Chapter 4
Stochastic Programming

With the requirement of considering randomness, different types of stochastic programming have been developed to suit the different purposes of management. The first type of stochastic programming is the *expected value model*, which optimizes the expected objective functions subject to some expected constraints. The second, *chance-constrained programming*, was pioneered by Charnes and Cooper [37] as a means of handling uncertainty by specifying a confidence level at which it is desired that the stochastic constraint holds. After that, Liu [174] generalized chance-constrained programming to the case with not only stochastic constraints but also stochastic objectives. In practice, there usually exist multiple events in a complex stochastic decision system. Sometimes the decision-maker wishes to maximize the chance functions of satisfying these events. In order to model this type of problem, Liu [166] provided a theoretical framework of the third type of stochastic programming, called *dependent-chance programming*.

This chapter will give some basic concepts of probability theory and introduce a spectrum of stochastic programming. A hybrid intelligent algorithm is also documented.

4.1 Random Variables

Before introducing the concept of random variable, let us define a probability measure by an axiomatic approach.

Definition 4.1. *Let Ω be a nonempty set, and $\mathcal{A}$ a σ-algebra of subsets (called events) of Ω. The set function* Pr *is called a probability measure if*
Axiom 1. *(Normality)* $\Pr\{\Omega\} = 1$*;*
Axiom 2. *(Nonnegativity)* $\Pr\{A\} \ge 0$ *for any event A;*
Axiom 3. *(Countable Additivity) For every countable sequence of mutually disjoint events $\{A_i\}$, we have*

B. Liu: Theory and Practice of Uncertain Programming, STUDFUZZ 239, pp. 25–56.
springerlink.com

$$\Pr\left\{\bigcup_{i=1}^{\infty} A_i\right\} = \sum_{i=1}^{\infty} \Pr\{A_i\}. \tag{4.1}$$

Example 4.1. Let $\Omega = \{\omega_1, \omega_2, \cdots\}$, and let $\mathcal{A}$ be the power set of Ω. Assume that $p_1, p_2, \cdots$ are nonnegative numbers such that $p_1 + p_2 + \cdots = 1$. Define a set function on $\mathcal{A}$ as

$$\Pr\{A\} = \sum_{\omega_i \in A} p_i, \quad A \in \mathcal{A}. \tag{4.2}$$

Then Pr is a probability measure.

Example 4.2. Let $\Omega = [0, 1]$ and let $\mathcal{A}$ be the Borel algebra over Ω. If Pr is the Lebesgue measure, then Pr is a probability measure.

Example 4.3. Let ϕ be a nonnegative and integrable function on $\Re$ (the set of real numbers) such that $\int_{\Re} \phi(x)\mathrm{d}x = 1$. Then for any Borel set A, the set function

$$\Pr\{A\} = \int_A \phi(x)\mathrm{d}x \tag{4.3}$$

is a probability measure on $\Re$.

Theorem 4.1. *Let Ω be a nonempty set, $\mathcal{A}$ a σ-algebra over Ω, and* Pr *a probability measure. Then* $\Pr\{\emptyset\} = 0$ *and* $0 \le \Pr\{A\} \le 1$ *for any event A.*

Proof: Since $\emptyset$ and Ω are disjoint events and $\emptyset \cup \Omega = \Omega$, we have $\Pr\{\emptyset\} + \Pr\{\Omega\} = \Pr\{\Omega\}$ which makes $\Pr\{\emptyset\} = 0$. By the nonnegativity axiom, we have $\Pr\{A\} \ge 0$ for any event A. By the countable additivity axiom, we get $\Pr\{A\} = 1 - \Pr\{A^c\} \le 1$.

Definition 4.2. *Let Ω be a nonempty set, $\mathcal{A}$ a σ-algebra of subsets of Ω, and* Pr *a probability measure. Then the triplet* $(\Omega, \mathcal{A}, \Pr)$ *is called a probability space.*

Definition 4.3. *A random variable is a measurable function from a probability space* $(\Omega, \mathcal{A}, \Pr)$ *to the set of real numbers, i.e., for any Borel set B of real numbers, the set*

$$\{\xi \in B\} = \{\omega \in \Omega \mid \xi(\omega) \in B\} \tag{4.4}$$

is an event.

Definition 4.4. *Let $f : \Re^n \to \Re$ be a measurable function, and $\xi_1, \xi_2, \cdots, \xi_n$ random variables defined on the probability space* $(\Omega, \mathcal{A}, \Pr)$. *Then $\xi = f(\xi_1, \xi_2, \cdots, \xi_n)$ is a random variable defined by*

$$\xi(\omega) = f(\xi_1(\omega), \xi_2(\omega), \cdots, \xi_n(\omega)), \quad \forall \omega \in \Omega. \tag{4.5}$$

Definition 4.5. *An n-dimensional random vector is a measurable function from a probability space* $(\Omega, \mathcal{A}, \Pr)$ *to the set of n-dimensional real vectors, i.e., for any Borel set B of* $\Re^n$*, the set*

$$\{\boldsymbol{\xi} \in B\} = \left\{\omega \in \Omega \;\middle|\; \boldsymbol{\xi}(\omega) \in B\right\} \tag{4.6}$$

is an event.

Theorem 4.2. *The vector* $(\xi_1, \xi_2, \cdots, \xi_n)$ *is a random vector if and only if* $\xi_1, \xi_2, \cdots, \xi_n$ *are random variables.*

Proof: Write $\boldsymbol{\xi} = (\xi_1, \xi_2, \cdots, \xi_n)$. Suppose that $\boldsymbol{\xi}$ is a random vector on the probability space $(\Omega, \mathcal{A}, \Pr)$. For any Borel set B of $\Re$, the set $B \times \Re^{n-1}$ is also a Borel set of $\Re^n$. Thus we have

$$\begin{aligned}\left\{\omega \in \Omega \;\middle|\; \xi_1(\omega) \in B\right\} &= \left\{\omega \in \Omega \;\middle|\; \xi_1(\omega) \in B, \xi_2(\omega) \in \Re, \cdots, \xi_n(\omega) \in \Re\right\} \\ &= \left\{\omega \in \Omega \;\middle|\; \boldsymbol{\xi}(\omega) \in B \times \Re^{n-1}\right\} \in \mathcal{A}\end{aligned}$$

which implies that ξ_1 is a random variable. A similar process may prove that $\xi_2, \xi_3, \cdots, \xi_n$ are random variables. Conversely, suppose that all $\xi_1, \xi_2, \cdots, \xi_n$ are random variables on the probability space $(\Omega, \mathcal{A}, \Pr)$. We define

$$\mathcal{B} = \left\{B \subset \Re^n \;\middle|\; \{\omega \in \Omega | \boldsymbol{\xi}(\omega) \in B\} \in \mathcal{A}\right\}.$$

The vector $\boldsymbol{\xi} = (\xi_1, \xi_2, \cdots, \xi_n)$ is proved to be a random vector if we can prove that $\mathcal{B}$ contains all Borel sets of $\Re^n$. First, the class $\mathcal{B}$ contains all open intervals of $\Re^n$ because

$$\left\{\omega \in \Omega \;\middle|\; \boldsymbol{\xi}(\omega) \in \prod_{i=1}^{n}(a_i, b_i)\right\} = \bigcap_{i=1}^{n}\left\{\omega \in \Omega \;\middle|\; \xi_i(\omega) \in (a_i, b_i)\right\} \in \mathcal{A}.$$

Next, the class $\mathcal{B}$ is a σ-algebra of $\Re^n$ because (i) we have $\Re^n \in \mathcal{B}$ since $\{\omega \in \Omega | \boldsymbol{\xi}(\omega) \in \Re^n\} = \Omega \in \mathcal{A}$; (ii) if $B \in \mathcal{B}$, then $\{\omega \in \Omega | \boldsymbol{\xi}(\omega) \in B\} \in \mathcal{A}$, and

$$\{\omega \in \Omega \mid \boldsymbol{\xi}(\omega) \in B^c\} = \{\omega \in \Omega \mid \boldsymbol{\xi}(\omega) \in B\}^c \in \mathcal{A}$$

which implies that $B^c \in \mathcal{B}$; (iii) if $B_i \in \mathcal{B}$ for $i = 1, 2, \cdots$, then $\{\omega \in \Omega | \boldsymbol{\xi}(\omega) \in B_i\} \in \mathcal{A}$ and

$$\left\{\omega \in \Omega \;\middle|\; \boldsymbol{\xi}(\omega) \in \bigcup_{i=1}^{\infty} B_i\right\} = \bigcup_{i=1}^{\infty}\{\omega \in \Omega \mid \boldsymbol{\xi}(\omega) \in B_i\} \in \mathcal{A}$$

which implies that $\cup_i B_i \in \mathcal{B}$. Since the smallest σ-algebra containing all open intervals of $\Re^n$ is just the Borel algebra of $\Re^n$, the class $\mathcal{B}$ contains all Borel sets of $\Re^n$. The theorem is proved.

Probability Distribution

Definition 4.6. *The probability distribution Φ: $\Re \to [0,1]$ of a random variable ξ is defined by*

$$\Phi(x) = \Pr\left\{\omega \in \Omega \;\middle|\; \xi(\omega) \le x\right\}. \tag{4.7}$$

That is, $\Phi(x)$ is the probability that the random variable ξ takes a value less than or equal to x.

Definition 4.7. *The probability density function ϕ: $\Re \to [0,+\infty)$ of a random variable ξ is a function such that*

$$\Phi(x) = \int_{-\infty}^{x} \phi(y)\mathrm{d}y \tag{4.8}$$

holds for all $x \in \Re$, where Φ is the probability distribution of the random variable ξ.

Uniform Distribution: A random variable ξ has a uniform distribution if its probability density function is

$$\phi(x) = \begin{cases} \dfrac{1}{b-a}, & \text{if } a \le x \le b \\ 0, & \text{otherwise} \end{cases} \tag{4.9}$$

denoted by $\mathcal{U}(a,b)$, where a and b are given real numbers with $a < b$.

Exponential Distribution: A random variable ξ has an exponential distribution if its probability density function is

$$\phi(x) = \begin{cases} \dfrac{1}{\beta}\exp\left(-\dfrac{x}{\beta}\right), & \text{if } x \ge 0 \\ 0, & \text{otherwise} \end{cases} \tag{4.10}$$

denoted by $\mathcal{EXP}(\beta)$, where β is a positive number.

Normal Distribution: A random variable ξ has a normal distribution if its probability density function is

$$\phi(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \quad x \in \Re \tag{4.11}$$

denoted by $\mathcal{N}(\mu,\sigma^2)$, where μ and σ are real numbers.

Theorem 4.3 *(Probability Inversion Theorem)***.** *Let ξ be a random variable whose probability density function ϕ exists. Then for any Borel set B of $\Re$, we have*

$$\Pr\{\xi \in B\} = \int_B \phi(y)\mathrm{d}y. \tag{4.12}$$

Proof: Let $\mathcal{C}$ be the class of all subsets C of $\Re$ for which the relation

$$\Pr\{\xi \in C\} = \int_C \phi(y)\mathrm{d}y \tag{4.13}$$

holds. We will show that $\mathcal{C}$ contains all Borel sets of $\Re$. It follows from the probability continuity theorem and relation (4.13) that $\mathcal{C}$ is a monotone class. It is also clear that $\mathcal{C}$ contains all intervals of the form $(-\infty, a]$, $(a, b]$, (b, ∞) and $\emptyset$ since

$$\Pr\{\xi \in (-\infty, a]\} = \Phi(a) = \int_{-\infty}^{a} \phi(y)\mathrm{d}y,$$

$$\Pr\{\xi \in (b, +\infty)\} = \Phi(+\infty) - \Phi(b) = \int_{b}^{+\infty} \phi(y)\mathrm{d}y,$$

$$\Pr\{\xi \in (a, b]\} = \Phi(b) - \Phi(a) = \int_{a}^{b} \phi(y)\mathrm{d}y,$$

$$\Pr\{\xi \in \emptyset\} = 0 = \int_{\emptyset} \phi(y)\mathrm{d}y$$

where Φ is the probability distribution of ξ. Let $\mathcal{F}$ be the algebra consisting of all finite unions of disjoint sets of the form $(-\infty, a]$, $(a, b]$, (b, ∞) and $\emptyset$. Note that for any disjoint sets $C_1, C_2, \cdots, C_m$ of $\mathcal{F}$ and $C = C_1 \cup C_2 \cup \cdots \cup C_m$, we have

$$\Pr\{\xi \in C\} = \sum_{j=1}^{m} \Pr\{\xi \in C_j\} = \sum_{j=1}^{m} \int_{C_j} \phi(y)\mathrm{d}y = \int_C \phi(y)\mathrm{d}y.$$

That is, $C \in \mathcal{C}$. Hence we have $\mathcal{F} \subset \mathcal{C}$. Since the smallest σ-algebra containing $\mathcal{F}$ is just the Borel algebra of $\Re$, the monotone class theorem implies that $\mathcal{C}$ contains all Borel sets of $\Re$.

Example 4.4. Let ξ be a uniformly distributed random variable on $[a, b]$. Then for any number $c \in [a, b]$, it follows from probability inversion theorem that

$$\Pr\{\xi \le c\} = \int_a^c \phi(x)\mathrm{d}x = \int_a^c \frac{1}{b-a}\mathrm{d}x = \frac{c-a}{b-a}.$$

Independence

Definition 4.8. *The random variables* $\xi_1, \xi_2, \cdots, \xi_m$ *are said to be independent if*

$$\Pr\left\{\bigcap_{i=1}^{m} \{\xi_i \in B_i\}\right\} = \prod_{i=1}^{m} \Pr\{\xi_i \in B_i\} \tag{4.14}$$

for any Borel sets $B_1, B_2, \cdots, B_m$ *of real numbers.*

Theorem 4.4. *Let ξ_i be random variables with probability distributions Φ_i, $i = 1, 2, \cdots, m$, respectively, and Φ the probability distribution of the random vector $(\xi_1, \xi_2, \cdots, \xi_m)$. Then $\xi_1, \xi_2, \cdots, \xi_m$ are independent if and only if*

$$\Phi(x_1, x_2, \cdots, x_m) = \Phi_1(x_1)\Phi_2(x_2)\cdots\Phi_m(x_m) \tag{4.15}$$

for all $(x_1, x_2, \cdots, x_m) \in \Re^m$.

Proof: If $\xi_1, \xi_2, \cdots, \xi_m$ are independent random variables, then we have

$$\begin{aligned}\Phi(x_1, x_2, \cdots, x_m) &= \Pr\{\xi_1 \le x_1, \xi_2 \le x_2, \cdots, \xi_m \le x_m\}\\ &= \Pr\{\xi_1 \le x_1\}\Pr\{\xi_2 \le x_2\}\cdots\Pr\{\xi_m \le x_m\}\\ &= \Phi_1(x_1)\Phi_2(x_2)\cdots\Phi_m(x_m)\end{aligned}$$

for all $(x_1, x_2, \cdots, x_m) \in \Re^m$. Conversely, assume that (4.15) holds. Let $x_2, x_3, \cdots, x_m$ be fixed real numbers, and $\mathcal{C}$ the class of all subsets C of $\Re$ for which the relation

$$\Pr\{\xi_1 \in C, \xi_2 \le x_2, \cdots, \xi_m \le x_m\} = \Pr\{\xi_1 \in C\}\prod_{i=2}^{m}\Pr\{\xi_i \le x_i\} \tag{4.16}$$

holds. We will show that $\mathcal{C}$ contains all Borel sets of $\Re$. It follows from the probability continuity theorem and relation (4.16) that $\mathcal{C}$ is a monotone class. It is also clear that $\mathcal{C}$ contains all intervals of the form $(-\infty, a]$, $(a, b]$, (b, ∞) and $\emptyset$. Let $\mathcal{F}$ be the algebra consisting of all finite unions of disjoint sets of the form $(-\infty, a]$, $(a, b]$, (b, ∞) and $\emptyset$. Note that for any disjoint sets $C_1, C_2, \cdots, C_k$ of $\mathcal{F}$ and $C = C_1 \cup C_2 \cup \cdots \cup C_k$, we have

$$\begin{aligned}&\Pr\{\xi_1 \in C, \xi_2 \le x_2, \cdots, \xi_m \le x_m\}\\ =&\sum_{j=1}^{m}\Pr\{\xi_1 \in C_j, \xi_2 \le x_2, \cdots, \xi_m \le x_m\}\\ =&\Pr\{\xi_1 \in C\}\Pr\{\xi_2 \le x_2\}\cdots\Pr\{\xi_m \le x_m\}.\end{aligned}$$

That is, $C \in \mathcal{C}$. Hence we have $\mathcal{F} \subset \mathcal{C}$. Since the smallest σ-algebra containing $\mathcal{F}$ is just the Borel algebra of $\Re$, the monotone class theorem implies that $\mathcal{C}$ contains all Borel sets of $\Re$. Applying the same reasoning to each ξ_i in turn, we obtain the independence of the random variables.

Theorem 4.5. *Let ξ_i be random variables with probability density functions ϕ_i, $i = 1, 2, \cdots, m$, respectively, and ϕ the probability density function of the random vector $(\xi_1, \xi_2, \cdots, \xi_m)$. Then $\xi_1, \xi_2, \cdots, \xi_m$ are independent if and only if*

$$\phi(x_1, x_2, \cdots, x_m) = \phi_1(x_1)\phi_2(x_2)\cdots\phi_m(x_m) \tag{4.17}$$

for almost all $(x_1, x_2, \cdots, x_m) \in \Re^m$.

Proof: If $\phi(x_1, x_2, \cdots, x_m) = \phi_1(x_1)\phi_2(x_2)\cdots\phi_m(x_m)$ a.e., then we have

$$\begin{aligned}\Phi(x_1, x_2, \cdots, x_m) &= \int_{-\infty}^{x_1}\int_{-\infty}^{x_2}\cdots\int_{-\infty}^{x_m} \phi(t_1, t_2, \cdots, t_m)\mathrm{d}t_1\mathrm{d}t_2\cdots\mathrm{d}t_m \\ &= \int_{-\infty}^{x_1}\int_{-\infty}^{x_2}\cdots\int_{-\infty}^{x_m} \phi_1(t_1)\phi_2(t_2)\cdots\phi_m(t_m)\mathrm{d}t_1\mathrm{d}t_2\cdots\mathrm{d}t_m \\ &= \int_{-\infty}^{x_1}\phi_1(t_1)\mathrm{d}t_1\int_{-\infty}^{x_2}\phi_2(t_2)\mathrm{d}t_2\cdots\int_{-\infty}^{x_m}\phi_m(t_m)\mathrm{d}t_m \\ &= \Phi_1(x_1)\Phi_2(x_2)\cdots\Phi_m(x_m)\end{aligned}$$

for all $(x_1, x_2, \cdots, x_m) \in \Re^m$. Thus $\xi_1, \xi_2, \cdots, \xi_m$ are independent. Conversely, if $\xi_1, \xi_2, \cdots, \xi_m$ are independent, then for any $(x_1, x_2, \cdots, x_m) \in \Re^m$, we have $\Phi(x_1, x_2, \cdots, x_m) = \Phi_1(x_1)\Phi_2(x_2)\cdots\Phi_m(x_m)$. Hence

$$\Phi(x_1, x_2, \cdots, x_m) = \int_{-\infty}^{x_1}\int_{-\infty}^{x_2}\cdots\int_{-\infty}^{x_m} \phi_1(t_1)\phi_2(t_2)\cdots\phi_m(t_m)\mathrm{d}t_1\mathrm{d}t_2\cdots\mathrm{d}t_m$$

which implies that $\phi(x_1, x_2, \cdots, x_m) = \phi_1(x_1)\phi_2(x_2)\cdots\phi_m(x_m)$ a.e.

Example 4.5. Let $\xi_1, \xi_2, \cdots, \xi_m$ be independent random variables with probability density functions $\phi_1, \phi_2, \cdots, \phi_m$, respectively, and $f : \Re^m \to \Re$ a measurable function. Then for any Borel set B of real numbers, the probability $\Pr\{f(\xi_1, \xi_2, \cdots, \xi_m) \in B\}$ is

$$\int\int\cdots\int_{f(x_1,x_2,\cdots,x_m)\in B} \phi_1(x_1)\phi_2(x_2)\cdots\phi_m(x_m)\mathrm{d}x_1\mathrm{d}x_2\cdots\mathrm{d}x_m.$$

Expected Value

Expected value is the average value of random variable in the sense of probability measure. It may be defined as follows.

Definition 4.9. *Let ξ be a random variable. Then the expected value of ξ is defined by*

$$E[\xi] = \int_0^{+\infty} \Pr\{\xi \ge r\}\mathrm{d}r - \int_{-\infty}^0 \Pr\{\xi \le r\}\mathrm{d}r \tag{4.18}$$

provided that at least one of the two integrals is finite.

Let ξ and η be random variables with finite expected values. For any numbers a and b, it has been proved that $E[a\xi + b\eta] = aE[\xi] + bE[\eta]$. That is, the expected value operator has the linearity property.

Example 4.6. Assume that ξ is a discrete random variable taking values x_i with probabilities p_i, $i = 1, 2, \cdots, m$, respectively. It follows from the definition of expected value operator that

$$E[\xi] = \sum_{i=1}^{m} p_i x_i.$$

Theorem 4.6. *Let ξ be a random variable whose probability density function ϕ exists. If the Lebesgue integral*

$$\int_{-\infty}^{+\infty} x\phi(x)\mathrm{d}x$$

is finite, then we have

$$E[\xi] = \int_{-\infty}^{+\infty} x\phi(x)\mathrm{d}x. \tag{4.19}$$

Proof: It follows from Definition 4.9 and Fubini Theorem that

$$\begin{aligned}
E[\xi] &= \int_{0}^{+\infty} \Pr\{\xi \ge r\}\mathrm{d}r - \int_{-\infty}^{0} \Pr\{\xi \le r\}\mathrm{d}r \\
&= \int_{0}^{+\infty} \left[\int_{r}^{+\infty} \phi(x)\mathrm{d}x\right] \mathrm{d}r - \int_{-\infty}^{0} \left[\int_{-\infty}^{r} \phi(x)\mathrm{d}x\right] \mathrm{d}r \\
&= \int_{0}^{+\infty} \left[\int_{0}^{x} \phi(x)\mathrm{d}r\right] \mathrm{d}x - \int_{-\infty}^{0} \left[\int_{x}^{0} \phi(x)\mathrm{d}r\right] \mathrm{d}x \\
&= \int_{0}^{+\infty} x\phi(x)\mathrm{d}x + \int_{-\infty}^{0} x\phi(x)\mathrm{d}x \\
&= \int_{-\infty}^{+\infty} x\phi(x)\mathrm{d}x.
\end{aligned}$$

The theorem is proved.

Example 4.7. Let ξ be a uniformly distributed random variable on the interval $[a, b]$. Then its expected value is

$$E[\xi] = \int_{a}^{b} \frac{x}{b-a}\mathrm{d}x = \frac{a+b}{2}$$

.

Example 4.8. Let ξ be an exponentially distributed random variable $\mathcal{EXP}(\beta)$. Then its expected value is

$$E[\xi] = \int_{0}^{+\infty} \frac{x}{\beta} \exp\left(-\frac{x}{\beta}\right) \mathrm{d}x = \beta.$$

Example 4.9. Let ξ be a normally distributed random variable $\mathcal{N}(\mu, \sigma^2)$. Then its expected value is

$$E[\xi] = \int_{-\infty}^{+\infty} \frac{x}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \mathrm{d}x = \mu.$$

Critical Values

Let ξ be a random variable. In order to measure it, we may use its expected value. Alternately, we may employ α-optimistic value and α-pessimistic value as a ranking measure.

Definition 4.10. *Let ξ be a random variable, and $\alpha \in (0,1]$. Then*

$$\xi_{\sup}(\alpha) = \sup\left\{r \;\middle|\; \Pr\{\xi \geq r\} \geq \alpha\right\} \tag{4.20}$$

is called the α-optimistic value of ξ; and

$$\xi_{\inf}(\alpha) = \inf\left\{r \;\middle|\; \Pr\{\xi \leq r\} \geq \alpha\right\} \tag{4.21}$$

is called the α-pessimistic value of ξ.

This means that the random variable ξ will reach upwards of the α-optimistic value $\xi_{\sup}(\alpha)$ at least α of time, and will be below the α-pessimistic value $\xi_{\inf}(\alpha)$ at least α of time.

Theorem 4.7. *Let ξ be a random variable. Then we have*
(a) $\xi_{\inf}(\alpha)$ is an increasing and left-continuous function of α;
(b) $\xi_{\sup}(\alpha)$ is a decreasing and left-continuous function of α.

Proof: (a) It is easy to prove that $\xi_{\inf}(\alpha)$ is an increasing function of α. Next, we prove the left-continuity of $\xi_{\inf}(\alpha)$ with respect to α. Let $\{\alpha_i\}$ be an arbitrary sequence of positive numbers such that $\alpha_i \uparrow \alpha$. Then $\{\xi_{\inf}(\alpha_i)\}$ is an increasing sequence. If the limitation is equal to $\xi_{\inf}(\alpha)$, then the left-continuity is proved. Otherwise, there exists a number z^* such that

$$\lim_{i\to\infty} \xi_{\inf}(\alpha_i) < z^* < \xi_{\inf}(\alpha).$$

Thus $\Pr\{\xi \leq z^*\} \geq \alpha_i$ for each i. Letting $i \to \infty$, we get $\Pr\{\xi \leq z^*\} \geq \alpha$. Hence $z^* \geq \xi_{\inf}(\alpha)$. A contradiction proves the left-continuity of $\xi_{\inf}(\alpha)$ with respect to α. The part (b) may be proved similarly.

Ranking Criteria

Let ξ and η be two random variables. Different from the situation of real numbers, there does not exist a natural ordership in a random world. Thus an important problem appearing in this area is how to rank random variables. Here we give four ranking criteria.

Expected Value Criterion: We say $\xi > \eta$ if and only if $E[\xi] > E[\eta]$, where E is the expected value operator of random variables.

Optimistic Value Criterion: We say $\xi > \eta$ if and only if, for some predetermined confidence level $\alpha \in (0,1]$, we have $\xi_{\sup}(\alpha) > \eta_{\sup}(\alpha)$, where $\xi_{\sup}(\alpha)$ and $\eta_{\sup}(\alpha)$ are the α-optimistic values of ξ and η, respectively.

Pessimistic Value Criterion: We say $\xi > \eta$ if and only if, for some predetermined confidence level $\alpha \in (0,1]$, we have $\xi_{\inf}(\alpha) > \eta_{\inf}(\alpha)$, where $\xi_{\inf}(\alpha)$ and $\eta_{\inf}(\alpha)$ are the α-pessimistic values of ξ and η, respectively.

Probability Criterion: We say $\xi > \eta$ if and only if $\Pr\{\xi \ge \overline{r}\} > \Pr\{\eta \ge \overline{r}\}$ for some predetermined level $\overline{r}$.

Random Number Generation

Random number generation is a very important issue in Monte Carlo simulation. Generally, let ξ be a random variable with a probability distribution $\Phi(\cdot)$. Since $\Phi(\cdot)$ is an increasing function, the inverse function $\Phi^{-1}(\cdot)$ is defined on $[0,1]$. Assume that u is a uniformly distributed random variable on the interval $[0,1]$. Then we have

$$\Pr\left\{\Phi^{-1}(u) \le y\right\} = \Pr\{u \le \Phi(y)\} = \Phi(y) \tag{4.22}$$

which proves that the variable $\xi = \Phi^{-1}(u)$ has the probability distribution $\Phi(\cdot)$. In order to get a random variable ξ with probability distribution $\Phi(\cdot)$, we can produce a uniformly distributed random variable u from the interval $[0,1]$, and ξ is assigned to be $\Phi^{-1}(u)$. The above process is called the *inverse transform method*. But for the main known distributions, instead of using the inverse transform method, we have direct generating processes. For detailed expositions, the interested readers may consult Fishman [67], Law and Kelton [147], Bratley et al. [23], Rubinstein [268], and Liu [181]. Here we give some generating methods for probability distributions frequently used in this book.

The subfunction of generating pseudorandom numbers has been provided by the C library for any type of computer, defined as

```
int rand(void)
```

which produces a pseudorandom integer between 0 and RAND_MAX, where RAND_MAX is defined in stdlib.h as $2^{15} - 1$. Thus the uniform distribution, exponential distribution, and normal distribution can be generated by the following way:

Algorithm 4.1 (Uniform Distribution $\mathcal{U}(a,b)$)
Step 1. $u = \text{rand}(\,)$.
Step 2. $u \leftarrow u/\text{RAND\_MAX}$.
Step 3. Return $a + u(b - a)$.

Algorithm 4.2 (Exponential Distribution $\mathcal{EXP}(\beta)$)
Step 1. Generate u from $\mathcal{U}(0,1)$.
Step 2. Return $-\beta\ln(u)$.

Algorithm 4.3 (Normal Distribution $\mathcal{N}(\mu,\sigma^2)$)
Step 1. Generate μ_1 and μ_2 from $\mathcal{U}(0,1)$.
Step 2. $y=[-2\ln(\mu_1)]^{\frac{1}{2}}\sin(2\pi\mu_2)$.
Step 3. Return $\mu+\sigma y$.

4.2 Expected Value Model

The first type of stochastic programming is the so-called *expected value model* (EVM), which optimizes some expected objective function subject to some expected constraints, for example, minimizing expected cost, maximizing expected profit, and so forth.

Now let us recall the well-known newsboy problem in which a boy operating a news stall has to determine the number x of newspapers to order in advance from the publisher at a cost of \$$c$/newspaper every day. It is known that the selling price is \$$a$/newspaper. However, if the newspapers are not sold at the end of the day, then the newspapers have a small value of \$$b$/newspaper at the recycling center. Assume that the demand for newspapers is denoted by ξ in a day, then the number of newspapers at the end of the day is clearly $x-\xi$ if $x>\xi$, or 0 if $x\le\xi$. Thus the profit of the newsboy should be

$$f(x,\xi)=\begin{cases}(a-c)x, & \text{if } x\le\xi\\ (b-c)x+(a-b)\xi, & \text{if } x>\xi.\end{cases}$$

In practice, the demand ξ for newspapers is usually a stochastic variable, so is the profit function $f(x,\xi)$. Since we cannot predict how profitable the decision of ordering x newspapers will actually be, a natural idea is to employ the expected profit $E[f(x,\xi)]$. The newsboy problem is related to determining the optimal integer number x of newspapers such that the expected profit $E[f(x,\xi)]$ achieves the maximal value, i.e.,

$$\begin{cases}\max\ E[f(x,\xi)]\\ \text{subject to:}\\ \qquad x\ge 0,\quad \text{integer.}\end{cases}$$

This is a typical example of EVM. Generally, if we want to find a decision with maximum expected return subject to some expected constraints, then we have the following EVM,

$$\begin{cases} \max\ E[f(\boldsymbol{x},\boldsymbol{\xi})] \\ \text{subject to:} \\ \quad E[g_j(\boldsymbol{x},\boldsymbol{\xi})] \le 0,\ j=1,2,\cdots,p \end{cases} \tag{4.23}$$

where $\boldsymbol{x}$ is a decision vector, $\boldsymbol{\xi}$ is a stochastic vector, $f(\boldsymbol{x},\boldsymbol{\xi})$ is the return function, $g_j(\boldsymbol{x},\boldsymbol{\xi})$ are stochastic constraint functions for $j=1,2,\cdots,p$.

Definition 4.11. *A solution $\boldsymbol{x}$ is feasible if and only if $E[g_j(\boldsymbol{x},\boldsymbol{\xi})] \le 0$ for $j=1,2,\cdots,p$. A feasible solution $\boldsymbol{x}^*$ is an optimal solution to EVM (4.23) if $E[f(\boldsymbol{x}^*,\boldsymbol{\xi})] \ge E[f(\boldsymbol{x},\boldsymbol{\xi})]$ for any feasible solution $\boldsymbol{x}$.*

In many cases, there are multiple objectives. Thus we have to employ the following expected value multiobjective programming (EVMOP),

$$\begin{cases} \max\ \left[E[f_1(\boldsymbol{x},\boldsymbol{\xi})], E[f_2(\boldsymbol{x},\boldsymbol{\xi})], \cdots, E[f_m(\boldsymbol{x},\boldsymbol{\xi})]\right] \\ \text{subject to:} \\ \quad E[g_j(\boldsymbol{x},\boldsymbol{\xi})] \le 0,\ j=1,2,\cdots,p \end{cases} \tag{4.24}$$

where $f_i(\boldsymbol{x},\boldsymbol{\xi})$ are return functions for $i=1,2,\cdots,m$.

Definition 4.12. *A feasible solution $\boldsymbol{x}^*$ is said to be a Pareto solution to EVMOP (4.24) if there is no feasible solution $\boldsymbol{x}$ such that*

$$E[f_i(\boldsymbol{x},\boldsymbol{\xi})] \ge E[f_i(\boldsymbol{x}^*,\boldsymbol{\xi})], \quad i=1,2,\cdots,m \tag{4.25}$$

and $E[f_j(\boldsymbol{x},\boldsymbol{\xi})] > E[f_j(\boldsymbol{x}^*,\boldsymbol{\xi})]$ for at least one index j.

We can also formulate a stochastic decision system as an expected value goal programming (EVGP) according to the priority structure and target levels set by the decision-maker:

$$\begin{cases} \min \sum_{j=1}^{l} P_j \sum_{i=1}^{m} (u_{ij} d_i^+ \vee 0 + v_{ij} d_i^- \vee 0) \\ \text{subject to:} \\ \quad E[f_i(\boldsymbol{x},\boldsymbol{\xi})] - b_i = d_i^+,\ i=1,2,\cdots,m \\ \quad b_i - E[f_i(\boldsymbol{x},\boldsymbol{\xi})] = d_i^-,\ i=1,2,\cdots,m \\ \quad E[g_j(\boldsymbol{x},\boldsymbol{\xi})] \le 0, \qquad j=1,2,\cdots,p \end{cases} \tag{4.26}$$

where P_j is the preemptive priority factor which expresses the relative importance of various goals, $P_j \gg P_{j+1}$, for all j, u_{ij} is the weighting factor corresponding to positive deviation for goal i with priority j assigned, v_{ij} is the weighting factor corresponding to negative deviation for goal i with priority j assigned, $d_i^+ \vee 0$ is the positive deviation from the target of goal i, $d_i^- \vee 0$ is the negative deviation from the target of goal i, f_i is a function in goal constraints, g_j is a function in real constraints, b_i is the target value according to goal i, l is the number of priorities, m is the number of goal constraints, and p is the number of real constraints.

4.3 Chance-Constrained Programming

As the second type of stochastic programming developed by Charnes and Cooper [37], chance-constrained programming (CCP) offers a powerful means of modeling stochastic decision systems with assumption that the stochastic constraints will hold at least α of time, where α is referred to as the *confidence level* provided as an appropriate safety margin by the decision-maker. After that, Liu [174] generalized CCP to the case with not only stochastic constraints but also stochastic objectives.

Assume that $\boldsymbol{x}$ is a decision vector, $\boldsymbol{\xi}$ is a stochastic vector, $f(\boldsymbol{x}, \boldsymbol{\xi})$ is a return function, and $g_j(\boldsymbol{x}, \boldsymbol{\xi})$ are stochastic constraint functions, $j = 1, 2, \cdots, p$. Since the stochastic constraints $g_j(\boldsymbol{x}, \boldsymbol{\xi}) \leq 0, j = 1, 2, \cdots, p$ do not define a deterministic feasible set, it is desired that the stochastic constraints hold with a confidence level α. Thus we have a chance constraint as follows,

$$\Pr\left\{g_j(\boldsymbol{x}, \boldsymbol{\xi}) \leq 0, j = 1, 2, \cdots, p\right\} \geq \alpha \tag{4.27}$$

which is called a joint chance constraint.

Definition 4.13. *A point $\boldsymbol{x}$ is called feasible if and only if the probability measure of the event $\{g_j(\boldsymbol{x}, \boldsymbol{\xi}) \leq 0, j = 1, 2, \cdots, p\}$ is at least α.*

In other words, the constraints will be violated at most $(1 - \alpha)$ of time. Sometimes, the joint chance constraint is separately considered as

$$\Pr\{g_j(\boldsymbol{x}, \boldsymbol{\xi}) \leq 0\} \geq \alpha_j, \quad j = 1, 2, \cdots, p \tag{4.28}$$

which is referred to as a separate chance constraint.

Maximax Chance-Constrained Programming

In a stochastic environment, in order to maximize the optimistic return with a given confidence level subject to some chance constraint, Liu [174] gave the following CCP:

$$\begin{cases} \max\limits_{\boldsymbol{x}} \max\limits_{\overline{f}} \overline{f} \\ \text{subject to:} \\ \qquad \Pr\left\{f(\boldsymbol{x}, \boldsymbol{\xi}) \geq \overline{f}\right\} \geq \beta \\ \qquad \Pr\left\{g_j(\boldsymbol{x}, \boldsymbol{\xi}) \leq 0, j = 1, 2, \cdots, p\right\} \geq \alpha \end{cases} \tag{4.29}$$

where α and β are the predetermined confidence levels, and $\max \overline{f}$ is the β-optimistic return.

In practice, we may have multiple objectives. Thus we have to employ the following chance-constrained multiobjective programming (CCMOP),

$$\begin{cases} \max\limits_{\boldsymbol{x}} \left[\max\limits_{\overline{f}_1} \overline{f}_1, \max\limits_{\overline{f}_2} \overline{f}_2, \cdots, \max\limits_{\overline{f}_m} \overline{f}_m \right] \\ \text{subject to:} \\ \qquad \Pr\left\{ f_i(\boldsymbol{x}, \boldsymbol{\xi}) \ge \overline{f}_i \right\} \ge \beta_i, \; i = 1, 2, \cdots, m \\ \qquad \Pr\left\{ g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0 \right\} \ge \alpha_j, \;\; j = 1, 2, \cdots, p \end{cases} \tag{4.30}$$

where $\alpha_1, \alpha_2, \cdots, \alpha_p, \beta_1, \beta_2, \cdots, \beta_m$ are the predetermined confidence levels, and $\max \overline{f}_i$ are the β_i-optimistic values to the ith return functions $f_i(\boldsymbol{x}, \boldsymbol{\xi})$, $i = 1, 2, \cdots, m$, respectively.

Sometimes, we may formulate a stochastic decision system as a chance-constrained goal programming (CCGP) according to the priority structure and target levels set by the decision-maker:

$$\begin{cases} \min\limits_{\boldsymbol{x}} \sum\limits_{j=1}^{l} P_j \sum\limits_{i=1}^{m} \left(u_{ij} \left(\min\limits_{d_i^+} d_i^+ \vee 0 \right) + v_{ij} \left(\min\limits_{d_i^-} d_i^- \vee 0 \right) \right) \\ \text{subject to:} \\ \qquad \Pr\left\{ f_i(\boldsymbol{x}, \boldsymbol{\xi}) - b_i \le d_i^+ \right\} \ge \beta_i^+, \; i = 1, 2, \cdots, m \\ \qquad \Pr\left\{ b_i - f_i(\boldsymbol{x}, \boldsymbol{\xi}) \le d_i^- \right\} \ge \beta_i^-, \; i = 1, 2, \cdots, m \\ \qquad \Pr\left\{ g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0 \right\} \ge \alpha_j, \qquad\quad j = 1, 2, \cdots, p \end{cases} \tag{4.31}$$

where P_j is the preemptive priority factor which expresses the relative importance of various goals, $P_j \gg P_{j+1}$, for all j, u_{ij} is the weighting factor corresponding to positive deviation for goal i with priority j assigned, v_{ij} is the weighting factor corresponding to negative deviation for goal i with priority j assigned, $\min d_i^+ \vee 0$ is the β_i^+-optimistic positive deviation from the target of goal i, $\min d_i^- \vee 0$ is the β_i^--optimistic negative deviation from the target of goal i, f_i is a function in goal constraints, g_j is a function in system constraints, b_i is the target value according to goal i, l is the number of priorities, m is the number of goal constraints, and p is the number of system constraints.

Remark 4.1. In a deterministic goal programming, at most one of positive deviation and negative deviation takes a positive value. However, for a CCGP, it is possible that both of them are positive.

Minimax Chance-Constrained Programming

In a stochastic environment, in order to maximize the pessimistic return with a given confidence level subject to some chance constraint, Liu [181] provided the following minimax CCP model:

$$\begin{cases} \max\limits_{\boldsymbol{x}} \min\limits_{\overline{f}} \overline{f} \\ \text{subject to:} \\ \qquad \Pr\left\{f(\boldsymbol{x},\boldsymbol{\xi}) \le \overline{f}\right\} \ge \beta \\ \qquad \Pr\left\{g_j(\boldsymbol{x},\boldsymbol{\xi}) \le 0, j=1,2,\cdots,p\right\} \ge \alpha \end{cases} \tag{4.32}$$

where α and β are the given confidence levels, and $\min \overline{f}$ is the β-pessimistic return.

If there are multiple objectives, then we may employ the following minimax CCMOP,

$$\begin{cases} \max\limits_{\boldsymbol{x}} \left[\min\limits_{\overline{f}_1} \overline{f}_1,\ \min\limits_{\overline{f}_2} \overline{f}_2,\ \cdots,\ \min\limits_{\overline{f}_m} \overline{f}_m\right] \\ \text{subject to:} \\ \qquad \Pr\left\{f_i(\boldsymbol{x},\boldsymbol{\xi}) \le \overline{f}_i\right\} \ge \beta_i,\ i=1,2,\cdots,m \\ \qquad \Pr\left\{g_j(\boldsymbol{x},\boldsymbol{\xi}) \le 0\right\} \ge \alpha_j,\ \ j=1,2,\cdots,p \end{cases} \tag{4.33}$$

where α_j and β_i are confidence levels, and $\min \overline{f}_i$ are the β_i-pessimistic values to the return functions $f_i(\boldsymbol{x},\boldsymbol{\xi})$, $i=1,2,\cdots,m$, $j=1,2,\cdots,p$, respectively.

We can also formulate a stochastic decision system as a minimax CCGP according to the priority structure and target levels set by the decision-maker:

$$\begin{cases} \min\limits_{\boldsymbol{x}} \sum\limits_{j=1}^{l} P_j \sum\limits_{i=1}^{m} \left[u_{ij}\left(\max\limits_{d_i^+} d_i^+ \vee 0\right) + v_{ij}\left(\max\limits_{d_i^-} d_i^- \vee 0\right)\right] \\ \text{subject to:} \\ \qquad \Pr\left\{f_i(\boldsymbol{x},\boldsymbol{\xi}) - b_i \ge d_i^+\right\} \ge \beta_i^+,\ i=1,2,\cdots,m \\ \qquad \Pr\left\{b_i - f_i(\boldsymbol{x},\boldsymbol{\xi}) \ge d_i^-\right\} \ge \beta_i^-,\ i=1,2,\cdots,m \\ \qquad \Pr\left\{g_j(\boldsymbol{x},\boldsymbol{\xi}) \le 0\right\} \ge \alpha_j, \qquad\quad j=1,2,\cdots,p \end{cases} \tag{4.34}$$

where P_j is the preemptive priority factor which expresses the relative importance of various goals, $P_j \gg P_{j+1}$, for all j, u_{ij} is the weighting factor corresponding to positive deviation for goal i with priority j assigned, v_{ij} is the weighting factor corresponding to negative deviation for goal i with priority j assigned, $\max d_i^+ \vee 0$ is the β_i^+-pessimistic positive deviation from the target of goal i, $\max d_i^- \vee 0$ is the β_i^--pessimistic negative deviation from the target of goal i, f_i is a function in goal constraints, g_j is a function in system constraints, b_i is the target value according to goal i, l is the number of priorities, m is the number of goal constraints, and p is the number of system constraints.

Deterministic Equivalents

The traditional solution methods require conversion of the chance constraints to their respective deterministic equivalents. As we know, this process is usually hard to perform and only successful for some special cases. Let us consider the following form of chance constraint,

$$\Pr\{g(\boldsymbol{x}, \boldsymbol{\xi}) \le 0\} \ge \alpha. \tag{4.35}$$

It is clear that

(a)the chance constraints (4.28) are a set of form (4.35);
(b)the stochastic objective constraint $\Pr\{f(\boldsymbol{x}, \boldsymbol{\xi}) \ge \overline{f}\} \ge \beta$ coincides with the form (4.35) by defining $g(\boldsymbol{x}, \boldsymbol{\xi}) = \overline{f} - f(\boldsymbol{x}, \boldsymbol{\xi})$;
(c)the stochastic objective constraint $\Pr\{f(\boldsymbol{x}, \boldsymbol{\xi}) \le \overline{f}\} \ge \beta$ coincides with the form (4.35) by defining $g(\boldsymbol{x}, \boldsymbol{\xi}) = f(\boldsymbol{x}, \boldsymbol{\xi}) - \overline{f}$;
(d)the stochastic goal constraints $\Pr\{b - f(\boldsymbol{x}, \boldsymbol{\xi}) \le d^-\} \ge \beta$ and $\Pr\{f(\boldsymbol{x}, \boldsymbol{\xi}) - b \le d^+\} \ge \beta$ coincide with the form (4.35) by defining $g(\boldsymbol{x}, \boldsymbol{\xi}) = b - f(\boldsymbol{x}, \boldsymbol{\xi}) - d^-$ and $g(\boldsymbol{x}, \boldsymbol{\xi}) = f(\boldsymbol{x}, \boldsymbol{\xi}) - b - d^+$, respectively; and
(e)the stochastic goal constraints $\Pr\{b - f(\boldsymbol{x}, \boldsymbol{\xi}) \ge d^-\} \ge \beta$ and $\Pr\{f(\boldsymbol{x}, \boldsymbol{\xi}) - b \ge d^+\} \ge \beta$ coincide with the form (4.35) by defining $g(\boldsymbol{x}, \boldsymbol{\xi}) = f(\boldsymbol{x}, \boldsymbol{\xi}) + d^- - b$ and $g(\boldsymbol{x}, \boldsymbol{\xi}) = b - f(\boldsymbol{x}, \boldsymbol{\xi}) + d^+$, respectively.

This section summarizes some known results.

Theorem 4.8. *Assume that the stochastic vector $\boldsymbol{\xi}$ degenerates to a random variable ξ with probability distribution Φ, and the function $g(\boldsymbol{x}, \boldsymbol{\xi})$ has the form $g(\boldsymbol{x}, \boldsymbol{\xi}) = h(\boldsymbol{x}) - \xi$. Then $\Pr\{g(\boldsymbol{x}, \boldsymbol{\xi}) \le 0\} \ge \alpha$ if and only if $h(\boldsymbol{x}) \le K_\alpha$, where K_α is the maximal number such that $\Pr\{K_\alpha \le \xi\} \ge \alpha$.*

Proof: The assumption implies that $\Pr\{g(\boldsymbol{x}, \boldsymbol{\xi}) \le 0\} \ge \alpha$ can be written in the following form,

$$\Pr\{h(\boldsymbol{x}) \le \xi\} \ge \alpha. \tag{4.36}$$

For each given confidence level α $(0 < \alpha \le 1)$, let K_α be the maximal number (maybe $+\infty$) such that

$$\Pr\{K_\alpha \le \xi\} \ge \alpha. \tag{4.37}$$

Note that the probability $\Pr\{K_\alpha \le \xi\}$ will increase if K_α is replaced with a smaller number. Hence $\Pr\{h(\boldsymbol{x}) \le \xi\} \ge \alpha$ if and only if $h(\boldsymbol{x}) \le K_\alpha$.

Remark 4.2. For a continuous random variable ξ, the equation $\Pr\{K_\alpha \le \xi\} = 1 - \Phi(K_\alpha)$ always holds, and we have, by (4.37),

$$K_\alpha = \Phi^{-1}(1 - \alpha) \tag{4.38}$$

where Φ^{-1} is the inverse function of Φ.

Example 4.10. Assume that we have the following chance constraint,

$$\begin{cases} \Pr\{3x_1 + 4x_2 \le \xi_1\} \ge 0.8 \\ \Pr\{x_1^2 - x_2^3 \le \xi_2\} \ge 0.9 \end{cases} \tag{4.39}$$

where ξ_1 is an exponentially distributed random variable $\mathcal{EXP}(2)$ whose probability distribution is denoted by Φ_1, and ξ_2 is a normally distributed random variable $\mathcal{N}(2,1)$ whose probability distribution is denoted by Φ_2. It follows from Theorem 4.8 that the chance constraint (4.39) is equivalent to

$$\begin{cases} 3x_1 + 4x_2 \le \Phi_1^{-1}(1-0.8) = 0.446 \\ x_1^2 - x_2^3 \le \Phi_2^{-1}(1-0.9) = 0.719. \end{cases}$$

Theorem 4.9. *Assume that the stochastic vector* $\boldsymbol{\xi} = (a_1, a_2, \cdots, a_n, b)$ *and the function* $g(\boldsymbol{x}, \boldsymbol{\xi})$ *has the form* $g(\boldsymbol{x}, \boldsymbol{\xi}) = a_1x_1 + a_2x_2 + \cdots + a_nx_n - b$*. If* a_i *and* b *are assumed to be independently normally distributed random variables, then* $\Pr\{g(\boldsymbol{x}, \boldsymbol{\xi}) \le 0\} \ge \alpha$ *if and only if*

$$\sum_{i=1}^{n} E[a_i]x_i + \Phi^{-1}(\alpha)\sqrt{\sum_{i=1}^{n} V[a_i]x_i^2 + V[b]} \le E[b] \tag{4.40}$$

where Φ *is the standardized normal distribution function.*

Proof: The chance constraint $\Pr\{g(\boldsymbol{x}, \boldsymbol{\xi}) \le 0\} \ge \alpha$ can be written in the following form,

$$\Pr\left\{\sum_{i=1}^{n} a_ix_i \le b\right\} \ge \alpha. \tag{4.41}$$

Since a_i and b are assumed to be independently normally distributed random variables, the quantity

$$y = \sum_{i=1}^{n} a_ix_i - b$$

is also normally distributed with the following expected value and variance,

$$E[y] = \sum_{i=1}^{n} E[a_i]x_i - E[b],$$
$$V[y] = \sum_{i=1}^{n} V[a_i]x_i^2 + V[b].$$

We note that

$$\frac{\sum_{i=1}^{n} a_ix_i - b - \left(\sum_{i=1}^{n} E[a_i]x_i - E[b]\right)}{\sqrt{\sum_{i=1}^{n} V[a_i]x_i^2 + V[b]}}$$

must be standardized normally distributed. Since the inequality $\sum_{i=1}^{n} a_i x_i \le b$ is equivalent to

$$\frac{\sum_{i=1}^{n} a_i x_i - b - \left(\sum_{i=1}^{n} E[a_i]x_i - E[b]\right)}{\sqrt{\sum_{i=1}^{n} V[a_i]x_i^2 + V[b]}} \le -\frac{\sum_{i=1}^{n} E[a_i]x_i - E[b]}{\sqrt{\sum_{i=1}^{n} V[a_i]x_i^2 + V[b]}},$$

the chance constraint (4.41) is equivalent to

$$\Pr\left\{\eta \le -\frac{\sum_{i=1}^{n} E[a_i]x_i - E[b]}{\sqrt{\sum_{i=1}^{n} V[a_i]x_i^2 + V[b]}}\right\} \ge \alpha \tag{4.42}$$

where η is the standardized normally distributed random variable. Then the chance constraint (4.42) holds if and only if

$$\Phi^{-1}(\alpha) \le -\frac{\sum_{i=1}^{n} E[a_i]x_i - E[b]}{\sqrt{\sum_{i=1}^{n} V[a_i]x_i^2 + V[b]}}. \tag{4.43}$$

That is, the deterministic equivalent of chance constraint is (4.40). The theorem is proved.

Example 4.11. Suppose that the chance constraint set has the following form,

$$\Pr\{a_1x_1 + a_2x_2 + a_2x_3 \le b\} \ge 0.95 \tag{4.44}$$

where a_1, a_2, a_3, and b are normally distributed random variables $\mathcal{N}(1,1)$, $\mathcal{N}(2,1)$, $\mathcal{N}(3,1)$, and $\mathcal{N}(4,1)$, respectively. Then the formula (4.40) yields the deterministic equivalent of (4.44) as follows,

$$x_1 + 2x_2 + 3x_3 + 1.645\sqrt{x_1^2 + x_2^2 + x_3^2 + 1} \le 4$$

by the fact that $\Phi^{-1}(0.95) = 1.645$.

4.4 Dependent-Chance Programming

In practice, there usually exist multiple events in a complex stochastic decision system. Sometimes, the decision-maker wishes to maximize the probabilities of meeting these events. In order to model this type of stochastic decision system, Liu [166] provided the third type of stochastic programming, called *dependent-chance programming* (DCP), in which the underlying philosophy is based on selecting the decision with maximal chance to meet the event.

DCP theory breaks the concept of feasible set and replaces it with uncertain environment. Roughly speaking, DCP involves maximizing chance functions of events in an uncertain environment. In deterministic model, EVM and CCP, the feasible set is essentially assumed to be deterministic after the real problem is modeled. That is, an optimal solution is given regardless of whether it can be performed in practice. However, the given solution may be impossible to perform if the realization of uncertain parameter is unfavorable. Thus DCP theory never assumes that the feasible set is deterministic. In fact, DCP is constructed in an uncertain environment. This special feature of DCP is very different from the other existing types of stochastic programming. However, such problems do exist in the real world.

Now we introduce the concepts of uncertain environment, event and chance function, and discuss the principle of uncertainty, thus offering a spectrum of DCP models. We will take a supply system, represented by Figure 4.1 as the background.

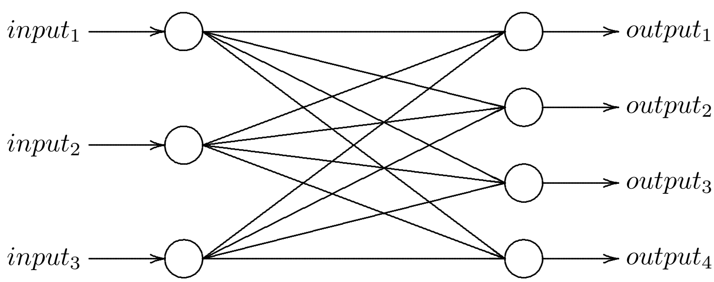

Fig. 4.1 A Supply System

As an illustrative example, in Figure 4.1 there are 3 inputs representing 3 locations of resources and 4 outputs representing the demands of 4 users. We must answer the following supply problem: What is the appropriate combination of resources such that certain goals of supply are achieved?

In order to obtain the appropriate combination of resources for the supply problem, we use 12 decision variables $x_1, x_2, \cdots, x_{12}$ to represent an action, where x_1, x_2, x_3, x_4 are quantities ordered from $input_1$ to outputs 1,2,3,4 respectively; x_5, x_6, x_7, x_8 from $input_2$; $x_9, x_{10}, x_{11}, x_{12}$ from $input_3$. In practice, some variables may vanish due to some physical constraints.

We note that the inputs are available outside resources. Thus they have their own properties. For example, the capacities of resources are finite. Let ξ_1, ξ_2, ξ_3 be the maximum quantities supplied by the three resources. Then we have the following constraint,

$$\begin{cases} x_1^+ + x_2^+ + x_3^+ + x_4^+ \le \xi_1 \\ x_5^+ + x_6^+ + x_7^+ + x_8^+ \le \xi_2 \\ x_9^+ + x_{10}^+ + x_{11}^+ + x_{12}^+ \le \xi_3 \\ x_i \ge 0, \quad i = 1, 2, \cdots, 12 \end{cases} \tag{4.45}$$

which represents that the quantities ordered from the resources are nonnegative and cannot exceed the maximum quantities, where x_i^+ represents x_i if x_i takes positive value, and vanishes otherwise. This means that the decision variable $x_i = 0$ must be able to perform for any realization of stochastic resources.

If at least one of ξ_1, ξ_2, and ξ_3 is really stochastic, then the constraint (4.45) is uncertain because we cannot make a decision such that it can be performed certainly before knowing the realization of ξ_1, ξ_2, and ξ_3. We will call this type of constraint the uncertain environment, and in this case the stochastic environment.

Definition 4.14. *By uncertain environment we mean the following stochastic constraint,*

$$g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, \quad j = 1, 2, \cdots, p, \tag{4.46}$$

where $\boldsymbol{x}$ is a decision vector, and $\boldsymbol{\xi}$ is a stochastic vector.

In the supply system, we should satisfy the demands of the 4 users, marked by c_1, c_2, c_3, and c_4. Then we have the following four events:

$$x_1 + x_5 + x_9 = c_1, \quad x_2 + x_6 + x_{10} = c_2,$$
$$x_3 + x_7 + x_{11} = c_3, \; x_4 + x_8 + x_{12} = c_4.$$

These equalities mean that the decision should satisfy the demands of users. Generally, an event is defined as follows.

Definition 4.15. *By event we mean a system of stochastic inequalities,*

$$h_k(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, \quad k = 1, 2, \cdots, q \tag{4.47}$$

where $\boldsymbol{x}$ is a decision vector, and $\boldsymbol{\xi}$ is a stochastic vector.

In view of the uncertainty of this system, we are not sure whether a decision can be performed before knowing the realization of stochastic variables. Thus we wish to employ the following chance functions to evaluate these four events,

$$f_1(\boldsymbol{x}) = \Pr\{x_1 + x_5 + x_9 = c_1\}, \quad f_2(\boldsymbol{x}) = \Pr\{x_2 + x_6 + x_{10} = c_2\},$$
$$f_3(\boldsymbol{x}) = \Pr\{x_3 + x_7 + x_{11} = c_3\}, \; f_4(\boldsymbol{x}) = \Pr\{x_4 + x_8 + x_{12} = c_4\},$$

subject to the uncertain environment (4.45).

Definition 4.16. *The chance function of an event $\mathcal{E}$ characterized by (4.47) is defined as the probability measure of the event, i.e.,*

$$f(\boldsymbol{x}) = \Pr\{h_k(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, k = 1, 2, \cdots, q\} \tag{4.48}$$

subject to the uncertain environment (4.46).

Usually, we hope to maximize the four chance functions $f_1(\boldsymbol{x})$, $f_2(\boldsymbol{x})$, $f_3(\boldsymbol{x})$ and $f_4(\boldsymbol{x})$. Here we remind the reader once more that the events like $x_1 + x_5 + x_9 = c_1$ do possess uncertainty because they are in an uncertain environment. *Any event is uncertain if it is in an uncertain environment*! This is an important law in the uncertain world. In fact, the randomness of the event is caused by the stochastic parameters ξ_1, ξ_2, ξ_3, and ξ_4 in the uncertain environment.

Until now we have formulated a stochastic programming model for the supply problem in an uncertain environment as follows,

$$\begin{cases} \max f_1(\boldsymbol{x}) = \Pr\{x_1 + x_5 + x_9 = c_1\} \\ \max f_2(\boldsymbol{x}) = \Pr\{x_2 + x_6 + x_{10} = c_2\} \\ \max f_3(\boldsymbol{x}) = \Pr\{x_3 + x_7 + x_{11} = c_3\} \\ \max f_4(\boldsymbol{x}) = \Pr\{x_4 + x_8 + x_{12} = c_4\} \\ \text{subject to:} \\ \qquad x_1^+ + x_2^+ + x_3^+ + x_4^+ \le \xi_1 \\ \qquad x_5^+ + x_6^+ + x_7^+ + x_8^+ \le \xi_2 \\ \qquad x_9^+ + x_{10}^+ + x_{11}^+ + x_{12}^+ \le \xi_3 \\ \qquad x_i \ge 0, \quad i = 1, 2, \cdots, 12 \end{cases} \tag{4.49}$$

where ξ_1, ξ_2, and ξ_3 are stochastic variables. In this stochastic programming model, some variables (for example, x_1, x_2, x_3, x_4) are stochastically dependent because they share a common uncertain resource ξ_1. This also implies that the chance functions are stochastically dependent. We will call the stochastic programming (4.49) *dependent-chance programming* (DCP).

Principle of Uncertainty

How do we compute the chance function of an event $\mathcal{E}$ in an uncertain environment? In order to answer this question, we first give some definitions.

Definition 4.17. *Let $r(x_1, x_2, \cdots, x_n)$ be an n-dimensional function. The ith decision variable x_i is said to be degenerate if*

$$r(x_1, \cdots, x_{i-1}, x_i', x_{i+1}, \cdots, x_n) = r(x_1, \cdots, x_{i-1}, x_i'', x_{i+1}, \cdots, x_n)$$

for any x_i' and x_i''; otherwise it is nondegenerate.

For example, $r(x_1, x_2, x_3, x_4, x_5) = (x_1 + x_3)/x_4$ is a 5-dimensional function. The variables x_1, x_3, x_4 are nondegenerate, but x_2 and x_5 are degenerate.

Definition 4.18. *Let $\mathcal{E}$ be an event $h_k(\boldsymbol{x}, \boldsymbol{\xi}) \le 0$, $k = 1, 2, \cdots, q$. The support of the event $\mathcal{E}$, denoted by $\mathcal{E}^*$, is defined as the set consisting of all nondegenerate decision variables of functions $h_k(\boldsymbol{x}, \boldsymbol{\xi})$, $k = 1, 2, \cdots, q$.*

For example, let $\boldsymbol{x} = (x_1, x_2, \cdots, x_{12})$ be a decision vector, and let $\mathcal{E}$ be an event characterized by $x_1 + x_5 + x_9 = c_1$ and $x_2 + x_6 + x_{10} = c_2$. It is clear that x_1, x_5, x_9 are nondegenerate variables of the function $x_1 + x_5 + x_9$, and x_2, x_6, x_{10} are nondegenerate variables of the function $x_2 + x_6 + x_{10}$. Thus the support $\mathcal{E}^*$ of the event $\mathcal{E}$ is $\{x_1, x_2, x_5, x_6, x_9, x_{10}\}$.

Definition 4.19. *The jth constraint $g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0$ is called an active constraint of the event $\mathcal{E}$ if the set of nondegenerate decision variables of $g_j(\boldsymbol{x}, \boldsymbol{\xi})$ and the support $\mathcal{E}^*$ have nonempty intersection; otherwise it is inactive.*

Definition 4.20. *Let $\mathcal{E}$ be an event $h_k(\boldsymbol{x}, \boldsymbol{\xi}) \le 0$, $k = 1, 2, \cdots, q$ in the uncertain environment $g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, j = 1, 2, \cdots, p$. The dependent support of the event $\mathcal{E}$, denoted by $\mathcal{E}^{**}$, is defined as the set consisting of all nondegenerate decision variables of $h_k(\boldsymbol{x}, \boldsymbol{\xi}), k = 1, 2, \cdots, q$ and $g_j(\boldsymbol{x}, \boldsymbol{\xi})$ in the active constraints to the event $\mathcal{E}$.*

Remark 4.3. It is obvious that $\mathcal{E}^* \subset \mathcal{E}^{**}$ holds.

Definition 4.21. *The jth constraint $g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0$ is called a dependent constraint of the event $\mathcal{E}$ if the set of nondegenerate decision variables of $g_j(\boldsymbol{x}, \boldsymbol{\xi})$ and the dependent support $\mathcal{E}^{**}$ have nonempty intersection; otherwise it is independent.*

Remark 4.4. An active constraint must be a dependent constraint.

Definition 4.22. *Let $\mathcal{E}$ be an event $h_k(\boldsymbol{x}, \boldsymbol{\xi}) \le 0$, $k = 1, 2, \cdots, q$ in the uncertain environment $g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, j = 1, 2, \cdots, p$. For each decision $\boldsymbol{x}$ and realization $\boldsymbol{\xi}$, the event $\mathcal{E}$ is said to be consistent in the uncertain environment if the following two conditions hold: (i) $h_k(\boldsymbol{x}, \boldsymbol{\xi}) \le 0$, $k = 1, 2, \cdots, q$; and (ii) $g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0$, $j \in J$, where J is the index set of all dependent constraints.*

Intuitively, an event can be met by a decision provided that the decision meets both the event itself and the dependent constraints. We conclude it with the following principle of uncertainty.

Principle of Uncertainty: *The chance of a random event is the probability that the event is consistent in the uncertain environment.*

Assume that there are m events $\mathcal{E}_i$ characterized by $h_{ik}(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, k = 1, 2, \cdots, q_i$ for $i = 1, 2, \cdots, m$ in the uncertain environment $g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, j = 1, 2, \cdots, p$. The principle of uncertainty implies that the chance function of the ith event $\mathcal{E}_i$ in the uncertain environment is

$$f_i(\boldsymbol{x}) = \Pr\left\{\begin{array}{l} h_{ik}(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, k = 1, 2, \cdots, q_i \\ g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, j \in J_i \end{array}\right\} \tag{4.50}$$

where J_i are defined by

$$J_i = \left\{ j \in \{1, 2, \cdots, p\} \;\middle|\; g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0 \text{ is a dependent constraint of } \mathcal{E}_i \right\}$$

for $i = 1, 2, \cdots, m$.

Remark 4.5. The principle of uncertainty is the basis of solution procedure of DCP that we shall encounter throughout the remainder of the book. However, the principle of uncertainty does not apply in all cases. For example, consider an event $x_1 \ge 6$ in the uncertain environment $x_1 - x_2 \le \xi_1, x_2 - x_3 \le \xi_2, x_3 \le \xi_3$. It follows from the principle of uncertainty that the chance of the event is $\Pr\{x_1 \ge 6, x_1 - x_2 \le \xi_1, x_2 - x_3 \le \xi_2\}$, which is clearly wrong because the realization of $x_3 \le \xi_3$ must be considered. Fortunately, such a case does not exist in real-life problems.

General Models

In this subsection, we consider the single-objective DCP. A typical DCP is represented as maximizing the chance function of an event subject to an uncertain environment,

$$\begin{cases} \max \Pr\left\{h_k(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, k = 1, 2, \cdots, q\right\} \\ \text{subject to:} \\ \qquad g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, \quad j = 1, 2, \cdots, p \end{cases} \tag{4.51}$$

where $\boldsymbol{x}$ is an n-dimensional decision vector, $\boldsymbol{\xi}$ is a random vector of parameters, the system $h_k(\boldsymbol{x}, \boldsymbol{\xi}) \le 0$, $k = 1, 2, \cdots, q$ represents an event $\mathcal{E}$, and the constraints $g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0$, $j = 1, 2, \cdots, p$ are an uncertain environment.

DCP (4.51) reads as "maximizing the probability of the random event $h_k(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, k = 1, 2, \cdots, q$ subject to the uncertain environment $g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, j = 1, 2, \cdots, p$".

We now go back to the supply system. Assume that there is only one event $\mathcal{E}$ that satisfies the demand c_1 of *output*$_1$ (i.e., $x_1 + x_5 + x_9 = c_1$). If we want to find a decision $\boldsymbol{x}$ with maximum probability to meet the event $\mathcal{E}$, then we have the following DCP model,

$$\begin{cases} \max \Pr\{x_1 + x_5 + x_9 = c_1\} \\ \text{subject to:} \\ \qquad x_1^+ + x_2^+ + x_3^+ + x_4^+ \le \xi_1 \\ \qquad x_5^+ + x_6^+ + x_7^+ + x_8^+ \le \xi_2 \\ \qquad x_9^+ + x_{10}^+ + x_{11}^+ + x_{12}^+ \le \xi_3 \\ \qquad x_i \ge 0, i = 1, 2, \cdots, 12. \end{cases} \tag{4.52}$$

It is clear that the support of the event $\mathcal{E}$ is $\mathcal{E}^* = \{x_1, x_5, x_9\}$. If $x_1 \neq 0, x_5 \neq 0, x_9 \neq 0$, then the uncertain environment is

$$\begin{cases} x_1 + x_2 + x_3 + x_4 \le \xi_1 \\ x_5 + x_6 + x_7 + x_8 \le \xi_2 \\ x_9 + x_{10} + x_{11} + x_{12} \le \xi_3 \\ x_i \ge 0, i = 1, 2, \cdots, 12. \end{cases}$$

Thus the dependent support $\mathcal{E}^{**} = \{x_1, x_2, \cdots, x_{12}\}$, and all constraints are dependent constraints. It follows from the principle of uncertainty that the chance function of the event $\mathcal{E}$ is

$$f(\boldsymbol{x}) = \Pr\left\{\begin{array}{l} x_1 + x_5 + x_9 = c_1 \\ x_1 + x_2 + x_3 + x_4 \le \xi_1 \\ x_5 + x_6 + x_7 + x_8 \le \xi_2 \\ x_9 + x_{10} + x_{11} + x_{12} \le \xi_3 \\ x_i \ge 0, i = 1, 2, \cdots, 12 \end{array}\right\}.$$

If $x_1 = 0, x_5 \neq 0, x_9 \neq 0$, then the uncertain environment is

$$\begin{cases} 0 + x_2 + x_3 + x_4 \le \xi_1 \\ x_5 + x_6 + x_7 + x_8 \le \xi_2 \\ x_9 + x_{10} + x_{11} + x_{12} \le \xi_3 \\ x_i \ge 0, i = 1, 2, \cdots, 12. \end{cases}$$

Thus the dependent support $\mathcal{E}^{**} = \{x_5, x_6, \cdots, x_{12}\}$. It follows from the principle of uncertainty that the chance function of the event $\mathcal{E}$ is

$$f(\boldsymbol{x}) = \Pr\left\{\begin{array}{l} x_1 + x_5 + x_9 = c_1 \\ x_5 + x_6 + x_7 + x_8 \le \xi_2 \\ x_9 + x_{10} + x_{11} + x_{12} \le \xi_3 \\ x_i \ge 0, i = 5, 6, \cdots, 12 \end{array}\right\}.$$

Similarly, if $x_1 \neq 0, x_5 = 0, x_9 \neq 0$, then the chance function of the event $\mathcal{E}$ is

$$f(\boldsymbol{x}) = \Pr\left\{\begin{array}{l} x_1 + x_5 + x_9 = c_1 \\ x_1 + x_2 + x_3 + x_4 \le \xi_1 \\ x_9 + x_{10} + x_{11} + x_{12} \le \xi_3 \\ x_i \ge 0, i = 1, 2, 3, 4, 9, 10, 11, 12 \end{array}\right\}.$$

If $x_1 \neq 0, x_5 \neq 0, x_9 = 0$, then the chance function of the event $\mathcal{E}$ is

$$f(\boldsymbol{x}) = \Pr\left\{\begin{array}{l} x_1 + x_5 + x_9 = c_1 \\ x_1 + x_2 + x_3 + x_4 \le \xi_1 \\ x_5 + x_6 + x_7 + x_8 \le \xi_2 \\ x_i \ge 0, i = 1, 2, \cdots, 8 \end{array}\right\}.$$

If $x_1 = 0, x_5 = 0, x_9 \neq 0$, then the chance function of the event $\mathcal{E}$ is

$$f(\boldsymbol{x}) = \Pr\left\{\begin{array}{l} x_1 + x_5 + x_9 = c_1 \\ x_9 + x_{10} + x_{11} + x_{12} \le \xi_3 \\ x_i \ge 0, i = 9, 10, \cdots, 12 \end{array}\right\}.$$

If $x_1 = 0, x_5 \neq 0, x_9 = 0$, then the chance function of the event $\mathcal{E}$ is

$$f(\boldsymbol{x}) = \Pr\left\{\begin{array}{l} x_1 + x_5 + x_9 = c_1 \\ x_5 + x_6 + x_7 + x_8 \le \xi_2 \\ x_i \ge 0, i = 5, 6, \cdots, 8 \end{array}\right\}.$$

If $x_1 \neq 0, x_5 = 0, x_9 = 0$, then the chance function of the event $\mathcal{E}$ is

$$f(\boldsymbol{x}) = \Pr\left\{\begin{array}{l} x_1 + x_5 + x_9 = c_1 \\ x_1 + x_2 + x_3 + x_4 \le \xi_1 \\ x_i \ge 0, i = 1, 2, \cdots, 4 \end{array}\right\}.$$

Note that the case $x_1 = x_5 = x_9 = 0$ is impossible because $c_1 \neq 0$. It follows that DCP (4.52) is equivalent to the unconstrained model "$\max f(\boldsymbol{x})$".

Dependent-Chance Multiobjective Programming

Since a complex decision system usually undertakes multiple events, there undoubtedly exist multiple potential objectives (some of them are chance functions) in a decision process. A typical formulation of dependent-chance multiobjective programming (DCMOP) is represented as maximizing multiple chance functions subject to an uncertain environment,

$$\left\{\begin{array}{l} \max \left[\begin{array}{l} \Pr\{h_{1k}(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, k = 1, 2, \cdots, q_1\} \\ \Pr\{h_{2k}(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, k = 1, 2, \cdots, q_2\} \\ \quad \cdots \\ \Pr\{h_{mk}(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, k = 1, 2, \cdots, q_m\} \end{array}\right] \\ \text{subject to:} \\ \qquad g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, \quad j = 1, 2, \cdots, p \end{array}\right. \tag{4.53}$$

where $h_{ik}(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, k = 1, 2, \cdots, q_i$ represent events $\mathcal{E}_i$ for $i = 1, 2, \cdots, m$, respectively.

It follows from the principle of uncertainty that we can construct a relationship between decision vectors and chance functions, thus calculating the chance functions by stochastic simulations or traditional methods. Then we can solve DCMOP by utility theory if complete information of the preference function is given by the decision-maker or search for all of the efficient solutions if no information is available. In practice, the decision-maker can provide only partial information. In this case, we have to employ the interactive methods.

Dependent-Chance Goal Programming

When some management targets are given, the objective function may minimize the deviations, positive, negative, or both, with a certain priority structure set by the decision-maker. Then we may formulate the stochastic decision system as the following dependent-chance goal programming (DCGP),

$$\begin{cases} \min \sum_{j=1}^{l} P_j \sum_{i=1}^{m} (u_{ij} d_i^+ \vee 0 + v_{ij} d_i^- \vee 0) \\ \text{subject to:} \\ \quad \Pr\{h_{ik}(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, k = 1, 2, \cdots, q_i\} - b_i = d_i^+, \; i = 1, 2, \cdots, m \\ \quad b_i - \Pr\{h_{ik}(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, k = 1, 2, \cdots, q_i\} = d_i^-, \; i = 1, 2, \cdots, m \\ \quad g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, \qquad j = 1, 2, \cdots, p \end{cases}$$

where P_j is the preemptive priority factor, u_{ij} is the weighting factor corresponding to positive deviation for goal i with priority j assigned, v_{ij} is the weighting factor corresponding to negative deviation for goal i with priority j assigned, $d_i^+ \vee 0$ is the positive deviation from the target of goal i, $d_i^- \vee 0$ is the negative deviation from the target of goal i, b_i is the target value according to goal i, l is the number of priorities, and m is the number of goal constraints.

4.5 Hybrid Intelligent Algorithm

From the mathematical viewpoint, there is no difference between deterministic mathematical programming and stochastic programming except for the fact that there exist uncertain functions in the latter. If the uncertain functions can be converted to their deterministic forms, then we can obtain equivalent deterministic models. However, generally speaking, we cannot do so. It is thus more convenient to deal with them by stochastic simulations. Essentially, there are three types of uncertain functions in stochastic programming as follows:

$$\begin{aligned} &U_1 : \boldsymbol{x} \to E[f(\boldsymbol{x}, \boldsymbol{\xi})], \\ &U_2 : \boldsymbol{x} \to \Pr\{g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, j = 1, 2, \cdots, p\}, \\ &U_3 : \boldsymbol{x} \to \max\left\{\overline{f} \;\middle|\; \Pr\left\{f(\boldsymbol{x}, \boldsymbol{\xi}) \ge \overline{f}\right\} \ge \alpha\right\}. \end{aligned} \tag{4.54}$$

Stochastic Simulation for $U_1(\boldsymbol{x})$

In order to compute the uncertain function $U_1(\boldsymbol{x})$, we generate ω_k from the probability space $(\Omega, \mathcal{A}, \Pr)$ and produce $\boldsymbol{\xi}_k = \boldsymbol{\xi}(\omega_k)$ for $k = 1, 2, \cdots, N$. *Equivalently, we generate random vectors $\boldsymbol{\xi}_k$ according to the probability distribution Φ for $k = 1, 2, \cdots, N$.* It follows from the strong law of large numbers that

$$\frac{\sum_{k=1}^{N} f(\boldsymbol{x},\boldsymbol{\xi}_k)}{N} \longrightarrow U_1(\boldsymbol{x}), \quad \text{a.s.} \tag{4.55}$$

as $N \to \infty$. Therefore, the value $U_1(\boldsymbol{x})$ is estimated by

$$\frac{1}{N}\sum_{k=1}^{N} f(\boldsymbol{x},\boldsymbol{\xi}_k)$$

provided that N is sufficiently large.

Algorithm 4.4 (Stochastic Simulation for $U_1(\boldsymbol{x})$)
Step 1. Set $e = 0$.
Step 2. Generate ω from the probability space $(\Omega, \mathcal{A}, \Pr)$ and produce $\boldsymbol{\xi} = \boldsymbol{\xi}(\omega)$. Equivalently, generate a random vector $\boldsymbol{\xi}$ according to its probability distribution.
Step 3. $e \leftarrow e + f(\boldsymbol{x},\boldsymbol{\xi})$.
Step 4. Repeat the second and third steps N times.
Step 5. $U_1(\boldsymbol{x}) = e/N$.

Stochastic Simulation for $U_2(\boldsymbol{x})$

In order to compute the uncertain function $U_2(\boldsymbol{x})$, we generate ω_k from the probability space $(\Omega, \mathcal{A}, \Pr)$ and produce $\boldsymbol{\xi}_k = \boldsymbol{\xi}(\omega_k)$ for $k = 1, 2, \cdots, N$. *Equivalently, we generate random vectors $\boldsymbol{\xi}_k$ according to the probability distribution Φ for* $k = 1, 2, \cdots, N$. Let N' denote the number of occasions on which $g_j(\boldsymbol{x},\boldsymbol{\xi}_k) \le 0, j = 1, 2, \cdots, p$ for $k = 1, 2, \cdots, N$ (i.e., the number of random vectors satisfying the system of inequalities). Let us define

$$h(\boldsymbol{x},\boldsymbol{\xi}_k) = \begin{cases} 1, \text{ if } g_j(\boldsymbol{x},\boldsymbol{\xi}_k) \le 0, j = 1, 2, \cdots, p \\ 0, \text{ otherwise.} \end{cases}$$

Then we have $E[h(\boldsymbol{x},\boldsymbol{\xi}_k)] = U_2(\boldsymbol{x})$ for all k, and $N' = \sum_{k=1}^{N} h(\boldsymbol{x},\boldsymbol{\xi}_k)$. It follows from the strong law of large numbers that

$$\frac{N'}{N} = \frac{\sum_{k=1}^{N} h(\boldsymbol{x},\boldsymbol{\xi}_k)}{N}$$

converges a.s. to $U_2(\boldsymbol{x})$. Thus $U_2(\boldsymbol{x})$ can be estimated by N'/N provided that N is sufficiently large.

Algorithm 4.5 (Stochastic Simulation for $U_2(\boldsymbol{x})$)

Step 1. Set $N' = 0$.

Step 2. Generate ω from the probability space $(\Omega, \mathcal{A}, \Pr)$ and produce $\boldsymbol{\xi} = \boldsymbol{\xi}(\omega)$. Equivalently, generate a random vector $\boldsymbol{\xi}$ according to its probability distribution.

Step 3. If $g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0$ for $j = 1, 2, \cdots, p$, then $N' \leftarrow N' + 1$.

Step 4. Repeat the second and third steps N times.

Step 5. $U_2(\boldsymbol{x}) = N'/N$.

Stochastic Simulation for $U_3(\boldsymbol{x})$

In order to compute the uncertain function $U_3(\boldsymbol{x})$, we generate ω_k from the probability space $(\Omega, \mathcal{A}, \Pr)$ and produce $\boldsymbol{\xi}_k = \boldsymbol{\xi}(\omega_k)$ for $k = 1, 2, \cdots, N$. *Equivalently, we generate random vectors $\boldsymbol{\xi}_k$ according to the probability distribution Φ for $k = 1, 2, \cdots, N$.* Now we define

$$h(\boldsymbol{x}, \boldsymbol{\xi}_k) = \begin{cases} 1, \text{ if } f(\boldsymbol{x}, \boldsymbol{\xi}_k) \ge \overline{f} \\ 0, \text{ otherwise} \end{cases}$$

for $k = 1, 2, \cdots, N$, which are random variables, and $E[h(\boldsymbol{x}, \boldsymbol{\xi}_k)] = \alpha$ for all k. By the strong law of large numbers, we obtain

$$\frac{\sum\limits_{k=1}^{N} h(\boldsymbol{x}, \boldsymbol{\xi}_k)}{N} \longrightarrow \alpha, \quad \text{a.s.}$$

as $N \to \infty$. Note that the sum $\sum_{k=1}^{N} h(\boldsymbol{x}, \boldsymbol{\xi}_k)$ is just the number of $\boldsymbol{\xi}_k$ satisfying $f(\boldsymbol{x}, \boldsymbol{\xi}_k) \ge \overline{f}$ for $k = 1, 2, \cdots, N$. Thus $\overline{f}$ is just the N'th largest element in the sequence $\{f(\boldsymbol{x}, \boldsymbol{\xi}_1), f(\boldsymbol{x}, \boldsymbol{\xi}_2), \cdots, f(\boldsymbol{x}, \boldsymbol{\xi}_N)\}$, where N' is the integer part of αN.

Algorithm 4.6 (Stochastic Simulation for $U_3(\boldsymbol{x})$)

Step 1. Generate ω_k from the probability space $(\Omega, \mathcal{A}, \Pr)$ and produce $\boldsymbol{\xi}_k = \boldsymbol{\xi}(\omega_k)$ for $k = 1, 2, \cdots, N$. Equivalently, generate random vectors $\boldsymbol{\xi}_k$ according to the probability distribution for $k = 1, 2, \cdots, N$.

Step 2. Set $f_i = f(\boldsymbol{x}, \boldsymbol{\xi}_k)$ for $k = 1, 2, \cdots, N$.

Step 3. Set N' as the integer part of βN.

Step 4. Return the N'th largest element in $\{f_1, f_2, \cdots, f_N\}$ as $U_3(\boldsymbol{x})$.

Neural Network for Approximating Uncertain Functions

Although stochastic simulations are able to compute the uncertain functions, we need relatively simple functions to approximate the uncertain functions because the stochastic simulations are a time-consuming process. In order to speed up the solution process, neural network (NN) is employed to approximate uncertain functions due to the following reasons: (i) NN has the ability to approximate the uncertain functions by using the training data; (ii) NN can compensate for the error of training data (all input-output data obtained by stochastic simulation are clearly not precise); and (iii) NN has the high speed of operation after they are trained.

Hybrid Intelligent Algorithm

Liu [181] integrated stochastic simulation, NN and GA to produce a hybrid intelligent algorithm for solving stochastic programming models.

Algorithm 4.7 (Hybrid Intelligent Algorithm)

Step 1. Generate training input-output data for uncertain functions like

$$\begin{aligned} &U_1: \boldsymbol{x} \to E[f(\boldsymbol{x}, \boldsymbol{\xi})], \\ &U_2: \boldsymbol{x} \to \Pr\{g_j(\boldsymbol{x}, \boldsymbol{\xi}) \le 0, j = 1, 2, \cdots, p\}, \\ &U_3: \boldsymbol{x} \to \max\left\{\overline{f} \;\middle|\; \Pr\left\{f(\boldsymbol{x}, \boldsymbol{\xi}) \ge \overline{f}\right\} \ge \alpha\right\} \end{aligned}$$

by the stochastic simulation.

Step 2. Train a neural network to approximate the uncertain functions according to the generated training input-output data.

Step 3. Initialize *pop_size* chromosomes whose feasibility may be checked by the trained neural network.

Step 4. Update the chromosomes by crossover and mutation operations in which the feasibility of offspring may be checked by the trained neural network.

Step 5. Calculate the objective values for all chromosomes by the trained neural network.

Step 6. Compute the fitness of each chromosome according to the objective values.

Step 7. Select the chromosomes by spinning the roulette wheel.

Step 8. Repeat the fourth to seventh steps for a given number of cycles.

Step 9. Report the best chromosome as the optimal solution.

4.6 Numerical Experiments

In order to illustrate its effectiveness, a set of numerical examples has been done, and the results are successful. Here we give some numerical examples which are all performed on a personal computer with the following parameters: the population size is 30, the probability of crossover P_c is 0.3, the probability of mutation P_m is 0.2, and the parameter a in the rank-based evaluation function is 0.05.

Example 4.12. Now we consider the following EVM,

$$\begin{cases} \min E\left[\sqrt{(x_1-\xi_1)^2+(x_2-\xi_2)^2+(x_3-\xi_3)^2}\right] \\ \text{subject to:} \\ \qquad x_1^2+x_2^2+x_3^2 \le 10 \end{cases}$$

where ξ_1 is a uniformly distributed random variable $\mathcal{U}(1,2)$, ξ_2 is a normally distributed random variable $\mathcal{N}(3,1)$, and ξ_3 is an exponentially distributed random variable $\mathcal{EXP}(4)$.

In order to solve this model, we generate input-output data for the uncertain function

$$U:(x_1,x_2,x_3)\to E\left[\sqrt{(x_1-\xi_1)^2+(x_2-\xi_2)^2+(x_3-\xi_3)^2}\right]$$

by stochastic simulation. Then we train an NN (3 input neurons, 5 hidden neurons, 1 output neuron) to approximate the uncertain function U. After that, the trained NN is embedded into a GA to produce a hybrid intelligent algorithm.

A run of the hybrid intelligent algorithm (3000 cycles in simulation, 2000 data in NN, 300 generations in GA) shows that the optimal solution is

$$(x_1^*,x_2^*,x_3^*)=(1.1035,2.1693,2.0191)$$

whose objective value is 3.56.

Example 4.13. Let us consider the following CCP in which there are three decision variables and nine stochastic parameters,

$$\begin{cases} \max \overline{f} \\ \text{subject to:} \\ \qquad \Pr\left\{\xi_1x_1+\xi_2x_2+\xi_3x_3 \ge \overline{f}\right\} \ge 0.90 \\ \qquad \Pr\left\{\eta_1x_1^2+\eta_2x_2^2+\eta_3x_3^2 \le 8\right\} \ge 0.80 \\ \qquad \Pr\left\{\tau_1x_1^3+\tau_2x_2^3+\tau_3x_3^3 \le 15\right\} \ge 0.85 \\ \qquad x_1,x_2,x_3 \ge 0 \end{cases}$$

where ξ_1,η_1, and τ_1 are uniformly distributed random variables $\mathcal{U}(1,2)$, $\mathcal{U}(2,3)$, and $\mathcal{U}(3,4)$, respectively, ξ_2,η_2, and τ_2 are normally distributed random variables $\mathcal{N}(1,1)$, $\mathcal{N}(2,1)$, and $\mathcal{N}(3,1)$, respectively, and ξ_3, η_3, and τ_3 are exponentially distributed random variables $\mathcal{EXP}(1)$, $\mathcal{EXP}(2)$, and $\mathcal{EXP}(3)$, respectively,

We employ stochastic simulation to generate input-output data for the uncertain function $U : \boldsymbol{x} \to (U_1(\boldsymbol{x}), U_2(\boldsymbol{x}), U_3(\boldsymbol{x}))$, where

$$\begin{aligned}
U_1(\boldsymbol{x}) &= \max\left\{\overline{f} \;\middle|\; \Pr\left\{\xi_1 x_1 + \xi_2 x_2 + \xi_3 x_3 \ge \overline{f}\right\} \ge 0.90\right\},\\
U_2(\boldsymbol{x}) &= \Pr\left\{\eta_1 x_1^2 + \eta_2 x_2^2 + \eta_3 x_3^2 \le 8\right\},\\
U_3(\boldsymbol{x}) &= \Pr\left\{\tau_1 x_1^3 + \tau_2 x_2^3 + \tau_3 x_3^3 \le 15\right\}.
\end{aligned}$$

Then we train an NN (3 input neurons, 15 hidden neurons, 3 output neurons) to approximate the uncertain function U. Finally, we integrate the trained NN and GA to produce a hybrid intelligent algorithm.

A run of the hybrid intelligent algorithm (5000 cycles in simulation, 3000 training data in NN, 1000 generations in GA) shows that the optimal solution is

$$(x_1^*, x_2^*, x_3^*) = (1.458,\ 0.490,\ 0.811)$$

with objective value $\overline{f}^* = 2.27$.

Example 4.14. Let us now turn our attention to the following DCGP,

$$\begin{cases}
\text{lexmin}\left\{d_1^- \vee 0, d_2^- \vee 0, d_3^- \vee 0\right\}\\
\text{subject to:}\\
\qquad 0.92 - \Pr\{x_1 + x_4^2 = 4\} = d_1^-\\
\qquad 0.85 - \Pr\{x_2^2 + x_6 = 3\} = d_2^-\\
\qquad 0.85 - \Pr\{x_3^2 + x_5^2 + x_7^2 = 2\} = d_3^-\\
\qquad x_1 + x_2 + x_3 \le \xi_1\\
\qquad x_4 + x_5 \le \xi_2\\
\qquad x_6 + x_7 \le \xi_3\\
\qquad x_i \ge 0, \quad i = 1, 2, \cdots, 7
\end{cases}$$

where ξ_1, ξ_2, and ξ_3 are uniformly distributed random variable $\mathcal{U}[3,5]$, normally distributed random variable $\mathcal{N}(3.5,1)$, and exponentially distributed random variable $\mathcal{EXP}(9)$, respectively.

In the first priority level, there is only one event $\mathcal{E}_1$ which will be fulfilled by $x_1 + x_4^2 = 4$. It is clear that the support $\mathcal{E}_1^* = \{x_1, x_4\}$ and the dependent support $\mathcal{E}_1^{**} = \{x_1, x_2, x_3, x_4, x_5\}$. Thus the dependent constraints of $\mathcal{E}_1$ are

$$x_1 + x_2 + x_3 \le \xi_1, \quad x_4 + x_5 \le \xi_2, \quad x_1, x_2, x_3, x_4, x_5 \ge 0.$$

It follows from the principle of uncertainty that the chance function $f_1(\boldsymbol{x})$ of $\mathcal{E}_1$ is

$$f_1(\boldsymbol{x}) = \Pr\left\{\begin{array}{l} x_1 + x_4^2 = 4 \\ x_1 + x_2 + x_3 \le \xi_1 \\ x_4 + x_5 \le \xi_2 \\ x_1, x_2, x_3, x_4, x_5 \ge 0 \end{array}\right\}.$$

At the second priority level, there is an event $\mathcal{E}_2$ which will be fulfilled by $x_2^2 + x_6 = 3$. The support $\mathcal{E}_2^* = \{x_2, x_6\}$ and the dependent support $\mathcal{E}_2^{**} = \{x_1, x_2, x_3, x_6, x_7\}$. Thus the dependent constraints of $\mathcal{E}_2$ are

$$x_1 + x_2 + x_3 \le \xi_1, \quad x_6 + x_7 \le \xi_3, \quad x_1, x_2, x_3, x_6, x_7 \ge 0.$$

The principle of uncertainty implies that the chance function $f_2(\boldsymbol{x})$ of the event $\mathcal{E}_2$ is

$$f_2(\boldsymbol{x}) = \Pr\left\{\begin{array}{l} x_2^2 + x_6 = 3 \\ x_1 + x_2 + x_3 \le \xi_1 \\ x_6 + x_7 \le \xi_3 \\ x_1, x_2, x_3, x_6, x_7 \ge 0 \end{array}\right\}.$$

At the third priority level, there is an event $\mathcal{E}_3$ which will be fulfilled by $x_3^2 + x_5^2 + x_7^2 = 2$. The support $\mathcal{E}_3^* = \{x_3, x_5, x_7\}$ and the dependent support $\mathcal{E}_3^{**}$ includes all decision variables. Thus all constraints are dependent constraints of $\mathcal{E}_3$. It follows from the principle of uncertainty that the chance function $f_3(\boldsymbol{x})$ of the event $\mathcal{E}_3$ is

$$f_3(\boldsymbol{x}) = \Pr\left\{\begin{array}{l} x_3^2 + x_5^2 + x_7^2 = 2 \\ x_1 + x_2 + x_3 \le \xi_1 \\ x_4 + x_5 \le \xi_2 \\ x_6 + x_7 \le \xi_3 \\ x_1, x_2, x_3, x_4, x_5, x_6, x_7 \ge 0 \end{array}\right\}.$$

We encode a solution into a chromosome $V = (v_1, v_2, v_3, v_4)$, and decode a chromosome into a feasible solution in the following way,

$$\begin{array}{llll} x_1 = v_1, & x_2 = v_2, & x_3 = v_3, & x_4 = \sqrt{4 - v_1}, \\ x_5 = v_4, & x_6 = 3 - v_2^2, & x_7 = \sqrt{2 - v_3^2 - v_4^2}. & \end{array}$$

We first employ stochastic simulation to generate input-output data for the uncertain function $U : (v_1, v_2, v_3, v_4) \to (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), f_3(\boldsymbol{x}))$. Then we train an NN (4 input neurons, 10 hidden neurons, 3 output neurons) to approximate it. Finally, we embed the trained NN into a GA to produce a hybrid intelligent algorithm.

A run of the hybrid intelligent algorithm (6000 cycles in simulation, 3000 data in NN, 1000 generations in GA) shows that the optimal solution is

$$\boldsymbol{x}^* = (0.1180, 1.7320, 0.1491, 1.9703, 0.0000, 0.0000, 1.4063)$$

which can satisfy the first two goals, but the third objective is 0.05.