Effects of Crossover Operations on the Performance of EMO Algorithms

Hisao Ishibuchi

Department of Computer Science and Intelligent Systems, Osaka Prefecture University
1-1 Gakuen-cho, Sakai, Osaka, 599-8531, Japan
hisaoi@cs.osakafu-u.ac.jp

Abstract. This paper visually demonstrates the effect of crossover operations on the performance of EMO algorithms through computational experiments on multi-objective 0/1 knapsack problems. In our computational experiments, we use the NSGA-II algorithm as a representative EMO algorithm. First we compare the performance of the NSGA-II algorithm between two cases: NSGA-II with/without crossover. Experimental results show that the crossover operation has a positive effect on the convergence of solutions to the Pareto front and a negative effect on the diversity of solutions. That is, the crossover operation decreases the diversity of solutions while it improves the convergence of solutions to the Pareto front. Next we examine the effects of recombining similar or dissimilar parents using a similarity-based mating scheme. Experimental results show that the performance of the NSGA-II algorithm is improved by recombining similar parents and degraded by recombining dissimilar ones. Finally we show that the recombination of extreme and similar parents using the similarity-based mating scheme drastically improves the diversity of obtained non-dominated solutions without severely degrading their convergence to the Pareto front. An idea of dynamically controlling the selection pressure toward extreme and similar parents is also illustrated through computational experiments.

1 Introduction

Recently developed EMO algorithms usually share some common ideas such as elitism, fitness sharing and Pareto ranking. While mating restriction has been often discussed in the literature, it has not been used in many EMO algorithms. Even the necessity of crossover operations in EMO algorithms has not been clearly demonstrated in the literature. In this paper, we first examine the necessity of crossover operations in EMO algorithms. Then we examine the effect of mating restriction on the performance of EMO algorithms to find well-distributed Pareto-optimal or near Pareto-optimal solutions.

A similarity-based mating scheme was proposed in Ishibuchi & Shibata [1] to examine positive and negative effects of mating restriction on the performance 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)

Dagstuhl Seminar Proceedings 04461
Practical Approaches to Multi-Objective Optimization
http://drops.dagstuhl.de/opus/volltexte/2005/239

was chosen among a pre-specified number of candidates (say β candidates) based on their similarity or dissimilarity to Parent A. To find β candidates, the standard fitness-based binary tournament selection was iterated β times. Ishibuchi & Shibata [2] 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 vector of those candidates was calculated in the objective space. The most dissimilar candidate to the average vector was chosen as Parent A. On the other hand, the most similar one to Parent A among β candidates was chosen as Parent B. Furthermore, it was demonstrated in [3] that the diversity-convergence balance can be dynamically adjusted by controlling the values of the two parameters α and β in the mating scheme in Fig. 1. Ishibuchi & Narukawa [4] also examined the relation between the similarity of recombined parents and the performance of EMO algorithms.

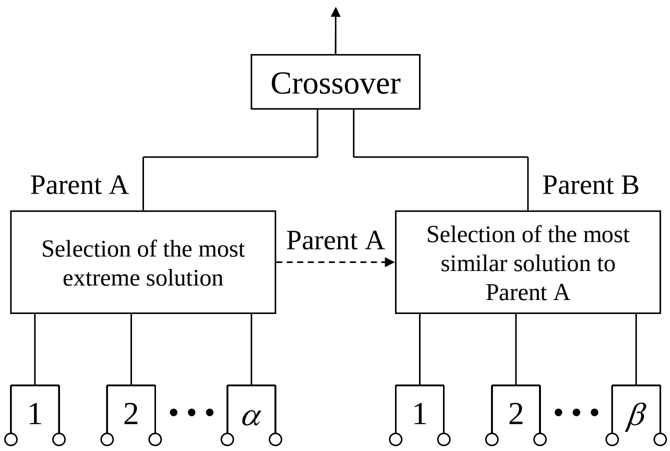


Fig. 1. Mating scheme in Ishibuchi & Shibata [2].

In this paper, we examine the effect of crossover operations on the performance of EMO algorithms through computational experiments on multiobjective 0/1 knapsack problems using the NSGA-II algorithm of Deb et al. [5]. We also examine the effect of the similarity-based mating scheme in Fig. 1 on the performance of the NSGA-II algorithm.

2 Test Problems

A multiobjective 0/1 knapsack problem with k knapsacks (i.e., k objectives and k constraints) and n items in Zitzler & Thiele [6] can be written as follows:

[*k*-*n* knapsack problem]

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

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

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

In this formulation, $\mathbf{x}$ is an n-dimensional binary vector (i.e., $(x_1, x_2, ..., x_n) \in \{0, 1\}^n$), p_{ij} is the profit of item j according to knapsack i, w_{ij} is the weight of item j according to knapsack i, and c_i is the capacity of knapsack i. Each solution $\mathbf{x}$ is handled as a binary string of length n in EMO algorithms. The k-objective n-item knapsack problem is referred to as a k-n knapsack problem in this paper. Zitzler & Thiele [6] examined the performance of several EMO algorithms using nine test problems. In this paper, we use the two-objective 500-item knapsack problem (i.e., 2-500 test problem).

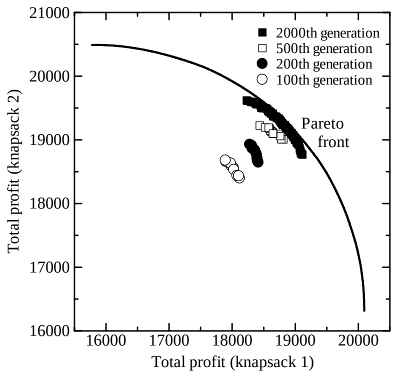
The distance between two solutions in the objective space is calculated by the Euclidean distance. That is, the distance between two solutions $\mathbf{x}$ and $\mathbf{y}$ is calculated in the objective space as

$$|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y})| = \sqrt{|f_1(\mathbf{x}) - f_1(\mathbf{y})|^2 + \cdots + |f_k(\mathbf{x}) - f_k(\mathbf{y})|^2}\ , \tag{4}$$

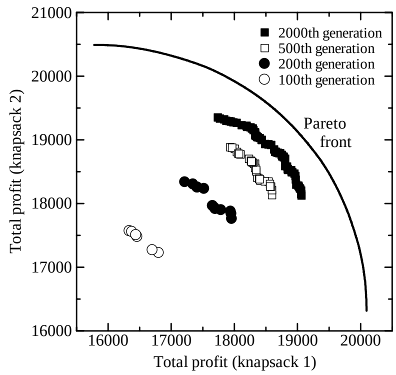
where $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}),\ ...,\ f_k(\mathbf{x}))$ is the k-dimensional objective vector corresponding to the solution $\mathbf{x}$.

3 Computational Experiments

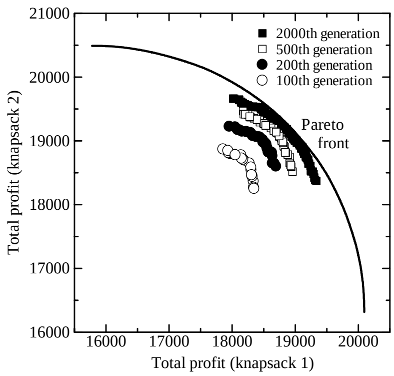
We applied the NSGA-II algorithm with the standard one-point crossover operation to the 2-500 test problem. The population size was specified as 200. In Fig. 2, we show non-dominated solutions at each generation of a single run of the NSGA-II algorithm with crossover (in the left column with the crossover rate 0.8) and without crossover (in the right column). Three different specifications of the mutation rate were examined in Fig. 2. From Fig. 2, we can see that the crossover operation has a positive effect on the convergence of solutions to the Pareto front while it has a negative effect on the diversity of solutions. This is because the crossover operation is likely to generate new offspring in the intermediate region between their parents in the objective space. That is, the crossover operation is likely to shrink the spread of the population. As a result, extreme regions in the objective space are not frequently explored by the NSGA-II algorithm with the crossover operation if compared with the case of no crossover. Experimental results in Fig. 2 suggest that the recombination of similar parents may improve the performance of the NSGA-II algorithm because such recombination is not likely to shrink the spread of the population. Fig. 2 also suggests the necessity of a diversity-increasing mechanism in the NSGA-II algorithm.



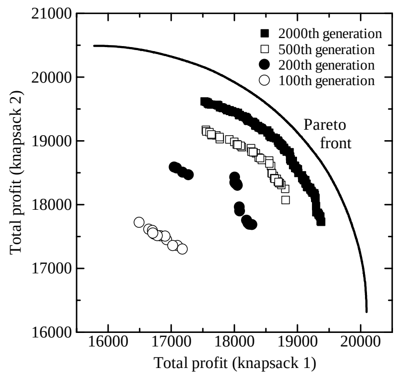
(a) With crossover (mutation rate is 0.001).



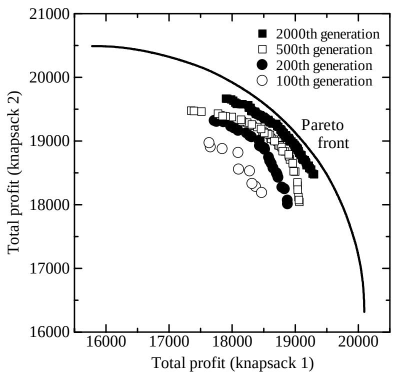
(b) No crossover (mutation rate is 0.001).



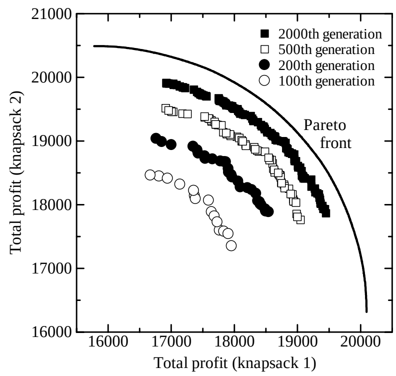
(c) With crossover (mutation rate is 0.002).



(d) No crossover (mutation rate is 0.002).



(e) With crossover (mutation rate is 0.01).

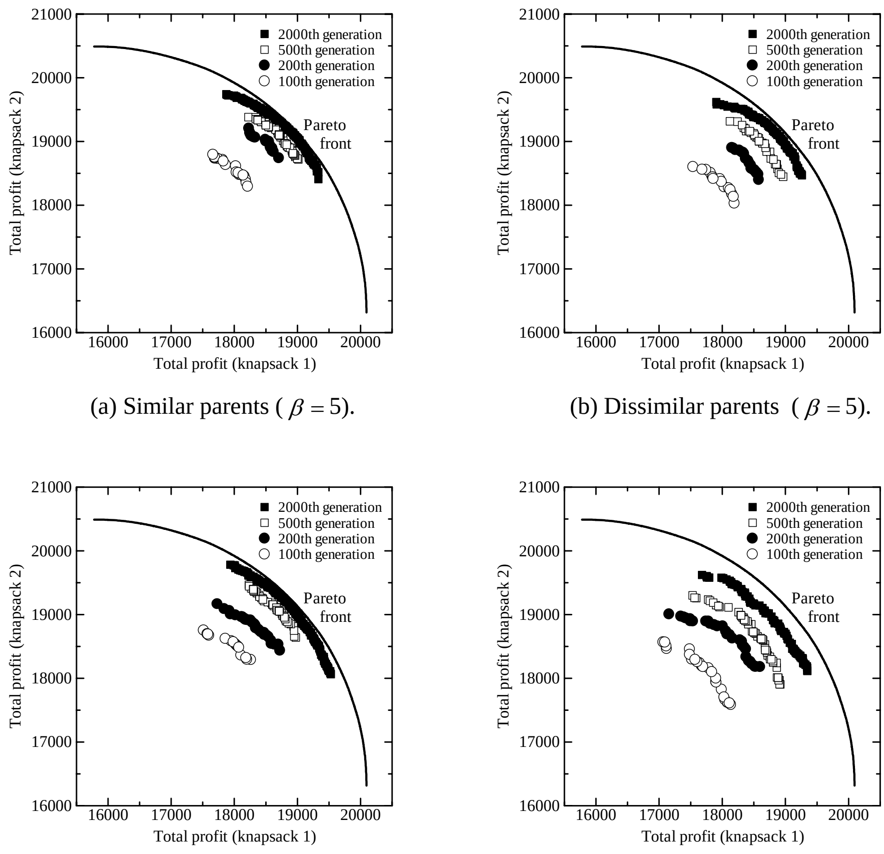


(f) No crossover (mutation rate is 0.01).

Fig. 2. Effects of crossover on the performance of NSGA-II for the 2-500 test problem.

Next we examine the effect of recombining similar parents using the similarity-based mating scheme in Fig. 1. When a pair of parents is to be chosen, one parent (say Parent A) is selected by the standard fitness-based binary tournament selection in the same manner as the NSGA-II algorithm. That is, the value of α in Fig. 1 is fixed as $\alpha = 1$ in order to focus on the effect of recombining similar parents. Next we iterate the standard fitness-based binary tournament selection β times to find β candidates for the selection of the other parent (say Parent B). The most similar candidate to Parent A is chosen as Parent B (i.e., as the mate of Parent A). For comparison, we also examine the choice of the most dissimilar parent to Parent A.

We applied the NSGA-II algorithm with the mating scheme in Fig. 1 to the 2-500 test problem using various values of β while fixing the value of α as $\alpha = 1$. Experimental results are shown in Fig. 3 where the crossover rate and the mutation rate are 0.8 and 0.002, respectively.



(a) Similar parents ($\beta = 5$).

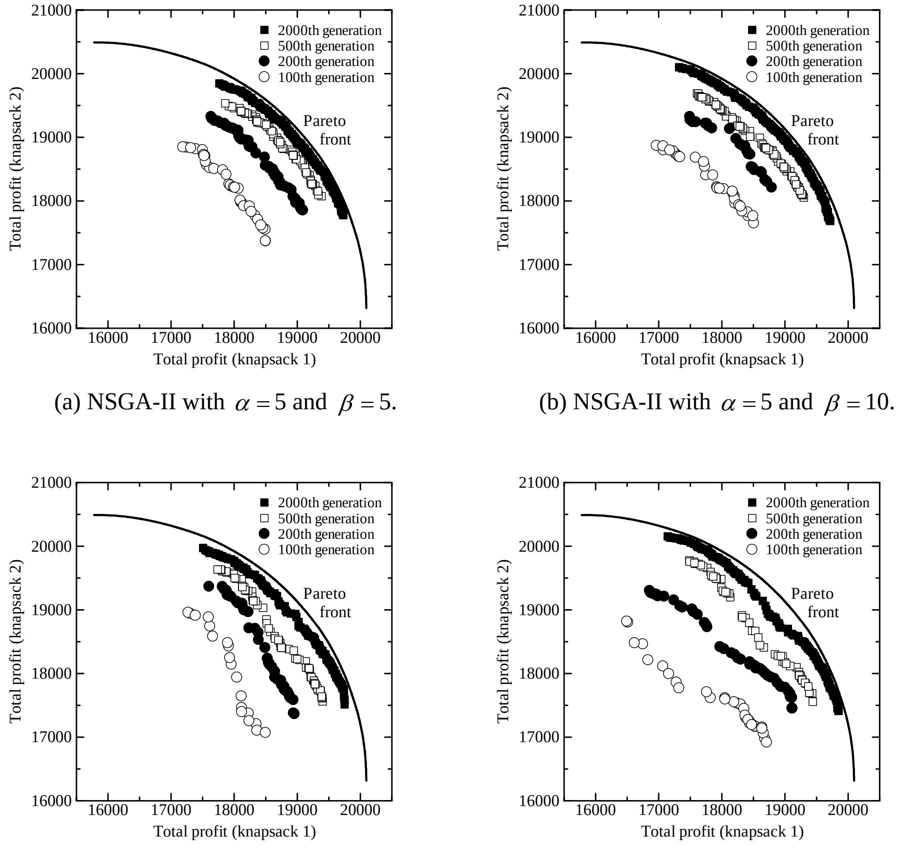
(b) Dissimilar parents ($\beta = 5$).

(c) Similar parents ($\beta = 10$).

(d) Dissimilar parents ($\beta = 10$).

Fig. 3. Effects of recombining similar or dissimilar parents on the performance of NSGA-II.

From the comparison between Fig. 2 (c) and Fig. 3, we can see that the performance of the NSGA-II algorithm was slightly improved by the recombination of similar parents and degraded by the recombination of dissimilar parents. Since the performance improvement of the NSGA-II algorithm was not significant in Fig. 3, we further examine not only the recombination of similar parents but also the selection of extreme parents using the similarity-based mating scheme in Fig. 1. Experimental results by the NSGA-II algorithm with the mating scheme are shown in Fig. 4 for various values of α and β. As in Fig. 2 (c) and Fig. 3, we used the crossover rate 0.8 and the mutation rate 0.002 in Fig. 4. From the comparison between Fig. 2 (c) by the original NSGA-II algorithm and Fig. 4 with the similarity-based mating scheme, we can see that the performance of the NSGA-II algorithm with respect to the diversity of solutions was significantly improved without severe deterioration in the convergence by recombining extreme and similar parents in Fig. 4.



(a) NSGA-II with $\alpha = 5$ and $\beta = 5$.

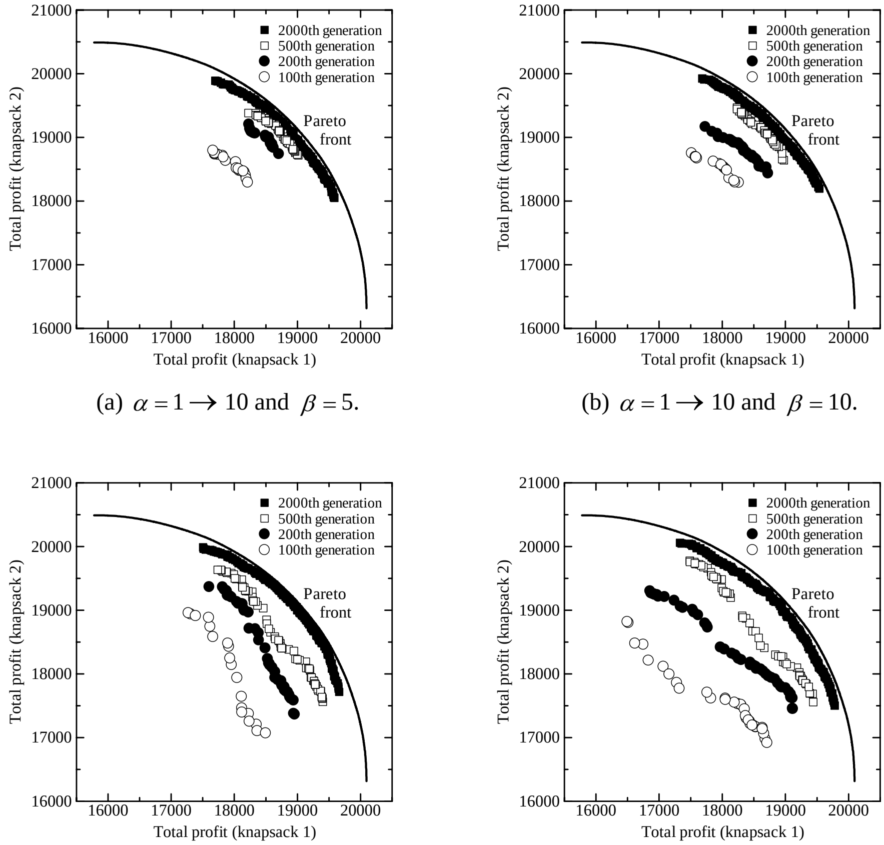
(b) NSGA-II with $\alpha = 5$ and $\beta = 10$.

(c) NSGA-II with $\alpha = 10$ and $\beta = 5$.

(d) NSGA-II with $\alpha = 10$ and $\beta = 10$.

Fig. 4. Effects of choosing extreme parents and recombining similar parents.

While the recombination of extreme and similar parents significantly improved the diversity of solutions obtained by the NSGA-II algorithm, some experimental results in Fig. 4 were inferior to those in Fig. 2 (c) by the original NSGA-II algorithm with respect to the convergence of solutions to the Pareto front. So we further examine the effect of dynamically changing the parameter values in the similarity-based mating scheme in Fig. 1 (see Ishibuchi & Shibata [3] for the dynamic version of the similarity-based mating scheme). In computational experiments, we changed the value of α at the 1000th generation (e.g., from $\alpha = 10$ to $\alpha = 1$). Experimental results are shown in Fig. 5. Very good results were obtained in Fig. 5 (c) and Fig. 5 (d) where the value of α was specified as $\alpha = 10$ in the first half and $\alpha = 1$ in the second half of the multiobjective evolution by the NSGA-II algorithm with the similarity-based mating scheme. It should be noted the first half in Fig. 5 (c) and Fig. 5 (d) were exactly the same as Fig. 4 (c) and Fig. 4 (d), respectively.



(a) $\alpha = 1 \rightarrow 10$ and $\beta = 5$.

(b) $\alpha = 1 \rightarrow 10$ and $\beta = 10$.

(c) $\alpha = 10 \rightarrow 1$ and $\beta = 5$.

(d) $\alpha = 10 \rightarrow 1$ and $\beta = 10$.

Fig. 5. Effects of the dynamic control of the parameters in the similarity-based mating scheme.

4 Concluding Remarks

Through computational experiments on multiobjective 0/1 knapsack problems, we examined the effect of the standard one-point crossover operation on the performance of the NSGA-II algorithm. First we showed that the crossover operation improved the convergence of solutions to the Pareto front while it decreased the diversity of solutions. Next we showed that the recombination of similar parents slightly improved the performance of the NSGA-II algorithm through computational experiments using a similarity-based mating scheme. Then we showed that the recombination of extreme and similar parents significantly improved the diversity of solutions. Finally we showed that the convergence and the diversity were dynamically controlled to find well-distributed near Pareto-optimal solutions using a dynamic version of the similarity-based mating scheme.

The effect of recombining similar parents was not large when the similarity-based mating scheme was not used for choosing extreme parents. The effect of choosing extreme parents was not large, either, when it was not used for recombining similar parents. That is, the recombination of extreme and similar parents had a large synergy effect. We also examined the effect of the similarity-based mating scheme through computational experiments on function optimization problems with real-number decision variables. Since the NSGA-II algorithm worked well on such an optimization problem, we did not observe any further improvement by the mating scheme.

The author would like to thank the financial support from Japan Society for the Promotion of Science (JSPS) through Grand-in-Aid for Scientific Research (B): KAKENHI (14380194). The author also would like to thank Mr. Kaname Narukawa for his help with some computational experiments.

References

1. 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, Vol. 2632 (Proc. of EMO 2003), Springer, Berlin (2003) 433-447.
2. Ishibuchi, H. and Shibata, Y.: A Similarity-based Mating Scheme for Evolutionary Multiobjective Optimization. Lecture Notes in Computer Science, Vol. 2723 (Proc. of GECCO 2003), Springer, Berlin (2003) 1065-1076.
3. Ishibuchi, H. and Shibata, Y.: Mating Scheme for Controlling the Diversity-Convergence Balance for Multiobjective Optimization. Lecture Notes in Computer Science, Vol. 3102 (Proc. of GECCO 2004), Springer, Berlin (2004) 1259-1271.
4. Ishibuchi, H. and Narukawa, K.: Recombination of Similar Parents in EMO algorithms. Lecture Notes in Computer Science, Vol. 3410 (Proc. of EMO 2005), Springer, Berlin (2005) 265-279.
5. Deb, K., Agrawal, S., Pratap, A., and Meyarivan, T.: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. IEEE Trans. on Evolutionary Computation **6** (2002) 182-197.
6. 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.