A mapping-based constraint-handling technique for evolutionary algorithms with its applications to portfolio optimization problems

Kiyoharu Tagawa and Yukiko Orito

School of Science and Engineering, Kindai University, Higashi-Osaka, Japan; Graduate School of Humanities and Social Sciences, Hiroshima University, Higashi-Hiroshima, Japan

ABSTRACT

A novel Constraint-Handling Technique (CHT) for Evolutionary Algorithms (EAs) applied to constrained optimization problems is proposed. It is assumed that the feasible region of the constrained optimization problem is defined by a convex-hull of multiple vertices. On the other hand, without loss of generality, the search space of EA is given by a hyper-cube. The proposed CHT called Convex-Hull Mapping (CHM) transforms the real vector in the search space of EA into the solution in the feasible region. It is also proven that CHM performs a surjective mapping from the search space of EA to the feasible region. Although the proposed CHM can be applied to any EAs, one of the latest EAs, or Adaptive Differential Evolution (ADE), is used in this paper. By using ADE, CHM is compared with conventional CHTs in a real-world optimization problem in the field of finance, namely the portfolio optimization problem. Portfolio optimization is the process of determining the best proportion of investment in different assets according to some objective. Specifically, to reveal the characteristic of CHM depending on the number of the above vertices, three different formulations of the portfolio optimization problem are employed to evaluate the performance of ADE using CHM. Numerical experiments show that CHM is better than conventional CHTs in most cases. Moreover, the hybrid method combining CHM with a conventional CHT outperforms the original CHT.

ARTICLE HISTORY

Received 18 October 2021
Revised 3 January 2022
Accepted 2 February 2022

KEYWORDS

Evolutionary algorithms; constraint-handling; portfolio optimization problems

1. Introduction

Various Evolutionary Algorithms (EAs) have been proposed nowadays as excellent and practical numerical optimization methods. However, most of EAs are originally designed for solving unconstrained optimization problems. On the other hand, real-world optimization problems are formulated as constrained ones in many cases. Thus Constraint-Handling Techniques (CHTs) for handling constraints with EAs are essential to cope with real-world optimization problems by using EAs effectively.

During the last few decades, a number of CHTs have been proposed for EAs [1]. Those CHTs can be categorized as four different classes:

(1) Penalty functions
(2) Mapping-based techniques
(3) Special operators
(4) Separation of objective function and constraints.

Except mapping-based techniques, CHTs modify the procedures of original EAs to deal with constraints. Without loss of generality, the search space of EA is given by a hyper-cube. Mapping-based techniques, which is also called decoders [2], transform real vectors in the search space of EA into feasible solutions of the constrained optimization problem. The largest advantage of mapping-based CHTs is that original EAs can be applied directly to the constrained optimization problem without making any changes on the problem formulation nor the procedures of EAs. On the other hand, it is difficult to provide a simple and generic mapping-based CHT as discussed below.

This paper proposes a novel mapping-based CHT called Convex-Hull Mapping (CHM) for EAs applied to constrained optimization problems. Let $X \subseteq \mathbb{R}^n$ be the feasible region of the constrained optimization problem. We assume that the feasible region $X$ is given by a convex hull of multiple vertexes. Therefore, the feasible region is a convex polyhedron. As stated above, the search space of EA is defined by a hyper-cube $S = [0, 1]^q$. Thereby, the proposed CHM provides

$\tau : S \rightarrow X$.

Figure 1 shows an image of the proposed CHM-mapping. The procedure of CHM is very simple. Furthermore, it is proven that CHM performs a surjective mapping from the search space $S \subseteq \mathbb{R}^q$ of EA to the feasible region $X \subseteq \mathbb{R}^n$. On the other hand, not all constrained optimization problems have the feasible region defined by a convex polyhedron. Besides, it is hard to obtain all vertexes of the convex polyhedrons for some cases. Therefore, a real-world optimization problem, or
the portfolio optimization problem, is used to demonstrate the usefulness of the proposed CHM.

The proposed CHM can be used for any EAs. Therefore, to evaluate the effectiveness of CHM, we have employed the basic Differential Evolution (DE) as an instance of EA in our previous paper [3]. DE has been shown to be a simple yet powerful EA [4–6]. However, the performance of DE depends on the setting of control parameters, namely the scale factor and crossover rate. Actually, we cannot expect any EA that is able to perform the best on all problems as No Free Lunch Theorem [7] has suggested. To eliminate the influence of differences in the search performances of EAs and to evaluate the effectiveness of CHM as fairly as possible, we employ a state-of-the-art EA called JADE [8]. For changing the values of the control parameters of DE adaptively, many Adaptive DE (ADE) have been proposed nowadays [8, 9]. Among them, JADE has been reported to be one of the most powerful ADE [10].

This paper is an extended version of the paper presented in the SICE Annual Conference 2021 [3] and differs from the paper in the following three points:

1. In addition to the basic DE, the latest ADE is used as an example of EAs.
2. The performance of CHM is compared with conventional CHTs.
3. Three different formulations are used for the portfolio optimization problem.

As we will discuss later, a variety of problem formulations using various constraints are reported for the portfolio optimization problem. The three types of formulations in this paper have suitable constraints, respectively, to evaluate the characteristics of CHM. Besides, these constraints are very common in many problem formulations of the portfolio optimization problem. Experimental results also show that CHM enhances the performance of EA rather than conventional CHTs in many cases.

The remainder of this paper is organized as follows. Section 2 describes the existing works related to mapping-based CHTs and portfolio optimization problems. Section 3 formulates the constrained optimization problem in which the feasible region is defined by a convex hull of vertices. Section 3 also describes the procedure of CHM. Section 4 explains the basic DE and the latest ADE called JADE. From Section 5 to Section 7, three different formulations of the portfolio optimization problem are described respectively. Then the performance of CHM is compared with conventional CHTs through the three types of portfolio optimization problems. Section 8 discusses the characteristic of CHM in terms of constraints for the three portfolio optimization problems. Finally, Section 9 concludes this paper and provides future work.

2. Related works

2.1. Constraint-Handling technique

Constrained optimization problems are usually formulated as

\[
\begin{align*}
\min \quad & f(x) \\
\text{s. t.} \quad & h_i(x) = 0, \quad i = 1, \ldots, b \\
& g_i(x) \leq 0, \quad i = b + 1, \ldots, m \\
& x_j \in \bar{x}_j, \quad j = 1, \ldots, n
\end{align*}
\]

(1)

where \( x = (x_1, \ldots, x_n) \in \mathbb{R}^n \) is the vector of decision variables. The number of decision variables is \( n \). The objective function and constraints can be linear or nonlinear. For handling equality constraints \( h_1(x) = 0 \), they are usually transformed into inequality ones such as \( g_i(x) = |h_i(x)| \leq \epsilon \) by using a very small value \( \epsilon > 0 \).

Even though many EAs have been proposed nowadays, the search space \( S \subseteq \mathbb{R}^n \) of EA is usually restricted to a hyper-cube such as \( S = [0, 1]^n \). Therefore, EAs cannot deal with constraints for the constrained optimization problem in (1) except the box-constraint: \( \bar{x}_j - x_j \leq 0, j = 1, \ldots, n \). The box-constraint can be eliminated easily by using a simple transformation from \( z \in S \) to \( x \in \mathbb{R}^n \) as follows:

\[
x_j = (\bar{x}_j - x_j) z_j + \bar{x}_j, \quad j = 1, \ldots, n
\]

(2)

where \( z = (z_1, \ldots, z_n) \in S \) and \( 0 \leq z_j \leq 1, j = 1, \ldots, n \).

To apply EAs to constrained optimization problems, many Constraint-Handling Techniques (CHTs) have been proposed for EAs [1]. Let \( X \subseteq \mathbb{R}^n \) be the feasible region of a constrained optimization problem. Thus every \( x \in X \) satisfies all constraints. Many CHTs consider the search space \( \tilde{S} \subseteq \mathbb{R}^n \) that contains the feasible region such as \( X \subseteq \tilde{S} \). Specifically, the search space of EA is defined as \( \tilde{S} = [\bar{x}_j, \bar{x}_j]^n \) by using the transformation in (2). Therefore, original EAs need to be modified to evaluate infeasible solutions \( x \notin X \) existing in the search space \( \tilde{S} \subseteq \mathbb{R}^n \).

An EA called GENOCOP [11] has been proposed for optimization problems in which the feasible regions \( X \subseteq \mathbb{R}^n \) are convex. GENOCOP uses a number of special operators to generate a new feasible solution from existing feasible ones. For example, by using a random
value $\alpha \in (0, 1)$, the arithmetical crossover of GENOCOP generates a feasible solution $x \in X$ from two feasible ones $x_1 \in X$ and $x_2 \in X$ as follows:

$$x = \alpha x_1 + (1 - \alpha) x_2.$$ (3)

GENOCOP is not a generic CHT but a kind of EA for solving convex-constrained optimization problems because the types of available operators are limited.

Several mapping-based CHTs have been proposed to provide a mapping from the search space $S \subseteq \mathbb{R}^n$ of EA to the feasible region $X \subseteq \mathbb{R}^n$. In Koziels's decoder [2], the search space of EA is given as $S = [-1, 1]^n \subseteq \mathbb{R}^n$. Then the homomorphous mapping [2] provides $\tau : S \rightarrow X$ by using a reference point $x_0 \in X$. Specifically, the vector $y \in S$ is transformed into the feasible solution $x \in X$ as

$$x = x_0 + \beta y$$ (4)

where the value of the scale factor $\beta \in \mathbb{R}$, $\beta \geq 0$ is decided for each $y \in S$.

For the homomorphous mapping in (4), the reference point $x_0 \in X$ has to be given properly because the performance of EA depends on the selection of $x_0 \in X$. Besides, if the boundary of the feasible region $X \subseteq \mathbb{R}^n$ is unknown, a binary search is needed to decide the value of the scale factor $\beta \in \mathbb{R}$ for each $y \in S = [-1, 1]^n$. The homomorphous mapping has also been extended to deal with non-convex feasible regions [2, 12]. However, its actual implementation is far from trivial and involves a high computational cost when it is used for practical optimization problems.

Mapping-based CHTs using the grid generation and the circle packing have been reported, respectively, to approximate Riemann mapping from the search space of EA to the feasible region $X \subseteq \mathbb{R}^n$. In Koziels's decoder [2], the search space of EA is given as $S = [-1, 1]^n \subseteq \mathbb{R}^n$. Then the homomorphous mapping [2] provides $\tau : S \rightarrow X$ by using a reference point $x_0 \in X$. Specifically, the vector $y \in S$ is transformed into the feasible solution $x \in X$ as

$$x = x_0 + \beta y$$ (4)

where the value of the scale factor $\beta \in \mathbb{R}$, $\beta \geq 0$ is decided for each $y \in S$.

For the homomorphous mapping in (4), the reference point $x_0 \in X$ has to be given properly because the performance of EA depends on the selection of $x_0 \in X$. Besides, if the boundary of the feasible region $X \subseteq \mathbb{R}^n$ is unknown, a binary search is needed to decide the value of the scale factor $\beta \in \mathbb{R}$ for each $y \in S = [-1, 1]^n$. The homomorphous mapping has also been extended to deal with non-convex feasible regions [2, 12]. However, its actual implementation is far from trivial and involves a high computational cost when it is used for practical optimization problems.

A Support Vector Machine (SVM) has been used to provide a mapping $\tau : S \rightarrow X$ [14]. The SVM has been trained in advance to discriminate feasible solutions from infeasible ones in a higher-dimensional feature space $H$. Then infeasible solutions are converted into feasible ones in the feature space. Thus the mapping $\tau : S \rightarrow X$ is defined as $\tau = \varphi^{-1} \circ \psi$ where $\varphi : S \rightarrow H$ and $\varphi^{-1} : H \rightarrow X$. However, it is non-trivial to prepare a good training dataset for the SVM. Besides, it is hard to realize the inverse mapping $\varphi^{-1} : H \rightarrow X$. Consequently, the SVM is not a generic CHT.

A Deep Neural Network (DNN) has also been used to provide a mapping $\tau : S \rightarrow X$ where the feasible region $X \subseteq \mathbb{R}^n$ is given by a set of non-convex areas [15]. To prepare a good training dataset for the DNN, a number of machine learning techniques are employed. Nevertheless, it is hard to provide a surjective mapping by the DNN. Thus the DNN-mapping needs to be used in combination with other CHTs.

In the portfolio replication problem, Orito et al. [16, 17] have proposed mapping-based CHTs. The search space $S \subseteq \mathbb{R}^q$ of EA is given by a hyper-cube, while the feasible region $X \subseteq \mathbb{R}^n$ is a hyper-plane. Even though the CHTs can be used to eliminate only one equality constraint, they are effective for large-scale optimization problems. That is because the dimensionality of the feasible region $X \subseteq \mathbb{R}^n$ is reduced in the corresponding search space $S \subseteq \mathbb{R}^q$ such that $n > q$. On the other hand, since each of the mappings $\psi : S \rightarrow X$ is defined by the product of multiple trigonometric functions, they cannot generate solutions $x \in X$ uniformly on the hyper-plane.

A certain mapping-based CHT has been used widely for portfolio optimization problems to eliminate one equality constraint [18]. The mapping-based CHT is called Similarity Transformation (ST) in this paper, and it will be detailed later.

### 2.2. Portfolio optimization problem

Portfolio optimization is the process of determining the best proportion of investment in different assets according to some objective. Since portfolio optimization is one of the most challenging problems in the field of finance [19], a number of problem formulations are reported. Optimization methods of mathematical programming [20] have been used widely to solve portfolio optimization problems formulated based on Markowitz’s model [21]. Recently, various EAs have also been applied to portfolio optimization problems which are extended variants of Markowitz’s model.

The cardinality constraint limits the portfolio to have a specified number of assets. Genetic Algorithm (GA), Tabu Search (TS), and Simulated Annealing (SA) have been applied to the portfolio optimization problem with the cardinality constraint [18]. An extended GA has been proposed to solve the portfolio optimization problem considering the costs for selling and buying assets to change a portfolio structure [22]. Evolutionary Strategy (ES) and Differential Evolution (DE) have been applied to the portfolio optimization problem including the assets with long and short positions [23]. Particle Swarm Optimization (PSO) has been used to optimize the portfolio for a cardinality constrained Markowitz’s model [24]. Fireworks Algorithm (FWA) [25] and Artificial Bee Colony (ABC) algorithm [26] have been proposed for solving the portfolio optimization problem based on the efficient frontier model [18]. An Estimation of Distribution Algorithm (EDA) has been proposed for the optimization problem of creating a replicate portfolio for a given portfolio with good returns [27].

All the above works have formulated portfolio optimization problems as constrained optimization problems and applied various EAs to them by using some CHTs.
3. Convex-Hull mapping

3.1. Constrained optimization problem

In this paper, the constrained optimization problem is formulated as

$$\min f(x) \quad \text{s.t.} \quad x \in X \subseteq \Re^n$$

(5)

where $X \subseteq \Re^n$ denotes the feasible region.

We assume that the feasible region $X \subseteq \Re^n$ of the optimization problem in (5) can be represented as a convex hull of multiple vertexes $\hat{x}_i \in \Re^n$, $i = 1, \ldots, p$ as

$$X = \text{Conv}(\hat{x}_1, \ldots, \hat{x}_i, \ldots, \hat{x}_p)$$

(6)

where $p$ is the number of vertexes $\hat{x}_i \in X \subseteq \Re^n$ and $p \geq n$.

By using all vertexes $\hat{x}_i \in X$, $i = 1, \ldots, p$ shown in (6), every feasible solution $x \in X$ of the constrained optimization problem in (5) can be represented as

$$x = \sum_{i=1}^{p} a_i \hat{x}_i$$

$$a_1 + \cdots + a_i + \cdots + a_p = 1$$

$$a_i \geq 0, \quad i = 1, \ldots, p.$$  

(7)

3.2. Procedure of CHM

EAs are applied to the constrained optimization problem in (5). We assume that all vertexes $\hat{x}_i \in X$, $i = 1, \ldots, p$ are known. The search space of EA is defined by a hyper-cube $S = [0, 1]^q \subseteq \Re^n$. Therefore, the vector $z \in S$ has $q$ elements as

$$z = (z_1, \ldots, z_j, \ldots, z_q)$$

(8)

where $0 \leq z_j \leq 1$, $j = 1, \ldots, q$ and $q = p-1$.

The proposed CHM transforms the vector $z \in S$ into the feasible solution $x \in X$. The procedure of CHM, namely $\tau : S \rightarrow X$, is described as follows:

**Step 1:** Sort all elements $z_j \in [0, 1], j = 1, \ldots, q$ of the vector $z \in S$ in ascending order. Then generate a new vector $\tilde{z} = (\tilde{z}_1, \ldots, \tilde{z}_j, \ldots, \tilde{z}_q)$ such as

$$0 \leq \tilde{z}_1 \leq \tilde{z}_2 \leq \ldots \leq \tilde{z}_j \leq \ldots \leq \tilde{z}_{q-1} \leq \tilde{z}_q \leq 1.$$  

(9)

**Step 2:** Decide the values of coefficients $a_i \in \Re$, $i = 1, \ldots, p$ as

$$a_1 = \tilde{z}_1$$

$$a_2 = \tilde{z}_2 - \tilde{z}_1$$

$$\vdots$$

$$a_i = \tilde{z}_i - \tilde{z}_{i-1}$$

$$\vdots$$

$$a_{p-1} = \tilde{z}_q - \tilde{z}_{q-1}$$

$$a_p = 1 - \tilde{z}_q$$

(10)

**Step 3:** Compose the feasible solution $x \in X$ as

$$x = \sum_{i=1}^{p} a_i \hat{x}_i$$

(11)

where $\hat{x}_i \in \Re^n$, $i = 1, \ldots, p$ are the vertexes of the feasible region $X \subseteq \Re^n$.

**Theorem 3.1:** CHM $\tau : S \rightarrow X$ is a surjective mapping.

**Proof:** See Appendix 1.

3.3. Example of CHM

Let $X \subseteq \Re^2$ be a feasible region of the constrained optimization problem in (5). The feasible region $X \subseteq \Re^2$ is a convex hull of the following three vertexes:

$$\hat{x}_1 = (0, 0), \quad \hat{x}_2 = (2, 0.5), \quad \hat{x}_3 = (1, 1.5).$$

Since the number of vertexes is $p = 3$, the search space of EA is given as $S = [0, 1]^q$, $q = p-1 = 2$. Figure 2 shows 50 vectors $z_k \in S$, $k = 1, \ldots, 50$ generated randomly. Figure 3 shows a set of the corresponding feasible solutions $x_k \in X$ generated from the vectors $z_k \in S$ by using CHM such as $x_k = \tau(z_k)$, $k = 1, \ldots, 50$.

For example, from $z = (0.5, 0.2) \in S$, coefficients in (10) are calculated as

$$a_1 = 0.2, \quad a_2 = 0.5 - 0.2 = 0.3, \quad a_3 = 1 - 0.5 = 0.5.$$  

From (11), the corresponding feasible solution $x \in X$ is composed as

$$x = a_1 \hat{x}_1 + a_2 \hat{x}_2 + a_3 \hat{x}_3 = (1.1, 0.9).$$

![Figure 2. Search space: $S = [0, 1]^2$.](image1)

![Figure 3. Feasible region: $X \subseteq \Re^2$.](image2)
4. Evolutionary algorithm

4.1. Optimization problem

Without loss of generality, most EAs that do not have some kind of CHT can only be applied to the unconstrained optimization problem formulated as

$$\min \ g(z)$$

s. t. $$z \in S = [0, 1]^q$$ \hspace{1cm} (12)

where $$S = [0, 1]^q \subseteq \mathbb{R}^q$$ denotes the search of EA.

If the objective function in (12) is defined as $$g = f \circ \tau$$ by using CHM $$\tau: S \rightarrow X$$, any EAs can be applied to the constrained optimization problem in (5). Specifically, we have only to solve the unconstrained optimization problem in (12) by using EA. Then the best solution $$z_k \in S$$ obtained by EA is converted into the best solution $$x_k \in X$$ of the constrained optimization problem in (5) such as $$x_k = \tau(z_k)$$.

4.2. Differential evolution

As an instance of EAs for solving the optimization problem in (12), we explain the basic DE [4]. DE has NP individuals, or vectors $$z_k \in P_t$$, $$k = 1, \ldots, NP$$, in the population $$P_t \subseteq S$$ where $$t$$ denotes the current generation. The initial population $$P_0 \subseteq S$$ is generated randomly. Then each $$z_k \in P_t$$, $$k = 1, \ldots, NP$$ is assigned to the target vector in turn. The basic DE uses the strategy called “DE/rand/1/bin” [4] to generate a new vector $$u_k \in S$$, which is compared with $$z_k \in P_t$$, as follows.

By using three vectors $$z_{k1}, z_{k2},$$ and $$z_{k3}$$ ($$k \neq k1 \neq k2 \neq k3$$) selected from $$P_t$$ randomly, the mutation vector $$v_k = (v_{1,k}, \ldots, v_{q,k}) \in \mathbb{R}^q$$ is generated as

$$v_k = z_{k1} + F(z_{k2} - z_{k3})$$ \hspace{1cm} (13)

where the value of the scale factor $$F \in (0, 1]$$ is given in advance.

In the case of $$v_k \not\in S$$, outside elements $$u_{j,k} \in \mathbb{R}$$ are corrected as

$$v_{j,k} = \begin{cases} 0 & \text{if } v_{j,k} < 0 \\ 1 & \text{if } v_{j,k} > 1. \end{cases}$$ \hspace{1cm} (14)

By merging the elements of the target vector $$z_k \in P_t$$ and $$v_k \in S$$ in (13), a new vector $$u_k = (u_{1,k}, \ldots, u_{q,k}) \in S$$ called the trial vector is generated as

$$u_{j,k} = \begin{cases} v_{j,k} & \text{if } \text{rand } j \leq CR \lor j = j_r \\ z_{j,k} & \text{otherwise} \end{cases}$$ \hspace{1cm} (15)

where the value of the crossover rate $$CR \in [0, 1]$$ is given in advance. rand $$j \in [0, 1]$$, $$j = 1, \ldots, q$$ denote uniform random numbers. The index of element $$j_r \in [1, q]$$ is also selected randomly to avoid generating redundant vectors like $$u_k = z_k$$.

In each of the generations, the trial vector $$u_k \in S$$, $$k = 1, \ldots, NP$$ is compared with the corresponding target vector $$z_k \in P_t$$. Then either the trial vector $$u_k \in S$$ or the target vector $$z_k \in P_t$$ is selected for a member of the next generation $$z_k \in P_{t+1}$$. Specifically, $$u_k \in S$$ is chosen if $$g(u_k) \leq g(z_k)$$. Otherwise, $$z_k \in P_t$$ is chosen.

In this paper, the termination condition of DE is given by the maximum number of generations $$NG$$. Then the algorithm of the basic DE is described as follows:

**Step 1:** Generate the initial $$P_0 \subseteq S = [0, 1]^q$$ randomly.

**Step 2:** Evaluate $$g(z_k)$$ for $$z_k \in P_0$$, $$k = 1, \ldots, NP$$.

**Step 3:** If $$t = NG$$ holds, output the best vector $$z_k \in P_t$$ and terminate.

**Step 4:** For each $$z_k \in P_t$$, generate $$u_k \in S$$, $$k = 1, \ldots, NP$$.

**Step 5:** Evaluate $$g(u_k)$$ for $$u_k \in S$$, $$k = 1, \ldots, NP$$.

**Step 6:** Select either $$z_k \in P_t$$ or $$u_k \in S$$ for $$z_k \in P_{t+1}, k = 1, \ldots, NP$$.

**Step 7:** $$t = t + 1$$. Go back to Step 3.

4.3. Adaptive differential evolution

DE has been shown to be a simple yet efficient EA for many optimization problems. Its performance, however, is still quite dependent on the setting of control parameters, namely the scale factor $$F \in (0, 1]$$ in (13) and the crossover rate $$CR \in [0, 1]$$ in (15). Therefore, various parameter control methods have been reported for DE [10]. Adaptive DE (ADE) employs a parameter control method in which feedback from the evolution- ary search is used to dynamically change the control parameters.

In this paper, a powerful ADE called JADE [8] is chosen among various ADE to solve the unconstrained optimization problem in (12). As well as the basic DE, JADE assigns each control method in which feedback from the evolutionary search is used to dynamically change the control parameters.

In this paper, a powerful ADE called JADE [8] is chosen among various ADE to solve the unconstrained optimization problem in (12). As well as the basic DE, JADE assigns each control method in which feedback from the evolutionary search is used to dynamically change the control parameters.

DE has been shown to be a simple yet efficient EA for many optimization problems. Its performance, however, is still quite dependent on the setting of control parameters, namely the scale factor $$F \in (0, 1]$$ in (13) and the crossover rate $$CR \in [0, 1]$$ in (15). Therefore, various parameter control methods have been reported for DE [10]. Adaptive DE (ADE) employs a parameter control method in which feedback from the evolutionary search is used to dynamically change the control parameters.

In this paper, a powerful ADE called JADE [8] is chosen among various ADE to solve the unconstrained optimization problem in (12). As well as the basic DE, JADE assigns each control method in which feedback from the evolutionary search is used to dynamically change the control parameters.
distribution of mean $\mu_{CR}$ and standard deviation 0.1 as

$$CR_k \sim \text{Normal}(\mu_{CR}, 0.1^2)$$  \hspace{1cm} (18)

and then truncated to be $CR_k \in [0, 1]$ if $CR_k < 0$ or $CR_k > 1$.

Instead of $CR$, $CR_k$ is used in (15) to generate $u_k \in S$. If $u_k \in S$ is better than $z_k \in P_t$, associate control parameters $F_k$ and $CR_k$ are called successful ones.

The location $\mu_F$ in (16) and the mean $\mu_{CR}$ in (18) are initialized to be 0.5. After that, by using the sets of successful control parameters $S_F = \{F_k\}$ and $S_{CR} = \{CR_k\}$, they are updated, respectively, at the end of each generation as follows:

$$\mu_F = 0.9 \mu_F + 0.1 \text{mean}_E(S_F)$$  \hspace{1cm} (19)

where $\text{mean}_E(S_F)$ denotes the Lehmer mean of $F_k \in S_F$ [8].

$$\mu_{CR} = 0.9 \mu_{CR} + 0.1 \text{mean}_A(S_{CR})$$  \hspace{1cm} (20)

where $\text{mean}_A(S_{CR})$ denotes the arithmetic mean of $CR_k \in S_{CR}$ [8].

Except the above parameter control method and the greedy strategy in (17), the algorithm of JADE is almost the same with the basic DE.

### 4.4. Adaptive DE using feasibility rule

The feasibility rule [1] is one of the most widely used CHTs because of its simplicity and efficiency. By using the feasibility rule and the transformation in (2), JADE can be applied to the constrained optimization problem formulated in (1).

From the constraints $g_i(x) \leq 0$, $i = 1, \ldots, m$ of the optimization problem in (1), the constraint violation is defined for a candidate solution $x \in \mathbb{R}^n$ as

$$\phi(x) = \max[0, \max_i g_i(x)]$$  \hspace{1cm} (21)

where $x \in \mathbb{R}^n$ is infeasible if $\phi(x) > 0$, while $x \in \mathbb{R}^n$ is feasible if $\phi(x) = 0$.

The search space of JADE applied to the constrained optimization problem in (1) is defined as $S = [x_L, x_U]^n$. The search space $\tilde{S} \subseteq \mathbb{R}^n$ contains the feasible region $\tilde{X} \subseteq \mathbb{R}^n$ such as $X \subseteq \tilde{S}$. The algorithm of JADE using the feasibility rule is the same with the original JADE except the range of the search space and the selection method for the member of the next generation $z_{k} \in P_{t+1}$. As stated above, the trial vector $u_k \in \tilde{S}$ is generated and compared with the target vector $z_k \in P_t$. The values of the objective function in (1) and the constraint violation in (21) are evaluated for $u_k \in \tilde{S}$ and $z_k \in P_t$. Then if either of the following two conditions is satisfied,

$$\begin{cases}
\phi(u_k) < \phi(z_k) \quad \text{or} \quad (\phi(u_k) = \phi(z_k)) \land (f(u_k) \leq f(z_k))
\end{cases}$$  \hspace{1cm} (22)

$u_k \in \tilde{S}$ is chosen for $z_k \in P_{t+1}$. Otherwise, $z_k \in P_t$ is chosen for $z_k \in P_{t+1}$.

## 5. Efficient frontier model

### 5.1. Problem formulation

We invest our money in $n$ assets. Let $x_j \in \mathbb{R}$, $j = 1, \ldots, n$ be the proportion held of asset $j$. The portfolio is defined as $x = (x_1, \ldots, x_n)$. A long-only portfolio means to buy an asset, while a sort position means to sell an asset. In a long trade, we buy an asset and wait to sell it when the price rises. In the problem formulation, however, a long-only portfolio $x \in \mathbb{R}^n$ is considered over a single period. Thus $x \in \mathbb{R}^n$ is constrained as

$$\begin{align*}
x_1 + \cdots + x_j + \cdots + x_n &= 1 \\
0 &\leq x_j \leq 1, \quad j = 1, \ldots, n.
\end{align*}$$  \hspace{1cm} (23)

Let $\mu_j \in \mathbb{R}$, $j = 1, \ldots, n$ be the expected return of asset $j$ in the period. Besides, let $\sigma_{ij} \in \mathbb{R}$ be the covariance between the expected returns of assets $i$ and $j$.

The return of a portfolio $x \in \mathbb{R}^n$ is evaluated as

$$\mu_r(x) = \sum_{j=1}^{n} \mu_j x_j = \mu^T x$$  \hspace{1cm} (24)

where $\mu = (\mu_1, \ldots, \mu_n) \in \mathbb{R}^n$ and $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$.

The risk of a portfolio $x \in \mathbb{R}^n$ is evaluated as

$$\sigma_r(x) = \sqrt{x^T C x}$$  \hspace{1cm} (25)

where $C = [\sigma_{ij}] \in \mathbb{R}^{n \times n}$ is the covariance matrix.

There is a trade-off relationship between the return and the risk. By using a risk aversion indicator $\lambda \in [0, 1]$, the portfolio optimization problem is formulated as

$$\begin{cases}
\min \quad f(x) = \lambda \sigma_r(x) - (1 - \lambda) \mu_r(x) \\
\text{s.t.} \quad x_1 + \cdots + x_j + \cdots + x_n &= 1 \\
0 &\leq x_j \leq 1, \quad j = 1, \ldots, n.
\end{cases}$$  \hspace{1cm} (26)

where the risk aversion indicator $\lambda \in [0, 1]$ is given by investors.

The gradient of the objective function $f(x)$ in (26) can be derived as

$$\nabla f(x) = \lambda \frac{x C}{\sqrt{x^T C x}} - (1 - \lambda) \mu.$$  \hspace{1cm} (27)

From constraints in (26), we define the activity of a feasible solution $x \in X$ as

$$\gamma(x) = \min \left\{ \min_j (x_j), \min_j (1 - x_j) \right\}.$$  \hspace{1cm} (28)

If $\gamma(x) = 0$ holds, at least one constraint is active with $x \in X$. From Karush–Kuhn–Tucker (KKT) conditions [28], the optimal solution $x^* \in X$ of the portfolio optimization problem in (26) satisfies either $|\nabla f(x^*)| = 0$ or $\gamma(x^*) = 0$. 
of EA is defined as a solution problem as shown in (12). Besides, the search space is transformed into an unconstrained optimization problem in (26). The feasible region of EA in (26) is transformed into a set of feasible solutions \( S \). Hence, the feasible region of EA can be represented by a convex-hull of the following \( p \) vertices \( \hat{x}_i \in \mathbb{R}^n \), \( p = n \):

\[
\begin{align*}
\hat{x}_1 &= (1, 0, 0, \ldots, 0) \\
\hat{x}_2 &= (0, 1, 0, \ldots, 0) \\
&\vdots \\
\hat{x}_{n-1} &= (0, \ldots, 0, 1, 0) \\
\hat{x}_n &= (0, \ldots, 0, 1, 0)
\end{align*}
\]  

By using CHM, the portfolio optimization problem in (26) is transformed into an unconstrained optimization problem as shown in (12). Besides, the search space of EA is defined as \( S = [0, 1]^q, q = n-1 \). Therefore, the dimension of the feasible region \( X \subseteq \mathbb{R}^n \) is reduced in the search space of EA \( S = [0, 1]^q \subseteq \mathbb{R}^q \) such as \( q < n \).

Actually, many EAs have been applied to portfolio optimization problems which are constrained by (23). For the conventional EAs, the search space of EA has been defined as \( S = [0, 1]^n \). Then a real vector \( z \in S = [0, 1]^n \) is converted into a feasible solution \( x \in X \subseteq \mathbb{R}^n \) by using a mapping-based CHT as follows:

\[
x_j = \frac{z_j}{z_1 + \cdots + z_j + \cdots + z_n}, \quad j = 1, \ldots, n
\]  

where \( z = (z_1, \ldots, z_n) \in \mathbb{R}^n \) and \( x = (x_1, \ldots, x_n) \in \mathbb{R}^n \).

The mapping-based CHT in (30) is called Similarity Transformation (ST) in this paper. By using ST in (30), the portfolio optimization problem in (26) can also be transformed into an unconstrained optimization problem as shown in (12).

CHM is compared with ST in the diversity of solutions distributed on the feasible region \( X \subseteq \mathbb{R}^n \). As shown in Figure 4, the feasible region \( X \subseteq \mathbb{R}^n \) is a hyper-plane. Let \( P_0 \subseteq S \) be an initial population generated randomly. The initial population \( P_0 \subseteq S \) is transformed into a set of feasible solutions \( X_0 \subseteq X \) by using

\[
\begin{align*}
\text{ST} &\rightarrow \text{CHM} \\
\text{ST} &\rightarrow \text{CHM}
\end{align*}
\]  

Figure 5. The diversity of the set of \( x_k \in X_0 \subseteq \mathbb{R}^n, k = 1, \ldots, NP \) for the dimension \( n \). The population size is chosen as \( NP = 400 \). The feasible solutions \( x_k \in X_0 \) are generated by using CHM and ST.

Figure 6. The diversity of the set of \( x_k \in X_0 \subseteq \mathbb{R}^n, k = 1, \ldots, NP \) for the population size \( NP \). The dimension is chosen as \( n = 40 \). The feasible solutions \( x_k \in X_0 \) are generated by using CHM and ST.

CHM and ST respectively. Then the diversity of the set of feasible solutions \( x_k \in X_0, k = 1, \ldots, NP \) is measured by using an evaluation function \( \eta(X_0) \) defined as follows:

\[
\eta(X_0) = \frac{1}{NP} \sum_{k=1}^{NP} \left( \sum_{j=1}^{n} (x_{j,k} - \theta_j)^2 \right)
\]

where \( \theta_j \) is the average of the \( j \)-dimension over all solutions \( x_k \in X_0 \).

Figure 5 compares CHM with ST in the diversity of solutions \( x_k \in X_0 \) evaluated by (31). The population size, or the number of \( x_k \in X_0 \), is chosen as \( NP = 400 \). The horizontal axis of Figure 5 is the dimension of \( x_k \in \mathbb{R}^n \). The results in Figure 5 are the average of 100 runs. From Figure 5, the proposed CHM outperforms the conventional ST in the diversity of feasible solutions \( x_k \in X_0 \) transformed from \( P_0 \subseteq S \).

Figure 6 compares CHM with ST in the same way with Figure 5. The dimension of \( x_k \in X_0 \subseteq \mathbb{R}^n \) is fixed at \( n = 40 \). The horizontal axis of Figure 6 is the population size. From Figure 6, we can confirm that CHM outperforms ST in the diversity of solutions \( x_k \in X_0 \). Furthermore, from Figure 6, we can see that the diversity of solutions \( x_k \in X_0, k = 1, \ldots, NP \) cannot be enhanced only by increasing the population size \( NP \).

The basic DE and JADE are applied, respectively, to the portfolio optimization problem in (26) by using the following three methods. From the result of a preliminary experiment, the parameters \( NP = 100 \) and \( NG = 300 \) are chosen for them.
Figure 7. Convergence graph: the objective function values \( f(x_k) \) of the best solutions \( x_k \in P_t \) obtained by the basic DE using three methods for the portfolio optimization problem in (26) with \( \lambda = 0.1 \).

Figure 8. Convergence graph: the objective function values \( f(x_k) \) of the best solutions \( x_k \in P_t \) obtained by the basic DE using three methods for the portfolio optimization problem in (26) with \( \lambda = 0.9 \).

(1) CHM-based: \( z_k \in P_t \) is transformed into \( x_k \in X \) by using CHM.

(2) ST-based: \( z_k \in P_t \) is transformed into \( x_k \in X \) by using ST.

(3) CHM-ST hybrid: A set of initial solutions \( X_0 \subseteq X \) is generated by using CHM. Let \( P_0 = X_0 \). After that, \( z_k \in P_t \) is transformed into \( x_k \in X \) by using ST.

A data set of assets provided by OR-Library [29] is used for an instance of the portfolio optimization problem in (26) where the number of assets is \( n = 31 \). Two values are chosen for the risk aversion indicator \( \lambda \in [0, 1] \), namely \( \lambda = 0.1 \) and \( \lambda = 0.9 \). We can obtain a risk-loving portfolio, or a solution \( x \in \mathbb{R}^n \), if \( \lambda = 0.1 \) is used. On the other hand, we can obtain a risk-averse portfolio \( x \in \mathbb{R}^n \) if \( \lambda = 0.9 \) is used.

### 5.3. Experimental results of using DE

By using the above three methods, the basic DE is applied to the portfolio optimization problem in (26). The parameters \( F = 0.5 \) and \( CR = 0.9 \) are chosen for DE. Figure 7 shows the objective function values \( f(x_k) \) of the best solutions \( x_k \in P_t \) at generation \( t \) where the risk aversion indicator is chosen as \( \lambda = 0.1 \). The results of Figure 7 are the average of 30 runs. Similarly, Figure 8 shows the values of \( f(x_k) \) for \( \lambda = 0.9 \).

From Figure 7 and Figure 8, the best solution \( x_k \in P_t \) obtained by using CHM-based method is obviously better than the other methods. On the other hand, the best solutions \( x_k \in P_t \) obtained by ST-based and CHM-ST hybrid methods have converged to poor solutions because of the lack of diversity in the population \( P_t \).

Table 1 compares the best solutions \( x_k \in X \) obtained for \( \lambda = 0.1 \) by DE using the three methods respectively. Table 1 shows the objective function \( f(x_k) \) in (26), the absolute value of the gradient \( |\nabla f(x_k)| \) in (27), the activity of solution \( \gamma(x_k) \) in (28), and the distance \( d(x_k) \) from the origin \( 0 \in \mathbb{R}^n \) to the solution \( x \in \mathbb{R}^n \) defined as

\[
d(x) = \sqrt{x^2} = \sqrt{x_1^2 + \cdots + x_n^2}
\]

where \( 1/\sqrt{n} \leq d(x) \leq 1 \) holds for the portfolio optimization problem in (26).

The results of Table 1 are the average of 30 runs. Similarly, Table 2 compares the best solutions \( x_k \in X \) obtained by DE using the three methods for \( \lambda = 0.9 \).

From the values of \( f(x_k) \) in Table 1 and Table 2, we can see that CHM-based method is obviously better than the other methods. Furthermore, the best solution \( x_k \in X \) obtained by using CHM might be an optimal one because of \( \gamma(x_k) \approx 0 \).

### 5.4. Experimental results of using JADE

By using the three methods, JADE is applied to the portfolio optimization problem in (26). Figure 9 shows the objective function values \( f(x_k) \) of the best solutions \( x_k \in P_t \) for the portfolio optimization problem with \( \lambda = 0.1 \). The results in Figure 9 are the average of 30 runs. Similarly, Figure 10 shows the values of \( f(x_k) \) for \( \lambda = 0.9 \).

From Figure 9 and Figure 10, the best solutions \( x_k \in P_t \) obtained by JADE seem to have converged sufficiently at \( t = 200 \) in every case. Therefore, we can confirm that the number of generations, or \( NG = 300 \), chosen for JADE is enough large.

In the same way with Table 1, Table 3 compares the best solutions \( x_k \in X \) obtained by JADE using three methods for the portfolio optimization problem in (26) with \( \lambda = 0.1 \). Similarly, Table 4 compares the best solutions \( x_k \in X \) for \( \lambda = 0.9 \).
$\lambda = 0.1 \quad \text{CHM} \quad \text{ST} \quad \text{CHM-ST}$

$\lambda = 0.9 \quad \text{CHM} \quad \text{ST} \quad \text{CHM-ST}$

From the values of $\gamma(x_b)$ $\simeq 0$ in Table 3 and Table 4, the best solutions $x_b \in X$ obtained by JADE using the three methods satisfy the optimality condition.

Unfortunately, from the objective function values $f(x_b)$ in Table 3 and Table 4, we cannot confirm the difference between the best solutions $x_b \in X$ obtained by using the three methods. Therefore, by using Steel–Dwass’s test, we evaluate the average ranking of the objective function values. Table 5 shows the result of Steel–Dwass’s test about the value of $f(x_b)$ in (26) obtained for $\lambda = 0.1$ in which * means $p < 0.05$; ** means $p < 0.01$; n.s. means not significant. Similarly, Table 6 shows the result of Steel–Dwass’s test about the value of $f(x_b)$ in (26) obtained for $\lambda = 0.9$.

From Table 5, CHM-based method is better than ST-based method for the case of $\lambda = 0.1$. Besides, there is no significant difference between CHM-based method and CHM-ST hybrid method. On the other hand, from Table 6, ST-based method is better than CHM-based method for the case of $\lambda = 0.9$. However, CHM-ST hybrid method is significantly better than ST-based method and is the best method in Table 6.

From the values of $d(x_b)$ in Table 3 and Table 4, the location of the best solution $x_b \in X$ for the portfolio optimization problem in (26) clearly depends on the value of $\lambda \in [0, 1]$. The risk-loving portfolio obtained with $\lambda = 0.1$ is far from the center of the feasible region $X \subseteq \mathbb{R}^n$. On the other hand, the risk-averse portfolio obtained with $\lambda = 0.9$ exists near the center of the feasible region $X \subseteq \mathbb{R}^n$. From Figure 5, ST tends to generate more solutions $x_k \in X$ in the center of the feasible region $X \subseteq \mathbb{R}^n$. As a result, ST-based method is better than CHM-based method for $\lambda = 0.9$.

### 6. Problem formulation

#### 6.1. Portfolios including deposit

The deposit means an asset that has a constant rate $r_0 \in \mathbb{R}$. The rate $r_0 \in \mathbb{R}$ is small but positive. Let $x_0 \in \mathbb{R}^n$ be the proportion for the deposit. Considering the deposit as an investment option [30], the portfolio $x \in \mathbb{R}^n$ is constrained as

$$x_0 + x_1 + \cdots + x_j + \cdots + x_n = 1,$$

$$0 \leq x_0 \leq 1, \quad 0 \leq x_j \leq 1, \quad j = 1, \ldots, n. \quad (33)$$

The proportion of the deposit $x_0 \in \mathbb{R}$ can be eliminated from (33) as

$$x_1 + \cdots + x_j + \cdots + x_n \leq 1,$$

$$0 \leq x_j \leq 1, \quad j = 1, \ldots, n. \quad (34)$$

From (24), the return of the portfolio $x \in \mathbb{R}^n$ including the deposit becomes

$$\mu_r(x) + r_0 x_0 = \mu_r(x) + r_0 (1 - 1^T x) \quad (35)$$

where $1 = (1, \ldots, 1) \in \mathbb{R}^n$. 

---

**Table 5.** Steel–Dwass’s test of $f(x_b)$ in (26) for $\lambda = 0.1$.

| Group   | Rank | $p$-value |
|---------|------|-----------|
| CHM     | 24.8 | 0.031     |
| ST      | 36.2 | *         |
| CHM     | 25.7 | 0.090     |
| CHM-ST  | 35.2 | n.s.      |
| ST      | 35.2 | n.s.      |

**Table 6.** Steel–Dwass’s test of $f(x_b)$ in (26) for $\lambda = 0.9$.

| Group   | Rank | $p$-value |
|---------|------|-----------|
| CHM     | 45.5 | 0.000     |
| ST      | 15.5 | **        |
| CHM     | 45.5 | 0.000     |
| CHM-ST  | 15.5 | **        |
| CHM-ST  | 19.4 | 0.000     |
| ST      | 41.5 | **        |
where the risk aversion indicator \( \lambda \in [0, 1] \) is given by investors. The gradient of the objective function \( f(x) \) in (36) can be derived as

\[
\nabla f(x) = \lambda \frac{x C}{\sqrt{x^T C x}} - (1 - \lambda) (\mu - r_0 I).
\]

From constraints in (36), we define the activity of a solution \( x \in \mathbb{R}^n \) as

\[
\gamma(x) = \min \left\{ \min_j \{ |x_j| \}, \min_j \{ |1 - x_j| \}, |1 - \sum_{j=1}^n x_j| \right\}
\]

where the solution \( x \in \mathbb{R}^n \) is not always feasible one.

6.2. Experimental setup

Figure 11 illustrates the feasible region \( X \subseteq \mathbb{R}^n \) of the portfolio optimization problem in (36) for the case of \( n = 2 \). Black three dots denote vertexes. From the linear constraints of the portfolio optimization problem in (36), the feasible region \( X \subseteq \mathbb{R}^n \) can be given by a convex-hull of the following \( p \) vertexes \( \hat{x}_i \in \mathbb{R}^n \), \( p = n + 1 \):

\[
\begin{align*}
\hat{x}_1 &= (1, 0, 0, \ldots, 0, 0) \\
\hat{x}_2 &= (0, 1, 0, \ldots, 0, 0) \\
&\vdots \\
\hat{x}_n &= (0, 0, 0, \ldots, 0, 1) \\
\hat{x}_{n+1} &= (0, 0, 0, \ldots, 0, 0).
\end{align*}
\]

By using CHM, the portfolio optimization problem in (36) is transformed into an unconstrained optimization problem as shown in (12). Besides, the search space of EA is defined as \( S = [0, 1]^q \), \( q = n \). As stated above, JADE using the Feasibility Rule (FR) can also be applied to the portfolio optimization problem in (36). In the case of JADE with FR, the search space of EA is given as \( S = [0, 1]^n \). Furthermore, from the constraints in (36), the constraint violation in (21) is defined as

\[
\phi(x) = \max \left\{ 0, \sum_{j=1}^n |x_j - 1| \right\}.
\]

JADE is applied to the portfolio optimization problem in (36) by using the following three methods in which \( NP = 100 \) and \( NG = 300 \) are chosen for JADE.

1. CHM-based: \( z_k \in P_f \) is transformed into \( x_k \in X \) by using CHM.
2. FR-based: JADE is used with FR. Hence \( z_k = x_k \in X \) holds if \( \phi(z_k) = 0 \).
3. CHM-FR hybrid: A set of initial solutions \( X_0 \subseteq X \) is generated by using CHM. Let \( P_0 = X_0 \). After that, JADE is used with FR to evolve the population \( P_t \).

The data set of assets provided by OR-Library [29] is also used for an instance of the portfolio optimization problem in (36) where the number of assets is \( n = 31 \).

6.3. Experimental results of using JADE

Figure 12 shows the objective function values \( f(x_b) \) of the best solutions \( z_b \in P_t \) at generation \( t \) obtained by JADE using the three methods for the portfolio optimization problem in (36) with \( \lambda = 0.1 \). The results of Figure 12 are the average of 30 runs. Similarly, Figure 13 shows the values of \( f(x_b) \) in (36) obtained for \( \lambda = 0.9 \).

Figure 14 shows the constraint violation values \( \phi(z_b) \) of the best solutions \( z_b \in P_t \) obtained by FR-based method for the portfolio optimization problem in (36) with \( \lambda = 0.1 \) and \( \lambda = 0.9 \) respectively. From Figure 14, we can see that it will take many generations to find a feasible solution \( z_k \in P_t \), \( \phi(z_k) = 0 \) by FR-based method. In the same way with Table 3, Table 7 compares the best solutions \( x_b \in X \) obtained by JADE using three methods for the portfolio optimization problem in (36)
and Table 8, the location of the best solution \( x_b \in P \) obtained by three methods for the portfolio optimization problem in (36) with \( \lambda = 0.1 \) and \( \lambda = 0.9 \).

**Table 7.** Comparison of three best solutions \( x_b \in X \) by JADE for the portfolio optimization problem in (36) with \( \lambda = 0.1 \).

| \( \lambda = 0.1 \) | CHM | ST | CHM-FR |
|----------------------|-----|----|--------|
| \( f(x_b) \)        | \(-2.131 \times 10^{-2}\) | \(-1.967 \times 10^{-2}\) | \(-2.096 \times 10^{-2}\) |
| \( |\nabla f(x_b)|\) | \(1.000 \times 10^{-1}\) | \(9.677 \times 10^{-2}\) | \(9.982 \times 10^{-2}\) |
| \( \gamma(x_b) \)  | \(6.006 \times 10^{-16}\) | \(0.0\) | \(0.0\) |
| \( d(x_b) \)       | \(4.154 \times 10^{-1}\) | \(1.489 \times 10^{-1}\) | \(3.488 \times 10^{-1}\) |

**Table 8.** Comparison of three best solutions \( x_b \in X \) by JADE for the portfolio optimization problem in (36) with \( \lambda = 0.9 \).

| \( \lambda = 0.9 \) | CHM | ST | CHM-FR |
|----------------------|-----|----|--------|
| \( f(x_b) \)        | \(-9.999 \times 10^{-5}\) | \(-9.999 \times 10^{-5}\) | \(-9.999 \times 10^{-5}\) |
| \( |\nabla f(x_b)|\) | \(1.023 \times 10^{-1}\) | \(1.536 \times 10^{-1}\) | \(1.534 \times 10^{-1}\) |
| \( \gamma(x_b) \)  | \(0.0\) | \(0.0\) | \(0.0\) |
| \( d(x_b) \)       | \(7.134 \times 10^{-39}\) | \(8.395 \times 10^{-18}\) | \(6.176 \times 10^{-20}\) |

with \( \lambda = 0.1 \). Similarly, Table 8 compares the best solutions \( x_b \in X \) for \( \lambda = 0.9 \).

From the values of \( d(x_b) \), \( 0 \leq d(x_b) \leq 1 \) in Table 7 and Table 8, the location of the best solution \( x_b \in X \) for the portfolio optimization problem in (36) clearly depends on the value of \( \lambda \in [0,1] \). In the case of \( \lambda = 0.9 \), the best solutions \( x_b \in X \) exist at the origin of the feasible region \( X \subseteq \mathbb{R}^n \) because \( d(x_b) \approx 0 \) holds for them.

From the values of \( f(x_b) \) in Table 7, CHM-based method seems to be better than the other methods. On the other hand, we cannot confirm the difference between the best solutions \( x_b \in X \) obtained by JADE using the three methods in Table 8.

Table 9 shows the result of Steel–Dwass’s test about the value of \( f(x_b) \) in (36) obtained by JADE using the three methods for \( \lambda = 0.1 \). Similarly, Table 10 shows the result of Steel–Dwass’s test about the value of \( f(x_b) \) in (36) for \( \lambda = 0.9 \). From the results in Table 9 and Table 10, the proposed CHM-based method significantly outperforms the other methods regardless of the value of \( \lambda \in [0,1] \).

### 7. Long-short portfolios

#### 7.1. Problem formulation

We consider a long-short portfolio that includes the deposit for a single period. In other words, we can sell one asset and buy another. However, the total amount of transactions is regulated [31]. Therefore, the portfolio \( x \in \mathbb{R}^n \) is constrained as

\[
|x_1| + \cdots + |x_j| + \cdots + |x_n| \leq 1
\]

\[
-1 \leq x_j \leq 1, \quad j = 1, \ldots, n.
\]

From (35) and (41), the portfolio optimization problem is formulated as

\[
\begin{align*}
\min & \quad f(x) = \lambda \sigma_r(x) - (1 - \lambda) \\
\text{s.t.} & \quad |x_1| + \cdots + |x_j| + \cdots + |x_n| \leq 1 \quad (42)
\end{align*}
\]

where the risk aversion indicator \( \lambda \in [0,1] \) is given by investors.

The gradient of the objective function \( f(x) \) in (42) is also given by (37). From constraints in (42), we define the activity of a solution \( x \in \mathbb{R}^n \) as

\[
\gamma(x) = \min \left\{ \min_j |1 + x_j|, \min_j |1 - x_j|, \left| -1 - \sum_{j=1}^n x_j \right| \right\}
\]

(43)

where the solution \( x \in \mathbb{R}^n \) is not always feasible one.
In (21) is defined as from the constraints in (42), the constraint violation of the portfolio optimization problem in (42), Black four dots denote vertexes. From the linear constraints of the portfolio optimization problem in (42), the feasible region of the portfolio optimization problem in (42) for the case of "n" is defined as $S = [0, 1]^q$, $q = 2n - 1$. JADE using FR can also be applied to the portfolio optimization problem in (42). In the case of JADE with FR, the search space of EA is given as $S = [-1, 1]^n$. Furthermore, from the constraints in (42), the constraint violation in (21) is defined as

$$
\phi(x) = \max \left\{ 0, \sum_{j=1}^{n} |x_j| - 1 \right\}. \tag{45}
$$

JADE is applied to the portfolio optimization problem in (42) by using the above three methods, namely (1) CHM-based, (2) FR-based, and (3) CHM-FR hybrid. In the three methods, NP = 100 and NG = 300 are chosen for JADE.

The data set of assets provided by OR-Library [29] is also used for an instance of the portfolio optimization problem in (42) where the number of assets is $n = 31$.

### 7.3. Experimental results of using JADE

Figure 16 shows the objective function values $f(x_b)$ of the best solutions $x_b \in P_t$ at generation $t$ obtained by the three methods for the portfolio optimization problem in (42) with $\lambda = 0.1$. The results of Figure 16 are the average of 30 runs. Similarly, Figure 17 shows the values of $f(x_b)$ obtained by the three methods for $\lambda = 0.9$.

Figure 18 shows the constraint violation values $\phi(x_b)$ of the best solutions $x_b \in P_t$ obtained by FR-based method for the portfolio optimization problem in (42) with $\lambda = 0.1$ and $\lambda = 0.9$. Even though the dimension of its search space $S = [0, 1]^q$, $q = 2n - 1$ is very high, CHM-based method outperforms the three methods for the portfolio optimization problem in (42) with $\lambda = 0.1$. The results of Figure 16 are the average of 30 runs. Similarly, Figure 17 shows the values of $f(x_b)$ obtained by the three methods for $\lambda = 0.9$.

Figure 18 shows the constraint violation values $\phi(x_b)$ of the best solutions $x_b \in P_t$ obtained by FR-based method for the portfolio optimization problem in (42) with $\lambda = 0.1$ and $\lambda = 0.9$. Even though the dimension of its search space $S = [0, 1]^q$, $q = 2n - 1$ is very high, CHM-based method outperforms the three methods for the portfolio optimization problem in (42) with $\lambda = 0.1$. The results of Figure 16 are the average of 30 runs. Similarly, Figure 17 shows the values of $f(x_b)$ obtained by the three methods for $\lambda = 0.9$.

Figure 18 shows the constraint violation values $\phi(x_b)$ of the best solutions $x_b \in P_t$ obtained by FR-based method for the portfolio optimization problem in (42) with $\lambda = 0.1$ and $\lambda = 0.9$. Even though the dimension of its search space $S = [0, 1]^q$, $q = 2n - 1$ is very high, CHM-based method outperforms the three methods for the portfolio optimization problem in (42) with $\lambda = 0.1$. The results of Figure 16 are the average of 30 runs. Similarly, Figure 17 shows the values of $f(x_b)$ obtained by the three methods for $\lambda = 0.9$.

Table 13 shows the result of Steel–Dwass’s test about the value of $f(x_b)$ obtained for the portfolio optimization problem in (42) with $\lambda = 0.1$. Even though the dimension of its search space $S = [0, 1]^q$, $q = 2n - 1$ is very high, CHM-based method outperforms the three methods for the portfolio optimization problem in (42) with $\lambda = 0.1$. The results of Figure 16 are the average of 30 runs. Similarly, Figure 17 shows the values of $f(x_b)$ obtained by the three methods for $\lambda = 0.9$.

Table 13 shows the result of Steel–Dwass’s test about the value of $f(x_b)$ obtained for the portfolio optimization problem in (42) with $\lambda = 0.1$. Even though the dimension of its search space $S = [0, 1]^q$, $q = 2n - 1$ is very high, CHM-based method outperforms the three methods for the portfolio optimization problem in (42) with $\lambda = 0.1$. The results of Figure 16 are the average of 30 runs. Similarly, Figure 17 shows the values of $f(x_b)$ obtained by the three methods for $\lambda = 0.9$. Table 13 shows the result of Steel–Dwass’s test about the value of $f(x_b)$ obtained for the portfolio optimization problem in (42) with $\lambda = 0.1$.
Table 11. Comparison of three best solutions $x_9 \in X$ by JADE for the portfolio optimization problem in (42) with $\lambda = 0.1$.

| Group | $f(x_9)$ | $|V f(x_9)|$ | $|f(x_9)|$ | $d(x_9)$ |
|-------|---------|-------------|-------------|---------|
| CHM   | $-6.825 \times 10^{-12}$ | $9.777 \times 10^{-5}$ | $9.985 \times 10^{-5}$ | $3.394 \times 10^{-2}$ |
| FR    | $-1.217 \times 10^{-2}$  | $1.915 \times 10^{-2}$  | $2.064 \times 10^{-2}$  | $2.549 \times 10^{-1}$  |

Table 12. Comparison of three best solutions $x_9 \in X$ by JADE for the portfolio optimization problem in (42) with $\lambda = 0.9$.

| Group | $f(x_9)$ | $|V f(x_9)|$ | $|f(x_9)|$ | $d(x_9)$ |
|-------|---------|-------------|-------------|---------|
| CHM   | $1.575 \times 10^{-2}$ | $2.578 \times 10^{-3}$ | $2.684 \times 10^{-3}$ | $3.094 \times 10^{-2}$ |
| FR    | $1.667 \times 10^{-2}$ | $2.789 \times 10^{-3}$ | $2.895 \times 10^{-3}$ | $3.304 \times 10^{-2}$ |

Table 13. Steel–Dwass’s test of $f(x_9)$ in (42) for $\lambda = 0.1$.

| Group | Rank | $p$-value |
|-------|------|-----------|
| CHM   | 15.5 | 0.000     |
| FR    | 45.5 | **        |
| CHM   | 16.4 | 0.000     |
| CHM-FR| 44.6 | **        |
| CHM-FR| 15.5 | 0.007     |
| FR    | 45.5 | **        |

Table 14. Steel–Dwass’s test of $f(x_9)$ in (42) for $\lambda = 0.9$.

| Group | Rank | $p$-value |
|-------|------|-----------|
| CHM   | 45.5 | 0.000     |
| FR    | 15.5 | **        |
| CHM   | 45.5 | 0.000     |
| CHM-FR| 15.5 | **        |
| CHM-FR| 15.5 | 0.000     |
| FR    | 45.5 | **        |

Table 15. Dimension of the search space $S \subseteq N^q$. 

| Problem | $q$ | $n$ | $2n - 1$ |
|---------|----|----|--------|
| EF      | $n - 1$ | $n$ | $2n - 1$ |
| EF_D   |      |    |        |
| EF_LS  |      |    |        |

Figure 19. The diversity of the set of $x_9 \in X_0 \subseteq N^n$, $k = 1, \ldots, NP$ for the dimension $n$. The population size is chosen as $NP = 400$. The set of feasible solutions $x_9 \in X_0 \subseteq X$ is generated by using CHM.

8. Discussion

From the viewpoints of constraint setting, we discuss the characteristics of the three types of portfolio optimization problems, namely Efficient Frontier (EF) model in (26), EF model including Deposit (EF_D) in (36), and EF model of Long-Short portfolio (EF_LS) in (42). Table 15 shows the dimension of the search space $S = [0, 1]^q$ provided by CHM for each of the portfolio optimization problems with the $n$-dimensional feasible region $X \subseteq N^n$. The dimension of $S = [0, 1]^q$ depends on the number of vertexes $\hat{x}_k \in N^n$, $i = 1, \ldots, p$ used to defined the feasible region such as $q = p - 1$. From the dimension of $S = [0, 1]^q$ in Table 15, EF_LS seems to be the most difficult for EAs using CHM. Actually, the best solutions $x_9 \in X$ in Table 12 (EF_LS with $\lambda = 0.9$) have not fully satisfied the optimality condition: $|V f(x_9)| = 0$ or $|f(x_9)| = 0$.

By using CHM, a set of feasible solutions $x_k \in X_0 \subseteq X$, $k = 1, \ldots, NP$ are generated randomly for each of the three types of portfolio optimization problems. Figure 19 shows the diversity of the set of feasible solutions $x_9 \in X_0$ defined by (31). The population size is chosen as $NP = 400$. The horizontal axis of Figure 19 is the dimension of the feasible region $X \subseteq N^n$. The results in Figure 19 are the average of 100 runs. From Figure 19, we can see that the diversity of the set of $x_k \in X_0$ decreases in proportion to the dimension of the feasible region $X \subseteq N^n$. The diversity of the set of $x_k \in X_0$ also depends on the dimension of the search space $S \subseteq N^q$ in Table 15. From Figure 19, we can confirm that EF_LS is the most difficult problem for EAs.

Next, we discuss the issue of redundancy in the solution representation by CHM. As shown in the proof of Theorem 3.1, CHM $\tau : S \rightarrow X$ is composed of three different mappings such as $\tau = \tau_1 \circ \tau_2 \circ \tau_3$. The first mapping $\tau_1 : S \rightarrow Z$ denotes the sorting of the elements of $Z = (z_1, \ldots, z_q) \in N^q$. Thus the first mapping is surjective. There are $q!$ redundant representations of $z \in S$ for each $\hat{z} \in Z$. However, we can expect that there are no redundant vectors $z_k \in P$, in the population, or a set of some vectors generated stochastically in real $q$-space $S \subseteq N^q$. Let $A \subseteq N^p$ be a set of $a = (a_1, \ldots, a_p) \in N^p$ in (10). The second mapping $\tau_2 : Z \rightarrow A$ defined by (10) is bijection. Therefore, the probability that the same solution is generated from different vectors, namely $Pr(x_9 = x_1 | z_k \neq z_i)$, depends on the third mapping $\tau_3 : A \rightarrow X$ defined by (11). As...
a result, the probability is proportional to the difference in dimension between the two spaces $X \subseteq \mathbb{R}^n$ and $A \subseteq \mathbb{R}^p, p \geq n$.

Experimental results show that CHM enhances the performance of EA rather than conventional CHTs in many cases, but evaluating the influence of the representational redundancy of solutions on the search performance of EA is a future challenge.

9. Conclusion

A novel mapping-based Constraint-Handling Technique (CHT) called Convex-Hull Mapping (CHM) is proposed for EAs applied to constrained optimization problems. The search space of EA is given by a hypercube. On the other hand, we assume that the feasible region of the constrained optimization problem is given by a convex hull of multiple vertexes. The proposed CHM transforms the real vector $z \in \mathbb{R}^q$ in the search space $S = [0, 1]^q$ of EA into the solution $x \in \mathbb{R}^n$ in the feasible region $X \subseteq \mathbb{R}^n$.

The advantages of the proposed CHM are

- It provides a surjective mapping $\tau : S \rightarrow X$.
- It does not require additional parameters.
- It can be used directly for any EAs.
- It does not require any modifications for original EAs.
- It can also be combined with conventional CHTs.
- It can be calculated easily if all vertexes are given.

The performance of CHM is compared with conventional CHTs through three types of portfolio optimization problems. The results of Steel–Dwass’s test on the objective function values of the best solutions obtained by JADE show that the proposed CHM is significantly better than conventional CHTs in many cases. Furthermore, a hybrid method combining CHM with a conventional CHT outperforms the original CHT.

Since the constraints effectively handled by CMH are very common in many problem formulations of portfolio optimization problems, CHM has the potential to contribute to design powerful EAs for solving various portfolio optimization problems.

Unfortunately, even if all constraints are linear, the proposed CHM cannot be used for EAs which are applied to an optimization problem in which

- It is hard to obtain all vertexes of a convex hull.
- Vertexes are too many in comparison with variables.
- The feasible region forms a concave polyhedron.

To enumerate all vertexes of the feasible region, something like the simplex algorithm [32] for the linear programming may be used. Furthermore, if a feasible region defined by a concave polyhedron or many vertexes can be divided into several convex polyhedrons, CHM can be used for EAs to solve the constrained optimization problem. Specifically, CHM can be used to solve a set of unconstrained optimization problems which are defined, respectively, in each of the convex polyhedrons. Therefore, in future work, we would like to develop a divide-and-conquer method for CHM to deal with constrained optimization problems that have complex feasible regions.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Notes on contributors

Kiyoharu Tagawa

He received his Ph. D. degree from Kobe University, Japan, in 1997. He is a Professor in School of Science and Engineering, Kindai University, Japan since 2007. His current research interests include evolutionary algorithms, optimization methods under uncertainties, and their applications to real-world problems. He is a member of SICE, IEEJ, IPSJ, and IEEE.

Yukiko Orito

She received her Ph.D. degree from Tokyo Metropolitan Institute of Technology in 2003. She is an associate professor in department of economics, Hiroshima University, Japan. Her current research interests are in evolutionary algorithms and its applications to engineering, finance, and economics. She is a member of IEEJ, JSEC, IPSJ and IEEE.

ORCID

Kiyoharu Tagawa http://orcid.org/0000-0002-3485-1730

References

[1] Mezura-Montes E, Coello Coello CA. Constraint-handling in nature-inspired numerical optimization: past, present and future. Swarm Evol Comput. 2011;1:173–194.

[2] Koziel S, Michalewicz Z. Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization. Evol Comput. 1999;7(1):19–44.

[3] Tagawa K. A mapping-based convex constraint-handling technique for evolutionary algorithms. Proc. of the SICE Annual Conference 2021; 2021. p. 808–813.

[4] Storn R, Price K. Differential evolution – a simple and efficient heuristic for global optimization over continuous space. J Global Optim. 1997;11(4):341–359.

[5] Price K, Storn RM, Lampinen JA. Differential evolution: a practical approach to global optimization. Berlin, Heidelberg: Springer; 2005.

[6] Das S, Suganthan PN. Differential evolution: a survey of the state-of-the-art. IEEE Trans Evol Comput. 2011;15(1):4–31.

[7] Wolper DH, Macready WG. No free lunch theorems for optimization. IEEE Trans Evol Comput. 1997;1(1):67–82.
Proof of Theorem 3.1

Let $A \subseteq \mathbb{R}^p$ be a set of vectors $a = (a_1, \ldots, a_p) \in \mathbb{R}^p$. Each $a \in \mathbb{R}^p$ consists of the coefficients $a_i \in \mathbb{R}$, $i = 1, \ldots, p$ in (7). Let $X \subseteq \mathbb{R}^n$ be the feasible region of the constrained optimization problem in (5). Since every solution $x \in X$ can be represented by a convex combination of all vertexes $\tilde{x}_i \in \mathbb{R}^n$, $i = 1, \ldots, p$ as shown in (7), we can define a surjective mapping $t_3 : A \rightarrow X$ as follows:

$$\forall x \in X, \exists a \in A; \ x = t_3(a) \quad (A1)$$

where $t_3 : A \rightarrow X$ is also a bijective mapping if $p = n$.

Let $Z \subseteq S = [0, 1]^q$ be a set of sorted vectors $\tilde{z} = (\tilde{z}_1, \ldots, \tilde{z}_q) \in \mathbb{R}^q$ as shown in (9). Then, the affine transformation given by (10) provides a bijective mapping $t_2 : Z \rightarrow A$ such as $a = t_2(\tilde{z})$. Furthermore, the sorting of elements $z_j \in \mathbb{R}$, $j = 1, \ldots, q$ of $z \in S$ provides a surjective mapping $t_1 : S \rightarrow Z$ as follows:

$$\forall \tilde{z} \in Z, \exists z \in S; \ \tilde{z} = t_1(z). \quad (A2)$$

Since the proposed CHM $\tau : S \rightarrow X$ can be composed as $\tau = t_3 \circ t_2 \circ t_1$, CHM performs a surjective mapping as follows:

$$\forall x \in X, \exists z \in S; \ x = \tau(z). \quad (A3)$$