Hindawi
Mathematical Problems in Engineering
Volume 2021, Article ID 9096272, 13 pages
https://doi.org/10.1155/2021/9096272

Research Article

A New Methodology for Solving Multiobjective Chance-Constrained Problems: An Application on IoT Systems

Kumru Didem Atalay,[1] **Tacettin Sercan Pekin**,[2] **and Ayşen Apaydin**[3]

[1]*Baskent University, Faculty of Engineering, Department of Industrial Engineering, Eskisehir Yolu 20 km Baglica, TR-06970, Ankara, Turkey*
[2]*Biligoba Technologies, Middle East Technical University Arge Ve Eğitim Merkezi, No. 13, Baraka 2, Cankaya, Ankara, Turkey*
[3]*Ankara University, Faculty of Applied Sciences, Department of Insurance and Actuarial Sciences, Cebeci, Ankara, Turkey*

Correspondence should be addressed to Kumru Didem Atalay; katalay@baskent.edu.tr

Received 17 November 2019; Revised 29 August 2021; Accepted 30 September 2021; Published 19 October 2021

Academic Editor: Alessandro Palmeri

This study presents a newly developed methodology to transform the chance-constrained problem into a deterministic problem and then solving this multiobjective deterministic problem with the proposed method. Chance-constrained problem contains independent gamma random variables that are denoted as a_{ij}. Two methods are proposed to obtain the deterministic equivalent of chance-constrained problem. The first of the methods is directly based on using the distribution, and the second consists of normalizing probabilistic constraints using Lyapunov's central limit theorem. An algorithm which uses the Global Criterion Method is developed to solve the multiobjective deterministic equivalent of chance-constrained problem. The methodology is applied to a real-life engineering problem that consists of an IoT device and its data sending process. Using Lyapunov's central limit theorem for large numbers of random variables is found to be more appropriate.

1. Introduction

Most decision problems, whether real world or hypothetical, can be defined with a mathematical programming model. Unfortunately, in many practical systems, it is very difficult to gather all necessary information to derive such models. This difficulty arises from the uncertainty of many of the factors in modeling equations. One possible reason for this uncertainty is the randomness of these factors, and stochastic programming is a branch of mathematical programming that theoretically and empirically focuses on "conditional extremum problems under incomplete information about random coefficients" [1].

In terms of the investigation and optimization of complicated systems, the deterministic approach broadly dominates projection, planning, and management processes in economic and military sciences. However, by assuming that all initial data exhibit the same basic characteristics, the deterministic approach ignores the probabilistic properties of complicated systems. On the one hand, these deterministic systems emphasize objective laws. On the other hand, the deterministic approach is constantly undermined by random factors in real-life data. As such, a stochastic approach should be more useful for most real-life problems [2].

Assume that a decision-maker wants to find the so-called "absolutely best" decision that corresponds to the minimum (or maximum) of a suitable objective function, while satisfying a given set of feasibility constraints. If the decision models are very complex, standard optimization strategies cannot be used to solve them directly. Therefore, a global optimization technique should be used. Global optimization applications are popular in all disciplines, including science, engineering, economy, and applied science. Global optimization is aimed at finding the best global solution of a model, in the presence of multiple local optima [3].

The classical linear programming problem, which is a specific class of mathematical programming problems, is formulated as follows:

$$\begin{aligned} &\max z(x) = \sum_{j=1}^{n} c_j x_j, \\ &\sum_{j=1}^{n} a_{ij} x_j \le b_i, \quad i = 1, \ldots, m, \\ &x_j \ge 0, \quad j = 1, \ldots, n. \end{aligned} \tag{1}$$

Note that all coefficients (i.e., technologic coefficients a_{ij}, right-hand side values b_i, and objective function coefficients c_j for $j = 1, .., n$ and $i = 1, \ldots, m$) are deterministic. However, when at least one coefficient is a random variable, the problem becomes a stochastic programming problem.

Stochastic programming problems can be classified into the following three categories: probability distribution problems, chance-constrained problems, and recourse problems.

This paper attempts to analyze a class of chance-constrained stochastic programming problems. Chance-constrained programming assumes the presence of random data variations and permits constraint violations up to specified probability limits. To transform equation (1) into a chance-constrained problem, the original constraints $\sum_{j=1}^{n} a_{ij}x_j \le b_i$, $i = 1, \ldots, m$, are replaced with the chance-constrained programming formulation $P(\sum_{j=1}^{n} a_{ij}x_j \le b_i) \ge 1 - \delta_i$, $i = 1, \ldots, m$. The term "chance-constrained" is used because of the constraints of $P(.)$, which designate the probability of the event in parentheses. The parameters $\delta_i\,(0 < \delta_i \le 1)$ are assumed to be deterministically fixed. Chance-constrained programming is one of the major approaches for addressing random coefficients in mathematical programming problems [1, 4–9].

Chance-constrained programming was first developed by Charnes and Cooper [5]. A variant of the Charnes and Cooper model, which utilized joint chance constraints, is given by Miller and Wagner [10]. Prekopa [11] introduced a more general case in which the right-hand side is allowed to have stochastically dependent components. Contini [12] developed an algorithm for stochastic goal programming which assumes that the random variable coefficients are normally distributed with known means and variances. In doing so, he transformed the stochastic problem into an equivalent deterministic quadratic programming problem. Rather than assuming that the coefficients are normally distributed, Sengupta [8] assumed a chi-squared distribution, which has a nonnegative range and is more applicable to economic models of production and investment. Jagannathan [7] obtained equivalent equations for chance-constrained programming models based on coefficient matrices with normally distributed elements and dependent random right-hand side elements. Regarding multiobjective stochastic linear programming problems, Stancu-Minasian [13] suggested a minimum risk approach, while both Leclercq [14] and Teghem et al. [15] have proposed interactive methods. Extensive discussion on the convex properties of stochastic problems can be found in [16]. Meanwhile, a fuzzy approach to stochastic programming has been presented by Mohan and Nguyen [17]. An interactive fuzzy satisfying method was also developed by Sakawa et al. [18]. Yang and Wen [19] presented a chance-constrained programming model for transmission system planning in the competitive electricity market environment. Huang [20] provided two types of credibility-based chance-constrained models for portfolio selection with fuzzy returns. Ağpak and Gökçen [21] developed new mathematical models for stochastic traditional and U-type assembly lines with a chance-constrained 0-1 integer programming technique. Henrion and Strugarek [22] investigated the convexity of chance constraints with independent random variables. Parpas et al. [23] proposed a stochastic algorithm for the global optimization of chance-constrained problems. They assumed that the probability measure used to evaluate the constraints is known only through its moments. Xu et al. [24] developed a robust hybrid stochastic chance-constraint programming model to support municipal solid waste management under uncertainty. Ben Abdelaziz and Masri [25] proposed a chance-constrained approach and a compromise programming approach to transform the multiobjective stochastic linear program with partial linear information on the probability distribution into its equivalent uniobjective problem. Goyal and Ravi [26] presented a polynomial time approximation scheme for the chance-constrained knapsack problem when item sizes are normally distributed and independent of other items. Zhang and Peng [27] proposed uncertain optimal assignment problem in which the uncertain factor is described by uncertainty theory. In order to construct an uncertain programming model, the concept of α-optimal assignment is proposed. After that, α-optimal model is given. Chen et al. [28] investigated an uncertain bicriteria solid transportation problem. Based on these equivalence relations, the optimal transportation plans were obtained by expected value goal programming and chance-constrained goal programming. Zhang et al. [29] investigated the fixed-charge solid transportation problem under uncertainty. Three models under different criteria were presented, which include expected value model, chance-constrained programming model, and measure-chance programming model. Majumder et al. [30] proposed an uncertain multiobjective Chinese postman problem under the framework of uncertainty theory, for an undirected connected network. The model is then solved using two classical multiobjective techniques, the Global Criterion Method and fuzzy programming method, and two multi-objective genetic algorithms. Majumder et al. [31] proposed a biobjective rough-fuzzy quadratic minimum spanning tree problem which is modeled using rough-fuzzy chance-constrained programming technique. The linear and quadratic weights of the proposed model are considered as two different objectives which are optimized simultaneously. Atalay and Apaydın [4] proposed a new method using Esseen inequality to obtain deterministic equivalent of chance-constraint problems with gamma random variables. The method in this paper uses estimation of the distance between distribution of sum of these independent random variables having gamma distribution and normal distribution; probabilistic constraint obtained via Esseen inequality has been made deterministic using the approach suggested by Polya. The main purpose of the article is to transform the chance-constrained model into a deterministic model based on the Esseen inequality. As a result, the upper bounds of the chance constraints are derived by the Esseen inequality and approximate deterministic equivalent of the model is developed. Sarkar et al. [32] have reduced the chance constraints with independent gamma parameters to deterministic constraints. They have used geometric inequality and some other inequalities for obtained deterministic equivalent of chance constraints where the objective function may be considered as linear or nonlinear with deterministic cost coefficient. In contrast to the two papers discussed above, in our paper, distribution function is directly used and multiobjective mathematical model solution is obtained with the use of Global Criterion Method. Distribution function is on the left-hand side of the chance constraint. Besides, Pareto optimal solutions are also provided. Furthermore, deterministic equivalent of chance constraint which is derived by direct distribution function is compared to the approximate model which is derived using Lyapunov's central limit theorem.

To the best of our knowledge, there is not any study that proposes an algorithm for solving multiobjective chance-constraint problems with gamma random variables by directly using the distribution and then using Lyapunov's central limit theorem and Global Criterion Method. Instead of using quantile rules, we directly used the distribution function. In this case, since a_{kj} are gamma random variables, $\sum_{j=1}^{n} a_{kj}x_j$ also becomes gamma random variables. Instead of using quantile function, distribution function of sum of independent random variables is directly used. According to this, distribution function $F(.)$ is obtained and it is assigned as the left-hand side value of the chance constraint. In the mathematical model, the parameters of the chance constraint which are random variables have a distribution function and this distribution function provides the direct calculation of probability. Therefore, probabilistic chance-constraint corresponding to the direct distribution function is itself deterministic equivalent.

In this paper, we consider a multiobjective chance-constrained stochastic programming problem, where the technologic coefficients a_{ij} are independent gamma random variables with known means and variances. The decision why we use the gamma distribution is motivated by the fact that it has relationships with gamma and normal distributions. If univariate distribution relationship is examined, Gamma random variables consisting $\alpha\beta$ and $\alpha^2\beta$ parameters converge to normal distribution with parameters $\mu = \alpha\beta$ and $\sigma^2 = \alpha^2\beta$ and $\beta \longrightarrow \infty$ and $X \sim N(\mu, \sigma^2)$ [33]. Bivariate gamma distributions, such as by marginal transformation, are mainly derived from the bivariate normal in some fashion. A univariate chi-squared distribution can be derived from one or more independent and identically distributed normal variables. Moreover, a chi-squared random variable is a special case of gamma. As a result of this, a bivariate gamma model is related to the bivariate normal one [34].

In addition, other distributions such as the exponential and chi-square distributions can be closely approximated by a gamma distribution. For our chance-constrained model, we developed a deterministic equivalent model without random variables. A transformation was implemented to convert the chance-constrained programming problem into a deterministic equivalent problem. This deterministic equivalent problem is obtained by two methods, that is, by deriving the direct distribution and then normalizing probabilistic constraints using Lyapunov's central limit theorem [35]. After converting the problem into a deterministic model, we used global optimization to propose an algorithm that utilizes the Global Criterion Method to solve the multiobjective problems. The solution was applied to a real-life engineering problem. IoT devices and distributed computing are recently a hot topic, and there are fairly new problems arising rapidly following the emerging new hardware and software running on these IoT devices. These devices have limitations on energy usage, data transfer rate, and continuous usage time. These limitations and the need for maximum data transfer amount make the application a suitable problem for the model. We have applied the model for such an IoT application in this study. Using numerical examples, we compared the objective functions of models obtained from these exact and approximate methods.

2. Chance-Constrained Programming

An ordinary linear programming model is said to be chance-constrained if its linear constraints are associated with a set of probability measures that indicate the degree to which constraints can be violated [36]. As a stochastic programming method, chance-constrained programming includes fixing certain appropriate levels for random constraints. Thus, it is generally used for modeling technical or economic systems, particularly economic planning, input control, structural design, inventory, and air and water quality management problems. Under chance-constrained programming, each constraint can be realized with a certain probability.

When some or all of the coefficients of the model are independent random variables, the chance-constrained programming formulation requires only that each constraint must hold for most of these combinations [6, 8, 37].

A multiobjective chance-constrained programming problem can be stated as follows:

$$\begin{gathered}
\max z_1(x) = \sum_{j=1}^{n} c_{1j}x_j = f_1(x), \\
\max z_2(x) = \sum_{j=1}^{n} c_{2j}x_j = f_2(x), \\
\vdots \\
\max z_K(x) = \sum_{j=1}^{n} c_{Kj}x_j = f_K(x), \\
P\left(\sum_{j=1}^{n} a_{ij}x_j \le b_i\right) \ge 1 - \delta_i, \quad \text{for } i = 1, 2, \dots, m, \\
x_j \ge 0, \quad j = 1, 2, \dots, n, \\
\delta_i \in (0, 1), \quad i = 1, 2, \dots, m.
\end{gathered} \tag{2}$$

Note that δ_i are specified probabilities and that c_{lj}, $l = 1, 2, ..., K$, a_{ij}, and b_i are independent random variables. A general CCP problem is defined with model (2). In the model, one of the parameters, two of the parameters, or three of the parameters can be random variables. However, the main focus of this work is on the case where a_{ij} parameters are gamma random variables. Here, it is assumed that the decision variables x_i are deterministic.

3. The Deterministic Equivalent of Chance-Constraint Models

In this section, we derive a deterministic equivalent of model (2), where a_{ij}, $j = 1, ..., n$, are independent random variables with known means and variances and $c_{lj}$and b_i are constants. The k^{th} chance constraint is given in model (2).

$$P\left(\sum_{j=1}^{n} a_{kj} x_j \le b_k\right) \ge 1 - \delta_k. \tag{3}$$

Note that each a_{kj} has a finite third central moment. If the deterministic equivalent of constraint (3) can be derived, the problem can be converted by determining the distribution of $\sum_{j=1}^{n} a_{kj} x_j$, which is the linear combination of a_{kj}. To derive the deterministic equivalent of the chance constraint (3), we propose two methods, namely, deriving the direct distribution and using Lyapunov's central limit theorem. Both methods are first defined, and then they are adapted to the gamma distribution.

The most suitable method for deriving the deterministic equivalent of a chance-constrained programming model is to find the direct distribution. However, as the number of decision variables increases to a large value of n (more precisely, $n \ge 5$), it becomes very difficult to obtain the probability $P(\sum_{j=1}^{n} a_{kj} x_j \le b_k)$. Therefore, in this case Lyapunov's central limit theorem can be used to obtain an approximate solution.

3.1. Method I: Transformation of a Chance-Constrained Programming Model into a Deterministic Model Using the Distribution Directly—Multivariate Transformation. Let $X_1, X_2, \ldots, X_n$ be a random vector with a probability density function $f_X(x_1, x_2, ..., x_n)$. Let $A = \{x \colon f_X(x) > 0\}$. Consider a new random vector, $(U_1, U_2, \ldots, U_n)$, which is defined as follows:

$$\begin{aligned}
U_1 &= g_1(X_1, X_2, ..., X_n),\\
U_2 &= g_2(X_1, X_2, ..., X_n),\\
&\vdots\\
U_n &= g_n(X_1, X_2, ..., X_n).
\end{aligned} \tag{4}$$

Suppose that $A_0, A_1, \ldots, A_n$ is from a partition of $\mathscr{A}$. The set A_0, which may be an empty set, satisfies $P((X_1, X_2, \ldots, X_n) \in A_0) = 0$. The transformation $(U_1, U_2, \ldots, U_n) = (g_1(X_1, .., X_n), \ldots, g_n(X_1, .., X_n))$ is a one-to-one transformation from A_j onto $\mathscr{B} = \{u_1, u_2, \ldots, u_n \colon u_1 = g_1(x_1, x_2, \ldots, x_n), \ldots, u_n = g_n(x_1, x_2, \ldots, x_n)\}$ for each $j = 1, 2, \ldots, k$. Then, for each j, the inverse function from $\mathscr{B}$ to A_j can be obtained. Denote the j^{th} inverse as follows:

$$\begin{aligned}
x_1 &= h_{1j}(u_1, ..., u_n),\\
x_2 &= h_{2j}(u_1, ..., u_n),\\
&\vdots\\
x_n &= h_{nj}(u_1, ..., u_n).
\end{aligned} \tag{5}$$

For $(u_1, u_2, \ldots, u_n) \in \mathscr{B}$, the j^{th} inverse yields a unique $(x_1, x_2, \ldots, x_n) \in A_j$ such that $(u_1, u_2, \ldots, u_n) = (g_1(x_1, x_2, \ldots, x_n), \ldots, u_n = g_n(x_1, x_2, \ldots, x_n))$. Let J_j denote the Jacobian computed from the j^{th} inverse. Assuming that these Jacobians do not vanish identically on $\mathscr{B}$, we obtain the following representation of the joint probability density function $f_U(u_1, ..., u_n)$ for $u \in \mathscr{B}$ [38]:

$$f_U(u_1, ..., u_n) = \sum_{j=1}^{k} f_X\left(h_{1j}(u_1, ..., u_n), ..., h_{nj}(u_1, ..., u_n)\right)\left|J_j\right|. \tag{6}$$

To obtain the deterministic equivalent of chance constraint (3), transformation methods can be used. Let $a_{k1}, a_{k2}, ..., a_{kn}$ be independent random variables. Consider the random variables $(U_1, U_2, \ldots, U_n)$, which are defined as follows:

$$\begin{aligned}
U_1 &= a_{k1} x_1,\\
U_2 &= a_{k1} x_1 + a_{k2} x_2 = U_1 + a_{k2} x_2,\\
&\vdots\\
U_n &= a_{k1} x_1 + a_{k2} x_2 + \ldots + a_{kn} x_n = U_{n-1} + a_{kn} x_n.
\end{aligned} \tag{7}$$

By deriving the probability density functions of the random variables $U_1, U_2, ..., U_n$, the deterministic equivalents of chance constraints can be obtained. Hence, chance constraint (3) can be converted to a deterministic equivalent as follows:

$$F_{U_n}(b_k) \ge 1 - \delta_k. \tag{8}$$

3.2. Method II: Transformation of a Chance-Constrained Programming Model into a Deterministic Model Using Lyapunov's Central Limit Theorem. Let $X_1, X_2, ..., X_n$ be independent random variables. Suppose that each X_j has a finite expected value and finite variance. Suppose also that the third central moments satisfy $E(|X_j - E(X_j)|^3) < \infty$ $(j = 1, 2, ..., n)$ and the following Lyapunov condition [35]:

$$\frac{\lim_{n \longrightarrow \infty} \sum_{j=1}^{n} E\left(\left|X_j - E\left(X_j\right)\right|^3\right)}{\sqrt{\operatorname{Var}\left(\sum_{j=1}^{n} X_j\right)}} = 0. \tag{9}$$

Then, the normalized sum of the random variables X_j converges to a standard normal random variable as $n \longrightarrow \infty$.

$$\begin{aligned} P\left(\sum_{j=1}^{n} X_j \leq x\right) &= P\left(\frac{\sum_{j=1}^{n} X_j - E\sum_{j=1}^{n} X_j}{\sqrt{Var\left(\sum_{j=1}^{n} X_j\right)}} \leq \frac{x - E\sum_{j=1}^{n} X_j}{\sqrt{Var\left(\sum_{j=1}^{n} X_j\right)}}\right) \\ &\cong P\left(Z \leq \frac{x - E\sum_{j=1}^{n} X_j}{\sqrt{Var\left(\sum_{j=1}^{n} X_j\right)}}\right) \\ &= \Phi\left(\frac{x - E\sum_{j=1}^{n} X_j}{\sqrt{Var\left(\sum_{j=1}^{n} X_j\right)}}\right). \end{aligned} \tag{10}$$

The deterministic equivalent of chance constraint (3) can be obtained using normalization as follows:

$$\begin{gathered} \Phi\left(\frac{b_k - E\left(\sum_{j=1}^{n} a_{kj}x_j\right)}{\sqrt{Var\left(\sum_{j=1}^{n} a_{kj}x_j\right)}}\right) \geq 1 - \delta_k, \\ b_k \geq \sqrt{Var\left(\sum_{j=1}^{n} a_{kj}x_j\right)}\Phi^{-1}\left(1 - \delta_k\right) + E\left(\sum_{j=1}^{n} a_{kj}x_j\right). \end{gathered} \tag{11}$$

In the following section, we provide a deterministic equivalent of constraint (3) for the two methods mentioned above when coefficients a_{ij} are independent gamma variables with different parameters.

4. The Deterministic Model for the Gamma Distribution

Definition 1. Let $a_{k1}, a_{k2}, \ldots, a_{kn}$ be independent random variables with parameters $(\alpha_{kj}, \beta_{kj})$. The probability density function for each random variable a_{kj} can be stated as follows:

$$f\left(a_{kj}\right) = \begin{cases} \dfrac{1}{\Gamma\left(\alpha_{kj}\right)\beta_{kj}^{\alpha_{kj}}} a_{kj}^{\alpha_{kj}-1} \exp\left(\dfrac{-a_{kj}}{\beta_{kj}}\right), & a_{kj} > 0,\ \beta_{kj} > 0,\ \alpha_{kj} > 0, \\ 0, & ow. \end{cases} \tag{12}$$

Method I: we start by considering a mathematical model with only two random variables. For $j = 1, 2$, chance constraint (3) can be expressed as follows:

$$P\left(a_{k1}x_1 + a_{k2}x_2 \leq b_k\right) \geq 1 - \delta_k. \tag{13}$$

Note that a_{k1} and a_{k2} are independent gamma random variables with $\Gamma(\alpha_{k1}, \beta_{k1})$ and $\Gamma(\alpha_{k2}, \beta_{k2})$, respectively. To calculate the deterministic equivalent of chance constraint (13), we must first find the density of the random variable $a_{k1}x_1 + a_{k2}x_2$. Therefore, we define random variables U_1 and U_2 as follows:

$$\begin{aligned} U_1 &= a_{k1}x_1 = Y_1, \\ U_2 &= a_{k1}x_1 + a_{k2}x_2 = Y_1 + Y_2 = U_1 + Y_2. \end{aligned} \tag{14}$$

Note that the random variables Y_j have a gamma distribution, with $\Gamma(\alpha_{kj}, \beta_{kj}x_j) = \Gamma(\alpha_{kj}, \beta_{kj}^{*})$. Let V_2 be an auxiliary variable used to find the probability density function of U_2.

$$\begin{aligned} U_2 &= a_{k1}x_1 + a_{k2}x_2 = Y_1 + Y_2 = U_1 + Y_2, \\ V_2 &= Y_2 = a_{k2}x_2. \end{aligned} \tag{15}$$

Note that if $y_1 = u_2 - v_2$ and $y_2 = v_2$, then $|J| = 1$. The joint probability density function of U_2 and V_2 is as follows:

$$\begin{aligned} f_{U_2,V_2}\left(u_2, v_2\right) &= f_{Y_1}\left(u_2 - v_2\right) f_{Y_2}\left(v_2\right)|J| \\ &= \frac{\left(u_2 - v_2\right)^{\alpha_{k1}-1} v_2^{\alpha_{k2}-1}}{\Gamma\left(\alpha_{k1}\right)\Gamma\left(\alpha_{k2}\right)\left(\beta_{k1}^{*}\right)^{\alpha_{k1}}\left(\beta_{k2}^{*}\right)^{\alpha_{k2}}} \exp\left(-\frac{u_2}{\beta_{k1}^{*}}\right) \exp\left[-v_2\left(\frac{1}{\beta_{k2}^{*}} - \frac{1}{\beta_{k1}^{*}}\right)\right]. \end{aligned} \tag{16}$$

The marginal probability density function of U_2 can be computed as follows:

$$f_{U_2}(u_2) = \frac{\exp\left(-u_2/\beta^*_{k1}\right)}{\Gamma(\alpha_{k1})\Gamma(\alpha_{k2})\left(\beta^*_{k1}\right)^{\alpha_{k1}}\left(\beta^*_{k2}\right)^{\alpha_{k2}}} \cdot \int_0^{u_2} \left(u_2 - v_2\right)^{\alpha_{k1}-1} v_2^{\alpha_{k2}-1} \exp\left[-v_2\left(\frac{1}{\beta^*_{k2}} - \frac{1}{\beta^*_{k1}}\right)\right] dv_2. \tag{17}$$

After integrating the above integral, we obtain the following expression:

$$f_{U_2}(u_2) = \frac{u_2^{\alpha_{k1}+\alpha_{k2}-1}\exp\left(-u_2/\beta^*_{k1}\right)}{\left(\beta^*_{k1}\right)^{\alpha_{k1}}\left(\beta^*_{k2}\right)^{\alpha_{k2}}\Gamma\left(\alpha_{k1}+\alpha_{k2}\right)} \cdot hypergeom\left(\alpha_{k2}, \alpha_{k1} + \alpha\alpha_{k2}, -\left(\frac{1}{\beta^*_{k2}} - \frac{1}{\beta^*_{k1}}\right)u_2\right). \tag{18}$$

Note that the term *hypergeom* is the generalized hypergeometric function defined by

$$hypergeom[a, b; x] = F(a, b; x) = \sum_{k=0}^{\infty} \frac{(a)_k x^k}{(b)_k k!}, \quad (b \neq 0, -1, -2, \ldots), \tag{19}$$

with

$$(a)_k = a(a+1)\ldots(a+k-1) = \frac{\Gamma(a+k)}{\Gamma(a)}. \tag{20}$$

By substituting equation (19) into (18), the probability density function of U_2 can be rewritten as follows:

$$f_{U_2}(u_2) = \frac{u_2^{\alpha_{k1}+\alpha_{k2}-1}\exp\left(-\left(u_2/\beta^*_{k1}\right)\right)}{\left(\beta^*_{k1}\right)^{\alpha_{k1}}\left(\beta^*_{k2}\right)^{\alpha_{k2}}\Gamma\left(\alpha_{k1}+\alpha_{k2}\right)} \sum_{j=1}^{\infty} \frac{\left(\alpha_{k2}\right)_j\left(-u_2\left(\left(1/\beta^*_{k2}\right) - \left(1/\beta^*_{k1}\right)\right)\right)^j}{\left(\alpha_{k1}+\alpha_{k2}\right)_j j!}. \tag{21}$$

The distribution function of U_2, which is the left side of the chance constraint given in (13) can be rewritten as follows:

$$\begin{aligned} P\left(U_2 \leq b_k\right) &= F_{U_2}\left(b_k\right) \\ &= \frac{1}{\left(\beta^*_{k1}\right)^{\alpha_{k1}}\left(\beta^*_{k2}\right)^{\alpha_{k2}}\Gamma\left(\alpha_{k1}+\alpha_{k2}\right)} \int_0^{b_k} u_2^{\alpha_{k1}+\alpha_{k2}-1} \exp\left(-\frac{u_2}{\beta^*_{k1}}\right) \cdot \sum_{j=1}^{\infty} \frac{\left(\alpha_{k2}\right)_j\left(-u_2\left(\left(1/\beta^*_{k2}\right) - \left(1/\beta^*_{k1}\right)\right)\right)^j}{\left(\alpha_{k1}+\alpha_{k2}\right)_j j!} du_2. \end{aligned} \tag{22}$$

Thus, the deterministic equivalent of chance constraint (13) is as follows:

$$\frac{1}{\left(\beta_{k1}^{*}\right)^{\alpha_{k1}}\left(\beta_{k2}^{*}\right)^{\alpha_{k2}}\Gamma\left(\alpha_{k1}+\alpha_{k2}\right)}\int_{0}^{b_k} u_2^{\alpha_{k1}+\alpha_{k2}-1}\exp\left(-\frac{u_2}{\beta_{k1}^{*}}\right)\sum_{j=1}^{\infty}\frac{\left(\alpha_{k2}\right)_j\left(-u_2\left(\left(1/\beta_{k2}^{*}\right)-\left(1/\beta_{k1}^{*}\right)\right)\right)^j}{\left(\alpha_{k1}+\alpha_{k2}\right)_j j!}\mathrm{d}u_2 \geq 1-\delta_k. \tag{23}$$

We now consider a chance-constraint with three random variables. The k^{th} chance constraint involving these random variables can be presented as follows:

$$P\left(a_{k1}x_1 + a_{k2}x_2 + a_{k3}x_3 \leq b_k\right) \geq 1-\delta_k. \tag{24}$$

Note that a_{k1}, a_{k2}, and a_{k3} are gamma random variables with $\Gamma(\alpha_{k1},\beta_{k1})$, $\Gamma(\alpha_{k2},\beta_{k2})$, and $\Gamma(\alpha_{k3},\beta_{k3})$, respectively. We define the random variable U_3 and auxiliary variable V_3 as follows:

$$\begin{aligned} U_3 &= a_{k1}x_1 + a_{k2}x_2 + a_{k3}x_3 = Y_1 + Y_2 + Y_3 = U_2 + Y_3, \\ V_3 &= Y_3 = a_{k3}x_3. \end{aligned} \tag{25}$$

Note that if $u_2 = u_3 - v_3$ and $y_3 = v_3$, then $|J| = 1$. The joint probability density function of U_3 and V_3 is as follows:

$$\begin{aligned} f_{U_3,V_3}\left(u_3,v_3\right) &= f_{U_2}\left(u_3-v_3\right)F_{Y_3}\left(v_3\right)|J| \\ &= \frac{\left(u_3-v_3\right)^{\alpha_{k1}+\alpha_{k2}-1}\exp\left(-\left(\left(u_3-v_3\right)/\beta_{k1}^{*}\right)\right)\exp\left(-\left(v_3/\beta_{k3}^{*}\right)\right)v_3^{\alpha_{k3}-1}}{\left(\beta_{k1}^{*}\right)^{\alpha_{k1}}\left(\beta_{k2}^{*}\right)^{\alpha_{k2}}\left(\beta_{k3}^{*}\right)^{\alpha_{k3}}\Gamma\left(\alpha_{k1}+\alpha_{k2}\right)\Gamma\left(\alpha_{k3}\right)} \\ &\quad \cdot \sum_{j=1}^{\infty}\frac{\left(\alpha_{k2}\right)_j\left(-\left(u_3-v_3\right)\left(\left(1/\beta_{k2}^{*}\right)-\left(1/\beta_{k1}^{*}\right)\right)\right)^j}{\left(\alpha_{k1}+\alpha_{k2}\right)_j j!}. \end{aligned} \tag{26}$$

The marginal probability density function of U_3 can be computed as follows:

$$f_{U_3}\left(u_3\right) = \int_0^{u_3} f_{U_3,V_3}\left(u_3,v_3\right)\mathrm{d}v_3. \tag{27}$$

The distribution function of U_3, which is on the left side of the chance constraints given in (24), can be written as follows:

$$P\left(U_3 \leq b_k\right) = F_{U_3}\left(b_k\right) = \int_0^{b_k} f_{U_3}\left(u_3\right)\mathrm{d}u_3. \tag{28}$$

Now, we consider a chance constraint involving n random variables. To derive the marginal probability density function of U_n, we define the random variable U_n and auxiliary variable V_n as follows:

$$\begin{aligned} U_n &= a_{k1}x_1 + a_{k2}x_2 + \cdots + a_{kn}x_n = Y_1 + Y_2 + \cdots + Y_n = U_{n-1} + Y_n, \\ V_n &= Y_n = a_{kn}x_n. \end{aligned} \tag{29}$$

Note that if $u_{n-1} = u_n - v_n$ and $y_n = v_n$, then $|J| = 1$. The joint probability density function of U_n and V_n can be stated as follows:

$$f_{U_n,V_n}\left(u_n,v_n\right) = f_{U_{n-1}}\left(u_n-v_n\right)f_{Y_n}\left(v_n\right)|J|. \tag{30}$$

The marginal probability density function of U_n can be computed as follows:

$$f_{U_n}\left(u_n\right) = \int_0^{u_n} f_{U_n,V_n}\left(u_n,v_n\right)\mathrm{d}v_n. \tag{31}$$

Hence, the deterministic equivalent of the left side of chance constraint (3) is

$$P\left(U_n \leq b_k\right) = F_{U_n}\left(b_k\right) = \int_0^{b_k} f_{U_n}\left(u_n\right)\mathrm{d}u_n. \tag{32}$$

Thus, we have reached a generalized result for n random variables. The deterministic equivalent of chance constraint (3) can be easily calculated with equation (32) for known parameters of gamma distributions.

Method II: in this method, we apply Lyapunov's central limit theorem to the gamma distributions with large values of n. As such, we are able to transform chance constraint (3) to deterministic equivalent (11).

$$b_k - \sqrt{\sum_{j=1}^{n} \alpha_{kj}\beta_{kj}^2 x_j^2}\Phi^{-1}\left(1-\delta_k\right) - E\left(\sum_{j=1}^{n} \alpha_{kj}\beta_{kj}x_j\right) \geq 0. \tag{33}$$

5. A New Algorithm for Multiobjective Chance-Constrained Problem Procedure

We now present a methodology for solving a multiobjective chance-constrained stochastic programming model. To derive a deterministic model, we use the proposed Methods I and II, as presented in Section 4.

Step 1: convert the given chance-constraint programming problem into an equivalent deterministic programming problem as discussed in Section 4.

Step 2: solve the deterministic problem obtained in Step 1, using only one objective at a time in model (2) and ignoring the others.

Step 3: calculate $f_l(x^*)$, $l = 1, \ldots, K$, which is the feasible solution of each lth objective function with respect to $f_l(x)$. Find the corresponding values of the objective functions at each solution for comparison.

Step 4: using the solutions obtained from Step 3 and the deterministic constraint obtained from Step 1, solve the following global criterion problem:

$$\min z = \sum_{l=1}^{K}\left[\frac{f_l(x^*) - f_l(x)}{f_l(x^*)}\right]^p, \quad x \in X. \tag{34}$$

Set $p = 1$ or $p = 2$. While the value of p is increasing, total distance between $f_l(x^*)$ and $f_l(x)$ is decreasing.

6. Application of the Newly Proposed Method to an IoT End-System

The Internet of Things [39] is an Internet-based information infrastructure which enables several smart "things" to communicate with each other and servers. These "things" are in general small devices with computing and communicating skills where most of the time measures some value. They need a power supply to operate, and since there are distributed, they often rely on battery power. This energy limitation requires some kind of intelligent power consumption and thus a good way of battery usage optimization.

The battery usage optimization process has a number of constraints, and the result should maximize the total amount of data sending time. The data sending time is directly related to the accuracy of updating the current state of the nodes on the server. According to this, at each sending process, the node sends either the current measurement or its current location to the server. The measurement data contain a measured height in centimeters, and location data contain world coordinates of the node. Once the data are sent to the server, the value on the server is updated and the number of data sending processes provides a better result on the server side for further calculations. Thus, maximizing the data sending time directly improves the quality of server-side jobs. There are certain constraints while maximizing the data sending time. Height change can occur with a random frequency through a day. Similarly, device location can change with a random frequency. With these in mind, constraints are as follows.

The first constraint is as follows: height change occurs with a random frequency every day and the assumption is that the height change is a gamma random variable with the parameters provided in Table 1. Every time a static duration of time is required for sending process. Similarly, location change occurs with a random frequency every day, and every time a static duration of time is required for sending process. The total of this time cannot exceed 3 seconds a day because the battery has a 3-second limit daily.

The second constraint is as follows: height change occurs with a random frequency every day and the assumption is that the height change is a gamma random variable with the parameters provided in Table 1. Every time a static duration of time is required for sending process. Similarly, location change occurs with a random frequency every day, and every time a static duration of time is required for sending process. The total of this time cannot exceed 2 seconds without changing the measurement equipment because of its limitations.

The third constraint is as follows: height change occurs with a random frequency every day and the assumption is that the height change is a gamma random variable with the parameters provided in Table 1. Every time a static duration of time is required for sending process. Similarly, location change occurs with a random frequency every day, and every time a static duration of time is required for sending process. The total of this time cannot exceed 1 second on consecutive sending operations because the device cannot handle that much memory.

This application requires 3 objective functions. The first objective function includes sending the data only as integer values where the total transmitted data at each time are the least and the objective is maximizing these transmitted data overall. The second objective function includes sending the data as float values where the total transmitted data are more than just sending integer values at each time and the objective is maximizing these transmitted data overall. The third objective function includes sending the data with data overheads which include the data types such as measurement or location. The final one has the most transmitted data at each time, and the objective is maximizing these transmitted data overall.

The mathematical model of the above engineering problem with multiobjective, chance-constrained programming problem by using the methodology presented in Section 5 is given as follows:

Table 1: Characteristics of a_{ij}.

$(i,\ j)$	$E(a_{ij})$	$Var(a_{ij})$	$a_{ij} \sim \Gamma(\alpha_{ij}, \beta_{ij})$	$E(a_{ij}x_j)$	$Var(a_{ij}x_j)$	$a_{ij}x_j \sim \Gamma(\alpha_{ij}, \beta_{ij}x_j)$
$(1,1)$	1	1	$a_{11} \sim \Gamma(1,1)$	x_1	x_1^2	$a_{11}x_1 \sim \Gamma(1, x_1)$
$(1,2)$	1	1	$a_{12} \sim \Gamma(1,1)$	x_2	x_2^2	$a_{12}x_2 \sim \Gamma(1, x_2)$
$(2,1)$	1	1	$a_{21} \sim \Gamma(1,1)$	x_1	x_1^2	$a_{21}x_1 \sim \Gamma(1, x_1)$
$(2,2)$	3	9	$a_{22} \sim \Gamma(1,3)$	$3x_2$	$9x_2^2$	$a_{22}x_2 \sim \Gamma(1, 3x_2)$
$(3,1)$	2	2	$a_{31} \sim \Gamma(2,1)$	$2x_1$	$2x_1^2$	$a_{31}x_1 \sim \Gamma(2, x_1)$
$(3,2)$	2	4	$a_{32} \sim \Gamma(1,2)$	$2x_2$	$4x_2^2$	$a_{32}x_2 \sim \Gamma(1, 2x_2)$

$$
\begin{aligned}
\max z_1(x) &= \sum_{j=1}^{2} c_{1j}x_j = f_1(x), \\
\max z_2(x) &= \sum_{j=1}^{2} c_{2j}x_j = f_2(x), \\
\max z_3(x) &= \sum_{j=1}^{2} c_{3j}x_j = f_3(x), \\
&P\left(\sum_{j=1}^{2} a_{1j}x_j \le b_1\right) \ge 1 - \delta_1, \\
&P\left(\sum_{j=1}^{2} a_{2j}x_j \le b_2\right) \ge 1 - \delta_2, \\
&P\left(\sum_{j=1}^{2} a_{3j}x_j \le b_3\right) \ge 1 - \delta_3, \\
&x_j \ge 0, \quad j = 1, 2.
\end{aligned} \tag{35}
$$

Note that the objective function coefficients, the right-hand side values, and the specified probabilities are determined as $(c_{11}, c_{12}, c_{21}, c_{22}, c_{31}, c_{32}) = (1, 1, 5, 6, 6, 8)$, $(b_1, b_2, b_3) = (3, 2, 1)$, and $(\delta_1, \delta_2, \delta_3) = (0.4, 0.4, 0.25)$, respectively. In this problem, a_{kj} is a gamma random variable with the characteristics shown in Table 1.

From Method I, we obtain the following deterministic problem:

$$
\begin{aligned}
&\max z_1 = x_1 + x_2 = f_1, \\
&\max z_2 = 5x_1 + 6x_2 = f_2, \\
&\max z_3 = 6x_1 + 8x_2 = f_3, \\
&\left[-x_2 \exp\left(\frac{3}{x_1}\right) + x_1 \exp\left(\frac{3}{x_2}\right)\right] \exp\left[-3\left(\frac{x_1 + x_2}{x_1 x_2}\right)\right] + x_2 - x_1 \ge 0.6\left(x_2 - x_1\right), \\
&\left[-3x_2 \exp\left(\frac{2}{x_1}\right) + x_1 \exp\left(\frac{2}{3x_2}\right)\right] \exp\left[-2\left(\frac{x_1 + 3x_2}{3x_1 x_2}\right)\right] + 3x_2 - x_1 \ge 0.6\left(3x_2 - x_1\right), \\
&- 4 \exp\left[-\frac{x_1 - 2x_2}{2x_1 x_2}\right] x_2^2 \exp\left(-\frac{1}{x_1}\right) + 4 \exp\left(-\frac{1}{x_1}\right) x_1 x_2 + 2 \exp\left(-\frac{1}{x_1}\right) x_2 - \exp\left(-\frac{1}{x_1}\right) x_1^2 \\
&\quad + 4x_2^2 - 4x_1 x_2 + x_1^2 \ge 0.75\left(-2x_2 + x_1\right)^2, \\
&\qquad x_j > 0, \quad j = 1, 2.
\end{aligned} \tag{36}
$$

Note that nonnegative constraint $x_j \ge 0$ is replaced with $x_j > 0$ to avoid dividing by zero in equation (36). From Method II, we obtain the following deterministic problem:

$$\begin{aligned}
&\max z_1 = x_1 + x_2 = f_1,\\
&\max z_2 = 5x_1 + 6x_2 = f_2,\\
&\max z_3 = 6x_1 + 8x_2 = f_3,\\
&3 - x_1 - x_2 - 0.253347103\left(x_1^2 + x_2^2\right)^{0.5} \geq 0,\\
&2 - x_1 - 3x_2 - 0.253347103\left(x_1^2 + 9x_2^2\right)^{0.5} \geq 0,\\
&1 - 2x_1 - 2x_2 - 0.67449\left(2x_1^2 + 4x_2^2\right)^{0.5} \geq 0,\\
&\qquad x_j \geq 0, \quad j = 1, 2.
\end{aligned} \tag{37}$$

Then, we solve the deterministic models (36) and (37) for each objective function and derive the solution shown in Table 2.

Then, we apply the Global Criterion Method as the solution technique to consider the objectives together. Deterministic constraints that are obtained by these two methods were examined one by one for both $p = 1$ and $p = 2$ in the multiobjective model. The solutions obtained by substituting the values given in Table 2 in model (34) in Step 4 are shown in Table 3.

Goal programming (GP), proposed by Charnes and Cooper, is a classical method to solve multiobjective problems, which is widely used in the literature. This method uses an approach which converts a multiobjective problem into a single-objective problem. Thus, it is possible to achieve a single solution for the multiobjective problem [40]. The main idea of GP is to reach predefined goals for every single objective. Deviations of goals are tried to be minimized in GP because it is impossible to obtain a solution for every objective since the goals conflict with each other. Models (36) and (37) with d_l^- and d_l^+, $l = 1, 2, 3$, are the positive deviations and negative deviations that are modeled and solved as goal programming which is proposed as an alternative to the Global Criterion Method which is given in Table 3. The result of the goal programming solution is given in Table 4.

Membership function method is used for Pareto optimal solution. The procedure is as follows: first, structure membership function $\mu_l(f_l(x))$ is generated for every objective function $f_l(x)$. Second, $\mu_l(f_l(x))$ is used as the new objective function for the structure of the new multiobjective programming problem. Then, this converts the new multiobjective programming problem into a single-objective programming problem by using appropriate methods. Finally, single-objective programming problem is solved to get the optimal solution, which is also the Pareto optimal solution of the original multiobjective programming problem [41]. Structure of the membership function for every objective function is as follows:

$$\mu_l(f_l(x)) = \frac{f_l(x) - \min_{x\in X} f_l(x)}{\max_{x\in X} f_l(x) - \min_{x\in X} f_l(x)}, \quad l = 1, 2, \ldots, K. \tag{38}$$

Pareto optimal solutions for model (36) is given in Table 5.

Table 2: Model solutions.

		Method I (from model (36))	Method II (from model (37))
For f_1	f_1^*	0.40822855	0.3598632
	x_1	0.2041427	0.2399088
	x_2	0.2041427	0.1199544
	f_2	2.245570	1.919270
	f_3	2.857998	2.399088
For f_2	f_2^*	2.245570	1.946318
	x_1	0.2041427	0.1817437
	x_2	0.2041427	0.1729333
	f_1	0.40822855	0.3546770
	f_3	2.857998	2.473929
For f_3	f_3^*	2.888421	2.487224
	x_1	0.02543799	0.1397264
	x_2	0.3419742	0.2061083
	f_1	0.3674122	0.3458346
	f_2	2.179035	1.935281

Table 3: Solutions from the Global Criterion Method.

	Method I (from model (36))		Method II (from model (37))	
	$p = 1$	$p = 2$	$p = 1$	$p = 2$
min z	0.01963806	0.00018279	0.01886591	0.00017017
f_1	0.3795788	0.3802155	0.3560863	0.3565821
f_2	2.191725	2.190862	1.945743	1.945196
f_3	2.865135	2.860862	2.467140	2.464064
x_1	0.08574758	0.09043120	0.1907753	0.1942964
x_2	0.2938312	0.2897843	0.1653110	0.1622857

Table 4: Solutions from the goal programming.

	Method I (from model (36))	Method II (from model (37))
Min z	0.03803001	0.03132063
f_1	0.4082941	0.3518664
f_2	2.245617	1.944759
f_3	2.858059	2.482053
x_1	0.2041470	0.1664387
x_2	0.2041470	0.1854276
d_1^-	0.001705926	0.008133627
d_1^+	0.000000	0.000000
d_2^-	0.004382595	0.005240505
d_2^+	0.000000	0.000000
d_3^-	0.03194148	0.01794650
d_3^+	0.000000	0.000000

If model (36) is analyzed for Pareto optimal solutions, optimal solution for f_1 is obtained when $w_1 = 1$ and $w_2 = w_3 = 0$. According to this, functions are resulted as $f_1 = 0.408286$, $f_2 = 2.24557$, and $f_3 = 2.857998$. Optimal solution for $f_2$is obtained when $w_2 = 1$ and $w_1 = w_3 = 0$. According to this, functions are resulted as $f_1 = 0.405686$, $f_2 = 2.247886$, and $f_3 = 2.873029$. Optimal solution for f_3 is obtained when $w_3 = 1$ and $w_2 = w_1 = 0$. According to this, functions are resulted as $f_1 = 0.396843$, $f_2 = 2.236849$, and

Table 5: Pareto optimal solution for Method I (from model (36)).

w_1	w_2	w_3	f_1	f_2	f_3	x_1	x_2	Max z
0,4	0,4	0,2	0.408285	2.24557	2.857998	0.204143	0.204143	0.997662
0,2	0,8	0,0	0.407612	2.246736	2.863025	0.198935	0.208677	1.000124
0,0	1,0	0,0	0.405686	2.247886	2.873029	0.186228	0.219458	1.001144
0,2	0,6	0,2	0.40656	2.247679	2.869119	0.191678	0.214881	0.998278
0,2	0,4	0,4	0.405335	2.247857	2.874374	0.184152	0.221183	0.996816
0,6	0,0	0,4	0.40888	2.244004	2.852489	0.209273	0.199606	0.995368
0,6	0,2	0,2	0.409551	2.241286	2.844367	0.216022	0.19353	0.99809
0,4	0,0	0,6	0.406246	2.247804	2.870622	0.189674	0.216572	0.993855
0,0	0,2	0,8	0.399065	2.241	2.88574	0.153391	0.245674	0.998724
0,8	0,2	0,0	0.408285	2.24557	2.857998	0.204143	0.204143	0.997662
0,2	0,0	0,8	0.402318	2.245732	2.88219	0.168179	0.23414	0.995099
0,0	0,8	0,2	0.404335	2.247485	2.877629	0.178526	0.225809	0.999927
0,8	0,0	0,2	0.408285	2.24557	2.857998	0.204143	0.204143	0.997662
0,0	0,0	1,0	0.396843	2.236849	2.886325	0.144211	0.252633	0.999193
0,4	0,2	0,4	0.407349	2.247058	2.86472	0.197036	0.210313	0.995566
0,6	0,4	0,0	0.408286	2.24557	2.857998	0.204143	0.204143	0.997662
0,0	0,4	0,6	0.401041	2.244105	2.884045	0.162143	0.238899	0.998701
1,0	0,0	0,0	0.408286	2.24557	2.857998	0.204145	0.204140	0.999999
0,2	0,2	0,6	0.403926	2.247226	2.878749	0.176329	0.227597	0.995752
0,4	0,6	0,0	0.409065	2.243374	2.850487	0.211018	0.198047	1.00013
0,0	0,6	0,4	0.402792	2.246244	2.881319	0.170508	0.232284	0.999108

$f_3 = 2.886325$. Graphical representations of Pareto optimal solution for model (36) are given in Figure 1.

If model (37) is analyzed for Pareto optimal solutions, optimal solution for f_1 is obtained when $w_1 = 1$ and $w_2 = w_3 = 0$. According to this, functions are resulted as $f_1 = 0.3598632$, $f_2 = 1.919270$, and $f_3 = 2.399088$. Optimal solution for f_2 is obtained when $w_2 = 1$ and $w_1 = w_3 = 0$. According to this, functions are resulted as $f_1 = 0.3546770$, $f_2 = 1.946318$, and $f_3 = 2.473929$. Optimal solution for f_3 is obtained when $w_3 = 1$ and $w_2 = w_1 = 0$. According to this, functions are resulted as $f_1 = 0.3458346$, $f_2 = 1.935281$, and $f_3 = 2.487224$. Graphical representations of Pareto optimal solution for model (37) are given in Figure 1.

7. Results and Discussions

In this paper, we have considered a chance-constrained programming problem under the assumption that the random variables a_{ij} are independent with gamma distributions. The main purpose of this paper is to transform the chance-constrained model into a deterministic model based on two proposed methods. Method I provides exact solutions for the conversion of a chance constraint into a deterministic constraint. Using Method I, we found the probability density function of a linear combination of n independent random variables. Then, using the probability density function, the chance constraints are transformed into deterministic constraints. For large values of n, we proposed Method II, which is an approximation method. As can be seen in the example in Table 1, the numerical results for the exact and approximate models are rather close to each other. In this case, we first used Method I; however, because of the complex and difficult calculations in Method I, Method II can also be used.

As can be seen in Table 2, for multiobjective problems, we calculated the objective value by considering all objective functions and then summarized the solutions of the deterministic model under the two methods. We compared the solutions of the deterministic model according to the decision variables and objective functions. Even for the special case with n = 2, the objective values obtained from Method I and Method II are very similar. According to Lyapunov's central limit theorem, it is obvious that we can obtain more approximate solutions for larger values of n.

For the multiobjective case, solutions are presented in Table 5. In the case of p = 2, the decrease in deviations from the ideal solutions can be observed in objective values. As such, the numerical results from these two methods are very close to one another. For multiobjective problems, the Global Criterion Method provides more effective solutions after converting chance constraints to deterministic constraints.

Considering the results obtained from the real-life problem in Section 6, Table 3 is created as a summary. The multiobjective problems that are solved by Global Criterion Method are also solved by goal programming, and the results are given in Table 4. Also, Pareto optimal solutions for these problems are obtained, and the results are given in Tables 5 and 6, respectively. Graphical representations of these Pareto optimal solutions are also provided. According to the results provided in the table, optimal solutions for the tree objective functions can be summarized as follows:

> 1st objective function: Method I (p = 2) is the optimal solution, where $f_1 = 0.3802155 \cong 0.38$. The result demonstrates that, with sending minimum data as only integer value, the maximum total data sent can reach 0.38 MB of data.

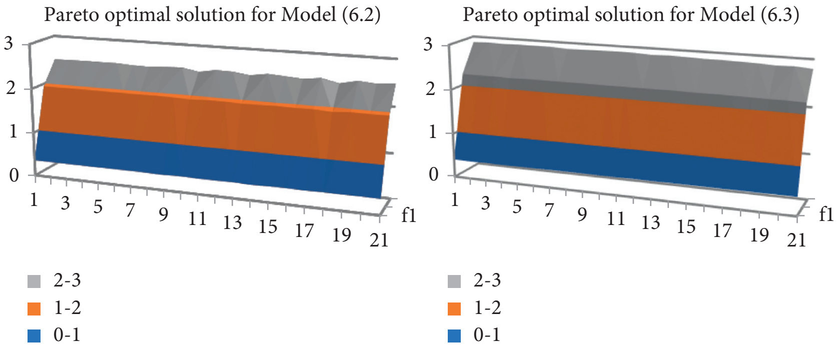

Figure 1: Graphical representations of the Pareto optimal solutions.

Table 6: Pareto optimal solution for Method II (from model (37)).

w_1	w_2	w_3	f_1	f_2	f_3	x_1	x_2	Max z
0,4	0,4	0,2	0.3572767	1.944002	2.458897	0.1996583	0.1576184	0.9943718
0,2	0,8	0,0	0.3566031	1.945168	2.463924	0.1944504	0.1621527	0.9977161
0,0	1,0	0,0	0.3546770	1.946318	2.473929	0.1817437	0.1729333	1.000001
0,2	0,6	0,2	0.3555507	1.946111	2.470018	0.1871939	0.1683569	0.9961564
0,2	0,4	0,4	0.3543261	1.946289	2.475273	0.1796676	0.1746585	0.9949954
0,6	0,0	0,4	0.3578708	1.942436	2.453389	0.2047889	0.1530819	0.9912371
0,6	0,2	0,2	0.3585426	1.939718	2.445266	0.2115374	0.1470052	0.9937465
0,4	0,0	0,6	0.3552376	1.946236	2.471521	0.1851899	0.1700477	0.9910710
0,0	0,2	0,8	0.3480565	1.939433	2.486639	0.1489064	0.1991501	0.9991052
0,8	0,2	0,0	0.3596667	1.928812	2.418957	0.2291883	0.1304784	0.9977646
0,2	0,0	0,8	0.3513097	1.944164	2.483090	0.1636941	0.1876156	0.9939172
0,0	0,8	0,2	0.3533263	1.945916	2.478527	0.1740416	0.1792848	0.9991363
0,8	0,0	0,2	0.3593769	1.933343	2.429178	0.2229186	0.1364584	0.9942518
0,0	0,0	1,0	0.3458346	1.935281	2.487224	0.1397263	0.2061083	1.000001
0,4	0,2	0,4	0.3563402	1.945490	2.465619	0.1925512	0.1637890	0.9925251
0,6	0,4	0,0	0.3590700	1.936333	2.436387	0.2180866	0.1409834	0.9966258
0,0	0,4	0,6	0.3500325	1.942537	2.484944	0.1576581	0.1923745	0.9986738
1,0	0,0	0,0	0.3598632	1.919270	2.399088	0.2399088	0.1199544	1.000000
0,2	0,2	0,6	0.3529170	1.945658	2.479647	0.1718445	0.1810726	0.9942446
0,4	0,6	0,0	0.3580567	1.941806	2.451386	0.2065339	0.1515228	0.9966015
0,0	0,6	0,4	0.3517832	1.944675	2.482218	0.1660240	0.1857593	0.9986893

2nd objective function: Method I ($p = 1$) is the optimal solution, where $f_2 = 2.191725 \cong 2.19$. The result demonstrates that, with sending medium size data as float values, the maximum total data sent can reach 2.19 MB of data.

3rd objective function: Method I ($p = 1$) is the optimal solution, where $f_3 = 2.865135 \cong 2.87$. The result demonstrates that, with sending minimum data as float values and their headers, the maximum total data sent can reach 2.87 MB of data.

The obtained results show that the methods provided can be applicable to such problems and the solutions prove that an optimal design can be achieved by our study. By comparing the values of objective functions, the designers may benefit the most applicable solution to their problems.

Data Availability

The datasets generated and/or analyzed during the current study are available from the corresponding author on reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] F. S. Hillier and G. J. Lieberman, *Introduction to Mathematical Programming*, Hill Publishing Company, New York, NY, USA, 1990.

[2] V. V. Kolbin, *Stochastic Programming*, D. Reidel Publishing Company, Boston, MA, USA, 1977.

[3] J. D. Pintér, “Global optimization: software, test problems, and applications,” *Nonconvex Optimization and Its Applications*, vol. 2, pp. 515–569, 2002.

[4] K. D. Atalay and A. Apaydin, “Gamma distribution approach in chance-constrained stochastic programming model,” *Journal of Inequalities and Applications*, vol. 2011, no. 1, p. 108, 2011.

[5] A. Charnes and W. W. Cooper, “Chance-constrained programming,” *Management Science*, vol. 6, no. 1, pp. 73–79, 1959.

[6] S. Hulsurkar, M. P. Biswal, and S. B. Sinha, “Fuzzy programming approach to multi-objective stochastic linear

programming problems," *Fuzzy Sets and Systems*, vol. 88, no. 2, pp. 173–181, 1997.

[7] R. Jagannathan, "Chance constrained programming with joint constraints," *Operations Research*, vol. 22, pp. 359–372, 1974.

[8] J. K. Sengupta, "Generalization of some distribution aspects of chance constrained linear programming," *International Economic Review*, vol. 11, no. 1, pp. 287–304, 1970.

[9] J. K. Sengupta, *Stochastic Programming: Methods and Applications*, Holland Publishing Company, Amsterdam, Netherlands, 1972.

[10] B. L. Miller and H. M. Wagner, "Chance constrained programming with joint constraints," *Operations Research*, vol. 13, no. 6, pp. 930–945, 1965.

[11] A. Prekopa, "On probabilistic constrained programming," *Mathematical Programming Study*, vol. 28, pp. 113–138, 1970.

[12] B. Contini, "A stochastic approach to goal programming," *Operations Research*, vol. 16, no. 3, pp. 576–586, 1968.

[13] L. M. Stancu-Minasian, *Stochastic Programming with Multiple Objective Functions*, D. Reidel Publishing Company, Boston, MA, USA, 1984.

[14] J.-P. Leclercq, "Stochastic programming: an interactive multicriteria approach," *European Journal of Operational Research*, vol. 10, no. 1, pp. 33–41, 1982.

[15] J. Teghem Jr., D. Dufrane, M. Thauvoye, and P. Kunsch, "Strange: an interactive method for multi-objective linear programming under uncertainty," *European Journal of Operational Research*, vol. 26, no. 1, pp. 65–82, 1986.

[16] A. Prekopa, *Stochastic Programming*, Kluwer Academic, Dordrecht, Netherlands, 1995.

[17] C. Mohan and H. T. Nguyen, "A fuzzifying approach to stochastic programming," *Opsearch*, vol. 34, no. 2, pp. 73–96, 1997.

[18] M. Sakawa, K. Kato, and I. Nishizaki, "An interactive fuzzy satisficing method for multiobjective stochastic linear programming problems through an expectation model," *European Journal of Operational Research*, vol. 145, no. 3, pp. 665–672, 2003.

[19] N. Yang and F. Wen, "A chance constrained programming approach to transmission system expansion planning," *Electric Power Systems Research*, vol. 75, no. 2-3, pp. 171–177, 2005.

[20] X. Huang, "Fuzzy chance-constrained portfolio selection," *Applied Mathematics and Computation*, vol. 177, no. 2, pp. 500–507, 2006.

[21] K Ağpak and H. Gökçen, "A chance constrained approach to stochastic line balancing problem," *European Journal of Operational Research*, vol. 180, pp. 1098–1115, 2007.

[22] R. Henrion and C. Strugarek, "Convexity of chance constraints with independent random variables," *Computational Optimization and Applications*, vol. 41, no. 2, pp. 263–276, 2008.

[23] P. Parpas, B. Rüstem, and E. N. Pistikopoulos, "Global optimization of robust chance constrained problems," *Journal of Global Optimization*, vol. 43, no. 2-3, pp. 231–247, 2009.

[24] Y. Xu, G. H. Huang, X. S. Qin, and M. F. Cao, "SRCCP: a Stochastic robust chance-constrained programming model for municipal solid waste management under uncertainty," *Resources, Conservation and Recycling*, vol. 53, no. 6, pp. 352–363, 2009.

[25] F. Ben Abdelaziz and H. Masri, "A compromise solution for the multiobjective stochastic linear programming under partial uncertainty," *European Journal of Operational Research*, vol. 202, no. 1, pp. 55–59, 2010.

[26] V. Goyal and R. Ravi, "A PTAS for the chance-constrained knapsack problem with random item sizes," *Operations Research Letters*, vol. 38, no. 3, pp. 161–164, 2010.

[27] B. Zhang and J. Peng, "Uncertain programming model for uncertain optimal assignment problem," *Applied Mathematical Modelling*, vol. 37, no. 9, pp. 6458–6468, 2013.

[28] L. Chen, J. Peng, and B. Zhang, "Uncertain goal programming models for bicriteria solid transportation problem," *Applied Soft Computing*, vol. 51, pp. 49–59, 2017.

[29] B. Zhang, J. Peng, S. Li, and L. Chen, "Fixed charge solid transportation problem in uncertain environment and its algorithm," *Computers & Industrial Engineering*, vol. 102, pp. 186–197, 2016.

[30] S. Majumder, S. Kar, and T. Pal, "Uncertain multi-objective Chinese postman problem," *Soft Computing*, vol. 23, no. 22, pp. 11557–11572, 2019.

[31] S. Majumder, S. Kar, and T. Pal, "Rough-fuzzy quadratic minimum spanning tree problem," *Expert Systems*, vol. 36, no. 2, Article ID e12364, 2019.

[32] V. Sarkar, K. Chaudhuri, and R. Mukherjee, "Mathematical programming problems having parameters as gamma random variables in chance constraints," *Advances in Inequalities and Applications*, vol. 5, 2016.

[33] L. M. Leemis, "Relationships among common univariate distributions," *The American Statistician*, vol. 40, no. 2, pp. 143–146, 1986.

[34] N. Balakrishna and C. D. Lai, "Bivariate gamma and related distributions," in *Continuous Bivariate Distributions*, Springer, New York, NY, USA, 2009.

[35] V. V. Petrov, *Sums of Independent Random Variables*, Springer-Verlag, New York, NY, USA, 1975.

[36] J. K. Sengupta, "Stochastic linear programming with chance constraints," *International Economic Review*, vol. 11, no. 1, pp. 101–116, 1970.

[37] S. B. Sinha, S. Hulsurkar, and M. P. Biswal, "Fuzzy programming approach to multi-objective stochastic programming problems when bi's follow joint normal distribution," *Fuzzy Sets and Systems*, vol. 109, no. 1, pp. 91–96, 2000.

[38] G. Castella and R. L. Berger, *Statistical Inference*, Duxbury Press, An Imprint of Wadsworth Publishing Company, Pacific Grove, CA, USA, 1990.

[39] K. Ashton, "That "internet of things" thing," *RFID Journal*, vol. 22, no. 7, pp. 97–114, 2009.

[40] A. Charnes and W. W. Cooper, *Management Models and the Industrial Applications of Linear Programming*, John Wiley, New York, NY, USA, 1961.

[41] G. Zhang and H. Zuo, "Pareto optimal solution analysis of convex multi-objective programming problem," *Journal of Networks*, vol. 8, no. 2, p. 437, 2013.