Stochastic Optimization
Multistage stochastic programming

V. Leclère

November 25, 2026





Presentation Outline

1. From two stages to T stages
2. Information constraint
3. Progressive hedging on the tree
4. Nested Benders on the tree
5. Bounds and heuristics

After this session, you should be able to

- write a multistage problem, and say for each decision what it knows;
- write its extensive formulation on a scenario tree;
- run progressive hedging and nested Benders on that tree;
- bracket the value of a problem you cannot solve.

Presentation Outline

1. From two stages to T stages
2. Information constraint
3. Progressive hedging on the tree
4. Nested Benders on the tree
5. Bounds and heuristics

From two stages to T stages

- A dam operator, a distributor, an investor: each decides, observes, and decides again.
- T stages, the noise of each preceding its decision:

$$\xi_1 \rightsquigarrow y_1 \rightsquigarrow \boldsymbol{\xi}_2 \rightsquigarrow \boldsymbol{y}_2 \rightsquigarrow \cdots \rightsquigarrow \boldsymbol{\xi}_T \rightsquigarrow \boldsymbol{y}_T \,.$$

- ξ_1 is data, not random: y_1 is a vector.
- History $\xi_{[t]} := (\xi_r)_{r\in[t]}$, decisions so far $y_{[t]} := (y_r)_{r\in[t]}$.
- A stage may depend on everything before it: $y_t \in \mathcal{Y}_t(y_{[t-1]}, \xi_{[t]})$, at the cost $\ell_t(y_{[t-1]}, \xi_{[t]}, y_t)$.
- At $T = 2$: session 2a's two-stage problem, $\boldsymbol{\xi}_2$ for $\boldsymbol{\xi}$.

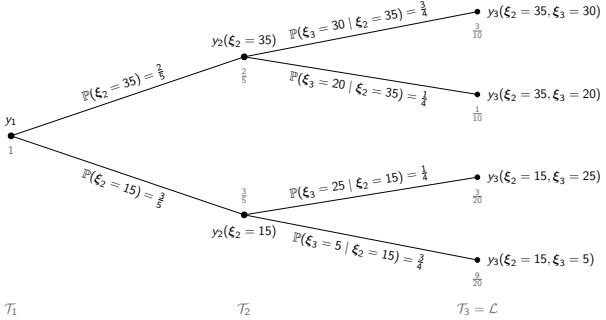
The multistage stochastic problem

$$\begin{aligned} \min_{(\boldsymbol{y}_t)_{t\in[T]}} \quad & \mathbb{E}_{\mathbb{P}}\Big[\sum_{t=1}^{T} \ell_t(\boldsymbol{y}_{[t-1]}, \boldsymbol{\xi}_{[t]}, \boldsymbol{y}_t)\Big] \\ \text{s.t.} \quad & \boldsymbol{y}_t \in \mathcal{Y}_t(\boldsymbol{y}_{[t-1]}, \boldsymbol{\xi}_{[t]}), && \forall t \in [T], \ \ \mathbb{P}\text{-a.s.} \\ & \boldsymbol{y}_t \preceq \boldsymbol{\xi}_{[t]}, && \forall t \in [T] \end{aligned}$$

- The first constraint says which decisions a stage admits.
- The information constraint has no counterpart in a deterministic problem: $\boldsymbol{y}_t$ may depend on $\boldsymbol{\xi}_{[t]}$, not on the noise still to come.
- (FSN): each $\boldsymbol{\xi}_t$ takes finitely many values. Then $\boldsymbol{y}_t \preceq \boldsymbol{\xi}_{[t]}$ means $\boldsymbol{y}_t = \phi_t(\boldsymbol{\xi}_{[t]})$: the problem is over policies $\phi = (\phi_t)_{t\in[T]}$, of expected cost $J(\phi)$, and $v^{\mathrm{SP}} = \inf_{\phi \text{ admissible}} J(\phi)$.
- A policy can be simulated even when its tree cannot be written down.

Example: shipping over three stages

Session 2a's distributor now delivers on two dates, demand ξ_2 then ξ_3, met in full: a boat at 1 a unit at stage 1, a second boat at $\frac{3}{2}$ at stage 2, a plane at 3 for at most 15 units at either date, no holding cost.



Three decisions that know three different things: y_1 nothing, stage 2 the first demand, stage 3 both. A node is a history, an arc a probability *given the node it leaves*; the two stage-2 nodes branch differently. By the end of the session: $v^{\mathrm{PI}} = 40 \leq v^{\mathrm{SP}} = 55 \leq v^{\mathrm{NA}} = 65$.

Extensive formulation: node form

- Scenario tree $\mathcal{T}$: a node $\nu \in \mathcal{T}_t$ is a history observable at stage t, of probability π_ν; its children ν^+ are the values the next noise can still take. The leaves $\mathcal{L} := \mathcal{T}_T$ are the scenarios.
- Node form: one decision per node,

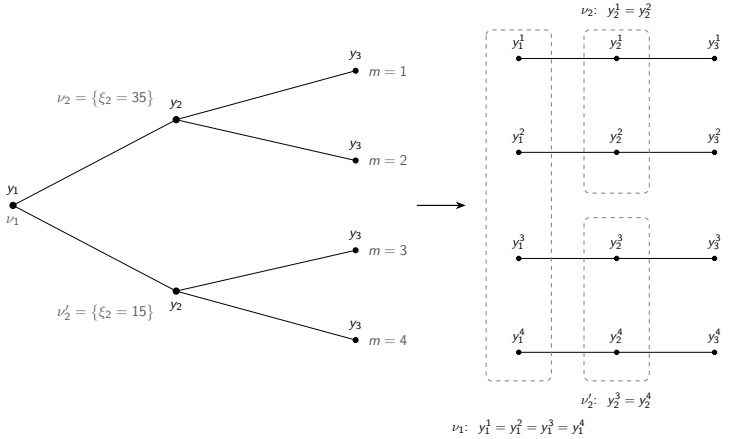
$$\begin{aligned} \min_{(y_\nu)_{\nu\in\mathcal{T}}} \quad & \sum_{t=1}^{T} \sum_{\nu\in\mathcal{T}_t} \pi_\nu \, \ell_t\big(y_{[\nu^-]}, \nu, y_\nu\big) \\ \text{s.t.} \quad & y_\nu \in \mathcal{Y}_t\big(y_{[\nu^-]}, \nu\big), \qquad \forall t \in [T], \ \ \forall \nu \in \mathcal{T}_t \ , \end{aligned}$$

  with $y_{[\nu^-]}$ the decisions on the path to ν.
- No random variable, no expectation, no information constraint: a deterministic problem, linear if the stages are.

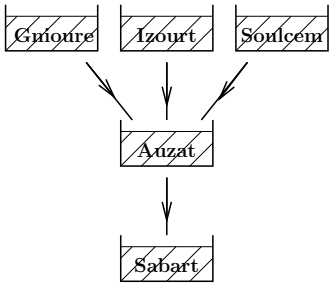
Extensive formulation: scenario form

Scenario form: one trajectory per leaf m, $(y_t^m)_{t\in[T]} \in \mathcal{C}^m$, weighted by π_m, the copies of a node tied by

$$y_t^m = y_t^{m'}, \quad \forall t \in [T], \ \forall \nu \in \mathcal{T}_t, \ \forall m, m' \in \mathcal{L}(\nu) \ ,$$



The tree is exponential in the number of stages



- b values per stage: b^{t-1} nodes at stage t, b^{T-1} leaves.
- Five dams, week by week over a year, ten possible weekly inflow vectors: $T = 52$, $b = 10$, so 10^{51} scenarios.
- $T = 7$: 10^6 scenarios, a laptop writes them down. $T = 13$: 10^{12}, a cluster does not. There are about 10^{24} stars in the observable universe.

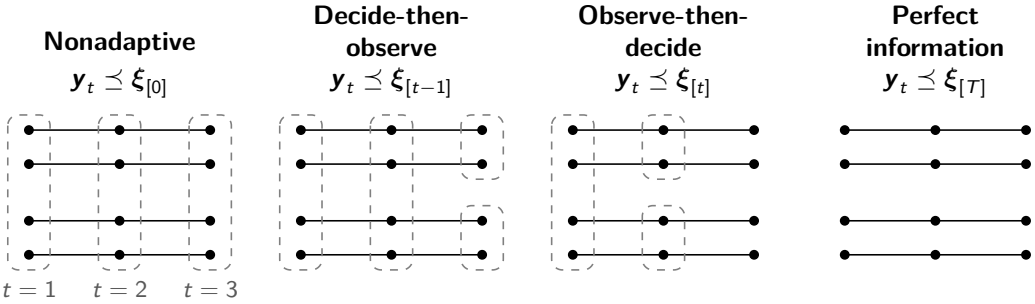
One decision per node, and the number of nodes is exponential in the number of stages: no solver or hardware changes that.

Presentation Outline

1. From two stages to T stages
2. Information constraint
3. Progressive hedging on the tree
4. Nested Benders on the tree
5. Bounds and heuristics

Four information structures

One constraint, four subscripts: $y_t \preceq \xi_{[0]}$, $\xi_{[t-1]}$, $\xi_{[t]}$ or $\xi_{[T]}$.



- Each structure is a partition of the scenarios' copies; each refines the one before.
- A finer partition admits more decisions: $v^{\mathrm{PI}} \leq v^{\mathrm{OTD}} \leq v^{\mathrm{DTO}} \leq v^{\mathrm{NA}}$, and $v^{\mathrm{OTD}} = v^{\mathrm{SP}}$ is the one we assume.
- On the shipping problem: $40 \leq 55 \leq 65 = 65$.

Choosing a structure

- Nonadaptive: a plan, decided in advance (a timetable, a maintenance programme). If in reality the plan could move, v^{NA} is an upper bound.
- Decide-then-observe and observe-then-decide differ by half a stage; observe-then-decide gives a value no larger.
- Perfect information: nobody operates under it, but it is a lower bound, one deterministic problem per scenario, estimated by Monte Carlo.

We assume observe-then-decide.

Choosing a structure

- Nonadaptive: a plan, decided in advance (a timetable, a maintenance programme). If in reality the plan could move, v^{NA} is an upper bound.
- Decide-then-observe and observe-then-decide differ by half a stage; observe-then-decide gives a value no larger.
- Perfect information: nobody operates under it, but it is a lower bound, one deterministic problem per scenario, estimated by Monte Carlo.

We assume observe-then-decide.

Choosing a structure

- Nonadaptive: a plan, decided in advance (a timetable, a maintenance programme). If in reality the plan could move, v^{NA} is an upper bound.
- Decide-then-observe and observe-then-decide differ by half a stage; observe-then-decide gives a value no larger.
- Perfect information: nobody operates under it, but it is a lower bound, one deterministic problem per scenario, estimated by Monte Carlo.

We assume observe-then-decide.

Choosing a structure

- Nonadaptive: a plan, decided in advance (a timetable, a maintenance programme). If in reality the plan could move, v^{NA} is an upper bound.
- Decide-then-observe and observe-then-decide differ by half a stage; observe-then-decide gives a value no larger.
- Perfect information: nobody operates under it, but it is a lower bound, one deterministic problem per scenario, estimated by Monte Carlo.

We assume observe-then-decide.

A decision that cannot see its noise meets its constraint robustly



- A decision chosen after ξ_t meets $\mathcal{Y}_t(y_{[t-1]}, \xi_{[t]})$ at the value that occurred.
- Chosen before (decide-then-observe, nonadaptive), it must meet it at every value the noise can still take: an almost-sure constraint on a decision that cannot see its noise is a robust constraint.
- There may be none, although each value alone admits one.

Half a stage. A warehouse manager orders

- on Monday for the week's demand: decide-then-observe;
- on Friday, knowing the week's sales: observe-then-decide;
- on Monday with a forecast: in between.

A stage is a moment of decision, not a time step of the data: with hourly data and a decision every twelve hours, a 48-hour problem has four stages, each with twelve hourly values in its noise.

A decision that cannot see its noise meets its constraint robustly



- A decision chosen after ξ_t meets $\mathcal{Y}_t(y_{[t-1]}, \xi_{[t]})$ at the value that occurred.
- Chosen before (decide-then-observe, nonadaptive), it must meet it at every value the noise can still take: an almost-sure constraint on a decision that cannot see its noise is a robust constraint.
- There may be none, although each value alone admits one.

Half a stage. A warehouse manager orders

- on Monday for the week's demand: decide-then-observe;
- on Friday, knowing the week's sales: observe-then-decide;
- on Monday with a forecast: in between.

A stage is a moment of decision, not a time step of the data: with hourly data and a decision every twelve hours, a 48-hour problem has four stages, each with twelve hourly values in its noise.

Exercise: which information structure?

For each, what does the decision know when it is taken?

1. An electricity producer commits tomorrow's output on the day-ahead market; tomorrow's wind is uncertain.
2. A grain merchant contracts the season's shipments once the yield is known.
3. A hospital rosters next month's night shifts before the month begins.
4. A maintenance crew is dispatched each morning to the failures reported overnight.

1. Decide-then-observe: the commitment precedes the stage's own noise, $y_t \preceq \xi_{[t-1]}$.

2, 4. Observe-then-decide: $y_t \preceq \xi_{[t]}$.

3. Nonadaptive, $y_t \preceq \xi_{[0]}$: the roster is a plan. Revise it weekly and the question returns.

Exercise: which information structure?

For each, what does the decision know when it is taken?

1. An electricity producer commits tomorrow's output on the day-ahead market; tomorrow's wind is uncertain.
2. A grain merchant contracts the season's shipments once the yield is known.
3. A hospital rosters next month's night shifts before the month begins.
4. A maintenance crew is dispatched each morning to the failures reported overnight.

1. Decide-then-observe: the commitment precedes the stage's own noise, $\boldsymbol{y}_t \preceq \boldsymbol{\xi}_{[t-1]}$.

2, 4. Observe-then-decide: $\boldsymbol{y}_t \preceq \boldsymbol{\xi}_{[t]}$.

3. Nonadaptive, $\boldsymbol{y}_t \preceq \boldsymbol{\xi}_{[0]}$: the roster is a plan. Revise it weekly and the question returns.

Presentation Outline

1. From two stages to T stages
2. Information constraint
3. Progressive hedging on the tree
4. Nested Benders on the tree
5. Bounds and heuristics

Progressive hedging on the tree

The scenario form couples the leaves through $\mathcal{N}$ alone: session 2a's algorithm applies, changed only where red.

- Average per node, over the leaves that extend it:

$$\bar{y}_\nu := \frac{1}{\pi_\nu} \sum_{m \in \mathcal{L}(\nu)} \pi_m \, y_t^m = \mathbb{E}_{\mathbb{P}}\big[\boldsymbol{y}_t \mid \boldsymbol{\xi}_{[t]} = \nu \big] , \qquad \nu \in \mathcal{T}_t \, .$$

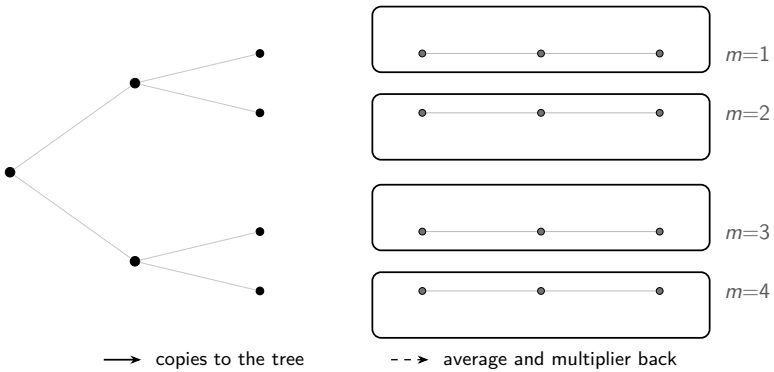
- Subproblem of leaf m, with $\nu = m_{[t]}$ its node at stage t:

$$\min_{(y_t)_{t \in [T]} \in \mathcal{C}^m} \sum_{t \in [T]} \ell_t\big(y_{[t-1]}, m_{[t]}, y_t\big) + \sum_{t \in [T-1]} \lambda_t^{m\top} y_t + \frac{\tau}{2} \sum_{t \in [T-1]} \| y_t - \bar{y}_\nu \|^2 \, .$$

- The leaf stage carries no term: at stage T each leaf is its own node.
- Multipliers centred within each node: $\mathbb{E}_{\mathbb{P}}\big[\boldsymbol{\lambda}_t \mid \boldsymbol{\xi}_{[t]} \big] = 0$.
- Convergence carries over: $\mathcal{N}$ is still a closed subspace.
- One iteration: $|\mathcal{L}|$ deterministic T-stage problems, each with a quadratic term.

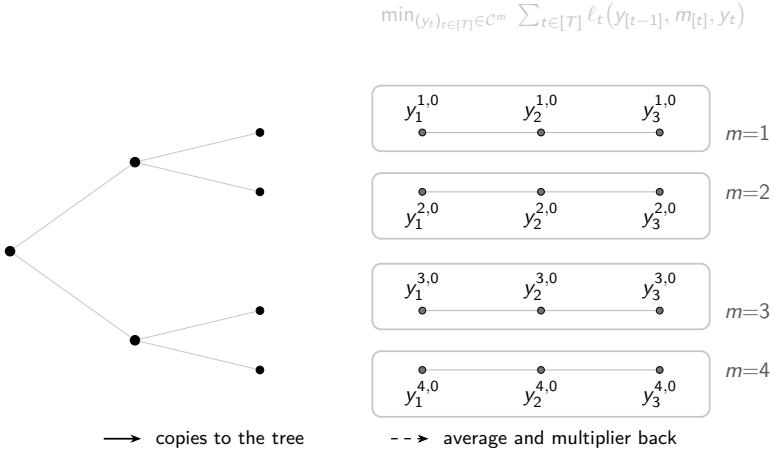
Two iterations of progressive hedging on the tree

$\min_{(y_t)_{t\in[T]}\in\mathcal{C}^m} \sum_{t\in[T]} \ell_t\big(y_{[t-1]}, m_{[t]}, y_t\big)$



Start. Each leaf m has its own deterministic problem (a box); at the start it is solved alone, with no multiplier and no proximal term.

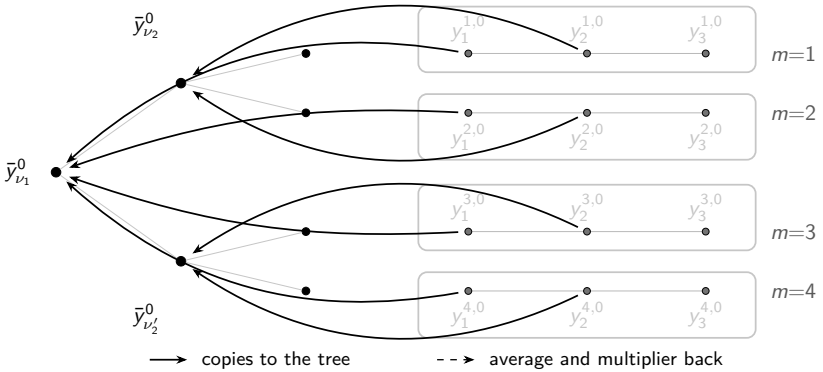
Two iterations of progressive hedging on the tree



Copies. Each scenario returns a trajectory $(y_1^{m,0}, y_2^{m,0}, y_3^{m,0})$: the copies of one node disagree.

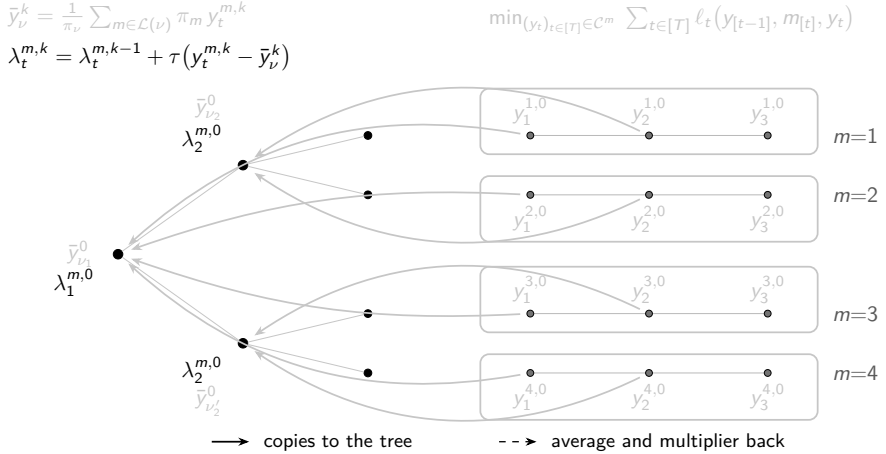
Two iterations of progressive hedging on the tree

$\bar{y}_\nu^k = \frac{1}{\pi_\nu} \sum_{m \in \mathcal{L}(\nu)} \pi_m \, y_t^{m,k}$ $\quad \min_{(y_t)_{t\in[T]} \in \mathcal{C}^m} \sum_{t\in[T]} \ell_t\big(y_{[t-1]}, m_{[t]}, y_t\big)$



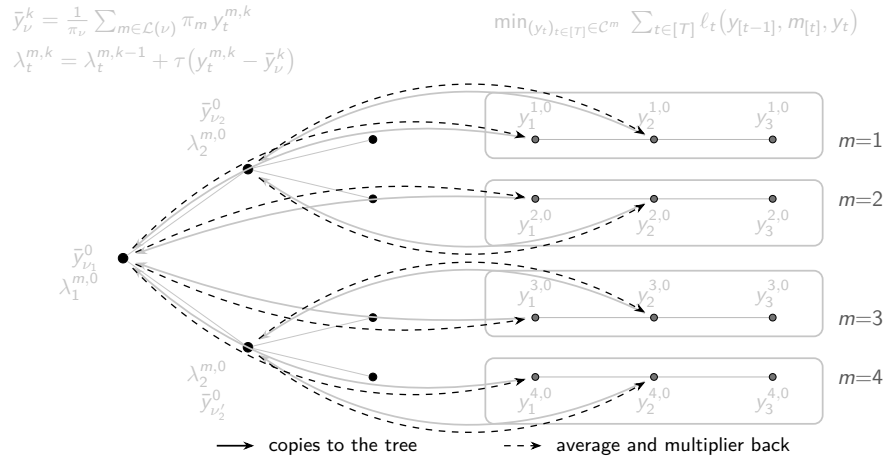
To the tree. The copies of stages 1 and 2 go to the node of their history, which averages those of the leaves that extend it.

Two iterations of progressive hedging on the tree



Multipliers. At each node, every copy's multiplier is moved by its own gap to the average. A leaf is its own node: no multiplier.

Two iterations of progressive hedging on the tree



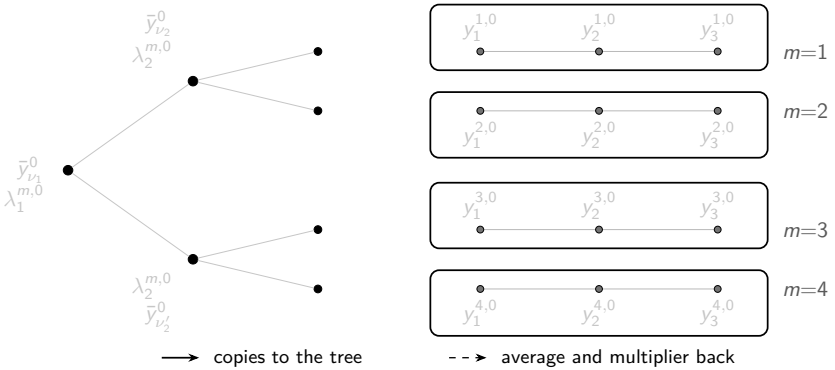
Back to the scenarios. Each copy gets its multiplier $\lambda_t^{m,0}$ and the average $\bar{y}_\nu^0$ of its node.

Two iterations of progressive hedging on the tree

$\bar{y}_\nu^k = \frac{1}{\pi_\nu} \sum_{m \in \mathcal{L}(\nu)} \pi_m \, y_t^{m,k}$

$\lambda_t^{m,k} = \lambda_t^{m,k-1} + \tau \left(y_t^{m,k} - \bar{y}_\nu^k \right)$

$\min_{(y_t)_{t \in [T]} \in \mathcal{C}^m} \sum_{t \in [T]} \ell_t(y_{[t-1]}, m_{[t]}, y_t)$
$+ \sum_{t \in [T-1]} \lambda_t^{m,k-1\top} y_t + \frac{\tau}{2} \sum_{t \in [T-1]} \| y_t - \bar{y}_\nu^{k-1} \|^2$



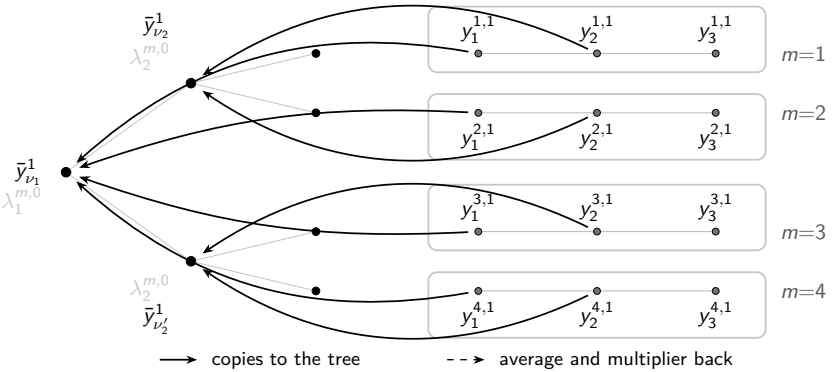
Iteration 1. Each problem now has the linear term in λ and the proximal term towards $\bar{y}$, on stages 1 and 2 only; solved again, still alone.

Two iterations of progressive hedging on the tree

$\bar{y}_\nu^k = \frac{1}{\pi_\nu} \sum_{m \in \mathcal{L}(\nu)} \pi_m \, y_t^{m,k}$

$\lambda_t^{m,k} = \lambda_t^{m,k-1} + \tau\big(y_t^{m,k} - \bar{y}_\nu^k\big)$

$\min_{(y_t)_{t\in[T]} \in \mathcal{C}^m} \sum_{t\in[T]} \ell_t\big(y_{[t-1]}, m_{[t]}, y_t\big)$
$+ \sum_{t\in[T-1]} \lambda_t^{m,k-1\top} y_t + \frac{\tau}{2} \sum_{t\in[T-1]} \|y_t - \bar{y}_\nu^{k-1}\|^2$



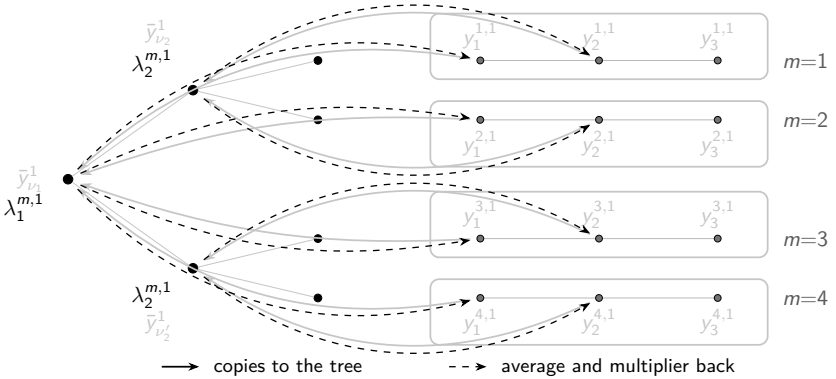
New copies. Solutions $y_t^{m,1}$, sent to the tree, whose nodes recompute their averages $\bar{y}_\nu^1$.

Two iterations of progressive hedging on the tree

$\bar{y}_\nu^k = \frac{1}{\pi_\nu} \sum_{m \in \mathcal{L}(\nu)} \pi_m \, y_t^{m,k}$

$\lambda_t^{m,k} = \lambda_t^{m,k-1} + \tau\left(y_t^{m,k} - \bar{y}_\nu^k\right)$

$\min_{(y_t)_{t\in[T]} \in \mathcal{C}^m} \sum_{t\in[T]} \ell_t\left(y_{[t-1]}, m_{[t]}, y_t\right)$
$+ \sum_{t\in[T-1]} \lambda_t^{m,k-1\top} y_t + \frac{\tau}{2} \sum_{t\in[T-1]} \|y_t - \bar{y}_\nu^{k-1}\|^2$



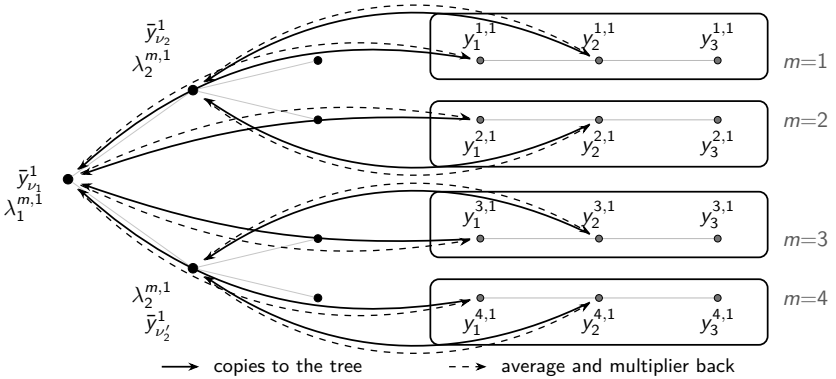
New multipliers. Moved again by the new gaps and sent back with the new averages: ready for iteration 2.

Two iterations of progressive hedging on the tree

$\bar{y}_\nu^k = \frac{1}{\pi_\nu} \sum_{m \in \mathcal{L}(\nu)} \pi_m \, y_t^{m,k}$

$\lambda_t^{m,k} = \lambda_t^{m,k-1} + \tau\big(y_t^{m,k} - \bar{y}_\nu^k\big)$

$\min_{(y_t)_{t\in[T]} \in \mathcal{C}^m} \sum_{t\in[T]} \ell_t\big(y_{[t-1]}, m_{[t]}, y_t\big)$
$+ \sum_{t\in[T-1]} \lambda_t^{m,k-1\top} y_t + \frac{\tau}{2} \sum_{t\in[T-1]} \|y_t - \bar{y}_\nu^{k-1}\|^2$



The whole iteration. $|\mathcal{L}|$ deterministic T-stage problems solved alone; each node averages the copies of its leaves and moves their multipliers by their gaps.

Presentation Outline

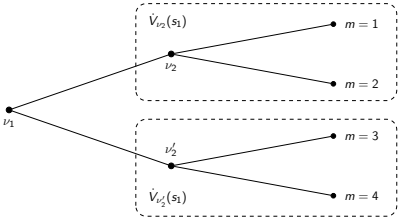
1. From two stages to T stages
2. Information constraint
3. Progressive hedging on the tree
4. Nested Benders on the tree
5. Bounds and heuristics

Nested Benders I: linear stages and the cost-to-go

- A subtree receives the whole decision history; most problems hand down far less (shipping: the stock).
- Linear stages: stage t hands on a state s_t and uses the past only through s_{t-1}:

$$A_t\, s_t + B_t\, s_{t-1} + C_t\, y_t = d_t\ , \quad y_t \geq 0\ ,$$

cost $c_t^\top y_t$, data functions of ξ_t, polyhedral final cost $V_{T+1}(s_T)$; the law of the noise may still depend on the whole history.



- Once a node's state is fixed, the subtrees of its children share no variable.

$\dot{V}_\nu(s)$, the expected cost of the subtree rooted at ν, handed the state s by its parent:

$$\dot{V}_\nu(s) = \min_{y\geq 0,\ s'} \left\{ c_\nu^\top y + \sum_{\nu' \in \nu^+} \frac{\pi_{\nu'}}{\pi_\nu}\, \dot{V}_{\nu'}(s') \mid A_\nu\, s' + B_\nu\, s + C_\nu\, y = d_\nu \right\} .$$

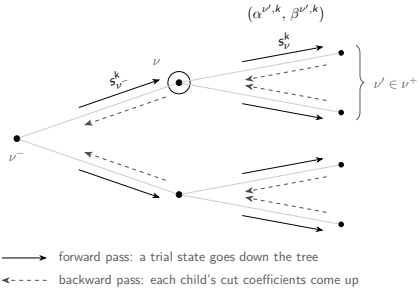
Nested Benders II: the node problem and the passes

At the state s_{ν^-} handed down, node ν solves

$$\min_{y_\nu \geq 0,\, s_\nu,\, \theta_{\nu'}} \; c_\nu^\top y_\nu + \sum_{\nu' \in \nu^+} \frac{\pi_{\nu'}}{\pi_\nu} \theta_{\nu'}$$

$$\text{s.t. } A_\nu\, s_\nu + B_\nu\, s_{\nu^-} + C_\nu\, y_\nu = d_\nu$$

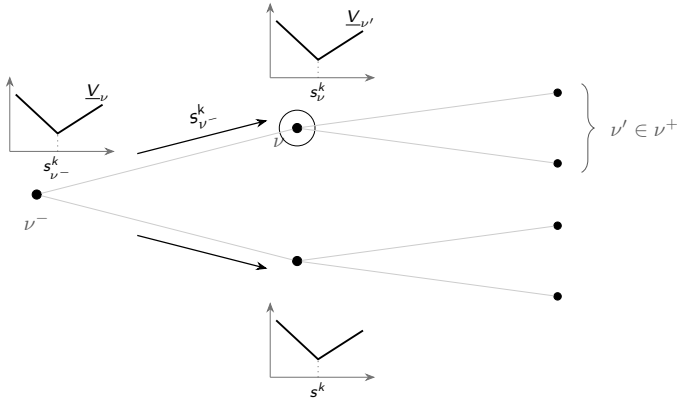
$$\theta_{\nu'} \geq \left(\alpha^{\nu',i}\right)^\top \left(s_\nu - s_\nu^i\right) + \beta^{\nu',i} \; .$$



Master and subproblem at once: the master of its children, the subproblem of its parent.

- **Forward pass**, root to leaves: trial states handed down.
- **Backward pass**, leaves to root: each node, at its parent's trial state, returns an optimality cut, as in L-shaped.
- Start with a backward pass at an admissible first trajectory.
- Finite convergence: each node has finitely many cuts, and the tree is finite.

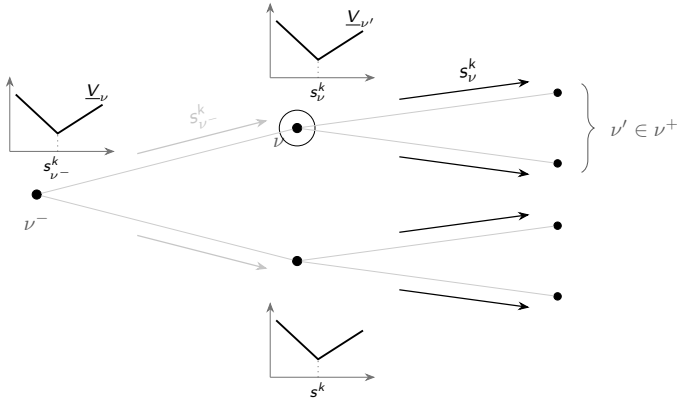
One iteration of nested Benders



Forward pass. The root ν^- solves with no cut yet, picks $s^k_{\nu^-}$ and hands it to each child.

Note that node ν is the master of its children and the subproblem of its parent at once.

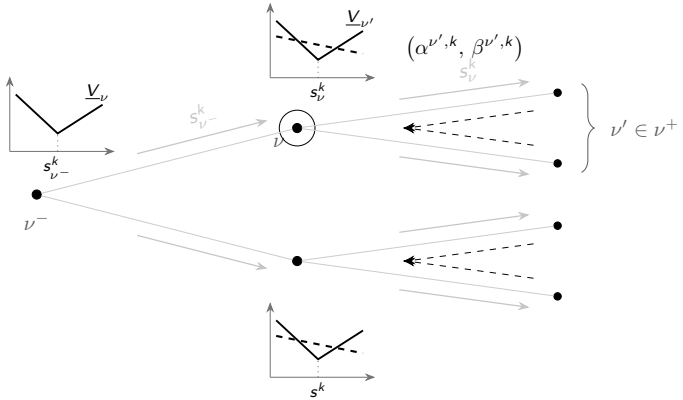
One iteration of nested Benders



Forward pass. Node ν solves at the state $s^k_{\nu^-}$ it was handed, picks s^k_ν and hands it to each of its children: every node now has a trial state.

Note that node ν is the master of its children and the subproblem of its parent at once.

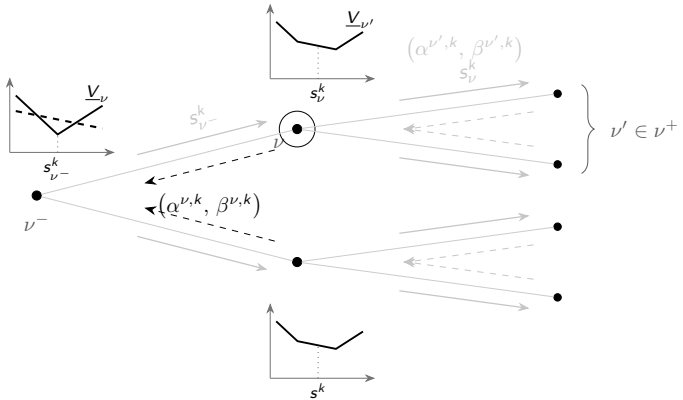
One iteration of nested Benders



Backward pass. Each leaf returns an optimality cut to its parent, built from its optimal dual: every model gains one.

Note that node ν is the master of its children and the subproblem of its parent at once.

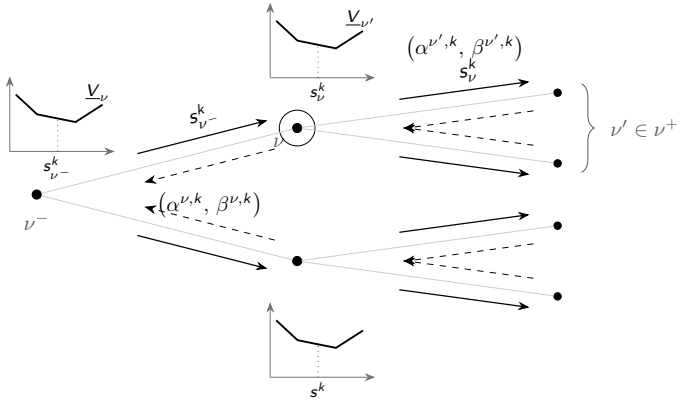
One iteration of nested Benders



Backward pass. Node ν, holding the cuts of its children, solves again at $s^k_{\nu^-}$ and returns its own cut to ν^-.

Note that node ν is the master of its children and the subproblem of its parent at once.

One iteration of nested Benders



One iteration later. Every model has gained a cut, so the next forward pass hands down states read off models that lie higher: the bounds close.

Note that node ν is the master of its children and the subproblem of its parent at once.

Presentation Outline

1. From two stages to T stages
2. Information constraint
3. Progressive hedging on the tree
4. Nested Benders on the tree
5. Bounds and heuristics

Bounds and heuristics



- Beyond 4 or 5 stages the tree cannot be enumerated. To know how far a heuristic is from optimal, bracket v^{SP}:
  - below, by relaxing the information constraint;
  - above, by the cost of a policy.
- An upper bound is the cost of a policy, admissible at every node and simulated on scenarios the optimizer never saw (session 1), with its confidence interval.
- Exact algorithms for long horizons need assumptions on the noise (stagewise independence): dynamic programming, next sessions.
- Decision rules (affine in the history) give another heuristic: the adaptive robust session.

Lower bounds



- **A ladder.** $v^{(k)}$: the information constraint kept for $t \le k$, perfect information $y_t \preceq \xi_{[T]}$ for $t > k$. Then

$$v^{\mathrm{PI}} = v^{(0)} \le v^{\mathrm{2SA}} = v^{(1)} \le \cdots \le v^{(T-1)} = v^{(T)} = v^{\mathrm{SP}} .$$

  In practice only $k = 0, 1$ are computed.

- **Perfect information.** One deterministic problem per leaf:

$$v^{\mathrm{PI}} = \sum_{m \in \mathcal{L}} \pi_m \min_{(y_t)_{t \in [T]} \in \mathcal{C}^m} \sum_{t \in [T]} \ell_t(y_{[t-1]}, m_{[t]}, y_t) .$$

  Draw trajectories, solve each, average, with a confidence interval: an unbiased estimate. Shipping: $v^{\mathrm{PI}} = 40$.

- **Any multiplier gives a bound.** $v^{\mathrm{D}}(\lambda) \le v^{\mathrm{SP}}$ for every centred λ: each iterate of progressive hedging gives a lower bound, at one solve per leaf without the proximal term.

Lower bounds



- **A ladder.** $v^{(k)}$: the information constraint kept for $t \leq k$, perfect information $y_t \preceq \xi_{[T]}$ for $t > k$. Then

$$v^{\mathrm{PI}} = v^{(0)} \leq v^{\mathrm{2SA}} = v^{(1)} \leq \cdots \leq v^{(T-1)} = v^{(T)} = v^{\mathrm{SP}} .$$

  In practice only $k = 0, 1$ are computed.

- **Perfect information.** One deterministic problem per leaf:

$$v^{\mathrm{PI}} = \sum_{m \in \mathcal{L}} \pi_m \min_{(y_t)_{t \in [T]} \in \mathcal{C}^m} \sum_{t \in [T]} \ell_t\big(y_{[t-1]}, m_{[t]}, y_t\big) .$$

  Draw trajectories, solve each, average, with a confidence interval: an unbiased estimate. Shipping: $v^{\mathrm{PI}} = 40$.

- **Any multiplier gives a bound.** $v^{\mathrm{D}}(\lambda) \leq v^{\mathrm{SP}}$ for every centred λ: each iterate of progressive hedging gives a lower bound, at one solve per leaf without the proximal term.

Lower bounds



- **A ladder.** $v^{(k)}$: the information constraint kept for $t \leq k$, perfect information $y_t \preceq \xi_{[T]}$ for $t > k$. Then

$$v^{\mathrm{PI}} = v^{(0)} \leq v^{\mathrm{2SA}} = v^{(1)} \leq \cdots \leq v^{(T-1)} = v^{(T)} = v^{\mathrm{SP}} .$$

  In practice only $k = 0, 1$ are computed.

- **Perfect information.** One deterministic problem per leaf:

$$v^{\mathrm{PI}} = \sum_{m \in \mathcal{L}} \pi_m \min_{(y_t)_{t \in [T]} \in \mathcal{C}^m} \sum_{t \in [T]} \ell_t\big(y_{[t-1]}, m_{[t]}, y_t\big) .$$

  Draw trajectories, solve each, average, with a confidence interval: an unbiased estimate. Shipping: $v^{\mathrm{PI}} = 40$.

- **Any multiplier gives a bound.** $v^{\mathrm{D}}(\lambda) \leq v^{\mathrm{SP}}$ for every centred λ: each iterate of progressive hedging gives a lower bound, at one solve per leaf without the proximal term.

The (sampled) two-stage relaxation



- Keep nonanticipativity at the root alone: y_1 is the first stage, the rest of the trajectory a recourse with hindsight. Its value is v^{2SA}.
- As many scenarios as the original problem, so sample it as in session 1: draw N trajectories, solve the two-stage problem on them.
- Bias, for the sampled value $\widehat{v}_N$:

$$\mathbb{E}\left[\widehat{v}_N\right] \leq v^{\mathrm{2SA}} \leq v^{\mathrm{SP}} .$$

  The average of $\widehat{v}_N$ over independent samples estimates a lower bound; a single sampled value is neither a bound nor an unbiased estimate of one.
- N independent trajectories make a fan: whatever N, sampling a multistage problem that way approximates its two-stage relaxation. A tree needs conditional sampling, $\prod_{t=2}^{T} N_t$ trajectories.

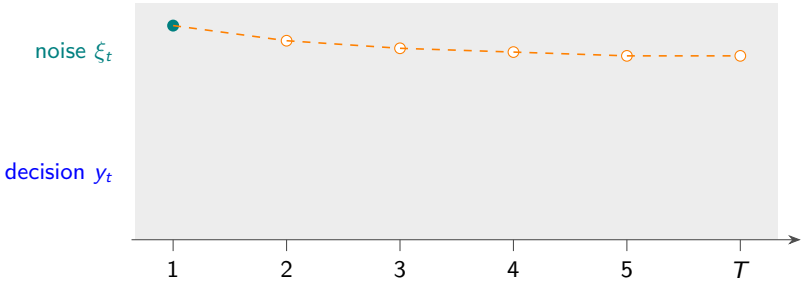
The expected-value plan need not be admissible

- Replace every noise by its expectation and solve the deterministic problem: a plan.
- Shipping: demands 23 and 17, so it books $y_1 = 40$ by boat and nothing else. After a first date at 35 it holds 5, a plane carries at most 15: a second date at 30 cannot be served.
- The best plan is worth v^{NA} (65 here), $+\infty$ if no plan meets every scenario's constraints; a stochastic gradient method can search for it.
- A good first decision; the two heuristics that follow re-solve the problem at every stage, on what has been observed by then.

Model predictive control

At stage t, with $y_{[t-1]}$ applied, $\xi_{[t]}$ observed and each later noise at its forecast $\mathbb{E}_{\mathbb{P}}\left[\boldsymbol{\xi}_\tau \mid \boldsymbol{\xi}_{[t]}\right]$, solve over the stages left (a shrinking horizon) and apply its first decision:

$$\min_{y_1,\dots,y_T} \sum_{\tau=1}^{T} \ell_\tau\left(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\right) .$$

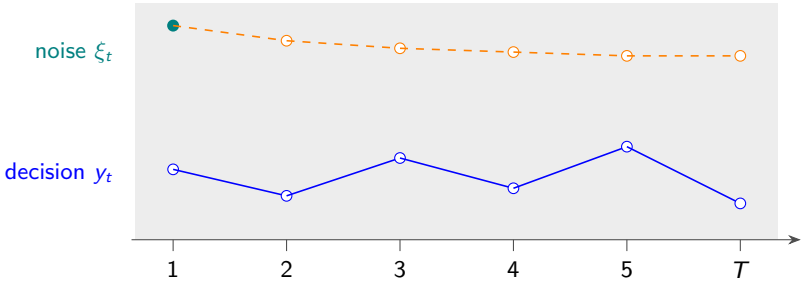


$t = 1$. Beyond ξ_1 there is only the forecast, and the forecast is one number per stage.

Model predictive control

At stage t, with $y_{[t-1]}$ applied, $\xi_{[t]}$ observed and each later noise at its forecast $\mathbb{E}_{\mathbb{P}}\left[\boldsymbol{\xi}_\tau \mid \boldsymbol{\xi}_{[t]}\right]$, solve over the stages left (a shrinking horizon) and apply its first decision:

$$\min_{y_1,\dots,y_T} \sum_{\tau=1}^{T} \ell_\tau\left(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\right) .$$

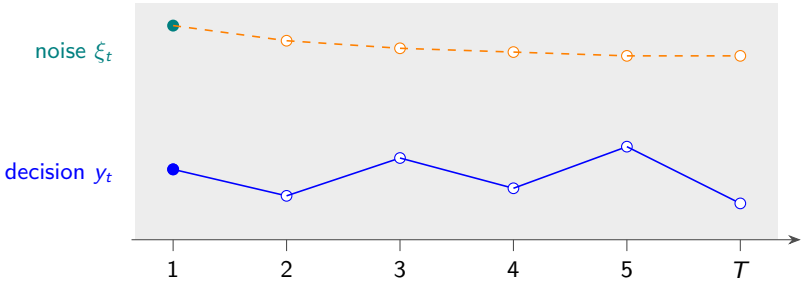


Solve. One deterministic problem, over every stage that is left.

Model predictive control

At stage t, with $y_{[t-1]}$ applied, $\xi_{[t]}$ observed and each later noise at its forecast $\mathbb{E}_{\mathbb{P}}\left[\boldsymbol{\xi}_\tau \mid \boldsymbol{\xi}_{[t]}\right]$, solve over the stages left (a shrinking horizon) and apply its first decision:

$$\min_{y_1,\dots,y_T} \sum_{\tau=1}^{T} \ell_\tau\left(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\right) .$$

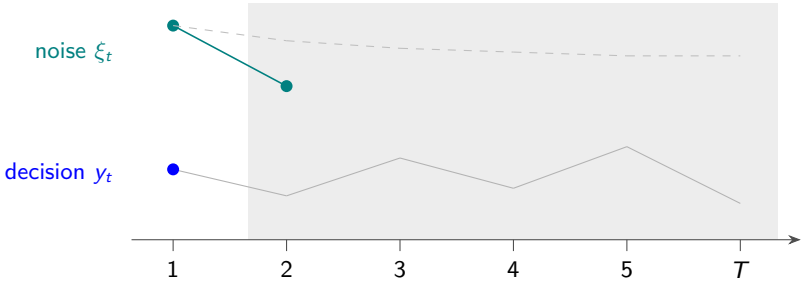


Apply, then discard. Only y_1 is played; the rest of the plan is thrown away.

Model predictive control

At stage t, with $y_{[t-1]}$ applied, $\xi_{[t]}$ observed and each later noise at its forecast $\mathbb{E}_{\mathbb{P}}\left[\boldsymbol{\xi}_\tau \mid \boldsymbol{\xi}_{[t]}\right]$, solve over the stages left (a shrinking horizon) and apply its first decision:

$$\min_{y_2,\dots,y_T} \sum_{\tau=2}^{T} \ell_\tau\left(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\right) .$$

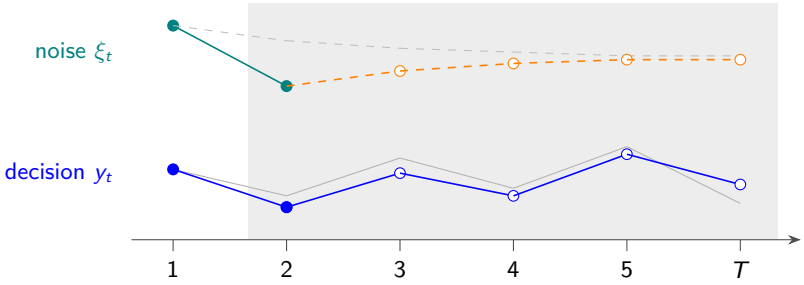


$t = 2$. ξ_2 comes in below what stage 1 predicted, which is now the first grey dashes.

Model predictive control

At stage t, with $y_{[t-1]}$ applied, $\xi_{[t]}$ observed and each later noise at its forecast $\mathbb{E}_{\mathbb{P}}\left[\boldsymbol{\xi}_\tau \mid \boldsymbol{\xi}_{[t]}\right]$, solve over the stages left (a shrinking horizon) and apply its first decision:

$$\min_{y_2,\dots,y_T} \sum_{\tau=2}^{T} \ell_\tau\left(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\right) .$$

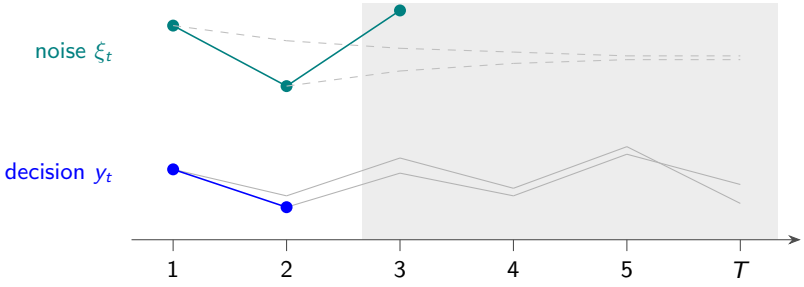


Re-solve. A new forecast, a new plan, and again only its first decision is applied.

Model predictive control

At stage t, with $y_{[t-1]}$ applied, $\xi_{[t]}$ observed and each later noise at its forecast $\mathbb{E}_{\mathbb{P}}\left[\boldsymbol{\xi}_\tau \mid \boldsymbol{\xi}_{[t]}\right]$, solve over the stages left (a shrinking horizon) and apply its first decision:

$$\min_{y_3,\dots,y_T} \sum_{\tau=3}^{T} \ell_\tau\left(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\right) .$$

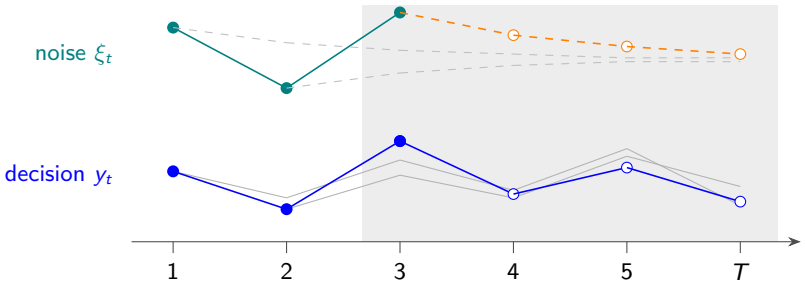


$t = 3$. This time the noise comes in above the forecast.

Model predictive control

At stage t, with $y_{[t-1]}$ applied, $\xi_{[t]}$ observed and each later noise at its forecast $\mathbb{E}_{\mathbb{P}}\left[\boldsymbol{\xi}_\tau \mid \boldsymbol{\xi}_{[t]}\right]$, solve over the stages left (a shrinking horizon) and apply its first decision:

$$\min_{y_3,\dots,y_T} \sum_{\tau=3}^{T} \ell_\tau\left(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\right) .$$

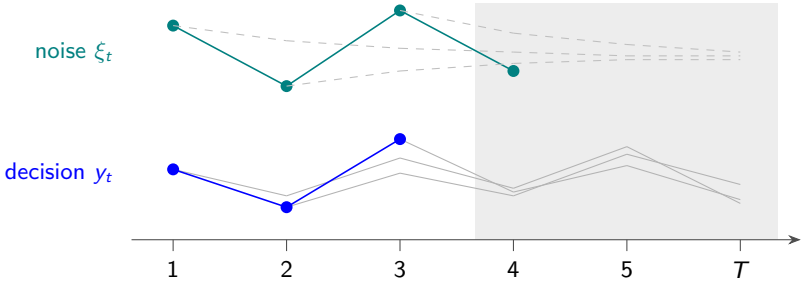


Re-solve. Each forecast differs from the one before, so each plan differs from the one before.

Model predictive control

At stage t, with $y_{[t-1]}$ applied, $\xi_{[t]}$ observed and each later noise at its forecast $\mathbb{E}_{\mathbb{P}}\left[\boldsymbol{\xi}_\tau \mid \boldsymbol{\xi}_{[t]}\right]$, solve over the stages left (a shrinking horizon) and apply its first decision:

$$\min_{y_4,\dots,y_T} \sum_{\tau=4}^{T} \ell_\tau\left(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\right) .$$

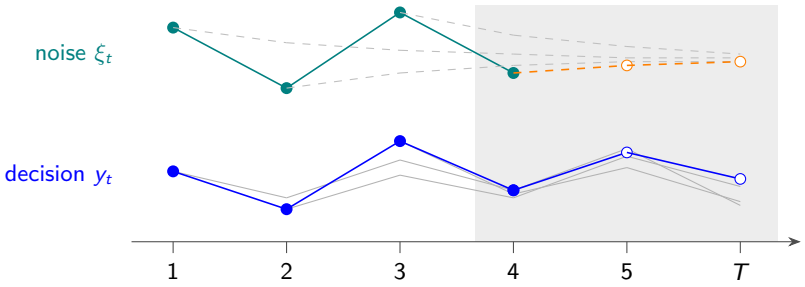


$t = 4$. Observed, and again away from the prediction.

Model predictive control

At stage t, with $y_{[t-1]}$ applied, $\xi_{[t]}$ observed and each later noise at its forecast $\mathbb{E}_{\mathbb{P}}\left[\boldsymbol{\xi}_\tau \mid \boldsymbol{\xi}_{[t]}\right]$, solve over the stages left (a shrinking horizon) and apply its first decision:

$$\min_{y_4,\dots,y_T} \sum_{\tau=4}^{T} \ell_\tau\left(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\right) .$$

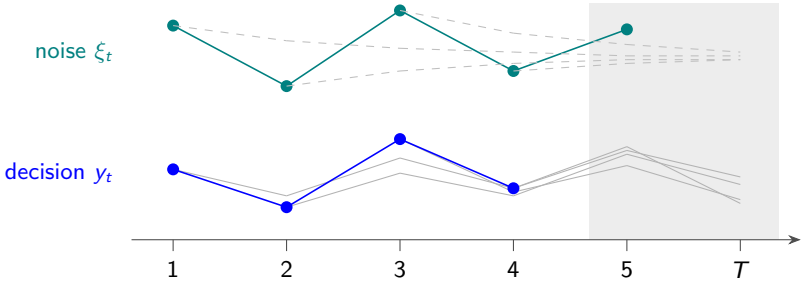


Re-solve. The blue path at the bottom is the only thing that was ever played.

Model predictive control

At stage t, with $y_{[t-1]}$ applied, $\xi_{[t]}$ observed and each later noise at its forecast $\mathbb{E}_{\mathbb{P}}\left[\boldsymbol{\xi}_\tau \mid \boldsymbol{\xi}_{[t]}\right]$, solve over the stages left (a shrinking horizon) and apply its first decision:

$$\min_{y_5,\dots,y_T} \sum_{\tau=5}^{T} \ell_\tau\left(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\right) .$$

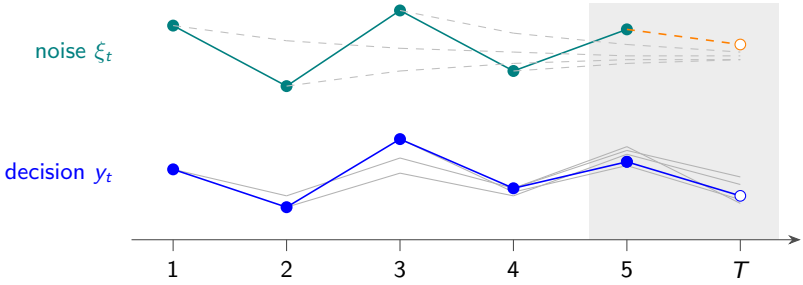


$t = 5$. Two stages are left to predict.

Model predictive control

At stage t, with $y_{[t-1]}$ applied, $\xi_{[t]}$ observed and each later noise at its forecast $\mathbb{E}_{\mathbb{P}}\left[\boldsymbol{\xi}_\tau \mid \boldsymbol{\xi}_{[t]}\right]$, solve over the stages left (a shrinking horizon) and apply its first decision:

$$\min_{y_5,\dots,y_T} \sum_{\tau=5}^{T} \ell_\tau\left(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\right) .$$

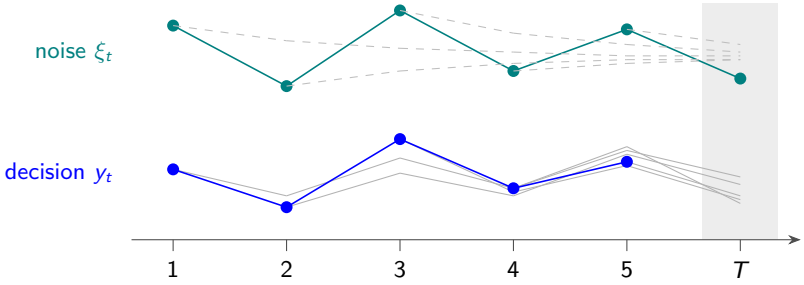


Re-solve. Grey is everything planned and dropped: T plans made, T decisions played.

Model predictive control

At stage t, with $y_{[t-1]}$ applied, $\xi_{[t]}$ observed and each later noise at its forecast $\mathbb{E}_{\mathbb{P}}\left[\boldsymbol{\xi}_\tau \mid \boldsymbol{\xi}_{[t]}\right]$, solve over the stages left (a shrinking horizon) and apply its first decision:

$$\min_{y_T} \sum_{\tau=T}^{T} \ell_\tau\left(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\right) .$$

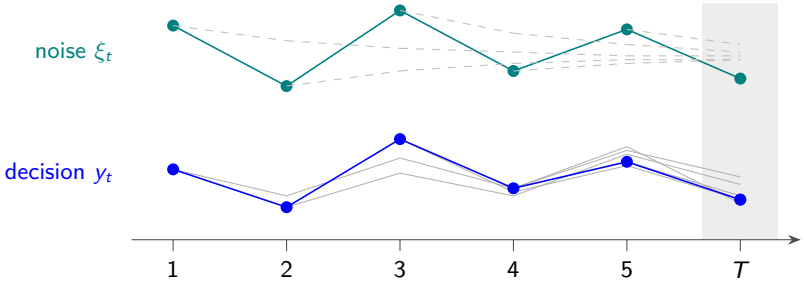


$t = T$. The last noise is observed and there is nothing left to forecast.

Model predictive control

At stage t, with $y_{[t-1]}$ applied, $\xi_{[t]}$ observed and each later noise at its forecast $\mathbb{E}_{\mathbb{P}}\left[\boldsymbol{\xi}_\tau \mid \boldsymbol{\xi}_{[t]}\right]$, solve over the stages left (a shrinking horizon) and apply its first decision:

$$\min_{y_T} \sum_{\tau=T}^{T} \ell_\tau\left(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\right) .$$

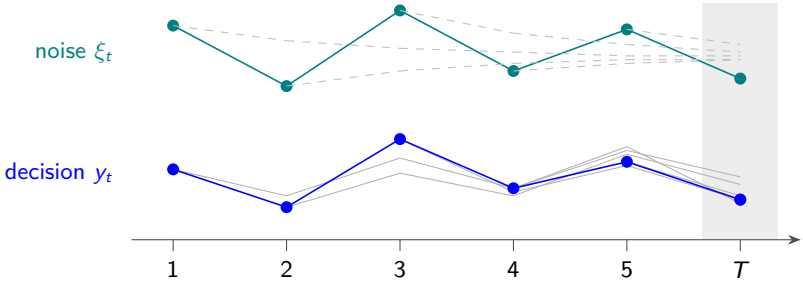


The last window is one stage. Solve it, apply its decision, and the trajectory is complete.

Model predictive control

At stage t, with $y_{[t-1]}$ applied, $\xi_{[t]}$ observed and each later noise at its forecast $\mathbb{E}_{\mathbb{P}}\left[\boldsymbol{\xi}_\tau \mid \boldsymbol{\xi}_{[t]}\right]$, solve over the stages left (a shrinking horizon) and apply its first decision:

$$\min_{y_T} \sum_{\tau=T}^{T} \ell_\tau\left(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\right) .$$

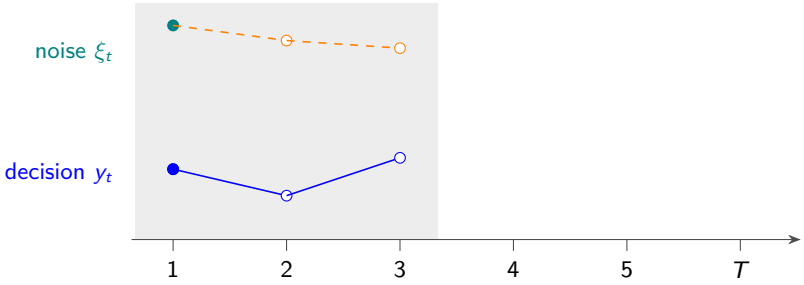


Admissibility. Admissible as long as every window problem is feasible, and nothing guarantees feasibility, not even relatively complete recourse: a forecast is not a value of the noise.

Model predictive control: rolling horizon

The window can instead keep a fixed length H and move forward with t, over $t, \dots, t+H-1$ (here $H = 3$), truncated at T: a rolling horizon.

$$\min_{y_1,\dots,y_3} \sum_{\tau=1}^{3} \ell_\tau\big(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\big) \ .$$

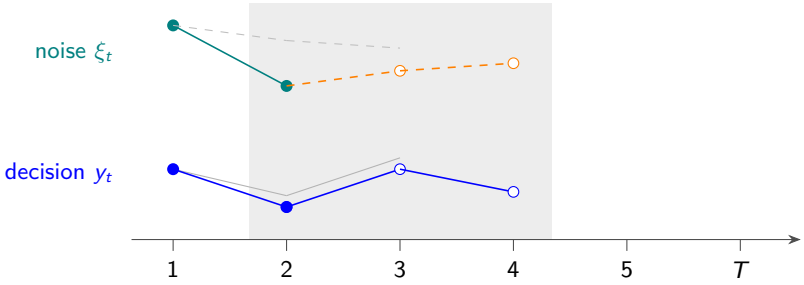


$t = 1$. Three stages are planned, and what happens after stage 3 is not in the problem at all.

Model predictive control: rolling horizon

The window can instead keep a fixed length H and move forward with t, over $t, \dots, t+H-1$ (here $H = 3$), truncated at T: a rolling horizon.

$$\min_{y_2,\dots,y_4} \sum_{\tau=2}^{4} \ell_\tau\big(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\big) \ .$$

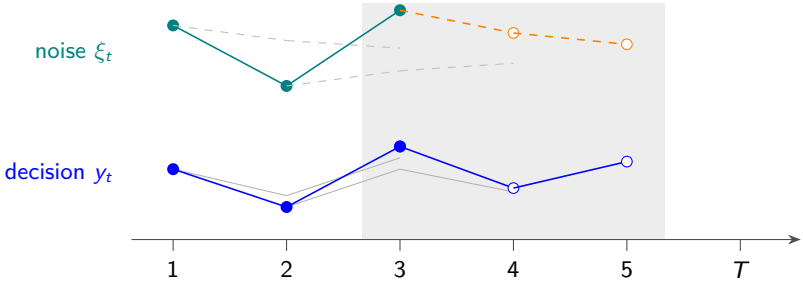


$t = 2$. The window rolls forward instead of shrinking: three stages again, re-solved on what is now known.

Model predictive control: rolling horizon

The window can instead keep a fixed length H and move forward with t, over $t, \dots, t+H-1$ (here $H = 3$), truncated at T: a rolling horizon.

$$\min_{y_3,\dots,y_5} \sum_{\tau=3}^{5} \ell_\tau\big(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\big) \ .$$

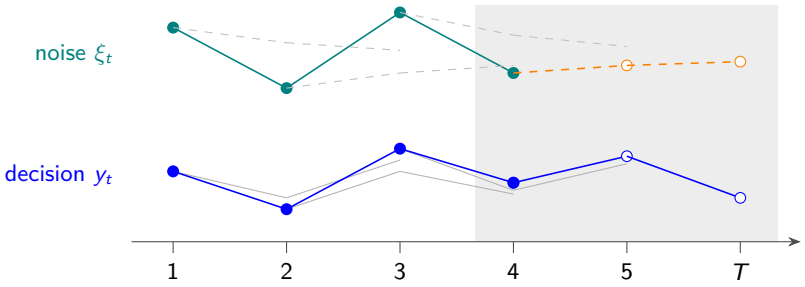


$t = 3$. Every solve has H stages, whatever T is, which is what makes a long horizon tractable.

Model predictive control: rolling horizon

The window can instead keep a fixed length H and move forward with t, over $t, \dots, t+H-1$ (here $H = 3$), truncated at T: a rolling horizon.

$$\min_{y_4,\dots,y_T} \sum_{\tau=4}^{T} \ell_\tau\big(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\big) \ .$$

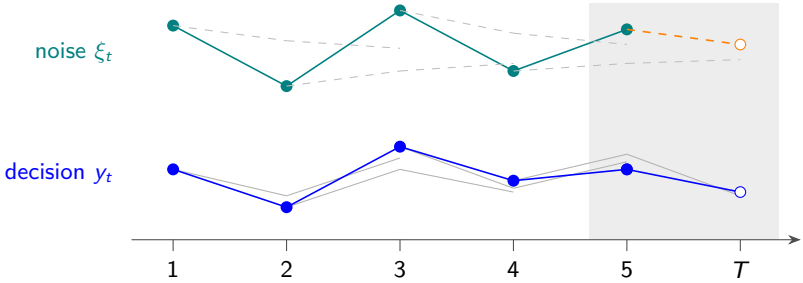


$t = 4$. The blue path is what was played, the grey what was planned and dropped.

Model predictive control: rolling horizon

The window can instead keep a fixed length H and move forward with t, over $t, \dots, t+H-1$ (here $H = 3$), truncated at T: a rolling horizon.

$$\min_{y_5,\dots,y_T} \sum_{\tau=5}^{T} \ell_\tau\big(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\big) \ .$$

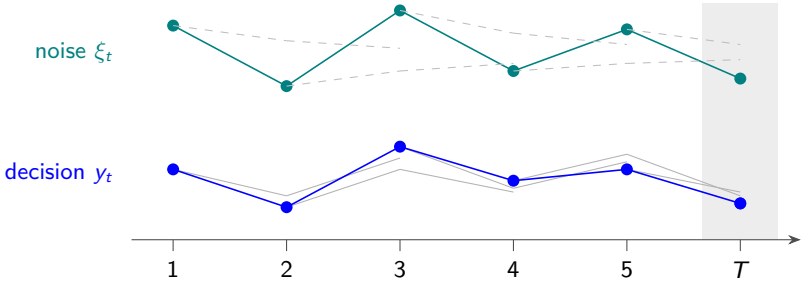


$t = 5$. Near the end the window is truncated at T, and the two horizons meet.

Model predictive control: rolling horizon

The window can instead keep a fixed length H and move forward with t, over $t, \dots, t+H-1$ (here $H = 3$), truncated at T: a rolling horizon.

$$\min_{y_T} \sum_{\tau=T}^{T} \ell_\tau\big(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau\big) \ .$$

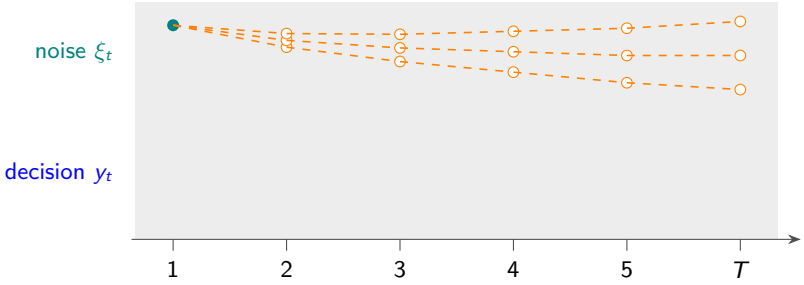


The tail. What lies beyond the window is ignored, unless a value at its end stands in for it: the cost-to-go of the next sessions.

The repeated two-stage heuristic

The same loop, with the two-stage relaxation of what remains in place of the deterministic problem: one decision now, common to every continuation, and hindsight after it.

$$\min_{y_1} \ \ell_1(y_{[0]}, \xi_{[1]}, y_1) + \mathbb{E}_{\mathbb{P}}\left[\min_{y_2,\dots,y_T} \sum_{\tau=2}^{T} \ell_\tau(y_{[\tau-1]}, \boldsymbol{\xi}_{[\tau]}, y_\tau) \ \middle| \ \boldsymbol{\xi}_{[1]} \right].$$

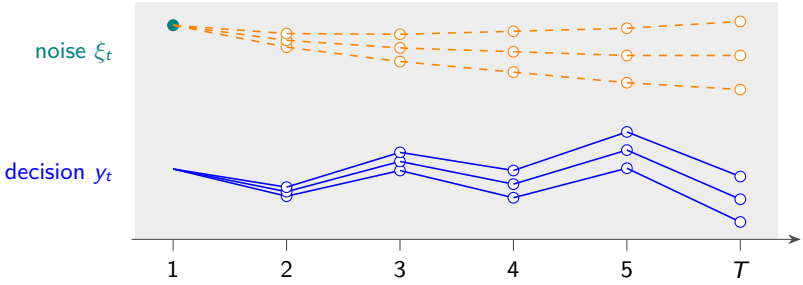


$t = 1$. What follows stage 1 is a conditional law, drawn here as three continuations.

The repeated two-stage heuristic

The same loop, with the two-stage relaxation of what remains in place of the deterministic problem: one decision now, common to every continuation, and hindsight after it.

$$\min_{y_1} \ \ell_1(y_{[0]}, \xi_{[1]}, y_1) + \mathbb{E}_{\mathbb{P}}\left[\min_{y_2,\dots,y_T} \sum_{\tau=2}^{T} \ell_\tau(y_{[\tau-1]}, \boldsymbol{\xi}_{[\tau]}, y_\tau) \ \middle| \ \boldsymbol{\xi}_{[1]} \right].$$

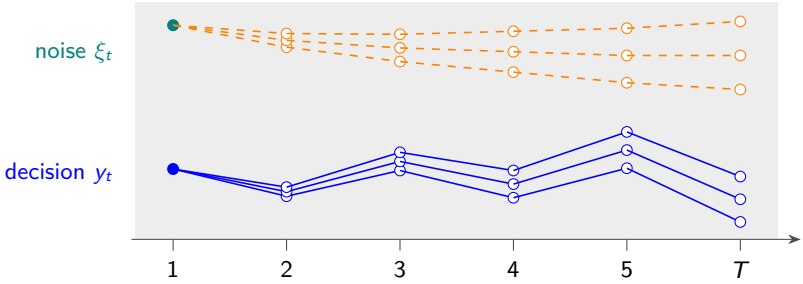


Solve. One decision now, common to every continuation, then a recourse that sees its own continuation to the end.

The repeated two-stage heuristic

The same loop, with the two-stage relaxation of what remains in place of the deterministic problem: one decision now, common to every continuation, and hindsight after it.

$$\min_{y_1} \ \ell_1(y_{[0]}, \xi_{[1]}, y_1) + \mathbb{E}_{\mathbb{P}}\left[\min_{y_2,\dots,y_T} \sum_{\tau=2}^{T} \ell_\tau(y_{[\tau-1]}, \boldsymbol{\xi}_{[\tau]}, y_\tau) \ \middle|\ \boldsymbol{\xi}_{[1]} \right].$$

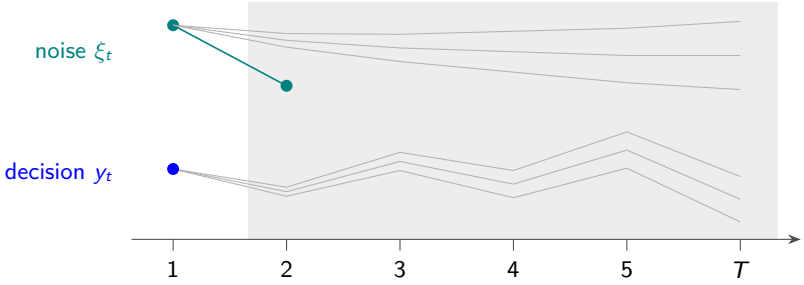


Apply, then discard. Only the common decision is played; every recourse is thrown away.

The repeated two-stage heuristic

The same loop, with the two-stage relaxation of what remains in place of the deterministic problem: one decision now, common to every continuation, and hindsight after it.

$$\min_{y_2} \ \ell_2(y_{[1]}, \xi_{[2]}, y_2) + \mathbb{E}_{\mathbb{P}}\left[\min_{y_3,\dots,y_T} \sum_{\tau=3}^{T} \ell_\tau(y_{[\tau-1]}, \boldsymbol{\xi}_{[\tau]}, y_\tau) \ \middle|\ \boldsymbol{\xi}_{[2]} \right].$$

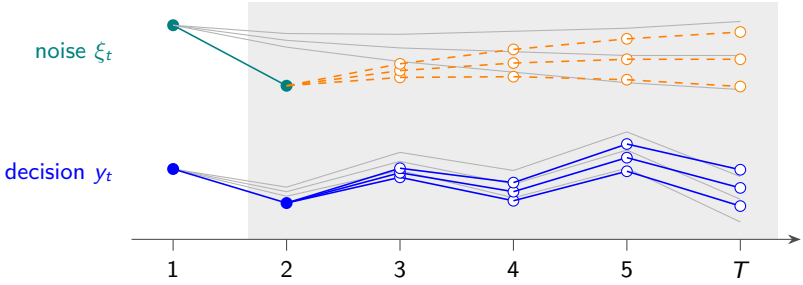


$t = 2$. The noise is observed, and the fan drawn at stage 1 goes with it.

The repeated two-stage heuristic

The same loop, with the two-stage relaxation of what remains in place of the deterministic problem: one decision now, common to every continuation, and hindsight after it.

$$\min_{y_2} \ \ell_2(y_{[1]}, \xi_{[2]}, y_2) + \mathbb{E}_{\mathbb{P}}\left[\min_{y_3,\dots,y_T} \sum_{\tau=3}^{T} \ell_\tau(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau) \ \middle|\ \xi_{[2]} \right].$$

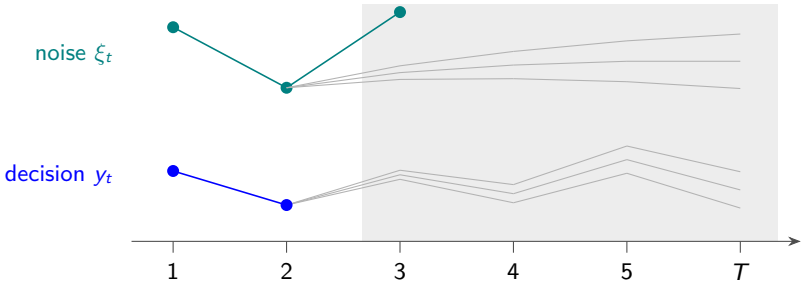


Re-solve. A new fan, under the law conditional on what is now known (sampling it is allowed, if the draw is conditional).

The repeated two-stage heuristic

The same loop, with the two-stage relaxation of what remains in place of the deterministic problem: one decision now, common to every continuation, and hindsight after it.

$$\min_{y_3} \ \ell_3(y_{[2]}, \xi_{[3]}, y_3) + \mathbb{E}_{\mathbb{P}}\left[\min_{y_4,\dots,y_T} \sum_{\tau=4}^{T} \ell_\tau(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau) \ \middle|\ \xi_{[3]} \right].$$

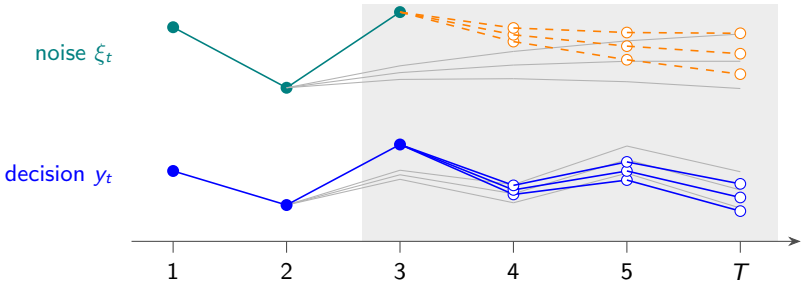


$t = 3$. The next noise, and another fan to draw.

The repeated two-stage heuristic

The same loop, with the two-stage relaxation of what remains in place of the deterministic problem: one decision now, common to every continuation, and hindsight after it.

$$\min_{y_3} \ \ell_3(y_{[2]}, \xi_{[3]}, y_3) + \mathbb{E}_{\mathbb{P}}\left[\min_{y_4,\dots,y_T} \sum_{\tau=4}^{T} \ell_\tau(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau) \ \middle|\ \xi_{[3]} \right].$$

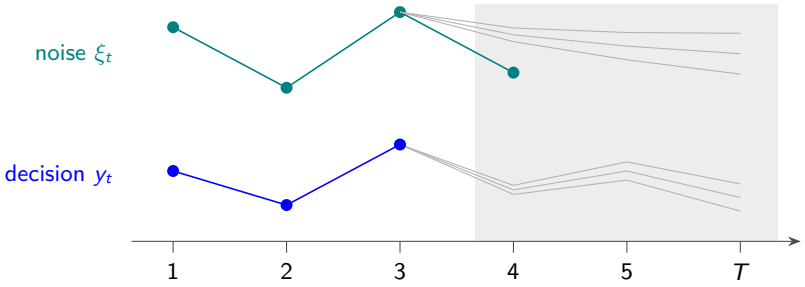


Re-solve. Each fan still runs to T, so it is one stage shorter than the one before.

The repeated two-stage heuristic

The same loop, with the two-stage relaxation of what remains in place of the deterministic problem: one decision now, common to every continuation, and hindsight after it.

$$\min_{y_4} \ \ell_4(y_{[3]}, \xi_{[4]}, y_4) + \mathbb{E}_{\mathbb{P}}\left[\min_{y_5,\dots,y_T} \sum_{\tau=5}^{T} \ell_\tau(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau) \ \middle|\ \xi_{[4]} \right].$$

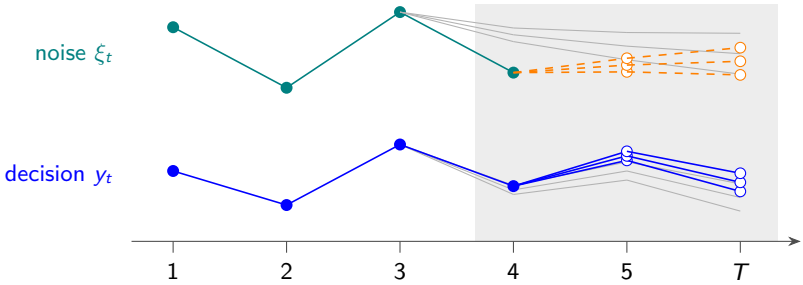


$t = 4$. Observed, and the fan goes grey with it.

The repeated two-stage heuristic

The same loop, with the two-stage relaxation of what remains in place of the deterministic problem: one decision now, common to every continuation, and hindsight after it.

$$\min_{y_4} \ \ell_4(y_{[3]}, \xi_{[4]}, y_4) + \mathbb{E}_{\mathbb{P}}\left[\min_{y_5,\dots,y_T} \sum_{\tau=5}^{T} \ell_\tau(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau) \ \middle|\ \xi_{[4]} \right].$$

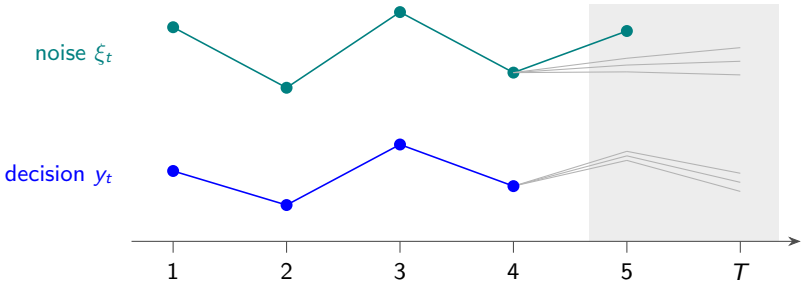


Re-solve. In grey, the fan of the stage before: every recourse computed there, and never played.

The repeated two-stage heuristic

The same loop, with the two-stage relaxation of what remains in place of the deterministic problem: one decision now, common to every continuation, and hindsight after it.

$$\min_{y_5} \ \ell_5(y_{[4]}, \xi_{[5]}, y_5) + \mathbb{E}_{\mathbb{P}}\left[\min_{y_T} \sum_{\tau=T}^{T} \ell_\tau(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau) \ \middle| \ \xi_{[5]} \right].$$

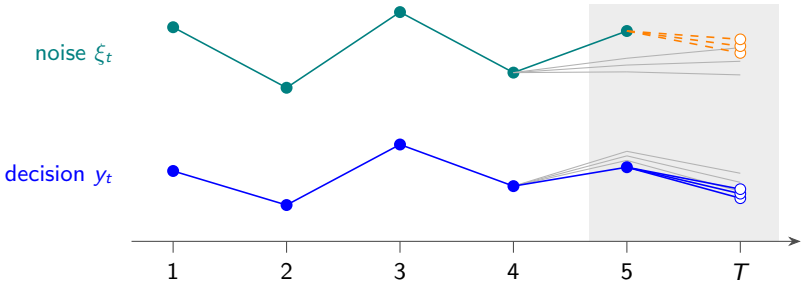


$t = 5$. Two stages are left.

The repeated two-stage heuristic

The same loop, with the two-stage relaxation of what remains in place of the deterministic problem: one decision now, common to every continuation, and hindsight after it.

$$\min_{y_5} \ \ell_5(y_{[4]}, \xi_{[5]}, y_5) + \mathbb{E}_{\mathbb{P}}\left[\min_{y_T} \sum_{\tau=T}^{T} \ell_\tau(y_{[\tau-1]}, \xi_{[\tau]}, y_\tau) \ \middle| \ \xi_{[5]} \right].$$

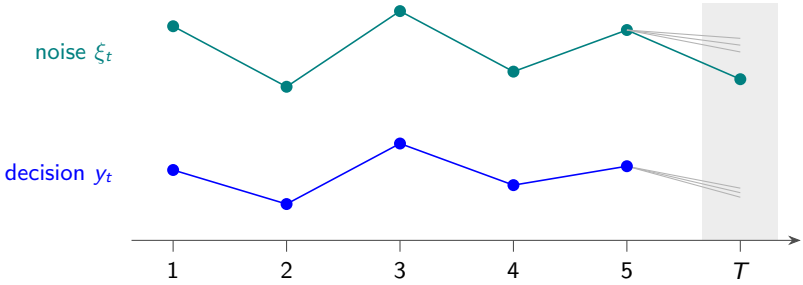


Re-solve. The blue path at the bottom is all that was ever applied.

The repeated two-stage heuristic

The same loop, with the two-stage relaxation of what remains in place of the deterministic problem: one decision now, common to every continuation, and hindsight after it.

$$\min_{y_T} \ \ell_T\big(y_{[T-1]}, \xi_{[T]}, y_T\big) \ .$$

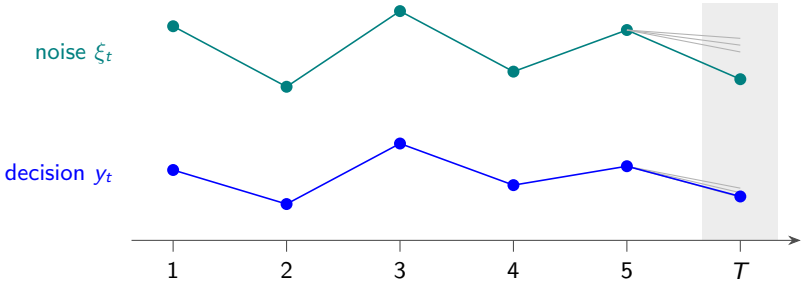


$t = T$. The last noise is observed and there is no continuation left.

The repeated two-stage heuristic

The same loop, with the two-stage relaxation of what remains in place of the deterministic problem: one decision now, common to every continuation, and hindsight after it.

$$\min_{y_T} \ \ell_T\big(y_{[T-1]}, \xi_{[T]}, y_T\big) \ .$$

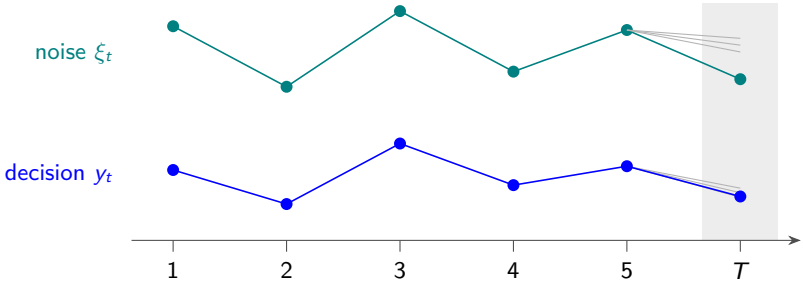


The last problem is one stage, with no recourse after it.

The repeated two-stage heuristic

The same loop, with the two-stage relaxation of what remains in place of the deterministic problem: one decision now, common to every continuation, and hindsight after it.

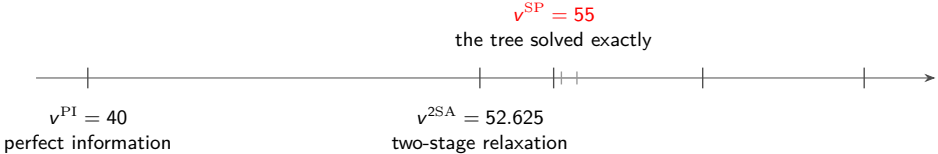
$$\min_{y_T} \ \ell_T\big(y_{[T-1]}, \xi_{[T]}, y_T\big) \ .$$



Still optimistic. Only the decision taken now faces an unknown future; every later one is answered with hindsight. Admissible under relatively complete recourse.

Closing the bracket on the shipping tree

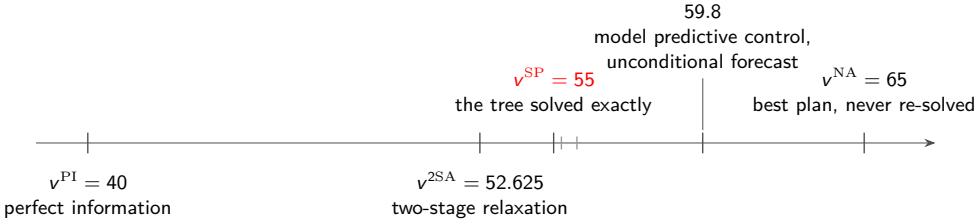
This session's methods, all run on the three-stage tree of the opening frame.



Below the line, the bounds. Each relaxes the information constraint, and each is one solve per leaf or one deterministic problem.

Closing the bracket on the shipping tree

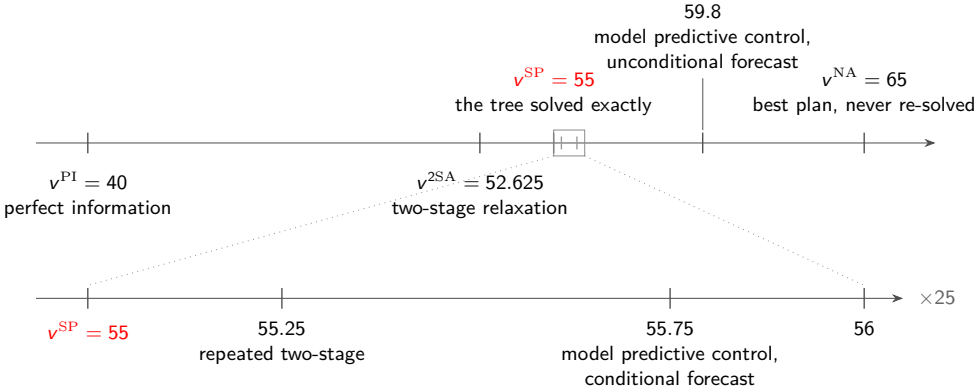
This session's methods, all run on the three-stage tree of the opening frame.



Above it, policies. A heuristic becomes an upper bound only once it is simulated on the tree, decision by decision.

Closing the bracket on the shipping tree

This session's methods, all run on the three-stage tree of the opening frame.



Both heuristics land within 1.5 **per cent of the optimum**, where the plan that is never re-solved is 18 per cent above it.

What you must remember

- Start by setting the information structure: what is known when a decision is taken.
- Evaluate the number of scenarios: beyond a handful of stages only bounds and heuristics are practical.
- Judge a heuristic by the policy it defines, admissible at every node and simulated.

What you should remember

- The information at a stage is the history of the noise, and a decision allowed to depend on it is a function of it.
- Widening what a decision may depend on relaxes the problem, stage by stage.
- A decision taken before its noise must meet its constraint at every value the noise can still take.
- Both decompositions extend to a tree (progressive hedging averages node by node, nested Benders builds cuts on the cost-to-go of each node), with limited scalability.