Introduction to Stochastic Programming

PhD Course

Patrizia Beraldi
DIMEG - UNICAL

patrizia.beraldi@unical.it

Introduction

Introduction

- Multistage stochastic programming formulations arise naturally as a generalization of two-stage models

Introduction

- Multistage stochastic programming formulations arise naturally as a generalization of two-stage models
- In the two-stage case, the decision process can be represented as follows

$$x \xrightarrow{\omega} y(\omega)$$

Introduction

- Multistage stochastic programming formulations arise naturally as a generalization of two-stage models
- In the two-stage case, the decision process can be represented as follows

$$x \xrightarrow{\omega} y(\omega)$$

- In the multi-stage case, we consider a planning horizon divided in a given number of stages $t = 1, \ldots, T$

Introduction

- Multistage stochastic programming formulations arise naturally as a generalization of two-stage models
- In the two-stage case, the decision process can be represented as follows

$$x \xrightarrow{\omega} y(\omega)$$

- In the multi-stage case, we consider a planning horizon divided in a given number of stages $t = 1, \dots, T$
- Thus, the decision process can be represented as

$$x_1 \quad \xrightarrow{\omega_1} x_2 \dots\dots x_t \quad \xrightarrow{\omega_t} \quad x_{t+1} \xrightarrow{\omega_{t+2}} \quad x_{t+2} \dots\dots$$

Introduction

- Multistage stochastic programming formulations arise naturally as a generalization of two-stage models
- In the two-stage case, the decision process can be represented as follows

$$x \xrightarrow{\omega} y(\omega)$$

- In the multi-stage case, we consider a planning horizon divided in a given number of stages $t = 1, \dots, T$
- Thus, the decision process can be represented as

$$x_1 \quad \xrightarrow{\omega_1} x_2 \dots\dots x_t \quad \xrightarrow{\omega_t} \quad x_{t+1} \xrightarrow{\omega_{t+2}} \quad x_{t+2} \dots\dots$$

  - $x_t \in \mathbb{R}^{n_t}$ denotes the decisions taken at stage t

Introduction

- Multistage stochastic programming formulations arise naturally as a generalization of two-stage models
- In the two-stage case, the decision process can be represented as follows

$$x \xrightarrow{\omega} y(\omega)$$

- In the multi-stage case, we consider a planning horizon divided in a given number of stages $t = 1, \ldots, T$
- Thus, the decision process can be represented as

$$x_1 \quad \xrightarrow{\omega_1} x_2 \ldots \ldots x_t \quad \xrightarrow{\omega_t} \quad x_{t+1} \xrightarrow{\omega_{t+2}} \quad x_{t+2} \ldots \ldots$$

  - $x_t \in \mathbb{R}^{n_t}$ denotes the decisions taken at stage t
  - ω_t represents the uncertainty whose realizations become known at time t

Remarks

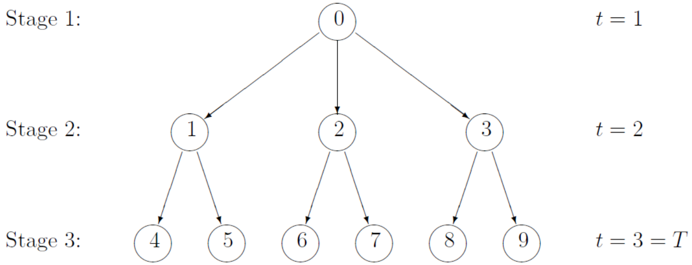
Remarks

- **stage** is a moment in time, when decisions are taken
- **time period** is a time interval between two stages

Remarks

- **stage** is a moment in time, when decisions are taken
- **time period** is a time interval between two stages
- The multi-stage models can be introduced by considering alternative formulations
- We shall start by introducing the most intuitive one, where the evolution of the uncertain parameters is represented by a scenario tree

Scenario tree for a three-stage problem



The scenario tree: notation

- The root node 0 is associated with the first stage and refers to deterministic data

The scenario tree: notation

- The root node 0 is associated with the first stage and refers to deterministic data
- Each node n at level $t \geq 2$ represents a possible realization of the random event ω_t

The scenario tree: notation

- The root node 0 is associated with the first stage and refers to deterministic data
- Each node n at level $t \geq 2$ represents a possible realization of the random event ω_t
- Each node n at level t has a unique ancestor (father) at level $t-1$ denoted by $a(n)$ and certain number of successors (children) at level $t+1$

The scenario tree: notation

- The root node 0 is associated with the first stage and refers to deterministic data
- Each node n at level $t \geq 2$ represents a possible realization of the random event ω_t
- Each node n at level t has a unique ancestor (father) at level $t-1$ denoted by $a(n)$ and certain number of successors (children) at level $t+1$
- Nodes with no children are called **leaf nodes**

The scenario tree: notation

- The root node 0 is associated with the first stage and refers to deterministic data
- Each node n at level $t \geq 2$ represents a possible realization of the random event ω_t
- Each node n at level t has a unique ancestor (father) at level $t-1$ denoted by $a(n)$ and certain number of successors (children) at level $t+1$
- Nodes with no children are called **leaf nodes**
- There is a correspondence between the leaf nodes and the scenarios

The scenario tree: notation

- The root node 0 is associated with the first stage and refers to deterministic data
- Each node n at level $t \geq 2$ represents a possible realization of the random event ω_t
- Each node n at level t has a unique ancestor (father) at level $t-1$ denoted by $a(n)$ and certain number of successors (children) at level $t+1$
- Nodes with no children are called **leaf nodes**
- There is a correspondence between the leaf nodes and the scenarios
- A **scenario** is path from the root node to a leaf node, i.e. it is a joint realization of the random parameters over all the time stages

The scenario tree: notation

The scenario tree: notation

- We denote by $\mathcal{N}$ the set of nodes in the scenario tree

The scenario tree: notation

- We denote by $\mathcal{N}$ the set of nodes in the scenario tree
- Let l be the ancestor of node n ($l = a(n)$)

The scenario tree: notation

- We denote by $\mathcal{N}$ the set of nodes in the scenario tree
- Let l be the ancestor of node n ($l = a(n)$)
- Let ρ_{ln} be the probability of moving from node l to node n

The scenario tree: notation

- We denote by $\mathcal{N}$ the set of nodes in the scenario tree
- Let l be the ancestor of node n ($l = a(n)$)
- Let ρ_{ln} be the probability of moving from node l to node n
- Remark:

The scenario tree: notation

- We denote by $\mathcal{N}$ the set of nodes in the scenario tree
- Let l be the ancestor of node n ($l = a(n)$)
- Let ρ_{ln} be the probability of moving from node l to node n
- Remark:
  - ρ_{ln} is a conditional probability
  - $\rho_{ln} \geq 0$ and the sum of the ρ_{ln} associated with the children of l should be 1

The scenario tree: notation

- We denote by $\mathcal{N}$ the set of nodes in the scenario tree
- Let l be the ancestor of node n ($l = a(n)$)
- Let ρ_{ln} be the probability of moving from node l to node n
- Remark:
  - ρ_{ln} is a conditional probability
  - $\rho_{ln} \geq 0$ and the sum of the ρ_{ln} associated with the children of l should be 1
- Starting from the ρ_{ln} it is possible to compute the probability associated with each scenario

The scenario tree: notation

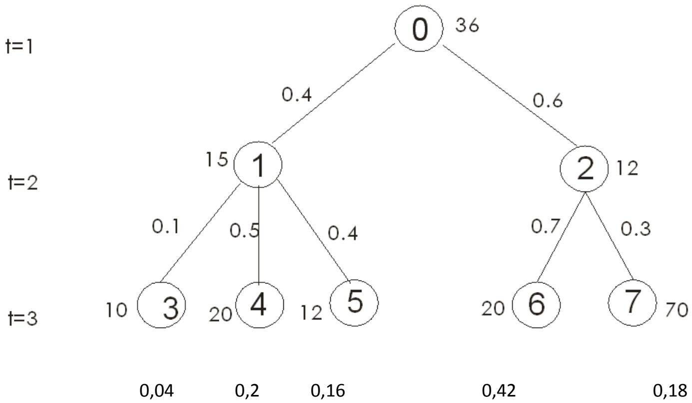
- We denote by $\mathcal{N}$ the set of nodes in the scenario tree
- Let l be the ancestor of node n ($l = a(n)$)
- Let ρ_{ln} be the probability of moving from node l to node n
- Remark:
  - ρ_{ln} is a conditional probability
  - $\rho_{ln} \geq 0$ and the sum of the ρ_{ln} associated with the children of l should be 1
- Starting from the ρ_{ln} it is possible to compute the probability associated with each scenario
- Let n_1, n_2, n_T be the nodes forming the path from the root node to a leaf node

The scenario tree: notation

- We denote by $\mathcal{N}$ the set of nodes in the scenario tree
- Let l be the ancestor of node n ($l = a(n)$)
- Let ρ_{ln} be the probability of moving from node l to node n
- Remark:
  - ρ_{ln} is a conditional probability
  - $\rho_{ln} \geq 0$ and the sum of the ρ_{ln} associated with the children of l should be 1
- Starting from the ρ_{ln} it is possible to compute the probability associated with each scenario
- Let n_1, n_2, n_T be the nodes forming the path from the root node to a leaf node
- The probability associated with the scenario is defined by

$$\rho_{n_1 n_2} * \rho_{n_2 n_3} * \cdots * \rho_{n_{T-1} n_T}$$

The scenario tree: example



Remarks

Remarks

- The branching factor may be arbitrary in principle

Remarks

- The branching factor may be arbitrary in principle
- The more branches we use, the better our ability to model uncertainty

Remarks

- The branching factor may be arbitrary in principle
- The more branches we use, the better our ability to model uncertainty
- Unfortunately, the number of nodes grows exponentially with the number of stages, as well as the computational effort

Remarks

- The branching factor may be arbitrary in principle
- The more branches we use, the better our ability to model uncertainty
- Unfortunately, the number of nodes grows exponentially with the number of stages, as well as the computational effort
- In practice, we are interested in the decisions that must be implemented here and now, i.e., those corresponding to the root node of the tree

Remarks

- The branching factor may be arbitrary in principle
- The more branches we use, the better our ability to model uncertainty
- Unfortunately, the number of nodes grows exponentially with the number of stages, as well as the computational effort
- In practice, we are interested in the decisions that must be implemented here and now, i.e., those corresponding to the root node of the tree
- The other (recourse) decision variables are instrumental to the aim of devising a robust plan, but they are not implemented in practice, as the multistage model is solved on a rolling horizon basis

Remarks

- In order to model the uncertainty as accurately as possible with a limited computational effort, a possible idea is to branch many paths from the root node, and less from the subsequent nodes

Remarks

- In order to model the uncertainty as accurately as possible with a limited computational effort, a possible idea is to branch many paths from the root node, and less from the subsequent nodes
- The design of appropriate scenario generation techniques is currently a subject of intensive research since the quality of scenario tree impacts on the recommendations provided by the solution of the stochastic programming formulations

Remarks

- In order to model the uncertainty as accurately as possible with a limited computational effort, a possible idea is to branch many paths from the root node, and less from the subsequent nodes
- The design of appropriate scenario generation techniques is currently a subject of intensive research since the quality of scenario tree impacts on the recommendations provided by the solution of the stochastic programming formulations
- Given the limited time, we shall not address this important issue in this short course

Multi-stage problems: the compact formulation

Multi-stage problems: the compact formulation

For each node $n \in \mathcal{N}$ we denote by

Multi-stage problems: the compact formulation

For each node $n \in \mathcal{N}$ we denote by

- p_n the node probability

Multi-stage problems: the compact formulation

For each node $n \in \mathcal{N}$ we denote by

- p_n the node probability
- x_n the vector of associated decision variables

Multi-stage problems: the compact formulation

For each node $n \in \mathcal{N}$ we denote by

- p_n the node probability
- x_n the vector of associated decision variables
- c_n, h^n, T^n, W^n the corresponding matrices and vectors

$$\min \qquad z = \sum_{n \in \mathcal{N}} p_n c_n^T x_n$$

Multi-stage problems: the compact formulation

For each node $n \in \mathcal{N}$ we denote by

- p_n the node probability
- x_n the vector of associated decision variables
- c_n, h^n, T^n, W^n the corresponding matrices and vectors

$$\min \quad z = \sum_{n \in \mathcal{N}} p_n c_n^T x_n$$
$$Ax_0 = b$$

Multi-stage problems: the compact formulation

For each node $n \in \mathcal{N}$ we denote by

- p_n the node probability
- x_n the vector of associated decision variables
- c_n, h^n, T^n, W^n the corresponding matrices and vectors

$$
\begin{aligned}
\min \quad & z = \sum_{n \in \mathcal{N}} p_n c_n^T x_n \\
& A x_0 = b \\
& T^n x_{a(n)} + W^n x_n = h^n \quad \forall n \in \mathcal{N} - \{0\}
\end{aligned}
$$

Multi-stage problems: the compact formulation

For each node $n \in \mathcal{N}$ we denote by

- p_n the node probability
- x_n the vector of associated decision variables
- c_n, h^n, T^n, W^n the corresponding matrices and vectors

$$
\begin{aligned}
\min \quad & z = \sum_{n \in \mathcal{N}} p_n c_n^T x_n \\
& A x_0 = b \\
& T^n x_{a(n)} + W^n x_n = h^n \quad \forall n \in \mathcal{N} - \{0\} \\
& x_n \geq 0 \quad \forall n \in \mathcal{N}
\end{aligned}
$$

Multi-stage problems: The split-variable formulation

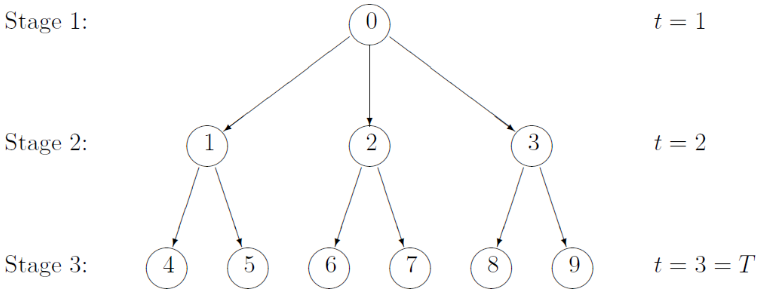
- In the compact formulation, the non-anticipativity constraints are implicitly included

Multi-stage problems: The split-variable formulation

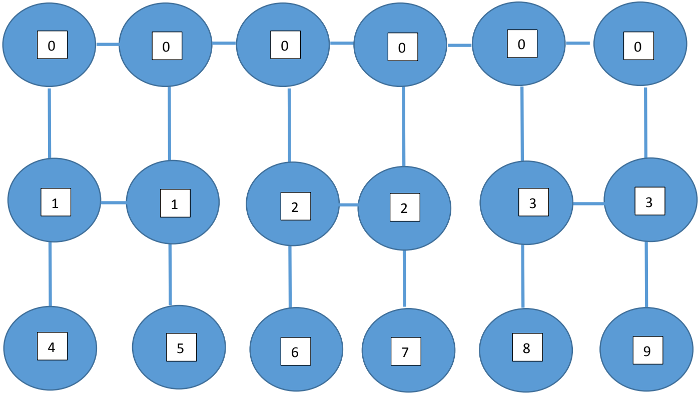
- In the compact formulation, the non-anticipativity constraints are implicitly included
- These constraints state that the decisions associated with two scenarios that share the same history for a given number k of periods, should be identical up to k

- In the compact formulation, the non-anticipativity constraints are implicitly included
- These constraints state that the decisions associated with two scenarios that share the same history for a given number k of periods, should be identical up to k
- The issue may be understood by looking at the following figure, where horizontal lines correspond to non-anticipativity requirements

The original tree



The split tree



The split-variable formulation

The split-variable formulation

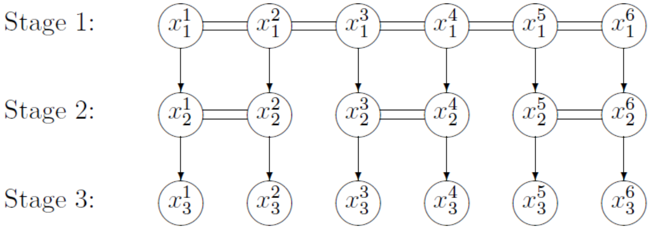
- We make a copy of all variables for each scenario

The split-variable formulation

- We make a copy of all variables for each scenario
- We denote by x_t^s the decisions taken at time t under scenario s
- We add non-anticipativity constraints to force logical identical variables to agree across scenarios

The split-variable formulation

- We make a copy of all variables for each scenario
- We denote by x_t^s the decisions taken at time t under scenario s
- We add non-anticipativity constraints to force logical identical variables to agree across scenarios



The non-anticipativity condition

The non-anticipativity condition

- In our example

$$
\begin{aligned}
x_1^1 &= x_1^2 = \cdots = x_1^6 = x_0 \\
x_2^1 &= x_2^2 = x_1 \\
x_2^3 &= x_2^4 = x_2 \\
x_2^4 &= x_2^5 = x_3
\end{aligned}
$$

The non-anticipativity condition

- In our example

$$
\begin{aligned}
x_1^1 &= x_1^2 = \cdots = x_1^6 = x_0 \\
x_2^1 &= x_2^2 = x_1 \\
x_2^3 &= x_2^4 = x_2 \\
x_2^4 &= x_2^5 = x_3
\end{aligned}
$$

- In general

$$x_{t(n)}^s = x_n \quad \forall n \quad \forall s \in S(n)$$

The non-anticipativity condition

- In our example

$$x_1^1 = x_1^2 = \cdots = x_1^6 = x_0$$
$$x_2^1 = x_2^2 = x_1$$
$$x_2^3 = x_2^4 = x_2$$
$$x_2^4 = x_2^5 = x_3$$

- In general

$$x_{t(n)}^s = x_n \;\; \forall n \quad \forall s \in S(n)$$

  - $t(n)$ stage of node n

The non-anticipativity condition

- In our example

$$
\begin{aligned}
x_1^1 &= x_1^2 = \cdots = x_1^6 = x_0 \\
x_2^1 &= x_2^2 = x_1 \\
x_2^3 &= x_2^4 = x_2 \\
x_2^4 &= x_2^5 = x_3
\end{aligned}
$$

- In general

$$x_{t(n)}^s = x_n \quad \forall n \quad \forall s \in S(n)$$

  - $t(n)$ stage of node n
  - $S(n)$ set of scenarios passing through node n

$$
\begin{aligned}
S(0) &= \{1, 2, 3, 4, 5, 6\} \\
S(1) &= \{1, 2\} \\
S(2) &= \{3, 4\} \\
S(3) &= \{5, 6\}
\end{aligned}
$$

The split-variable formulation

$$
\begin{aligned}
\min \quad & z = \sum_{s=1}^{S} p_s \sum_{t=1}^{T} c_t^s x_t^s \\
& A x_1^s = b \quad s = 1, \dots S \\
& T_t^s x_{t-1}^s + W_t^s x_t^s = h_t^s \;\; \forall t \geq 2, s = 1, \dots S \\
& x_n - x_{t(n)}^s = 0 \quad \forall n \in \mathcal{N} \quad \forall s \in S(n) \\
& x_t^s \geq 0 \quad t = 1, \dots T \quad s = 1, \dots, S
\end{aligned}
$$

Stochastic Programming in Finance

Stochastic Programming in Finance

- Finance represents one of the main application domains where the SP framework has been widely used to model real-life problems

Stochastic Programming in Finance

- Finance represents one of the main application domains where the SP framework has been widely used to model real-life problems
- The main motivation is related to high volatility of financial markets that makes the explicit inclusion of the uncertainty in the mathematical model mandatory to derive financial plans that can be used in practice

- Finance represents one of the main application domains where the SP framework has been widely used to model real-life problems
- The main motivation is related to high volatility of financial markets that makes the explicit inclusion of the uncertainty in the mathematical model mandatory to derive financial plans that can be used in practice
- In particular, the multistage framework offers a valuable paradigm allowing to define optimal investment plans that can revised over the time as new information becomes available

A multi-stage model for Portfolio Optimization

- We consider a planning horizon divided into a number of elementary periods $t = 1, 2, \dots T$
- At each period t of the planning horizon, the investor must decide:
  - The amount of security i to be purchased B_{it}
  - The amount of security i to sell S_{it}
  - The amount of security i to be maintained in the portfolio H_{it}
  - The monetary amount to invest in a risk-free asset v_t

The deterministic model

- **Physical balance constraints** $(i = 1, \ldots N)$

The deterministic model

- **Physical balance constraints** $(i = 1, \dots N)$

$$H_{it} = H_{it-1} + B_{it} - S_{it} \qquad \text{t=2,...,T-1}$$

The deterministic model

- **Physical balance constraints** $(i = 1, \dots N)$

$$H_{it} = H_{it-1} + B_{it} - S_{it} \qquad \text{t=2,...,T-1}$$
$$H_{i1} = InitHold_i + B_{i1} - S_{i1} \quad \text{t=1}$$

The deterministic model

- **Physical balance constraints** $\quad (i = 1, \ldots N)$

$$H_{it} = H_{it-1} + B_{it} - S_{it} \qquad \text{t=2,...,T-1}$$
$$H_{i1} = InitHold_i + B_{i1} - S_{i1} \quad \text{t=1}$$
$$(S_{iT} = H_{1T-1})$$

- **Monetary balance constraints**

The deterministic model

- **Physical balance constraints** $(i = 1, \ldots N)$

$$H_{it} = H_{it-1} + B_{it} - S_{it} \qquad \text{t=2,...,T-1}$$
$$H_{i1} = InitHold_i + B_{i1} - S_{i1} \quad \text{t=1}$$
$$(S_{iT} = H_{1T-1})$$

- **Monetary balance constraints**

$$(1-g)\sum_{i=1}^{N} P_{it}S_{it} + F_t + (1+r_t)v_{t-1} =$$
$$(1+g)\sum_{i=1}^{N} P_{it}B_{it} + L_t + v_t \qquad t = 2, \ldots, T-1$$

The deterministic model

- **Physical balance constraints** $(i = 1, \dots N)$

$$H_{it} = H_{it-1} + B_{it} - S_{it} \qquad \text{t=2,...,T-1}$$
$$H_{i1} = InitHold_i + B_{i1} - S_{i1} \quad \text{t=1}$$
$$(S_{iT} = H_{1T-1})$$

- **Monetary balance constraints**

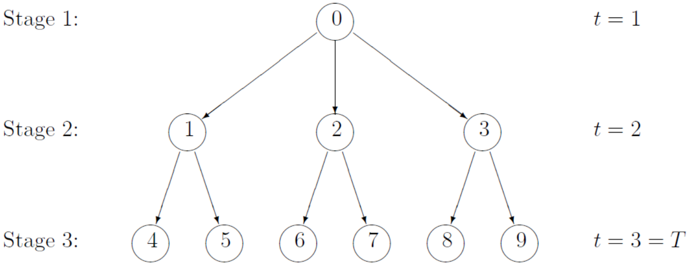
$$(1-g)\sum_{i=1}^{N} P_{it}S_{it} + F_t + (1+r_t)v_{t-1} =$$
$$(1+g)\sum_{i=1}^{N} P_{it}B_{it} + L_t + v_t \qquad t = 2, \dots, T-1$$
$$(1-g)\sum_{i=1}^{N} P_{i1}S_{i1} + F_1 + = (1+g)\sum_{i=1}^{N} P_{i1}B_{i1} + L_1 + v_1$$

The deterministic model

- **The objective function**

$$max \quad W_T = (1-g)\sum_{i=1}^{N} P_{iT} H_{iT-1} + (1+r_T) v_{T-1} + F_T - L_T$$

Scenario tree for a three-stage problem



The multi-stage node formulation

The multi-stage node formulation

- In this case the decision variables are associated with the nodes of the scenario tree which we use to represent the dynamic evolution of the uncertain parameters

The multi-stage node formulation

- In this case the decision variables are associated with the nodes of the scenario tree which we use to represent the dynamic evolution of the uncertain parameters
- Let $\mathcal{N}$ denote the set of nodes of the scenario tree

The multi-stage node formulation

- In this case the decision variables are associated with the nodes of the scenario tree which we use to represent the dynamic evolution of the uncertain parameters
- Let $\mathcal{N}$ denote the set of nodes of the scenario tree
- For each node $n \in \mathcal{N}$ we denote by

The multi-stage node formulation

- In this case the decision variables are associated with the nodes of the scenario tree which we use to represent the dynamic evolution of the uncertain parameters
- Let $\mathcal{N}$ denote the set of nodes of the scenario tree
- For each node $n \in \mathcal{N}$ we denote by
  - P_{in} the price of asset i
  - L_n the liability
  - F_n the available fund to invest
  - r_n the risk-free interest rate

The multi-stage node formulation

For each node $n \in \mathcal{N}$ we denote by

- S_{in} the amount of security i to sell
- B_{in} the amount of security i to be purchased
- H_{in} the amount of security i to be maintained in the portfolio
- v_n the monetary amount invested in a risk-free asset

The multi-stage node formulation

The multi-stage node formulation

- **Physical balance constraints**

$$H_{in} = H_{ia(n)} + B_{in} - S_{in} \; \forall i \; \forall n \in \mathcal{N} - \{0\} - \{\text{leaf node}\}$$
$$H_{i0} = InitHold_i + B_{i0} - S_{i0} \quad i = 1, \dots N$$
$$S_{in} = H_{ia(n)} \qquad i = 1, \dots N \quad \forall n \in \{\text{leaf node}\}$$

The multi-stage node formulation

- **Physical balance constraints**

$$H_{in} = H_{ia(n)} + B_{in} - S_{in} \ \forall i \ \forall n \in \mathcal{N} - \{0\} - \{\text{leaf node}\}$$
$$H_{i0} = InitHold_i + B_{i0} - S_{i0} \quad i = 1, \dots N$$
$$S_{in} = H_{ia(n)} \qquad i = 1, \dots N \quad \forall n \in \{\text{leaf node}\}$$

- **Monetary balance constraints**

$$(1-g)\sum_{i=1}^{N} P_{i0}S_{i0} + F_0 = (1+g)\sum_{i=1}^{N} P_{i0}B_{i0} + L_0 + v_0$$
$$(1-g)\sum_{i=1}^{N} P_{in}S_{in} + F_n + (1+r_n)v_{a(n)} =$$
$$(1+g)\sum_{i=1}^{N} P_{in}B_{in} + L_n + v_n \quad n \in \mathcal{N} - \{0\} - \{\text{leaf node}\}$$

The multi-stage node formulation

- **Definition of the wealth at the leaf nodes**

The multi-stage node formulation

- **Definition of the wealth at the leaf nodes**

$$W_n = (1-g)\sum_{i=1}^{N} P_{in} H_{ia(n)} + (1+r_n) v_{a(n)} + F_n - L_n \ \forall n \text{ leaf node}$$

- **The objective function**

$$\max z = \sum_{n \ \in \{\text{leaf nodes}\}} p_n * W_n$$

The multi-stage split formulation

The multi-stage split formulation

In this case the decision variables have a double index (stage, scenario)
We denote by

The multi-stage split formulation

In this case the decision variables have a double index (stage, scenario)
We denote by

- P_{it}^{s} the price of asset i at stage t under scenario s
- L_{t}^{s} the liability at stage t under scenario s
- F_{t}^{s} the available fund to invest at stage t under scenario s
- r_{t}^{s} the risk-free interest rate at stage t under scenario s

The multi-stage split formulation

For t and$s$ we denote by

The multi-stage split formulation

For t and$s$ we denote by

- S_{it}^{s} the amount of security i to sell at stage t under scenario s
- B_{it}^{s} the amount of security i to be purchased at stage t under scenario s
- H_{it}^{s} the amount of security i to be maintained in the portfolio at stage t under scenario s
- v_{t}^{s} the monetary amount invested in a risk-free asset at stage t under scenario s

The multi-stage split formulation

The multi-stage split formulation

- **Physical balance constraints**

$$H^s_{it} = H^s_{it-1} + B^s_{it} - S^s_{it} \quad i = 1, \dots N \quad t = 2, \dots T-1 \; \forall s$$
$$H^s_{i1} = InitHold_i + B^s_{i1} - S^s_{i1} \quad i = 1, \dots N \quad s = 1, \dots S$$
$$S^s_{iT} = H^s_{1T-1} \quad i = 1, \dots N \; \forall s$$

The multi-stage split formulation

- **Physical balance constraints**

$$H_{it}^s = H_{it-1}^s + B_{it}^s - S_{it}^s \quad i = 1, \dots N \quad t = 2, \dots T-1 \ \forall s$$
$$H_{i1}^s = InitHold_i + B_{i1}^s - S_{i1}^s \quad i = 1, \dots N \quad s = 1, \dots S$$
$$S_{iT}^s = H_{1T-1}^s \quad i = 1, \dots N \ \forall s$$

- **Monetary balance constraints**

$$(1-g)\sum_{i=1}^{N} P_{i1}^s S_{i1}^s + F_1^s = (1+g)\sum_{i=1}^{N} P_{i1}^s B_{i1}^s + L_1^s + v_1^s \ \forall s$$

$$(1-g)\sum_{i=1}^{N} P_{it}^s S_{it}^s + F_t^s + (1+r_t^s)v_{t-1}^s =$$
$$(1+g)\sum_{i=1}^{N} P_{it}^s B_{it}^s + L_t^s + v_t^s \quad t = 2, \dots T-1 \ \forall s$$

The multi-stage node formulation

The multi-stage node formulation

- **Definition of the last stage**

$$W_T^s = (1-g)\sum_{i=1}^{N} P_{iT}^s H_{iT-1}^s + (1+r_T^s)v_{T-1}^s + F_T^s - L_T^s \ \forall s$$

The multi-stage node formulation

- **Definition of the last stage**

$$W_T^s = (1-g)\sum_{i=1}^{N} P_{iT}^s H_{iT-1}^s + (1+r_T^s)v_{T-1}^s + F_T^s - L_T^s \ \forall s$$

- **The non-anticipativity constraints**

The multi-stage node formulation

- **Definition of the last stage**

$$W_T^s = (1-g)\sum_{i=1}^{N} P_{iT}^s H_{iT-1}^s + (1+r_T^s)v_{T-1}^s + F_T^s - L_T^s \ \forall s$$

- **The non-anticipativity constraints**

$$\begin{aligned}
H_{in} &= H_{it(n)}^s \quad \forall n \in \mathcal{N} \quad s \in S(n)\\
S_{in} &= S_{it(n)}^s \quad \forall n \in \mathcal{N} \quad s \in S(n)\\
B_{in} &= B_{it(n)}^s \quad \forall n \in \mathcal{N} \quad s \in S(n)\\
v_n &= v_{t(n)}^s \quad \forall n \in \mathcal{N} \quad s \in S(n)
\end{aligned}$$

The multi-stage node formulation

- **Definition of the last stage**

$$W_T^s = (1-g)\sum_{i=1}^{N} P_{iT}^s H_{iT-1}^s + (1+r_T^s)v_{T-1}^s + F_T^s - L_T^s \;\; \forall s$$

- **The non-anticipativity constraints**

$$\begin{aligned}
H_{in} &= H_{it(n)}^s \quad \forall n \in \mathcal{N} \quad s \in S(n)\\
S_{in} &= S_{it(n)}^s \quad \forall n \in \mathcal{N} \quad s \in S(n)\\
B_{in} &= B_{it(n)}^s \quad \forall n \in \mathcal{N} \quad s \in S(n)\\
v_{n} &= v_{t(n)}^s \quad \forall n \in \mathcal{N} \quad s \in S(n)
\end{aligned}$$

- **The objective function**

$$\max z = \sum_{s=1}^{S} p_s W_T^s$$

Exercise

- Let us consider a classical production problem defined over a time horizon divided in a given number T of periods.
- For the sake of simplicity, we shall consider a single product.
- At each period t, we have to decide the amount to produce so to satisfy a given demand
- We consider the possibility to stock the eventual amount in excess
- Define the deterministic and the stochastic models assuming that the evolution of the uncertain demand can be represented as the scenario tree reported below

The deterministic version

For each time period $t = 1, \ldots, T$ we denote by

- c_t the unitary production cost
- π_t the unitary inventory cost
- d_t the demand
- x_t the amount to produce
- I_t the amount to stock

$$\min z = \sum_{t=1}^{T} (c_t * x_t + \pi_t * I_t)$$

$$I_t = I_{t-1} + x_t - d_t \qquad t = 1, \ldots T$$

$$x_t \geq 0, \qquad I_t \geq 0 \qquad t = 1, \ldots, T$$

The scenario tree: example