Search Space Reduction Model with Trigonometric Function for Linear Equality Constraint-handling: The Case of Portfolio Replication Problem

1st Yukiko Orito
Department of Economics
Hiroshima University
Hiroshima, Japan
orito@hiroshima-u.ac.jp

2nd Yoshiko Hanada
Faculty of Engineering Science
Kansai University
Osaka, Japan
hanada@kansai-u.ac.jp

3rd Junzhi Li
School of Electronics Engineering
and Computer Science
Peking University
Beijing, P.R. China
ljz@pku.edu.cn

Abstract—In the constrained optimization problem, it is difficult that the evolutionary algorithm finds the optimal solution on the whole solution space. The algorithm needs a constraint-handling technique for obtaining the solutions only in the feasible region. In this paper, for the portfolio replication problem which is the linear equality constrained optimization problem, we propose the search space reduction model with larger dimensionality using trigonometric functions as a new constraint-handling technique. The proposed model is employed in an estimation of distribution algorithm and it is applied to the portfolio replication problem. In the numerical experiments, we show the usefulness of the proposed model as compared with a standard constraint-handling technique.

Index Terms—linear equality constraint-handling technique, large scale global optimization problem, trigonometric function, Pythagorean theorem

I. Introduction

In the unconstrained optimization problem, the evolutionary algorithm demonstrates high performance for finding the optimal solution even if it is faced with the complex and nonlinear problem. On the other hand, in the constrained optimization problem, it is difficult to find the optimal solution on the whole solution space consisting not only of feasible solutions but also of infeasible solutions. The evolutionary algorithm needs a constraint-handling technique for obtaining the solutions only in the feasible region, however it brings about the evolutionary stagnation as a critical problem due to this constraint-handling.

Many researchers have proposed the constraint-handling techniques used in the evolutionary algorithms. Coello Coello [1] provided the survey of popular constraint-handling techniques: the techniques are classified into five categories; penalty functions, special representations and operators, repair algorithms, separation of objectives and constraints, and hybrid methods. Mezura-Montes and Coello Coello [2] provided the survey of constraint-handling techniques in nature-inspired numerical optimization. Koziel and Michalewicz [3] proposed a homomorphous mapping technique for the inequality constrained optimization problem. Their technique partially transforms the whole solution space into the unconstrained search space. On the unconstrained search space consisting only of feasible solutions, the evolutionary algorithm does not need to handle the constraints, and therefore it works very well.

For the portfolio replication problem which is one of linear equality constrained optimization problems, Orito and Hanada [4] proposed a search space reduction model as one of mapping techniques. This model partially transforms the whole solution space into the unconstrained search space by reducing the search space size. However, in [4], the size of the unconstrained search space is much smaller than that of the whole solution space.

In this paper, for the portfolio replication problem, we propose the search space reduction model with larger dimensionality as a new constraint-handling technique, and then the proposed model is employed in an estimation of distribution algorithm (EDA). In the numerical experiments, the proposed model is compared with the model having smaller dimensionality.

This paper is organized as follows: The search space reduction model is proposed in Section II. The EDA with the search space reduction model is presented in Section III. The portfolio replication problem is described in Section IV. The results of numerical experiments are shown in Section V and we conclude this paper in Section VI.

II. Search Space Reduction Model

Let $\boldsymbol{x} = (x_1, \cdots, x_N)$ be a solution consisting of N design variables on the whole solution space. The linear equality constrained optimization problem in this paper is represented as follows.

$$\min f(\boldsymbol{x}), \tag{1}$$
$$s.t. \ \ g(\boldsymbol{x}) = \sum_{i=1}^{N} x_i = 1$$
$$x_i \geq 0 \ \ (i = 1, \cdots, N)$$

where x_i is the i-th variable of solution $\boldsymbol{x}$.

Equation (1) is one of general allocation problems where the constraint is denoted by the total of variable values.
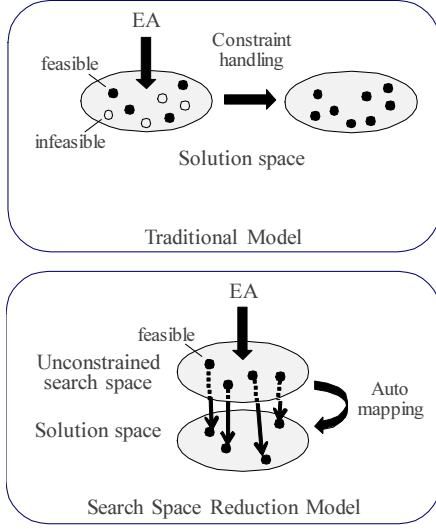
Fig. 1. Examples of constraint-handling techniques.

While the evolutionary algorithm is applied on the whole solution space for the optimization problem given by (1), the infeasible solution generated by it needs to be removed, evolved into, or repaired to the feasible solution. Therefore, the constraint-handling brings about the evolutionary stagnation.

In order to overcome such a problem, the search space reduction model has proposed as a constraint-handling technique. The model transforms the part of the whole solution space to the unconstrained search space. In using the model, the evolutionary algorithm finds only the feasible solution on the unconstrained search space transformed by the model. The solution on the unconstrained search space is mapped to the feasible solution on the whole solution space. Therefore, it is expected that the evolutionary algorithm effectively finds the feasible solution without evolutionary stagnation. The examples of two constraint-handling techniques, the traditional model and the search space reduction model, are shown in Fig. 1.

A. Search Space Reduction Model with Small Dimensionality (SSR_SD)

As defined as (1), the number of dimensions of the whole solution space is N and the solution on this space is represented as $\boldsymbol{x} = (x_1, \cdots, x_N)$. On the other hand, let $\boldsymbol{\theta} = (\theta_1, \cdots, \theta_M)$ be a solution consisting of M design variables on the unconstrained search space transformed by the search space reduction model.

In [4], the search space reduction model using trigonometric functions was proposed. We call it "SSR_SD" in this paper. In SSR_SD, the relationship between N and M is defined as follows.

$$M = \log_2 N \quad (N, M \in \mathbb{N}) \tag{2}$$

From (2), the solution on the N-dimensional solution space is mapped to the solution on the M $(= \log_2 N)$-dimensional unconstrained search space through SSR_SD.

Let $a_j \in \{0, 1\}$ $(j = 1, \cdots, M)$ be the j-th value of the binary number converted from the decimal number $i - 1$ $(i = 1, \cdots, N)$. In the case that, the feasible solution on the whole solution space by using SSR_SD is defined as follows.

$$\boldsymbol{x}^* = (x_1^*, \cdots, x_N^*) \quad (\boldsymbol{x}^* \subset \boldsymbol{x}) \tag{3}$$

$$x_i^* = \prod_{j=1}^{M} \left(\cos^2 \theta_j \right)^{a_j} \left(\sin^2 \theta_j \right)^{1-a_j} \quad (i = 1, \cdots, N)$$

From the Pythagorean theorem, that is,

$$\sin^2 \theta_j + \cos^2 \theta_j = 1 \quad (j = 1, \cdots, M), \tag{4}$$

the following equation is satisfied for any variable values of $\boldsymbol{\theta} = (\theta_1, \cdots, \theta_M)$.

$$\sum_{i=1}^{N} x_i^* = \prod_{j=1}^{M} \left(\sin^2 \theta_j + \cos^2 \theta_j \right) = 1 \tag{5}$$

Therefore, in the optimization problem given by (1), the evolutionary algorithm can obtain the feasible solution $\boldsymbol{x}^*$ on the whole solution space when it finds the solution $\boldsymbol{\theta}$ on the unconstrained search space.

For example, the size of the whole solution space is set to $N = 4$ and then the size of the unconstrained search space in SSR_SD is set to $M = \log_2 N = 2$. When the size is $M = 2$, the decimal number $i - 1$ $(i = 1, \cdots, 4)$ is expressed as the binary number $00_{(2)}$, $01_{(2)}$, $10_{(2)}$, and $11_{(2)}$, respectively. In the case that, the feasible solution on the solution space is expressed as follows.

$$\boldsymbol{x}^* = (x_1^*, \cdots, x_4^*) \tag{6}$$
$$x_1^* = \sin^2 \theta_1 \sin^2 \theta_2 \quad x_2^* = \sin^2 \theta_1 \cos^2 \theta_2$$
$$x_3^* = \cos^2 \theta_1 \sin^2 \theta_2 \quad x_4^* = \cos^2 \theta_1 \cos^2 \theta_2$$

Based on the Pythagorean theorem, the following equation is satisfied for any variable values of $\boldsymbol{\theta} = (\theta_1, \theta_2)$.

$$\sum_{i=1}^{4} x_i^* = \left(\sin^2 \theta_1 + \cos^2 \theta_1 \right) \left(\sin^2 \theta_2 + \cos^2 \theta_2 \right) = 1 \tag{7}$$

B. Search Space Reduction Model with Large Dimensionality (SSR_LD)

In SSR_SD, the size of the unconstrained search space M depends strongly on that of the whole solution space N. M is much smaller than N. If M is set as $M = 1, 2, 3, 4, \cdots$, N is given as $N = 2^M = 2, 4, 8, 16, \cdots$, respectively. In addition, for solving the general optimization problem, the size of the whole solution space N has already been decided, and therefore M of SSR_SD cannot express any N.

In order to overcome such a problem, the new search space reduction model based on SSR_SD is proposed in this paper.

We call it "SSR_LD." In SSR_LD, the relationship between N and M is defined as follows.

$$M = N - 1 \quad (N, M \in \mathbb{N}) \tag{8}$$

From (8), the solution on the N-dimensional solution space is mapped to the solution on the $M \,(= N - 1)$-dimensional unconstrained search space through SSR_LD.

The feasible solution on the whole solution space through SSR_LD is defined as follows.

$$\boldsymbol{x}^* = (x_1^*, \cdots, x_N^*) \quad (\boldsymbol{x}^* \subset \boldsymbol{x}) \tag{9}$$

$$x_i^* = \begin{cases} \sin^2 \theta_i & (i = 1) \\ \sin^2 \theta_i \prod_{j=1}^{i-1} \cos^2 \theta_j & (i = 2, \cdots, N-1) \\ \prod_{j=1}^{i-1} \cos^2 \theta_j & (i = N) \end{cases}$$

From the Pythagorean theorem given by (4), the following equation is satisfied for any variable values of $\boldsymbol{\theta} = (\theta_1, \cdots, \theta_M)$.

$$\sum_{i=1}^{N} x_i^* = \left(\sin^2 \theta_1 + \cos^2 \theta_1 \left(\sin^2 \theta_2 + \cos^2 \theta_2 \left(\cdots \right.\right.\right.$$
$$\left.\left.\left. \left(\sin^2 \theta_M + \cos^2 \theta_M\right)\right)\right)\right)$$
$$= 1 \tag{10}$$

Therefore, as in SSR_SD, the evolutionary algorithm used SSR_LD can obtain the feasible solution $\boldsymbol{x}^*$ on the whole solution space when it finds the solution $\boldsymbol{\theta}$ on the unconstrained search space.

For example, the size of the whole solution space is set to $N = 4$ and then the size of the unconstrained search space in SSR_LD is set to $M = N - 1 = 3$. In the case that, the feasible solution on the solution space is expressed as follows.

$$\boldsymbol{x}^* = (x_1^*, \cdots, x_4^*) \tag{11}$$
$$x_1^* = \sin^2 \theta_1 \quad x_2^* = \sin^2 \theta_2 \cos^2 \theta_1$$
$$x_3^* = \sin^2 \theta_3 \cos^2 \theta_1 \cos^2 \theta_2$$
$$x_4^* = \cos^2 \theta_1 \cos^2 \theta_2 \cos^2 \theta_3$$

Based on the Pythagorean theorem, the following equation is satisfied for any variable values of $\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3)$.

$$\sum_{i=1}^{4} x_i^* = \left(\sin^2 \theta_1 + \cos^2 \theta_1 \left(\sin^2 \theta_2 + \cos^2 \theta_2 \right.\right.$$
$$\left.\left. \left(\sin^2 \theta_3 + \cos^2 \theta_3\right)\right)\right)$$
$$= 1 \tag{12}$$

Therefore, SSR_LD has the contribution that the degree of freedom for the unconstrained search space is much higher than that of SSR_SD.

III. EDA with Search Space Reduction Model

As the EDAs in the evolutionary algorithms, Univariate Marginal Distribution Algorithm (UMDA) [5], Population-Based Incremental Learning (PBIL) [6], and Compact Genetic Algorithm (cGA) [7] are well known algorithms. The EDAs generate the new population according to the probability distribution which is estimated from the variables of the previous generation instead of the traditional crossover and mutation operations. They provide better results in the optimization problem with no significant interaction among design variables. In this paper, the EDA whose probability distribution is estimated from the histogram of the variables (Histogram EDA) [8] is applied to the linear equality constrained optimization problem. Although this Histogram EDA is employed as one of evolutionary algorithms, the search space reduction model can be adapted to other algorithms.

In the genetic representation of Histogram EDA, the individual is defined as the candidate solution on the unconstrained search space through the search space reduction model as follows.

$$\boldsymbol{\theta} = (\theta_1, \cdots, \theta_M) \tag{13}$$

When the individual $\boldsymbol{\theta}$ is given, the feasible solution $\boldsymbol{x}^*$ on the solution space is also mapped through the search space reduction model. In this paper, the evaluation function of Histogram EDA is defined as follows.

$$f(\boldsymbol{x}^*), \tag{14}$$

where x_i^* is the i-th variable of the feasible solution $\boldsymbol{x}^*$.

The procedure of Histogram EDA with search space reduction model consists of the following five steps.

1) Initial State
   At the initial generation of Histogram EDA, N_p individuals are randomly generated in the initial parents' population. It is defined as follows.

$$\boldsymbol{\theta}^{(1)} = (\theta_1, \cdots, \theta_M)^{(1)}$$
$$\vdots$$
$$\boldsymbol{\theta}^{(N_p)} = (\theta_1, \cdots, \theta_M)^{(N_p)}$$

   Each variable value is determined as a value on the range of $[0, \pi]$.
2) Histogram of Parents' Population
   For the parents' population $\boldsymbol{\theta}^{(k)}$, the variable on the range of $[0, \pi]$ is divided into H discrete intervals. The histogram consists of N_p individuals in the parents' population, and thereby Bin h $(h = 1, \cdots, H)$ is the bin of the discrete interval on the range of $[\pi(h-1)/H, \pi h/H]$. The histogram with frequencies to the bin h is defined as follows.

$$v_j[h] = \# \left\{ k \left| \frac{\pi(h-1)}{H} \le \theta_j^{(k)} < \frac{\pi h}{H} \right. \right\}, \quad (15)$$
$$(k \in \{1, \cdots, N_p\}, \ j = 1, \cdots, M, \ h = 1, \cdots, H)$$

where $\#$ means the frequencies at the bin h.

3) Probability Distribution

The offspring population consisting of N_o individuals are generated from the current parents' population. We assume that the histogram of j-th variable defined by (15) is the probability distribution of j-th variable for producing the offspring population.

Let $p_j[h]$ be the probability of j-th variable to the bin h. The probability distribution of j-th variable is defined as follows.

$$p_j[h] = \frac{1}{N_p} v_j[h] \quad (j = 1, \cdots, M, \ h = 1, \cdots, H)$$
$$(16)$$

The bin number of each variable is randomly selected according to the probability distributions given by (16). Let h' be the selected bin number. For producing the offspring population, the new variable value is randomly determined on the range of $[\pi(h'-1)/H, \pi h'/H]$.

4) Evaluation and Selection

When the individual $\boldsymbol{\theta}_k^{(k)}$ is determined, the feasible solution $\boldsymbol{x}_k^*$ is also mapped through the search space reduction model. For each of all individuals, the evaluation function value is obtained by (14). According to the evaluation function values, N_p individuals are selected by the elitism and the roulette wheel selections from the current parents' and offspring populations for the next generation.

5) Termination Criterion

The operations of producing the parents' population, making the probability distribution, producing the offspring population, and performing the evaluation and the selection are repeated until the maximal number of the repetitions is satisfied. The termination criterion of the iterative process is the generation size in Histogram EDA.

In the final generation, the best solution with the lowest evaluation function value of all is defined as the optimal or the quasi-optimal solution per one simulation.

IV. PORTFOLIO REPLICATION PROBLEM

As a real world application of the evolutionary algorithm, we focus on the portfolio replication problem, one of the popular equality constrained portfolio optimization problems. Many researchers have applied various evolutionary algorithms to the portfolio optimization problems [8]–[14].

In the portfolio replication problem, from a practical viewpoint, an asset management firm desires to replicate the portfolio of another firm if this portfolio has delivered good performances. Generally, the total return of portfolio provided by the asset management firm is opened to the public as the performance of the portfolio. However, the proportion-weighted combination in the portfolio is closed to the public. Therefore, it is difficult to replicate this portfolio even if all assets included in the portfolio are opened to the public.

In this portfolio replication problem, we optimize the replication portfolio such that its total return mimics the total return of the given benchmark portfolio in this paper. The replication portfolio consisting of N assets is represented as the vector $\boldsymbol{x} = (x_1, \cdots, x_N)$. Let $r_i(t)$ be the return of Asset i $(i = 1, \cdots, N)$ at t. The returns of replication portfolio consisting of N assets over the period between $t = 1$ and $t = T$ are represented as follows.

$$\begin{pmatrix} r_{\boldsymbol{x}}(1) \\ \vdots \\ r_{\boldsymbol{x}}(T) \end{pmatrix} = \begin{pmatrix} r_1(1) & \cdots & r_N(1) \\ & \vdots & \\ r_1(T) & \cdots & r_N(T) \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_N \end{pmatrix}$$
$$(17)$$

In (17), the total return of the optimal replication portfolio must be same as that of the given benchmark portfolio because the benchmark portfolio is the optimal solution. Therefore, the total return of the replication portfolio $(r_{\boldsymbol{x}}(1), \cdots, r_{\boldsymbol{x}}(T))$ is known data as the same return of the benchmark portfolio. The return of each of assets, $(r_i(1), \cdots, r_i(T))$ $(i = 1, \cdots, N)$, is also known data. If the number of days data T is equal to the number of assets N, the optimal solution is solved by using the simultaneous equations. On the other hand, if T is much less than N, it is difficult to find the optimal solution. In this context, we try to optimize the replication portfolio such that its total return mimics the total return of the given benchmark portfolio.

Let $\boldsymbol{x}^B = (x_1^B, \cdots, x_N^B)$ be the benchmark portfolio and $r_{\boldsymbol{x}^B}(t)$ be the its total return. For evaluating the replication portfolio, the function consisting of the errors of total returns and the rates of changes of total returns between the replication and the benchmark portfolios is defined as follows.

$$f(\boldsymbol{x}) = \sum_{t=1}^{T} \left(r_{\boldsymbol{x}}(t) - r_{\boldsymbol{x}^B}(t) \right)^2$$
$$+ \rho \sum_{t=1}^{T-1} \left(1 - \frac{r_{\boldsymbol{x}}(t+1) - r_{\boldsymbol{x}}(t)}{r_{\boldsymbol{x}^B}(t+1) - r_{\boldsymbol{x}^B}(t)} \right)^2, \quad (18)$$

where ρ is a parameter given in the numerical experiment.

In the linear equality constrained optimization problem, (18) is employed as the evaluation function in Histogram EDA.

V. NUMERICAL EXPERIMENTS

In the experiments, the benchmark portfolio consists of N assets with high turnover on the Tokyo Stock Exchange. The length of each of seven periods from 2011 to 2016 is set to $T = 100$, respectively. The parameter ρ in (18) is set to $\rho = 1.0E - 08$ as in [4]. The numerical experiments are implemented in C++.

The parameter setting in Histogram EDA is as follows.

- Parents' population size: $N_p = 100$
- Offspring population size: $N_o = 200$
- Elitist rate: 0.01
- The number of bins: $H = 100$ (The width of bin is set to π/H.)
- Generation size: 100
- Simulation size: 10

In order to analyze the usefulness of the search space reduction model, the following three models are compared in the numerical experiments.

1) Standard Repair Model (SRM)

   Many researchers proposed the evolutionary algorithms in the portfolio optimization problems [8]–[14]. Their all constraint-handling techniques are almost similar. When the evolutionary algorithm generates the infeasible solution, it is repaired to the feasible solution by the following model.

$$\boldsymbol{x} = (x_1, \cdots, x_N) = \frac{1}{\displaystyle\sum_{i=1}^{N} x_i'} (x_1', \cdots, x_N'), \quad (19)$$

   where x_i' $(i = 1, \cdots, N)$ is the infeasible solution. We call this model with N-dimensionality "SRM" in this paper. The SRM restores the feasible solution from the infeasible solution. This is the standard repair technique in the linear equality constrained optimization problem, however to construct the feasible solution may bring about evolutionary stagnation because of the constraint-handling.

2) Search Space Reduction Model with Small Dimensionality (SSR_SD)

   The search space reduction model with $(\log_2 N)$-dimensionality, SSR_SD, is described in Section II-A.

3) Search Space Reduction Model with Large Dimensionality (SSR_LD)

   As described in Section II-B, the search space reduction model with $(N-1)$-dimensionality, SSR_LD, is proposed in this paper.

As described in Section IV, the benchmark portfolio is defined as the optimal solution for the portfolio replication problem. In the numerical experiments, the following two benchmark portfolios, BP1 and BP2, are employed as the optimal solutions, respectively.

1) Benchmark portfolio consisting of $\log_2 N$ variables (BP1)

   For BP1, the $\log_2 N$ variable values of $\boldsymbol{\theta}$ are randomly determined on the unconstrained search space, and then they are mapped to the N variable values $\boldsymbol{x}^*$ on the solution space through SSR_SD. Therefore, when the BP1 is employed as the benchmark portfolio, it is

expected that the Histogram EDA with SSR_SD works better than with SSR_LD.

2) Benchmark portfolio consisting of $N-1$ variables (BP2)

   For BP2, the $N-1$ variable values of $\boldsymbol{\theta}$ are randomly determined on the unconstrained search space, and then they are mapped to the N variable values $\boldsymbol{x}^*$ on the solution space through SSR_LD. Therefore, when the BP2 is employed as the benchmark portfolio, it is expected that the Histogram EDA with SSR_LD works better than with SSR_SD.

For each of BP1 and BP2, the Histogram EDAs with SRM, SSR_SD, and SSR_LD are applied to the case studies of $N = 16$, 32, 64, 128, and 256 in each of all periods, respectively. The evaluation functions of the best solutions obtained by these three models are shown in Tables I-V, respectively. In the tables, the best result of three models is written by boldface.

From Tables I-V, for BP1, the evaluation functions of SRM are better than those of SSR_SD and SSR_LD on almost all periods when the solution space size is smaller. On the other hand, the evaluation functions of SSR_SD are much better than those of SRM and SSR_LD when the solution space size is larger. These results suggest that SRM is sufficient to optimize the small scale optimization problem, however it is not sufficient for the large scale optimization problem because it brings about evolutionary stagnation. For the large scale optimization problem, SSR_SD is very effective as the constraint-handling technique used in the evolutionary algorithm. For BP2, the evaluation functions of SSR_SD and SSR_LD are much better than those of SRM regardless of the size of solution space. From these all results, we can conclude that the search space reduction models are very effective for the large scale optimization problem as compared with the standard constraint-handling technique.

However, for BP2, the evaluation functions of SSR_LD are not better than those of SSR_SD though BP2 is employed as the benchmark portfolio which has the advantage to SSR_LD. Why SSR_LD is not better than SSR_SD as the search space reduction model? As mentioned in Section II, SSR_LD has the larger number of dimensions on the unconstrained search space, $M = N-1$ as compared with $M = \log_2 N$ of SSR_SD. However, the feasible solution x_i^* $(i = 1, \cdots, N)$ on the solution space is defined as the product of multiple $\cos^2 \theta_j$s. The value of cosine function $\cos^2 \theta_j$ depends on the value of θ_j. If θ_j is given as a value near $\pi/2$ on the range of $[0, \pi]$, $\cos^2 \theta_j$ in (9) is given as a small value near zero, and then the mapped x_i^* is also determined by zero or a much small value. Therefore, the feasible solution mapped by SSR_LD easily falls into a local optimal solution whose many variables are determined by zero or a much small value. To improve this problem is our future work.

VI. CONCLUSION

In this paper, we proposed the search space reduction model with large dimensionality as a new linear equality constraint-handling technique that partially transforms the feasible so-

TABLE I
EVALUATION FUNCTION ($N = 16$)

Period	BP1			BP2		
	SRM	*SSR_SD*	*SSR_LD*	*SRM*	*SSR_SD*	*SSR_LD*
1	**3.400E-05**	2.900E-04	1.640E-04	7.080E-04	6.660E-06	**2.390E-07**
2	**7.230E-05**	4.140E-04	5.050E-04	4.100E-04	7.360E-05	**5.160E-05**
3	2.300E-05	6.070E-04	**2.120E-05**	1.890E-04	3.040E-04	**4.080E-05**
4	3.260E-05	1.980E-04	**3.170E-05**	5.540E-04	5.840E-04	**1.300E-05**
5	**1.420E-05**	1.950E-04	1.220E-04	3.190E-04	**2.890E-06**	6.190E-06
6	**1.870E-05**	3.360E-04	1.320E-04	2.630E-04	**7.470E-05**	2.120E-04
7	**1.770E-04**	9.280E-04	3.440E-04	1.370E-04	**5.610E-05**	7.580E-05

TABLE II
EVALUATION FUNCTION ($N = 32$)

Period	BP1			BP2		
	SRM	*SSR_SD*	*SSR_LD*	*SRM*	*SSR_SD*	*SSR_LD*
1	**1.700E-04**	5.865E-04	5.120E-04	7.090E-03	**5.500E-05**	5.690E-05
2	**2.390E-04**	4.593E-04	4.130E-03	5.210E-02	**1.880E-04**	8.110E-04
3	6.890E-04	4.322E-04	**3.660E-04**	1.710E-02	9.380E-05	**6.520E-05**
4	2.530E-04	2.488E-04	**1.920E-04**	3.570E-03	**1.230E-05**	1.230E-04
5	**2.270E-04**	3.823E-04	1.070E-03	1.500E-03	**5.650E-05**	6.060E-04
6	**2.670E-04**	5.100E-04	9.970E-04	2.020E-03	**1.560E-04**	2.860E-04
7	**2.420E-04**	8.190E-04	7.350E-04	4.300E-03	1.340E-04	**1.060E-04**

TABLE III
EVALUATION FUNCTION ($N = 64$)

Period	BP1			BP2		
	SRM	*SSR_SD*	*SSR_LD*	*SRM*	*SSR_SD*	*SSR_LD*
1	8.660E-04	**8.366E-04**	1.830E-03	6.180E-03	**1.820E-04**	4.99E-04
2	**8.250E-04**	1.056E-03	2.590E-03	6.560E-03	**2.470E-04**	9.28E-04
3	8.920E-04	1.284E-03	**8.490E-04**	8.450E-03	**5.840E-04**	7.04E-04
4	1.060E-03	**7.994E-04**	1.930E-03	1.250E-02	**2.170E-04**	5.09E-04
5	5.810E-04	1.038E-03	**5.800E-04**	8.320E-03	1.400E-04	**1.07E-04**
6	**6.470E-04**	1.669E-03	1.570E-03	1.870E-02	2.010E-04	**6.36E-05**
7	8.460E-04	**6.912E-04**	1.330E-03	7.550E-03	**1.120E-05**	1.41E-04

lutions on the whole solution space into the solutions on the unconstrained search space. The evolutionary algorithm with this model has the contribution that it can work only on the unconstrained search space without evolutionary stagnation.

In the numerical experiments, we showed that the proposed model is effective for the large scale optimization problems as compared with the standard constraint-handling technique.

However, the proposed model has the problem that it is defined as the product of multiple trigonometric functions. In the case that, the feasible solution mapped by the model easily falls into a local optimal solution. We need to improve this problem as our future work.

REFERENCES

[1] C. A. C. Coello, "Theoretical and Numerical Constraint-handling Techniques Used with Evolutionary Algorithms: A Survey of the State of the Art," *Computer Methods in Applied Mechanics and Engineering*, vol. 191, pp. 1245–1287, 2002.

[2] E. Mezura-Montes and C. A. C. Coello, "Constraint-handling in Nature-inspired Numerical Optimization: Past, Present and Future," *Swarm and Evolutionary Computation*, vol. 1, pp. 173–194, 2011.

[3] S. Koziel, and Z. Michalewicz, "Evolutionary Algorithms, Homomorphous Mappings, and Constrained Parameter Optimization," *Evolutionary Computation*, vol. 7, no. 1, pp. 19–44, 1999.

[4] Y. Orito and Y. Hanada, "Equality Constraint-handling Technique with Various Mapping Points: The Case of Portfolio Replication Problem," *Proceedings of the 2015 IEEE Congress on Evolutionary Computation*, pp. 2573–2580, 2015.

[5] H. Muhlenbein, "The Equation for Response to Selection and Its Use for Prediction", *Evolutionary Computation*, vol. 5, issue. 3, pp. 303-346, 1997.

[6] S. Baluja, "Population-based Incremental Learning: A Method for Integrating Genetic Search Based Function Optimization and Competitive Learning", *Technical Report*, CMU-CS-94-163, *Carnegie Mellon University*, 1994.

[7] G. R. Harik, F. G. Lobo, and D. E. Goldberg, "The Compact Genetic Algorithm", *Technical Report*, 97006, *IlliGAL Report*, 1997.

[8] Y. Orito, H. Yamamoto, and Y. Tsujimura, "Equality Constrained Long-Short Portfolio Replication by Using Probabilistic Model-building GA," *Proceedings of the 2012 IEEE Congress on Evolutionary Computation*, pp. 513–520, 2012.

[9] Y. Xia, B. Liu, S. Wang, and K. K. Lai, "A Model for Portfolio Selection with Order of Expected Returns," *Computers & Operations Research*, vol. 27, pp. 409–422, 2000.

[10] C. C. Lin and Y. T. Liu, "Genetic Algorithms for Portfolio Selection

TABLE IV
EVALUATION FUNCTION ($N = 128$)

Period	BP1			BP2		
	SRM	*SSR_SD*	*SSR_LD*	*SRM*	*SSR_SD*	*SSR_LD*
1	2.759E-03	**6.512E-04**	3.694E-03	1.597E-02	2.958E-04	**1.856E-04**
2	5.066E-03	**9.211E-04**	1.151E-03	7.239E-03	**1.952E-04**	2.342E-04
3	1.351E-03	**5.875E-04**	8.623E-04	1.952E-02	2.835E-04	**1.917E-04**
4	1.629E-03	**6.194E-04**	3.257E-03	8.738E-03	2.745E-04	**2.083E-04**
5	1.128E-03	**5.549E-04**	1.680E-03	3.916E-03	**5.200E-04**	5.698E-04
6	2.152E-03	**6.211E-04**	1.459E-03	8.000E-03	**1.892E-04**	6.086E-03
7	2.093E-03	**1.210E-03**	3.113E-03	7.007E-03	**5.126E-04**	6.813E-03

TABLE V
EVALUATION FUNCTION ($N = 256$)

Period	BP1			BP2		
	SRM	*SSR_SD*	*SSR_LD*	*SRM*	*SSR_SD*	*SSR_LD*
1	2.550E-03	**8.690E-04**	2.480E-03	1.280E-02	**6.790E-04**	1.200E-02
2	1.790E-03	**1.220E-03**	2.310E-03	1.010E-02	4.970E-04	**3.070E-04**
3	2.050E-03	**9.560E-04**	2.900E-03	5.330E-03	**2.480E-04**	5.010E-03
4	2.620E-03	**1.830E-03**	3.800E-03	1.160E-02	**1.650E-03**	7.770E-03
5	1.690E-03	**8.190E-04**	1.980E-03	6.540E-03	**1.480E-03**	6.290E-03
6	1.570E-03	**1.250E-03**	1.700E-03	4.030E-02	**2.550E-04**	3.830E-02
7	4.090E-03	**1.680E-03**	3.310E-03	8.470E-03	**1.070E-03**	6.910E-03

Problems with Minimum Transaction Lots," *European Journal of Operational Research*, vol. 85, no. 1, pp. 393–404, 2008.

[11] T. J. Chang, N. Meade, J. E. Beasley, and Y. M. Sharaiha, "Heuristics for Cardinality Constrained Portfolio Optimization," *Computers & Operations Research*, vol. 27, pp. 1271–1302, 2000.

[12] Y. Crama and M. Schyns, "Simulated Annealing for Complex Portfolio Selection Problems," *European Journal of Operational Research*, vol. 150, pp. 546–571, 2003.

[13] K. J. Oh, T. Y. Kim, and S. Min, "Using Genetic Algorithm to Support Portfolio Optimization for Index Fund Management," *Expert Systems with Applications*, vol. 28, pp. 371–379, 2005

[14] Y. Orito, M. Inoguchi, and H. Yamamoto, "Index Fund Optimization Using a Genetic Algorithm and a Heuristic Local Search," *Electronics and Communications in Japan*, vol. 93, no. 10, pp. 42–52, 2010.