Discussion of Clustering and Network Inversion for Multi-Objective Genetic Algorithms

Kenji Kobayashi
Graduate Student, Department of Knowledge Engineering and Computer Sciences, Doshisha University, 1-3 Tatara Miyakodani,Kyo-tanabe, Kyoto, 610-0321, JAPAN
kkobayashi@mikilab.doshisha.ac.jp

Tomoyuki Hiroyasu
Department of Knowledge Engineering and Computer Sciences, Doshisha University, 1-3 Tatara Miyakodani,Kyo-tanabe, Kyoto, 610-0321, JAPAN
tomo@is.doshisha.ac.jp

Mitsunori Miki
Department of Knowledge Engineering and Computer Sciences, Doshisha University, 1-3 Tatara Miyakodani,Kyo-tanabe, Kyoto, 610-0321, JAPAN
mmiki@mail.doshisha.ac.jp

Abstract

Recently, multi-objective genetic algorithms have been applied to real-world problems and the good results were derived. When evaluation function needs a huge calculation time to derive fitness value, high calculation cost becomes a problem. One solution to this problem is to perform the search with a small population size. With this solution, however, the diversity of the solutions is often lost. This means that Pareto optimal solutions with high diversity may not be obtained. To resolve this issue, we proposed a new diversity maintenance method using Artificial Neural Network (ANN). In this method, the converged solutions on certain points are relocated uniformly in the objective space by inverse analysis using ANN. In this paper, we improved the proposed method using clustering method before inverse analysis. This mechanism improves the approximation ability of ANN. The proposed method was introduced to NSGA-II, and its effectiveness was examined on mathematical test functions. In some test functions, the proposed method provided solutions with a high degree of diversity, even when the search is performed with a small number of solutions on the high dimensional problems. The effectiveness of the clustering before inverse analysis was also discussed through the experiments.

Keywords: Multi-objective genetic algorithm, Neural network, Inverse problem, Diversity, Clustering.

1 Introduction

Recently, multi-objective genetic algorithms have been applied to real-world problems and the good results were derived [1, 2]. However, in real-world problems, it usually takes a large amount of time to evaluate one parameter set. Therefore, in these problems, even if fine algorithms with effective search ability are applied, satisfactory results cannot be obtained with insufficient calculation time. To handle this problem, two approaches have been mainly proposed.

The first approach is to perform a MOGA search with parallel processing. In such parallel models, MOGA operations are executed in many computational resources sharing the processes among them. Such parallel studies have been reported by many researchers, most of which have made use of the master-slave or island model [3, 4, 5, 6, 7, 8]. Especially, it is reported that Deb's parallel model can obtain good Pareto optimal solutions while reducing the calculation time.

The other approach is to reduce the calculation cost of MOGA operations by improving the design of algorithms. In this paper, such algorithms requiring low calculation cost are discussed.

2 Approach for Reducing Calculation Cost and Its Issues

2.1 Approach for Reducing Calculation Cost

It is important to consider not only the quality of Pareto optimal solutions, but also the length of calculation time. Therefore, we need to design algorithms that can derive reasonable solutions with fewer evaluation calls in MOGA search. There are two approaches to develop such algorithms.

The first approach is to use the response surface methodology [9], which is a technique for approximating objective functions. This method reduces the calculation cost by generating approximations of objective functions and treating those as the objective functions for each evaluation. There are several response surface methodologies, such as the quadratic polynomial model [10], the neural network model [11, 12, 13], and Kriging model [14]. Among these, the quadratic polynomial model is commonly used, because it is the simplest and has low calculation cost for approximation. Although the approximation cost associated with the other models is greater than that of the quadratic model, the neural network model and Kriging model allow approximation of more complicated objective functions [15].

On the other hand, the method discussed here involves a search with a small number of solutions. This approach can reduce the calculation cost, but the search solutions often converge to a few points in the search process. This may have a negative influence on the quality of the derived Pareto solutions, especially with regard to uniform distribution. We have proposed a mechanism that restores the diversity of the solutions using an Artificial Neural Network (ANN) [16]. In this mechanism, converged solutions are relocated according to target solutions that are distributed evenly on derived Pareto front in the proposed model. Using this mechanism, Pareto optimal solutions with a high degree of diversity can be obtained even when the search is performed with a small population size. In this paper, we improved the proposed method using clustering method before inverse analysis.

2.2 Issues to Overcome in the Proposed Mechanism

In our relocation mechanism, target solutions with uniform distribution are required to relocate converged solutions in the objective space. However, it is very difficult to determine the design variable values of these targets, as they exist in the objective space but not in the design space. Therefore, to obtain such desirable solutions, the design values must be determined through inverse analysis. A technique for ANN known as network inversion [17] was used here for this inverse analysis.

Similar to my research, many multi-objective optimization methods using ANN are reported in recent years, however, our goal is different from those other methods. Their goals can be classified mainly into two types: methods that reduce the calculation cost for each evaluation using an approximation function of the forward objective function [18], and those to obtain approximation of the inverse and apply it for local searches [19, 20]. On the other hand, our approach is to restore the diversity of solutions by relocating them.

But, in this relocation, there is another problem involving the solutions for training ANN. To generate fine approximations, solutions that exist in the neighborhood area of each other should be applied as the training data set for ANN. However, in some cases, even when the solutions derived by MOGA are closely distributed in the objective space, they may not be so in the design space. Therefore, we attempt to improve the accuracy of an approximation by applying clustering to such solutions.

3 Diversity Restoration Mechanism Using Clustering and Network Inversion

In this paper, a mechanism to restore the solution diversity using clustering and network inversion is proposed. This mechanism is composed of four processes: MOGA search, clustering, training ANN, and relocation. Details of these processes are described in the following sub-sections. With this mechanism, it is expected that the diversity will be restored, even when the solution diversity is lost in the small population search.

In addition, we propose the diversity maintenance method by iteration of the proposed mechanism. The application timing of diversity restoration mechanism is determined by the following two steps: 1)The first application is when the number of non-dominated solutions equals to the search population size in MOGA search. 2)The remaining application is determined by the equal interval to the remaining generations. The concept of the proposed method is shown in Figure. 1.

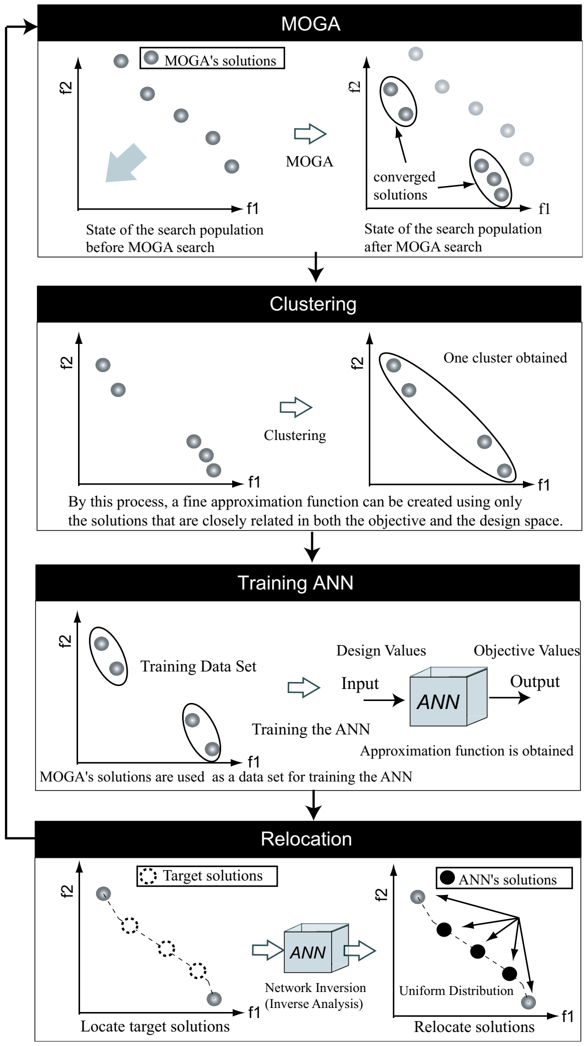
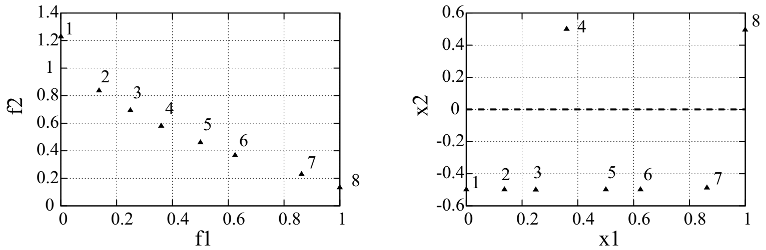


Figure 1: Concept of the Proposed Method.

3.1 Clustering

We can obtain Pareto solutions by MOGA search in various complex functions. However, even if they are adjacent in the objective space, they may not be so in the design space. The distribution of the population in multi-modal test problem ZDT4 is shown in Figure. 2 as an example of this situation.

In the left and right figures of Figure. 2, the ID indicates the same solution, and this ID is determined by the following clustering algorithm. In Figure. 2, solutions 3 and 4 are adjacent in the objective space, but not in the design space. When all of these solutions are used to generate an approximation function, it is difficult to obtain a fine approximation due to the noise effects of such solutions. To resolve this issue, before the approximation, it is required to select the solutions that exist close to each other in both spaces as the training data set. For this purpose, we proposed a new clustering algorithm as follows:



a)Objective Space b)Design Space

Figure 2: Distribution of the Population in the Objective and the Design Space.

base solution
$i = 0$: the solution with the minimum f1 value.
$i = 1$: the solution with the maximum f1 value.

Step 1-1: In the objective space, calculate the Euclid distances between all solutions and the base solution. (i=0)
Step 1-2: After sorting the solutions in ascending order by the distance, assign them IDs in ascending order.
Step 1-3: In the design space, calculate the Euclid distances between all solutions and the base solution. Then, sort by the distance in ascending order.
Step 1-4: From the sorted list of solutions, select solutions in ascending order while the following condition is satisfied: the solution has the larger (i=0) or smaller (i=1) ID than the previous solution.
Step 1-5: If i=0, update the base solution (i=i+1), and return to Step1-3.
Step 1-6: End if selected solutions in both cases are the same. If not, return to Step1-1 using the solutions of each cluster. (i=0)

The sorting process of the data set in Figure. 2 is shown in Figure. 3.

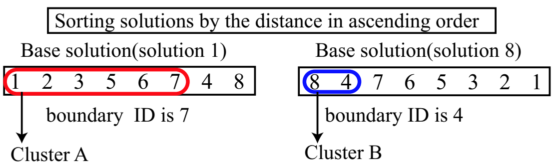


Figure 3: Sorting Process of Clustering.

By applying this algorithm to derived solutions of a MOGA search, a training data set without noise can be applied for approximations. For example, when this algorithm is applied to the data of Figure. 2, two clusters can be made: one cluster is composed of solutions 1, 2, 3, 5, 6, 7, and the other is 4 and 8. By creating an approximation by the solutions of each cluster, a fine approximation can be obtained.

3.2 Training ANN and Relocation

In this section, the relocation mechanism using network inversion is described. The algorithm of the proposed relocation mechanism is as follows:
Step 2-1: A linear line passing through the set of non-dominated solutions is obtained through interpolation.
Step 2-2: In a set of n non-dominated solutions , all solutions are removed except those on both ends. Then, $n - 2$ targets are created uniformly on the linear line by f1.
Step 2-3: The set of solutions after clustering is used as a data set for training ANN, and an approximation function is created.
(Input: design values; Output: objective values)

Step 2-4: Network inversion is performed and the design values corresponding to the objective values of the targets are obtained.
Step 2-5: The obtained design values are evaluated using the real objective function.
Step 2-6: Archive and solutions obtained from ANN are combined, and the archive update mechanism of NSGA-II is executed.

Inverse Analysis by Network Inversion

The design values of targets are obtained through network inversion. In this technique, the approximation function is trained by the forward relation of the objective function, and non-dominated solutions derived by MOGA search and clustering are used as the training data set. After obtaining approximations, a set of target objective values is applied as an output data to the fixed forward network, and a set of design values of solutions obtained by MOGA search and clustering is applied as an input. Then, preserving the weights of the fixed network, the input values are updated according to the calculated input correction signal. From this, the design values of the target solutions can be determined.

Determination of the Target Positions

Next, we describe how to determine the objective function values of the target solutions. There are two steps to obtain those values. In the first step, a linear interpolation line is obtained (Step 1). To this interpolation, a linear interpolation method was adopted, because it showed more positive results in preliminary experiments than two-dimensional interpolation. In the second step, targets are allocated to satisfy the following two conditions: 1) solutions lie on the interpolation line; 2) solutions are at equal distances with regard to f1. As it is difficult to set the targets in many objectives (more than three), this paper focuses on two-objective optimization problems. Many objectives problems (more than three) will be examined in future studies. The scheme of the determination of the target position is shown in Figure. 4.

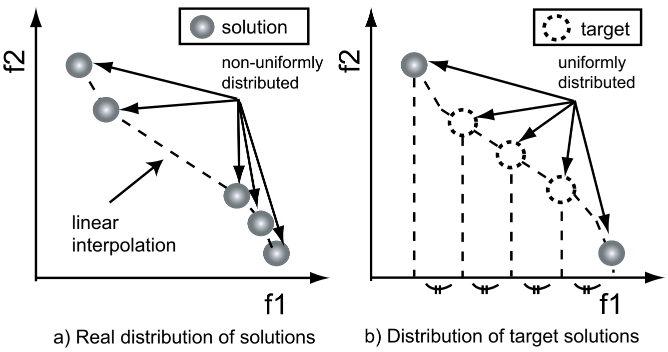


Figure 4: Concept of Detemination of the Target Positions.

The left diagram in Figure. 4 shows Pareto-optimal solutions, and the relocation concept is illustrated to the right in Figure. 4.

4 Effectiveness of Diversity Restoration Mechanism using Clustering and Network Inversion

In this section, we discuss the effectiveness of the proposed mechanism with clustering and network inversion. It is expected that this mechanism will provide a high degree of diversity to the derived solutions from MOGA search using network inversion. However, in some cases, even when these solutions are closely distributed in the objective space, they may not be so in the design space. For example, when a set of objective values can be obtained by multiple sets of design values, such situations tend to occur. In such cases, fine approximation cannot be obtained. Therefore, we attempt to resolve this problem by applying clustering before network inversion. Using the clustering operation, the approximation can be created by the solutions that exist closely in both the spaces. In this paper, as a preliminary study, we chose the ZDT1, ZDT2, and ZDT4 [21] problem as the problems with relatively smooth function landscape. Especially, ZDT4 is a multi-modal test problem requiring the clustering process to create the fine approximations. The equations of ZDT1, ZDT2, and ZDT4 are shown in Table. 1. The fixed parameters regarding NSGA-II in all experiments are illustrated in Table. 2.

Table 1: Test Problem (ZDT1, ZDT2, ZDT4)

Problem	Functions
ZDT1	min $f_1 = x_1$ min $f_2 = g \times h$ $g = (1 - (x_1/g)^(1/2))$ $g = (1 + 9(\sum_{i=2}^{n})/(n-1)$ $h = 1 - (x_1/g)^(1/2)$ $x_i \in [0,1]$
ZDT2	min $f_1 = x_1$ min $f_2 = g \times h$ $g = (1 - (x_1/g)^(2))$ $g = (1 + 9(\sum_{i=2}^{n})/(n-1)$ $h = 1 - (x_1/g)^(1/2)$ $x_i \in [0,1]$
ZDT4	min $f_1 = x_1$ min $f_2 = g \times h$ $g = 1 + 10(N-1)$ $+ \sum_{i=2}^{n}(x_i^2 - 10cos(4\pi x_i))$ $h = 1 - (f_1/g)^{0.5}$ $x_1 \in [0,1]$, $x_i \in [-5,5]$, $i = 2, \ldots, 10$

Table 2: Fixed Parameter Settings in All Experiments

Crossover rate	1.0
Method of crossover	One-point crossover
Length of chromosome	20 × dimension
Mutation rate	1/Length of chromosome

Evaluation Method

CoverRate —method to evaluate the distribution of the solutions in the objective space — is used to evaluate a set of non-dominated solutions. The value of Cover Rate is calculated by the following procedures. In a set of non-dominated solutions of the archive, the width, which is defined by the difference of the maximum and minimum value in an objective, is divided equally into K parts (K=the number of search solutions). Then, the number k_i of dominated domains that solutions exist in are counted. Cover Rate is calculated by iterating this process for all N objectives.

$$\text{Cover Rate} = \frac{1}{N}\sum_{i=1}^{N}\frac{k_i}{K}$$

When the solutions exist in each domain without duplication as much as possible, the indicator value is closer to 1.0. The concept of Cover Rate is shown in Figure. 5

GeneralDistance(**GD**)—method to evaluate the convergence of the solutions to the Pareto front — is used to evaluate a set of non-dominated solutions. The value of GD is defined as the following equation.

$$\text{GD} = \frac{\sqrt{\sum_{i=1}^{n} d_i^2}}{n}$$

n is the number of non-dominated solutions obtained by seach algorithms, and d_i is the Euclidian distance measured in the objective space between the each solution and the

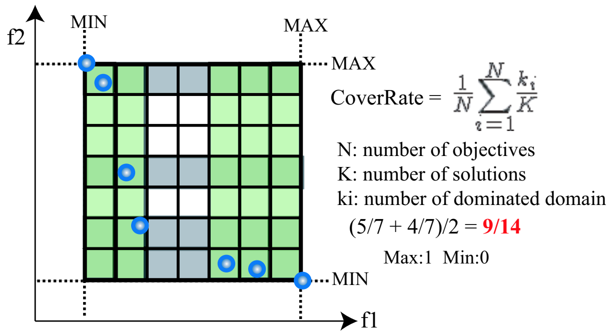


Figure 5: Concept of Cover Rate.

nearest one of the Pareto front. In this metric, GD=0 indicates that the all solutions exist on the Pareto front. Therefore, the smaller value means the better solutions obtained with regard to the convergence to the Pareto front.

4.1 Assessment of Approximation Ability using Clustering and Network Inversion

We proposed a new clustering algorithm regarding the vicinity among non-dominated solutions obtained by MOGA search in both the objective and the design space. In this experiment, the effects of clustering were examined by comparing two cases, with and without clustering. ZDT4 is selected as the test problem, because it is a difficult problem due to its multi-modality, that one function value is derived by multi-sets of the design variable values. In this case, clustering is an essential technique to generate fine approximations. The test data set is shown in Figure. 6. NSGA-II is applied as the MOGA method to introduce the proposed mechanism in all experiments. The population size was set to 10, and the number of design variables was 2.

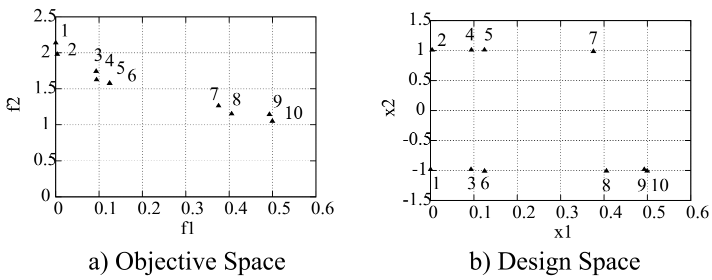


Figure 6: Data Set for Clustering.

In Figure. 6, the left figure shows the objective space of the test data set, and the right is the design space. In this test data set, some solutions adjacent in the objective space are not in the design space. To these solutions, two patterns of handling are compared: One is the method with clustering, and the other is without clustering. Then, after this operation, network inversion is applied to these solutions. The results are shown in Figure. 7.

In Figure. 7, the above figure shows the distribution of the target solutions in the objective space. The figures on the left and right show the distributions of the solutions with and without clustering respectively. Figure. 7 indicates that when the adjacent relation of the solutions are different in the objective and design space, application of clustering before network inversion improves the accuracy of the approximations.

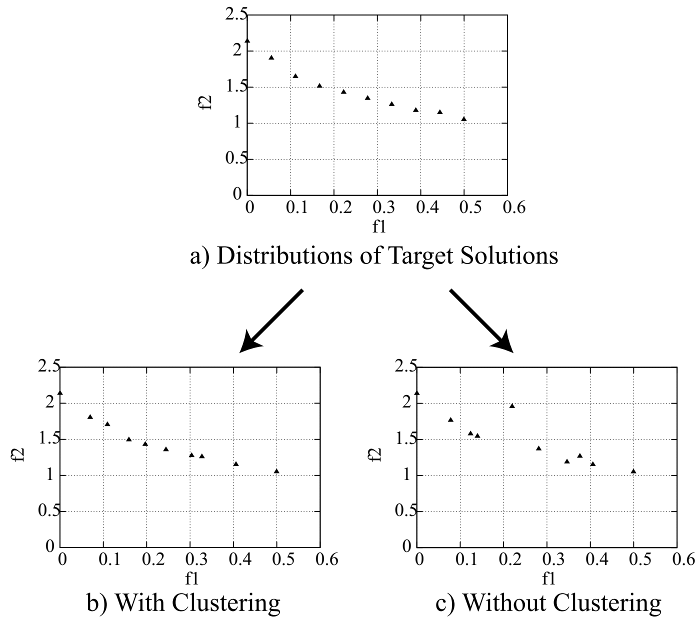


Figure 7: Comparison of the Results with and without Clustering.

4.2 Verification of the Effects of the Proposed Mechanism on Cover Rate

We next examined the improvement of solution diversity with the proposed mechanism by a numerical value. The proposed mechanism is mainly composed of a MOGA search and the diversity restoration step. In this experiment, the case with only a MOGA search and the case with both processes were compared in 30 trials. Three test problems (ZDT1, ZDT2, and ZDT4) were tested with the three patterns of the number of design variables. Cover Rate was adopted as the evaluation indicator of the solution diversity. The experiment parameters are shown in Table. 3. The results are shown in Figure. 8 to Figure. 10.

Table 3: Parameter Settings in the Verification of the Effects of the Proposed Mechanism on Cover Rate

Problem	ZDT1	ZDT2	ZDT4
Population size	10		
Generation	50		100
Dimension	2,5,10		
Number of application of the proposed mechanism	1		

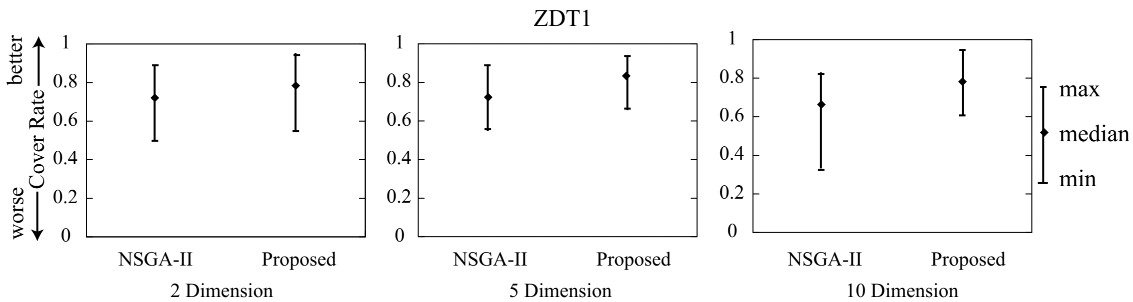


Figure 8: Cover Rate (Max, Median, Min) in ZDT1.

From Figure. 8 to Figure. 10, we can infer that the proposed mechanism can obtain solutions with a higher degree of diversity than those of NSGA-II. In addition, though

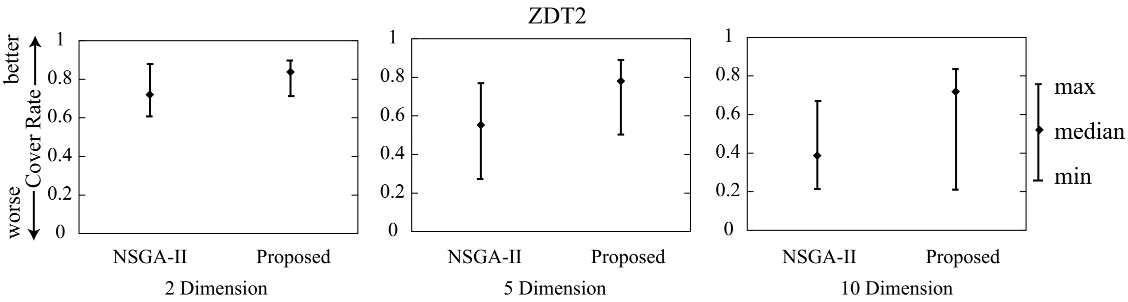


Figure 9: Cover Rate (Max, Median, Min) in ZDT2.

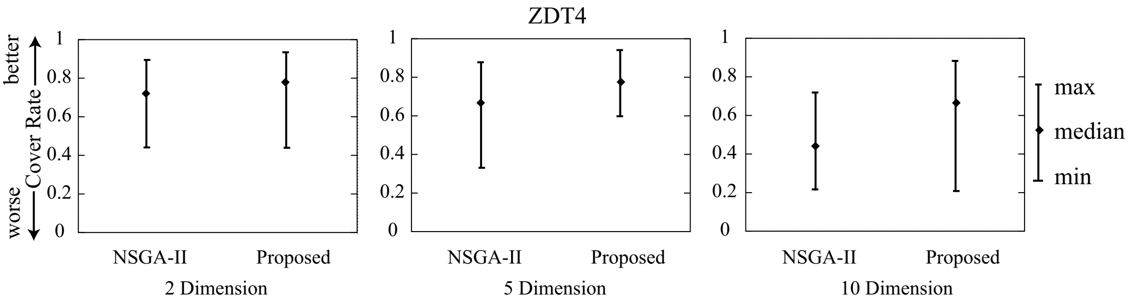


Figure 10: Cover Rate (Max, Median, Min) in ZDT4.

median quality regarding diversity is lower in higher dimensions with the same number of evaluations, the proposed method provides a high degree of diversity constantly in even such severe situations.

4.3 Verification of the Effects of Iterating the Proposed Mechanism

From the previous experiments, we found that the proposed mechanism can improve the diversity of solutions. The preservation of the high degree of diversity is one of the most important matters in the quality of Pareto-optimal solutions. This is desired not only in the last result of the MOGA search, but also during the search. We assume that the high diversity during the search is not preserved. In such cases, if the search is stopped short in a problem requiring a large calculation cost, the derived solutions would have only low degree of diversity. On the other hand, if high diversity is preserved, the solutions with high diversity can be obtained. To enable such diversity maintenance search, in this paper, the new diversity maintenance method with iteration of the proposed mechanism is proposed.

In this experiment, we examined the variance of Cover Rate in every generation. Three test problems are tested with three patterns of design variables. The parameters in this experiment are shown in Table. 4.

Table 4: Parameter Settings in the Verification of the Effects of Iterating the Proposed Mechanism

Problem	ZDT1	ZDT2	ZDT4
Population size		10	20
Generation		100	200
Dimension			10
Number of application of the proposed mechanism		6	10

The investigation of the number of the applications of the proposed mechanism is in the future works. The results of this experiment in Cover Rate are shown in Figure. 11 to

Figure. 13, and GD Result is shown in Table. 5 to confirm the convergence of the solutions to the Pareto front. In this comparison, the number of evaluation calls are set to the same in both methods.

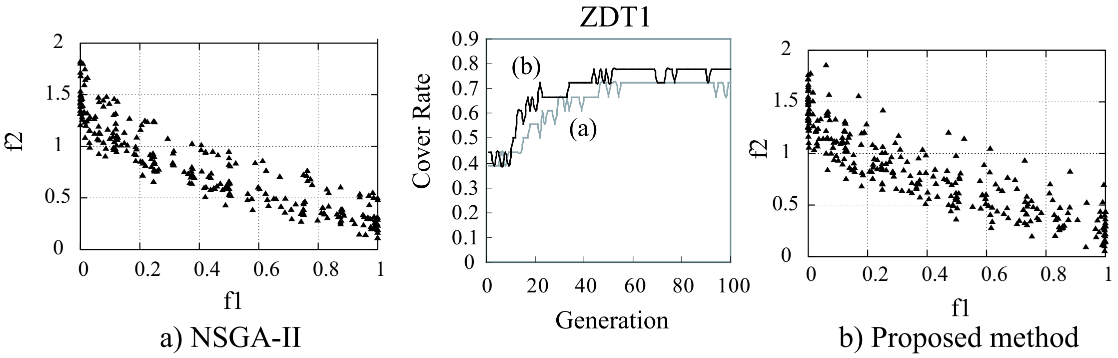


Figure 11: Variance of Cover Rate During the Search (ZDT1).

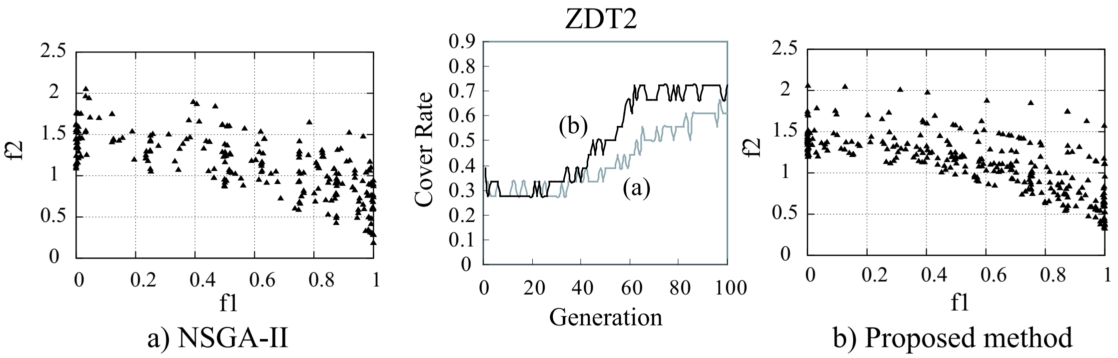


Figure 12: Variance of Cover Rate During the Search (ZDT2).

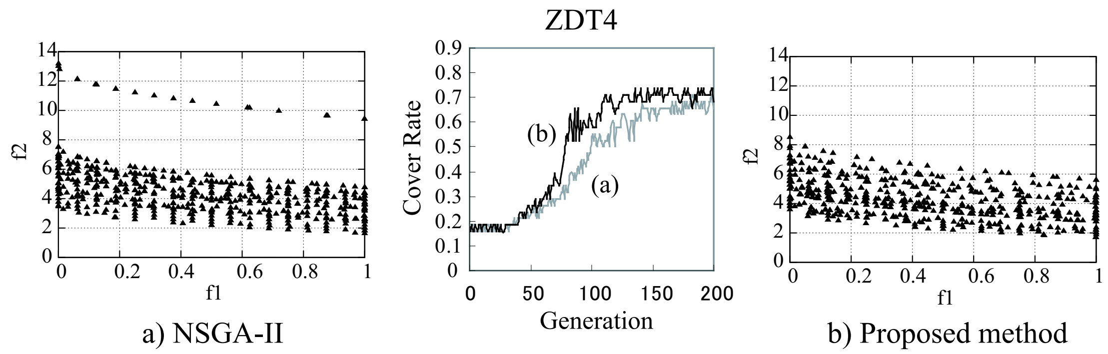


Figure 13: Variance of Cover Rate During the Search (ZDT4).

Table 5: GD Results

Problem		ZDT1	ZDT2	ZDT4
NSGA-II	mean	0.209384841	0.347565320	3.532395769
	std dev	0.105594909	0.128004631	1.659361456
Proposed Method	mean	0.237733100	0.362001831	3.060460718
	std dev	0.130098275	0.214421932	1.312268380

From Figure. 11 to Figure. 13, side figures show the plots of all non-dominated solutions of NSGA-II and the proposed method in 30 trials, and center figure shows the variance of Cover Rate. The results from these figures indicate that the search with the proposed method can preserve a high degree of diversity during the search. And from Figure. 5, the accuracy of the solutions which is obtained by the proposed method is comparable to the conventional one. From these experimental results, we can conclude that the proposed mechanism can improve the diversity of the solutions even in high dimensional search. Moreover, the proposed method with iteration can preserve a high degree of diversity during the search without deterioration of the search performance regarding accuracy.

5 Conclusions

In this paper, a new diversity restoration mechanism was proposed. This mechanism is composed of the four parts, MOGA search, clustering, training ANN, and relocation. MOGA search is a conventional search with MOGA, such as NSGA-II. In this search, many types of MOGA methods can be applied to this mechanism. Clustering is the procedure to select the solutions which exist close to each other in both the objective and the design space. In training ANN, an approximation function is trained by forward relation of objective function, and the solutions obtained by the previous two processes are used as the training data set. In relocation process, the solutions are relocated uniformly in the objective space. To obtain the solutions with uniform distribution, the design variable values of the target solutions, which are distributed uniformly in the objective space, must be determined by inverse analysis. For this purpose, network inversion is performed by the fixed approximation function created from training ANN. Thus, this proposed mechanism can restore a high degree of diversity of solutions with uniform distribution. In addition, we proposed the new diversity maintenance method with iteration of its mechanism. It maintains the solution diversity during the search, even when searching with a small number of solutions. Here, the proposed mechanism was introduced to NSGA-II, and its effectiveness was examined on mathematical test functions. From the results of the numerical experiments, we can conclude that the proposed mechanism can improve the diversity of solutions, even with the small population search on the high dimensional problems. Also, the proposed method with iteration of the proposed mechanism can preserve a high degree of diversity of the solutions during the search in some mathematical test functions.

References

[1] K.Chiba, S.Obayashi. High-fidelity multidisciplinary design optimization of aerostructural wing shape for regional jet. In *23rd Applied Aerodynamics Conference*, 2005.

[2] J.Kamiura S.Watanabe H.Hiroyasu T.Hiroyasu, M.Miki. Multi-objective optimization of diesel engine emissions and fuel economy using genetic algorithms and phenomenological model. In *SAE 2002 Powertrain and Fluid Systems Conference*.

[3] Kalyanmoy Deb, Pawan Zope, and Abhishek Jain. Distributed computing of pareto-optimal solutions with evolutionary algorithms. In *EMO*, pp. 534–549, 2003.

[4] David A. van Veldhuizen, Jesse B. Zydallis, and Gary B. Lamont. Considerations in engineering parallel multiobjective evolutionary algorithms. *IEEE Trans. Evolutionary Computation*, Vol. 7, No. 2, pp. 144–173, 2003.

[5] F. Streichert, H. Ulmer, and A. Zell. Parallelization of multi-objective evolutionary algorithms using clustering algorithms. In Carlos A. Coello Coello, Arturo Hernandez Aguirre, and Eckart Ziztler, editors, *Conference on Evolutionary Multi-Criterion Optimization (EMO 2005)*, Vol. 3410 of *LNCS*, pp. 92–107, Guanajuato, Mexico, 9-11 March 2005.

[6] F. de Toro Negro, J. Ortega, E. Ros, S. Mota, B. Paechter, and J. M. Martín. Psfga: parallel processing and evolutionary computation for multiobjective optimisation. *Parallel Comput.*, Vol. 30, No. 5-6, pp. 721–739, 2004.

[7] Gary B. Lamont Carlos A. Coello and David A. Van Veldhuizen. *Evolutionary Algorithms for Solving Multi-Objective Problems.* Kluwer Academic Publishers, Norwell, MA, USA, 2002.

[8] Jürgen Branke, Hartmut Schmeck, Kalyanmoy Deb, and M. Reddy. Parallelizing multi-objective evolutionary algorithms: cone separation. In *Congress on Evolutionary Computation*, Vol. 2, pp. 1952–1957. IEEE, JUN 2004.

[9] A.Todoroki. Response surface method,
http://florida.mes.titech.ac.jp/responsesurface.pdf.

[10] Julio l, hierarchical variable selection in polynomial regression models.
http://www.jstor.org/view/00031305/di020596/02p0164o/0.

[11] C.Bishop. Neural networks for pattern recognition. *Oxford University Press*, 1997.

[12] J.Henseler. Back propagation. *Lect Notes Comput Sci*, Vol. 931, pp. 37–66, 1999.

[13] HAYMAN S. The mcculloch-pitts model. *IEEE Int Conf Neural Netw Proc*, Vol. 6, pp. 4438–4439, 1999.

[14] J. et al. Sacks. *Statistical Science*, Vol. 4, pp. pp.409–435, 1989.

[15] Y.Kounoe and K.Fujita. Third-order polynominal response surface with optimal selection of interaction terms for global approximation. In *Design and Systems*, Vol. 15, pp. 31–34.

[16] K.Kobayashi and M.Miki T.Hiroyasu. Mechanism of multi-objective genetic algorithm for maintaining the solution diversity using neural network. In *Lecture Notes in Computer Science*, Vol. 4403, pp. 216–226. IEEE, 2007.

[17] A.Linden and J.Kindermann. Inversion of multilayer nets, proc. int. joing conf. neural networks. pp. pp 425–430, 1993.

[18] T.Sirai, M.Arakawa, and H.Nakayama. Application of rbf network in approximate optimization in satisfcing method. In *Design and Systems*, pp. Vol14th page85–86, 2004.

[19] A.G.Cunha and A.Vieira. A hybrid multi-objective evolutionary algorithm using an inverse neural network. 2004.

[20] Ian Griffin Salem F. Adra, Ahmed I. Hamody and Peter J. Fleming. A hybrid multi-objective evolutionary algorithm using an inverse neural network for aircraft control system designa. *2005 IEEE Congress on Evolutionary Computation (CEC'2005)*, Vol. 1, pp. 1–8, 2005.

[21] K. Deb and T. Meyarivan. Constrained test problems for multi-objective evolutionary optimization. In *KanGAL report 200005, Indian Institute of Technology, Kanpur, India*, 2000.