An Approach to Chance Constrained Problems using Truncated Halton Sequence and Differential Evolution With Application to Flood Control Planning

KIYOHARU TAGAWA
Kindai University
School of Science and Engineering
Higashi-Osaka 577-8502
JAPAN
tagawa@info.kindai.ac.jp

Abstract: This paper proposes a new optimization method for solving Chance Constrained Problems (CCPs). Specifically, instead of the conventional Monte Carlo simulation based on random sampling, Truncated Halton Sequence (THS) is used to evaluate the probabilistic constraints in CCP. Then a group-based adaptive differential evolution called JADE2G is combined with THS and used to solve CCP efficiently. Actually, there are two types of CCPs, namely Joint CCP (JCCP) and Separate CCP (SCCP). Even though the proposed optimization method is applicable to both JCCP and SCCP, it is demonstrated through the flood control planning formulated as JCCP.

Key–Words: Chance constrained problem, Differential evolution, Stochastic programming
Received: December 22, 2018. Revised: January 19, 2019. Accepted:March 7, 2019.
Published: April 30, 2019

1 Introduction

In real-world optimization problems, a wide range of uncertainties have to be taken into account. Thus optimization problems under uncertainties have been studied for many years. Generally speaking, there are two problem formulations for handling uncertainties in optimization problems. The first one is the deterministic optimization problem [1]. The second one is the stochastic optimization problem [2].

Robust optimization problem is a deterministic formulation. Since robust optimization problem always considers the worst-case performance under uncertainties [1], the overestimation of uncertainties may lead to a conservative decision in practice.

Chance Constrained Problem (CCP) [2] is one of the possible formulations of the stochastic optimization problem. CCP is also referred to as probabilistic constrained problem [3]. Since the balance between optimality and reliability can be taken with a given probability, or a sufficiency level, a lot of real-world applications have been formulated as CCP [3–6].

CCP has been studied in the field of stochastic programming [2]. In the stochastic programming, the optimization methods of nonlinear programming are used to solve CCP. These optimization methods assume that the functions in CCP are differentiable and convex. Furthermore, a time-consuming Monte Carlo simulation based on random sampling is needed to evaluate the feasibility of the solutions of CCP. Even though Evolutionary Algorithms (EAs) have been reported recently to solve CCPs [4,7,8], they also spend long times for Monte Carlo simulations.

Actually, there are two types of CCPs, namely Joint CCP (JCCP) and Separate CCP (SCCP) [2]. In our previous study [9,10], Weighted Empirical Cumulative Distribution Function (W_ECDF) was proposed to evaluate the feasibility of the solutions of SCCP. For solving SCCP, Differential Evolution (DE) [11] was combined with W_ECDF. However, the previous optimization method was only applicable to SCCP. For solving JCCP with W_ECDF, the joint probability of JCCP has to be approximated by using Bonferroni inequality [10], which includes a large error.

In this paper, we propose a new optimization method to solve CCP without using the time-consuming Monte Carlo simulation based on random sampling. By using the empirical probability based on Truncated Halton Sequence (THS) [12], i.e. a novel sampling technique, CCP is transformed to a relaxation problem. Furthermore, for solving the relaxation problem efficiently, a new Adaptive DE (ADE) called on JADE2G is used. The new optimization method with THS is applicable to both JCCP and SCCP. We demonstrate the proposed optimization method with THS through a real-world application formulated as JCCP, namely the flood control planning to make an economical plan to protect an urban area from the flood damage caused by torrential rain [10].

The remainder of this paper is organized as follows. Section 2 describes two types of CCPs, namely JCCP and SCCP. Section 3 explains THS. Section 4 proposes a new optimization method for solving CCP, which is based on a new ADE combined with THS. Section 5 formulates the flood control planning as JCCP. Section 6 shows the results of numerical experiments. Finally, Section 7 concludes this paper.

2 Problem Formulation

The uncertainties included in optimization problems are represented by random variables $\boldsymbol{\xi} \in \boldsymbol{\Omega}$. Symbols used to describe problems are defined below.

- Decision variable $\boldsymbol{x} = (x_1, \cdots, x_D) \in \boldsymbol{X} \subseteq \Re^D$.
- Random variable $\boldsymbol{\xi} = (\xi_1, \cdots, \xi_K) \in \boldsymbol{\Omega} \subseteq \Re^K$.
- Measurable function $g_m : \boldsymbol{X} \times \boldsymbol{\Omega} \to \Re, \;\; m = 1, \cdots, M$.
- Objective function $g_0 : \boldsymbol{X} \to \Re$.
- Sufficiency level (Probability) $\alpha \in (0, 1)$.
- $\Pr(A)$ is occurrence probability of event A.

2.1 Robust Optimization Problem

Optimization problem with $M \geq 1$ constraints and one objective to be minimized is written as

$$\left[\begin{array}{ll} \displaystyle\min_{\boldsymbol{x} \in \boldsymbol{X}} & g_0(\boldsymbol{x}) \\ \text{sub. to} & g_m(\boldsymbol{x}) \leq 0, \; m = 1, \cdots, M \end{array} \right. \tag{1}$$

where a solution $\boldsymbol{x} \in \boldsymbol{X}$ is called feasible if it satisfies all constraints. Otherwise, $\boldsymbol{x} \in \boldsymbol{X}$ is called infeasible.

Uncertainties are given by a vector of random variables $\boldsymbol{\xi} \in \boldsymbol{\Omega}$ with support $\boldsymbol{\Omega} \subseteq \Re^K$. Thereby, the robust optimization problem [1] is defined as

$$\left[\begin{array}{ll} \displaystyle\min_{\boldsymbol{x} \in \boldsymbol{X}} & g_0(\boldsymbol{x}) \\ \text{sub. to} & \forall \boldsymbol{\xi} \in \boldsymbol{\Omega} : g_m(\boldsymbol{x}, \boldsymbol{\xi}) \leq 0, \\ & m = 1, \cdots, M. \end{array} \right. \tag{2}$$

Since the feasible solution $\boldsymbol{x} \in \boldsymbol{X}$ of robust optimization problem in (2) has to satisfy all constraints for every $\boldsymbol{\xi} \in \boldsymbol{\Omega}$, it seems to be too conservative from a practical engineering perspective.

2.2 Chance Constrained Problem (CCP)

By introducing a sufficiency level $\alpha \in (0, 1)$ into the constraints in (2), CCP reduces the conservativism of robust optimization problem. As stated above, there are two types of formulations for CCP.

Firstly, JCCP is formulated as

$$\left[\begin{array}{ll} \displaystyle\min_{\boldsymbol{x} \in \boldsymbol{X}} & g_0(\boldsymbol{x}) \\ \text{sub. to} & h(\boldsymbol{x}, \boldsymbol{\Omega}) \geq \alpha \end{array} \right. \tag{3}$$

where the probability $h(\boldsymbol{x}, \boldsymbol{\Omega})$ is defined as

$$h(\boldsymbol{x}, \boldsymbol{\Omega}) = \Pr\left(\begin{array}{l} \forall \boldsymbol{\xi} \in \boldsymbol{\Omega} : g_m(\boldsymbol{x}, \boldsymbol{\xi}) \leq 0, \\ m = 1, \cdots, M \end{array} \right). \tag{4}$$

Secondly, SCCP is formulated as

$$\left[\begin{array}{ll} \displaystyle\min_{\boldsymbol{x} \in \boldsymbol{X}} & g_0(\boldsymbol{x}) \\ \text{sub. to} & h_m(\boldsymbol{x}, \boldsymbol{\Omega}) \geq \alpha, \; m = 1, \cdots, M \end{array} \right. \tag{5}$$

where the probability $h_m(\boldsymbol{x}, \boldsymbol{\Omega})$ is defined as

$$h_m(\boldsymbol{x}, \boldsymbol{\Omega}) = \Pr(\forall \boldsymbol{\xi} \in \boldsymbol{\Omega} : g_m(\boldsymbol{x}, \boldsymbol{\xi}) \leq 0). \tag{6}$$

3 Probability Estimation

For evaluating the chance constraint included in CCP empirically, the indicator function is defined as

$$\mathbb{1}(\mathcal{A}) = \begin{cases} 1 & \text{if} \quad \text{event } \mathcal{A} \text{ is true} \\ 0 & \text{if} \quad \text{event } \mathcal{A} \text{ is false.} \end{cases} \tag{7}$$

3.1 Random Sampling (RS)

CCP is usually difficult to solve because the time-consuming Monte Carlo simulation is required to evaluate the empirical probability that the chance constraint of CCP is satisfied with a solution $\boldsymbol{x} \in \boldsymbol{X}$ by using a huge number of samples of $\boldsymbol{\xi} \in \boldsymbol{\Omega}$.

In the conventional Monte Carlo simulation [2], a set of samples $\boldsymbol{\xi}^n \in \boldsymbol{\Xi} \subseteq \boldsymbol{\Omega}$, $n = 1, \cdots, N$ is selected randomly. For each of the samples $\boldsymbol{\xi}^n \in \boldsymbol{\Xi}$, the function value $g_m(\boldsymbol{x}, \boldsymbol{\xi}^n)$ is evaluated. Thereby, the empirical probability $h(\boldsymbol{x}, \boldsymbol{\Xi})$ that approximates the probability $h(\boldsymbol{x}, \boldsymbol{\Omega})$ in (4) is calculated as

$$\begin{aligned} & h(\boldsymbol{x}, \boldsymbol{\Xi}) \\ & = \Pr\left(\begin{array}{l} \forall \boldsymbol{\xi}^n \in \boldsymbol{\Xi} : g_m(\boldsymbol{x}, \boldsymbol{\xi}^n) \leq 0, \\ m = 1, \cdots, M \end{array} \right) \\ & = \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}\left(\begin{array}{l} g_m(\boldsymbol{x}, \boldsymbol{\xi}^n) \leq 0, \\ m = 1, \cdots, M \end{array} \right). \end{aligned} \tag{8}$$

From the law of large number [13], the empirical probability should be close to the true value as

$$\lim_{N \to \infty} h(\boldsymbol{x}, \boldsymbol{\Xi}) \to h(\boldsymbol{x}, \boldsymbol{\Omega}). \tag{9}$$

As a drawback of the empirical probability based on Random Sampling (RS), the number of samples $\boldsymbol{\xi}^n \in \boldsymbol{\Xi}$ must be sufficiently large to approximate the true probability accurately. That is because few samples are obtained from the tail part of support $\boldsymbol{\Omega}$.

3.2 Weighted Halton Sequence (WHS)

In order to take samples from support $\boldsymbol{\Omega}$ uniformly, we use Halton Sequence (HS) [14], i.e. a low-discrepancy sequence, instead of the random samples $\boldsymbol{\xi}^n \in \boldsymbol{\Xi}$.

HS generates points $\boldsymbol{\theta}^n \in \widehat{\boldsymbol{\Theta}}$, $n = 1, \cdots, N$ uniformly in a hyper-cube $\widehat{\boldsymbol{\Theta}} \subseteq \boldsymbol{\Omega}$. We suppose that the Probability Density Function (PDF) of $\boldsymbol{\xi} \in \boldsymbol{\Omega}$ is known. Let $f : \boldsymbol{\Omega} \to [0, \infty)$ be the PDF of $\boldsymbol{\xi} \in \boldsymbol{\Omega}$. Each point $\boldsymbol{\theta}^n \in \widehat{\boldsymbol{\Theta}}$ is weighted by the value of PDF such as $f(\boldsymbol{\theta}^n)$. By using Weighted HS (WHS) [9], the empirical probability $h(\boldsymbol{x}, \widehat{\boldsymbol{\Theta}})$ is calculated as

$$h(\boldsymbol{x}, \widehat{\boldsymbol{\Theta}}) = \frac{1}{W} \sum_{n=1}^{N} f(\boldsymbol{\theta}^n)\, \mathbb{1}\left(\begin{array}{l} g_m(\boldsymbol{x}, \boldsymbol{\theta}^n) \leq 0, \\ m = 1, \cdots, M \end{array} \right) \tag{10}$$

where $W = f(\boldsymbol{\theta}^1) + \cdots + f(\boldsymbol{\theta}^n) + \cdots + f(\boldsymbol{\theta}^N)$.

As a drawback of the empirical probability based on WHS, a set of points $\boldsymbol{\theta}^n \in \widehat{\boldsymbol{\Theta}}$ includes many futile points weighted by a small value as $f(\boldsymbol{\theta}^n) \approx 0$.

3.3 Truncated Halton Sequence (THS)

Truncated HS (THS) $\boldsymbol{\theta}^n \in \boldsymbol{\Theta} \subseteq \widehat{\boldsymbol{\Theta}}$, $n = 1, \cdots N$ is a subset of WHS. THS is defined as

$$\boldsymbol{\Theta} = \{\boldsymbol{\theta}^n \in \widehat{\boldsymbol{\Theta}} \mid f(\boldsymbol{\theta}^n) \geq f_{\min}\} \tag{11}$$

where the minimum PDF value $f_{\min}$ is a parameter.

By using the proposed THS [12], the empirical probability $h(\boldsymbol{x}, \boldsymbol{\Theta})$ is calculated as

$$h(\boldsymbol{x}, \boldsymbol{\Theta}) = \frac{1}{W} \sum_{n=1}^{N} f(\boldsymbol{\theta}^n)\, \mathbb{1}\left(\begin{array}{l} g_m(\boldsymbol{x}, \boldsymbol{\theta}^n) \leq 0, \\ m = 1, \cdots, M \end{array} \right) \tag{12}$$

where $W = f(\boldsymbol{\theta}^1) + \cdots + f(\boldsymbol{\theta}^n) + \cdots + f(\boldsymbol{\theta}^N)$.

3.4 Examples of RS, WHS, and THS

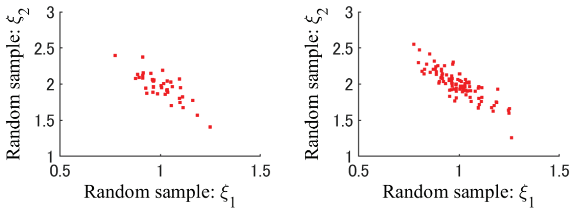
The random variables $\boldsymbol{\xi} = (\xi_1, \xi_2) \in \boldsymbol{\Omega} \subseteq \Re^2$ are supposed to be following normal distributions as

$$\left\{ \begin{array}{l} \xi_1 \sim \mathcal{N}(\mu_1, \sigma_1^2) = \mathcal{N}(1, 0.1^2) \\ \xi_2 \sim \mathcal{N}(\mu_2, \sigma_2^2) = \mathcal{N}(2, 0.2^2). \end{array} \right. \tag{13}$$

The correlation matrix of $\boldsymbol{\xi} \in \boldsymbol{\Omega}$ is given as

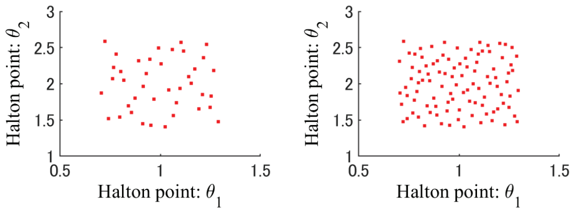
$$\boldsymbol{R} = \left(\begin{array}{cc} 1 & \rho_{12} \\ \rho_{21} & 1 \end{array} \right) = \left(\begin{array}{cc} 1 & -0.8 \\ -0.8 & 1 \end{array} \right) \tag{14}$$

where ρ_{ij} is a correlation coefficient.



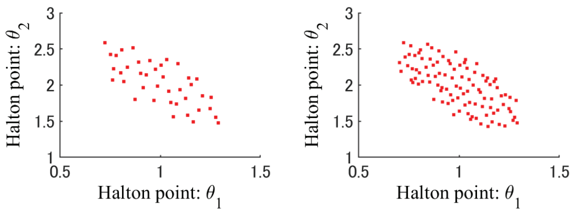
(a) $N = 40$ (b) $N = 100$

Figure 1: Random samples $\boldsymbol{\xi}^n \in \boldsymbol{\Xi} \subseteq \boldsymbol{\Omega}$



(a) $N = 40$ (b) $N = 100$

Figure 2: Weighted Halton sequence $\boldsymbol{\theta}^n \in \widehat{\boldsymbol{\Theta}}$



(a) $N = 40$ (b) $N = 100$

Figure 3: Truncated Halton sequence $\boldsymbol{\theta}^n \in \boldsymbol{\Theta}$

Fig. 1 shows $\boldsymbol{\xi}^n \in \boldsymbol{\Xi}$, $n = 1, \cdots, N$ generated by RS from (13). Fig. 2 shows $\boldsymbol{\theta}^n \in \widehat{\boldsymbol{\Theta}}$ generated as WHS. Fig. 3 shows $\boldsymbol{\theta}^n \in \boldsymbol{\Theta}$ generated as THS with $f_{\min} = 0.01$ in (11). From Fig. 1 and Fig. 3, we can confirm that THS covers a wider range than RS.

Let us consider a stochastic function defined as

$$g(\boldsymbol{x}, \boldsymbol{\xi}) = x_1\, \xi_1 + x_2\, \xi_2 + b. \tag{15}$$

From the linearity of the normal distribution, the value of $g(\boldsymbol{x}, \boldsymbol{\xi})$ also follows a normal distribution as

$$\left\{ \begin{array}{rcl} g(\boldsymbol{x}, \boldsymbol{\xi}) & \sim & \mathcal{N}(\mu_g(\boldsymbol{x}), \sigma_g^2(\boldsymbol{x})) \\ \mu_g(\boldsymbol{x}) & = & \mu_1\, x_1 + \mu_2\, x_2 + b \\ \sigma_g^2(\boldsymbol{x}) & = & \sigma_1^2\, x_1^2 + \sigma_2^2\, x_2^2 \\ & & +2\, \rho_{12}\, \sigma_1\, \sigma_2\, x_1\, x_2 \end{array} \right. \tag{16}$$

where ρ_{12} is the correlation coefficient in (14).

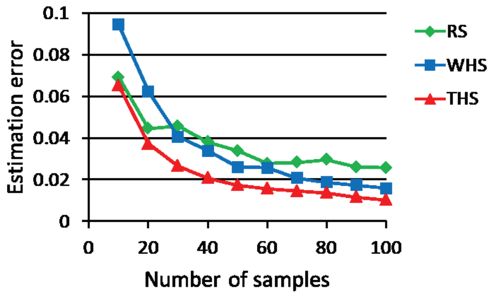


Figure 4: Estimation error for sample size

From (16), we can derive the probability as

$$\begin{aligned}&\Pr(\forall\, \boldsymbol{\xi} \in \boldsymbol{\Omega} : \; g(\boldsymbol{x},\, \boldsymbol{\xi}) \leq 0) \\ &= \Pr\left(\frac{g(\boldsymbol{x},\, \boldsymbol{\xi}) - \mu_g(\boldsymbol{x})}{\sigma_g(\boldsymbol{x})} \leq \frac{-\mu_g(\boldsymbol{x})}{\sigma_g(\boldsymbol{x})} \right) \\ &= \Phi\left(\frac{-\mu_g(\boldsymbol{x})}{\sigma_g(\boldsymbol{x})} \right)\end{aligned} \tag{17}$$

where Φ is the Cumulative Distribution Function (CDF) of the standard normal distribution: $\mathcal{N}(0,\, 1)$.

We choose a solution $\hat{\boldsymbol{x}} = (1,\, 1)$ and $b = -3.172$ for the function in (15). From (17), we can obtain the true probability such as $\Pr(g(\hat{\boldsymbol{x}},\, \boldsymbol{\xi}) \leq 0) \approx 0.900$. In order to estimate the true probability, we evaluate the empirical probabilities with RS, WHS, and THS.

Fig. 4 compares the estimation errors between RS, WHS, and THS. The estimation errors in Fig. 4 are averaged over 100 runs. Actually, the original HS [14] is generated deterministically. Thus we use the randomized HS [15] for WHS and THS. From Fig. 4, we can confirm that the probability can be estimated most correctly by using THS. Therefore, the proposed THS is better than the other sampling techniques.

4 Proposed Approach to CCP

4.1 CCP formulation through THS

By using the empirical probability in (12) that is based on THS $\boldsymbol{\theta}^n \in \boldsymbol{\Theta}$, $n = 1, \cdots, N$ and a correction level $\beta \geq \alpha$, JCCP in (3) is reformed as

$$\left[\begin{array}{ll} \min\limits_{\boldsymbol{x} \in \boldsymbol{X}} & g_0(\boldsymbol{x}) \\ \text{sub. to} & h(\boldsymbol{x},\, \boldsymbol{\Theta}) \geq \beta. \end{array} \right. \tag{18}$$

The correction level is initialized as $\beta = \alpha$ and regulated through the optimization described later.

Similarly, SCPP in (5) is reformed as

$$\left[\begin{array}{ll} \min\limits_{\boldsymbol{x} \in \boldsymbol{X}} & g_0(\boldsymbol{x}) \\ \text{sub. to} & h_m(\boldsymbol{x},\, \boldsymbol{\Theta}) \geq \beta,\; m = 1, \cdots, M \end{array} \right. \tag{19}$$

where the empirical probability is defined as

$$\begin{aligned}&h_m(\boldsymbol{x},\, \boldsymbol{\Theta}) \\ &= \frac{1}{W} \sum_{n=1}^{N} f(\boldsymbol{\theta}^n)\; \mathbb{1}(g_m(\boldsymbol{x},\, \boldsymbol{\theta}^n) \leq 0).\end{aligned} \tag{20}$$

4.2 Adaptive Differential Evolution

DE has been proven to be one of the most powerful global numerical optimization methods [16]. The performance of DE depends on control parameters, namely the scale factor and the crossover rate [11]. Thus, various Adaptive DEs (ADEs) have been reported with the adjusting techniques of these control parameters [17,18]. One of the most successful ADEs is JADE [18]. However, as well as DE [11], JADE can't handle constrained optimization problems.

ADEGL [19] is an extended JADE which is also contrived to solve unconstrained optimization problems. In ADEGL, the population is divided into multiple groups according to the ranks of their objective values. The groups have their own probability distributions to adjust the control parameters. It has been shown that ADEGL outperforms JADE [19].

Inspired by ADEGL, we have proposed a new ADE for solving constrained optimization problems, which is called JADE based on 2 Groups (JADE2G) [20]. In JADE2G, the population is also divided into two groups, namely feasible ones and infeasible ones. The two groups have their own probability distributions to adjust the control parameters respectively. The procedure of JADE2G is detailed below.

4.2.1 Initialization

At each generation t, JADE2G has a set of solutions $\boldsymbol{x}_i \in \boldsymbol{P}_t$, $i = 1, \cdots, N_P$ called population. Each solution $\boldsymbol{x}_i \in \boldsymbol{P}_t$ is a real vector represented as

$$\boldsymbol{x}_i = (x_{1,i}, \cdots, x_{j,i}, \cdots, x_{D,i}) \in \Re^D \tag{21}$$

where $\underline{x}_j \leq x_{j,i} \leq \overline{x}_j$, $j = 1, \cdots, D$.

By using a uniform distribution, a set of initial solution $\boldsymbol{x}_i \in \boldsymbol{P}_0 \subseteq \boldsymbol{X}$, $i = 1, \cdots, N_P$ of JADE2G is generated randomly as

$$x_{j,i} \sim \mathcal{U}(\underline{x}_j,\, \overline{x}_j),\; j = 1, \cdots, D \tag{22}$$

where the population size N_P is a fixed parameter.

4.2.2 Constraint-Handling

Even though the original versions of EAs including JADE and ADEGL can't deal with the constraints of optimization problems, a number of Constraint Handling Techniques (CHTs) have been reported [21].

The feasibility rule [22] is one of the most widely used CHTs because of its simplicity and efficiency. Thus we also employ the feasibility rule for JADE2G.

First of all, the constraint violation $\phi(\boldsymbol{x}_i)$ of JCCP in (18) is defined for a solution $\boldsymbol{x}_i \in \boldsymbol{P}_t$ as

$$\phi(\boldsymbol{x}_i) = \max\{\beta - h(\boldsymbol{x}, \boldsymbol{\Theta}), 0\}. \tag{23}$$

Similarly, the constraint violation $\phi(\boldsymbol{x}_i)$ of SCCP in (19) is defined for a solution $\boldsymbol{x}_i \in \boldsymbol{P}_t$ as

$$\phi(\boldsymbol{x}_i) = \sum_{m=1}^{M} \max\{\beta - h_m(\boldsymbol{x}, \boldsymbol{\Theta}), 0\}. \tag{24}$$

From the constraint violation, the feasibility of a solution $\boldsymbol{x}_i \in \boldsymbol{P}_t$ can be discriminated as

$$\begin{pmatrix} \phi(\boldsymbol{x}_i) = 0 & \text{if } \boldsymbol{x}_i \in \boldsymbol{P}_t \text{ is feasible} \\ \phi(\boldsymbol{x}_i) > 0 & \text{if } \boldsymbol{x}_i \in \boldsymbol{P}_t \text{ is infeasible.} \end{pmatrix} \tag{25}$$

4.2.3 Mutation

Each solution $\boldsymbol{x}_i \in \boldsymbol{P}_t$ is assigned to a target vector in turn. Then the mutation operator creates a mutation vector $\boldsymbol{v}_i \in \boldsymbol{X}$ from the target vector $\boldsymbol{x}_i \in \boldsymbol{P}_t$.

By using either of two Cauchy distributions, the scale factor $F_i \in [0, 1]$ of the target vector $\boldsymbol{x}_i \in \boldsymbol{P}_t$ is generated in the same way with JADE [18] as

$$F_i \sim \begin{cases} \mathcal{C}(\mu_{F1}, 0.1) & \text{if } \phi(\boldsymbol{x}_i) = 0 \\ \mathcal{C}(\mu_{F2}, 0.1) & \text{if } \phi(\boldsymbol{x}_i) > 0 \end{cases} \tag{26}$$

where $F_i \in \Re$ is truncated to be 1 if $F_i > 1$ holds or regenerated from (26) if $F_i < 0$ holds.

A solution $\boldsymbol{x}_p \in \boldsymbol{P}_t$ is selected randomly from the top $100p\%$ solutions. Besides, two solutions $\boldsymbol{x}_{r1}$ and $\boldsymbol{x}_{r2}$, $r1 \neq r2 \neq i$ are also selected randomly from the population $\boldsymbol{P}_t$. Thereby, the mutation operator called "DE/current-to-$p$best/1" [18] is used to make a new mutation vector $\boldsymbol{v}_i \in \Re^D$ with $F_i \in [0, 1]$ as

$$\boldsymbol{v}_i = \boldsymbol{x}_i + F_i\,(\boldsymbol{x}_p - \boldsymbol{x}_i) + F_i\,(\boldsymbol{x}_{r1} - \boldsymbol{x}_{r2}) \tag{27}$$

where the outside element $v_{j,i} \in \Re$ is corrected as

$$v_{j,i} = \begin{cases} (\underline{x}_j + x_{j,i})/2 & \text{if } v_{j,i} < \underline{x}_j \\ (\overline{x}_j + x_{j,i})/2 & \text{if } v_{j,i} > \overline{x}_j. \end{cases} \tag{28}$$

4.2.4 Crossover

By using either of two normal distributions, the crossover rate $CR_i \in [0, 1]$ is generated in the same way with JADE for the target vector $\boldsymbol{x}_i \in \boldsymbol{P}_t$ as

$$CR_i \sim \begin{cases} \mathcal{N}(\mu_{CR1}, 0.1^2) & \text{if } \phi(\boldsymbol{x}_i) = 0 \\ \mathcal{N}(\mu_{CR2}, 0.1^2) & \text{if } \phi(\boldsymbol{x}_i) > 0 \end{cases} \tag{29}$$

where $CR_i \in \Re$ is truncated to $[0, 1]$.

According to $CR_i \in [0, 1]$ in (29), the binomial crossover makes a trial vector $\boldsymbol{z}_i \in \boldsymbol{X}$ by merging the target vector $\boldsymbol{x}_i$ and the mutation vector $\boldsymbol{v}_i$ as

$$z_{j,i} = \begin{cases} v_{j,i} & \text{if } \mathrm{rand}_j \leq CR_i \ \vee \ j = j_r \\ x_{j,i} & \text{otherwise} \end{cases} \tag{30}$$

where $\mathrm{rand}_j \in [0, 1]$ is a uniformly distributed random value. The subscript $j_r \in [0, D]$ is also selected randomly, which ensures that the newborn $\boldsymbol{z}_i \in \boldsymbol{X}$ differs from the existing solution $\boldsymbol{x}_i \in \boldsymbol{P}_t$ [11].

4.2.5 Selection

Each trial vector $\boldsymbol{z}_i \in \boldsymbol{X}$, $i = 1, \cdots, N_P$ is compared with the corresponding target vector $\boldsymbol{x}_i \in \boldsymbol{P}_t$. If the following condition is satisfied, $\boldsymbol{z}_i \in \boldsymbol{X}$ is discarded immediately and $\boldsymbol{x}_i \in \boldsymbol{P}_t$ is added to $\boldsymbol{P}_{t+1}$.

$$(\phi(\boldsymbol{x}_i) = 0) \wedge (g_0(\boldsymbol{x}_i) < g_0(\boldsymbol{z}_i)) \tag{31}$$

Since the condition in (31) doesn't require the evaluation of $\phi(\boldsymbol{z}_i)$ based on the empirical probability, this technique is very efficient. Only when the condition in (31) is not satisfied, $\phi(\boldsymbol{z}_i)$ is evaluated. Then if either of the following conditions is satisfied, the trial vector $\boldsymbol{z}_i \in \boldsymbol{X}$ is added to $\boldsymbol{P}_{t+1}$.

$$\begin{pmatrix} \phi(\boldsymbol{z}_i) < \phi(\boldsymbol{x}_i) \\ (\phi(\boldsymbol{z}_i) = \phi(\boldsymbol{x}_i)) \wedge (g_0(\boldsymbol{z}_i) \leq g_0(\boldsymbol{x}_i)) \end{pmatrix} \tag{32}$$

4.2.6 Parameter Adaptation

At the end of each generation t, the locations μ_{F1} and μ_{F2} of Cauchy distributions in (26) are updated in each of the two groups as

$$\begin{pmatrix} \mu_{F1} = (1-c)\,\mu_{F1} + c\,S_{F1}^2/S_{F1} \\ \mu_{F2} = (1-c)\,\mu_{F2} + c\,S_{F2}^2/S_{F2} \end{pmatrix} \tag{33}$$

where S_{Fk} and S_{Fk}^2, $k = 1, 2$ are the sums of F_i and F_i^2 in success cases. The success case means that the trial vector $\boldsymbol{z}_i \in \boldsymbol{X}$ generated with F_i has been better than the target vector $\boldsymbol{x}_i \in \boldsymbol{P}_t$ in each group. The recommended value $c = 0.1$ [18] is used in (33).

At the end of each generation t, the means μ_{CR1} and μ_{CR2} in (29) are also updated respectively as

$$\begin{pmatrix} \mu_{CR1} = (1-c)\,\mu_{CR1} + c\,S_{CR1}/S_{NS1} \\ \mu_{CR2} = (1-c)\,\mu_{CR2} + c\,S_{CR2}/S_{NS2} \end{pmatrix} \tag{34}$$

where S_{NSk}, $k = 1, 2$ denote the numbers of success cases. S_{CRk}, $k = 1, 2$ denote the sums of CR_i in success cases. The value $c = 0.1$ is used in (34).

4.3 Algorithm of JADE2G

Generally speaking, JADE2G applied to CCP will encounter the following three stages in the process of search: 1) the population contains infeasible solutions only; 2) the population consists of both feasible and infeasible solutions; 3) the population contains feasible solutions only. The constraint violation $\phi(\boldsymbol{x}_i)$ is minimized in the first stage, while the objective function $g_0(\boldsymbol{x}_i)$ is minimized in the final stage. As a result, the control parameters F_i and CR_i have to be changed drastically depending on the stages. From (33) and (34), JADE2G enables the quick change of them.

The maximum number of generations N_T is used as the termination condition of JADE2G. Thereby, the algorithm of JADE2G is described as follows.

Step 1: Initialize the population $\boldsymbol{P}_0 \subseteq \boldsymbol{X}$. $t := 0$.

Step 2: Set $\mu_{F1} := 0.5$ and $\mu_{F2} := 0.8$ in (26).

Step 3: Set $\mu_{CR1} := 0.5$ and $\mu_{CR2} := 0.8$ in (29).

Step 4: If the termination condition $t = N_T$ holds, output the best one $\boldsymbol{x}_b \in \boldsymbol{P}_{N_T}$ and terminate.

Step 5: By using the scale factor F_i and the crossover rate CR_i, generate the trial vectors $\boldsymbol{z}_i \in \boldsymbol{X}$ from the target vectors $\boldsymbol{x}_i \in \boldsymbol{P}_t$, $i = 1, \cdots, N_P$.

Step 6: Comparing $\boldsymbol{z}_i \in \boldsymbol{X}$ with $\boldsymbol{x}_i \in \boldsymbol{P}_t$, choose the better one for $\boldsymbol{x}_i \in \boldsymbol{P}_{t+1}$, $i = 1, \cdots, N_P$.

Step 7: Update the locations μ_{F1} and μ_{F2}.

Step 8: Update the means μ_{CR1} and μ_{CR2}.

Step 9: $t := t + 1$. Go back to Step 4.

4.4 Verification of Solution

The solution $\boldsymbol{x}_b \in \boldsymbol{X}$ of JCCP in (18) obtained by JADE2G is checked whether it satisfies the chance constraint of JCCP in (3). By using a huge number of samples $\boldsymbol{\xi}^n \in \widehat{\boldsymbol{\Xi}} \subseteq \boldsymbol{\Omega}$, $n = 1, \cdots, \widehat{N}$ selected randomly, the empirical probability $h(\boldsymbol{x}_b, \widehat{\boldsymbol{\Xi}})$ with the solution $\boldsymbol{x}_b \in \boldsymbol{X}$ is calculated as shown in (8). For any $\epsilon \in (0, 1]$ and $\delta \in (0, 1]$, if the following sample size $\hat{N}$ is used for $\boldsymbol{\xi}^n \in \widehat{\boldsymbol{\Xi}} \subseteq \boldsymbol{\Omega}$,

$$\widehat{N} \geq \frac{1}{2\,\epsilon^2} \log\left(\frac{2}{\delta}\right) \tag{35}$$

then Chernoff bound [23] ensures that

$$\Pr\left(\left|h(\boldsymbol{x}_b, \boldsymbol{\Omega}) - h(\boldsymbol{x}_b, \widehat{\boldsymbol{\Xi}})\right| \leq \epsilon\right) \geq 1 - \delta. \tag{36}$$

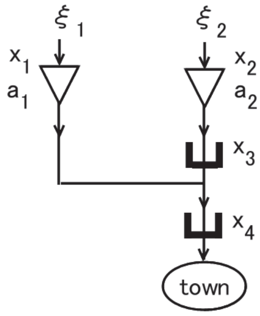


Figure 5: Topological river model for Case Study 1

In this paper, $\epsilon = 10^{-3}$ and $\delta = 0.01$ are chosen in (35). Therefore, $\hat{N} = 2,649,159$ samples are used to calculate the empirical probability $h(\boldsymbol{x}_b, \widehat{\boldsymbol{\Xi}})$.

If the solution $\boldsymbol{x}_b \in \boldsymbol{X}$ of JCCP in (18) obtained by JADE2G meets the condition in (37), we accept $\boldsymbol{x}_b \in \boldsymbol{X}$ as a solution of JCCP in (3). Otherwise, we increase the value of the correction level β in (18) just a little. Then, in order to obtain another solution, we apply JADE2G to JCCP in (18) again.

$$h(\boldsymbol{x}_b, \widehat{\boldsymbol{\Xi}}) \geq \alpha \tag{37}$$

5 Flood Control Planning

In the flood control planning [10], we make a plan to protect an urban area from the flood damage caused by torrential rain. The flood control reservoir system design has been formulated as JCCP by using several linear constraints [24]. In addition to the reservoir, we consider the water-retaining capacity of forest in the formulation of the flood control planning.

According to the model of a forest in watershed [25], the area of forest $a \in \Re$ is a constant. The amount of rainfall $\xi \in \Re$ per unit area is a random variable. The water-retaining capacity of forest $x \in \Re$ per unit area is regarded as a decision variable because it can be controlled through the forest maintenance such as afforestation. Thereby, the inflow of water $q \in \Re$ from the forest to river can be described as

$$q(x, \xi, a) = a\,(\xi - x\,(1 - \exp(-\xi/x))) \tag{38}$$

where the effect of past rainfall is not considered.

5.1 Case Study 1

Fig. 5 shows a topological river model. The arrow indicates the direction where water flows. Symbol $\bigtriangledown$ denotes a forest. There are two forests in Fig. 5. It rains in each of the two forests. The amounts of rainfall (ξ_1 and ξ_2) are random variables following the normal distributions defined by (13). The areas of two forests (a_1 and a_2) are constants. The water-retaining capacities of them (x_1 and x_2) are decision variables. From (38), the inflows of water $Q_j \in \Re$, $j = 1, 2$ from the two forests to the river are described as

$$Q_j = q(x_j,\, \xi_j,\, a_j),\ j = 1, 2. \tag{39}$$

Symbol $\sqcup$ denotes a reservoir. There are two reservoirs in Fig. 5. The capacities of reservoirs to be built (x_3 and x_4) are also decision variables. From (39), the inflow of water from the river to the town located at the lower part of the river is

$$\begin{aligned} & g(\boldsymbol{x},\, \boldsymbol{\xi}) \\ & = \max\{\max\{Q_2 - x_3,\ 0\} + Q_1 - x_4,\ 0\}. \end{aligned} \tag{40}$$

The inflow of water in (40) can be rewritten as

$$\begin{aligned} & g(\boldsymbol{x},\, \boldsymbol{\xi}) \\ & = \begin{cases} g_1(\boldsymbol{x},\, \boldsymbol{\xi}) & \text{if } (Q_2 > x_3) \\ & \quad \wedge (Q_1 + Q_2 > x_3 + x_4) \\ g_2(\boldsymbol{x},\, \boldsymbol{\xi}) & \text{if } (Q_2 \le x_3) \\ & \quad \wedge (Q_1 > x_4) \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{41}$$

where the functions $g_m(\boldsymbol{x},\, \boldsymbol{\xi})$, $m = 1, 2$ are given as

$$\left(\begin{array}{l} g_1(\boldsymbol{x},\, \boldsymbol{\xi}) = Q_1 + Q_2 - x_3 - x_4 \\ g_2(\boldsymbol{x},\, \boldsymbol{\xi}) = Q_1 - x_4. \end{array} \right. \tag{42}$$

We want to prevent the inflow of water from the river to the town in Fig. 5. The risk that the town is damaged by flooding is restricted by a sufficiency level $\alpha \in (0, 1)$. The cost for the flood control is the objective function to be minimized. The maintenance cost of a forest is proportional to its capacity. The construction cost of a reservoir is proportional to the square of its capacity. Consequently, from (41), the flood control planning is formulated as

$$\left(\begin{array}{l} \min \quad g_0(\boldsymbol{x}) = \displaystyle\sum_{j=1}^{2} a_j\, x_j + \sum_{j=3}^{4} c_j\, x_j^2 \\ \text{sub. to} \\ \quad \Pr \left(\begin{array}{l} \forall \boldsymbol{\xi} \in \boldsymbol{\Omega} :\ g_m(\boldsymbol{x},\, \boldsymbol{\xi}) \le 0, \\ m = 1, 2 \end{array} \right) \ge \alpha \\ \quad 0.5 \le x_1 \le 1.5,\ 0.5 \le x_2 \le 1.5 \\ \quad 0 \le x_3 \le 2,\ 0 \le x_4 \le 3 \end{array} \right. \tag{43}$$

where $a_1 = 2$, $a_2 = 2$, $c_3 = 3$, and $c_4 = 1$.

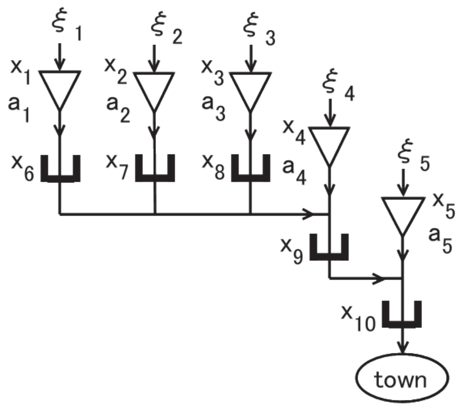


Figure 6: Topological river model for Case Study 2

5.2 Case Study 2

Fig. 6 shows a topological river model in the same way with Fig. 5. There are five forests in Fig. 6. The amounts of rainfall $\boldsymbol{\xi} \in \boldsymbol{\Omega} \subseteq \Re^5$ are given by a multivariate normal distribution with the following mean $\boldsymbol{\mu} \in \Re^5$ and variance $\boldsymbol{\sigma}^2 \in \Re^5$.

$$\left(\begin{array}{l} \boldsymbol{\mu} = (2.0,\, 1.5,\, 2.5,\, 0.8,\, 1.0) \\ \boldsymbol{\sigma}^2 = (0.2^2,\, 0.3^2,\, 0.2^2,\, 0.1^2,\, 0.1^2) \end{array} \right. \tag{44}$$

The correlation matrix of $\boldsymbol{\xi} \in \boldsymbol{\Omega}$ is given as

$$\boldsymbol{R} = \begin{pmatrix} 1.0 & -0.5 & 0.0 & 0.3 & -0.5 \\ -0.5 & 1.0 & -0.8 & 0.0 & 0.2 \\ 0.0 & -0.8 & 1.0 & 0.0 & 0.3 \\ 0.3 & 0.0 & 0.0 & 1.0 & 0.0 \\ -0.5 & 0.2 & 0.3 & 0.0 & 1.0 \end{pmatrix} \tag{45}$$

There is a town at the lower part of the river in Fig. 6. We want to prevent the inflow of water from the river to the town. The water-retaining capacities of five forests x_j, $j = 1, \cdots, 5$ and the capacities of five reservoirs x_j, $j = 6, \cdots, 10$ are decision variables. As well as JCCP in (43), the cost for the flood control is the objective function to be minimized. Thus the flood control planning in Fig. 6 is formulated as

$$\left(\begin{array}{l} \min \quad g_0(\boldsymbol{x}) = \displaystyle\sum_{j=1}^{5} a_j\, x_j + \sum_{j=6}^{10} c_j\, x_j^2 \\ \text{sub. to} \\ \quad \Pr \left(\begin{array}{l} \forall \boldsymbol{\xi} \in \boldsymbol{\Omega} :\ g_m(\boldsymbol{x},\, \boldsymbol{\xi}) \le 0, \\ m = 1, \cdots, 9 \end{array} \right) \ge \alpha \\ \quad 0.5 \le x_j \le 1.5,\ j = 1, \cdots, 5 \\ \quad 0 \le x_j \le 3,\ j = 6, 7, 8 \\ \quad 0 \le x_j \le 4,\ j = 9, 10 \end{array} \right. \tag{46}$$

where $a_j = 2$, $j = 1, \cdots, 5$, $c_j = 3$, $j = 6, 7, 8$, $c_9 = 2$, and $c_{10} = 1$. In the same way as we have

Table 1: Experimental results for JCCP in (43)

ρ_{12}	$g_0(\boldsymbol{x}_b)$	$h(\boldsymbol{x}_b, \boldsymbol{\Theta})$	$h(\boldsymbol{x}_b, \widehat{\boldsymbol{\Xi}})$	β
-0.8	11.018	0.901	0.913	0.918
0.0	11.583	0.921	0.908	0.938
$+0.8$	11.994	0.926	0.904	0.941

shown in (40), (41), and (42), the function $g_m(\boldsymbol{x}, \boldsymbol{\xi})$, $m = 1, \cdots, 9$ can be derived as

$$\begin{cases} g_1(\boldsymbol{x}, \boldsymbol{\xi}) = Q_1 + Q_2 + Q_3 + Q_4 + Q_5 \\ \qquad -x_6 - x_7 - x_8 - x_9 - x_{10} \\ g_2(\boldsymbol{x}, \boldsymbol{\xi}) = Q_1 + Q_2 + Q_4 + Q_5 \\ \qquad -x_6 - x_7 - x_9 - x_{10} \\ g_3(\boldsymbol{x}, \boldsymbol{\xi}) = Q_1 + Q_3 + Q_4 + Q_5 \\ \qquad - x_6 - x_8 - x_9 - x_{10} \\ g_4(\boldsymbol{x}, \boldsymbol{\xi}) = Q_2 + Q_3 + Q_4 + Q_5 \\ \qquad -x_7 - x_8 - x_9 - x_{10} \\ g_5(\boldsymbol{x}, \boldsymbol{\xi}) = Q_1 + Q_4 + Q_5 \\ \qquad -x_6 - x_9 - x_{10} \\ g_6(\boldsymbol{x}, \boldsymbol{\xi}) = Q_2 + Q_4 + Q_5 \\ \qquad -x_7 - x_9 - x_{10} \\ g_7(\boldsymbol{x}, \boldsymbol{\xi}) = Q_3 + Q_4 + Q_5 \\ \qquad -x_8 - x_9 - x_{10} \\ g_8(\boldsymbol{x}, \boldsymbol{\xi}) = Q_4 + Q_5 - x_9 - x_{10} \\ g_9(\boldsymbol{x}, \boldsymbol{\xi}) = Q_5 - x_{10} \end{cases} \tag{47}$$

where $Q_j = q(x_j, \xi_j, a_j)$, $j = 1, \cdots, 5$.

6 Numerical Experiment

6.1 Case Study 1

JADE2G is coded by MATLAB [26]. Then JADE2G is applied to the flood control planning stated as JCCP in (43). The control parameters of JADE2G are chosen as $N_P = 20$, $N_T = 60$, and $p = 0.2$. About $N = 60$ points $\boldsymbol{\theta}^n \in \boldsymbol{\Theta}$, $n = 1, \cdots, N$ are used to evaluate each solution $\boldsymbol{x}_i \in \boldsymbol{P}_t$. The sufficiency level in (43) is chosen as $\alpha = 0.9$. Three correlation coefficients, namely $\rho_{12} = \pm 0.8$ and $\rho_{12} = 0$, are considered in (14) for the amount of rainfall $\boldsymbol{\xi} \in \boldsymbol{\Omega}$. JADE2G is applied to each JCCP 10 times.

Table 1 shows the results of experiment averaged over 10 runs. The objective function value $g_0(\boldsymbol{x}_b)$ of the best solution $\boldsymbol{x}_b \in \boldsymbol{X}$ obtained by JADE2G, the empirical probabilities $h(\boldsymbol{x}_b, \boldsymbol{\Theta})$ and $h(\boldsymbol{x}_b, \widehat{\boldsymbol{\Xi}})$, and the correction level β used for JCCP in (18) are shown in Table 1. From (36), we can regard $h(\boldsymbol{x}_b, \widehat{\boldsymbol{\Xi}})$ as the true probability $h(\boldsymbol{x}_b, \boldsymbol{\Omega})$ achieved by $\boldsymbol{x}_b \in \boldsymbol{X}$. From Table 1, we can confirm that the condition in (37) is

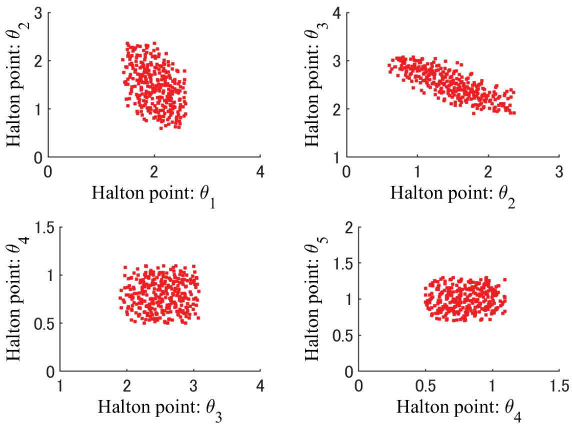


Figure 7: Truncated Halton Sequence $\boldsymbol{\theta}^n \in \boldsymbol{\Theta}$

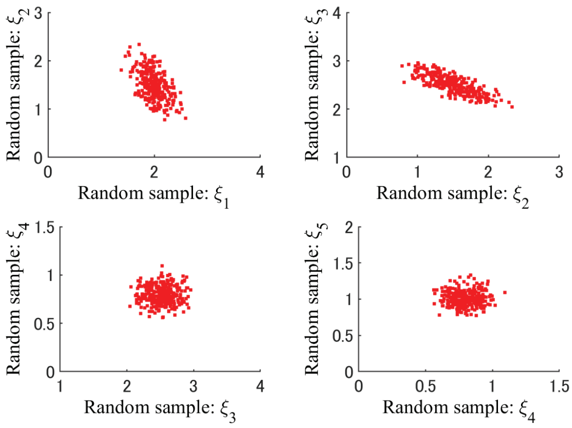


Figure 8: Random samples $\boldsymbol{\xi}^n \in \boldsymbol{\Xi}$

satisfied in all cases. Furthermore, we can see that the value of $g_0(\boldsymbol{x}_b)$ increases proportionally to ρ_{12}.

6.2 Case Study 2

By changing the value of sufficiency level $\alpha \in (0, 1)$, JADE2G is applied to to the flood control planning stated as JCCP in (46). The control parameters of JADE2G are chosen as $N_P = 50$, $N_T = 200$, and $p = 0.2$. About $N = 300$ points $\boldsymbol{\theta}^n \in \boldsymbol{\Theta}$ of THS are used to evaluate each $\boldsymbol{x}_i \in \boldsymbol{P}_t$. Fig. 7 shows the points $\boldsymbol{\theta}^n \in \boldsymbol{\Theta}$, $n = 1, \cdots, N$. Considering the PDF of $\boldsymbol{\xi} \in \boldsymbol{\Omega}$, $f_{\min} = 10^{-3}$ is used. Incidentally, Fig. 8 shows $N = 300$ samples $\boldsymbol{\xi}^n \in \boldsymbol{\Xi}$ generated by RS.

Table 2 shows the results of experiment. The objective function value $g_0(\boldsymbol{x}_b)$ of the best solution $\boldsymbol{x}_b \in \boldsymbol{X}$, the empirical probabilities $h(\boldsymbol{x}_b, \boldsymbol{\Theta})$ and $h(\boldsymbol{x}_b, \widehat{\boldsymbol{\Xi}})$ are shown in Table 2. The correction level is chosen as $\beta = \alpha$. Therefore, the condition in (37) is not yet satisfied by $\boldsymbol{x}_b \in \boldsymbol{X}$ in some cases.

Fig. 9 shows the trade-off relationship between

Table 2: Experimental results for JCCP in (46)

α	$g_0(\boldsymbol{x}_b)$	$h(\boldsymbol{x}_b, \boldsymbol{\Theta})$	$h(\boldsymbol{x}_b, \widehat{\boldsymbol{\Xi}})$	β
0.95	33.682	0.950	0.926	0.95
0.90	33.467	0.914	0.910	0.90
0.85	32.901	0.859	0.839	0.85
0.80	32.668	0.819	0.810	0.80
0.75	32.221	0.752	0.734	0.75
0.70	32.178	0.711	0.699	0.70
0.65	31.682	0.652	0.616	0.65

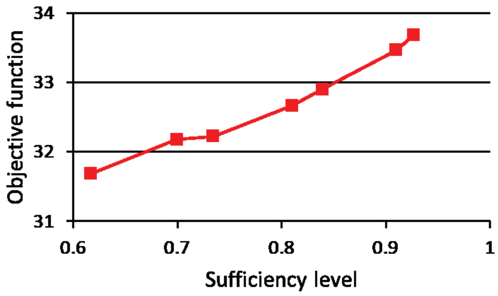


Figure 9: Trade-off between $g_0(\boldsymbol{x}_b)$ and α

the objective function value $g_0(\boldsymbol{x}_b)$ and the sufficiency level $\alpha \in (0, 1)$ that satisfies the condition in (37) for the solution $\boldsymbol{x}_b \in \boldsymbol{X}$. The experimental results in Table 2 are summarized by Fig. 9. From Fig. 9, we can see that the safety of the town shown in Fig. 6 depends on the budget for the flood control.

7 Conclusion

A number of real-world applications can be formulated as CCPs. However, the time-consuming Monte Carlo simulation based on random sampling has been required to solve CCP. In this paper, an optimization method based on THS and JADE2G was proposed to solve CCP efficiently. Even though there are two types of CCPs, namely JCCP and SCCP, the proposed method is applicable to both types of CCPs.

The performance of the optimization method was demonstrated through a real-world application, namely the flood control planning formulated as JCCP. By the way, in the flood control planning, we assumed the normal distribution for the amount of rainfall, but the probability estimation technique based on THS can be applied effectively to arbitrary probability distribution if its PDF is provided.

In the future work, we would like to demonstrate the usefulness of the proposed method through the various real-world applications formulated as CCPs.

Acknowledgements: The research was supported by JSPS KAKENHI (17K060508).

References:

[1] A. Ben-Tal, L. E. Ghaoui, and A. Nemirovski, *Robust Optimization*. Princeton University Press, 2009.

[2] A. Prékopa, *Stochastic Programming*. Kluwer Academic Publishers, 1995.

[3] S. P. Uryasev, *Probabilistic Constrained Optimization: Methodology and Applications*. Kluwer Academic Publishers, 2001.

[4] W. Jiekang, Z. Jianquan, C. Guotong, and Z. Hongliang, "A hybrid method for optimal scheduling of short-term electric power generation of cascaded hydroelectric plants based on particle swarm optimization and chance-constrained programming," *IEEE Trans. on Power Systems*, vol. 23, no. 4, pp. 1570–1579, 2008.

[5] M. Lubin, Y. Dvorkin, and S. Backhaus, "A robust approach to chance constrained optimal power flow with renewable generation," *IEEE Trans. on Power Systems*, vol. 31, no. 5, pp. 3840–3849, 2016.

[6] K. Tagawa, "Chebyshev inequality based approach to chance constrained portfolio optimization," *International Journal of Mathematical and Computational Methods*, vol. 2, pp. 66–71, 2017.

[7] C. A. Poojari and B. Varghese, "Genetic algorithm based technique for solving chance constrained problems," *European Journal of Operational Research*, vol. 185, pp. 1128–1154, 2008.

[8] B. Liu, Q. Zhang, F. V. Fernández, and G. G. E. Gielen, "An efficient evolutionary algorithm for chance-constrained bi-objective stochastic optimization," *IEEE Trans. on Evolutionary Computation*, vol. 17, no. 6, pp. 786–796, 2013.

[9] K. Tagawa and B. S. Miyanaga, "Weighted empirical distribution based approach to chance constrained optimization problems using differential evolution," in *Proc. of IEEE Congress on Evolutionary Computation (CEC2017)*, 2017, pp. 97–104.

[10] K. Tagawa and S. Miyanaga, "An approach to chance constrained problems using weighted

empirical distribution and differential evolution with application to flood control planning," *Electronics and Communications in Japan*, vol. 102, no. 3, pp. 45–55, 2019. [Online]. Available: https://doi.org/10.1002/ecj.12148

[11] K. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*. Springer, 2005.

[12] K. Tagawa, "Truncated Halton sequence and adaptive differential evolution to solve joint chance constrained problems with application to flood control planning," in *Proc. of 2nd European Conference on Electrical Engineering & Computer Science (EECS2018)*, 2018, pp. 1–6.

[13] R. B. Ash, *Basic Probability Theory*. Dover, Downers Grove, 2008.

[14] J. H. Halton, "On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals," *Numerische Mathematik*, vol. 2, no. 1, pp. 84–90, 1960.

[15] X. Wang, "Randomized Halton sequences," *Mathematical and Computer Modeling*, vol. 32, pp. 887–899, 2000.

[16] S. Das and P. N. Suganthan, "Differential evolution: a survey of the state-of-the-art," *IEEE Trans. on Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, 2011.

[17] J. Brest, S. Greiner, B. Bošković, M. Merink, and V. Žumer, "Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems," *IEEE Trans. on Evolutionary Computation*, vol. 10, no. 6, pp. 646–657, 2006.

[18] J. Zhang and A. C. Sanderson, "JADE: Adaptive differential evolution with optional external archive," *IEEE Trans. on Evolutionary Computation*, vol. 13, no. 5, pp. 945–958, 2009.

[19] T. Takahama and S. Sakai, "An adaptive differential evolution with learning parameters according to groups defined by the rank of objective values," in *Proc. of Int. Conference on Swarm Intelligence (ICSI2017)*, 2017, pp. 411–419.

[20] K. Tagawa, "Extension of weighted empirical distribution and group-based adaptive differential evolution for joint chance constrained problems," in *Proc. of Genetic and Evolutionary Computation Conference (GECCO'18) Companion*, 2018, pp. 223–224. [Online]. Available: https://doi.org/10.1145/3205651.3205711

[21] E. E. 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.

[22] K. Deb, "An efficient constraint handling method for genetic algorithms," *Computer Methods in Applied Mechanica and Engineering*, vol. 186, pp. 311–338, 2000.

[23] R. Tempo, G. Calafiore, and F. Dabbene, *Randomized Algorithms for Analysis and Control of Uncertain Systems: With Applications*. Springer, 2012.

[24] A. Prékopa and T. Szántai, "Flood control reservoir system design using stochastic programming," *Mathematical Programming Study*, vol. 9, pp. 138–151, 1978.

[25] E. Maita and M. Suzuki, "Quantitative analysis of direct runoff in a forested mountainous, small watershed," *Journal of Japan Society of Hydrology and Water Resources*, vol. 22, no. 5, pp. 342–355, 2009.

[26] A. R. Martinez and W. L. Martinez, *Computational Statistics Handbook with MATLAB ®, Second Edition*. Chapman & Hall/CRC, 2008.