

Chapter 5

RELIABILITY STOCHASTIC OPTIMIZATION WITH BRANCH AND BOUND TECHNIQUE

*An Application of Stochastic Programming with
Branch and Bound technique - n stage series system
with m chance constraints.*

A modified version of this chapter is submitted to South African Journal of Science.

5.1 INTRODUCTION

In the past three decades, numerous reliability optimization techniques have been proposed (Tillman et al 1977, 1980, Kuo et al. 1987, Chen 1992). Stochastic programming models for general redundancy-optimization problems have been studied by (Zhao et al 2003). Stochastic programming models arise as reformulations or extensions of reliability optimization problems with random parameters. Moreover, the resource elements vary and it is reasonable to regard them as stochastic variables. Problems in this area are not easy to solve. Most researchers in this area concentrated on developing approximate solution methods as optimal solutions. However, efficiency in the complex theoretical aspect is usually not considered. Quality statements mostly remain restricted to convergence to an optimal solution without accompanying implications on the running time of the algorithms for attaining most accurate solutions. Very recently the complexity of stochastic programming problems has been addressed, confirming these problems are harder than most combinatorial optimization problems.

This chapter addresses chance constrained reliability stochastic optimization (CCRSO) problem. Chance constraint programming technique has been first proposed by (Charnes and Cooper, 1954). The objective is to maximize system reliability for the given chance constraints. A methodology is illustrated to determine optimal solutions to n stage series system with m chance constraints of the redundancy allocation problem. Various cases of randomness have been discussed with known distributions like Uniform, Normal, and Log-normal distributions, when the resource variables are random. Once the real number solution is obtained using the technique of chance constraint, the B&B technique is used to obtain the integer

solution. In this chapter, a 4-stage series system with two chance constraints is numerically illustrated for the redundancy allocation problem.

This chapter has been organized as follows, stochastic integer programming problem for n stage series system with m chance constraint discussed and then the required algorithm to get integer solution is provided along with numerical example, which illustrate the model effectively.

1.1 Stochastic Integer Programming (SIP): n Stage Series System with m Chance Constraints

The chance constraint optimization problem for n stage series system with m chance constraint can be formulated as

$$\text{Max } R_s(X) = \prod_{j=1}^n [1 - (1 - r_j)^{x_j}] \quad (5.1)$$

Subject to, $P[g_i(x) \leq b_i] \geq 1 - \alpha_i, \quad i = 1, 2, \dots, m; \quad x_j \geq 1, j = 1, 2, \dots, n$, where resource vector b is random in nature,

R_s - reliability of the system

r_j, q_j - reliability, unreliability of components j ; $r_j + q_j = 1$

x_j - number of components used at stage j

$g_i(x)$ - chance constraint i

b_i - amount of resource i available (random

α_i - level of significance.

5.1.1 Case 1: b follows uniform distribution

Let $b_i \sim U(l_i, u_i)$, the constraint in system (1) is equivalent to $g_i(x) \leq \tau_i$, where $\beta_i = 1 - \alpha_i$,

$$\int_{\tau_i}^{u_i} \left(\frac{dx}{u_i - l_i} \right) = \beta_i$$

$$\tau_i = \alpha_i u_i + \beta_i l_i.$$

Hence, the deterministic equivalent of system (5.1) is:

$$\text{Max} R_s(X) = \prod_{j=1}^n [1 - (1 - r_j)^{x_j}]$$

(5.2)

subject to

$$g_i(x) \leq \alpha_i u_i + \beta_i l_i, \quad i = 1, 2, \dots, m; \quad x_j \geq 1, j = 1, 2, \dots, n.$$

5.1.2 Case 2: b follows normal distribution

Let $b_i \sim N(b_i, \sigma_{b_i}^2)$, where $\mu_{b_i}, \sigma_{b_i}^2$ are mean and variance of the normal random variable b_i . Using the i^{th} chance constraint of the system (5.1), restate the chance constraint as $P[b_i \geq g_i(x)] \geq 1 - \alpha_i$, $i = 1, 2, \dots, m$, this expression can be further stated as $P[(b_i - \mu_{b_i})/\sigma_{b_i} \geq (g_i(x) - \mu_{b_i})/\sigma_{b_i}] \geq 1 - \alpha_i$, $i = 1, 2, \dots, m$.

Using the cumulative density function of the standard normal random variable, it can be simplified as:

$$1 - \Phi[(g_i(x) - \mu_{b_i})/\sigma_{b_i}] \geq 1 - \alpha_i, \quad i = 1, 2, \dots, m,$$

where

$$\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^z \exp\left(-\frac{t^2}{2}\right) dt.$$

This can be further simplified as

$$\Phi[(g_i(x) - \mu_{b_i})/\sigma_{b_i}] \leq \Phi(-K_{\alpha_i}), \quad i = 1, 2, \dots, m.$$

The chance constraint can be transformed into deterministic constraint as

$$g_i(x) \leq \mu_{b_i} - \sigma_{b_i} K_{\alpha_i}, \quad i = 1, 2, \dots, m.$$

Hence, the deterministic equivalent of system (1) is:

$$\text{Max} R_s(\mathbf{X}) = \prod_{j=1}^n [1 - (1 - r_j)^{x_j}] \quad (5.3)$$

subject to

$$g_i(x) \leq \mu_{b_i} - \sigma_{b_i} K_{\alpha_i}, \quad i = 1, 2, \dots, m; \quad x_j \geq 1, \quad j = 1, 2, \dots, n.$$

5.1.3 Case 3: b follows log-normal distribution

Let $b_i \sim LN(\mu_i, \sigma_i^2)$, where μ_i, σ_i^2 are the mean and variance of the log normal random variable b_i . Using the i^{th} chance constraint of the system (5.1), we restate the chance constraint as

$$P[\ln b_i \geq \ln g_i(x)] \geq 1 - \alpha_i, \quad i = 1, 2, \dots, m.$$

This expression can be further stated as $P[(\ln b_i - \mu_i)/\sigma_i \geq (\ln g_i(x) - \mu_i)/\sigma_i] \geq 1 - \alpha_i, \quad i = 1, 2, \dots, m$. The following deterministic i^{th} constraint is obtained by similar arguments made in case 2.

$$g_i(x) \leq \exp(\mu_i - \sigma_i K_{\alpha_i}), \quad i = 1, 2, \dots, m.$$

Hence, the deterministic equivalent of system (1) is:

$$\text{Max} R_s(X) = \prod_{j=1}^n [1 - (1 - r_j)^{x_j}] \quad (5.4)$$

subject to

$$g_i(x) \leq \exp(\mu_i - \sigma_i K_{\alpha_i}), \quad i = 1, 2, \dots, m; \quad x_j \geq 1, \quad j = 1, 2, \dots, n.$$

5.2 General Algorithm

1. Convert the deterministic form of chance constraint into a linear constraint, adopting the technique of sequential linear programming (Rao 2000, Jeeva et al 2002,2004 , Charles and Dutta, 2003).
2. Code any one of the system (5.2) – (5.4) along with respective linearized constraint in MATLAB or LINGO and generate optimal solutions by inputting initial values using random function (in later stages one can use the existing real solution to generate integer solution using the step below given).
3. Apply the branch and bound algorithm given below to get integer solutions.

5.3 Branch-and-bound (B&B) technique

The B&B technique for CCRSO for stochastic optimization is given below:

1. Solve the problem as if all the variables were real numbers i.e. not integers, using the general algorithm given above. This solution is the upper bound (for maximization problem) of the CCRSO problem.

2. Choose one variable at a time that has a non-integer value, say x_j and branch that variable to the next higher integer value for one problem and to the next lower integer value for the other. The real valued solution of the variable j can be expressed as $x_j = [x_j] + x_j^*$, where $[x_j]$ is the integer part of x_j and x_j^* is the fractional part of x_j , $0 < x_j^* < 1$. The lower bound and upper bound constraints of the two mutually exclusive problems are $x_j = [x_j]$ and $x_j = [x_j] + 1$, respectively. Add these two constraints to both branched problems.

3. Now the variable x_j is an integer in either branch. Fix the integer of x_j for the following steps of branch-and-bound. Select the branch that yields the maximum objective function with all constraints satisfied. Then repeat step 2 on another variable $x_k \neq x_j$ for each of the new sub problems until all variables become integers.

4. Stop the particular branch if the solution is not satisfying the constraints of the original problem else stop the branch when all the desired integer values are obtained.

5.4 NUMERICAL EXAMPLE

Example 1

A four-stage system with chance constraints is formulated as a pure stochastic integer programming problem using the data given in table 5.1. The decision variables, $X = (x_1, \dots, x_4)$, are the number of redundancies at each stage. The problem is formulated as in Case 1.

Table 5.1: Data for Example 1

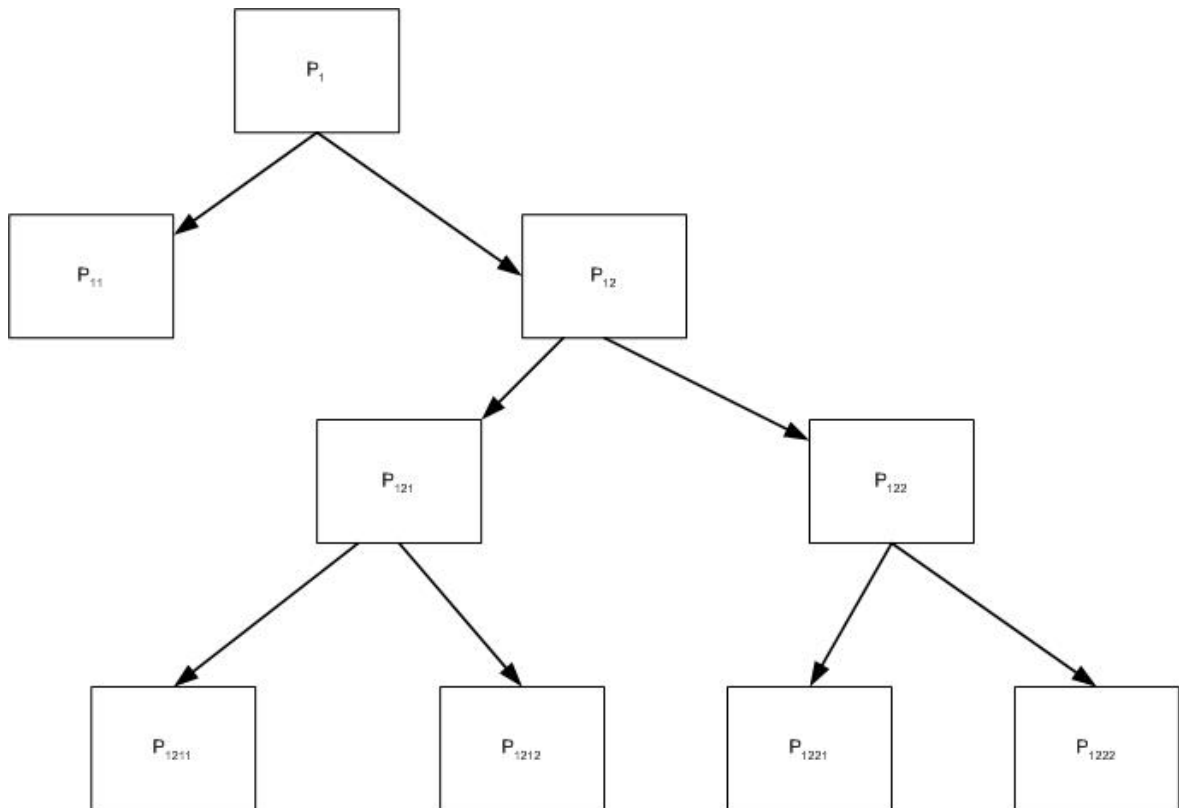
Stage, <i>j</i>	1	2	3	4	Available Resource			
r_j	0.75	0.80	0.75	0.85		l_i	u_i	α_i
c_{1j}	1.5	3.3	3.2	4.4	b_1	50	60	0.10
c_{2j}	4.0	5.0	7.0	9.0	b_2	110	140	0.15

Table 5.2: Solutions for Example 1

S. No.	Initial guess (obtained using rand())				x_1	x_2	x_3	x_4	$R_s(X)$
	x_1	x_1	x_2	x_3					
1	1.9501	1.2311	1.6068	1.4860	7.7656	9.5884	1.0344	1	1
2	1.8913	1.7621	1.4565	1.0185	10.857	8.2167	1	1	1
3	1.8214	1.4447	1.6154	1.7919	8.4843	8.6375	1	1.4931	1
4	1.9218	1.7382	1.1763	1.4057	7.7650	6.2088	1	1	0.9999
5	1.9355	1.9169	1.4103	1.8936	10.226	7.5664	1	1.7028	1
6	1.0579	1.3529	1.8132	1.0099	11.370	7.6831	1.3097	1	1
7	1.1389	1.2028	1.1987	1.6038	10.706	8.0460	1	1.1794	1
8	1.2722	1.1988	1.0153	1.7468	10.125	7.9687	1	1.4356	1
9	1.4451	1.9318	1.4660	1.4186	12.011	6.5778	1	1	1
10	1.8462	1.5252	1.2026	1.6721	9.3136	8.5091	1.0046	1.3034	1

With the data given in table 1, the real solutions are obtained using the general algorithm, which is exhibited in table 2. This paper suggests that the real solution be further elaborated by the B&B technique. Let us take one solution $X = (11.3697, 7.6831, 1.3097, 1.0000)$ from Table 5. 2. Now the integer solution is obtained using B&B technique. The following figure 1 gives clear picture about B&B network.

Figure 5. 1: A B&B Network Representation for Example 1



- P1** : $x_1 = 11.3697; x_2 = 7.6831; x_3 = 1.3097; x_4 = 1.0000; R = 1.0000$
P11 : **Fathomed**
P12 : $x_1 = 11.1175; x_2 = 7.1284; x_3 = 2.0000; x_4 = 1.0000; R = 1.0000$
P121 : $x_1 = 11.1175; x_2 = 7.0000; x_3 = 2.0000; x_4 = 1.0000; R = 1.0000$
P122 : $x_1 = 9.2000; x_2 = 8.0000; x_3 = 2.0000; x_4 = 1.0000; R = 1.0000$
P1211 : $x_1 = 11.0000; x_2 = 7.0000; x_3 = 2.0000; x_4 = 1.0000; R = 1.0000$
P1212 : **Fathomed**
P1221 : $x_1 = 9.0000; x_2 = 8.0000; x_3 = 2.0000; x_4 = 1.0000; R = 1.0000$
P1222 : **Fathomed**

Alternative optimal integer is obtained from the B&B process, $X = (11, 7, 2, 1)$ and $X = (9, 8, 2, 1)$.

5.5 CONCLUSION

The combination of the chance constraint technique and the B&B technique takes advantage of an exact method and enumerative method. In this paper the chance constraint technique, using MATLAB program, quickly reaches real solutions that is close to optimum. In addition, the B&B technique generates many sets of integer solutions. The competitive alternatives provide the management with several options and flexibility. Since a good approximation is obtained by the chance constraint technique, it does not take many branches for the B&B technique to reach the integer solution. The B&B algorithm given in this paper can be directly applied to the mixed integer stochastic programming problem (MISPP). For MISPP, only the integer variables need to be enumerated by the B&B procedure. The real variables are free of restriction after each step of the B&B technique.