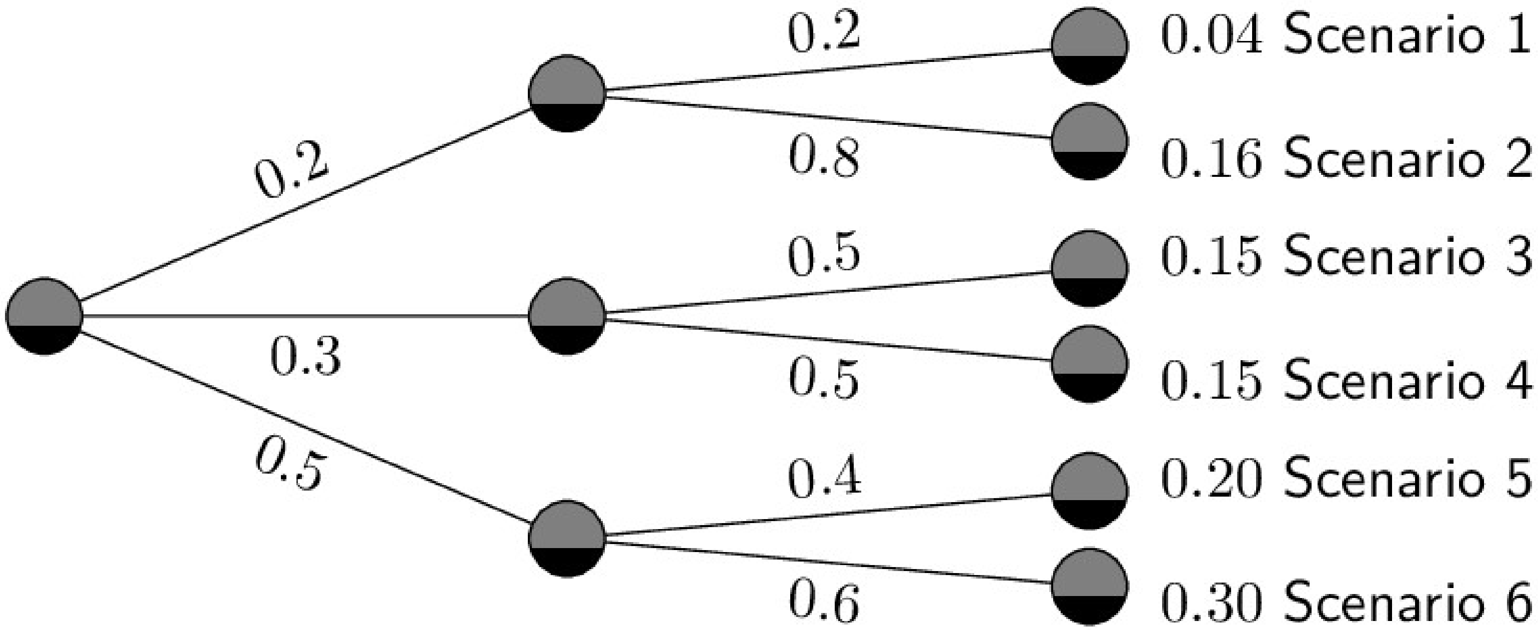
Decision Making under Uncertainty (02435)

Section for Dynamical Systems, DTU Compute.

DTU Compute

Department of Applied Mathematics and Computer Science

Scenario tree



Probabilities at the branches are conditional probabilities to pass along the branch.
Probabilities of all child nodes of each node sum up to 1.
The probability of a scenario is the product of probabilities along the path.

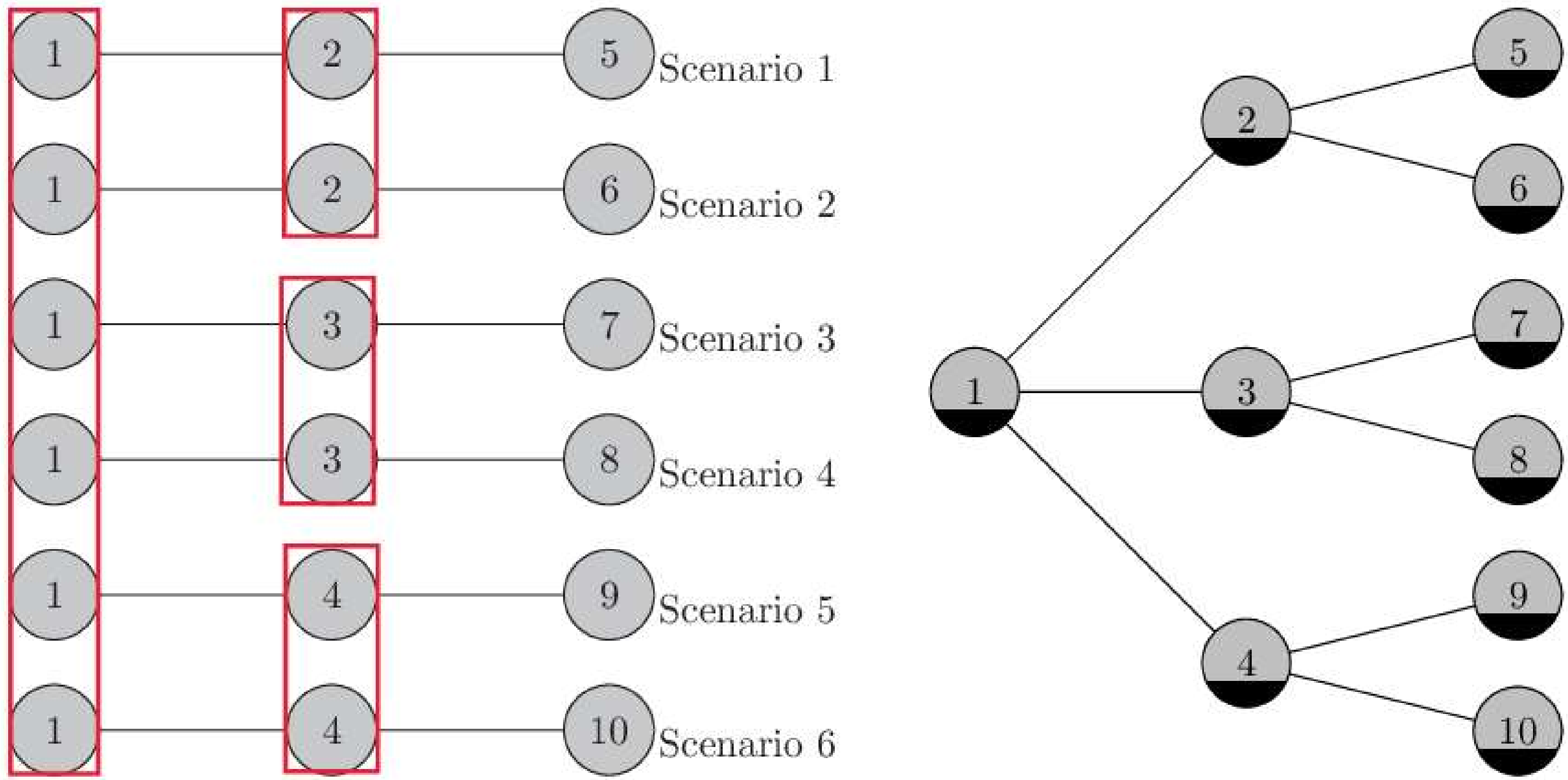
Nonanticipativity

Nonanticipativity

Decisions are made based on the information available up to the current stage and future information cannot be anticipated.

Scenario-based formulation

We define all variables based on scenarios and ensure non-anticipativity explicitly by constraints (marked in red).



New sets:
$\mathcal{S}_s^h$ = Scenarios that need the same value as scenario s on stage h

Scenario-based formulation: Wind power production

All decision variables are now defined based on the scenarios.

p_s^D	Amount to bid in the day-ahead market [MWh] $\forall s \in \mathcal{S}$ (1st stage)
p_s^A	Bid to the adjustment market production [MWh] $\forall s \in \mathcal{S}$ (2nd stage)
p_s^+	Excess production in scenario s [MWh] $\forall s \in \mathcal{S}$ (3rd stage)
p_s^-	Missing production in scenario s [MWh] $\forall s \in \mathcal{S}$ (3rd stage)

$$\text{Max} \sum_{s \in \mathcal{S}} \pi_s (\lambda^D p_s^D + \lambda^A P_s^A + \lambda^+ p_s^+ - \lambda^- p_s^-)$$

$$\begin{aligned}
\text{s.t. } & p_s^D \leq \overline{P} && \forall s \in \mathcal{S} && \text{(max production)} \\
& p_s^D + p_s^A \leq \overline{P} && \forall s \in \mathcal{S} && \text{(max production)} \\
& W_s - p_s^D - p_s^A = p_s^+ - p_s^- && \forall s \in \mathcal{S} && \text{(deviation)} \\
& p_s^D = p_{s'}^D && \forall s \in \mathcal{S}, s' \in \mathcal{S}_s^1 && \text{(non-anticipativity)} \\
& p_s^A = p_{s'}^A && \forall s \in \mathcal{S}, s' \in \mathcal{S}_s^2 && \text{(non-anticipativity)}
\end{aligned}$$

$$p_s^D \in \mathbb{R}^+ \quad \forall s \in \mathcal{S}, \qquad p_s^A \in \mathbb{R} \quad \forall s \in \mathcal{S}, \qquad p_s^+, p_s^- \in \mathbb{R}^+ \quad \forall s \in \mathcal{S}$$

Implicit vs Explicit Non-Anticipativity

Implicit non-anticipativity

$$x\_First\_Stage_{w}$$
$$x\_Second\_Stage_{w,s}$$

Explicit non-anticipativity

$$x_{w,t,s}$$
$$x_{w,1,s} = x_{w,1,s'}\,, \quad \forall w \in W, \quad s, s' \in S$$

Which is better?

Explicit non-anticipativity (Multi-stage)

$$x_{w,t,s}$$

$$x_{w,t,s} = x_{w,t,s'}\,, \forall w \in W, t \in T, s \in S, s' \in S_s^t$$

Nonanticipativity - Summary

Nonanticipativity

Decisions are made based upon the information available up to the current stage and future information can not be anticipated.

- When using a scenario-based formulation, we lose the so called **nonanticipativity**, because the decision variables are scenario-based, i.e., scenario-dependent decisions can be made. We need to enforce nonanticipativity **explicitly by constraints**.
- In the node-based formulation, the nonanticipativity is **implicitly encoded** in the definitions of the variables (history up to the current stage).

Sequential Decisions

Information → Decision → Information → Decision → …

Solution concept: *policy*

A policy is a mapping from the available information at stage t to a decision at t

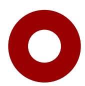
In this course:
Updated information →
Scenario Generation (for a lookahead horizon) →
Scenario Reduction →
SP (for the horizon, only the here-and-now decisions apply)

Within time

Stochastic Lookahead Policy

5-stage problem

Policy: Lookahead horizon of 3 stages with 6 scenarios

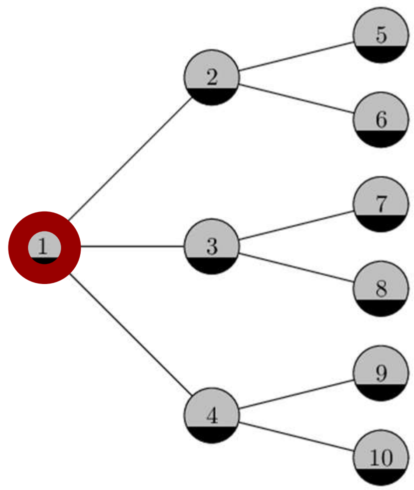


Stages: 1 2 3 4 5

Stochastic Lookahead Policy

5-stage problem

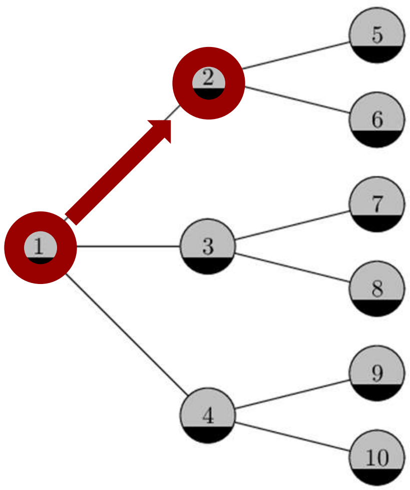
Policy: Lookahead horizon of 3 stages with 6 scenarios



Stages: 1 2 3 4 5

Stochastic Lookahead Policy

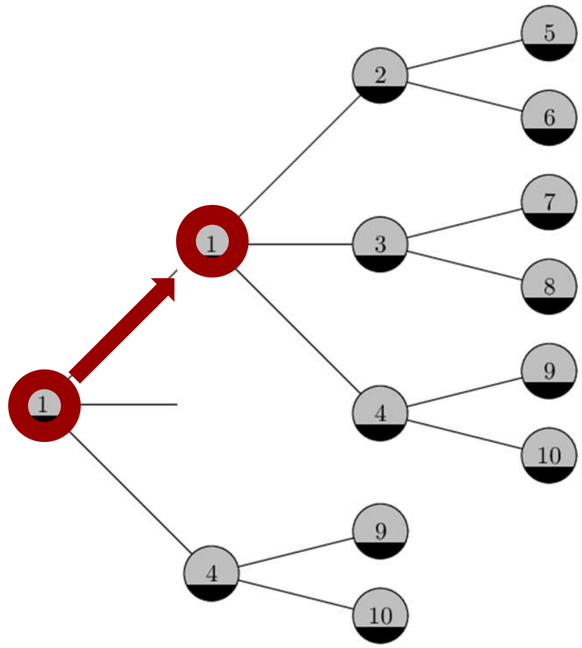
5-stage problem
Policy: Lookahead horizon of 3 stages with 6 scenarios



Stages: 1 2 3 4 5

Stochastic Lookahead Policy

5-stage problem
Policy: Lookahead horizon of 3 stages with 6 scenarios

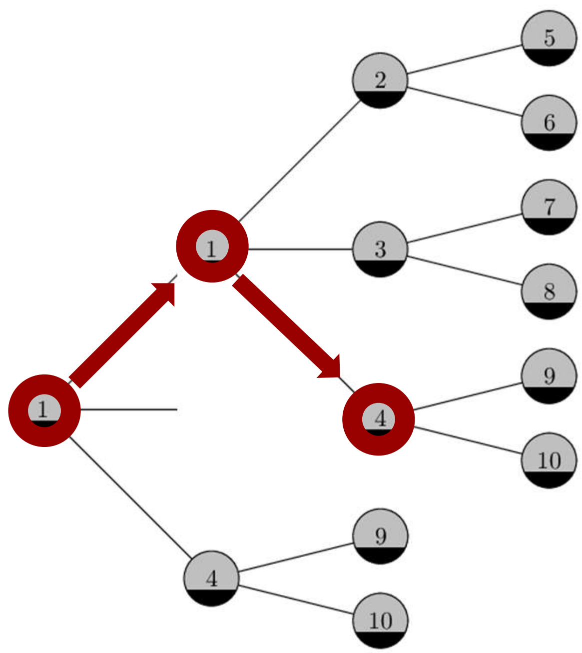


Stages: 1 2 3 4 5

Stochastic Lookahead Policy

5-stage problem

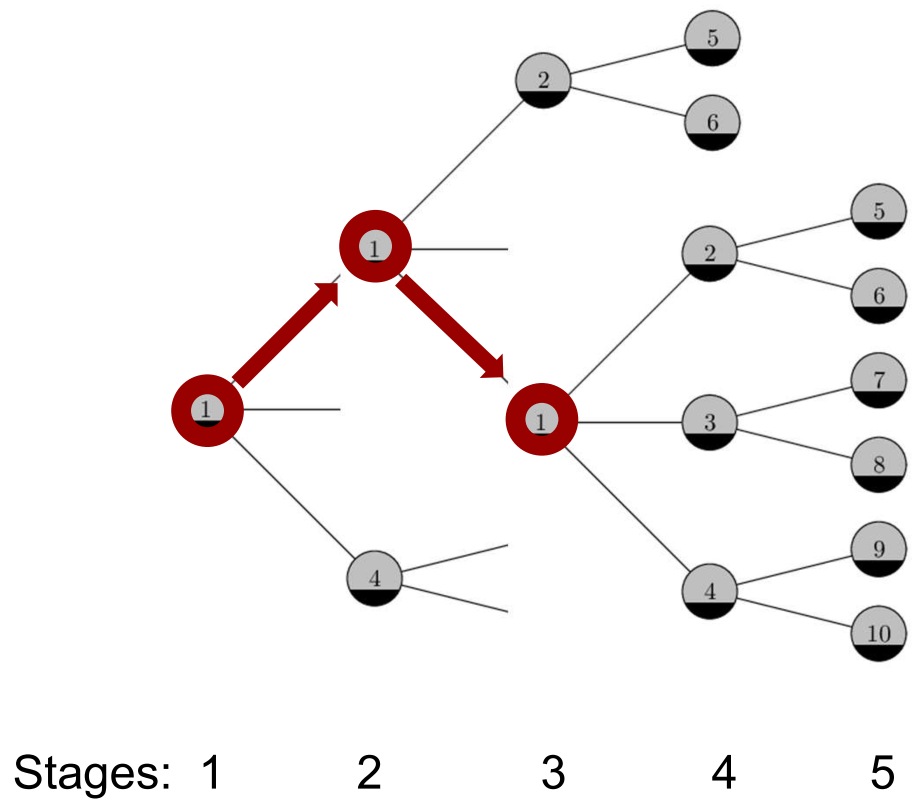
Policy: Lookahead horizon of 3 stages with 6 scenarios



Stages: 1 2 3 4 5

Stochastic Lookahead Policy

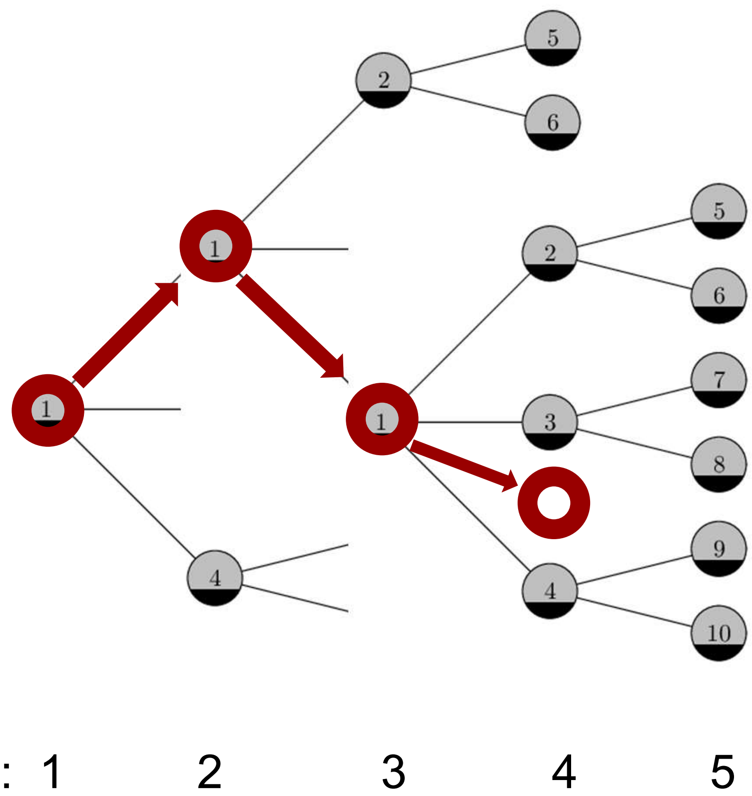
5-stage problem
Policy: Lookahead horizon of 3 stages with 6 scenarios



Stochastic Lookahead Policy

5-stage problem

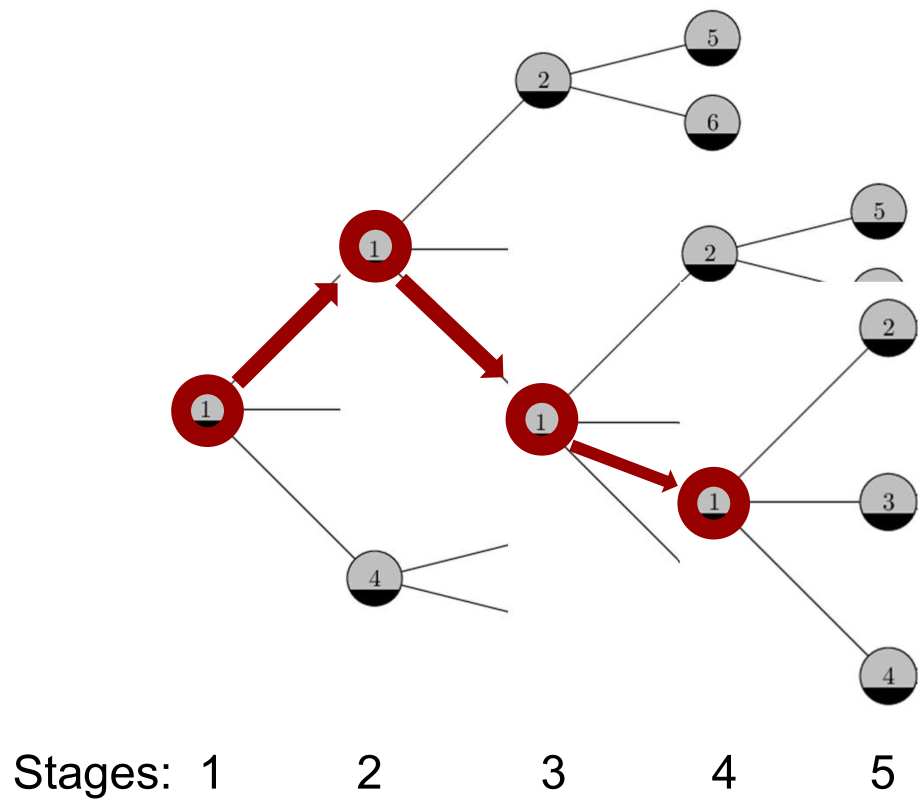
Policy: Lookahead horizon of 3 stages with 6 scenarios



Stochastic Lookahead Policy

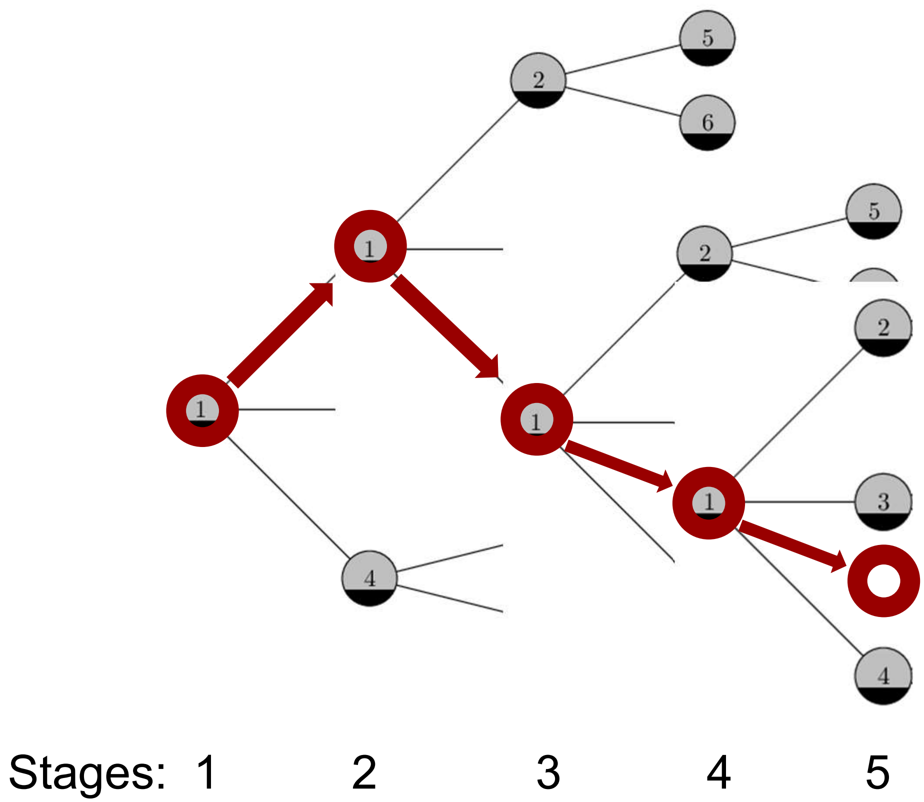
5-stage problem

Policy: Lookahead horizon of 3 stages with 6 scenarios



Stochastic Lookahead Policy

5-stage problem
Policy: Lookahead horizon of 3 stages with 6 scenarios



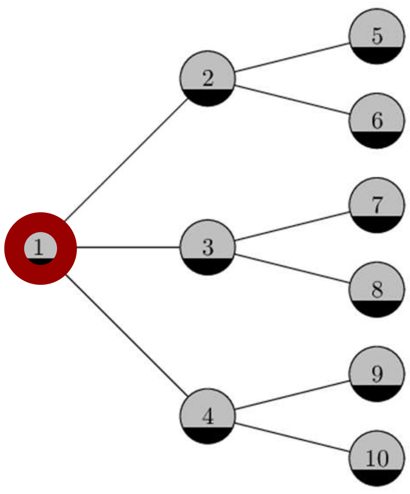
To Evaluate the policy:
I evaluate the cost of the *effective* decisions

How to build a Policy

Input:
```
make_multistage_here_and_now_decision(number_of_sim_periods, tau, current_stock, current_prices)
```

Output: here-and-now decisions
1) Define the number of look-ahead days
2) Define the initial number of scenarios
3) Generate your scenarios: Price(w,t,s)
- A scenario is a prices trajectory
- For stage 1: set the prices to the current prices
- For stage t: "sample_next" from Price(w,t-1,s)

1) Reduce scenarios
2) Reassign probabilities
3) Create and populate the "non-anticipativity" sets S_s^t (e.g. by simply iterating through all scenario pairs and, for each pair, checking up to which stage they share history.



Stages: 1 2 3 4 5

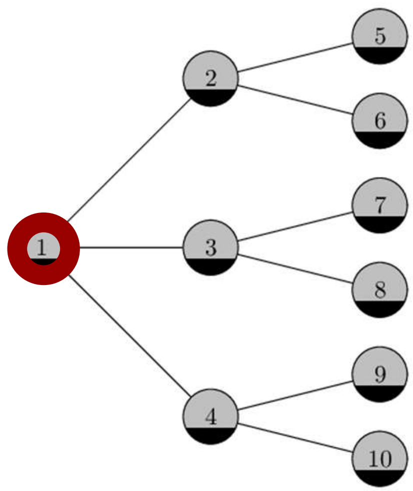
How many scenarios do you think will have shared history up to stage 1?

How many up to stage 2?

How to build a Policy

Input: current prices
Output: here-and-now decisions

1) Define the number of look-ahead days
2) Define the initial number of scenarios
3) Generate your scenarios: Price(w,t,s)
   - A scenario is a prices trajectory
   - For the stage 1: set the prices equal to the current prices
   - For stage t: “sample_next” from Price(w,t-1,s)



Stages: 1 2 3 4 5

→ Discretize Scenarios!

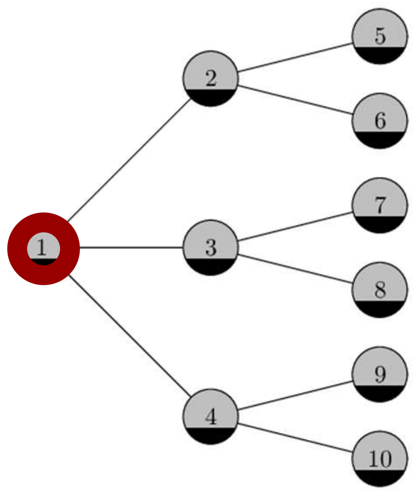
If you don’t discretize scenarios, you cannot encode non-anticipativity!

4) Reduce scenarios
5) Reassign probabilities
6) Create and populate the “non-anticipativity” sets S_s^t

How to build a Policy

Input: current prices
Output: here-and-now decisions

1) Define the number of look-ahead days
2) Define the initial number of scenarios
3) Generate your scenarios: Price(w,t,s)
   - A scenario is a prices trajectory
   - For the stage 1: set the prices equal to the current prices
   - For stage t: “sample_next” from Price(w,t-1,s)



Stages: 1 2 3 4 5

4) Discretize Scenarios
   define the discrete price values
   round each price to the closest value

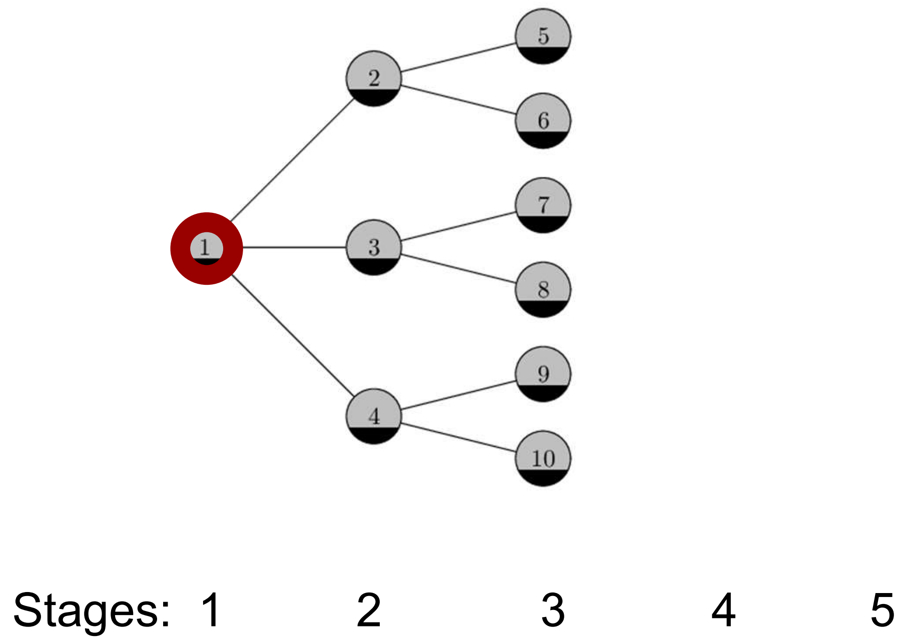
5) Reduce scenarios (down to how many?)
6) Reassign probabilities
7) Create and populate the “non-anticipativity” sets S_s^t

8) Solve the program…

How to build a Policy

Input: current prices
Output: here-and-now decisions

1) Define the number of look-ahead days
2) Define the initial number of scenarios
3) Generate your scenarios: Price(w,t,s)
   - A scenario is a prices trajectory
   - For the stage 1: set the prices equal to the current prices
   - For stage t: "sample_next" from Price(w,t-1,s)



4) Discretize Scenarios

   define the discrete price values
   round each price to the closest value

5) Reduce scenarios (down to how many?)
6) Reassign probabilities
7) Create and populate the "non-anticipativity" sets S_s^t

Design choices:
- Number of lookahead days
- Initial number of scenarios
- Discretization Granularity
- Number of final scenarios

8) Solve the program…

Design Choices and Considerations

Design choices:

- Initial number of scenarios **use no more than 1000 samples for scenario generation**

- Number of lookahead days
- Number of final scenarios

→ Number of variables for the stochastic program

use no more than 1000 samples for scenario generation, and no more than 6,500 variables for the stochastic program (hint: there is a trade-off between the number of scenarios and the length of the lookahead horizon).

- Discretization Granularity

→ If too fine: non-anticipativity is lost
→ If too coarse: statistical information of the uncertainty is lost