EFFECTIVENESS OF NEIGHBORHOOD CROSSOVER IN MULTIOBJECTIVE GENETIC ALGORITHM

Kengo Yoshii
Graduate School of Engineering
Doshisha University
Kyoto Kyotanabe-shi Japan
email: kyoshii@mikilab.doshisha.ac.jp

Tomoyuki Hiroyasu
Department of Knowledge Engineering
Doshisha University
Kyoto Kyotanabe-shi Japan
email: tomo@is.doshisha.ac.jp

Mitsunori Miki
Department of Knowledge Engineering
Doshisha University
Kyoto Kyotanabe-shi Japan
email: mmiki@mail.doshisha.ac.jp

ABSTRACT

In this paper, the effectiveness of the neighborhood crossover of EMO algorithms is discussed through the numerical experiments. The neighborhood crossover chooses two parents which are close to each other in the objective space. All the individuals are sorted in order of proximity in the objective space, and then neighborhood shuffle is conducted, which randomly replaces individuals in the population at a fixed width interval of population size. This operation prevents crossing over repeatedly between the same pair of individuals.The width of neighborhood shuffle is the parameter of this operation and this parameter determines the range of the population where individuals are shuffled. Therefore, this parameter affects the quality of the solutions. We implemented the NSGA-II with the neighborhood crossover and examined the effect of the width of neighborhood shuffle to further investigate the effectiveness of neighborhood crossover. The results of the numerical experiment indicated that the effect of neighborhood crossover can be achieved by applying neighborhood crossover to the search population created through copy selection. In addition, the necessity of neighborhood shuffle and an appropriate width of neighborhood shuffle were reviewed.

KEY WORDS

Multi-Objective Genetic Algorithms, Neighborhood Crossover, Neighborhood shuffle, Optimization

1 Introduction

A problem with multiple criteria where the criteria are in a trade-off relationship is called a multiobjective problem. There are various methods of solving multiobjective problems, but the focus of this study was evolutionary multiobjective optimization (EMO), where a set of solutions superior to any solution called Pareto optimal solutions is obtained at one time. The most important goal of EMO is to discover the non-dominated solutions equivalent to the Pareto optimum front, or a solution similar to that with great diversity. Various algorithms of such approaches have been proposed after Shaffer's VEGA [1]. Among them, Deb's Elitist Non-Dominated Sorting Genetic Algorithm (NSGA-II) [2] and Zizler's Strength Pareto Evolutionary Algorithm 2 (SPEA2) [3] have incorporated important mechanisms for the MOGAs, such as preservation of individuals with high degree of fitness and method for reduction of individuals based on diversity, and these have been reported to yield excellent solutions.

Meanwhile, we have been improving these important mechanisms by incorporating the neighborhood crossover that crosses over by individuals which are close to each other in the objective space, and the search ability has been improved [4, 5]. In the neighborhood crossover, individuals are sorted in the order of proximity of distance between the individuals in the objective space, and then the neighborhood shuffle is conducted to randomly rearrange the individuals in a fixed width of the population. The width of neighborhood shuffle over which this neighborhood shuffle is conducted is a parameter, and the proximity between individuals changes depending on this parameter, and it greatly influences the search performance of the solution. In this study, the neighborhood crossover was built into NSGA-II, which is a typical method, to further investigate neighborhood crossover and the influence of the width of neighborhood shuffle on neighborhood crossover. The required condition to achieve the effects of neighborhood crossover and appropriate width of neighborhood shuffle were also examined.

2 Evolutionary multiobjective optimization

The Genetic Algorithm (GA) is an optimization method modeled on the inheritance and evolution of organisms in the natural world. As GA is a multi point search, which is different from the conventional one point search method, two or more Pareto optimum solutions can be obtained through a single search.

In MOGAs that apply GA to multiobjective problems, the search goal to obtain various Pareto optimum solutions in the objective space with a high degree of accuracy. Important mechanisms that have been proposed to achieve this goal are summarized below.

1. Preservation to archive
   The preservation of the Pareto optimum solution to the archive has been incorporated into many recent algorithms. This operation is achieved by generating the archive separate from the search population, and preserving superior individuals of each search phase in the archive [6, 3, 2, 7, 8].

2. Environmental selection
   The selection of the solutions to be preserved in the archive is called environmental selection. The individuals preserved in the archive are generally those individuals with a high degree of fitness. However, when the number of non-dominated solutions exceeds the size of the archive, the solution are selected with consideration of the degree of overcrowding of the individuals. The selection technique that considers the level of overcrowding of the solutions includes methods that use sharing[9]: crowding distance is used in NSGA-II [2], and the Archive truncation method is used in SPEA2 [3].

3. Mating selection
   Selection of the search population of the next generations from the archive is called mating selection. In methods such as NSGA-II and SPEA2, the search has been accelerated by generating a search population from individuals with high degree of fitness that are preserved in the archive.

4. Fitness degree allocation
   As two or more objective functions exist in the MOGAs, the objective function value cannot be applied as a degree of fitness as in single-objective GA. In addition, the method of allocating the degree of fitness with consideration of the dominated relation between individuals is proposed. Typical methods use ranking [9], allocation of the degree of fitness based on the number of dominated individuals [3], and nondominated sorting method [2].

3 Application of neighborhood crossover to MOGAs

3.1 Crossover in MOGAs with consideration to proximity

The role of crossover is generating good offsprings from good parent individuals. In MOGAs, good individuals are nondominated solutions and are scattered in the population. On the other hand, a typical MOGAs method includes generally either one-point or multi-point crossover. In these crossovers, there is a problem such that individuals in a crossover pair are chosen at random, and the Euclid distance in the variable space between the individuals is too large that generated offsprings have a high possibility of being dominated solutions. To prevent this, it is necessary to consider the proximity of the individuals somehow, when the individuals to be crossed over are chosen. By crossing over individuals adjacent in the variable space, the offspring can be generated near the good parent individuals and would be also a nondomitend solution, thus creating a population with great diversity. However, there are some cases where the Euclid distance in the variable space cannot be defined, such as in the combinational function. Generally, in the continuous function, there is a high possibility of individuals adjacent in the objective space also being adjacent in the variable space. For the above reasons, in this study, crossover was conducted on individuals adjacent in the objective space rather than in the variable space. With this background, various studies have been conducted on neighborhood crossover, where crossover between individuals in close proximity to one another is conducted [4, 5].

3.2 Algorithm of neighborhood crossover

Neighborhood crossover is a crossover between individuals with short Euclid distance in the objective space. The search ability can be improved by specifically crossing over individuals that are close to each other in the objective space. The algorithm of the neighborhood crossover is shown below.

1. From the best individual for one of the function values, sort the population in close order in the objective space.

2. Neighborhood shuffle which changes individuals randomly in certain width of the population size is performed for the sorted population.

3. Select 2*ith and 2*i+1th individuals as parents and crossover is performed

Neighborhood shuffle is a very important operation in the neighborhood crossover. If neighborhood shuffle is not conducted, crossover will be conducted with the same pair in every generation, and thus it will be impossible to escape when falling into a localized solution. Therefore, it is important to perform neighborhood shuffle with a width of moderate size. The width of neighborhood shuffle in which this neighborhood shuffle is conducted is a parameter, and is decided depending on the ratio of the neighborhood shuffle width (R_{nsw}). R_{nsw} is a real number between 0 and 1.0, and the size of the width of neighborhood shuffle is represented as the ratio to the size of the population. For example, R_{nsw} 0.1 means that the neighborhood shuffle is conducted with a width that is 10% of the population. The proximity of the individuals changes depending on the size of R_{nsw}, and the proximity increases with decreasing this size, but this also increases the possibility that crossover will be conducted repeatedly in the same pair. In the next section, the influence of changes in width of neighborhood shuffle on solution search ability is examined through a numerical experiment, to investigate the effects of neighborhood crossover.

4 Effectiveness of neighborhood crossover

To investigate the effect of neighborhood crossover, a numerical experiment was conducted using a test function, and using a typical MOGA method, NSGA-II, incorporating neighborhood crossover. The influence of the width of neighborhood shuffle on neighborhood crossover was also examined.

4.1 Experimental method

The target problems used in this experiment were: KUR used for numerical experiment of Kursawe as a continuous function [10], and knapsack problem with two objectives and 750 items (KP750-2) from multiobjective knapsack problems used in the numerical experiments of Zitzler (knapsack problems) as a combinational function [11].

KUR

$$\begin{cases} \min \quad f_1 = \sum_{i=1}^{n}(-10\exp(-0.2\sqrt{x_i^2 + x_{i+1}^2})) \\ \min \quad f_2 = \sum_{i=1}^{n}(|x_i|^{0.8} + 5\sin(x_i)^3) \\ \text{s.t.} \\ \qquad\qquad x_i[-5,5],\ i = 1,\ldots,n,\ n = 100 \end{cases} \tag{1}$$

KUR is a problem that involves interaction between two consecutive variables in f1(x), and is multi-convex in f2(x). In this experiment, this problem was handled as a problem with 100 design variables, which made the search more difficult.

$\mathbf{KP750-n}$

$$\begin{cases} \max f_i(x) = \sum_{j=1}^{750} x_j \cdot p_{(i,j)} \\ \text{s.t.} \\ \qquad g_i(x) = \sum_{j=1}^{750} x_j \cdot w_{(i,j)} \le W_i \\ \qquad\qquad 1 \le i \le k, k = 2 \end{cases} \tag{2}$$

While the multiobjective knapsack problem is very simple and easy to implement, the search of the problem is very difficult. $p_{(i,j)}$ and $w_{(i,j)}$ in the above expression represent the profit value and the weight value that accompany the jth item when calculating the evaluation values of the ith item, respectively. In addition, W_i is the restriction value (upper limit) to the sum total of the weight value in calculating the evaluation values of the ith item.

Though there are various methods to evaluate the obtained non-dominated solutions, in this study, we will use the evaluation method outlined below.

1. cover rate: Icover
2. Spread [11]
3. Ratio of Non-dominated Individuals:RNI [12]

Icover is a method of absolutely evaluating the obtained non-dominated solution; it evaluates whether the solution set has a uniform distribution in the Pareto optimum solution domain in the objective space. Icover can be obtained from the proportion of the number of small areas k_i, where non-dominated solutions exist in objective function I, of which Pareto optimum solution areas for each objective function are divided into K sections. The following is the equation to obtain Icover in the target problem of N objective function.

$$I_{cover} = \frac{1}{N}\sum_{i=1}^{N}\frac{k_i}{K}$$

The above equation shows that the closer Icover is to 1.0, the more the solution is from all over the region. In this experiment, the population size was set to the division number K.

Spread is calculated using the following equation. The range of Pareto optimum solutions increases as larger values are used.

$$\text{Spread} = \sum_{i=1}^{N}[\max f_i(x) - \min f_i(x)]$$

RNI is the method used by Tan expanded to compare the two non-dominated solutions [12]. In RNI, the sumset of the solution sets X and Y obtained by two methods is set as S^U. Next, solutions non-dominated by any other solution are selected from S^U, and the selected set of solutions is S^P. Finally, the proportion of the solution from each method in S^P is selected as $I_{RNI(X,Y)}$. Therefore, the closer this ratio is to the maximum value of 100%, it is better than other methods; i.e., the obtained solutions are closer to the real solution.

Our numerical expriments on KUR are performed under the following parameter specifications:

Population Size : 100(KUR), 250(KP750-2)
The number of Dimension : 100(KUR))
Chromosome Length : 20*The number of Dimension(KUR), 750(KP750-2)
Crossover Probability : 1.0
Mutation Probability : 1/Chromosome Length
Stopping condition : 250 generations(KUR),
2000 generations(KP750-2)

The average value of each evaluation method is calculated over 30 runs with diffrent initial populations.

In each target problem, various R_{nsw} are examined. In KUR, because the size of the population is 100, R_{nsw} with values of 0.0, 0.05, 0.1, 0.2, 0.25, 0.5, and 1.0 are examined. R_{nsw}0.0 means that neighborhood shuffle is not executed after the sort, and as R_{nsw}1.0 executes neighborhood shuffle in width of the size of the population after the sort, it will be the same as the original NSGA-Ⅱ. In addition, in KP750-2, because the size of the population is 250, R_{nsw} with values 0.0, 0.02, 0.04, 0.1, 0.2, 0.5, and 1.0 are examined.

4.2 Examination of the effectiveness of the neighborhood crossover

Icover, Spread, and RNI compared to the original NSGA-I I through changes in R_{nsw} in KUR, are shown in Figure.1.

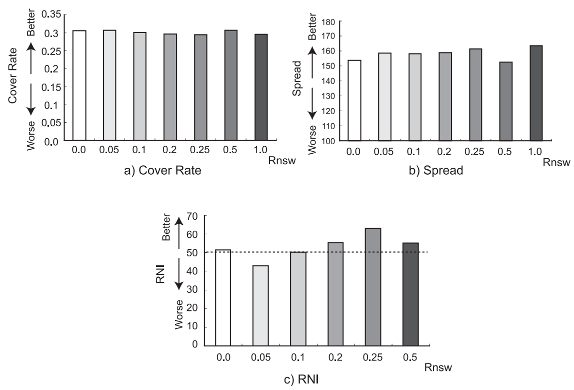

Figure 1. Results of Icover, Spread, RNI for KUR

From the results of Icover in Figure.1, no significant effect of neighborhood crossover is seen compared to the original NSGA-Ⅱ. In addition, there was no marked difference in the influence of R_{nsw} on neighborhood crossover. The results of Spread indicated that the performance of neighborhood crossover is worse than NSGA-Ⅱ for any R_{nsw}, and the broadness is lost. For RNI, although R_{nsw}0.25 had relatively good results, no marked difference was observed with changes in the R_{nsw}, and thus there was no effect of neighborhood crossover.

Similarly, the results for KP750-2 are shown in Figure.2.

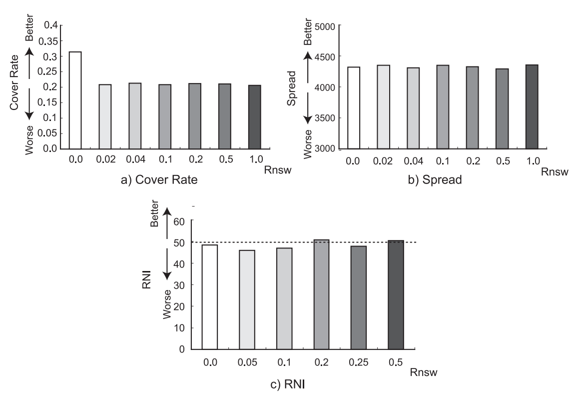

Figure 2. Results of Icover, Spread, RNI for KP750-2

No significant effect of the neighborhood crossover is seen for KP750-2. Thus, we believe that the effect is small just by simply introducing neighborhood crossover, and thus certain conditions are required to achieve effectiveness. The next section examines the conditions for achieving effectiveness of the neighborhood crossover.

5 Effectiveness of copy selection on neighborhood crossover

5.1 Examination of the copy selection

In chapter 4, neighborhood crossover was incorporated into NSGA-Ⅱ to investigate the effect of neighborhood crossover using various widths of neighborhood shuffle, comparing the results to the original NSGA-Ⅱ, but no significant differences were observed. Here, we consider the conditions that will generate effectiveness of neighborhood crossover. Typical methods, such as NSGA-Ⅱ and SPEA2, use tournament selection as the mating selection method to select the search population from the archive. This is to speed up convergence through a search using more superior individuals. However, superior individuals selected by tournament selection from the archive are selected redundantly more than once. On the other hand, as the neighborhood crossover first sorts the individuals in the order of proximity, in the population formed by tournament selection, the possibility of the same individuals being side by side is higher, and the possibility of unnecessary crossovers between the same individuals is therefore also higher. Thus, mating selection in neighborhood crossover, which requires a population that consists of individuals as different as possible, is believed to be more effective using copy selection that just copies the archive population, rather than using tournament selection. In next section, a comparative experiment is described where the search population of the next generations is generated with copy selection, instead of using tournament selection, from NSGA-Ⅱ incorporating neighborhood crossover and executing neighborhood crossover. The target problem and the parameters are the same as in the preceding section.

5.2 Experimental results

Icover, Spread, and RNI compared to the original NSGA-I I through changes in ratio of width of neighborhood shuffle (R_{nsw}) in KUR are shown in Figure.3. The results of original NSGA-Ⅱ for Icover and Spread are also shown for comparison.

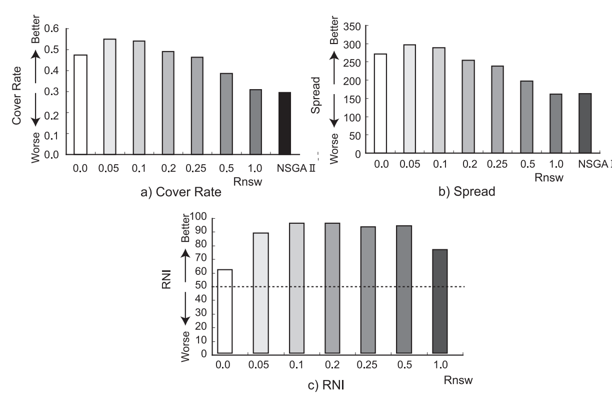

Figure 3. Results of Icover, Spread, RNI for KUR when copy selection is adapted

The results shown in Figure.3 confirm the effectiveness of neighborhood crossover. Especially, when R_{nsw} is between 0.05 and 0.2, the broadest Pareto optimum solutions are obtained. When R_{nsw}0.0, i.e., when the neighborhood shuffle is not conducted, the search was believed to be influenced because the frequency of crossover in the same pair increases. In original NSGA-II, as the search population is generated from the archive through sampling with replacement, convergence to the Pareto optimum solution of the search individuals becomes faster, but the population becomes more likely to lose its diversity, and falls more easily into localized solutions. Therefore, it is difficult to obtain an excellent solution in a multi-convex problem, as in KUR. However, the search ability will be improved by maintaining diversity through copy selection and by incorporating neighborhood crossovers. The Pareto optimum solutions for KUR obtained from the 30 trials using R_{nsw} 0.1 compared to NSGA-II are shown in Figure.4. The effectiveness of the neighborhood crossover was confirmed visually, and the obtained non-dominated solutions can be seen to have great diversity.

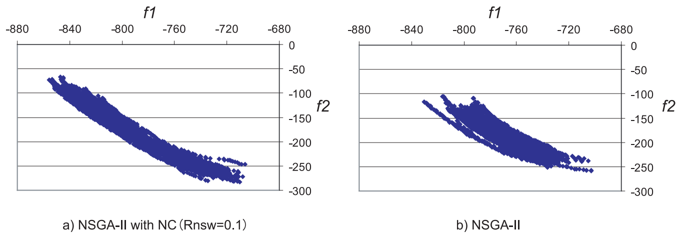

Figure 4. Solutions obtained by NSGA-I with NC and original NSGA-II for KUR

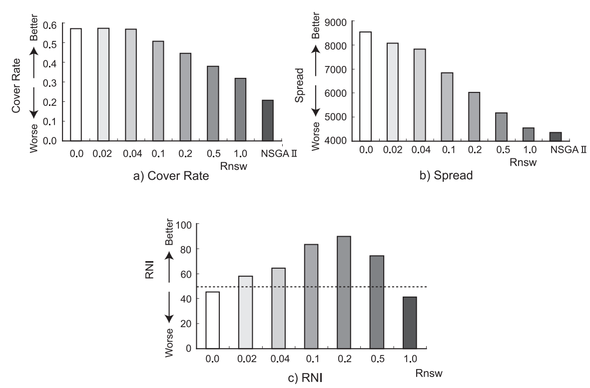

Figure 5. Results of Icover, Spread, RNI for KUR when copy selection is adapted

Similarly, the results of KP750-2 are shown in Figure.5. KP750-2 is a problem with a very wide Pareto optimum front, and the design variable value of the Pareto optimum individuals forming the Pareto optimum front also contains a large amount of diversity. Therefore, similar to KUR, to search for a wide range of Pareto optimum solutions, the diversity of the population is very important. Figure.5 shows that a Pareto optimum solution set with great diversity is obtained, and the effectiveness of neighborhood crossover was confirmed.

The above results show that neighborhood crossover is effective when the search population of next generation is generated from the archive using copy selection.

5.3 Necessity of neighborhood shuffle

We examined various ratios of neighborhood shuffle width in section 5.1, and found that RNI is worse using R_{nsw}0.0 in both target problems. When R_{nsw}0.0, Icover, and Spread increase, but as neighborhood shuffling will not be executed, the frequency of the crossovers between the same pairs of individuals as in the previous generation will also increase, and the search is influenced. To confirm this, we indicate the number of times crossovers took place in the same pair as in the previous generation, for each target problem Figure.6.

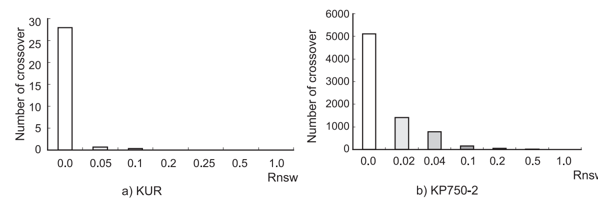

Figure 6. Number of times crossover was conducted by individuals of the same pair as in the previous generation.

From Figure.6, we can see that more crossovers took place in the same pair as in the previous generation when the neighborhood shuffle width is smaller. In addition, from the results shown in Figure.3 and Figure.5, proximity decreases, and the width and diversity are lost when R_{nsw} is larger than 0.5. Therefore, it is important for neighborhood crossover to have an appropriate neighborhood shuffle width for neighborhood shuffling.

5.4 Appropriate width of neighborhood shuffle

From the results of RNI in Figure.3 and Figure.5, we can see that better results are obtained than the original NSGA-I I even when R_{nsw} is 1.0. As there was no significant difference in the results of Spread, the convergence toward the Pareto optimum solution is great in RNI with good results. That is, in these target problems, good results are obtained simply by using copy selection instead of tournament selection. Search ability increases when neighborhood crossover is conducted after using copy selection. Thus, a wide range and great diversity of non-dominated solutions can be obtained. To confirm the effectiveness of neighborhood crossover using appropriate R_{nsw}, Figure.7 shows the RNI in comparison with R_{nsw}1.0 for each target problem. In Figure.7, good results are obtained in all R_{nsw} except R_{nsw}0.0. In addition, in Figure.7, we can see that the best ratio of neighborhood shuffle width is around R_{nsw}0.2.

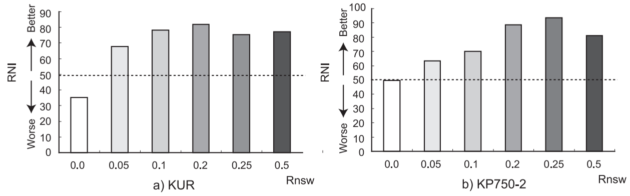

Figure 7. Comparison with R_{nsw}1.0 in RNI

5.5 Characteristics of a search when neighborhood crossover is introduced

To examine the search process of the neighborhood crossover with an appropriate width of neighborhood shuffle, the search process in KP750-2 at R_{nsw}0.2 is shown in Figure.8.

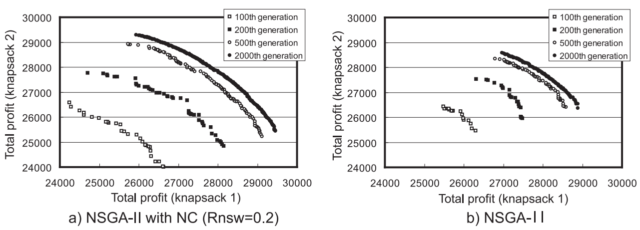

Figure 8. Solution search history for KP750-2

In comparison with the original NSGA-II search process, although the speed of convergence is inferior, wide-ranging non-dominated solutions with great diversity are obtained from the early stages of the search. These observations indicate that it is possible to conduct a search while maintaining the diversity of the population, and to obtain a wide range of Pareto optimum solutions using neighborhood crossover.

6 Conclusions

The influence of the width of neighborhood shuffle on solution search was examined to further investigate the neighborhood crossovers that have been proposed previously. The effect of neighborhood crossover is small even if applied to the search population generated with tournament selection used as a mating selection method. This is because more superior individuals are redundantly selected through tournament selection and the diversity is lost, and because the possibility of crossover taking place between the same individuals is high. On the other hand, when neighborhood crossover is applied to the population generated with copy selection used as a mating selection method, the search could be conducted while maintaining the diversity of the population. In addition, it was confirmed that the best Pareto optimal solutions are obtained when neighborhood shuffling is conducted with the width of neighborhood shuffle being approximately 20% of the population size.

References

[1] J. D. Schaffer. Multiple objective optimization with vector evaluated genetic algorithms. In *Proceedings of the 1st International Conference on Genetic Algorithms*, pages 93–100, Mahwah, NJ, USA, 1985. Lawrence Erlbaum Associates, Inc.

[2] A. Pratab K. Deb, S. Agrawal and T. Meyarivan. A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. In Marc Schoenauer, Kalyanmoy Deb, Günter Rudolph, Xin Yao, Evelyne Lutton, J. J. Merelo, and Hans-Paul Schwefel, editors, *Proceedings of the Parallel Problem Solving from Nature VI Conference*, pages 849–858, Paris, France, 2000. Springer. Lecture Notes in Computer Science No. 1917.

[3] M. Laumanns E. Zitzler and L. Thiele. SPEA2: Improving the Strength Pareto Evolutionary Algorithm. Technical Report 103, Gloriastrasse 35, CH-8092 Zurich, Switzerland, 2001.

[4] T. Hiroyasu M. Kim and M. Miki. Spea2+: Improving the performance of the strength pareto evolutionary algorithm2. *Parallel Problem Solving from Nature - PPSN VIII*, pages 742–751, 2004.

[5] T. Hiroyasu S. Watanabe and M. Miki. Neighborhood cultivation genetic algorithm for multi-objective optimization problems. *Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution And Learning (SEAL-2002)*, pages 198–202, 2002.

[6] K. Deb E. Zitzler and Lothar Thiele. Comparison of multi-objective evolutionary algorithms: Empirical results. *Evolutionary Computation*, 8(2):173–195, 2000.

[7] T. Murata and H. Ishibuchi. Moga : Multi-objective genetic algorithms. *Proceedings of the 2nd IEEE International Conference on Evolutionary Computing*, pages 289–294, 1995.

[8] K. Yoshida S. Kobayashi and M. Asada. Generating a set of pareto optimal decision trees by genetic algorithms. *Journal of the Japanese Society for Artificial Intelligence*, 11(5):725–732, 1996.

[9] N. Nafpliotis J. Horn and D. E. Goldberg. A Niched Pareto Genetic Algorithm for Multiobjective Optimization. In *Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*, volume 1, pages 82–87, Piscataway, New Jersey, 1994. IEEE Service Center.

[10] F. Kursawe. A Variant of Evolution Strategies for Vector Optimization. In H. P. Schwefel and R. Männer, editors, *Parallel Problem Solving from Nature. 1st Workshop, PPSN I*, volume 496 of *Lecture Notes in Computer Science*, pages 193–197, Berlin, Germany, oct 1991. Springer-Verlag.

[11] E. Zitzler and Lothar Thiele. Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. *IEEE Transactions on Evolutionary Computation*, 3(4):257–271, November 1999.

[12] T. H. Lee K. C. Tan and E. F. Khor. Increamenting Multi-objective Evolutionary Algorithms: Performance Studies and Comparisons. *First International Conference on Evolutionary Multi-Criterion Optimization*, pages 111–125, 2001.