Efficient Search Techniques Using Adaptive Discretization of Design Variables on Real-Coded Evolutionary Computations

Toshiki Kondo
Tokyo University of Science
Katsushika-ku, Tokyo
4416611@ed.tus.ac.jp

Tomoaki Tatsukawa
Tokyo University of Science
Katsushika-ku, Tokyo
tatsukawa@rs.tus.ac.jp

ABSTRACT

In this paper, we evaluate the effects of adaptive discretization of design variables in real-coded evolutionary computations (RCECs). While the appropriate granularity of design variables can improve convergence in RCECs, it is difficult to decide the appropriate one in advance in most of the practical optimization problems. Besides, when the granularity is too coarse, the diversity may be lost. To address these difficulties, we propose two adaptive discretization techniques that discretize each design variable using granularity determined according to the indicator of solution distribution state in design space. In this study, standard deviation(SD) or estimated probability density function(ePDF) is used as an indicator for determining granularities of design variables. We use NSGA-II as an RCEC and thirteen benchmark problems including engineering problems. The generational distance (GD) and inverted generational distance (IGD) metrics are used for investigating the performance of convergence and diversity, respectively. To make sure the statistical difference of results, the Wilcoxon rank-sum test and Welch's t-test are applied in each problem. The results of experiments show that both of the proposed methods can automatically improve convergence in many problems. In addition, it is confirmed that the diversity is also maintained.

CCS CONCEPTS

• **Mathematics of computing → Evolutionary algorithms**;

KEYWORDS

Evolutionary Computations, Multi-objective optimization

ACM Reference Format:
Toshiki Kondo and Tomoaki Tatsukawa. 2018. Efficient Search Techniques Using Adaptive Discretization of Design Variables on Real-Coded Evolutionary Computations. In *Proceedings of the Genetic and Evolutionary Computation Conference 2018 (GECCO '18),* Jennifer B. Sartor, Theo D'Hondt, and Wolfgang De Meuter (Eds.). ACM, New York, NY, USA, Article 4, 8 pages. https://doi.org/10.1145/3205455.3205636

ACM ISBN 978-1-4503-5618-3/18/07…$15.00
https://doi.org/10.1145/3205455.3205636

1 INTRODUCTION

Many studies have been carried out on multiobjective optimization with real-coded evolutionary computations (RCECs). The effectiveness of RCECs has been demonstrated in real-world optimization problems[13]. However, since the use of many computing resources over long periods of time are often required to evaluate objective functions and constraints in actual problems, it takes a long time to conduct optimization by RCECs. Therefore, it is necessary to search efficiently with fewer evaluations for accelerating optimization.

Unlike binary-coded evolutionary computations (BCECs), RCECs have design variables consisting of real value instead of binary value. Generally, in the case of using RCECs, the design variables are set real value close to the continuous value, so it is not so often to consider discretization of design variables. On the other hand, in real-world problems, the design variables are often set discrete value from the viewpoint of feasibility. However, there are not enough discussions on the effects of discretization of design variables in RCECs.

In BCECs, it was reported that the short bit length accelerates the convergence of solutions[7] because it discretizes and shrinks the design space. By utilizing this characteristic, a technique of dynamically changing the bit length in the process of optimization have been proposed[7, 9].

Conversely, there are few studies about methods that dynamically discretizes the design variables on RCECs. The effect of discretization of the design variables on RCECs has been reported by [10]. From this paper, it is found that coarse discretization of the design variables can improve better convergence than fine discretization. On the other hand, it is also reported that too coarse discretization worsens the diversity of solutions in some problems. These results imply that appropriate discretization of each design variable according to the problem can improve convergence while maintaining diversity. However, it is difficult to discretize each design variable properly according to the problem before optimization.

In this study, to address this difficulty, we propose the new techniques which adaptively discretize the design variables according to the distribution in design space and investigate the effects of proposed methods. Here, we use Non-dominated Sorting Genetic Algorithm-II (NSGA-II)[1], which is one of the most popular multiobjective evolutionary algorithm, and a set of benchmark problems, including some engineering problems.

2 RELATED WORKS

A brief review of the discretization in evolutionary multiobjective optimization algorithms is presented in this section.

2.1 Discretization in Objective Space

In recent years, studies about discretization of an objective space have been drawing considerable. The idea of epsilon dominance, which was proposed by Laumanns[11], is one of the most popular discretization techniques of the objective space. This technique makes all points within a small distance (the epsilon distance) from a set of Pareto-optimal points dominant. Epsilon dominance can reduce the number of Pareto-optimal solutions so that the selection pressure is increased over the Pareto front [12].

Ishibuchi, et.al.[6] examined the effects of discrete objective functions with various granularities. They focused on granularity differences between discrete objective functions in combinatorial optimization problems and showed that a distinct objective function with coarse granularity (low resolution) slowed down the search ability of evolutionary multiobjective optimization algorithms along that objective.

2.2 Discretization in Design Space

Studies related to discrete design variables have been conducted mainly in binary-coded evolutionary computations.

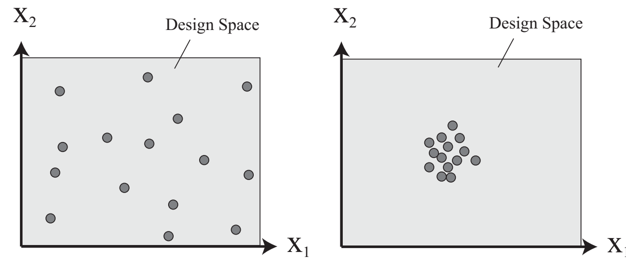
Jaimes, et. al.[7] proposed the island-type parallel multiobjective evolutionary algorithm (pMOEA) in which each island had a different resolution (different length of binary strings). In this algorithm, the resolution in each island starts from low and gradually increases in each generation so that rapid convergence can be achieved while maintaining diversity.

Kim, et. al.[9] proposed a variable chromosome length evolutionary computation for topology optimization problems. This algorithm also starts with low-resolution (a short chromosome) and increases the number of bits after finding an optimal solution at the current resolution.

These dynamic resolution techniques[7, 9], which are based on variable resolution, show better search performance than methods with the fixed resolutions.

In RCECs, Kondo, et. al. [10] investigated the effect of discretization of the design variables on evolution by using NSGA-II and reported that the coarse discretization of design variables tends to accelerate the convergence. Especially, in DTLZ problems[3], coarse discretization of design variables can suppress the occurrence of dominance resistant solutions (DRSs)[5] that prevents efficient search. However, in DTLZ4, which has strong nonlinearity, although coarse discretization of design variables can improve the convergence, it worsens the diversity. From these results, it is implied that appropriate discretization of each design variable according to the problem can improve convergence while maintaining diversity.

However, it is difficult to discretize each design variable accurately according to the problem before optimization.



(a) Sparse Distribution (b) Dense Distribution

Figure 1: Schematic diagrams of solution distribution in two dimensional design-variable space

3 ADAPTIVE DISCRETIZATION METHOD

In this study, we consider changing the granularity of each design variable according to the distribution of solutions in design space. Figure 1 shows the schematic diagrams of the solution distribution in two-dimensional design space. Generally, in an early generation, the solution distribution in the design space is scattered throughout the design space, which is a sparse distribution (Fig. 1(a)). As the generation progresses, since many solutions move around the optimal solution, the solution distribution becomes a dense distribution (Fig. 1(b)).

Here, when the solution distribution is sparse, it is assumed that a design variable has not converged to an optimal value yet. In this case, coarse discretization is performed to accelerate convergence. Conversely, when the solution distribution is dense, it is assumed that there is an optimal solution near the solution distribution. In this case, fine discretization is performed to search with high accuracy. It is expected that the convergence can improve while maintaining diversity, by controlling in this way. The important feature is that this method can be simply applied to most of RCECs because it is not necessary to modify the procedure of RCECs.

In this study, NSGA-II is used as an RCEC. Standard deviation (SD) or estimated probability density function (ePDF) is used as an indicator of the state of the solution distribution to control the granularity of each design variable.

Algorithm 1 shows the algorithm of NSGA-II with the adaptive discretization method.

After making initial population (P_0), each design variable is discretized according to an indicator that represents the distribution state of solutions. The indicator will be described later. After that, objective functions and constraints are evaluated. Then offsprings (Q_0) are generated from parent pairs by crossover and mutation. The indicator of the solution distribution is evaluated by offsprings again. Each design variable of offsprings is discretized according to the indicator. Finally, parent population is updated by non-dominated sorting and crowding distance sorting.

In this study, when step2 and step7 are executed, each design variable is projected on normalized space [0,1], and their decimal-place number is controlled in this space. Controlling the decimal-place number of design variables corresponds to discretization. Also, the decimal-place number corresponds to a granularity used in discretization. When the design variable is discretized using the

Algorithm 1 NSGA-II Algorithm with Proposed Method

Initialize parent population (P_0)
Discretize each design variable according to solution distribution of P_0 (Proposed method)
Evaluate P_0
for each $t = 0, \cdots, t_{\max}$ **do**
  Make parent pairs from the population (P_t) by the tournament selection
  Conduct a crossover (SBX) and a mutation (polynomial mutation) to generate an offspring population (Q_t)
  Discretize each design variable according to solution distribution of Q_t (Proposed method)
  Evaluate Q_t
  Combine P_t and Q_t into R_t
  Make a new parent population(P_{t+1}) from R_t (non-dominated sorting and crowding distance sorting)
end for

number of decimal place d, the division number of design space is 10^d. The number of decimal place d is determined in different ways according to each indicator (SD or ePDF).

3.1 SD-based Indicator

In this study, standard deviation (SD) is adopted as an indicator that represents the distribution state solutions. When the sparse distribution is given, SD indicates high value. On the other hand, when the dense distribution is given, SD indicates low value. Utilizing this property, the decimal-place number of each design variable is decided by SD. The number of decimal place d is decided by SD as follows:

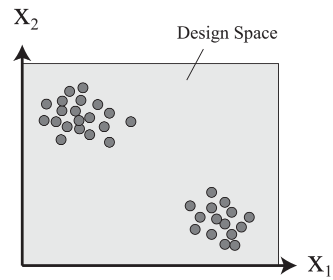
$$d_i = round\left(\left(1 - \frac{\sigma_i}{\sigma_{\max}}\right)(d_{\max} - d_{\min}) + d_{\min}\right) \quad (1)$$

d_i indicates the number of decimal place controlling design variable x_i. $d_{\max}$ and $d_{\min}$ are positive integers indicating upper and lower bounds parameters, respectively. σ_i indicates the standard deviation of design variable x_i. $\sigma_{\max}$ is the standard deviation of the sparsest distribution. In this study, uniform distribution is assumed as the sparsest distribution. Therefore, $\sigma_{\max}$ is the standard deviation of uniform distribution in normalized space [0,1] and becomes $1/\sqrt{12}$.

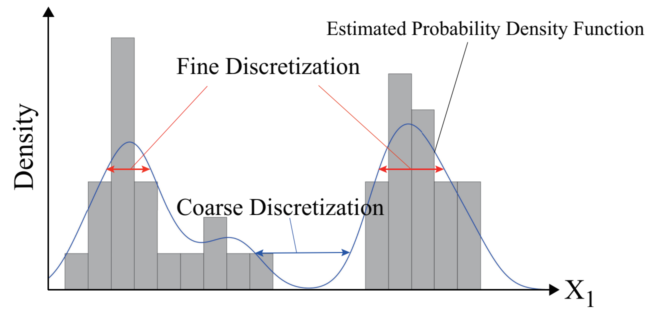
From Eq. (1), when the distribution in design variable x_i is sparse, σ_i is close to $\sigma_{\max}$, then d_i becomes close to $d_{\min}$. As a result, the design variable x_i is coarsely discretized. On the other hand, when the distribution in design variable x_i is dense, σ_i is close to 0, then d_i becomes close to $d_{\max}$. As a result, the design variable x_i is finely discretized.

3.2 ePDF-based Indicator

Since the SD-based indicator determines the granularity of the entire design space, there is a possibility that it can not work well with the case where there are multiple optimal values of a design variable. Therefore, in this study, estimated probability density function (ePDF) is adopted as another indicator of solutions-distribution state in this study. By using ePDF, features of solution distribution



(a) Multiple density distribution



(b) Histogram and ePDF in x_1 of (a).

Figure 2: Estimated Probability Density Function

can be accurately grasped more than SD. Figure 2(a) indicates the case of multiple dense distributions. In this case, the SD-based indicator may not work well because standard deviation cannot detect multiple dense distributions. On the other hand, ePDF can detect multiple dense distributions. Figure 2(b) shows the frequency histogram of x_1 in Fig. 2(a), and ePDF of one. In Fig. 2(b), blue line indicates ePDF. As shown in Fig. 2(b), there are two peaks in ePDF. Since ePDF can detect the detailed features of solution distribution, we consider that the design variables are partially discretized according to ePDF value for each region. When the sparse distribution is given, ePDF indicates low value. On the other hand, when the dense distribution is given, ePDF indicates high value. Utilizing this property, like Fig. 2(b), when the high ePDF value is given, the design variables are finely discretized. Conversely, when the low ePDF value is given, the design variables are coarsely discretized.

The number of decimal place d used in discretization is decided using ePDF as follows:

$$d_{i,j} = round\left(\left(1 - \frac{\hat{f}(x_{i,j})}{\max\limits_{x \in [0,1]} \hat{f}(x)}\right)(d_{\max} - d_{\min}) + d_{\min}\right) \quad (2)$$

$d_{i,j}$ indicates the number of decimal place controlling design variable x_i of an individual j. $d_{\max}$ and $d_{\min}$ are positive integers indicating upper and lower bounds parameters, respectively. $\hat{f}(x_{i,j})$ indicates estimated probability density value in design variable x_i of an individual j. $\max_{x \in [0,1]} \hat{f}(x)$ is the maximum estimated probability density value in normalized space [0,1].

In this study, kernel density estimation (KDE) is used to estimate the probability density function.

4 EXPERIMENTAL CONDITIONS

4.1 Benchmark Problems

In this study, 13 benchmark problems are considered: four DTLZ problems[3], two WFG problems[4], two CEC2009 problems[14], three CDTLZ problems[8], and two engineering problems[2, 8], which are described below. The properties of the adopted benchmark problems are summarized in Table 1, where each problem is classified under four characteristics[4]: separability, modality, bias, and geometry. These benchmark problems are representative problems selected from the above four characteristics.

Separability indicates whether a correlation relationship exists among the parameters. Separable problems are characterized by parameter independence, whereas non-separable problems are characterized by parameter dependencies, and are more difficult to analyze than separable problems. Modality describes the characteristics of fitness landscapes. Multimodal problems have many local optimal points, whereas unimodal problems have only one global optimal point. Bias also describes the fitness landscape and indicates the bias of distribution in objective space. Geometry refers to the shape of the Pareto-front surface.

As engineering problems, the car side impact problem[8] and the welded beam problem[2] are employed.

The car side impact problem has three objective functions; (1) minimizing of the total weight of a car, (2) minimizing of the pubic force experienced by a passenger, and (3) minimizing of the average velocity of the V-Pillar responsible for withstanding the impact load. This problem has ten constraints and seven real-parameter variables.

The welded beam design problem has two objective functions; (1) minimizing the cost of fabrication, and (2) minimizing of the end defection of the welded beam. This problem has four constraints and four real-parameter variables.

4.2 Computational Conditions

Non-dominated Sorting Genetic Algorithm-II (NSGA-II) is used in this study. The common parameter settings are shown in Table 2, where η_c and η_m are indices of simulated binary crossover (SBX) and polynomial mutation, respectively. The population size, the generation number, the number of objective functions and the number of design variables are different in each benchmark problem. All of these parameters are shown in Table 1. k and l, which are parameters used in WFGs, are set 6 and 4, respectively. The optimization performance is compared among NSGA-II with the SD-based indicator, the ePDF-based indicator and no discretization(ND). The parameters setting of SD and ePDF are shown in Table 3 and Table 4, respectively.

5 EXPERIMENTAL RESULTS

In this study, generational distance (GD) and inverted generational distance (IGD) are adopted to evaluate convergence and diversity, respectively. The average GD and IGD are calculated by using feasible non-dominated solutions at each generation. In the engineering problems, approximate Pareto-optimal solutions are created by merging non-dominated solutions of all trials. Furthermore, the Wilcoxon rank-sum test is applied to make sure statistically differences of GD trends between each indicator (SD or ePDF) and no discretization (ND). To confirm statistically differences of averaged final IGD between each indicator (SD or ePDF) and no discretization (ND), the Welch's t-test is applied in each problem. In each test, significance level is set 5%.

5.1 Convergence

Table 5 shows the averaged GD for using each indicator (SD or ePDF) and no discretization (ND) for each of the benchmark problem using NSGA-II. The averaged GD is evaluated at the 10th, half, and final generation because the convergence rate is important in MOEA as well as the final converged value. In Table 5, a checkmark (✓) indicates that there is significant difference between the indicator and ND in the problem, on the other hand, a hyphen (-) indicates that there is not significant difference among them.

As shown in Table 5, in final generation, SD-based indicator can improve better convergence than ND with 85% of benchmark problems. Similarly, in final generation, ePDF-based indicator can also improve better convergence than ND with 77% of benchmark problems. In addition, in 10 generation, SD-based indicator yields lower GD than ND with 85% of benchmark problems. ePDF-based indicator yields lower GD than ND with 62% of benchmark problems. From these results, it is expected that adaptive discretization can improve convergence performance of RCECs in various problems.

To confirm statistically differences of GD trends between each indicator (SD or ePDF) and ND, the Wilcoxon rank-sum test is applied in each problem. As a result, both adaptive discretization methods are statistically superior than ND in many problems.

On the other hand, in WFG8 and CF7, SD can not improve convergence. The common feature of these problems is non-separable. In the non-separable problem, there is a possibility that there are multiple dense distribution, so SD may worsen convergence. However, in WFG8 and CF7, because ePDF can improve convergence, it is implied that partially discretization by ePDF-based indicator can improve convergence in complex problems with non-separable feature.

5.2 Diversity

Table 6 shows the averaged final IGD for using each indicator (SD or ePDF) and no discretization (ND) for each of the benchmark problem using NSGA-II to confirm final diversity of solutions. In Table 6, a checkmark (✓) indicates that there is significant difference between the indicator and ND in the problem, on the other hand, hyphen (-) indicates that there is not significant difference among them.

As shown in Table 6, each adaptive discretization method can yield lower IGD than ND in final generation with about half problems. In the other half problems, although ND yields lower IGD, each method can yields IGD that is same in the order of magnitude as that of ND.

To confirm statistically differences of the final averaged IGD between each indicator (SD or ePDF) and ND, the Welch's t-test is applied in each problem. As a result, in the problems, where ND

Table 1: The benchmark problems used in this study.

Problem	Obj.	Var.	Con.	Separability	Modality	Bias	Geometry
DTLZ2	M	N	-	separable	uni	-	concave
DTLZ3	M	N	-	separable	multi	-	concave
DTLZ4	M	N	-	separable	uni	✓	concave
DTLZ7	M	N	-	separable	uni	-	concave, disconnected
WFG2	M	N	-	non-separable	multi	-	convex, disconnected
WFG8	M	N	-	non-separable	uni	✓	concave
UF9	3	N	-	non-separable	multi	-	linear, disconnected
CF7	2	N	2	non-separable	multi	-	convex
C1DTLZ3	M	N	1	separable	multi	-	concave
C2DTLZ2 convex	M	N	1	separable	uni	-	convex, disconnected
C3DTLZ1	M	N	M	separable	multi	-	convex (feasible surface is PF)
Car Side Impact	3	7	10	uncertain			
Welded Beam Design	2	4	4	uncertain			

Obj. is the number of objectives; Var. is the number of variables; Con. is the number of constraints.

M is the user predefined number of objectives; N is the user predefined number of variables.

- indicates no value or no characteristic; ✓ indicates the problem has the characteristic.

The Car Side Impact and Welded Beam Design problems have uncertain characteristics because of engineering problems.

Table 2: Values of the common parameters used in the experimental study.

parameter	value
Crossover rate	1.0
Mutation rate	1/ (# of variables)
η_c	30
η_m	20
Trial	10

Table 3: Values of the using SD parameters used in the experimental study.

parameter	value
d_{min}	2
d_{max}	8
σ_{max}	$1/\sqrt{12}$

$1/\sqrt{12}$ is standard deviation of uniform distribution.

Table 4: Values of the using estimated PDF parameters used in the experimental study.

parameter	value
d_{min}	2
d_{max}	8
h	0.05
Kernel Function	Standard Normal Distribution

h is bandwidth parameter in KDE.

Kernel Function is a function used by KDE.

yields lower IGD than each indicator, there is not statistically significant difference except for WFG2. Additionally, in objective space, there is no big distribution difference between final non-dominated solutions obtained by adaptive discretization and them obtained by ND. Therefore, it is confirmed that the adaptive discretization methods can maintain diversity.

5.3 Further Investigations of DTLZ3

To further investigate the effects of adaptive discretization on evolution, we focus on DTLZ3 because of the most considerable difference between adaptive discretizations and ND in convergence.

Figure 3 shows the distribution of non-dominated solutions obtained by adaptive discretization using each indicator (SD or ePDF) and ND. As shown in Fig. 3, adaptive discretization methods using either indicator can obtain better non-dominated solutions than ND. Interestingly, both of adaptive discretization methods have less number of dominance resistant solutions (DRSs) than ND. In DTLZ3, it is considered that because adaptive discretization tends to avoid the formation of DRSs, the performance of convergence can be improved better than ND.

Figure 4 shows the transition of division number of design variables used in discretization by the SD-based indicator in DTLZ3. The vertical line indicates the number of division, and the horizontal one shows the generation. The higher the number of division is, the finer the design variable is discretized. Position variables (x_1 and x_2) mean the variables which determine the position of the objective space and distance variables (from x_3 to x_{38}) mean the variables which determine the distance from the origin of the objective space in DTLZ3. As shown in Fig. 4, it is found that most of the design variables are changing from coarse discretization to fine discretization. On the other hand, the position variables are coarsely discretized from the begin to the end. This result means that fine discretization is not required for position variables in DTLZ3. It is considered that adaptive discretization based on the SD indicator is possible to discriminate design variables that require fine discretization or does not.

Figure 5(a) and (b) indicate the transition of ePDF value in the design variable space of position variable x_1 and distance variable

Table 5: Averaged generational distances (GD) at the 10th, half, and final generation using NSGA-II with each indicator (SD or ePDF) and no discretization (ND). Best cases are in bold font. Significance level of the Wilcoxon rank-sum test is set 5 %.

Problem	MG	PS	Obj.	Var.	Gen		SD		ePDF	ND
DTLZ2	100	100	3	38	10	✓	**1.628**	-	1.741	1.759
					50		**1.671×10^{-1}**		2.179×10^{-1}	2.565×10^{-1}
					100		**6.114×10^{-2}**		9.541×10^{-2}	1.011×10^{-1}
DTLZ3	200	500	3	38	10	✓	**2.175×10^{3}**	✓	2.484×10^{3}	2.508×10^{3}
					100		**1.508×10^{2}**		2.792×10^{2}	4.039×10^{2}
					200		**2.661×10**		4.349×10	6.669×10
DTLZ4	100	300	3	38	10	✓	**9.789×10^{-1}**	-	1.292	1.223
					50		**5.313×10^{-2}**		8.692×10^{-2}	7.147×10^{-2}
					100		**2.060×10^{-2}**		3.269×10^{-2}	2.522×10^{-2}
DTLZ7	100	100	3	38	10	✓	**7.060**	-	8.295	8.145
					50		**6.427×10^{-1}**		1.276	1.306
					100		**1.491×10^{-1}**		5.271×10^{-1}	5.442×10^{-1}
WFG2	100	100	3	10	10	✓	2.350×10^{-1}	✓	**2.232×10^{-1}**	2.428×10^{-1}
					50		**5.586×10^{-2}**		6.082×10^{-2}	7.141×10^{-2}
					100		3.821×10^{-2}		**3.748×10^{-2}**	4.220×10^{-2}
WFG8	100	100	3	10	10	-	7.337×10^{-1}	-	7.214×10^{-1}	**7.015×10^{-1}**
					50		2.934×10^{-1}		**2.882×10^{-1}**	2.939×10^{-1}
					100		2.198×10^{-1}		**1.977×10^{-1}**	2.002×10^{-1}
UF9	200	300	3	20	10	-	**2.752**	-	2.810	2.771
					100		**7.003×10^{-1}**		7.153×10^{-1}	7.343×10^{-1}
					200		**5.068×10^{-1}**		5.427×10^{-1}	5.320×10^{-1}
CF7	200	200	2	20	10	✓	**2.628×10**	-	2.686×10	2.706×10
					100		1.938		**1.535**	1.577
					200		9.725×10^{-1}		**5.473×10^{-1}**	7.615×10^{-1}
C1DTLZ3	300	100	3	38	10	✓	**2.573×10^{3}**	✓	2.757×10^{3}	2.871×10^{3}
					150		**1.361×10^{2}**		2.214×10^{2}	3.267×10^{2}
					300		**7.301×10**		9.030×10	1.243×10^{2}
C2DTLZ2 convex	300	100	3	38	10	✓	**3.948×10**	✓	5.962×10	7.017×10
					150		7.964×10^{-2}		**7.492×10^{-2}**	1.114×10^{-1}
					300		5.569×10^{-2}		**5.244×10^{-2}**	9.449×10^{-2}
C3DTLZ1	300	100	3	38	10	✓	**1.140×10^{3}**	✓	1.168×10^{3}	1.152×10^{3}
					150		**1.445×10^{2}**		2.307×10^{2}	3.801×10^{2}
					300		**6.083×10**		9.469×10	2.587×10^{2}
Car Side Impact	100	300	3	7	10	-	1.089×10^{-1}	-	**1.008×10^{-1}**	1.075×10^{-1}
					50		**3.751×10^{-2}**		3.765×10^{-2}	3.763×10^{-2}
					100		**3.109×10^{-2}**		3.194×10^{-2}	3.167×10^{-2}
Welded Beam Design	100	300	2	4	10	✓	**2.762**	✓	2.954	8.898
					50		**1.276×10^{-2}**		1.884×10^{-2}	4.047×10^{-2}
					100		4.390×10^{-3}		**2.665×10^{-3}**	8.827×10^{-3}

MG is the number of generation set; PS is the number of population size set.
Obj. is the number of objectives; Var. is the number of variables.
Gen indicates the generation that results were computed on.
✓ indicates that there is significant difference between using each indicator and ND in each problem.
- indicates that there is not significant difference between using each indicator and ND in each problem.

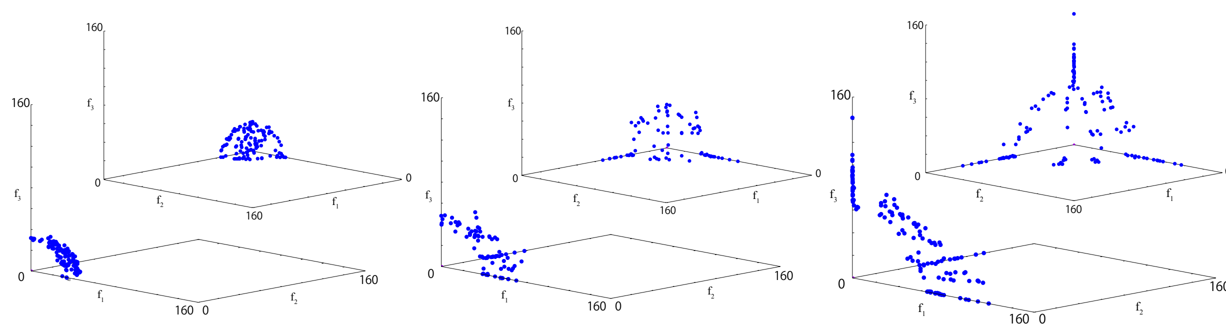
x_{10} in DTLZ3, respectively. The vertical line indicates the probability density value, and the horizontal one indicates the design variable space. The colored lines show ePDF of design variables in each generation. The higher the probability density value is, the finer the design variable is discretized.

As shown in Fig. 5(a), it is found that as evolution comes to an end, the ePDF value around 0 becomes high. Originally, the ePDF value of the position variable should be flat. However, since solutions that position variable is distributed around 0 tend to become non-dominated solutions easily and to survive next generation, it is considered that the ePDF value around 0 becomes high. Accurate search around 0 may induce the occurrence of DRSs. As a result, as shown in Fig. 3, it is considered that the distribution of ePDF becomes worse than the distribution of SD.

As shown in Fig. 5(b), it is found that as evolution comes to an end, the ePDF value around 0.5 becomes high. Because the optimal value of distance variables is 0.5 in DTLZ3 and solutions tend to be generated near 0.5 in distance variables, the ePDF value around 0.5 becomes high. Therefore, it is considered that ePDF-based indicator is working as intended. Since ePDF-based method finely searches around optimal value and coarsely searches the other regions, the convergence performance can be improved.

Table 6: Averaged inverted generational distances (IGD) at final generation using NSGA-II with each indicator (SD or ePDF) and no discretization (ND). Best cases are in bold font. Significance level of the Welch’s t-test is set 5 %.

Problem	MG	PS	Obj.	Var.		SD		ePDF	ND
DTLZ2	100	100	3	38	-	5.709×10^{-2}	✓	$\mathbf{5.549 \times 10^{-2}}$	6.213×10^{-2}
DTLZ3	200	500	3	38	✓	$\mathbf{2.270 \times 10}$	✓	3.245×10	4.607×10
DTLZ4	100	300	3	38	-	1.879×10^{-1}	-	$\mathbf{3.076 \times 10^{-2}}$	2.507×10^{-1}
DTLZ7	100	100	3	38	✓	$\mathbf{1.783 \times 10^{-1}}$	-	3.283×10^{-1}	3.037×10^{-1}
WFG2	100	100	3	10	✓	3.428×10^{-1}	✓	3.400×10^{-1}	$\mathbf{2.713 \times 10^{-1}}$
WFG8	100	100	3	10	-	2.882×10^{-1}	-	2.670×10^{-1}	$\mathbf{2.622 \times 10^{-1}}$
UF9	200	300	3	20	-	2.257×10^{-1}	-	2.146×10^{-1}	$\mathbf{1.880 \times 10^{-1}}$
CF7	200	200	2	20	-	4.401×10^{-1}	-	4.077×10^{-1}	$\mathbf{3.775 \times 10^{-1}}$
C1DTLZ3	300	100	3	38	✓	$\mathbf{6.410 \times 10}$	-	7.591×10	8.024×10
C2DTLZ2 convex	300	100	3	38	-	6.928×10^{-2}	✓	$\mathbf{6.840 \times 10^{-2}}$	7.260×10^{-2}
C3DTLZ1	300	100	3	38	✓	$\mathbf{2.551 \times 10}$	✓	2.630×10	3.508×10
Car Side Impact	100	300	3	7	-	3.509×10^{-2}	-	3.522×10^{-2}	$\mathbf{3.480 \times 10^{-2}}$
Welded Beam Design	100	300	2	4	-	9.295×10^{-2}	-	6.907×10^{-2}	$\mathbf{4.521 \times 10^{-2}}$



(a) SD-based indicator (b) ePDF-based indicator (c) No discretization

Figure 3: Distribution of non-dominated solutions applying each adaptive discretization (using SD or ePDF) and no discretization (ND)

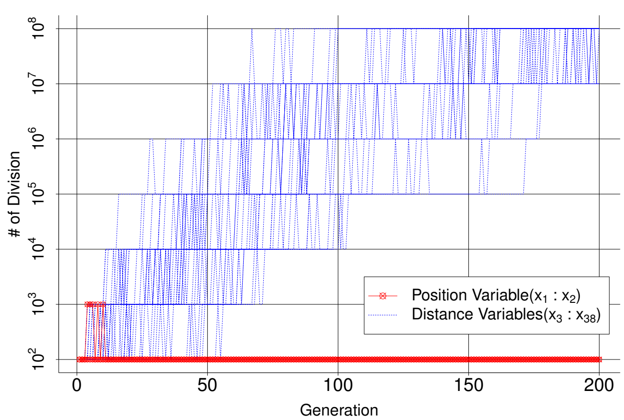


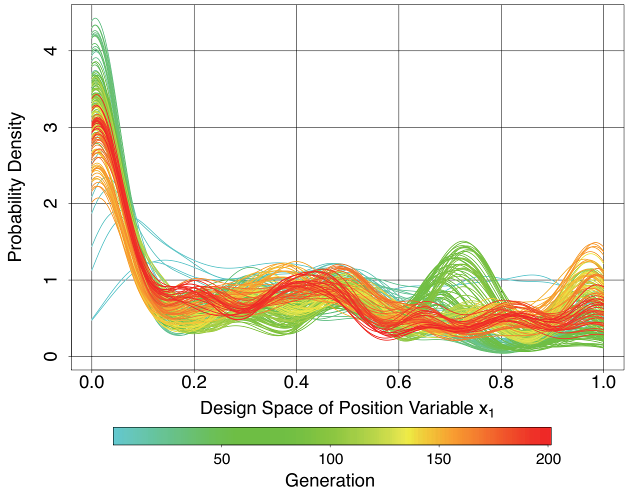
Figure 4: The transition of division number in DTLZ3.

6 CONCLUSION

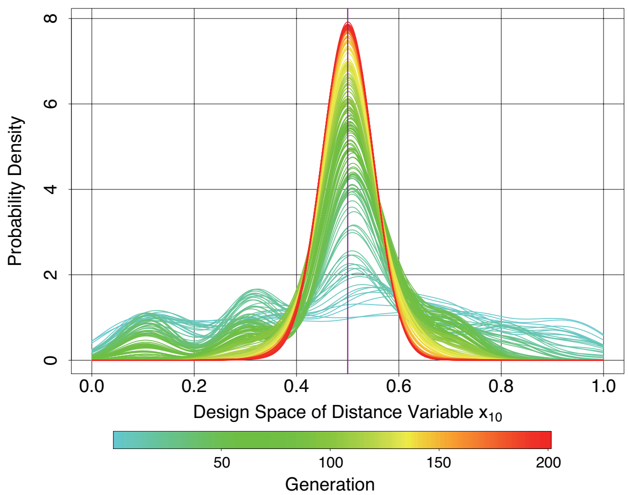
In this study, two kinds of new adaptive discretization methods of design variable on evolution for real-coded evolutionary computations (RCECs) were proposed. One is based on the standard deviation indicator, and the other is based on the estimated probability density function (ePDF) indicator. Combining the proposed methods and NSGA-II, the convergence and diversity performance of proposed methods were investigated by using 13 benchmark problems.

The results showed that adaptive discretization methods can improve better convergence performance than no discretization in many problems, especially using SD. In addition, the results also indicated that adaptive discretization methods can maintain the diversity in almost problems. From these results, it is expected that the adaptive discretization methods can improve convergence while maintaining the diversity in RCECs.

We also investigated the transition of the division number used in discretization by SD-based indicator and the transition of ePDF value used in discretization by ePDF-based indicator in DTLZ3. The transition of the division number showed that different granularity of discretization was automatically applied to position variables and distance variables, respectively. From this result, it is considered that SD is possible to discriminate design variables that require fine discretization or does not, so convergence performance can be improved. The transition of ePDF value showed that ePDF finely searches around optimal value and coarsely searches the other



(a) The transition of ePDF for x_1



(b) The transition of ePDF for x_{10}

Figure 5: The transition of ePDF in DTLZ3

regions. From these results, it is confirmed that both of the proposed adaptive discretization methods are working as intended. These methods can improve better convergence while maintaining the diversity.

As a future work, further investigation will be necessary to assess in the benchmark problems, where there are multiple optimal value of design variables, in order to investigate behavior of the adaptive discretization.

In real-world engineering optimization problems, turn-around time for optimization should be as short as possible, so adaptive discretization can be expected as an efficient search method.

7 ACKNOWLEDGMENT

This research was partially supported by RIKEN Advanced Institute for Computational Science as Priority Issue 8 (Sub-issue A) on Post-K computer development (Project ID:hp170238).

REFERENCES

[1] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. 2002. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6, 2 (April 2002), 182–197. https://doi.org/10.1109/4235.996017

[2] Kalyanmoy Deb and J Sundar. 2006. Reference point based multi-objective optimization using evolutionary algorithms. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation*. ACM, 635–642.

[3] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. 2002. Scalable multi-objective optimization test problems. In *Proceedings of IEEE Congress on Evolutionary Computation CEC '02*, Vol. 1. 825–830. https://doi.org/10.1109/CEC.2002.1007032

[4] S. Huband, P. Hingston, L. Barone, and L. While. 2006. A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation* 10, 5 (Oct. 2006), 477–506. https://doi.org/10.1109/TEVC.2005.861417

[5] K. Ikeda, H. Kita, and S. Kobayashi. 2001. Failure of Pareto-based MOEAs: does non-dominated really mean near to optimal?. In *Proceedings of IEEE Congress on Evolutionary Computation*, Vol. 2. 957–962. https://doi.org/10.1109/CEC.2001.934293

[6] Hisao Ishibuchi, Masakazu Yamane, and Yusuke Nojima. 2012. Effects of Discrete Objective Functions with Different Granularities on the Search Behavior of EMO Algorithms. In *Proceedings of the 14th Annual Conference on Genetic and Evolutionary Computation (GECCO '12)*. 481–488. https://doi.org/10.1145/2330163.2330232

[7] A. Lopez Jaimes and C. A. Coello Coello. 2005. MRMOGA: parallel evolutionary multiobjective optimization using multiple resolutions. In *Proceedings of IEEE Congress on Evolutionary Computation*, Vol. 3. 2294–2301. https://doi.org/10.1109/CEC.2005.1554980

[8] Himanshu Jain and Kalyamoy Deb. 2014. An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point Based Nondominated Sorting Approach, Part II: Handling Constraints and Extending to an Adaptive Approach. *IEEE Transactions on Evolutionary Computation* 18, 4 (Aug. 2014), 602–622. https://doi.org/10.1109/TEVC.2013.2281534

[9] I.Y. Kim and O.L. de Weck. 2005. Variable chromosome length genetic algorithm for progressive refinement in topology optimization. *Structural and Multidisciplinary Optimization* 29, 6 (June 2005), 445. https://doi.org/10.1007/s00158-004-0498-5

[10] T. Kondoh, T. Tatsukawa, A. Oyama, T. Watanabe, and K. Fujii. 2016. Effects of discrete design-variable precision on real-coded Genetic Algorithm. In *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*. 1–8. https://doi.org/10.1109/SSCI.2016.7850230

[11] Marco Laumanns, Lothar Thiele, Kalyanmoy Deb, and Eckart Zitzler. 2002. Combining Convergence and Diversity in Evolutionary Multiobjective Optimization. *Evolutionary Computation* 10, 3 (Sept. 2002), 263–282. https://doi.org/10.1162/106365602760234108

[12] BSP Mishra, Satchidanand Dehuri, and Sung-Bae Cho. 2015. Swarm Intelligence in Multiple and Many Objectives Optimization: A Survey and Topical Study on EEG Signal Analysis. In *Multi-objective Swarm Intelligence*. Springer, 27–73.

[13] A. Oyama, T. Nonomura, and K. Fujii. 2010. Data Mining of Pareto-Optimal Transonic Airfoil Shapes Using Proper Orthogonal Decomposition. *Journal of Aircraft* 47 (2010), 1756–1762.

[14] Qingfu Zhang, Aimin Zhou, Shizheng Zhao, Ponnuthurai Nagaratnam Suganthan, Wudong Liu, and Santosh Tiwari. 2008. Multiobjective optimization test instances for the CEC 2009 special session and competition. *University of Essex, Colchester, UK and Nanyang technological University, Singapore, special session on performance assessment of multi-objective optimization algorithms, technical report* 264 (2008).