our non-geometric crossover operator as in Fig. 12 and Fig. 13. In Fig. 13, the same parameter specifications were

used in our non-geometric crossover operator as in Fig. 11. From the comparison between Fig. 12 and Fig. 13, we can observe a large positive effect of our non-geometric crossover operator on the diversity of solutions in NSGA-II.

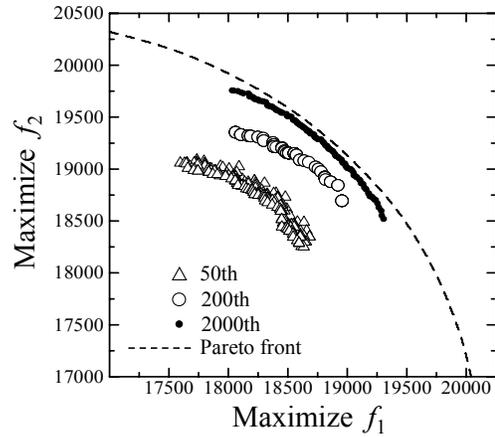


Figure 12. All individuals in some generations in a single run of NSGA-II.

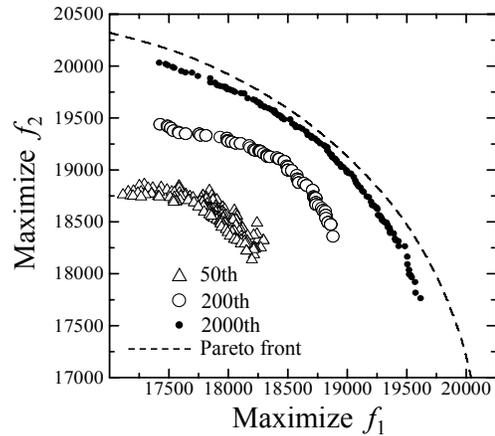


Figure 13. All individuals in some generations in a single run of NSGA-II with our non-geometric crossover operator.

Average values of the $D1_R$ measure are summarized in Fig. 14 for the better parent choice and Fig. 15 for the random parent choice. From these figures, we can see that the performance of NSGA-II was clearly improved by our non-geometric crossover operator (especially around the left-bottom corner in Fig. 14). We can also see that better results were obtained by the better parent choice in Fig. 14 than the random parent choice in Fig. 15.

We also observed performance improvement of NSGA-II by our non-geometric crossover operator for the hypervolume and range measures as shown in Fig. 16 and Fig. 17 (due to the page limitation, we only report experimental results with the better parent choice). Average values of the entropy were shown in Fig. 18. From the comparison between Fig. 18 and the other figures, we can see that the increase in the entropy is strongly related to the performance improvement of NSGA-II.

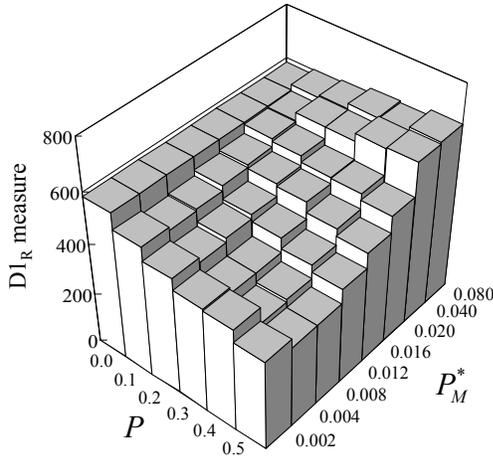


Figure 14. $D1_R$ measure with the better parent choice.

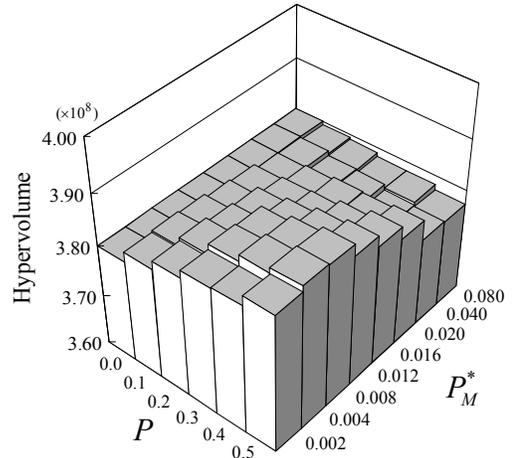


Figure 16. Hypervolume with the better parent choice.

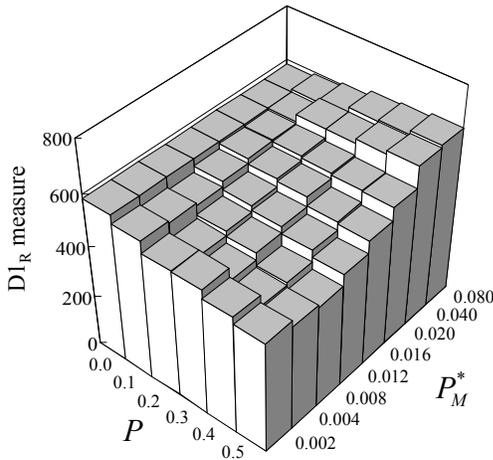


Figure 15. $D1_R$ measure with the random parent choice.

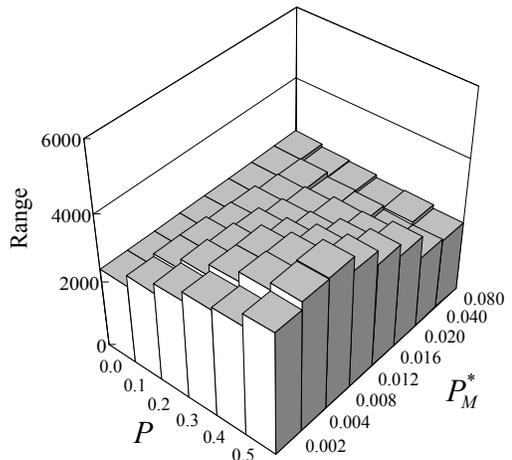


Figure 17. Range with the better parent choice.

5. CONCLUDING REMARKS

In this paper, we proposed the use of a non-geometric binary crossover operator to increase the diversity of solutions in EMO algorithms. The use of non-geometric crossover for binary strings was motivated by the fact that non-geometric crossover has been almost always used for real number strings whereas traditional mask-based binary crossover is geometric crossover. The effect of our non-geometric crossover operator on the performance of EMO algorithms was clearly demonstrated through computational experiments on a two-objective 500-item 0/1 knapsack problem using NSGA-II. Experimental results showed that the diversity of solutions was drastically improved without severely degrading their convergence to the Pareto front. It was shown that our non-geometric crossover operator improved the performance of not only EMO algorithms but also single-objective genetic algorithms.

Diversity improvement can be also easily realized by increasing the mutation probability in NSGA-II. The use of a large mutation probability, however, severely degrades the convergence of solutions to the Pareto front in NSGA-II [11].

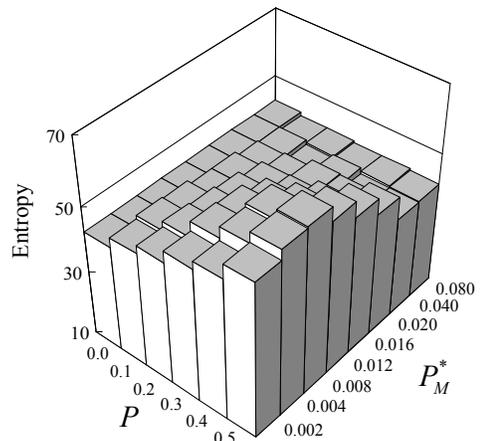


Figure 18. Entropy with the better parent choice.

6. REFERENCES

- [1] Bosman, P. A. N., and Thierens, D. The balance between proximity and diversity in multiobjective evolutionary algorithms. *IEEE Trans. on Evolutionary Computation* 7, 2 (2003) 174-188.
- [2] Czyzak, P., and Jaskiewicz, A. Pareto-simulated annealing - A metaheuristic technique for multi-objective combinatorial optimization. *Journal of Multi-Criteria Decision Analysis* 7, 1 (1998) 34-47.
- [3] Deb, K. *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, Chichester (2001).
- [4] Deb, K., and Agrawal, R. B. Simulated binary crossover for continuous search space. *Complex Systems*, 9, 2 (1995) 115-148.
- [5] Deb, D., and Kumar, A. Real-coded genetic algorithms with simulated binary crossover: Studies on multimodal and multiobjective problems. *Complex Systems*, 9, 6 (1995) 431-454.
- [6] Deb, K., Pratap, A., Agarwal, S., and Meyerivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. on Evolutionary Computation* 6, 2 (2002) 182-197.
- [7] Eshelman, L. J., and Schaffer, J. D. Real-coded genetic algorithms and interval-schemata. *Foundation of Genetic Algorithms 2*, Morgan Kaufmann, San Mateo (1993) 187-202.
- [8] Fonseca, C. M., and Fleming, P. J., 1996. On the performance assessment and comparison of stochastic multiobjective optimizers. *Lecture Notes in Computer Science 1141: Parallel Problem Solving from Nature - PPSN IV*, Springer, Berlin (1996) 584-593.
- [9] Ishibuchi, H., Kaige, S., and Narukawa, K. Comparison between Lamarckian and Baldwinian repair on multiobjective 0/1 knapsack problems. *Lecture Notes in Computer Science 3410: Evolutionary Multi-Criterion Optimization - EMO 2005*, Springer, Berlin (2005) 370-385.
- [10] Ishibuchi, H., and Narukawa, K. Recombination of similar parents in EMO algorithms. *Lecture Notes in Computer Science 3410: Evolutionary Multi-Criterion Optimization - EMO 2005*, Springer, Berlin (2005) 265-279.
- [11] Ishibuchi, H., Narukawa, K., Tsukamoto, N., and Nojima, Y. An empirical study on similarity-based mating for evolutionary multiobjective combinatorial optimization. *European Journal of Operational Research* (in press).
- [12] Ishibuchi, H., and Shibata, Y. Mating scheme for controlling the diversity-convergence balance for multiobjective optimization. *Lecture Notes in Computer Science 3102: Genetic and Evolutionary Computation - GECCO 2004*, Springer, Berlin (2004) 1259-1271.
- [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, 2 (2003) 204-223.
- [14] Jaskiewicz, 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, 4 (2002) 402-412.
- [15] Kim, M., Hiroyasu, T., Miki, M., and Watanabe, S. SPEA2+: Improving the performance of the strength Pareto evolutionary algorithm 2. *Lecture Notes in Computer Science 3242: Parallel Problem Solving from Nature - PPSN VIII*, Springer, Berlin (2004) 742-751.
- [16] Moraglio, A., and Poli, P. Topological interpretation of crossover. *Lecture Notes in Computer Science 3102: Genetic and Evolutionary Computation - GECCO 2004*, Springer, Berlin (2004) 1377-1388.
- [17] Moraglio, A., and Poli, P. Product geometric crossover. *Lecture Notes in Computer Science 4193: Parallel Problem Solving from Nature - PPSN IX*, Springer, Berlin (2006) 1018-1027.
- [18] Moraglio, A., and Poli, P. Inbreeding properties of geometric crossover and non-geometric crossover. *Proc. of Foundation of Genetic Algorithm Workshop - FOGA 2007* (Mexico City, January 8-11, 2007) 15 pages.
- [19] Muhlenbein, H., and Schlierkamp-Voosen, D. Predictive models for the breeder genetic algorithm, I: Continuous parameter optimization. *Evolutionary Computation*, 1, 1 (1993) 25-49.
- [20] Ono, I., and Kobayashi, S. A Real-coded genetic algorithm for function optimization using unimodal normal distribution crossover. *Proc. of 7th International Conference on Genetic Algorithms* (East Lansing, July 19-23, 1997) 246-253.
- [21] Sato, H., Aguirre, H., and Tanaka, K. Effects from local dominance and local recombination in enhanced MOEAs. *Proc. of 5th International Conference on Simulated Evolution and Learning* (Busan, Korea, October 26-29, 2004) CD-ROM Proceedings.
- [22] Van Veldhuizen, D. A. Multiobjective evolutionary algorithms: Classifications, analyses, and new innovations. *Ph. D dissertation*, Air Force Institute of Technology (1999).
- [23] Watanabe, S., Hiroyasu, T., and Miki, M. Neighborhood cultivation genetic algorithm for multi-objective optimization problems. *Proc. of 4th Asia-Pacific Conference on Simulated Evolution and Learning* (Singapore, November 18-22, 2002) 198-202.
- [24] Wright, A. H. Genetic algorithms for real parameter optimization. *Foundation of Genetic Algorithm*, Morgan Kaufmann, San Mateo (1991) 205-218.
- [25] Zitzler, E. Evolutionary algorithms for multiobjective optimization: Methods and applications. *Ph. D dissertation*, Swiss Federal Institute of Technology (1999).
- [26] Zitzler, E., and Thiele, L. Multiobjective optimization using evolutionary algorithms - A comparative case study. *Lecture Notes in Computer Science 1498: Parallel Problem Solving from Nature - PPSN V*, Springer, Berlin (1998) 292-301.
- [27] Zitzler, E., and Thiele, L. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Trans. on Evolutionary Computation* 3, 4 (1999) 257-271.