Stochastic Programming: Theory and Algorithms

Sinan Sabri

LCSEE
West Virginia University
Morgantown, WV USA

April 21, 2015

Outline

Outline

Outline

Outline

Outline

Outline

Introduction

- Stochastic programming assumes that the uncertain parameters are random variables with known probability distributions.

- Stochastic programming assumes that the uncertain parameters are random variables with known probability distributions.
- This information is then used to transform the stochastic program into a so-called deterministic equivalent which might be:

- Stochastic programming assumes that the uncertain parameters are random variables with known probability distributions.
- This information is then used to transform the stochastic program into a so-called deterministic equivalent which might be:
  - a linear program,

- Stochastic programming assumes that the uncertain parameters are random variables with known probability distributions.
- This information is then used to transform the stochastic program into a so-called deterministic equivalent which might be:
  - a linear program,
  - a nonlinear program, or

- Stochastic programming assumes that the uncertain parameters are random variables with known probability distributions.
- This information is then used to transform the stochastic program into a so-called deterministic equivalent which might be:
  - a linear program,
  - a nonlinear program, or
  - an integer program

- The uncertainty is described by:

- The uncertainty is described by:
  - a certain sample space Ω ,

- The uncertainty is described by:
    - a certain sample space Ω , Ω is often a finite set $\{\omega_1, \dots, \omega_S\}$

- The uncertainty is described by:
  - a certain sample space Ω , Ω is often a finite set $\{\omega_1, \ldots, \omega_S\}$
  - σ field of random events,

- The uncertainty is described by:
  - a certain sample space Ω , Ω is often a finite set $\{\omega_1, \ldots, \omega_S\}$
  - σ field of random events, and
  - a probability measure P

- The uncertainty is described by:
  - a certain sample space Ω , Ω is often a finite set $\{\omega_1, \ldots, \omega_S\}$
  - σ field of random events, and
  - a probability measure P
- The corresponding probabilities $p(\omega_k) \geq 0$ satisfy $\sum_{k=1}^{S} p(\omega_k) = 1$

- The uncertainty is described by:
  - a certain sample space Ω , Ω is often a finite set $\{\omega_1, \ldots, \omega_s\}$
  - σ field of random events, and
  - a probability measure P
- The corresponding probabilities $p(\omega_k) \geq 0$ satisfy $\sum_{k=1}^{S} p(\omega_k) = 1$
- For example,

- The uncertainty is described by:
  - a certain sample space Ω , Ω is often a finite set $\{\omega_1, \ldots, \omega_S\}$
  - σ field of random events, and
  - a probability measure P
- The corresponding probabilities $p(\omega_k) \geq 0$ satisfy $\sum_{k=1}^{S} p(\omega_k) = 1$
- For example, to represent the outcomes of flipping a fair coin twice in a row,

- The uncertainty is described by:
  - a certain sample space Ω , Ω is often a finite set $\{\omega_1, \ldots, \omega_s\}$
  - σ field of random events, and
  - a probability measure P
- The corresponding probabilities $p(\omega_k) \geq 0$ satisfy $\sum_{k=1}^{S} p(\omega_k) = 1$
- For example, to represent the outcomes of flipping a fair coin twice in a row, we would use four random events $\Omega = \{HH, HT, TH, TT\}$,

- The uncertainty is described by:
  - a certain sample space Ω , Ω is often a finite set $\{\omega_1, \ldots, \omega_s\}$
  - σ field of random events, and
  - a probability measure P
- The corresponding probabilities $p(\omega_k) \geq 0$ satisfy $\sum_{k=1}^{S} p(\omega_k) = 1$
- For example, to represent the outcomes of flipping a fair coin twice in a row, we would use four random events $\Omega = \{HH, HT, TH, TT\}$, each with probability $1/4$.

- Stochastic programming models can include anticipative and/or adaptive decision variables.

- Stochastic programming models can include anticipative and/or adaptive decision variables.
- Anticipative variables correspond to those decisions that must be made here-and-now and cannot depend on the future observations/partial realizations of the random parameters.

- Stochastic programming models can include anticipative and/or adaptive decision variables.
- Anticipative variables correspond to those decisions that must be made here-and-now and cannot depend on the future observations/partial realizations of the random parameters.
- Adaptive variables correspond to wait-and-see decisions that can be made after some (or, sometimes all) of the random parameters are observed.

- Stochastic programming models can include anticipative and/or adaptive decision variables.
- Anticipative variables correspond to those decisions that must be made here-and-now and cannot depend on the future observations/partial realizations of the random parameters.
- Adaptive variables correspond to wait-and-see decisions that can be made after some (or, sometimes all) of the random parameters are observed.
- Stochastic programming models that include both anticipative and adaptive variables are called recourse models.

- In investment planning, each new trading opportunity represents a new decision to be made.

- In investment planning, each new trading opportunity represents a new decision to be made.
- Therefore, trading dates where investment portfolios can be rebalanced become natural choices for decision stages.

- In investment planning, each new trading opportunity represents a new decision to be made.
- Therefore, trading dates where investment portfolios can be rebalanced become natural choices for decision stages.
- These problems can be formulated conveniently as multi-stage stochastic programming problems with recourse.

Probabilistic Constraints

Exercise 1:

- Suppose that we have two six-sided dice.

Exercise 1:

- Suppose that we have two six-sided dice.
  - Dice one gives a result a_1
  - Dice two gives a result a_2

Exercise 1:

- Suppose that we have two six-sided dice.
  - Dice one gives a result a_1
  - Dice two gives a result a_2
- Assume the dice are fair, we have discrete probability distributions for a_1 and a_2 as:

Exercise 1:

- Suppose that we have two six-sided dice.
  - Dice one gives a result a_1
  - Dice two gives a result a_2
- Assume the dice are fair, we have discrete probability distributions for a_1 and a_2 as:
  - $a_1 = i \qquad (i = 1, \ldots, 6)$ with probability $1/6$
  - $a_2 = j \qquad (j = 1, \ldots, 6)$ with probability $1/6$

Exercise 1:

- Suppose that we have two six-sided dice.
  - Dice one gives a result a_1
  - Dice two gives a result a_2
- Assume the dice are fair, we have discrete probability distributions for a_1 and a_2 as:
  - $a_1 = i \qquad (i = 1, \ldots, 6)$ with probability $1/6$
  - $a_2 = j \qquad (j = 1, \ldots, 6)$ with probability $1/6$
- Consider a simple LP with two variables and one constraint:

Exercise 1:

- Suppose that we have two six-sided dice.
  - Dice one gives a result a_1
  - Dice two gives a result a_2
- Assume the dice are fair, we have discrete probability distributions for a_1 and a_2 as:
  - $a_1 = i \qquad (i = 1, \ldots, 6)$ with probability $1/6$
  - $a_2 = j \qquad (j = 1, \ldots, 6)$ with probability $1/6$
- Consider a simple LP with two variables and one constraint:

$$\min 5x + 6y$$

Exercise 1:

- Suppose that we have two six-sided dice.
  - Dice one gives a result a_1
  - Dice two gives a result a_2
- Assume the dice are fair, we have discrete probability distributions for a_1 and a_2 as:
  - $a_1 = i \qquad (i = 1, \ldots, 6)$ with probability $1/6$
  - $a_2 = j \qquad (j = 1, \ldots, 6)$ with probability $1/6$
- Consider a simple LP with two variables and one constraint:

$$\min 5x + 6y \qquad \textit{subject to:}$$

Exercise 1:

- Suppose that we have two six-sided dice.
  - Dice one gives a result a_1
  - Dice two gives a result a_2
- Assume the dice are fair, we have discrete probability distributions for a_1 and a_2 as:
  - $a_1 = i \qquad (i = 1, \ldots, 6)$ with probability $1/6$
  - $a_2 = j \qquad (j = 1, \ldots, 6)$ with probability $1/6$
- Consider a simple LP with two variables and one constraint:

$$\min 5x + 6y \qquad \textit{subject to:}$$

$$a_1 x + a_2 y \geq 3$$

Exercise 1:

- Suppose that we have two six-sided dice.
  - Dice one gives a result a_1
  - Dice two gives a result a_2
- Assume the dice are fair, we have discrete probability distributions for a_1 and a_2 as:
  - $a_1 = i \qquad (i = 1, \ldots, 6)$ with probability $1/6$
  - $a_2 = j \qquad (j = 1, \ldots, 6)$ with probability $1/6$
- Consider a simple LP with two variables and one constraint:

$$\min 5x + 6y \qquad \textit{subject to:}$$

$$a_1 x + a_2 y \geq 3$$

$$x, y \geq 0$$

Exercise 1

- If a_1 and a_2 were known numbers we will have LP problem,

Exercise 1

- If a_1 and a_2 were known numbers we will have LP problem,**but they are not.**

Exercise 1

- If a_1 and a_2 were known numbers we will have LP problem,**but they are not.**
- One interpretation could be that we wish the constraint

Exercise 1

- If a_1 and a_2 were known numbers we will have LP problem,**but they are not.**
- One interpretation could be that we wish the constraint $a_1 x + a_2 y \geq 3$

Exercise 1

- If a_1 and a_2 were known numbers we will have LP problem,**but they are not.**
- One interpretation could be that we wish the constraint $a_1x + a_2y \geq 3$ to hold for all possible values of a_1 and a_2.

Exercise 1

- If a_1 and a_2 were known numbers we will have LP problem,**but they are not.**
- One interpretation could be that we wish the constraint $a_1x + a_2y \geq 3$ to hold for all possible values of a_1 and a_2.
- Then we simply have a deterministic LP with two variables and 36 constraints:

Exercise 1

- If a_1 and a_2 were known numbers we will have LP problem,**but they are not.**
- One interpretation could be that we wish the constraint $a_1x + a_2y \geq 3$ to hold for all possible values of a_1 and a_2.
- Then we simply have a deterministic LP with two variables and 36 constraints:

$$\min 5 \cdot x + 6 \cdot y$$

Exercise 1

- If a_1 and a_2 were known numbers we will have LP problem,**but they are not.**
- One interpretation could be that we wish the constraint $a_1 x + a_2 y \geq 3$ to hold for all possible values of a_1 and a_2.
- Then we simply have a deterministic LP with two variables and 36 constraints:

$$\min 5 \cdot x + 6 \cdot y \qquad \textit{subject to:}$$

Exercise 1

- If a_1 and a_2 were known numbers we will have LP problem,**but they are not.**
- One interpretation could be that we wish the constraint $a_1x + a_2y \geq 3$ to hold for all possible values of a_1 and a_2.
- Then we simply have a deterministic LP with two variables and 36 constraints:

$$\min 5 \cdot x + 6 \cdot y \qquad \textit{subject to:}$$

$$i \cdot x + j \cdot y \geq 3$$

Exercise 1

- If a_1 and a_2 were known numbers we will have LP problem,**but they are not.**
- One interpretation could be that we wish the constraint $a_1 x + a_2 y \geq 3$ to hold for all possible values of a_1 and a_2.
- Then we simply have a deterministic LP with two variables and 36 constraints:

$$\min 5 \cdot x + 6 \cdot y \qquad \textit{subject to:}$$

$$i \cdot x + j \cdot y \geq 3$$

$$i, j = 1, ..., 6$$

Exercise 1

- If a_1 and a_2 were known numbers we will have LP problem,**but they are not.**
- One interpretation could be that we wish the constraint $a_1x + a_2y \geq 3$ to hold for all possible values of a_1 and a_2.
- Then we simply have a deterministic LP with two variables and 36 constraints:

$$\min 5 \cdot x + 6 \cdot y \qquad \textit{subject to:}$$
$$i \cdot x + j \cdot y \geq 3$$
$$i, j = 1, ..., 6$$
$$x, y \geq 0$$

Exercise 1

- Suppose now that instead of insisting that the constraint

Exercise 1

- Suppose now that instead of insisting that the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$

Exercise 1

- Suppose now that instead of insisting that the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ holds for all possible values of a_1 and a_2.

Exercise 1

- Suppose now that instead of insisting that the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ holds for all possible values of a_1 and a_2.
- We insist that it holds only with a specified probability $1 - \alpha$ (where $0 < \alpha < 1$).

Exercise 1

- Suppose now that instead of insisting that the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ holds for all possible values of a_1 and a_2.
- We insist that it holds only with a specified probability $1 - \alpha$ (where $0 < \alpha < 1$).
- For example, if $\alpha = 0.05$ would mean that we want the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ to hold with probability 0.95.

Exercise 1

- Suppose now that instead of insisting that the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ holds for all possible values of a_1 and a_2.
- We insist that it holds only with a specified probability $1 - \alpha$ (where $0 < \alpha < 1$).
- For example, if $\alpha = 0.05$ would mean that we want the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ to hold with probability 0.95.
- Here, the constraint need not always be true now,

Exercise 1

- Suppose now that instead of insisting that the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ holds for all possible values of a_1 and a_2.
- We insist that it holds only with a specified probability $1 - \alpha$ (where $0 < \alpha < 1$).
- For example, if $\alpha = 0.05$ would mean that we want the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ to hold with probability 0.95.
- Here, the constraint need not always be true now, rather it need only be true 95% of the time.

Exercise 1

- Suppose now that instead of insisting that the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ holds for all possible values of a_1 and a_2.
- We insist that it holds only with a specified probability $1 - \alpha$ (where $0 < \alpha < 1$).
- For example, if $\alpha = 0.05$ would mean that we want the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ to hold with probability 0.95.
- Here, the constraint need not always be true now, rather it need only be true 95% of the time.
- Hence the problem is:

$$\min 5 \cdot x + 6 \cdot y \qquad \textit{subject to:}$$

Exercise 1

- Suppose now that instead of insisting that the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ holds for all possible values of a_1 and a_2.
- We insist that it holds only with a specified probability $1 - \alpha$ (where $0 < \alpha < 1$).
- For example, if $\alpha = 0.05$ would mean that we want the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ to hold with probability 0.95.
- Here, the constraint need not always be true now, rather it need only be true 95% of the time.
- Hence the problem is:

$$\min 5 \cdot x + 6 \cdot y \qquad \textit{subject to:}$$
$$Prob(a_1 \cdot x + a_2 \cdot y \geq 3) \geq 1 - \alpha$$

Exercise 1

- Suppose now that instead of insisting that the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ holds for all possible values of a_1 and a_2.
- We insist that it holds only with a specified probability $1 - \alpha$ (where $0 < \alpha < 1$).
- For example, if $\alpha = 0.05$ would mean that we want the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ to hold with probability 0.95.
- Here, the constraint need not always be true now, rather it need only be true 95% of the time.
- Hence the problem is:

$$\min 5 \cdot x + 6 \cdot y \qquad \textit{subject to:}$$
$$Prob(a_1 \cdot x + a_2 \cdot y \geq 3) \geq 1 - \alpha$$
$$x, y \geq 0$$

Exercise 1

To summarize:

- a_1 and a_2 are unknown here,

Exercise 1

To summarize:

- a_1 and a_2 are unknown here, we merely have probability distribution information for them.

Exercise 1

To summarize:

- a_1 and a_2 are unknown here, we merely have probability distribution information for them.
- We are required to choose values for x and y such that:

Exercise 1

To summarize:

- a_1 and a_2 are unknown here, we merely have probability distribution information for them.
- We are required to choose values for x and y such that:
  - The objective function is minimized

Exercise 1

To summarize:

- a_1 and a_2 are unknown here, we merely have probability distribution information for them.
- We are required to choose values for x and y such that:
  - The objective function is minimized and
  - The probability that the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ is satisfied is at least $1 - \alpha$.

Exercise 1

To summarize:

- a_1 and a_2 are unknown here, we merely have probability distribution information for them.
- We are required to choose values for x and y such that:
  - The objective function is minimized and
  - The probability that the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ is satisfied is at least $1 - \alpha$.

Now:

- For each pair of values (a_1, a_2) we have an associated joint probability $(1/36)$

Exercise 1

To summarize:

- a_1 and a_2 are unknown here, we merely have probability distribution information for them.
- We are required to choose values for x and y such that:
  - The objective function is minimized and
  - The probability that the constraint $a_1 \cdot x + a_2 \cdot y \geq 3$ is satisfied is at least $1 - \alpha$.

Now:

- For each pair of values (a_1, a_2) we have an associated joint probability $(1/36)$
- Given values for $x \geq 0$ and $y \geq 0$ we can easily check whether the constraint is true with probability $1 - \alpha$.

Exercise 1

- Let us enumerate possibilities for $x = 0$, $y = 1$ and $\alpha = 0.05$

Exercise 1

- Let us enumerate possibilities for $x = 0$, $y = 1$ and $\alpha = 0.05$

a_1	a_2	Is $a_1 \cdot 0 + a_2 \cdot 1 \geq 3$?	Probability
1	1	No	$1/36$
1	2	No	$1/36$

etc

Exercise 1

- Let us enumerate possibilities for $x = 0$, $y = 1$ and $\alpha = 0.05$

a_1	a_2	Is $a_1 \cdot 0 + a_2 \cdot 1 \geq 3$?	Probability
1	1	No	$1/36$
1	2	No	$1/36$

etc

- In this case, $x = 0$ and $y = 1$ is not a feasible solution, since we already have a probability of 2/36 = 0.0555 that the constraint is infeasible.
  It is impossible for the constraint to be feasible with probability 0.95 (since 1-0.0555 = 0.9445).
- This particular problem (because it contains just two variables) can be easily solved by a simple numeric search procedure.

Exercise 1

- Let us enumerate possibilities for $x = 0$, $y = 1$ and $\alpha = 0.05$

a_1	a_2	Is $a_1 \cdot 0 + a_2 \cdot 1 \geq 3$?	Probability
1	1	No	$1/36$
1	2	No	$1/36$

etc

- In this case, $x = 0$ and $y = 1$ is not a feasible solution, since we already have a probability of 2/36 = 0.0555 that the constraint is infeasible.
  It is impossible for the constraint to be feasible with probability 0.95 (since 1-0.0555 = 0.9445).
- This particular problem (because it contains just two variables) can be easily solved by a simple numeric search procedure.
- For example, for $\alpha = 0.01$ the solution is $x = 3$, $y = 0$ and for $\alpha = 0.05$ the solution is $x = 1$, $y = 1$

Recourse Problems

Two-Stage Problems with Recourse

Two-Stage Problems with Recourse

- The generic form of a two-stage stochastic linear program with recourse:

Two-Stage Problems with Recourse

- The generic form of a two-stage stochastic linear program with recourse:

$$\max a^T x + E[\max_{y(\omega)} c(\omega)^T y(\omega)] \qquad (1)$$

Two-Stage Problems with Recourse

- The generic form of a two-stage stochastic linear program with recourse:

$$\max a^T x + E[\max_{y(\omega)} c(\omega)^T y(\omega)] \qquad (1)$$

$$Ax = b$$

Two-Stage Problems with Recourse

- The generic form of a two-stage stochastic linear program with recourse:

$$\max a^T x + E[\max_{y(\omega)} c(\omega)^T y(\omega)] \qquad (1)$$
$$Ax = b$$
$$B(\omega)x + C(\omega)y(\omega) = d(\omega)$$

Two-Stage Problems with Recourse

- The generic form of a two-stage stochastic linear program with recourse:

$$\max a^T x + E[\max_{y(\omega)} c(\omega)^T y(\omega)] \quad (1)$$
$$Ax = b$$
$$B(\omega)x + C(\omega)y(\omega) = d(\omega)$$
$$x \geq 0,$$

Two-Stage Problems with Recourse

- The generic form of a two-stage stochastic linear program with recourse:

$$\max a^T x + E[\max_{y(\omega)} c(\omega)^T y(\omega)] \qquad (1)$$

$$Ax = b$$

$$B(\omega)x + C(\omega)y(\omega) = d(\omega)$$

$$x \geq 0, \qquad y(\omega) \geq 0$$

Two-Stage Problems with Recourse

- The generic form of a two-stage stochastic linear program with recourse:

$$\max a^T x + E[\max_{y(\omega)} c(\omega)^T y(\omega)] \qquad (1)$$
$$Ax = b$$
$$B(\omega)x + C(\omega)y(\omega) = d(\omega)$$
$$x \geq 0, \qquad y(\omega) \geq 0$$

- The first-stage:

Two-Stage Problems with Recourse

- The generic form of a two-stage stochastic linear program with recourse:

$$\max a^T x + E[\max_{y(\omega)} c(\omega)^T y(\omega)] \qquad (1)$$
$$Ax = b$$
$$B(\omega)x + C(\omega)y(\omega) = d(\omega)$$
$$x \geq 0, \qquad y(\omega) \geq 0$$

- The first-stage:
  - Decisions are represented by vector x.

Two-Stage Problems with Recourse

- The generic form of a two-stage stochastic linear program with recourse:

$$\max a^T x + E[\max_{y(\omega)} c(\omega)^T y(\omega)] \qquad (1)$$
$$Ax = b$$
$$B(\omega)x + C(\omega)y(\omega) = d(\omega)$$
$$x \geq 0, \qquad y(\omega) \geq 0$$

- The first-stage:
  - Decisions are represented by vector x. These decisions are made before the random event ω is observed.

Two-Stage Problems with Recourse

- The generic form of a two-stage stochastic linear program with recourse:

$$\max a^T x + E[\max_{y(\omega)} c(\omega)^T y(\omega)] \qquad (1)$$
$$Ax = b$$
$$B(\omega)x + C(\omega)y(\omega) = d(\omega)$$
$$x \geq 0, \qquad y(\omega) \geq 0$$

- The first-stage:
  - Decisions are represented by vector x. These decisions are made before the random event ω is observed.
- The second-stage:

Two-Stage Problems with Recourse

- The generic form of a two-stage stochastic linear program with recourse:

$$\max a^T x + E[\max_{y(\omega)} c(\omega)^T y(\omega)] \qquad (1)$$
$$Ax = b$$
$$B(\omega)x + C(\omega)y(\omega) = d(\omega)$$
$$x \geq 0, \qquad y(\omega) \geq 0$$

- The first-stage:
  - Decisions are represented by vector x. These decisions are made before the random event ω is observed.
- The second-stage:
  - Decisions are represented by vector $y(\omega)$.

Two-Stage Problems with Recourse

- The generic form of a two-stage stochastic linear program with recourse:

$$\max a^T x + E[\max_{y(\omega)} c(\omega)^T y(\omega)] \quad (1)$$
$$Ax = b$$
$$B(\omega)x + C(\omega)y(\omega) = d(\omega)$$
$$x \geq 0, \qquad y(\omega) \geq 0$$

- The first-stage:
  - Decisions are represented by vector x. These decisions are made before the random event ω is observed.
- The second-stage:
  - Decisions are represented by vector $y(\omega)$. These decisions are made after the random event ω has been observed,

Two-Stage Problems with Recourse

- The generic form of a two-stage stochastic linear program with recourse:

$$\max a^T x + E[\max_{y(\omega)} c(\omega)^T y(\omega)] \qquad (1)$$

$$Ax = b$$

$$B(\omega)x + C(\omega)y(\omega) = d(\omega)$$

$$x \geq 0, \qquad y(\omega) \geq 0$$

- The first-stage:
  - Decisions are represented by vector x. These decisions are made before the random event ω is observed.
- The second-stage:
  - Decisions are represented by vector $y(\omega)$. These decisions are made after the random event ω has been observed, and therefore the vector y is a function of ω.

Two-Stage Problems with Recourse

- A and b define deterministic constraints on the first-stage decisions x

Two-Stage Problems with Recourse

- A and b define deterministic constraints on the first-stage decisions x
- $B(\omega)$, $C(\omega)$, and $d(\omega)$ define stochastic constraints linking the recourse decisions $y(\omega)$ to the first-stage decisions x.

Two-Stage Problems with Recourse

- A and b define deterministic constraints on the first-stage decisions x
- $B(\omega)$, $C(\omega)$, and $d(\omega)$ define stochastic constraints linking the recourse decisions $y(\omega)$ to the first-stage decisions x.
- The objective function contains a deterministic term $a^T x$ and the expectation of the second-stage objective $C(\omega)^T y(\omega)$ taken over all realizations of the random event ω.

Two-Stage Problems with Recourse

- A and b define deterministic constraints on the first-stage decisions x
- $B(\omega)$, $C(\omega)$, and $d(\omega)$ define stochastic constraints linking the recourse decisions $y(\omega)$ to the first-stage decisions x.
- The objective function contains a deterministic term $a^T x$ and the expectation of the second-stage objective $C(\omega)^T y(\omega)$ taken over all realizations of the random event ω.
- Notice that the first-stage decisions will not necessarily satisfy the linking constraints $B(\omega)x + C(\omega)y(\omega) = d(\omega)$,

Two-Stage Problems with Recourse

- A and b define deterministic constraints on the first-stage decisions x
- $B(\omega)$, $C(\omega)$, and $d(\omega)$ define stochastic constraints linking the recourse decisions $y(\omega)$ to the first-stage decisions x.
- The objective function contains a deterministic term $a^T x$ and the expectation of the second-stage objective $C(\omega)^T y(\omega)$ taken over all realizations of the random event ω.
- Notice that the first-stage decisions will not necessarily satisfy the linking constraints $B(\omega)x + C(\omega)y(\omega) = d(\omega)$,
- Therefore, recourse allows one to make sure that the initial decisions can be ”corrected” with respect to this second set of feasibility equations.

Two-Stage Problems with Recourse

- Problem (1) can be represented in an alternative manner by considering the second-stage or recourse problem that is defined as follows,

Two-Stage Problems with Recourse

- Problem (1) can be represented in an alternative manner by considering the second-stage or recourse problem that is defined as follows, given x, the first-stage decisions:

Two-Stage Problems with Recourse

- Problem (1) can be represented in an alternative manner by considering the second-stage or recourse problem that is defined as follows, given x, the first-stage decisions:

$$f(x, \omega) = \max c(\omega)^T y(\omega) \qquad (2)$$

Two-Stage Problems with Recourse

- Problem (1) can be represented in an alternative manner by considering the second-stage or recourse problem that is defined as follows, given x, the first-stage decisions:

$$f(x,\omega) = \max c(\omega)^T y(\omega) \qquad (2)$$
$$C(\omega)y(\omega) = d(\omega) - B(\omega)x$$

Two-Stage Problems with Recourse

- Problem (1) can be represented in an alternative manner by considering the second-stage or recourse problem that is defined as follows, given x, the first-stage decisions:

$$\begin{aligned} f(x, \omega) &= \max c(\omega)^T y(\omega) \qquad (2) \\ C(\omega)y(\omega) &= d(\omega) - B(\omega)x \\ y(\omega) &\geq 0 \end{aligned}$$

Two-Stage Problems with Recourse

- Let $f(x) = E[f(x, \omega)]$ denote the expected value of this optimum.

Two-Stage Problems with Recourse

- Let $f(x) = E[f(x,\omega)]$ denote the expected value of this optimum. If the function $f(x)$ is available,

Two-Stage Problems with Recourse

- Let $f(x) = E[f(x, \omega)]$ denote the expected value of this optimum. If the function $f(x)$ is available, the two-stage stochastic linear program (1) reduces to a deterministic nonlinear program:

Two-Stage Problems with Recourse

- Let $f(x) = E[f(x, \omega)]$ denote the expected value of this optimum. If the function $f(x)$ is available, the two-stage stochastic linear program (1) reduces to a deterministic nonlinear program:

$$\max a^T x + f(x) \qquad (3)$$

Two-Stage Problems with Recourse

- Let $f(x) = E[f(x, \omega)]$ denote the expected value of this optimum.
  If the function $f(x)$ is available, the two-stage stochastic linear program (1) reduces to a deterministic nonlinear program:

$$\max a^T x + f(x) \qquad (3)$$
$$Ax = b,$$

Two-Stage Problems with Recourse

- Let $f(x) = E[f(x, \omega)]$ denote the expected value of this optimum. If the function $f(x)$ is available, the two-stage stochastic linear program (1) reduces to a deterministic nonlinear program:

$$\max a^T x + f(x) \qquad (3)$$
$$Ax = b, \qquad x \geq 0$$

Two-Stage Problems with Recourse

- Assume that $\Omega = \{\omega_1, ..., \omega_S\}$

Two-Stage Problems with Recourse

- Assume that $\Omega = \{\omega_1, ..., \omega_S\}$ and let $p = (p_1, ..., p_S)$ denote the probability distribution on this sample space.

Two-Stage Problems with Recourse

- Assume that $\Omega = \{\omega_1, ..., \omega_S\}$ and let $p = (p_1, ..., p_S)$ denote the probability distribution on this sample space.
- The S possibilities ω_k, for $k = 1, ..., S$ are also called SCENARIOS .

Two-Stage Problems with Recourse

- Assume that $\Omega = \{\omega_1, ..., \omega_S\}$ and let $p = (p_1, ..., p_S)$ denote the probability distribution on this sample space.
- The S possibilities ω_k, for $k = 1, ..., S$ are also called SCENARIOS .
- The expectation of the second-stage objective becomes:

Two-Stage Problems with Recourse

- Assume that $\Omega = \{\omega_1, ..., \omega_S\}$ and let $p = (p_1, ..., p_S)$ denote the probability distribution on this sample space.
- The S possibilities ω_k, for $k = 1, ..., S$ are also called SCENARIOS .
- The expectation of the second-stage objective becomes:

$$E[\max_{y(\omega)} c(\omega)^T y(\omega)] = \sum_{k=1}^{S} p_k \max_{y(\omega_k)} c(\omega_k)^T y(\omega_k)$$

Two-Stage Problems with Recourse

- For brevity, we write c_k instead of $c(\omega_k)$, etc.

Two-Stage Problems with Recourse

- For brevity, we write c_k instead of $c(\omega_k)$, etc.
- Under this scenario approach, the two-stage stochastic linear programming problem (1) takes the following form:

Two-Stage Problems with Recourse

- For brevity, we write c_k instead of $c(\omega_k)$, etc.
- Under this scenario approach, the two-stage stochastic linear programming problem (1) takes the following form:

$$\max_x a^T x + \sum_{k=1}^{S} p_k \max_{y_k} c_k^T y_k \qquad (4)$$

Two-Stage Problems with Recourse

- For brevity, we write c_k instead of $c(\omega_k)$, etc.
- Under this scenario approach, the two-stage stochastic linear programming problem (1) takes the following form:

$$\max_x a^T x + \sum_{k=1}^{S} p_k \max_{y_k} c_k^T y_k \qquad (4)$$
$$Ax = b$$

Two-Stage Problems with Recourse

- For brevity, we write c_k instead of $c(\omega_k)$, etc.
- Under this scenario approach, the two-stage stochastic linear programming problem (1) takes the following form:

$$\max_{x} a^T x + \sum_{k=1}^{S} p_k \max_{y_k} c_k^T y_k \qquad (4)$$

$$Ax = b$$

$$B_k x + C_k y_k = d_k \qquad \mathit{for} \;\; k = 1, ..., S$$

Two-Stage Problems with Recourse

- For brevity, we write c_k instead of $c(\omega_k)$, etc.
- Under this scenario approach, the two-stage stochastic linear programming problem (1) takes the following form:

$$\max_x a^T x + \sum_{k=1}^{S} p_k \max_{y_k} c_k^T y_k \qquad (4)$$

$$Ax = b$$

$$B_k x + C_k y_k = d_k \qquad \textit{for} \;\; k = 1, ..., S$$

$$x \geq 0,$$

Two-Stage Problems with Recourse

- For brevity, we write c_k instead of $c(\omega_k)$, etc.
- Under this scenario approach, the two-stage stochastic linear programming problem (1) takes the following form:

$$\begin{aligned} &\max_x a^T x + \sum_{k=1}^{S} p_k \max_{y_k} c_k^T y_k \qquad (4) \\ &Ax = b \\ &B_k x + C_k y_k = d_k \qquad \textit{for} \;\; k = 1, ..., S \\ &x \geq 0, \qquad y_k \geq 0 \qquad \textit{for} \;\; k = 1, ..., S \end{aligned}$$

Two-Stage Problems with Recourse

- There is a different second stage decision vector y_k for each scenario k.

Two-Stage Problems with Recourse

- There is a different second stage decision vector y_k for each scenario k.
- The maximum in the objective is achieved by optimizing over all variables x and y_k simultaneously.

Two-Stage Problems with Recourse

- There is a different second stage decision vector y_k for each scenario k.
- The maximum in the objective is achieved by optimizing over all variables x and y_k simultaneously.

  Therefore, this optimization problem is:

Two-Stage Problems with Recourse

- There is a different second stage decision vector y_k for each scenario k.
- The maximum in the objective is achieved by optimizing over all variables x and y_k simultaneously.
  Therefore, this optimization problem is:

$$\max_{x, y_1, \ldots, y_S} a^T x + p_1 c_1^T y_1 + \cdots + p_S c_S^T y_S \qquad (5)$$

Two-Stage Problems with Recourse

- There is a different second stage decision vector y_k for each scenario k.
- The maximum in the objective is achieved by optimizing over all variables x and y_k simultaneously.
  Therefore, this optimization problem is:

$$\max_{x, y_1, \ldots, y_S} a^T x + p_1 c_1^T y_1 + \cdots + p_S c_S^T y_S \qquad (5)$$

$$Ax = b$$

Two-Stage Problems with Recourse

- There is a different second stage decision vector y_k for each scenario k.
- The maximum in the objective is achieved by optimizing over all variables x and y_k simultaneously.
  Therefore, this optimization problem is:

$$\max_{x, y_1, \dots, y_S} a^T x + p_1 c_1^T y_1 + \cdots + p_S c_S^T y_S \qquad (5)$$

$$Ax = b$$

$$\begin{array}{ccccccc} B_1 x & + & C_1 y_1 & & & = & d_1 \\ \vdots & & & \ddots & & \vdots & \vdots \\ B_S x & & & + & C_S y_S & = & d_S \end{array}$$

Two-Stage Problems with Recourse

- There is a different second stage decision vector y_k for each scenario k.
- The maximum in the objective is achieved by optimizing over all variables x and y_k simultaneously.

  Therefore, this optimization problem is:

$$\max_{x, y_1, \dots, y_S} \; a^T x + p_1 c_1^T y_1 + \cdots + p_S c_S^T y_S \qquad (5)$$

$$Ax = b$$

$$\begin{array}{ccccccc} B_1 x & + & C_1 y_1 & & & = & d_1 \\ \vdots & & & \ddots & & \vdots & \vdots \\ B_S x & & & + & C_S y_S & = & d_S \end{array}$$

$$x \geq 0,$$

Two-Stage Problems with Recourse

- There is a different second stage decision vector y_k for each scenario k.
- The maximum in the objective is achieved by optimizing over all variables x and y_k simultaneously.

  Therefore, this optimization problem is:

$$\max_{x, y_1, \ldots, y_S} a^T x + p_1 c_1^T y_1 + \cdots + p_S c_S^T y_S \qquad (5)$$

$$Ax = b$$

$$\begin{array}{ccccccc} B_1 x & + & C_1 y_1 & & & = & d_1 \\ \vdots & & & \ddots & & \vdots & \vdots \\ B_S x & & & + & C_S y_S & = & d_S \end{array}$$

$$x \geq 0, \qquad y_k \geq 0 \qquad for \;\; k = 1, \ldots., S$$

Two-Stage Problems with Recourse - Exercise 2

Exercise 2:

- Suppose that we have to make a decision about the amount of a product X to produce.

Two-Stage Problems with Recourse - Exercise 2

Exercise 2:

- Suppose that we have to make a decision about the amount of a product X to produce.
- Each unit of X that we make costs us $2.

Two-Stage Problems with Recourse - Exercise 2

Exercise 2:

- Suppose that we have to make a decision about the amount of a product X to produce.
- Each unit of X that we make costs us $2.
- X is made to meet demand from customers in the next time period.

Two-Stage Problems with Recourse - Exercise 2

Exercise 2:

- Suppose that we have to make a decision about the amount of a product X to produce.
- Each unit of X that we make costs us $2.
- X is made to meet demand from customers in the next time period.
- However demand is stochastic,

Two-Stage Problems with Recourse - Exercise 2

Exercise 2:

- Suppose that we have to make a decision about the amount of a product X to produce.
- Each unit of X that we make costs us $2.
- X is made to meet demand from customers in the next time period.
- However demand is stochastic, with a discrete probability distribution:

Two-Stage Problems with Recourse - Exercise 2

Exercise 2:

- Suppose that we have to make a decision about the amount of a product X to produce.
- Each unit of X that we make costs us \$2.
- X is made to meet demand from customers in the next time period.
- However demand is stochastic, with a discrete probability distribution:

$$\text{demand} = D_s$$

with probability p_s ($s = 1, ..., S$)

Two-Stage Problems with Recourse - Exercise 2

Exercise 2:

- Suppose that we have to make a decision about the amount of a product X to produce.
- Each unit of X that we make costs us \$2.
- X is made to meet demand from customers in the next time period.
- However demand is stochastic, with a discrete probability distribution:

$$\text{demand} = D_s$$

  with probability p_s $(s = 1, ..., S)$
- S is the possible future demand scenarios.

Two-Stage Problems with Recourse - Exercise 2

- Customer demand must be met.

Two-Stage Problems with Recourse - Exercise 2

- Customer demand must be met.
- We have the flexibility to buy in the product from an external supplier to meet observed customer demand

Two-Stage Problems with Recourse - Exercise 2

- Customer demand must be met.
- We have the flexibility to buy in the product from an external supplier to meet observed customer demand **but this costs us $3 per unit**

Two-Stage Problems with Recourse - Exercise 2

- Customer demand must be met.
- We have the flexibility to buy in the product from an external supplier to meet observed customer demand **but this costs us $3 per unit** (i.e. we have recourse to an additional source of supply if demand exceeds production).

Two-Stage Problems with Recourse - Exercise 2

- Customer demand must be met.
- We have the flexibility to buy in the product from an external supplier to meet observed customer demand **but this costs us $3 per unit** (i.e. we have recourse to an additional source of supply if demand exceeds production).
- How much should we choose to make now before we know what customer demand is?

Two-Stage Problems with Recourse - Exercise 2

- Customer demand must be met.
- We have the flexibility to buy in the product from an external supplier to meet observed customer demand **but this costs us $3 per unit** (i.e. we have recourse to an additional source of supply if demand exceeds production).
- How much should we choose to make now before we know what customer demand is?
- One way to think of this two-stage model is:
  - Action, make a decision (amount to produce)

Two-Stage Problems with Recourse - Exercise 2

- Customer demand must be met.
- We have the flexibility to buy in the product from an external supplier to meet observed customer demand **but this costs us $3 per unit** (i.e. we have recourse to an additional source of supply if demand exceeds production).
- How much should we choose to make now before we know what customer demand is?
- One way to think of this two-stage model is:
  - Action, make a decision (amount to produce)
  - Observation, observe a realization of the stochastic elements (demand that occurs)

Two-Stage Problems with Recourse - Exercise 2

- Customer demand must be met.
- We have the flexibility to buy in the product from an external supplier to meet observed customer demand **but this costs us $3 per unit** (i.e. we have recourse to an additional source of supply if demand exceeds production).
- How much should we choose to make now before we know what customer demand is?
- One way to think of this two-stage model is:
  - Action, make a decision (amount to produce)
  - Observation, observe a realization of the stochastic elements (demand that occurs)
  - Reaction (recourse), further decisions, depending upon the realization observed (extra production to meet demand if necessary)

Two-Stage Problems with Recourse - Exercise 2

- For example, suppose $S = 2$ and $D_1 = 500$, $p_1 = 0.6$; $D_2 = 700$, $p_2 = 0.4$

Two-Stage Problems with Recourse - Exercise 2

- For example, suppose $S = 2$ and $D_1 = 500$, $p_1 = 0.6$; $D_2 = 700$, $p_2 = 0.4$
- We have to decide how much to produce now before demand is known.

Two-Stage Problems with Recourse - Exercise 2

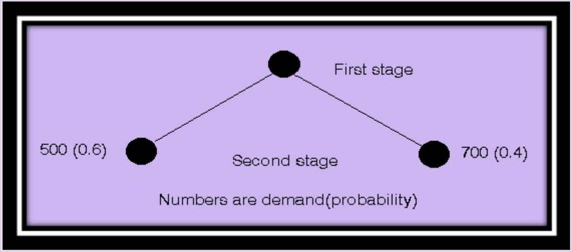
- For example, suppose $S = 2$ and $D_1 = 500$, $p_1 = 0.6$; $D_2 = 700$, $p_2 = 0.4$
- We have to decide how much to produce now before demand is known. If we were to produce 600, then if demand is 500 we are OK,

Two-Stage Problems with Recourse - Exercise 2

- For example, suppose $S = 2$ and $D_1 = 500$, $p_1 = 0.6$; $D_2 = 700$, $p_2 = 0.4$
- We have to decide how much to produce now before demand is known. If we were to produce 600, then if demand is 500 we are OK, if demand is 700 however we need recourse to an extra 100 units to meet it.

Two-Stage Problems with Recourse - Exercise 2

- For example, suppose $S = 2$ and $D_1 = 500$, $p_1 = 0.6$; $D_2 = 700$, $p_2 = 0.4$
- We have to decide how much to produce now before demand is known. If we were to produce 600, then if demand is 500 we are OK, if demand is 700 however we need recourse to an extra 100 units to meet it.



Two-Stage Problems with Recourse - Exercise 2

- Let $x_1 \geq 0$ be the number of units of X to produce now (at the first stage)

Two-Stage Problems with Recourse - Exercise 2

- Let $x_1 \geq 0$ be the number of units of X to produce now (at the first stage)
- Let $y_{2s} \geq 0$ be the number of units of X to buy from the external supplier at the second stage in scenario s when the stochastic realization of the demand is D_s $(s = 1, 2)$.

Two-Stage Problems with Recourse - Exercise 2

- Let $x_1 \geq 0$ be the number of units of X to produce now (at the first stage)
- Let $y_{2s} \geq 0$ be the number of units of X to buy from the external supplier at the second stage in scenario s when the stochastic realization of the demand is D_s $(s = 1, 2)$.
- Then the constraints to ensure that demand is always satisfied are:

Two-Stage Problems with Recourse - Exercise 2

- Let $x_1 \geq 0$ be the number of units of X to produce now (at the first stage)
- Let $y_{2s} \geq 0$ be the number of units of X to buy from the external supplier at the second stage in scenario s when the stochastic realization of the demand is D_s $(s = 1, 2)$.
- Then the constraints to ensure that demand is always satisfied are:

$$x_1 + y_{2s} \geq D_s \qquad s = 1, 2$$

Two-Stage Problems with Recourse - Exercise 2

- The SP is:

Two-Stage Problems with Recourse - Exercise 2

- The SP is:

$$\min 2 \cdot x_1 + \sum_{s=1}^{2} p_s (3 \cdot y_{2s})$$

Two-Stage Problems with Recourse - Exercise 2

- The SP is:

$$\min 2 \cdot x_1 + \sum_{s=1}^{2} p_s(3 \cdot y_{2s})$$
$$x_1 + y_{2s} \geq D_s \qquad s = 1, 2$$

Two-Stage Problems with Recourse - Exercise 2

- The SP is:

$$\min 2 \cdot x_1 + \sum_{s=1}^{2} p_s (3 \cdot y_{2s})$$
$$x_1 + y_{2s} \geq D_s \qquad s = 1, 2$$
$$x_1 \geq 0$$

Two-Stage Problems with Recourse - Exercise 2

- The SP is:

$$
\begin{aligned}
&\min 2 \cdot x_1 + \sum_{s=1}^{2} p_s(3 \cdot y_{2s}) \\
&x_1 + y_{2s} \geq D_s \qquad s = 1, 2 \\
&x_1 \geq 0 \\
&y_{2s} \geq 0 \qquad s = 1, 2
\end{aligned}
$$

Two-Stage Problems with Recourse - Exercise 2

- Recall here that $D_1 = 500$, $p_1 = 0.6$; $D_2 = 700$, $p_2 = 0.4$ with ordinary production costing 2 and recourse production costing 3.

Two-Stage Problems with Recourse - Exercise 2

- Recall here that $D_1 = 500$, $p_1 = 0.6$; $D_2 = 700$, $p_2 = 0.4$ with ordinary production costing 2 and recourse production costing 3.
- The complete SP is:

Two-Stage Problems with Recourse - Exercise 2

- Recall here that $D_1 = 500$, $p_1 = 0.6$; $D_2 = 700$, $p_2 = 0.4$ with ordinary production costing 2 and recourse production costing 3.
- The complete SP is:

$$\min 2 \cdot x_1 + 1.8 \cdot y_{21} + 1.2 \cdot y_{22}$$

Two-Stage Problems with Recourse - Exercise 2

- Recall here that $D_1 = 500$, $p_1 = 0.6$; $D_2 = 700$, $p_2 = 0.4$ with ordinary production costing 2 and recourse production costing 3.
- The complete SP is:

$$\min 2 \cdot x_1 + 1.8 \cdot y_{21} + 1.2 \cdot y_{22}$$
$$x_1 + y_{21} \geq 500$$

Two-Stage Problems with Recourse - Exercise 2

- Recall here that $D_1 = 500$, $p_1 = 0.6$; $D_2 = 700$, $p_2 = 0.4$ with ordinary production costing 2 and recourse production costing 3.
- The complete SP is:

$$\min 2 \cdot x_1 + 1.8 \cdot y_{21} + 1.2 \cdot y_{22}$$
$$x_1 + y_{21} \geq 500$$
$$x_1 + y_{22} \geq 700$$

Two-Stage Problems with Recourse - Exercise 2

- Recall here that $D_1 = 500$, $p_1 = 0.6$; $D_2 = 700$, $p_2 = 0.4$ with ordinary production costing 2 and recourse production costing 3.
- The complete SP is:

$$\min 2 \cdot x_1 + 1.8 \cdot y_{21} + 1.2 \cdot y_{22}$$
$$x_1 + y_{21} \geq 500$$
$$x_1 + y_{22} \geq 700$$
$$x_1 \geq 0$$

Two-Stage Problems with Recourse - Exercise 2

- Recall here that $D_1 = 500$, $p_1 = 0.6$; $D_2 = 700$, $p_2 = 0.4$ with ordinary production costing 2 and recourse production costing 3.
- The complete SP is:

$$
\begin{aligned}
&\min 2 \cdot x_1 + 1.8 \cdot y_{21} + 1.2 \cdot y_{22} \\
&x_1 + y_{21} \geq 500 \\
&x_1 + y_{22} \geq 700 \\
&x_1 \geq 0 \\
&y_{21}, y_{22} \geq 0
\end{aligned}
$$

Two-Stage Problems with Recourse - Exercise 2

- Recall here that $D_1 = 500$, $p_1 = 0.6$; $D_2 = 700$, $p_2 = 0.4$ with ordinary production costing 2 and recourse production costing 3.
- The complete SP is:

$$
\begin{aligned}
&\min 2 \cdot x_1 + 1.8 \cdot y_{21} + 1.2 \cdot y_{22} \\
&x_1 + y_{21} \geq 500 \\
&x_1 + y_{22} \geq 700 \\
&x_1 \geq 0 \\
&y_{21}, y_{22} \geq 0
\end{aligned}
$$

- The above program can be solved using Linear Programming.

Two-Stage Problems with Recourse - Exercise 2

- Recall here that $D_1 = 500$, $p_1 = 0.6$; $D_2 = 700$, $p_2 = 0.4$ with ordinary production costing 2 and recourse production costing 3.
- The complete SP is:

$$\begin{aligned}
&\min 2 \cdot x_1 + 1.8 \cdot y_{21} + 1.2 \cdot y_{22} \\
&x_1 + y_{21} \geq 500 \\
&x_1 + y_{22} \geq 700 \\
&x_1 \geq 0 \\
&y_{21}, y_{22} \geq 0
\end{aligned}$$

- The above program can be solved using Linear Programming.The production quantity that minimizes expected cost is $x_1 = 500$ and the minimum cost is 1240

Multi-Stage Problems

Multi-Stage Problems

- The recourse decisions can be taken at several points in time, called stages.

Multi-Stage Problems

- The recourse decisions can be taken at several points in time, called stages.
- Let $n \geq 2$ be the number of stages.

Multi-Stage Problems

- The recourse decisions can be taken at several points in time, called stages.
- Let $n \geq 2$ be the number of stages.
- The random event ω is a vector $(o_1, \ldots, o_{n-1})$ that gets revealed progressively over time.

Multi-Stage Problems

- The recourse decisions can be taken at several points in time, called stages.
- Let $n \geq 2$ be the number of stages.
- The random event ω is a vector $(o_1, \ldots, o_{n-1})$ that gets revealed progressively over time.
- The first-stage decisions are taken before any component of ω is revealed.

Multi-Stage Problems

- The recourse decisions can be taken at several points in time, called stages.
- Let $n \geq 2$ be the number of stages.
- The random event ω is a vector $(o_1, \ldots, o_{n-1})$ that gets revealed progressively over time.
- The first-stage decisions are taken before any component of ω is revealed.
- Then o_1 is revealed. With this knowledge, one takes the second-stage decisions.

Multi-Stage Problems

- The recourse decisions can be taken at several points in time, called stages.
- Let $n \geq 2$ be the number of stages.
- The random event ω is a vector $(o_1, \ldots, o_{n-1})$ that gets revealed progressively over time.
- The first-stage decisions are taken before any component of ω is revealed.
- Then o_1 is revealed. With this knowledge, one takes the second-stage decisions.
- After that, o_2 is revealed, and so on.

Multi-Stage Problems

- The recourse decisions can be taken at several points in time, called stages.
- Let $n \geq 2$ be the number of stages.
- The random event ω is a vector $(o_1, \ldots, o_{n-1})$ that gets revealed progressively over time.
- The first-stage decisions are taken before any component of ω is revealed.
- Then o_1 is revealed. With this knowledge, one takes the second-stage decisions.
- After that, o_2 is revealed, and so on.
- We assume that $\Omega = \{\omega_1, \ldots, \omega_S\}$ is a finite set.

Multi-Stage Problems

- The recourse decisions can be taken at several points in time, called stages.
- Let $n \geq 2$ be the number of stages.
- The random event ω is a vector $(o_1, \ldots, o_{n-1})$ that gets revealed progressively over time.
- The first-stage decisions are taken before any component of ω is revealed.
- Then o_1 is revealed. With this knowledge, one takes the second-stage decisions.
- After that, o_2 is revealed, and so on.
- We assume that $\Omega = \{\omega_1, \ldots, \omega_S\}$ is a finite set.
- Let p_k be the probability of scenario ω_k, for $k = 1, \ldots, S$.

Multi-Stage Problems

- Some scenarios ω_k may be identical in their first components and only become differentiated in the later stages.

Multi-Stage Problems

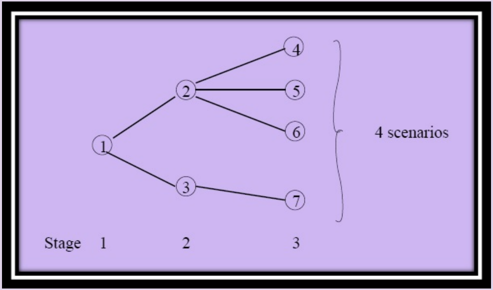
- Some scenarios ω_k may be identical in their first components and only become differentiated in the later stages.
- Therefore, it is convenient to introduce the scenario tree.

Multi-Stage Problems

- Some scenarios ω_k may be identical in their first components and only become differentiated in the later stages.
- Therefore, it is convenient to introduce the scenario tree.
- Scenario tree illustrates how the scenarios branch off at each stage.

Multi-Stage Problems

- Some scenarios ω_k may be identical in their first components and only become differentiated in the later stages.
- Therefore, it is convenient to introduce the scenario tree.
- Scenario tree illustrates how the scenarios branch off at each stage.



Multi-Stage Problems

- The nodes are labeled 1 through N

Multi-Stage Problems

- The nodes are labeled 1 through N
  - Node 1 is the root.

Multi-Stage Problems

- The nodes are labeled 1 through N
  - Node 1 is the root.
  - Each node is in one stage.

Multi-Stage Problems

- The nodes are labeled 1 through N
  - Node 1 is the root.
  - Each node is in one stage.
  - The root is the unique node in stage 1.

Multi-Stage Problems

- The nodes are labeled 1 through N
  - Node 1 is the root.
  - Each node is in one stage.
  - The root is the unique node in stage 1.
  - Each node i in stage $k \geq 2$ is adjacent to a unique node $a(i)$ in stage $k-1$.
  - Node $a(i)$ is called the father of node i.

Multi-Stage Problems

- The nodes are labeled 1 through N
  - Node 1 is the root.
  - Each node is in one stage.
  - The root is the unique node in stage 1.
  - Each node i in stage $k \geq 2$ is adjacent to a unique node $a(i)$ in stage $k - 1$.
  - Node $a(i)$ is called the father of node i.
- The paths from the root to the leaves (in stage n) represent the scenarios.

Multi-Stage Problems

- The nodes are labeled 1 through N
  - Node 1 is the root.
  - Each node is in one stage.
  - The root is the unique node in stage 1.
  - Each node i in stage $k \geq 2$ is adjacent to a unique node $a(i)$ in stage $k-1$.
  - Node $a(i)$ is called the father of node i.
- The paths from the root to the leaves (in stage n) represent the scenarios.
  - Thus the last stage has as many nodes as scenarios.

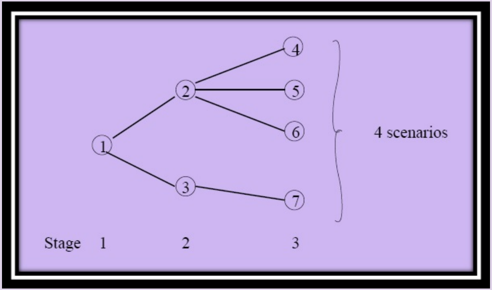
Multi-Stage Problems

- The nodes are labeled 1 through N
  - Node 1 is the root.
  - Each node is in one stage.
  - The root is the unique node in stage 1.
  - Each node i in stage $k \geq 2$ is adjacent to a unique node $a(i)$ in stage $k - 1$.
  - Node $a(i)$ is called the father of node i.
- The paths from the root to the leaves (in stage n) represent the scenarios.
  - Thus the last stage has as many nodes as scenarios.
  - These nodes are called the terminal nodes.

Multi-Stage Problems

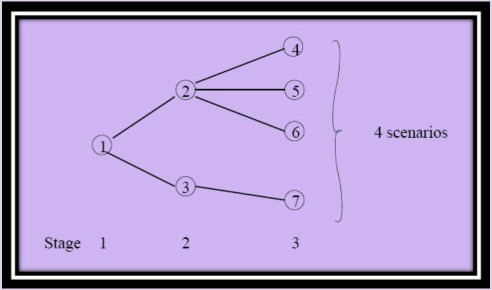
- The nodes are labeled 1 through N
  - Node 1 is the root.
  - Each node is in one stage.
  - The root is the unique node in stage 1.
  - Each node i in stage $k \geq 2$ is adjacent to a unique node $a(i)$ in stage $k-1$.
  - Node $a(i)$ is called the father of node i.
- The paths from the root to the leaves (in stage n) represent the scenarios.
  - Thus the last stage has as many nodes as scenarios.
  - These nodes are called the terminal nodes.
  - The collection of scenarios passing through node i in stage k have identical components $(o_1, \ldots, o_{k-1})$.

Multi-Stage Problems



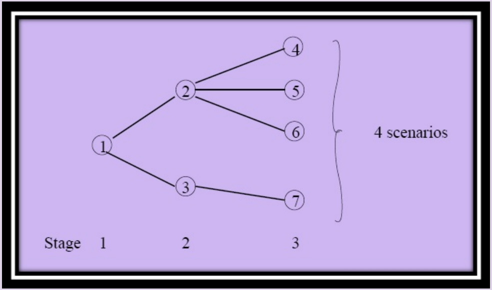
- Node 1 is the root,

Multi-Stage Problems



- Node 1 is the root,
- Nodes 4, 5, 6 and 7 are the terminal nodes.

Multi-Stage Problems



- Node 1 is the root,
- Nodes 4, 5, 6 and 7 are the terminal nodes.
- The father of Node 6 is Node 2, in other words $a(6) = 2$.

Multi-Stage Problems

- Associated with each node i is a recourse decision vector x_i.

Multi-Stage Problems

- Associated with each node i is a recourse decision vector x_i.
- For a node i in stage k, the decisions x_i are taken based on the information that has been revealed up to stage k.

Multi-Stage Problems

- Associated with each node i is a recourse decision vector x_i.
- For a node i in stage k, the decisions x_i are taken based on the information that has been revealed up to stage k.
- Let q_i be the probability of node i, conditional on being in Stage k.

Multi-Stage Problems

- Associated with each node i is a recourse decision vector x_i.
- For a node i in stage k, the decisions x_i are taken based on the information that has been revealed up to stage k.
- Let q_i be the probability of node i, conditional on being in Stage k.
- The multi-stage stochastic program with recourse can be formulated as follows:

Multi-Stage Problems

- Associated with each node i is a recourse decision vector x_i.
- For a node i in stage k, the decisions x_i are taken based on the information that has been revealed up to stage k.
- Let q_i be the probability of node i, conditional on being in Stage k.
- The multi-stage stochastic program with recourse can be formulated as follows:

$$\max_{x_1 \ldots x_N} \sum_{i=1}^{N} q_i c_i^T x_i \qquad (6)$$

Multi-Stage Problems

- Associated with each node i is a recourse decision vector x_i.
- For a node i in stage k, the decisions x_i are taken based on the information that has been revealed up to stage k.
- Let q_i be the probability of node i, conditional on being in Stage k.
- The multi-stage stochastic program with recourse can be formulated as follows:

$$\max_{x_1 \ldots x_N} \sum_{i=1}^{N} q_i c_i^T x_i \qquad (6)$$

$$Ax_1 = b$$

Multi-Stage Problems

- Associated with each node i is a recourse decision vector x_i.
- For a node i in stage k, the decisions x_i are taken based on the information that has been revealed up to stage k.
- Let q_i be the probability of node i, conditional on being in Stage k.
- The multi-stage stochastic program with recourse can be formulated as follows:

$$\max_{x_1 \ldots x_N} \sum_{i=1}^{N} q_i c_i^T x_i \qquad (6)$$

$$Ax_1 = b$$

$$B_i x_{a(i)} + C_i x_i = d_i \qquad \mathit{for} \quad i = 2 \ldots N$$

Multi-Stage Problems

- Associated with each node i is a recourse decision vector x_i.
- For a node i in stage k, the decisions x_i are taken based on the information that has been revealed up to stage k.
- Let q_i be the probability of node i, conditional on being in Stage k.
- The multi-stage stochastic program with recourse can be formulated as follows:

$$\max_{x_1 \dots x_N} \sum_{i=1}^{N} q_i c_i^T x_i \qquad (6)$$

$$Ax_1 = b$$

$$B_i x_{a(i)} + C_i x_i = d_i \qquad \mathit{for} \quad i = 2 \dots N$$

$$x_i \geq 0$$

Multi-Stage Problems

- Associated with each node i is a recourse decision vector x_i.
- For a node i in stage k, the decisions x_i are taken based on the information that has been revealed up to stage k.
- Let q_i be the probability of node i, conditional on being in Stage k.
- The multi-stage stochastic program with recourse can be formulated as follows:

$$
\begin{aligned}
&\max_{x_1 \ldots x_N} \sum_{i=1}^{N} q_i c_i^T x_i \qquad (6)\\
&Ax_1 = b\\
&B_i x_{a(i)} + C_i x_i = d_i \qquad \textit{for} \quad i = 2 \ldots N\\
&x_i \geq 0
\end{aligned}
$$

Multi-Stage Problems

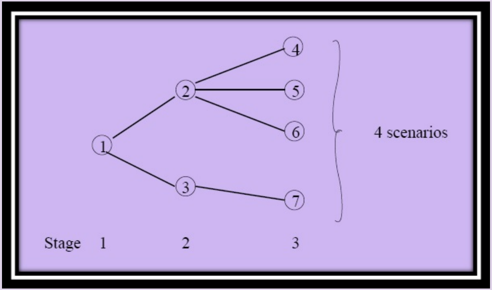
- A and b are deterministic constraints on the first stage decisions x_1

Multi-Stage Problems

- A and b are deterministic constraints on the first stage decisions x_1
- B_i, C_i, and d_i are stochastic constraints linking the recourse decisions x_i in node i to the recourse decisions $x_{a(i)}$ in its father node.

Multi-Stage Problems - Exercise 3

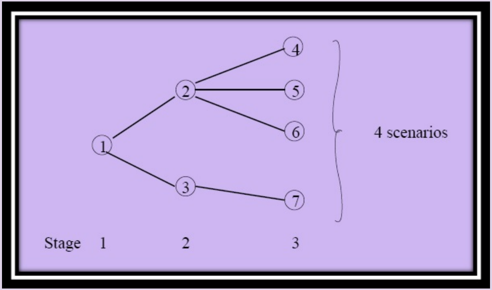
Exercise 3: Develop the Linear program (6) for the following scenario tree:



- The terminal nodes 4 to 7 correspond to scenarios 1 to 4 respectively.

Multi-Stage Problems - Exercise 3

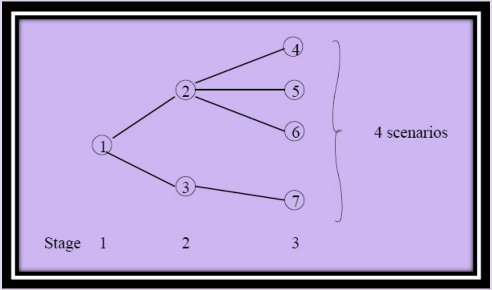
Exercise 3: Develop the Linear program (6) for the following scenario tree:



- The terminal nodes 4 to 7 correspond to scenarios 1 to 4 respectively.
- Thus we have $q_4 = p_1$, $q_5 = p_2$, $q_6 = p_3$ and $q_7 = p_4$, where p_k is the probability of scenario k.

Multi-Stage Problems - Exercise 3

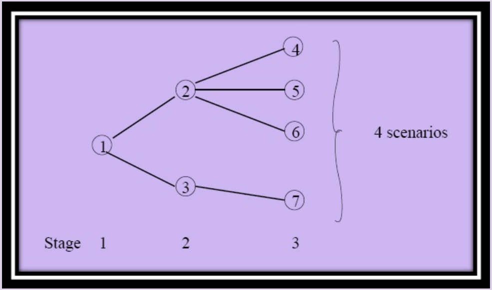
Exercise 3: Develop the Linear program (6) for the following scenario tree:



- The terminal nodes 4 to 7 correspond to scenarios 1 to 4 respectively.
- Thus we have $q_4 = p_1, q_5 = p_2, q_6 = p_3$ and $q_7 = p_4$, where p_k is the probability of scenario k. We also have $q_2 = p_1 + p_2 + p_3$,

Multi-Stage Problems - Exercise 3

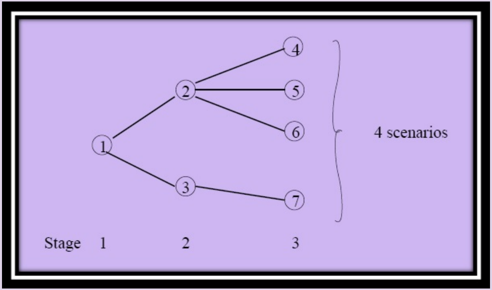
Exercise 3: Develop the Linear program (6) for the following scenario tree:



- The terminal nodes 4 to 7 correspond to scenarios 1 to 4 respectively.
- Thus we have $q_4 = p_1$, $q_5 = p_2$, $q_6 = p_3$ and $q_7 = p_4$, where p_k is the probability of scenario k. We also have $q_2 = p_1 + p_2 + p_3$, $q_3 = p_4$

Multi-Stage Problems - Exercise 3

Exercise 3: Develop the Linear program (6) for the following scenario tree:



- The terminal nodes 4 to 7 correspond to scenarios 1 to 4 respectively.
- Thus we have $q_4 = p_1$, $q_5 = p_2$, $q_6 = p_3$ and $q_7 = p_4$, where p_k is the probability of scenario k. We also have $q_2 = p_1 + p_2 + p_3$, $q_3 = p_4$ and $q_2 + q_3 = 1$.

Multi-Stage Problems - Exercise 3

$$\max c_1^T x_1 + q_2 c_2^T x_2 + q_3 c_3^T x_3 + p_1 c_4^T x_4 + p_2 c_5^T x_5 + p_3 c_6^T x_6 + p_4 c_7^T x_7$$

Multi-Stage Problems - Exercise 3

$$\max c_1^T x_1 + q_2 c_2^T x_2 + q_3 c_3^T x_3 + p_1 c_4^T x_4 + p_2 c_5^T x_5 + p_3 c_6^T x_6 + p_4 c_7^T x_7$$

$$Ax_1 = b$$

Multi-Stage Problems - Exercise 3

$$\max c_1^T x_1 + q_2 c_2^T x_2 + q_3 c_3^T x_3 + p_1 c_4^T x_4 + p_2 c_5^T x_5 + p_3 c_6^T x_6 + p_4 c_7^T x_7$$

$$Ax_1 = b$$

$$B_2 x_1 + C_2 x_2 = d_2$$

Multi-Stage Problems - Exercise 3

$$\max c_1^T x_1 + q_2 c_2^T x_2 + q_3 c_3^T x_3 + p_1 c_4^T x_4 + p_2 c_5^T x_5 + p_3 c_6^T x_6 + p_4 c_7^T x_7$$

$$Ax_1 = b$$

$$B_2 x_1 + C_2 x_2 = d_2$$

$$B_3 x_1 + C_3 x_3 = d_3$$

Multi-Stage Problems - Exercise 3

$$\max c_1^T x_1 + q_2 c_2^T x_2 + q_3 c_3^T x_3 + p_1 c_4^T x_4 + p_2 c_5^T x_5 + p_3 c_6^T x_6 + p_4 c_7^T x_7$$

$$Ax_1 = b$$

$$B_2 x_1 + C_2 x_2 = d_2$$

$$B_3 x_1 + C_3 x_3 = d_3$$

$$B_4 x_2 + C_4 x_4 = d_4$$

Multi-Stage Problems - Exercise 3

$$\max c_1^T x_1 + q_2 c_2^T x_2 + q_3 c_3^T x_3 + p_1 c_4^T x_4 + p_2 c_5^T x_5 + p_3 c_6^T x_6 + p_4 c_7^T x_7$$

$$Ax_1 = b$$

$$B_2 x_1 + C_2 x_2 = d_2$$

$$B_3 x_1 + C_3 x_3 = d_3$$

$$B_4 x_2 + C_4 x_4 = d_4$$

$$B_5 x_2 + C_5 x_5 = d_5$$

Multi-Stage Problems - Exercise 3

$$\max c_1^T x_1 + q_2 c_2^T x_2 + q_3 c_3^T x_3 + p_1 c_4^T x_4 + p_2 c_5^T x_5 + p_3 c_6^T x_6 + p_4 c_7^T x_7$$

$$Ax_1 = b$$

$$B_2 x_1 + C_2 x_2 = d_2$$

$$B_3 x_1 + C_3 x_3 = d_3$$

$$B_4 x_2 + C_4 x_4 = d_4$$

$$B_5 x_2 + C_5 x_5 = d_5$$

$$B_6 x_2 + C_6 x_6 = d_6$$

Multi-Stage Problems - Exercise 3

$$\max c_1^T x_1 + q_2 c_2^T x_2 + q_3 c_3^T x_3 + p_1 c_4^T x_4 + p_2 c_5^T x_5 + p_3 c_6^T x_6 + p_4 c_7^T x_7$$

$$Ax_1 = b$$

$$B_2 x_1 + C_2 x_2 = d_2$$

$$B_3 x_1 + C_3 x_3 = d_3$$

$$B_4 x_2 + C_4 x_4 = d_4$$

$$B_5 x_2 + C_5 x_5 = d_5$$

$$B_6 x_2 + C_6 x_6 = d_6$$

$$B_7 x_3 + C_7 x_7 = d_7$$

Multi-Stage Problems - Exercise 3

$$\max c_1^T x_1 + q_2 c_2^T x_2 + q_3 c_3^T x_3 + p_1 c_4^T x_4 + p_2 c_5^T x_5 + p_3 c_6^T x_6 + p_4 c_7^T x_7$$

$$Ax_1 = b$$

$$B_2 x_1 + C_2 x_2 = d_2$$

$$B_3 x_1 + C_3 x_3 = d_3$$

$$B_4 x_2 + C_4 x_4 = d_4$$

$$B_5 x_2 + C_5 x_5 = d_5$$

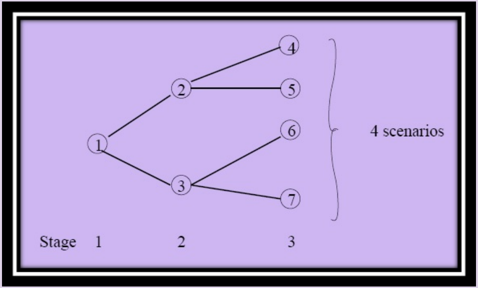
$$B_6 x_2 + C_6 x_6 = d_6$$

$$B_7 x_3 + C_7 x_7 = d_7$$

$$x_i \geq 0$$

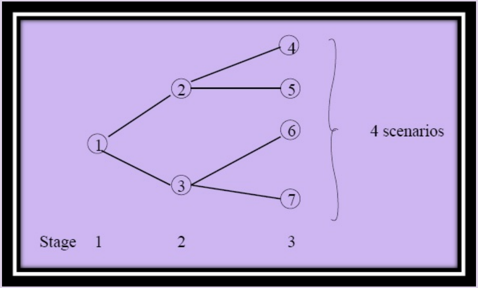
Multi-Stage Problems - Exercise 4

Exercise 4: Develop the Linear program (6) for the following scenario tree:



Multi-Stage Problems - Exercise 4

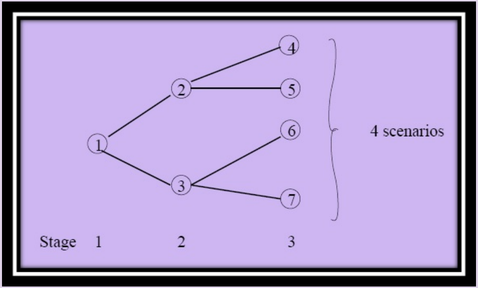
Exercise 4: Develop the Linear program (6) for the following scenario tree:



- The terminal nodes 4 to 7 correspond to scenarios 1 to 4 respectively.

Multi-Stage Problems - Exercise 4

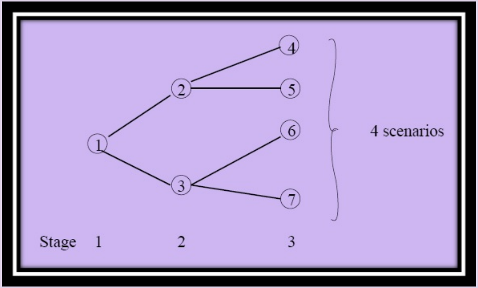
Exercise 4: Develop the Linear program (6) for the following scenario tree:



- The terminal nodes 4 to 7 correspond to scenarios 1 to 4 respectively.
- Thus we have $q_4 = p_1, q_5 = p_2, q_6 = p_3$ and $q_7 = p_4$, where p_k is the probability of scenario k.

Multi-Stage Problems - Exercise 4

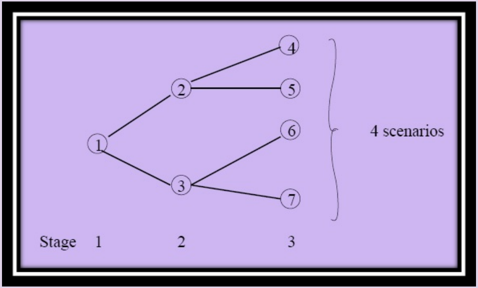
Exercise 4: Develop the Linear program (6) for the following scenario tree:



- The terminal nodes 4 to 7 correspond to scenarios 1 to 4 respectively.
- Thus we have $q_4 = p_1$, $q_5 = p_2$, $q_6 = p_3$ and $q_7 = p_4$, where p_k is the probability of scenario k. We also have $q_2 = p_1 + p_2$,

Multi-Stage Problems - Exercise 4

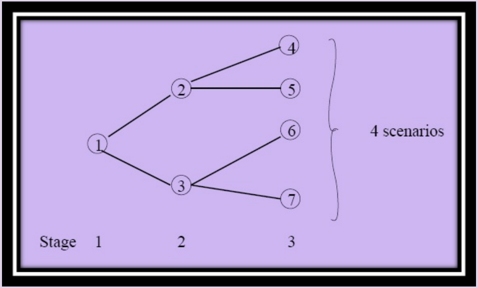
Exercise 4: Develop the Linear program (6) for the following scenario tree:



- The terminal nodes 4 to 7 correspond to scenarios 1 to 4 respectively.
- Thus we have $q_4 = p_1, q_5 = p_2, q_6 = p_3$ and $q_7 = p_4$, where p_k is the probability of scenario k. We also have $q_2 = p_1 + p_2$, $q_3 = p_4 + p_3$ and

Multi-Stage Problems - Exercise 4

Exercise 4: Develop the Linear program (6) for the following scenario tree:



- The terminal nodes 4 to 7 correspond to scenarios 1 to 4 respectively.
- Thus we have $q_4 = p_1, q_5 = p_2, q_6 = p_3$ and $q_7 = p_4$, where p_k is the probability of scenario k. We also have $q_2 = p_1 + p_2$, $q_3 = p_4 + p_3$ and $q_2 + q_3 = 1$.

Multi-Stage Problems - Exercise 5

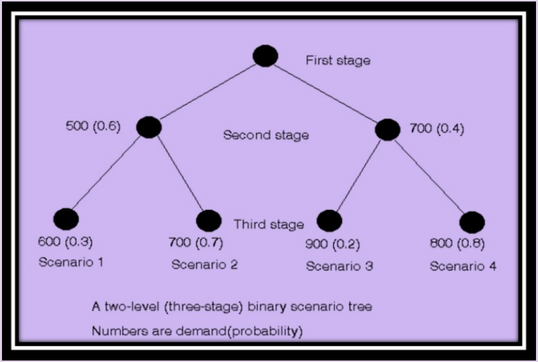
Exercise 5: Extend the two-stage problem we have considered in Exercise 2 to three stages.

Multi-Stage Problems - Exercise 5

Exercise 5: Extend the two-stage problem we have considered in Exercise 2 to three stages.Consider the three-stage binary scenario tree shown below.

Multi-Stage Problems - Exercise 5

Exercise 5: Extend the two-stage problem we have considered in Exercise 2 to three stages.Consider the three-stage binary scenario tree shown below.



Multi-Stage Problems - Exercise 5

- We initially make a decision about how much to produce.

Multi-Stage Problems - Exercise 5

- We initially make a decision about how much to produce.
- At the second stage, we have two possible realizations of the stochastic demand:

Multi-Stage Problems - Exercise 5

- We initially make a decision about how much to produce.
- At the second stage, we have two possible realizations of the stochastic demand:
  - a demand of 500 with probability 0.6

Multi-Stage Problems - Exercise 5

- We initially make a decision about how much to produce.
- At the second stage, we have two possible realizations of the stochastic demand:
  - a demand of 500 with probability 0.6
  - a demand of 700 with probability 0.4

Multi-Stage Problems - Exercise 5

- We initially make a decision about how much to produce.
- At the second stage, we have two possible realizations of the stochastic demand:
  - a demand of 500 with probability 0.6
  - a demand of 700 with probability 0.4
- After this realization, we make a decision as to how much to produce to meet demand in the next period (THE THIRD-STAGE).

Multi-Stage Problems - Exercise 5

- We initially make a decision about how much to produce.
- At the second stage, we have two possible realizations of the stochastic demand:
  - a demand of 500 with probability 0.6
  - a demand of 700 with probability 0.4
- After this realization, we make a decision as to how much to produce to meet demand in the next period (THE THIRD-STAGE).
- At the third stage, we have two possible realizations of the stochastic demand,

Multi-Stage Problems - Exercise 5

- We initially make a decision about how much to produce.
- At the second stage, we have two possible realizations of the stochastic demand:
  - a demand of 500 with probability 0.6
  - a demand of 700 with probability 0.4
- After this realization, we make a decision as to how much to produce to meet demand in the next period (THE THIRD-STAGE).
- At the third stage, we have two possible realizations of the stochastic demand, but these are different depending upon the realization at the second stage.

Multi-Stage Problems - Exercise 5

- We initially make a decision about how much to produce.
- At the second stage, we have two possible realizations of the stochastic demand:
  - a demand of 500 with probability 0.6
  - a demand of 700 with probability 0.4
- After this realization, we make a decision as to how much to produce to meet demand in the next period (THE THIRD-STAGE).
- At the third stage, we have two possible realizations of the stochastic demand, but these are different depending upon the realization at the second stage.
- For example, if the realized demand at the second stage was 500,

Multi-Stage Problems - Exercise 5

- We initially make a decision about how much to produce.
- At the second stage, we have two possible realizations of the stochastic demand:
  - a demand of 500 with probability 0.6
  - a demand of 700 with probability 0.4
- After this realization, we make a decision as to how much to produce to meet demand in the next period (THE THIRD-STAGE).
- At the third stage, we have two possible realizations of the stochastic demand, but these are different depending upon the realization at the second stage.
- For example, if the realized demand at the second stage was 500, then the possible realizations at the third stage are:

Multi-Stage Problems - Exercise 5

- We initially make a decision about how much to produce.
- At the second stage, we have two possible realizations of the stochastic demand:
  - a demand of 500 with probability 0.6
  - a demand of 700 with probability 0.4
- After this realization, we make a decision as to how much to produce to meet demand in the next period (THE THIRD-STAGE).
- At the third stage, we have two possible realizations of the stochastic demand, but these are different depending upon the realization at the second stage.
- For example, if the realized demand at the second stage was 500, then the possible realizations at the third stage are:
  - a demand of 600 with probability 0.3

Multi-Stage Problems - Exercise 5

- We initially make a decision about how much to produce.
- At the second stage, we have two possible realizations of the stochastic demand:
  - a demand of 500 with probability 0.6
  - a demand of 700 with probability 0.4
- After this realization, we make a decision as to how much to produce to meet demand in the next period (THE THIRD-STAGE).
- At the third stage, we have two possible realizations of the stochastic demand, but these are different depending upon the realization at the second stage.
- For example, if the realized demand at the second stage was 500, then the possible realizations at the third stage are:
  - a demand of 600 with probability 0.3
  - a demand of 700 with probability 0.7

Multi-Stage Problems - Exercise 5

- We initially make a decision about how much to produce.
- At the second stage, we have two possible realizations of the stochastic demand:
  - a demand of 500 with probability 0.6
  - a demand of 700 with probability 0.4
- After this realization, we make a decision as to how much to produce to meet demand in the next period (THE THIRD-STAGE).
- At the third stage, we have two possible realizations of the stochastic demand, but these are different depending upon the realization at the second stage.
- For example, if the realized demand at the second stage was 500, then the possible realizations at the third stage are:
  - a demand of 600 with probability 0.3
  - a demand of 700 with probability 0.7
- Note here at each level in the scenario tree the appropriate probabilities must sum to one.

Multi-Stage Problems - Exercise 5

- The four possible scenarios of the future:

Multi-Stage Problems - Exercise 5

- The four possible scenarios of the future:

Scenario	Second Stage	Third Stage	Probability

Multi-Stage Problems - Exercise 5

- The four possible scenarios of the future:

Scenario	Second Stage	Third Stage	Probability
1	500	600	0.6(0.3) = 0.18

Multi-Stage Problems - Exercise 5

- The four possible scenarios of the future:

Scenario	Second Stage	Third Stage	Probability
1	500	600	0.6(0.3) = 0.18
2	500	700	0.6(0.7) = 0.42

Multi-Stage Problems - Exercise 5

- The four possible scenarios of the future:

Scenario	Second Stage	Third Stage	Probability
1	500	600	0.6(0.3) = 0.18
2	500	700	0.6(0.7) = 0.42
3	700	900	0.4(0.2) = 0.08

Multi-Stage Problems - Exercise 5

- The four possible scenarios of the future:

Scenario	Second Stage	Third Stage	Probability
1	500	600	0.6(0.3) = 0.18
2	500	700	0.6(0.7) = 0.42
3	700	900	0.4(0.2) = 0.08
4	700	800	0.4(0.8) = 0.32

Multi-Stage Problems - Exercise 5

- The four possible scenarios of the future:

Scenario	Second Stage	Third Stage	Probability
1	500	600	0.6(0.3) = 0.18
2	500	700	0.6(0.7) = 0.42
3	700	900	0.4(0.2) = 0.08
4	700	800	0.4(0.8) = 0.32

- Note here that these probabilities add to one

Multi-Stage Problems - Exercise 5

- The four possible scenarios of the future:

Scenario	Second Stage	Third Stage	Probability
1	500	600	0.6(0.3) = 0.18
2	500	700	0.6(0.7) = 0.42
3	700	900	0.4(0.2) = 0.08
4	700	800	0.4(0.8) = 0.32

- Note here that these probabilities add to one (these 4 scenarios are the only possible futures that we can have).

Multi-Stage Problems - Exercise 5

- Let:

Multi-Stage Problems - Exercise 5

- Let:
  - $x_1 \geq 0$ be the number of units of X to produce now (at the first stage)

Multi-Stage Problems - Exercise 5

- Let:
  - $x_1 \geq 0$ be the number of units of X to produce now (at the first stage)
  - $y_{2s} \geq 0$ be the number of units of X to buy from the external supplier at the second stage in scenario s ($s = 1, \ldots, 4$)

Multi-Stage Problems - Exercise 5

- Let:
  - $x_1 \geq 0$ be the number of units of X to produce now (at the first stage)
  - $y_{2s} \geq 0$ be the number of units of X to buy from the external supplier at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $x_{2s} \geq 0$ be the number of units of X to produce at the second stage in scenario s ($s = 1, \ldots, 4$)

Multi-Stage Problems - Exercise 5

- Let:
  - $x_1 \geq 0$ be the number of units of X to produce now (at the first stage)
  - $y_{2s} \geq 0$ be the number of units of X to buy from the external supplier at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $x_{2s} \geq 0$ be the number of units of X to produce at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $y_{3s} \geq 0$ be the number of units of X to buy from the external supplier at the third stage in scenario s ($s = 1, \ldots, 4$)

Multi-Stage Problems - Exercise 5

- Let:
  - $x_1 \geq 0$ be the number of units of X to produce now (at the first stage)
  - $y_{2s} \geq 0$ be the number of units of X to buy from the external supplier at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $x_{2s} \geq 0$ be the number of units of X to produce at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $y_{3s} \geq 0$ be the number of units of X to buy from the external supplier at the third stage in scenario s ($s = 1, \ldots, 4$)
- To summarize:

Multi-Stage Problems - Exercise 5

- Let:
  - $x_1 \geq 0$ be the number of units of X to produce now (at the first stage)
  - $y_{2s} \geq 0$ be the number of units of X to buy from the external supplier at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $x_{2s} \geq 0$ be the number of units of X to produce at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $y_{3s} \geq 0$ be the number of units of X to buy from the external supplier at the third stage in scenario s ($s = 1, \ldots, 4$)
- To summarize:
  - In the first stage a decision as to how much to produce, x_1

Multi-Stage Problems - Exercise 5

- Let:
  - $x_1 \geq 0$ be the number of units of X to produce now (at the first stage)
  - $y_{2s} \geq 0$ be the number of units of X to buy from the external supplier at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $x_{2s} \geq 0$ be the number of units of X to produce at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $y_{3s} \geq 0$ be the number of units of X to buy from the external supplier at the third stage in scenario s ($s = 1, \ldots, 4$)
- To summarize:
  - In the first stage a decision as to how much to produce, x_1
  - In the second stage a realization of the stochastic element (demand)

Multi-Stage Problems - Exercise 5

- Let:
  - $x_1 \geq 0$ be the number of units of X to produce now (at the first stage)
  - $y_{2s} \geq 0$ be the number of units of X to buy from the external supplier at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $x_{2s} \geq 0$ be the number of units of X to produce at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $y_{3s} \geq 0$ be the number of units of X to buy from the external supplier at the third stage in scenario s ($s = 1, \ldots, 4$)
- To summarize:
  - In the first stage a decision as to how much to produce, x_1
  - In the second stage a realization of the stochastic element (demand)
  - A decision as to the values of the recourse variables, y_{2s}

Multi-Stage Problems - Exercise 5

- Let:
  - $x_1 \geq 0$ be the number of units of X to produce now (at the first stage)
  - $y_{2s} \geq 0$ be the number of units of X to buy from the external supplier at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $x_{2s} \geq 0$ be the number of units of X to produce at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $y_{3s} \geq 0$ be the number of units of X to buy from the external supplier at the third stage in scenario s ($s = 1, \ldots, 4$)
- To summarize:
  - In the first stage a decision as to how much to produce, x_1
  - In the second stage a realization of the stochastic element (demand)
  - A decision as to the values of the recourse variables, y_{2s}
  - In the second stage a decision as to how much to produce, x_{2s}

Multi-Stage Problems - Exercise 5

- Let:
  - $x_1 \geq 0$ be the number of units of X to produce now (at the first stage)
  - $y_{2s} \geq 0$ be the number of units of X to buy from the external supplier at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $x_{2s} \geq 0$ be the number of units of X to produce at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $y_{3s} \geq 0$ be the number of units of X to buy from the external supplier at the third stage in scenario s ($s = 1, \ldots, 4$)
- To summarize:
  - In the first stage a decision as to how much to produce, x_1
  - In the second stage a realization of the stochastic element (demand)
  - A decision as to the values of the recourse variables, y_{2s}
  - In the second stage a decision as to how much to produce, x_{2s}
  - In the third stage a realization of the stochastic element (demand)

Multi-Stage Problems - Exercise 5

- Let:
  - $x_1 \geq 0$ be the number of units of X to produce now (at the first stage)
  - $y_{2s} \geq 0$ be the number of units of X to buy from the external supplier at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $x_{2s} \geq 0$ be the number of units of X to produce at the second stage in scenario s ($s = 1, \ldots, 4$)
  - $y_{3s} \geq 0$ be the number of units of X to buy from the external supplier at the third stage in scenario s ($s = 1, \ldots, 4$)
- To summarize:
  - In the first stage a decision as to how much to produce, x_1
  - In the second stage a realization of the stochastic element (demand)
  - A decision as to the values of the recourse variables, y_{2s}
  - In the second stage a decision as to how much to produce, x_{2s}
  - In the third stage a realization of the stochastic element (demand)
  - A decision as to the values of the recourse variables, y_{3s}

Multi-Stage Problems - Exercise 5

- Consider the first stage, the constraints to ensure customer demand is satisfied are:

Multi-Stage Problems - Exercise 5

- Consider the first stage, the constraints to ensure customer demand is satisfied are:
  - $x_1 + y_{2s} \geq 500 \qquad (s = 1, 2)$

Multi-Stage Problems - Exercise 5

- Consider the first stage, the constraints to ensure customer demand is satisfied are:
  - $x_1 + y_{2s} \geq 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} \geq 700 \qquad (s = 3, 4)$

Multi-Stage Problems - Exercise 5

- Consider the first stage, the constraints to ensure customer demand is satisfied are:
  - $x_1 + y_{2s} \geq 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} \geq 700 \qquad (s = 3, 4)$
- Now at the second stage we will have units left over (i.e. inventory) to help meet future demand. This inventory level will be:

Multi-Stage Problems - Exercise 5

- Consider the first stage, the constraints to ensure customer demand is satisfied are:
  - $x_1 + y_{2s} \geq 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} \geq 700 \qquad (s = 3, 4)$
- Now at the second stage we will have units left over (i.e. inventory) to help meet future demand. This inventory level will be:
  - $x_1 + y_{2s} - 500 \qquad (s = 1, 2)$

Multi-Stage Problems - Exercise 5

- Consider the first stage, the constraints to ensure customer demand is satisfied are:
  - $x_1 + y_{2s} \geq 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} \geq 700 \qquad (s = 3, 4)$
- Now at the second stage we will have units left over (i.e. inventory) to help meet future demand. This inventory level will be:
  - $x_1 + y_{2s} - 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} - 700 \qquad (s = 3, 4)$

Multi-Stage Problems - Exercise 5

- Consider the first stage, the constraints to ensure customer demand is satisfied are:
  - $x_1 + y_{2s} \geq 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} \geq 700 \qquad (s = 3, 4)$
- Now at the second stage we will have units left over (i.e. inventory) to help meet future demand. This inventory level will be:
  - $x_1 + y_{2s} - 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} - 700 \qquad (s = 3, 4)$
- To ensure that demand is met in the third stage we have:

Multi-Stage Problems - Exercise 5

- Consider the first stage, the constraints to ensure customer demand is satisfied are:
  - $x_1 + y_{2s} \geq 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} \geq 700 \qquad (s = 3, 4)$
- Now at the second stage we will have units left over (i.e. inventory) to help meet future demand. This inventory level will be:
  - $x_1 + y_{2s} - 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} - 700 \qquad (s = 3, 4)$
- To ensure that demand is met in the third stage we have:

INVENTORY + AMOUNT PRODUCED + AMOUNT BOUGHT EXTERNALLY $\geq$ DEMAND

Multi-Stage Problems - Exercise 5

- Consider the first stage, the constraints to ensure customer demand is satisfied are:
  - $x_1 + y_{2s} \geq 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} \geq 700 \qquad (s = 3, 4)$
- Now at the second stage we will have units left over (i.e. inventory) to help meet future demand. This inventory level will be:
  - $x_1 + y_{2s} - 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} - 700 \qquad (s = 3, 4)$
- To ensure that demand is met in the third stage we have:

INVENTORY + AMOUNT PRODUCED + AMOUNT BOUGHT EXTERNALLY $\geq$ DEMAND

  - $x_1 + y_{2s} - 500 + x_{2s} + y_{3s} \geq 600 \qquad (s = 1)$

Multi-Stage Problems - Exercise 5

- Consider the first stage, the constraints to ensure customer demand is satisfied are:
  - $x_1 + y_{2s} \geq 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} \geq 700 \qquad (s = 3, 4)$
- Now at the second stage we will have units left over (i.e. inventory) to help meet future demand. This inventory level will be:
  - $x_1 + y_{2s} - 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} - 700 \qquad (s = 3, 4)$
- To ensure that demand is met in the third stage we have:

INVENTORY + AMOUNT PRODUCED + AMOUNT BOUGHT EXTERNALLY $\geq$ DEMAND

  - $x_1 + y_{2s} - 500 + x_{2s} + y_{3s} \geq 600 \qquad (s = 1)$
  - $x_1 + y_{2s} - 500 + x_{2s} + y_{3s} \geq 700 \qquad (s = 2)$

Multi-Stage Problems - Exercise 5

- Consider the first stage, the constraints to ensure customer demand is satisfied are:
  - $x_1 + y_{2s} \geq 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} \geq 700 \qquad (s = 3, 4)$
- Now at the second stage we will have units left over (i.e. inventory) to help meet future demand. This inventory level will be:
  - $x_1 + y_{2s} - 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} - 700 \qquad (s = 3, 4)$
- To ensure that demand is met in the third stage we have:

INVENTORY + AMOUNT PRODUCED + AMOUNT BOUGHT EXTERNALLY $\geq$ DEMAND

  - $x_1 + y_{2s} - 500 + x_{2s} + y_{3s} \geq 600 \qquad (s = 1)$
  - $x_1 + y_{2s} - 500 + x_{2s} + y_{3s} \geq 700 \qquad (s = 2)$
  - $x_1 + y_{2s} - 700 + x_{2s} + y_{3s} \geq 900 \qquad (s = 3)$

Multi-Stage Problems - Exercise 5

- Consider the first stage, the constraints to ensure customer demand is satisfied are:
  - $x_1 + y_{2s} \geq 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} \geq 700 \qquad (s = 3, 4)$
- Now at the second stage we will have units left over (i.e. inventory) to help meet future demand. This inventory level will be:
  - $x_1 + y_{2s} - 500 \qquad (s = 1, 2)$
  - $x_1 + y_{2s} - 700 \qquad (s = 3, 4)$
- To ensure that demand is met in the third stage we have:

INVENTORY + AMOUNT PRODUCED + AMOUNT BOUGHT EXTERNALLY $\geq$ DEMAND

  - $x_1 + y_{2s} - 500 + x_{2s} + y_{3s} \geq 600 \qquad (s = 1)$
  - $x_1 + y_{2s} - 500 + x_{2s} + y_{3s} \geq 700 \qquad (s = 2)$
  - $x_1 + y_{2s} - 700 + x_{2s} + y_{3s} \geq 900 \qquad (s = 3)$
  - $x_1 + y_{2s} - 700 + x_{2s} + y_{3s} \geq 800 \qquad (s = 4)$

Multi-Stage Problems - Exercise 5

- Scenarios with a common history must (logically) have the same set of decisions:

Multi-Stage Problems - Exercise 5

- Scenarios with a common history must (logically) have the same set of decisions:
  - Scenario 1 and 2, second stage:
    - $y_{21} = y_{22}$
    - $x_{21} = x_{22}$

Multi-Stage Problems - Exercise 5

- Scenarios with a common history must (logically) have the same set of decisions:
  - Scenario 1 and 2, second stage:
    - $y_{21} = y_{22}$
    - $x_{21} = x_{22}$
  - Scenario 3 and 4, second stage:
    - $y_{23} = y_{24}$
    - $x_{23} = x_{24}$

Multi-Stage Problems - Exercise 5

- Scenarios with a common history must (logically) have the same set of decisions:
  - Scenario 1 and 2, second stage:
    - $y_{21} = y_{22}$
    - $x_{21} = x_{22}$
  - Scenario 3 and 4, second stage:
    - $y_{23} = y_{24}$
    - $x_{23} = x_{24}$
- We have just one cost incurred with certainty, namely $2 \cdot x_1$,

Multi-Stage Problems - Exercise 5

- Scenarios with a common history must (logically) have the same set of decisions:
  - Scenario 1 and 2, second stage:
    - $y_{21} = y_{22}$
    - $x_{21} = x_{22}$
  - Scenario 3 and 4, second stage:
    - $y_{23} = y_{24}$
    - $x_{23} = x_{24}$
- We have just one cost incurred with certainty, namely $2 \cdot x_1$, all other costs are probabilistic.

Multi-Stage Problems - Exercise 5

- Scenarios with a common history must (logically) have the same set of decisions:
  - Scenario 1 and 2, second stage:
    - $y_{21} = y_{22}$
    - $x_{21} = x_{22}$
  - Scenario 3 and 4, second stage:
    - $y_{23} = y_{24}$
    - $x_{23} = x_{24}$
- We have just one cost incurred with certainty, namely $2 \cdot x_1$, all other costs are probabilistic.
- The easiest way to summarize these costs is to look at each scenario in turn:

Multi-Stage Problems - Exercise 5

- Scenarios with a common history must (logically) have the same set of decisions:
  - Scenario 1 and 2, second stage:
    - $y_{21} = y_{22}$
    - $x_{21} = x_{22}$
  - Scenario 3 and 4, second stage:
    - $y_{23} = y_{24}$
    - $x_{23} = x_{24}$
- We have just one cost incurred with certainty, namely $2 \cdot x_1$, all other costs are probabilistic.
- The easiest way to summarize these costs is to look at each scenario in turn:

Scenario	Probability	Cost

Multi-Stage Problems - Exercise 5

- Scenarios with a common history must (logically) have the same set of decisions:
  - Scenario 1 and 2, second stage:
    - $y_{21} = y_{22}$
    - $x_{21} = x_{22}$
  - Scenario 3 and 4, second stage:
    - $y_{23} = y_{24}$
    - $x_{23} = x_{24}$
- We have just one cost incurred with certainty, namely $2 \cdot x_1$, all other costs are probabilistic.
- The easiest way to summarize these costs is to look at each scenario in turn:

Scenario	Probability	Cost
1	0.18	$2 \cdot x_{21} + 3 \cdot y_{21} + 3 \cdot y_{31}$

Multi-Stage Problems - Exercise 5

- Scenarios with a common history must (logically) have the same set of decisions:
  - Scenario 1 and 2, second stage:
    - $y_{21} = y_{22}$
    - $x_{21} = x_{22}$
  - Scenario 3 and 4, second stage:
    - $y_{23} = y_{24}$
    - $x_{23} = x_{24}$
- We have just one cost incurred with certainty, namely $2 \cdot x_1$, all other costs are probabilistic.
- The easiest way to summarize these costs is to look at each scenario in turn:

Scenario	Probability	Cost
1	0.18	$2 \cdot x_{21} + 3 \cdot y_{21} + 3 \cdot y_{31}$
2	0.42	$2 \cdot x_{22} + 3 \cdot y_{22} + 3 \cdot y_{32}$

Multi-Stage Problems - Exercise 5

- Scenarios with a common history must (logically) have the same set of decisions:
  - Scenario 1 and 2, second stage:
    - $y_{21} = y_{22}$
    - $x_{21} = x_{22}$
  - Scenario 3 and 4, second stage:
    - $y_{23} = y_{24}$
    - $x_{23} = x_{24}$
- We have just one cost incurred with certainty, namely $2 \cdot x_1$, all other costs are probabilistic.
- The easiest way to summarize these costs is to look at each scenario in turn:

Scenario	Probability	Cost
1	0.18	$2 \cdot x_{21} + 3 \cdot y_{21} + 3 \cdot y_{31}$
2	0.42	$2 \cdot x_{22} + 3 \cdot y_{22} + 3 \cdot y_{32}$
3	0.08	$2 \cdot x_{23} + 3 \cdot y_{23} + 3 \cdot y_{33}$

Multi-Stage Problems - Exercise 5

- Scenarios with a common history must (logically) have the same set of decisions:
  - Scenario 1 and 2, second stage:
    - $y_{21} = y_{22}$
    - $x_{21} = x_{22}$
  - Scenario 3 and 4, second stage:
    - $y_{23} = y_{24}$
    - $x_{23} = x_{24}$
- We have just one cost incurred with certainty, namely $2 \cdot x_1$, all other costs are probabilistic.
- The easiest way to summarize these costs is to look at each scenario in turn:

Scenario	Probability	Cost
1	0.18	$2 \cdot x_{21} + 3 \cdot y_{21} + 3 \cdot y_{31}$
2	0.42	$2 \cdot x_{22} + 3 \cdot y_{22} + 3 \cdot y_{32}$
3	0.08	$2 \cdot x_{23} + 3 \cdot y_{23} + 3 \cdot y_{33}$
4	0.32	$2 \cdot x_{24} + 3 \cdot y_{24} + 3 \cdot y_{34}$

Multi-Stage Problems - Exercise 5

- Thus, The complete three- stage stochastic program is:

Multi-Stage Problems - Exercise 5

- Thus, The complete three- stage stochastic program is:

$$
\begin{aligned}
\min 2 \cdot x_1 &+ 0.18 \cdot (2 \cdot x_{21} + 3 \cdot y_{21} + 3 \cdot y_{31}) + 0.42 \cdot (2 \cdot x_{22} + 3 \cdot y_{22} + 3 \cdot y_{32}) \\
&+ 0.08 \cdot (2 \cdot x_{23} + 3 \cdot y_{23} + 3 \cdot y_{33}) + 0.32 \cdot (2 \cdot x_{24} + 3 \cdot y_{24} + 3 \cdot y_{34})
\end{aligned}
$$

Multi-Stage Problems - Exercise 5

- Thus, The complete three- stage stochastic program is:

$$\min 2 \cdot x_1 + 0.18 \cdot (2 \cdot x_{21} + 3 \cdot y_{21} + 3 \cdot y_{31}) + 0.42 \cdot (2 \cdot x_{22} + 3 \cdot y_{22} + 3 \cdot y_{32})$$
$$+0.08 \cdot (2 \cdot x_{23} + 3 \cdot y_{23} + 3 \cdot y_{33}) + 0.32 \cdot (2 \cdot x_{24} + 3 \cdot y_{24} + 3 \cdot y_{34})$$

subject to

Multi-Stage Problems - Exercise 5

- Thus, The complete three- stage stochastic program is:

$$\begin{aligned}\min 2 \cdot x_1 &+ 0.18 \cdot (2 \cdot x_{21} + 3 \cdot y_{21} + 3 \cdot y_{31}) + 0.42 \cdot (2 \cdot x_{22} + 3 \cdot y_{22} + 3 \cdot y_{32}) \\ &+ 0.08 \cdot (2 \cdot x_{23} + 3 \cdot y_{23} + 3 \cdot y_{33}) + 0.32 \cdot (2 \cdot x_{24} + 3 \cdot y_{24} + 3 \cdot y_{34})\end{aligned}$$

subject to

$$x_1 + y_{2s} - 500 + x_{2s} + y_{3s} \geq 600 \qquad (s = 1)$$

Multi-Stage Problems - Exercise 5

- Thus, The complete three- stage stochastic program is:

$$\begin{aligned}\min 2 \cdot x_1 &+ 0.18 \cdot (2 \cdot x_{21} + 3 \cdot y_{21} + 3 \cdot y_{31}) + 0.42 \cdot (2 \cdot x_{22} + 3 \cdot y_{22} + 3 \cdot y_{32}) \\ &+ 0.08 \cdot (2 \cdot x_{23} + 3 \cdot y_{23} + 3 \cdot y_{33}) + 0.32 \cdot (2 \cdot x_{24} + 3 \cdot y_{24} + 3 \cdot y_{34})\end{aligned}$$

subject to

$$\begin{aligned} x_1 + y_{2s} - 500 + x_{2s} + y_{3s} &\geq 600 \qquad (s = 1) \\ x_1 + y_{2s} - 500 + x_{2s} + y_{3s} &\geq 700 \qquad (s = 2) \end{aligned}$$

Multi-Stage Problems - Exercise 5

- Thus, The complete three- stage stochastic program is:

$$\min 2 \cdot x_1 + 0.18 \cdot (2 \cdot x_{21} + 3 \cdot y_{21} + 3 \cdot y_{31}) + 0.42 \cdot (2 \cdot x_{22} + 3 \cdot y_{22} + 3 \cdot y_{32})$$
$$+0.08 \cdot (2 \cdot x_{23} + 3 \cdot y_{23} + 3 \cdot y_{33}) + 0.32 \cdot (2 \cdot x_{24} + 3 \cdot y_{24} + 3 \cdot y_{34})$$

subject to

$$x_1 + y_{2s} - 500 + x_{2s} + y_{3s} \geq 600 \qquad (s = 1)$$
$$x_1 + y_{2s} - 500 + x_{2s} + y_{3s} \geq 700 \qquad (s = 2)$$
$$x_1 + y_{2s} - 700 + x_{2s} + y_{3s} \geq 900 \qquad (s = 3)$$

Multi-Stage Problems - Exercise 5

- Thus, The complete three- stage stochastic program is:

$$\min 2 \cdot x_1 + 0.18 \cdot (2 \cdot x_{21} + 3 \cdot y_{21} + 3 \cdot y_{31}) + 0.42 \cdot (2 \cdot x_{22} + 3 \cdot y_{22} + 3 \cdot y_{32}) \\ +0.08 \cdot (2 \cdot x_{23} + 3 \cdot y_{23} + 3 \cdot y_{33}) + 0.32 \cdot (2 \cdot x_{24} + 3 \cdot y_{24} + 3 \cdot y_{34})$$

subject to

$$\begin{aligned} x_1 + y_{2s} - 500 + x_{2s} + y_{3s} &\geq 600 \qquad (s = 1) \\ x_1 + y_{2s} - 500 + x_{2s} + y_{3s} &\geq 700 \qquad (s = 2) \\ x_1 + y_{2s} - 700 + x_{2s} + y_{3s} &\geq 900 \qquad (s = 3) \\ x_1 + y_{2s} - 700 + x_{2s} + y_{3s} &\geq 800 \qquad (s = 4) \end{aligned}$$

Multi-Stage Problems - Exercise 5

- Thus, The complete three- stage stochastic program is:

$$\min 2 \cdot x_1 + 0.18 \cdot (2 \cdot x_{21} + 3 \cdot y_{21} + 3 \cdot y_{31}) + 0.42 \cdot (2 \cdot x_{22} + 3 \cdot y_{22} + 3 \cdot y_{32})$$
$$+0.08 \cdot (2 \cdot x_{23} + 3 \cdot y_{23} + 3 \cdot y_{33}) + 0.32 \cdot (2 \cdot x_{24} + 3 \cdot y_{24} + 3 \cdot y_{34})$$

subject to

$$x_1 + y_{2s} - 500 + x_{2s} + y_{3s} \geq 600 \qquad (s = 1)$$
$$x_1 + y_{2s} - 500 + x_{2s} + y_{3s} \geq 700 \qquad (s = 2)$$
$$x_1 + y_{2s} - 700 + x_{2s} + y_{3s} \geq 900 \qquad (s = 3)$$
$$x_1 + y_{2s} - 700 + x_{2s} + y_{3s} \geq 800 \qquad (s = 4)$$
$$y_{21} = y_{22}, \quad x_{21} = x_{22}, \quad y_{23} = y_{24}, \quad x_{23} = x_{24}$$
$$all \;\; variables \geq 0$$

Multi-Stage Problems - Exercise 5

Solving the previous problem using LP solver:

Multi-Stage Problems - Exercise 5

Solving the previous problem using LP solver:

- Cost will be 2664

Multi-Stage Problems - Exercise 5

Solving the previous problem using LP solver:

- Cost will be 2664
- $x_1 = 700$

Multi-Stage Problems - Exercise 5

Solving the previous problem using LP solver:

- Cost will be 2664
- $x_1 = 700$
- $y_{21} = y_{22} = 0, \quad y_{23} = y_{24} = 0$

Multi-Stage Problems - Exercise 5

Solving the previous problem using LP solver:

- Cost will be 2664
- $x_1 = 700$
- $y_{21} = y_{22} = 0, \quad y_{23} = y_{24} = 0$
- $x_{21} = x_{22} = 500$

Multi-Stage Problems - Exercise 5

Solving the previous problem using LP solver:

- Cost will be 2664
- $x_1 = 700$
- $y_{21} = y_{22} = 0, \quad y_{23} = y_{24} = 0$
- $x_{21} = x_{22} = 500$
- $y_{31} = y_{32} = 0$

Multi-Stage Problems - Exercise 5

Solving the previous problem using LP solver:

- Cost will be 2664
- $x_1 = 700$
- $y_{21} = y_{22} = 0, \quad y_{23} = y_{24} = 0$
- $x_{21} = x_{22} = 500$
- $y_{31} = y_{32} = 0$
- $x_{23} = x_{24} = 800$

Multi-Stage Problems - Exercise 5

Solving the previous problem using LP solver:

- Cost will be 2664
- $x_1 = 700$
- $y_{21} = y_{22} = 0, \quad y_{23} = y_{24} = 0$
- $x_{21} = x_{22} = 500$
- $y_{31} = y_{32} = 0$
- $x_{23} = x_{24} = 800$
- $y_{33} = 100, y_{34} = 0$

Decomposition

Decomposition

- The size of the linear program (6) depends on the number of decision stages and the branching factor at each node of the scenario tree.

Decomposition

- The size of the linear program (6) depends on the number of decision stages and the branching factor at each node of the scenario tree.
- For example, a 4-stage model with 25 branches at each node has $25x25x25x25 = 390625$ scenarios.

Decomposition

- The size of the linear program (6) depends on the number of decision stages and the branching factor at each node of the scenario tree.
- For example, a 4-stage model with 25 branches at each node has $25x25x25x25 = 390625$ scenarios.
- Increasing the number of stages and branches quickly results in an explosion of dimensionality.

Decomposition

- The size of the linear program (6) depends on the number of decision stages and the branching factor at each node of the scenario tree.
- For example, a 4-stage model with 25 branches at each node has $25x25x25x25 = 390625$ scenarios.
- Increasing the number of stages and branches quickly results in an explosion of dimensionality.
- Obviously, the size of (6) can be a limiting factor in solving realistic problems.

Decomposition

- The size of the linear program (6) depends on the number of decision stages and the branching factor at each node of the scenario tree.
- For example, a 4-stage model with 25 branches at each node has $25x25x25x25 = 390625$ scenarios.
- Increasing the number of stages and branches quickly results in an explosion of dimensionality.
- Obviously, the size of (6) can be a limiting factor in solving realistic problems.
- When this occurs, it becomes essential to take advantage of the special structure of the linear program (6).

Decomposition- Bender Decomposition

Bender Decomposition

- Bender Decomposition is a technique in mathematical programming that allows the solution of very large linear programming problems that have a special block structure.

Decomposition- Bender Decomposition

Bender Decomposition

- Bender Decomposition is a technique in mathematical programming that allows the solution of very large linear programming problems that have a special block structure.
- This structure often occurs in applications such as stochastic programming.

Decomposition- Bender Decomposition

Bender Decomposition

- Bender Decomposition is a technique in mathematical programming that allows the solution of very large linear programming problems that have a special block structure.
- This structure often occurs in applications such as stochastic programming.
- The structure that we want to exploit is that of the two-stage problem (5).

Decomposition- Bender Decomposition

Bender Decomposition

- Bender Decomposition is a technique in mathematical programming that allows the solution of very large linear programming problems that have a special block structure.
- This structure often occurs in applications such as stochastic programming.
- The structure that we want to exploit is that of the two-stage problem (5).
- The constraint matrix of (5) has the following form:

Decomposition- Bender Decomposition

Bender Decomposition

- Bender Decomposition is a technique in mathematical programming that allows the solution of very large linear programming problems that have a special block structure.
- This structure often occurs in applications such as stochastic programming.
- The structure that we want to exploit is that of the two-stage problem (5).
- The constraint matrix of (5) has the following form:

$$\begin{pmatrix} A & & & \\ B_1 & C_1 & & \\ \vdots & & \ddots & \\ B_S & & & C_S \end{pmatrix}$$

Decomposition- Bender Decomposition

- Note that the blocks $C_1, \cdots, C_S$ of the constraint matrix are only interrelated through the blocks $B_1, \cdots, B_S$ which correspond to the first-stage decisions.

Decomposition- Bender Decomposition

- Note that the blocks $C_1, \cdots, C_S$ of the constraint matrix are only interrelated through the blocks $B_1, \cdots, B_S$ which correspond to the first-stage decisions.
- In other words, once the first-stage decisions x have been fixed,

Decomposition- Bender Decomposition

- Note that the blocks $C_1, \cdots, C_S$ of the constraint matrix are only interrelated through the blocks $B_1, \cdots, B_S$ which correspond to the first-stage decisions.
- In other words, once the first-stage decisions x have been fixed, (5) decomposes into S independent linear programs.

Decomposition- Bender Decomposition

- Note that the blocks $C_1, \cdots, C_S$ of the constraint matrix are only interrelated through the blocks $B_1, \cdots, B_S$ which correspond to the first-stage decisions.
- In other words, once the first-stage decisions x have been fixed, (5) decomposes into S independent linear programs.
- The idea of Benders decomposition is to solve:

Decomposition- Bender Decomposition

- Note that the blocks $C_1, \cdots, C_S$ of the constraint matrix are only interrelated through the blocks $B_1, \cdots, B_S$ which correspond to the first-stage decisions.
- In other words, once the first-stage decisions x have been fixed, (5) decomposes into S independent linear programs.
- The idea of Benders decomposition is to solve:
  - A *"master problem"* involving only the variables x

Decomposition- Bender Decomposition

- Note that the blocks $C_1, \cdots, C_S$ of the constraint matrix are only interrelated through the blocks $B_1, \cdots, B_S$ which correspond to the first-stage decisions.
- In other words, once the first-stage decisions x have been fixed, (5) decomposes into S independent linear programs.
- The idea of Benders decomposition is to solve:
  - A *"master problem"* involving only the variables x
  - A series of independent *"recourse problems"* each involving a different vector of variables y_k.

Decomposition- Bender Decomposition

- Note that the blocks $C_1, \cdots, C_S$ of the constraint matrix are only interrelated through the blocks $B_1, \cdots, B_S$ which correspond to the first-stage decisions.
- In other words, once the first-stage decisions x have been fixed, (5) decomposes into S independent linear programs.
- The idea of Benders decomposition is to solve:
  - A *"master problem"* involving only the variables x
  - A series of independent *"recourse problems"* each involving a different vector of variables y_k.
- The master problem and recourse problems are linear programs.

Decomposition- Bender Decomposition

- Note that the blocks $C_1, \cdots, C_S$ of the constraint matrix are only interrelated through the blocks $B_1, \cdots, B_S$ which correspond to the first-stage decisions.
- In other words, once the first-stage decisions x have been fixed, (5) decomposes into S independent linear programs.
- The idea of Benders decomposition is to solve:
  - A *"master problem"* involving only the variables x
  - A series of independent *"recourse problems"* each involving a different vector of variables y_k.
- The master problem and recourse problems are linear programs.
- The size of these linear programs is much smaller than the size of full model (5).

Decomposition- Bender Decomposition

- The recourse problems are solved for a given vector x and their solutions are used to generate inequalities that are added to the master problem.

Decomposition- Bender Decomposition

- The recourse problems are solved for a given vector x and their solutions are used to generate inequalities that are added to the master problem.
- Solving the new master problem produces a new x and the process is repeated.

Decomposition- Bender Decomposition

- The recourse problems are solved for a given vector x and their solutions are used to generate inequalities that are added to the master problem.
- Solving the new master problem produces a new x and the process is repeated.
- Let us write (5) as:

$$\max_x a^T x + P_1(x) + \cdots + P_S(x) \qquad (7)$$

Decomposition- Bender Decomposition

- The recourse problems are solved for a given vector x and their solutions are used to generate inequalities that are added to the master problem.
- Solving the new master problem produces a new x and the process is repeated.
- Let us write (5) as:

$$\max_x a^T x + P_1(x) + \cdots + P_S(x) \qquad (7)$$
$$Ax = b$$

Decomposition- Bender Decomposition

- The recourse problems are solved for a given vector x and their solutions are used to generate inequalities that are added to the master problem.
- Solving the new master problem produces a new x and the process is repeated.
- Let us write (5) as:

$$\max_x a^T x + P_1(x) + \cdots + P_S(x) \qquad (7)$$
$$Ax = b$$
$$x \geq 0$$

Decomposition- Bender Decomposition

- For $k = 1, \cdots, S$:

Decomposition- Bender Decomposition

- For $k = 1, \cdots, S$:

$$P_k(x) = \max_{y_k} p_k c_k^T y_k \qquad (8)$$

Decomposition- Bender Decomposition

- For $k = 1, \cdots, S$:

$$P_k(x) = \max_{y_k} p_k c_k^T y_k \qquad (8)$$
$$C_k y_k = d_k - B_k x$$

Decomposition- Bender Decomposition

- For $k = 1, \cdots, S$:

$$P_k(x) = \max_{y_k} p_k c_k^T y_k \qquad (8)$$
$$C_k y_k = d_k - B_k x$$
$$y_k \geq 0$$

Decomposition- Bender Decomposition

- The dual linear program of the recourse problem (8) is:

Decomposition- Bender Decomposition

- The dual linear program of the recourse problem (8) is:

$$P_k(x) = \min_{u_k} u_k^T (d_k - B_k x) \qquad (9)$$

Decomposition- Bender Decomposition

- The dual linear program of the recourse problem (8) is:

$$P_k(x) = \min_{u_k} u_k^T (d_k - B_k x) \qquad (9)$$

$$C_k^T u_k \geq p_k c_k$$

Decomposition- Bender Decomposition

- The dual linear program of the recourse problem (8) is:

$$P_k(x) = \min_{u_k} u_k^T (d_k - B_k x) \qquad (9)$$
$$C_k^T u_k \geq p_k c_k$$

- We assume that the dual (9) is feasible, which is the case of interest in applications.

Decomposition- Bender Decomposition

- The dual linear program of the recourse problem (8) is:

$$P_k(x) = \min_{u_k} u_k^T(d_k - B_k x) \qquad (9)$$
$$C_k^T u_k \geq p_k c_k$$

- We assume that the dual (9) is feasible, which is the case of interest in applications.
- The recourse linear program (8) will be solved for a sequence of vectors x^i.

Decomposition- Bender Decomposition

- The dual linear program of the recourse problem (8) is:

$$P_k(x) = \min_{u_k} u_k^T(d_k - B_k x) \qquad (9)$$

$$C_k^T u_k \geq p_k c_k$$

- We assume that the dual (9) is feasible, which is the case of interest in applications.
- The recourse linear program (8) will be solved for a sequence of vectors x^i.

Decomposition- Bender Decomposition

- The initial vector x^0 might be obtained by solving:

Decomposition- Bender Decomposition

- The initial vector x^0 might be obtained by solving:

$$\max_x a^T x \qquad (10)$$

Decomposition- Bender Decomposition

- The initial vector x^0 might be obtained by solving:

$$\max_x a^T x \qquad (10)$$
$$Ax = b,$$

Decomposition- Bender Decomposition

- The initial vector x^0 might be obtained by solving:

$$\max_x a^T x \qquad (10)$$
$$Ax = b, \qquad x \geq 0$$

- For a given vector x^i, two possibilities can occur for the recourse linear program (8):

Decomposition- Bender Decomposition

- The initial vector x^0 might be obtained by solving:

$$\max_x a^T x \qquad (10)$$
$$Ax = b, \qquad x \geq 0$$

- For a given vector x^i, two possibilities can occur for the recourse linear program (8): either (8) has an optimal solution

Decomposition- Bender Decomposition

- The initial vector x^0 might be obtained by solving:

$$\max_x a^T x \qquad (10)$$
$$Ax = b, \qquad x \geq 0$$

- For a given vector x^i, two possibilities can occur for the recourse linear program (8): either (8) has an optimal solution or it is infeasible.

Decomposition- Bender Decomposition

- If (8) has an optimal solution y_k^i, and u_k^i is the corresponding optimal dual solution,

Decomposition- Bender Decomposition

- If (8) has an optimal solution y_k^i, and u_k^i is the corresponding optimal dual solution, then (9) implies that

Decomposition- Bender Decomposition

- If (8) has an optimal solution y_k^i, and u_k^i is the corresponding optimal dual solution, then (9) implies that

$$P_k(x^i) = (u_k^i)^T(d_k - B_k x^i)$$

Decomposition- Bender Decomposition

- If (8) has an optimal solution y_k^i, and u_k^i is the corresponding optimal dual solution, then (9) implies that

$$P_k(x^i) = (u_k^i)^T(d_k - B_k x^i)$$

Since

$$P_k(x) \leq (u_k^i)^T(d_k - B_k x)$$

Decomposition- Bender Decomposition

- If (8) has an optimal solution y_k^i, and u_k^i is the corresponding optimal dual solution, then (9) implies that

$$P_k(x^i) = (u_k^i)^T(d_k - B_k x^i)$$

Since

$$P_k(x) \leq (u_k^i)^T(d_k - B_k x)$$

We get that

$$P_k(x) \leq (u_k^i)^T(d_k - B_k x^i) + P_k(x^i)$$

Decomposition- Bender Decomposition

- If (8) has an optimal solution y_k^i, and u_k^i is the corresponding optimal dual solution, then (9) implies that

$$P_k(x^i) = (u_k^i)^T(d_k - B_k x^i)$$

Since

$$P_k(x) \leq (u_k^i)^T(d_k - B_k x)$$

We get that

$$P_k(x) \leq (u_k^i)^T(d_k - B_k x^i) + P_k(x^i)$$

This inequality, which is called an *optimality cut*, can be added to the current master linear program.

Decomposition- Bender Decomposition

- If (8) has an optimal solution y_k^i, and u_k^i is the corresponding optimal dual solution, then (9) implies that

$$P_k(x^i) = (u_k^i)^T(d_k - B_k x^i)$$

Since

$$P_k(x) \leq (u_k^i)^T(d_k - B_k x)$$

We get that

$$P_k(x) \leq (u_k^i)^T(d_k - B_k x^i) + P_k(x^i)$$

This inequality, which is called an *optimality cut*, can be added to the current master linear program. Initially, the master linear program is just (10).

Decomposition- Bender Decomposition

- If (8) is infeasible,

Decomposition- Bender Decomposition

- If (8) is infeasible, then the dual problem is unbounded.

Decomposition- Bender Decomposition

- If (8) is infeasible, then the dual problem is unbounded. Let u_k^i a direction where (9) is unbounded

Decomposition- Bender Decomposition

- If (8) is infeasible, then the dual problem is unbounded. Let u_k^i a direction where (9) is unbounded

$$(u_k^i)^T(d_k - B_k x^i) < 0$$
$$C_k^T u_k^i \geq p_k c_k$$

Decomposition- Bender Decomposition

- If (8) is infeasible, then the dual problem is unbounded. Let u_k^i a direction where (9) is unbounded

$$(u_k^i)^T(d_k - B_k x^i) < 0$$
$$C_k^T u_k^i \geq p_k c_k$$

Since we are only interested in first-stage decisions x that lead to feasible second-stage decisions y_k,

Decomposition- Bender Decomposition

- If (8) is infeasible, then the dual problem is unbounded. Let u_k^i a direction where (9) is unbounded

$$(u_k^i)^T(d_k - B_k x^i) < 0$$
$$C_k^T u_k^i \geq p_k c_k$$

Since we are only interested in first-stage decisions x that lead to feasible second-stage decisions y_k, the following *feasibility cut* can be added to the current master linear program:

Decomposition- Bender Decomposition

- If (8) is infeasible, then the dual problem is unbounded. Let u_k^i a direction where (9) is unbounded

$$(u_k^i)^T(d_k - B_k x^i) < 0$$
$$C_k^T u_k^i \geq p_k c_k$$

Since we are only interested in first-stage decisions x that lead to feasible second-stage decisions y_k, the following *feasibility cut* can be added to the current master linear program:

$$(u_k^i)^T(d_k - B_k x) \geq 0$$

Decomposition- Bender Decomposition

- After solving the recourse problems (8) for each k,

Decomposition- Bender Decomposition

- After solving the recourse problems (8) for each k, we have the following lower bound on the optimal value of (5):

Decomposition- Bender Decomposition

- After solving the recourse problems (8) for each k, we have the following lower bound on the optimal value of (5):

$$LB = a^T x^i + P_1(x^i) + \cdots + P_S(x^i)$$

Decomposition- Bender Decomposition

- After solving the recourse problems (8) for each k, we have the following lower bound on the optimal value of (5):

$$LB = a^T x^i + P_1(x^i) + \cdots + P_S(x^i)$$

where we set $P_k(x^i) = -\infty$ if the corresponding recourse problem is infeasible.

Decomposition- Bender Decomposition

- After solving the recourse problems (8) for each k, we have the following lower bound on the optimal value of (5):

$$LB = a^T x^i + P_1(x^i) + \cdots + P_S(x^i)$$

where we set $P_k(x^i) = -\infty$ if the corresponding recourse problem is infeasible.

- Adding all the optimality and feasibility cuts found so far (for $j = 0, \cdots, i$) to the master linear program, we obtain:

Decomposition- Bender Decomposition

- After solving the recourse problems (8) for each k, we have the following lower bound on the optimal value of (5):

$$LB = a^T x^i + P_1(x^i) + \cdots + P_S(x^i)$$

where we set $P_k(x^i) = -\infty$ if the corresponding recourse problem is infeasible.

- Adding all the optimality and feasibility cuts found so far (for $j = 0, \cdots, i$) to the master linear program, we obtain:

$$\max_{x, z_1, \cdots, z_S} a^T x + \sum_{k=1}^{S} z_k$$

Decomposition- Bender Decomposition

- After solving the recourse problems (8) for each k, we have the following lower bound on the optimal value of (5):

$$LB = a^T x^i + P_1(x^i) + \cdots + P_S(x^i)$$

where we set $P_k(x^i) = -\infty$ if the corresponding recourse problem is infeasible.

- Adding all the optimality and feasibility cuts found so far (for $j = 0, \cdots, i$) to the master linear program, we obtain:

$$\begin{array}{l} \max_{x,z_1,\cdots,z_S} a^T x + \sum_{k=1}^{S} z_k \\ Ax = b \end{array}$$

Decomposition- Bender Decomposition

- After solving the recourse problems (8) for each k, we have the following lower bound on the optimal value of (5):

$$LB = a^T x^i + P_1(x^i) + \cdots + P_S(x^i)$$

where we set $P_k(x^i) = -\infty$ if the corresponding recourse problem is infeasible.

- Adding all the optimality and feasibility cuts found so far (for $j = 0, \cdots, i$) to the master linear program, we obtain:

$$\begin{aligned} &\max_{x, z_1, \cdots, z_S} a^T x + \textstyle\sum_{k=1}^S z_k \\ &Ax = b \\ &z_k \le (u_k^j)^T (B_k x^j - B_k x) + p_k(x^j) \qquad \text{for some pairs } (j,k) \end{aligned}$$

Decomposition- Bender Decomposition

- After solving the recourse problems (8) for each k, we have the following lower bound on the optimal value of (5):

$$LB = a^T x^i + P_1(x^i) + \cdots + P_S(x^i)$$

where we set $P_k(x^i) = -\infty$ if the corresponding recourse problem is infeasible.

- Adding all the optimality and feasibility cuts found so far (for $j = 0, \cdots, i$) to the master linear program, we obtain:

$$\begin{array}{ll} \max_{x,z_1,\cdots,z_S} a^T x + \sum_{k=1}^S z_k & \\ Ax = b & \\ z_k \le (u_k^j)^T (B_k x^j - B_k x) + p_k(x^j) & \text{for some pairs } (j,k) \\ 0 \le (u_k^j)^T (d_k - B_k x) & \text{for the remaining pairs } (j,k) \end{array}$$

Decomposition- Bender Decomposition

- After solving the recourse problems (8) for each k, we have the following lower bound on the optimal value of (5):

$$LB = a^T x^i + P_1(x^i) + \cdots + P_S(x^i)$$

where we set $P_k(x^i) = -\infty$ if the corresponding recourse problem is infeasible.

- Adding all the optimality and feasibility cuts found so far (for $j = 0, \cdots, i$) to the master linear program, we obtain:

$$\begin{array}{ll}
\max_{x, z_1, \cdots, z_S} a^T x + \sum_{k=1}^{S} z_k & \\
Ax = b & \\
z_k \leq (u_k^j)^T (B_k x^j - B_k x) + p_k(x^j) & \text{for some pairs } (j, k) \\
0 \leq (u_k^j)^T (d_k - B_k x) & \text{for the remaining pairs } (j, k) \\
x \geq 0 &
\end{array}$$

Decomposition- Bender Decomposition

- Denoting by $x^{i+1}, z_1^{i+1}, \cdots, z_S^{i+1}$ an optimal solution to this linear program,

Decomposition- Bender Decomposition

- Denoting by $x^{i+1}, z_1^{i+1}, \cdots, z_S^{i+1}$ an optimal solution to this linear program, we get an upper bound on the optimal value of (5):

Decomposition- Bender Decomposition

- Denoting by $x^{i+1}, z_1^{i+1}, \cdots, z_S^{i+1}$ an optimal solution to this linear program, we get an upper bound on the optimal value of (5):

$$UB = a^T x^{i+1} + z_1^{i+1} + \cdots + z_S^{i+1}$$

- Benders decomposition alternately solves the recourse problems (8)

Decomposition- Bender Decomposition

- Denoting by $x^{i+1}, z_1^{i+1}, \cdots, z_S^{i+1}$ an optimal solution to this linear program, we get an upper bound on the optimal value of (5):

$$UB = a^T x^{i+1} + z_1^{i+1} + \cdots + z_S^{i+1}$$

- Benders decomposition alternately solves the recourse problems (8)
- The master linear program with new optimality and feasibility cuts added at each iteration

Decomposition- Bender Decomposition

- Denoting by $x^{i+1}, z_1^{i+1}, \cdots, z_S^{i+1}$ an optimal solution to this linear program, we get an upper bound on the optimal value of (5):

$$UB = a^T x^{i+1} + z_1^{i+1} + \cdots + z_S^{i+1}$$

- Benders decomposition alternately solves the recourse problems (8)
- The master linear program with new optimality and feasibility cuts added at each iteration **until** the gap between the upper bound UB and the lower bound LB falls below a given threshold.

Decomposition- Bender Decomposition

Benders decomposition can also be used for multi-stage problems (6) in a straightforward way.

Decomposition- Bender Decomposition

Benders decomposition can also be used for multi-stage problems (6) in a straightforward way.

The stages are partitioned into a first set that gives rise to the ”master problem”

Decomposition- Bender Decomposition

Benders decomposition can also be used for multi-stage problems (6) in a straightforward way.

The stages are partitioned into a first set that gives rise to the ”master problem” and a second set that gives rise to the ”recourse problems”.

Decomposition- Bender Decomposition

Benders decomposition can also be used for multi-stage problems (6) in a straightforward way.

The stages are partitioned into a first set that gives rise to the ”master problem” and a second set that gives rise to the ”recourse problems”.

For example in a 6-stage problem,

Decomposition- Bender Decomposition

Benders decomposition can also be used for multi-stage problems (6) in a straightforward way.

The stages are partitioned into a first set that gives rise to the ”master problem” and a second set that gives rise to the ”recourse problems”.

For example in a 6-stage problem, the variables of the first 2 stages could define the master problem.

Decomposition- Bender Decomposition

Benders decomposition can also be used for multi-stage problems (6) in a straightforward way.

The stages are partitioned into a first set that gives rise to the ”master problem” and a second set that gives rise to the ”recourse problems”.

For example in a 6-stage problem, the variables of the first 2 stages could define the master problem.

When these variables are fixed,

Decomposition- Bender Decomposition

Benders decomposition can also be used for multi-stage problems (6) in a straightforward way.

The stages are partitioned into a first set that gives rise to the ”master problem” and a second set that gives rise to the ”recourse problems”.

For example in a 6-stage problem, the variables of the first 2 stages could define the master problem.

When these variables are fixed, (6) decomposes into separate linear programs each involving variables of the last 4 stages.

Decomposition- Bender Decomposition

Benders decomposition can also be used for multi-stage problems (6) in a straightforward way.

The stages are partitioned into a first set that gives rise to the ”master problem” and a second set that gives rise to the ”recourse problems”.

For example in a 6-stage problem, the variables of the first 2 stages could define the master problem.

When these variables are fixed, (6) decomposes into separate linear programs each involving variables of the last 4 stages.

The solutions of these recourse linear program provide optimality or feasibility cuts that can be added to the master problem.

Decomposition- Bender Decomposition

Benders decomposition can also be used for multi-stage problems (6) in a straightforward way.

The stages are partitioned into a first set that gives rise to the ”master problem” and a second set that gives rise to the ”recourse problems”.

For example in a 6-stage problem, the variables of the first 2 stages could define the master problem.

When these variables are fixed, (6) decomposes into separate linear programs each involving variables of the last 4 stages.

The solutions of these recourse linear program provide optimality or feasibility cuts that can be added to the master problem.

As before, upper and lower bounds are computed at each iteration and the algorithm stops when the difference drops below a given tolerance.

Scenario Generation

Scenario Generation

How should one generate scenarios in order to formulate a deterministic equivalent formulation (6) that accurately represents the underlying stochastic program?

Scenario Generation

How should one generate scenarios in order to formulate a deterministic equivalent formulation (6) that accurately represents the underlying stochastic program?

- First, one needs to model the correlation over time among the random parameters.
    - For a pension fund, such a model might relate wage inflation (which influences the liability side) to interest rates and stock prices (which influence the asset side).

Scenario Generation

How should one generate scenarios in order to formulate a deterministic equivalent formulation (6) that accurately represents the underlying stochastic program?

- First, one needs to model the correlation over time among the random parameters.
    - For a pension fund, such a model might relate wage inflation (which influences the liability side) to interest rates and stock prices (which influence the asset side).
    - Simpler autoregressive models can be used.

Scenario Generation

How should one generate scenarios in order to formulate a deterministic equivalent formulation (6) that accurately represents the underlying stochastic program?

- First, one needs to model the correlation over time among the random parameters.
  - For a pension fund, such a model might relate wage inflation (which influences the liability side) to interest rates and stock prices (which influence the asset side).
  - Simpler autoregressive models can be used.
- The second issue is the construction of a scenario tree from these models:

Scenario Generation

How should one generate scenarios in order to formulate a deterministic equivalent formulation (6) that accurately represents the underlying stochastic program?

- First, one needs to model the correlation over time among the random parameters.
  - For a pension fund, such a model might relate wage inflation (which influences the liability side) to interest rates and stock prices (which influence the asset side).
  - Simpler autoregressive models can be used.
- The second issue is the construction of a scenario tree from these models:
  - A finite number of scenarios must reflect as accurately as possible the random processes modeled in the previous step.
  - Random Sampling and Tree Fitting are used.

Scenario Generation

How should one generate scenarios in order to formulate a deterministic equivalent formulation (6) that accurately represents the underlying stochastic program?

- First, one needs to model the correlation over time among the random parameters.
  - For a pension fund, such a model might relate wage inflation (which influences the liability side) to interest rates and stock prices (which influence the asset side).
  - Simpler autoregressive models can be used.
- The second issue is the construction of a scenario tree from these models:
  - A finite number of scenarios must reflect as accurately as possible the random processes modeled in the previous step.
  - Random Sampling and Tree Fitting are used.
- On the other hand, the linear program (6) can only be solved if the size of the scenario tree is reasonably small, suggesting a rather limited number of scenarios.

References

References

References

References

References

1. G. Cornuejols, R. Tütüncü, Optimization methods in Finance, Cambridge University Press, 2007.

References

References

1. G. Cornuejols, R. Tütüncü, Optimization methods in Finance, Cambridge University Press, 2007.
2. A Shapiro, D Dentcheva, and A Ruszczynski, Lectures on Stochastic Programming: Modeling and Theory, SIAM, 2009.

References

References

1. G. Cornuejols, R. Tütüncü, Optimization methods in Finance, Cambridge University Press, 2007.
2. A Shapiro, D Dentcheva, and A Ruszczynski, Lectures on Stochastic Programming: Modeling and Theory, SIAM, 2009.
3. J E Beasley, OR-Notes: Stochastic Programming, http://people.brunel.ac.uk/ mastjjb/jeb/or/sp.html

References

References

1. G. Cornuejols, R. Tütüncü, Optimization methods in Finance, Cambridge University Press, 2007.
2. A Shapiro, D Dentcheva, and A Ruszczynski, Lectures on Stochastic Programming: Modeling and Theory, SIAM, 2009.
3. J E Beasley, OR-Notes: Stochastic Programming, http://people.brunel.ac.uk/ mastjjb/jeb/or/sp.html
4. Stochastic Programming, http://disopt.epfl.ch/cms/page-10544.html

References

References

1. G. Cornuejols, R. Tütüncü, Optimization methods in Finance, Cambridge University Press, 2007.
2. A Shapiro, D Dentcheva, and A Ruszczynski, Lectures on Stochastic Programming: Modeling and Theory, SIAM, 2009.
3. J E Beasley, OR-Notes: Stochastic Programming, http://people.brunel.ac.uk/ mastjjb/jeb/or/sp.html
4. Stochastic Programming, http://disopt.epfl.ch/cms/page-10544.html