Dhaka Univ. J. Sci. 74(2): 283–294, 2026 (July)
ISSN 1022-2502, E-ISSN 2408-8528
DOI: https://doi.org/10.3329/dujs.v74i2.84357

Analysis of Stochastic Integer Programming and Its Application in Real Life

M. Babul Hasan* **and Madina Akter**

Department of Mathematics, University of Dhaka, Dhaka-1000 Bangladesh

(Received: 16 September 2025; Accepted: 17 January 2026)

Abstract

Uncertainty is inherent in real-world decision-making. Traditional deterministic models often fail to capture such variability, leading to suboptimal or infeasible results. Stochastic programming provides a robust mathematical framework for optimization under uncertainty. This study presents an overview of stochastic integer programming (SIP) and its applications. After introducing the fundamentals of stochastic programming, we review existing literature and highlight key methodologies such as two-stage and multi-stage models, LP-relaxation, and recourse strategies. Practical examples solved using AMPL and Mathematica are provided to demonstrate the applicability of SIP in real-life decision problems. The findings emphasize the significance of SIP as a versatile tool for optimization under uncertainty and as a foundation for future research.

Keywords: Stochastic programming, Integer programming, Optimization under uncertainty, Two-stage models, AMPL, LP-relaxation.

I. Introduction

Mathematical optimization plays a vital role in decision-making across diverse domains such as supply chain management, finance, energy planning, and manufacturing. Many real-world decisions involve uncertain data, which traditional deterministic linear programming (DLP) models fail to capture adequately. Stochastic programming (SP) extends these models by incorporating probabilistic elements to represent uncertain parameters, thus providing more reliable and realistic decisionsupport[1-5].

The foundations of stochastic programming were laid by Dantzig[6], who introduced linear programming under uncertainty. Subsequent contributions, such as those by Van Slyke and Wets (1969), developed decomposition techniques like the L-shaped method for solving two-stage stochastic programs. Birge, J. R.[3], "Decomposition and partitioning methods for multistage stochastic linear programs Haneveld and Vlerk[9] provided a comprehensive overview of stochastic integer programming models and algorithms, highlighting their applications in industry.Laporte G., & Louveaux F. V., "The Integer L-shaped Method for Stochastic Integer Programs with Complete Recourse", Operations Research Letters[13] Grassmann, H. I., & A. M. Ireland[12], Scenario Formulationin an Algebraic Modelling Languages. Schultz R., Stougie L., & Vlerk M. H.,[19], Two-Stage Stochastic Integer Programming: A Survey.Van Slyke, R. M., & Wets, R. J.-B.[19]. L-shaped linear programs with applications to optimal control and stochastic programming. Recent works, including Ahmed[1], have focused on two-stage SIP with fixed recourse, introducing computationally efficient algorithms like the decomposition-based L-shaped method. Other studies, such as Carøe and Schultz[4], extended decomposition techniques to multi-stage SIPs using dual relaxation approaches. Collectively, these works form the theoretical and computational foundation of SIP, motivating its application in real-world decision problems.

These were the most important paper that helped most in our research. Besides these, we have gone through many more journals relative to our research topics. These papers were very helpful to develop and enhance our understanding. Also, these give us inspiration and direction for further research on this field. In the next section, we will reflect the knowledge that we have gathered from these papers.

This paper is organized as follows. Section 1 gives the basic concepts and properties of stochastic optimization and contains some literature review that is relevant to our work. Section 2 gives detail information about SP models and their solution procedure. Section 3 is about the part that is our main concept in this research. Here given stochastic integer programming and its application. Section 4 presents a detailed discussion of the results obtained in this research. conclusion of this research. Section 5 draws a conclusion of this research.

II. Methodology and Formulation of Stochastic Programs

Optimization under uncertainty can be approached through stochastic programming, which incorporates probability distributions for uncertain parameters.

Stochastic programming

In the field of mathematical optimization, stochastic programming is a framework for modeling optimization problems that involve uncertainty. Stochastic programs are mathematical programs where some of the date incorporated

* Author for correspondence. e-mail: babulhasan.math@du.ac.bd

into the objective function or constraints is uncertain. It is an optimization problem in which some or all parameters are uncertain but follow known probability distributions. Stochastic programming extends the scope of linear and nonlinear programming to include probabilistic or statistical information about one or more uncertain problem parameters. Similarly, when all the input data used in the mathematical formulation of the mathematical program is known with certainty then the corresponding model are called deterministic models.

In a typical optimization formulation:

Minimize $z = cz$ (2.1)

subject to $Ax \geq b, x \geq 0$

Uncertainty about the demand b, uncertainty about the input prices c, and uncertainty about the technical coefficient matrix A can be the treated in stochastic programming. These three types of uncertainty are related to parameters in stochastic programming. An ordinary LP problem becomes stochastic when any or all elements of the set $\beta = (A, b, c)$ of the parameters depends on random set of natures $i.e. \beta = \beta(s), s \epsilon W$ where w is the index set.

Stochastic integer programming (SIP), a subclass of SP, combines stochastic optimization with integrality constraints, making it highly relevant for problems involving discrete decisions. Applications include facility location, capital budgeting, logistics, and resource allocation under uncertainty. This paper explores the concepts, methodologies, and applications of SIP, supported by worked examples and computational results.

Most of the optimization problems applied in the real life present uncertain data: production costs and transport depend on fuel price, future demands depend on the uncertain market conditions or crop returns depend on the weather, among others. If we suppose that all the parameters are known, it could be produced not satisfactory result, or even disastrous. Hence, it seems more accurate to model the optimization problems taking into account unknown parameters. In, fact Stochastic Programming is an alternative to the deterministic problems.Stochastic Linear Programming are those linear optimization problems where some of the parameter's c, A,$b_l, and b_u$ of the model (2.1) are uncertain. So, uncertainty can be defined by random variables in the form of probability distributions, densities, or in general, probability measures.

Uncertainty appears everywhere. It seems that "uncertainty is reality". In real life we cannot ignore any possible uncertainty. It may be, Weather Related, Market Related Uncertainty, Demand in Supply Chain, Financial Uncertainty- Return in financial market, Noisy, biased, incomplete data in machine learning, Model Uncertainty, state uncertainty in sequential decision making, Competition, Technology Related, Arts of God

In an analysis of a decision, we would proceed through this list and identify those items that might interact with our decision in a meaningful way. For optimization under uncertainty stochastic programming is one of the best techniques. Stochastic programming models have the general characteristic of mathematical programming problems with the additional feature that one or some of the parameters are random variables. The stochastic programming problems can be classified as shown in Fig 2.1.

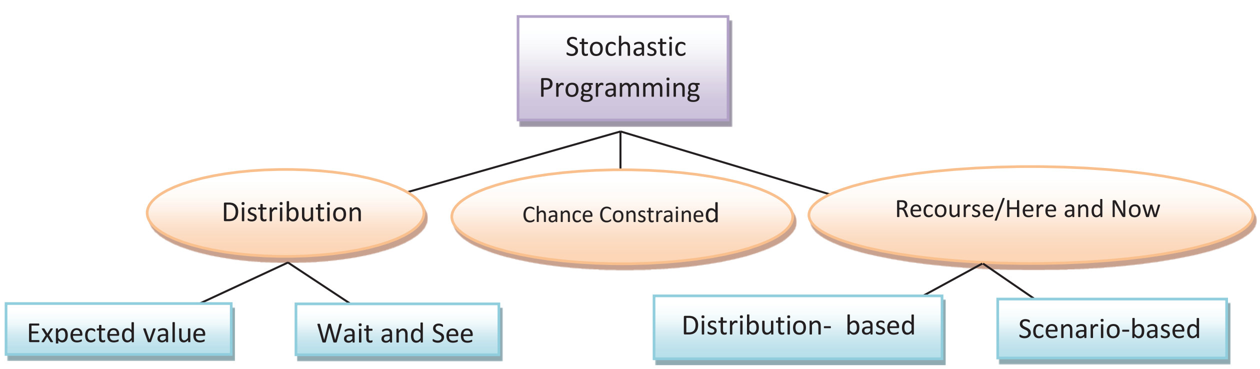

Fig. 2.1. Classification of SP

Two main formulations are commonly used:

Two-Stage Models

Decisions are divided into first-stage (here-and-now) and second-stage (wait-and-see) variables[1,2]. The first stagefixes initial decisions, while the second stage adapts to realized uncertainty.Computation in stochastic programs with recourse has focused on two-stage problems with finitenumbers of realizations. In this method, the set of decisions is divided into two groups such as First- stage decisions which involves the decisions that have to be taken before the experiment$i.e.$ without full information on some random events. And the period when these decisions are taken is called the first -stage. These decisions are usually represented bya vector$x$. And the other one is Second- stage decisions which involves the decisions that are taken after the experiment. The corresponding periodis called the second stage. When full information is received on the realization of some random vector $\xi(\omega)$ then, corrective actions$y(s)$ are taken to show explicit dependence on an underlying elementω or s.

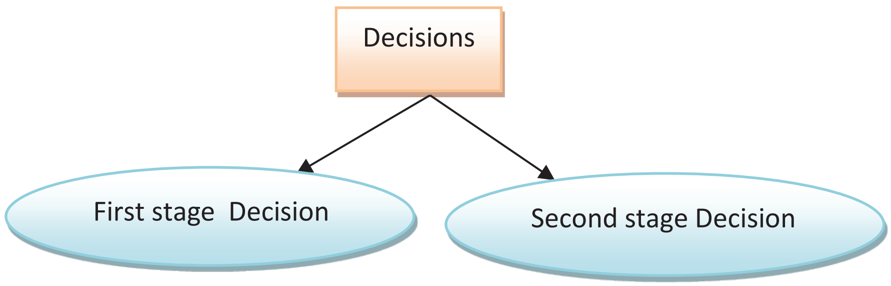

Fig. 2.2. Two-stage Decisions

We have to make a decision in the first stage. The principle of the SP problems can then be employed to make further decision in the second stage to avoid the problems constraints becoming infeasible.

General Form of Two-Stage Method

The general model is to choose some initial decision that minimizes current plus the expected value of future recourse actions. With a finite number of second stage realizations and all linear functions, we can always form the full deterministic equivalentlinear program or extensive form. Basically, the main idea of the concept of the model is the idea of recourse, which defines possibility to take right actions after a realization of the random event. General Mathematical Formulation is as follows.

Min $c^T x + E_\xi[Q(x,\xi)]$	Min cost: 1^{st}stage + Expected 2^{nd}stage
Subject to $Ax \le b$	1^{st}stage constraints
$x \ge 0$	Non negative restrictions
Where $Q(x,\xi) = min q^T y$	2^{nd}stage recourse action
Subject to $Tx + Wy \le h$	2^{nd}stage constraints
$y \ge 0$	Non negative restrictions

Fig 2.3. Two -Stage Model

Here ξ is the vector formed by the components which areq^T, h^T, W and T. E_ξ denotes the mathematical expectations with respect to ξ.

In the objective function, the first term is known as the first stage and the second term is known as the second stage decision. There is a difference between these two. The first-stage decisions are represented by the $n_1 \times 1$ $vector$ x. Corresponding to x are the first-stage vectors and matrices c, b and A, of sizes $n_1 \times 1$, $m_1 \times 1$ and $m_1 \times n_1$ respectively. In the second-stage, a number of random events$\omega \in \Omega$ may realize. For a given realization ω , the second –stage problem date$q(\omega)$, $h(\omega)$ and $T(\omega)$ become known, $q(\omega)$ is $n_2 \times 1$, $h(\omega)$ is $m_2 \times 1$ and $T(\omega)$ is $m_2 \times n_1$. Each component of q, T and h is thus a possible random variable. Let $T_i(\omega)$ be the ith row of $T(\omega)$. Piecing together the stochastic components of the second-stage data, weobtain a vector $\xi^T(\omega) = \left(q(\omega)^T,\ h(\omega)^T,\ T_{m1}(\omega), \dots\dots, T_{m2}(\omega)\right)$, with potentially up to $N = n_2 + m_2 + (m_2 \times n_1)$ components. In matrix formis as follows.

$$A = \begin{pmatrix} a_{11} & a_{12} \dots\dots & a_{1n} \\ a_{21} & a_{22} \dots\dots & a_{2n} \\ & \dots\dots\dots\dots\dots\dots\dots & \\ a_{m1} & a_{m2} & a_{mn} \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}, \quad x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix},$$

$$T = \begin{pmatrix} a_{11}(\omega) & a_{12}(\omega) \dots\dots & a_{1n}(\omega) \\ a_{21}(\omega) & a_{22}(\omega) \dots\dots & a_{2n}(\omega) \\ & \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots & \\ a_{m1}(\omega) & a_{m2}(\omega) & a_{mn}(\omega) \end{pmatrix}, H(\omega) \begin{pmatrix} h_1(\omega) \\ h_2(\omega) \\ \vdots \\ h_m(\omega) \end{pmatrix}$$

General Form of Multi-Stage Method

Decisions are made sequentially across multiple stages, incorporating evolving information and random events. We can see from the two-stage SP that the uncertainties are realized only after the first stage judgments are made. We will look at the stochastic programming method to these multi-stage problems in this section. We offer the same fundamental findings as in earlier sections. In real life practice, most of the problems are sequential decision problems, in which decisions are made at multiple stages. These are called “Multi-Stage Stochastic Problems”.

The multi-stage problems are difficult to solve than two-stage problems. Such problems are formulated as a mixed-integer linear programming model and then Bender’s decomposition algorithm is applied for solving it. Between two subsequent decisions, we may observe the random effects influencing our system; the next decisions take these into account, in addition to available information coming from past history. Decomposition technique is the easiest to solve these problems when random variables are successively independent. The general form of multi-stage SP is given as follows:

$$\min c_1^T x_1 + \mathbb{E}_{\xi[2,H]|||\xi[1,1]} \{\min c_1^T(\xi_2) x_2(\xi_2) + \cdots + \mathbb{E}_{\xi[H:H]|||\xi[1:H-1]} [\min c_H^T(\xi_H) x_H(\xi_H)]\}$$

subject to$W_1 x_1 \leq h_1$

$$T_2(\xi_2)x_1 + W_2 x_2(\xi_2) \leq h_2(\xi_2)$$

....

....

$$T_H(\xi_H)x_H + W_H x_H(\xi_H) \leq h_H(\xi_H)$$

$$x_1 \in X_1 \; ; \; x_t(\xi_t) \in X_{t,} \; t = 2, \dots\dots, \quad H$$

Where H denotes the stages. We can assume that all the constraints and objectives functions are linear. Consider the data$(\xi_2 , \dots\dots \xi_H)$ is uncertain. Here ξ_t is used to denote the random data vector in stage *t*. The parametersc_t , T_t , W_t , h_t are the function of ξ_t. We are similarly use $\xi_{[t,t']}$ to indicate the sequence of random data vectors corresponding to the stage t through t'. The decisions dynamics as follows. In stage t we first observe the data realization ξ_t and then taken anaction of x_t depending on the previous stage decisions x_{t-1} and the observed data ξ_t to optimize the futureexpected cost.$\mathbb{E}_{\xi[t,H]|||\xi[1,t-1]}$denotes three expectation operation in stage$t$ with respect to the conditional distribution of $\xi[t, H]$ given realization $\xi[1, t-1]$in stage $t-1$.The constraints are considered for all the possible realizations of the stochastic process till the stage t.Set$x_t$as the domain of variablesx_t, which can be continuous or mixed integer.

So the above equations can be represented as the deterministic equivalent of any multi-stage SLP with discrete or continuous distributions.

Theorem 3.1: Let $K_1 = \{x \setminus Ax \leq b \; x \geq 0\}$ be the set determined by the fixed constraints, namely those that do not depend on the particular realization of the random vector. For any given ξ, we may define a so-called elementary feasibility set as $K_2 = \{x \setminus y \geq 0 \; exists \; s.t. W(\omega)\mathrm{y} = \mathrm{h}(\omega) - \mathrm{T}(\omega)\mathrm{x} \}$. For a given ξ, the value function $Q(x, \xi)$ is a piecewise linear convex function in(h, T), a piecewise linear concave function in q, a piecewise linear convex function in x; for all $x \in K_2$. When ξ is a finite discrete random variable $Q(x)$is a piecewise linear concave function inK_2 .

The general stochastic programming problem can be expressed as minimizing an expected cost function subject to constraints involving random parameters. For integer problems, integrality restrictions are imposed on either first-stage or second-stage variables, resulting in stochastic integer programs.

Solution Procedure

Stochastic programs, as real world problems, are often computationally diffucult to solve. Before solving the stochastic model, we could be tempted to solve simpler problems: for examples, we could simplify the real imprecise data, replacing the unknown parameter with expected values those random variables and solve the obtained deterministic problem or alternatively, we could solve all related scenario sub models and compute the expectation of these different solutions. Those are:

Wait-and-See Solution (WS): Wait-and-See models assume that the decision maker is able to wait till the uncertainty is over, before calculating the optimal solutions.

Expected Value Problem (EVP): A usual simplification consist replacing each random variable with estimation, for example the expected value.

Expected Result of using the EV Solution (EEV)

This gives the expected result fixing the first-stage solution of the EV problem. The way to know if the simplified model is good enough is calculating these two measures are as the Expected value of perfect information (EVPI) where EVPI is defined as the difference between the wait-and-see, WS, and here-and-now solution, SP. In minimization problems: EVIP=SP=WS. And the value of the stochastic solution (VSS)where the VSS is defined as the difference between the Expected result of using EV problem and here-and-now solution, SP. In minimization problems: VSS=EEV-SP.

In this Section, we have given the basic techniques of modeling, important theorem and solution procedure of solving two-stage stochastic programming. In next Section we will discuss about stochastic integer programming and its application.

III. Stochastic Integer Programming and Applications

Stochastic integer programming (SIP) arises when some decision variables in a stochastic program are constrained to be integers. Applications typically involve binary first-stage decisions (e.g., facility openings, capital investment) and integer second-stage variables (e.g., resource allocations, production levels). Deterministic optimization problems are formulated with known parameters, but in real world problems almost invariably include some unknown parameters. In the field of mathematical optimization, stochastic programming is a framework for modeling optimization problems that involve uncertainty. In this section[4,5]. We will discuss about stochastic integer programs, LP –Relaxation, First-stage Binary Variables, Second-stage Binary Variables and applications. This section is our main concept in this research.

Stochastic integer programs with recourse (SIP) is a stochastic program, where some of the decisions are restricted to be integer. The general formulation of two-stage integer program requires that some variables in either the first-stage or second-stage are integer (Binary).

$$\min_{x \in X} z = c^T x + E_\xi \min_y \{q^T(\omega) y | W(\omega) y = h(\omega) - T(\omega)x, y \in Y\}$$

Subject to $Ax = b$ (3.1)

where the first-stage decisions are represented by the $n_1 \times 1$ $vector\ \ x$. Corresponding to $x$are the first-stage vectors and matricesc, b and A, of sizes $n_1 \times 1$, $m_1 \times 1$ and $m_1 \times n_1$ respectively. In the second-stage, a number of random events $\omega \in \Omega$ may realize. For a given realization ω , the second–stage problem date$q(\omega)$, $h(\omega)$ and $T(\omega)$ become known, $q(\omega)$ is $n_2 \times 1$, $h(\omega)$ is $m_2 \times 1$ and $T(\omega)$ is $m_2 \times n_1$. Each component of $q,\ T\ and\ h$ is thus a possible random variable.

Here Y always contains integrality restrictions on y and in some case X also contains integrality restrictions on x. The second-stage program is

$$Q(x,\xi) = \min_y\{q^T(\omega)y|W(\omega)y = h(\omega) - T(\omega)x, y \in Y\} \quad (3.2)$$

And its expection can be used to obtain a deterministic equivalent program (DEP)

$$\min_{x\in X} z = c^T x + Q(x), \text{ Subject to } Ax = b, x \geq 0.$$

LP-Relaxation

LP-relaxation is used to approximate integer problems by relaxing integrality constraints[3]. It provides bounds for the optimal solution and serves as the foundation for branch-and-bound and cutting-plane algorithms. Linear programming relaxation is a standard technique for designing approximation algorithms for hard optimization problems. LP –Relaxation is a model created by dropping integer (or binary) constraints.

ILP Model $Max\ \ z = 2x_1 + 3x_2$

Subject to $4x_1 + 12x_2 \leq 33$, $10x_1 + 4x_2 \leq 35$, $x_1,\ \ x_2 \geq 0$ and integer

Optimal solution is (2, 2) with Optimal objective function value as 10.

LP-Relaxation of the ILP Model

$$Max\ \ z = 2x_1 + 3x_2$$

Subject to$4x_1 + 12x_2 \leq 33$, $10x_1 + 4x_2 \leq 35$, $x_1,\ \ x_2 \geq 0$

Optimal solution is (2.769, 1.827) and Optimal objective function value is 11.019. Optimal solution of IPL is never better than optimal solution of its LP-Reclamation. MAX: LP-Relaxation gives upper bound for the ILP optimal objectivefunction value. MIN: LP-Relaxation gives lower bound for the ILP optimal objectivefunction value.In this example, optimal profit in LP-Relaxation is 11.019. From this we know that the optimal profit inthe ILP model will be≤ 11.019 (11.109 is an upper bound).LP-Relaxation of x_i binary is $0 \leq x_i \leq 1$.

Cases of Stochastic Integer Programs

There are two casesin stochastic integer programs which are **First-stage Binary Variables** (Integrality restrictions are in X) and **Second-stage Integer Variables**(Integrality restrictions are in Y).

First-stage Binary Variables

Problems with binary first-stage decisions often employ decomposition methods and optimality cuts to achieve convergence. These models capture strategic decisions that must be made before uncertainty is resolved. When the first-stage variables are binary variables, it is possible to drive specific optimality cuts in order to obtain a finitely convergent algorithm based on a branching system. The proposed method easily extents to the caseof mixed first-stage variables, provided the tender variables are binary. We assume the existence of a lower bound on $Q(x)$. There exists a finite lower bound L satisfying

$$L \leq \min_x\{Q(x)|Ax = b, x \in X\}. \quad (3.3)$$

In (3.3), there is no restriction about the tightness, although it is desirable to have as large as possible.

Proposition: Let $x_i = 1, i \in S\ and\ x_i = 0, i \notin S$ be some first-stage feasible solution. Let$qs = Q(x)$be the corresponding recourse function value. The optimality cut

$$\theta \geq (qs - L)(\textstyle\sum_{i\in S} x_i - \sum_{i\notin S} x_i) - (qs - L)(|S| - 1) + L \quad (3.4)$$

is valid.

Proof: Define $\delta(x,S) = (\sum_{i\in S} x_i - \sum_{i\notin S} x_i)$ (3.5)

Now, is always less than or equal to . It is equal to only if. In that case, the right-hand side of (3.4) takes the value and constraint is valid as. In all other cases, is smaller than or equal, which implies that the right-hand side of (3.5) takes a value smaller than or equal to which by (3.3) is a valid lower bound for all feasible. Readers are more familiar with geometrical representations may see (3.5) as a half-space, in the space, situated above a line going through the two points.

Example 1

Consider a two-stage program, where the1^{st}stage contains binary variables and the second stage is given by,

$$Min\ \ z = -2y_1 - y_2 - 3y_3 - 5y_4$$

Subject to

$$y_1 + 2y_2 + y_3 + 3y_4 \leq \xi_1 - x_1 - x_2, \qquad 2y_1 + y_2 + 3y_4 \leq \xi_2 - x_1$$

$2y_1 + y_3 \leq \xi_3 - x_2$, $y_1 \leq \xi_4$, $y_1,\ y_2, y_3,\ y_4 \geq 0$, integer

Assume $\xi = (2,2,2,2\ )^T\ or\ \ (5,3,2,2\ )^T$with equal probability $\frac{1}{2}$. If the current iteration point is $x = (0,1\ )^T$, find a lower bound L on $Q(x)$ and drive cut of type $\theta \geq (qs - L)(\sum_{i\in S} x_i - \sum_{i\notin S} x_i) - (qs - L)(|S| - 1) + L$.

Solution: Since the 1^{st}stage decisions are binary, the largest values of $y$are obtained with$x = (0,0\ )^T$. To obtain a lower bound$L$, we simply apply the LP-Relaxation ($i.e.$ drop the required that$y$ should be integer) and solve the resulting LP

$$Min\ \ z = -2y_1 - y_2 - 3y_3 - 5y_4$$

Subject to

$$y_1 + 2y_2 + y_3 + 3y_4 \leq \xi_1, 2\,y_1 + y_2 + 3y_4 \leq \xi_2$$

$$2y_1 + y_3 \leq \xi_3, y_1 \leq \xi_4, y_1,\ y_2, y_3,\ y_4 \geq 0$$

The above LP is solved by AMPL and the solution is $y = (0{,}0{,}2{,}0\ )^T$ and $Q(x,\xi) = -6$.

For $\xi = (2{,}2{,}2{,}2\ )^T$, we get,

$$Min\ \ z = -2y_1 - y_2 - 3y_3 - 5y_4$$

Subject to

$$y_1 + 2y_2 + y_3 + 3y_4 \leq 2, \quad 2\,y_1 + y_2 + 3y_4 \leq 2, \quad 2y_1 + y_3 \leq 2$$

$$y_1 \leq 2, y_1,\ y_2, y_3,\ y_4 \geq 0$$

The AMPL solution is$y = (0{,}0{,}2{,}1\ )^T$ and $Q(x,\xi) = -11$. This results in, the lower bound as

$L = .5 \times -6\ + .5 \times -11 = -8.5$.

Second part

Here bacause and . For , the second stage becomes

$$Min\ \ z = -2y_1 - y_2 - 3y_3 - 5y_4$$

subject to$y_1 + 2y_2 + y_3 + 3y_4 \leq 1,\ 2\,y_1 + y_2 + 3y_4 \leq 2,\ 2y_1 + y_3 \leq 1,\ y_1 \leq 2, y_1,\ y_2,\ \ y_3,\ y_4 \geq 0$, integer

The above LP is solved by AMPL and the solution is$y = (0{,}0{,}1{,}0\ )^T$ and $Q(x,\xi) = -3$.

The second stage becomes,

$$Min\ \ z = -2y_1 - y_2 - 3y_3 - 5y_4$$

Subject to

$$y_1 + 2y_2 + y_3 + 3y_4 \leq 4, \quad 2\,y_1 + y_2 + 3y_4 \leq 3$$

$2y_1 + y_3 \leq 1, \qquad y_1 \leq 2, y_1,\ y_2, y_3,\ y_4 \geq 0$, integer

The AMPL solution is$y = (\,0{,}0{,}1{,}1)^T$ and $Q(x,\xi) = -8$.

We conclude that$qs = \frac{-3+(-8)}{2} = -5.5$

The optimality cut is $\theta \geq (qs - L)(\sum_{i \in S} x_i - \sum_{i \notin S} x_i) - (qs - L)(\mid S \mid -1) + L$

$$\theta \geq (-5.5 + 8.5)(x_2 - x_1) - 8.5$$

$\theta \geq 3(x_2 - x_1) - 8.5$.

Here we represented the program of AMPL.

Lower Bound program: For $\xi = (2{,}2{,}2{,}2\ )^T$

Table 3.1. Data file for $\xi = (2, 2, 2, 2\ )^T$

probability.mod

```
param n;
param m;
set J := {1..n};
set I := {1..m};

param C {J};
param A {I,J};
param B {J};

var Y {J} >=0;

 minimize z: sum {j in J} C[j] * Y[j];

 s.t. Constraint {i in I}:
  sum {j in J} A[i,j] * Y[j] <= B[i];
```

probability.dat

```
param n := 4;
param m := 4;
param C :=
        1     -2
        2     -1
        3     -3
        4     -5;

param A :     1      2      3      4:=
        1     1      2      1      3
        2     2      1      0      3
        3     2      0      1      0
        4     1      0      0      0;


param B :=
        1    2
        2    2
        3    2
        4    2;
```

Table 3.2. Output file for $\xi = (2,2,2,2\ )^T$

probability.run

```
reset;
model probability.mod;
data probability.dat;
expand z, Constraint;
option solver cplex;
solve;
display Y, z;
```

```
ampl: include probability.run
minimize z:
	-2*Y[1] - Y[2] - 3*Y[3] - 5*Y[4];

subject to Constraint[1]:
	Y[1] + 2*Y[2] + Y[3] + 3*Y[4] <= 2;

subject to Constraint[2]:
	2*Y[1] + Y[2] + 3*Y[4] <= 2;

subject to Constraint[3]:
	2*Y[1] + Y[3] <= 2;

subject to Constraint[4]:
	Y[1] <= 2;

CPLEX 20.1.0.0: optimal solution; objective -6
2 dual simplex iterations (2 in phase I)
Y [*] :=
1  0
2  0
3  2
4  0
;

z = -6
```

Table 3.3. Model and data file for optimality cut for $\xi = (2,2,2,2\ )^T$

Optimality cut program: For $\xi = (2{,}2{,}2{,}2\ )^T$

probability0.mod

```
param n;
param m;
set J := {1..n};
set I := {1..m};

param C {J};
param A {I,J};
param B {J};

var Y {J} >=0;

 minimize z: sum {j in J} C[j] * Y[j];

 s.t. Constraint {i in I}:
  sum {j in J} A[i,j] * Y[j] <= B[i];
```

probability0.dat

```
param n := 4;
param m := 4;
param C :=
	1	-2
	2	-1
	3	-3
	4	-5;

param A :	1	2	3	4:=
	1	1	2	1	3
	2	2	1	0	3
	3	2	0	1	0
	4	1	0	0	0;


param B :=
	1	1
	2	2
	3	1
	4	2;
```

Table 3.4. Output file for optimality cut for $\xi = (2,2,2,2\ )^T$

```
| probability0.run ×
reset;
model probability0.mod;
data probability0.dat;
expand z, Constraint;
option solver cplex;
solve;
display Y, z;
```

```
ampl: include probability0.run
minimize z:
	-2*Y[1] - Y[2] - 3*Y[3] - 5*Y[4];

subject to Constraint[1]:
	Y[1] + 2*Y[2] + Y[3] + 3*Y[4] <= 1;

subject to Constraint[2]:
	2*Y[1] + Y[2] + 3*Y[4] <= 2;

subject to Constraint[3]:
	2*Y[1] + Y[3] <= 1;

subject to Constraint[4]:
	Y[1] <= 2;

CPLEX 20.1.0.0: optimal solution; objective -3
2 dual simplex iterations (2 in phase I)
Y [*] :=
1  0
2  0
3  1
4  0
;

z = -3
```

In a similar way, we assume $\xi = (2,2,2,2\ )^T or\ (5,3,2,2\ )^T\ or\ (7,3,3,2)^T\ or (8,5,5,3\ )^T$ with equal probability $\frac{1}{4}$. If the current iteration point is $x = (0,1\ )^T$, find a lower bound L on $Q(x)$ and drive cut of type $\theta \geq (qs - L)(\sum_{i\in S} x_i - \sum_{i\notin S} x_i) - (qs - L)(| S | -1) + L$. Detailed results are shown in Table 3.5.

Table 3.5. Lower bound and optimality cut for different data

Probability	ξ	Lower bound	Optimality cut	Iteration	
				Phase I	Phase II
$\frac{1}{2}$	$\xi = (2,2,2,2)^T$	- 3	- 3	2	0
	$\xi = (5,3,2,2)^T$	- 11	- 8	2	1
	$\xi = (7,3,3,2)^T$	- 14	- 11	2	2
	$\xi = (8,5,5,3)^T$	- 20	- 17	2	1
$\frac{1}{4}$	$\xi = (2,2,2,2)^T$	- 3	- 3	2	0
	$\xi = (5,3,2,2)^T$	- 11	- 8	2	1
	$\xi = (7,3,3,2)^T$	- 14	- 11	2	2
	$\xi = (8,5,5,3)^T$	- 20	- 17	2	1

Table 3.6. shows that, when probability is decreased lower bound is also reduced.

Probability	Lower bound	Optimality cut	Optimal solution
$\frac{1}{2}$	- 8.5	$\theta \geq 3(x_2 - x_1) - 8.5$	- 17
$\frac{1}{4}$	- 12.75	$\theta \geq 3(x_2 - x_1) - 12.75$	- 17

Second-stage Integer Variables

When second-stage decisions are integers, the problem becomes computationally challenging. Such models represent operational adjustments made after uncertainties are realized, such as demand fluctuations or cost variations.

We consider the case where the second-stage decisions areinteger, the random variable has a distribution, the technology matrix T is fixed and the recourse matrices$W_k$have integer coefficients. The latter can always be achieved by rescaling the second-stage constraints if the initial coefficients are rational.

The value of the second-stage program for one realization ξ_k reads as

$$Q(x,\xi_k) = \min_y\{q_k^T\, y \mid W_k y \geq\ h_k - Tx, y \in Z^{n_2}\} \quad (3.6)$$

The corresponding value function based on the tenders is

$$\psi(\chi,\xi_k) = \min_y\{q_k^T\, y \mid W_k y \geq\ h_k - \chi,\ \ y \in Z^{n_2}\} \quad (3.7)$$

As usual, $Q(x) = E_\xi Q(x,\xi) = \sum_{k=1}^{k} p_k\, Q(x,\xi_k)$. Similarly,

$$\psi(\chi) = E_\xi \psi(\chi,\xi_k) = \sum_{k=1}^{k} p_k\ \psi(\chi,\xi_k)\ . \quad (3.8)$$

For any, $Q(x) = \psi(-Tx)$. The classical problem$\min_x\{c^T x + Q(x) \mid x \in X\}$is thus equivalent to

$$Z^* = \min_{x,\chi}\{c^T x + \psi(\chi) \mid x \in X\ ,\ \ \chi = -Tx\} \quad (3.9)$$

The idea of branching on tenders is to partition the space of tenders $\chi = -Tx$ in an orthogonal partition and to use the non-decreasing property of the value function as a function of one component of the tender.

Example 2

Consider the following second-stage program with second-stage integer variables for one particular value of ξ

$$Q(x,\xi) = \min 5y_1 + 3y_2$$

Subject to

$2y_1 + 3y_2 \geq -3 + x_1 + 2x_2$, $4y_1 + y_2 \geq -2.4 + x_1 + x_2$, $y_1, y_2 \geq 0$, integer.

Due to the integer$y$, $Q(x,\xi)$ can only take finitely many different values. Describe the tender space and x- space regions where $Q(x,\xi)$ takes a given value.

Solution: If $y = (0,0)^T$ then $Q(x,\xi) = 0$, which is optimal$i.e.$

Region $R_1 = \{x \mid x_1 + 2x_2 \leq 3,\ x_1 + x_2 \leq 2.4\}$. This is convex.

If $y = (0,1)^T$ then $Q(x,\xi) = 3$, which is optimal$i.e.$

Region $R_2 = \{x \mid x_1 + 2x_2 \leq 6,\ x_1 + x_2 \leq 3.4\} \setminus R_1$. This is non- convex, due to the$x \notin R_1$ condition.

.If$y = (1,0)^T$ then $Q(x,\xi) = 5$, which is optimal $i.e.$

Region $R_3 = \{x \mid x_1 + 2x_2 \leq 5\} \setminus R_1 \setminus R_2$. This is convex.

.If$y = (0,2)^T$ then $Q(x,\xi) = 6$, which is optimal $i.e.$

Region $R_4 = \{x \mid x_1 + x_2 \leq 4.4\} \setminus R_1 \setminus R_2 \setminus R_3$. This is non- convex.

.If$y = (1,1)^T$ then $Q(x,\xi) = 8$, which is optimal $i.e.$

Region $R_5 = \{x \mid x_1 + 2x_2 \leq 8,\ x_1 + x_2 \leq 7.4\} \setminus R_1 \setminus R_2 \setminus R_3 \setminus R_4$. This is convex.

And so on. It turns out thatR_1, R_3 and R_5 are convex but R_2 and R_4 not. This is readily seen on graph of these regions. Figure 4.1 illustrates theabove regions, which are identified by the value taken by $Q(x,\xi)$.

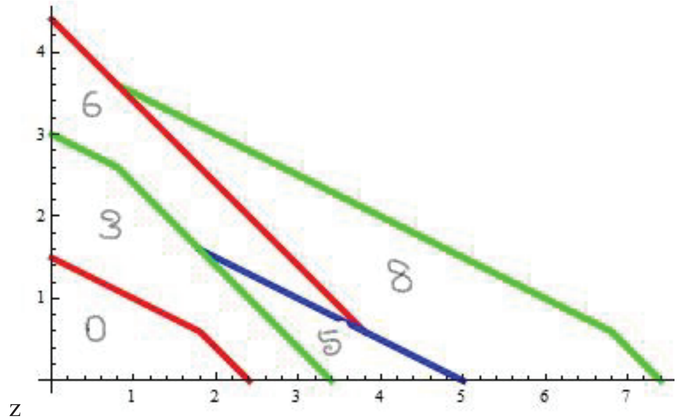

Fig. 3.1. Value of the second-stage solution in example 2 in the x-space.

Now we will look at the same description in the χ *space*

$$\psi(\chi,\xi) = \min 5y_1 + 3y_2$$

Subject to $2y_1 + 3y_2 \geq -3 + \chi_1,\ 4y_1 + y_2 \geq -2.4 + \chi_2, y_1, y_2 \geq 0$, integer.

with $\chi_1 = x_1 + 2x_2$ and $\chi_2 = x_1 + x_2$.

The corresponding regions

Table 3.7: Feasible region and convexity

y	$\psi(\chi,\xi)$	Region R	Convexity
$y = (0,0)^T$	0	$R_1 = \{\chi \mid \chi_1 \leq 3,\ \chi_2 \leq 2.4\}.$	Convex
$y = (1,0)^T$	3	$R_2 = \{\chi \mid \chi_1 \leq 3,\ \chi_2 \leq 2.4\} \backslash R_1$.	Convex
$y = (0,1)^T$	5	$R_3 = \{\chi \mid \chi_1 \leq 5,\ \chi_2 \leq 6.4\} \backslash R_1 \backslash R_2$	Convex
$y = (0,2)^T$	6	$R_4 = \{\chi \mid \chi_1 \leq 9,\ \chi_2 \leq 4.4\} \backslash R_1 \backslash R_2 \backslash R_3.$	Non-convex
$y = (1,1)^T$	8	$R_5 = \{\chi \mid \chi_1 \leq 8,\ \chi_2 \leq 7.4\} \backslash R_1 \backslash R_2 \backslash R_3 \backslash R_4.$	Non-convex

Figure 3.2 shows the regions in theχ space, each region being identified by the value of t$\psi(x)$. Here R_4 *and* R_5 are non-convex.

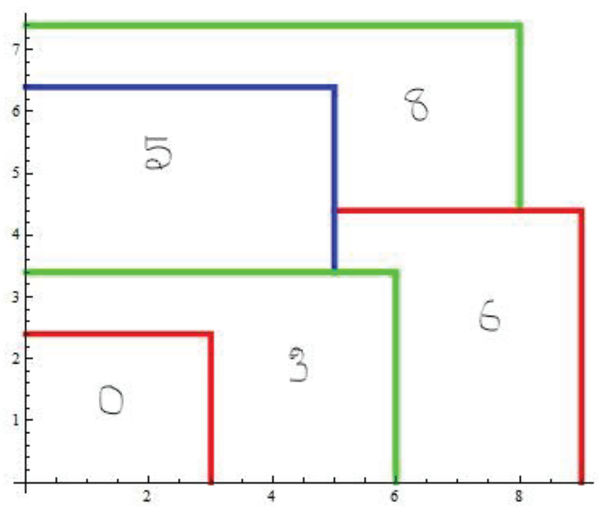

Fig. 3.2. Value of the second-stage solution in example 2 in the tender-space.

In this Section, we have given the basic concept of stochastic integer programs, LP-Relaxation, cases of stochastic integer programs and applications of first-stage binary variables, second-stage integer variables. Illustrative examples solved using AMPL demonstrate the power of SIP in handling binary and integer variables under uncertainty. Mathematica is employed to visualize feasible regions and solution spaces, supporting analytical insights.

IV. Results and Discussion

The worked examples highlight the differences between deterministic and stochastic models. In Example 1, the use of LP-relaxation provided an upper bound on the objective value, while probabilistic variations influencedthe lower bound and optimality cuts. Example 2 demonstrated how second-stage integer variables yield non-convex regions, emphasizing the complexity of SIP compared to standard SP.These findings illustrate that SIP provides more realistic solutions for decision problems under uncertainty. However, computational complexity increases with problem size, necessitating advanced algorithms and efficient software tools.

V. Conclusion and Future Work

This paper provided an overview of stochastic integer programming and its applications. SIP extends the capabilities of stochastic programming by incorporating integer constraints, making it suitable for discrete decision problems under uncertainty. Through examples, we demonstrated the applicability of SIP using AMPL and Mathematica. Future research may focus on scalable algorithms for large-scale SIPs, hybrid approaches combining decomposition and heuristics, and expanding applications to emerging areas such as renewable energy planning and data-driven optimization.

References

1. Ahmed, S. 2019. Two-stage stochastic integer programming: A brief introduction, In: Cochran et al. (eds.) Wiley Encyclopedia of Operations Research and Management Science.
2. Brige, J.R., and F. Louveaux, 1997 Introduction to Stochastic Programming. Springer-Verlag.
3. Birge, J. R. 1985, Decomposition and partitioning methods for multistage stochastic linear programs, Operations Research, **33,** 989–1007.
4. Carøe, C. C., & R. Schultz, 1999. Dual decomposition in stochastic integer programming. Operations Research Letters, **24(1)**, 37–45.
5. CarHea C. C., & R., 1999 Schultzb Dual decomposition in Stochastic Integer Programming. Operations Research Letters, **24**, 37–45.
6. Dantzig, G. B. 1955. Linear programming under uncertainty. Management Science, **1(3-4)**, 197–206.
7. Dantzig, G.B. & A. Mandan sky 1961, On the Solution of Two-stage Linear Programming Under Uncertainty", In I. J. Neyman, Editor, PProc, 4th Berkeley Symp, Math, Stat, and, 165-176, Berkeley.
8. Defourny, B. D. Ernst & L. Wehenkel 2013, Multi-stage stochastic Programming:A Scenario Tree Based Approach to Planning Under Uncertainty", INFORMS Journal on Computing, .**25**, **3**. 488-501.
9. Haneveld, W. K., & M. H Vlerk, . 1999. Stochastic integer programming: General models and algorithms. Annals of Operations Research, **85**, 39–57.
10. Hansman, R., T. Lohmann & TuBraunchweig, 1999 Stochastic programming Using Algebraic Modelling Languages". Northwestern University, Evanston, IL.
11. Hart S.M., & Y.-H., Liu 1995. The Application of Integer Linear Programming to the Implementation ofa Graphical User Interface: A New Rectangular Packing Problem". Appl. Math. Modelling. **19**, April.
12. Grassmann, H. I., & A. M. Ireland 1995 Scenario Formulationin an AlgebraicModelling Languages", Annals of Operations Research, .**59,** :45-75.
13. Laporte G.& F. V. Louveaux .,1993 The Integer L-shaped Method for Stochastic Integer Programs with Complete Recourse", Operations Research Letters **13** 133-142.
14. Mayer, J., & P . Kall, S, 2005 Stochastic. Linear Programming. Springer, USA.
15. Mishra, N., & V. D. Miguel, What Multi-Stage Stochastic Programming can do for Network Revenue Management." London Business School, 6 Sussex Place, Regent's Park, London, United Kingdom NW1, USA.
16. Ravindran, A., D. T. 1987 Philips & J. J. Solberg, Operation Research: Principles and Practices, Wiley & Sons, NY.
17. Richard D. Woller, 1991 Two-Stage Linear Programming Under Uncertainty with 0-1 Integer FirstStage Variables, California State University, Long Beach, CA, U.S.A.
18. Schultz R., Stougie L., & M. H., Vlerk, 1996.. Two-Stage Stochastic Integer Programming: A Survey", Statistical Norrlandic . **50**. **3**,. 404416.
19. Van Slyke, R. M., & R. J.-B.Wets,.1969. L-shaped linear programs with applications to optimal control and stochastic programming. SIAM Journal on Applied Mathematics, **17(4),** 638–663.