Decomposition techniques for stochastic optimal control problems

COPI'08

Pierre Carpentier[2], Pierre Girardeau[1,2,3]

[1]EDF R&D, France

[2]ENSTA ParisTech, France

[3]École des Ponts ParisTech, France

November 26, 2008

Outline

Outline

Introduction (1)

The problem we consider

$$\left\{\begin{array}{lll} \min\limits_{\mathbf{X},\mathbf{U}} & \mathbb{E}\left(\sum_{t=1}^{T-1}\sum_{i=1}^{n} L_t^i\left(\mathbf{X}_t^i,\mathbf{U}_t^i,\boldsymbol{\xi}_{t+1}\right) + \sum_{i=1}^{n} K^i\left(\mathbf{X}_T^i\right)\right) & \\ \text{s.t.} & \mathbf{X}_{t+1}^i = f_t^i\left(\mathbf{X}_t^i,\mathbf{U}_t^i,\boldsymbol{\xi}_{t+1}\right), & \boxed{\text{independent dynamics}} \\ & \sum_{i=1}^{n}\mathbf{U}_t^i = 0, & \boxed{\text{demand constraint (would be the same with } \mathbf{D}_t\ \mathcal{F}_t\text{-meas.)}} \\ & \mathbf{U}_t \text{ is } \mathcal{F}_t\text{-measurable}, & \boxed{\text{non-anticipativity constraints}} \\ & + \text{ bound constraints on } \mathbf{X} \text{ and } \mathbf{U}, & \end{array}\right.$$

with $\mathcal{F}_t = \sigma\left\{\boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_t\right\}$ the information available at time t.

$\hookrightarrow$ Application to power production management:

- n **power units** with stock $\mathbf{X}_t^i$ and production $\mathbf{U}_t^i$;
- the power producer has to supply a **global power demand**;
- the objective is to **minimize the production cost**.

Introduction (2)

- Consider a system that evolves through time depending on **random variables**.
- The objective is to **minimize a cost function** depending on the state of the system.

The context is the following:

- the system is made of **subsystems with independent dynamics**;
- the **cost is separable** with respect to the subsystems;
- the time is discrete, possibly with a **large number of time steps**;
- the **state dimension** of the system **is large**, but **each subsystem is small**;
- subsystems are linked by a **static coupling constraint** at each time step;

Introduction (2)

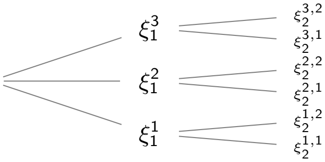
- Consider a system that evolves through time depending on **random variables**.
- The objective is to **minimize a cost function** depending on the state of the system.

The context is the following:

- the system is made of **subsystems with independent dynamics**;
- the **cost is separable** with respect to the subsystems;
- the time is discrete, possibly with a **large number of time steps**;
- the **state dimension** of the system **is large**, but **each subsystem is small**;
- subsystems are linked by a **static coupling constraint** at each time step;

Scenario trees

- When dealing with large-scale stochastic optimal control problems, one can **“discretize randomness”** using a scenario tree [Shapiro and Ruszczynski, 2003, Higle and Sen, 1996].



- One can then use some **deterministic techniques to decompose** the problem.

↪ **randomness representation**: exponential growth of the tree with respect to the time horizon;

↪ **feedback synthesis/interpolation**: how to build a control function?

Dynamic programming

When $\boldsymbol{\xi}_t$ is a **white noise process**, one can introduce the **value function** V and solve the **dynamic programming equation** [Bellman, 1957, Bertsekas, 2000]:

The finite horizon dynamic programming equation

$$V_T(x) = \sum_{i=1}^{n} K^i\left(x^i\right),$$

$$V_t(x) = \begin{cases} \min_u & \mathbb{E}\left(\sum_{i=1}^{n} L_t^i\left(x^i, u^i, \boldsymbol{\xi}_{t+1}\right) + V_{t+1}\left(f_t\left(x, u, \boldsymbol{\xi}_{t+1}\right)\right)\right), \\ \text{s.t.} & \sum_{i=1}^{n} u^i = 0. \end{cases}$$

Advantage: **feedback functions** are available.

Problem: **curse of dimensionality**.

Outline

Duality (1)

One can always write:

$$\min_{\mathbf{X},\mathbf{U}} \max_{\boldsymbol{\lambda}} \quad \mathbb{E}\left(\sum_{t=0}^{T-1}\left(\sum_{i=1}^{n} L_t^i\left(\mathbf{X}_t^i, \mathbf{U}_t^i, \boldsymbol{\xi}_{t+1}\right) - \boldsymbol{\lambda}_t \sum_{i=1}^{n} \mathbf{U}_t^i\right) + \sum_{i=1}^{n} K^i\left(\mathbf{X}_T^i\right)\right),$$

$$\text{s.t.} \quad \mathbf{X}_{t+1}^i = f_t^i\left(\mathbf{X}_t^i, \mathbf{U}_t^i, \boldsymbol{\xi}_{t+1}\right),$$

$$\mathbf{U}_t \text{ is } \mathcal{F}_t\text{-measurable.}$$

$\hookrightarrow$ **The Lagrange multiplier $\boldsymbol{\lambda}_t$ is a $\mathcal{F}_t$-measurable random variable.**

Duality (2)

When the Lagrangian has a saddle point, we can **interchange minimization and maximization**:

$$\max_{\boldsymbol{\lambda}} \min_{\mathbf{X},\mathbf{U}} \quad \mathbb{E}\left(\sum_{t=0}^{T-1}\left(\sum_{i=1}^{n} L_t^i\left(\mathbf{X}_t^i,\mathbf{U}_t^i,\boldsymbol{\xi}_{t+1}\right) - \boldsymbol{\lambda}_t \sum_{i=1}^{n}\mathbf{U}_t^i\right) + \sum_{i=1}^{n} K^i\left(\mathbf{X}_T^i\right)\right),$$

$$\text{s.t.} \quad \mathbf{X}_{t+1}^i = f_t^i\left(\mathbf{X}_t^i,\mathbf{U}_t^i,\boldsymbol{\xi}_{t+1}\right),$$

$$\mathbf{U}_t \text{ is } \mathcal{F}_t\text{-measurable.}$$

Duality (3)

With $\boldsymbol{\lambda}$ fixed, the minimization problem becomes decomposable:

$$\max_{\boldsymbol{\lambda}} \quad \left[\sum_{i=1}^{n} \min_{\mathbf{X}^i,\mathbf{U}^i} \mathbb{E}\left(\sum_{t=0}^{T-1}\left(L_t^i\left(\mathbf{X}_t^i,\mathbf{U}_t^i,\boldsymbol{\xi}_{t+1}\right) - \boldsymbol{\lambda}_t \mathbf{U}_t^i\right) + K^i\left(\mathbf{X}_T^i\right)\right)\right],$$

$$\text{s.t.} \quad \mathbf{X}_{t+1}^i = f_t^i\left(\mathbf{X}_t^i,\mathbf{U}_t^i,\boldsymbol{\xi}_{t+1}\right),$$

$$\mathbf{U}_t \text{ is } \mathcal{F}_t\text{-measurable.}$$

- We would like to use an **Uzawa algorithm**.
  - $\hookrightarrow$ we need to perform a gradient algorithm on random variables.
- Dynamic programming with state $\mathbf{X}_t^i$ would apply to the subproblems if $\boldsymbol{\lambda}$ were a **white noise** process.
  - $\hookrightarrow$ We have no information on the time correlations of $\boldsymbol{\lambda}$.

A new hope

[Strugarek, 2006] shows, on a specific instance of the problem, that **prices follow forward dynamics**:

$$\boldsymbol{\lambda}_{t+1} = \boldsymbol{\lambda}_t + \gamma \Big[(1+\alpha)\,\mathbf{D}_{t+1} - \mathbf{D}_t - \alpha\mathbb{E}\left(\mathbf{D}_{t+1}\right) - \alpha\left(\boldsymbol{\xi}_{t+1} - \mathbb{E}\left(\boldsymbol{\xi}_{t+1}\right)\right) \Big],$$

with $\mathbf{D}_t$ being a part of the noise.
$\hookrightarrow$ If we add $\mathbf{D}_t$ and λ_t in the state, subproblems can be solved using **dynamic programming in dimension 3**.

A new hope

[Strugarek, 2006] shows, on a specific instance of the problem, that **prices follow forward dynamics**:

$$\boldsymbol{\lambda}_{t+1} = \boldsymbol{\lambda}_t + \gamma \Big[(1+\alpha)\, \mathbf{D}_{t+1} - \mathbf{D}_t - \alpha \mathbb{E}\left(\mathbf{D}_{t+1}\right) - \alpha \left(\boldsymbol{\xi}_{t+1} - \mathbb{E}\left(\boldsymbol{\xi}_{t+1}\right)\right) \Big],$$

with $\mathbf{D}_t$ being a part of the noise.
$\hookrightarrow$ If we add $\mathbf{D}_t$ and $\boldsymbol{\lambda}_t$ in the state, subproblems can be solved using **dynamic programming in dimension 3**.

Prices dynamics

On more complicated examples, our heuristic consists in **choosing their shape a priori** and solve:

The i-th subproblem

$$\min_{\mathbf{X}^i,\mathbf{U}^i} \quad \mathbb{E}\left(\sum_{t=0}^{T-1}\left(L_t^i\left(\mathbf{X}_t^i,\mathbf{U}_t^i,\boldsymbol{\xi}_{t+1}\right)-\boldsymbol{\lambda}_t\mathbf{U}_t^i\right)+K^i\left(\mathbf{X}_T^i\right)\right),$$

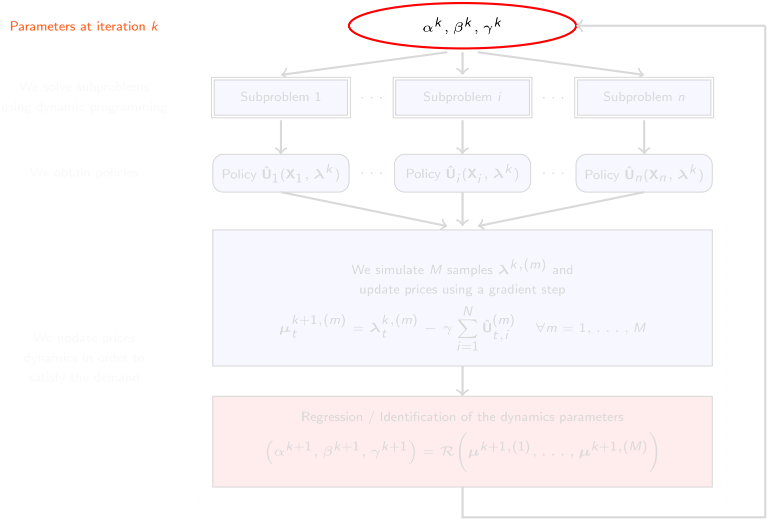
$$\text{s.t.} \quad \mathbf{X}_{t+1}^i = f_t^i\left(\mathbf{X}_t^i,\mathbf{U}_t^i,\boldsymbol{\xi}_{t+1}\right),$$

$$\boldsymbol{\lambda}_{t+1} = \alpha_t\boldsymbol{\lambda}_t + \beta_t\boldsymbol{\xi}_{t+1} + \gamma_t,$$

$\mathbf{U}_t^i$ is $\mathcal{F}_t$-measurable.

The algorithm consists in **iterating on the parameters** α, β, γ of these dynamics.

The algorithm



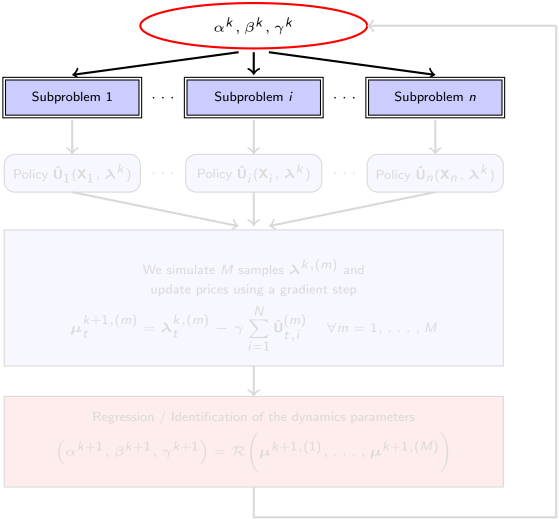
The algorithm

Parameters at iteration k

We solve subproblems using dynamic programming

We obtain policies

We update prices dynamics in order to satisfy the demand



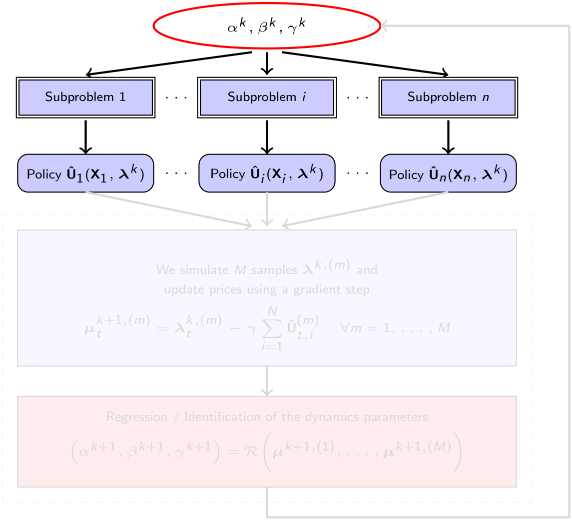
The algorithm

Parameters at iteration k

We solve subproblems using dynamic programming

We obtain policies

We update prices dynamics in order to satisfy the demand



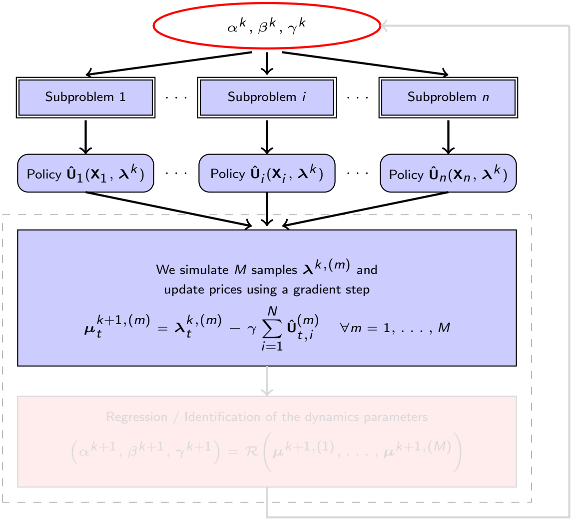
The algorithm

Parameters at iteration k

We solve subproblems using dynamic programming

We obtain policies

We update prices dynamics in order to satisfy the demand



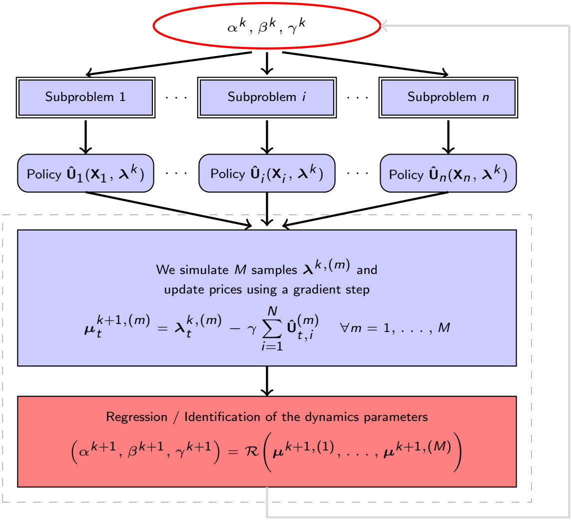
The algorithm

Parameters at iteration k

We solve subproblems using dynamic programming

We obtain policies

We update prices dynamics in order to satisfy the demand



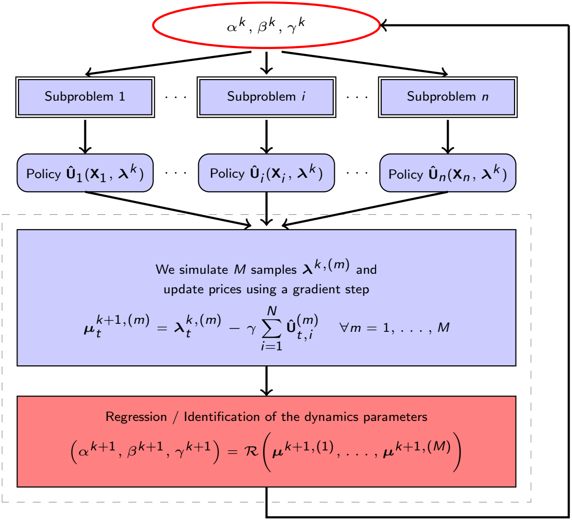
The algorithm

At iteration $k+1$

We solve subproblems using dynamic programming

We obtain policies

We update prices dynamics in order to satisfy the demand



Remark

Our heuristic can be seen as a **projected gradient algorithm on dual variables λ**.

↪ Since we restrict the dual space, the **coupling constraint may be unsatisfied**.

We replace a n-dimensional DP equation by n lower dimensional DP equations, but:

↪ The more we **enhance the prices dynamics, the higher the dimension of the subproblems' state** becomes.

Remark

Our heuristic can be seen as a **projected gradient algorithm on dual variables λ**.

↪ Since we restrict the dual space, the **coupling constraint may be unsatisfied**.

We replace a n-dimensional DP equation by n lower dimensional DP equations, but:

↪ The more we **enhance the prices dynamics, the higher the dimension of the subproblems' state** becomes.

Outline

The test case

We present a numerical experiment [Barty et al., 2008] on a **power production management** problem with:

- two hydraulic plants;
- one thermal plant with quadratic cost;
- power demand and water inflows are stochastic;
- time horizon is 2 years.

We apply:

- dynamic programming, which will be the reference solution;
- the decomposition method.

Random variables (1)

- We consider only white noises.

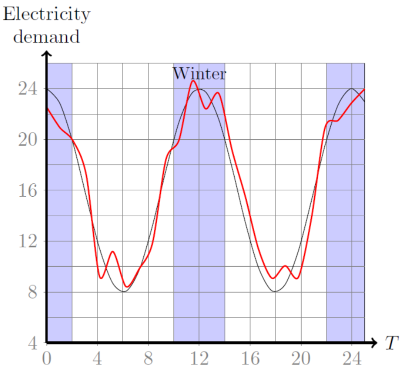


Figure: A drawing of the demand.

Random variables (2)

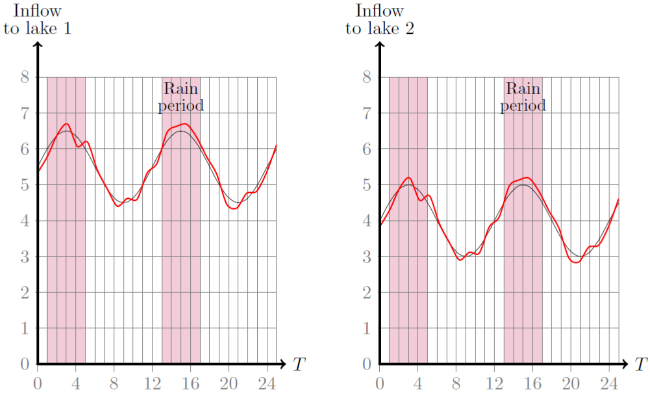


Figure: A drawing of the inflows.

Cost function

	Average cost	95% confidence interval
Dynamic programming (ref)	1371.7	$\pm$ 1.7
Decomposition	1384.6	$\pm$ 1.8

Cost along with iterations

We observe how average cost decreases along with the iterations in the decomposition method.

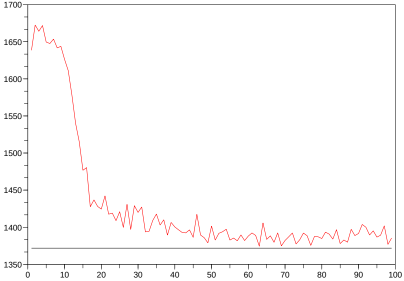
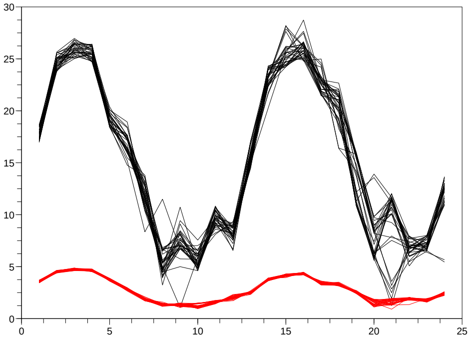


Figure: Red: decomposition method, black: dynamic programming (ref).

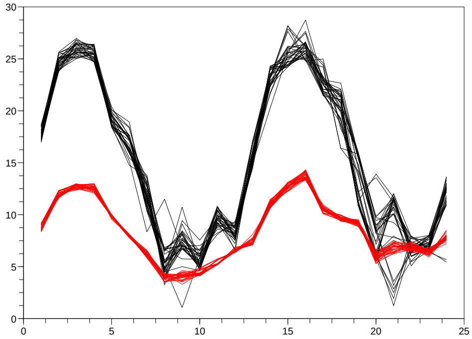
How prices converge to the optimal price

Iteration 1



- Optimal prices (black curves) are computed via dynamic programming.
- Instabilities in this curve come from numerical differentiation of the estimated Bellman functions.

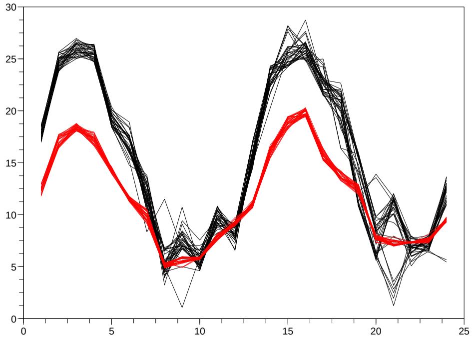
How prices converge to the optimal price
Iteration 10



- Optimal prices (black curves) are computed via dynamic programming.
- Instabilities in this curve come from numerical differentiation of the estimated Bellman functions.

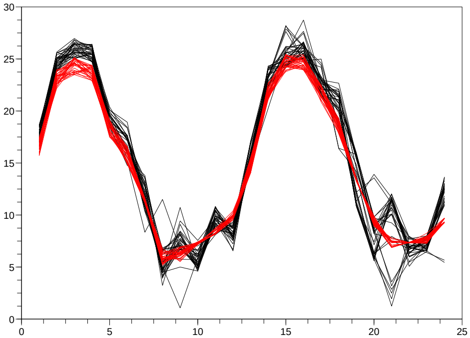
How prices converge to the optimal price

Iteration 20



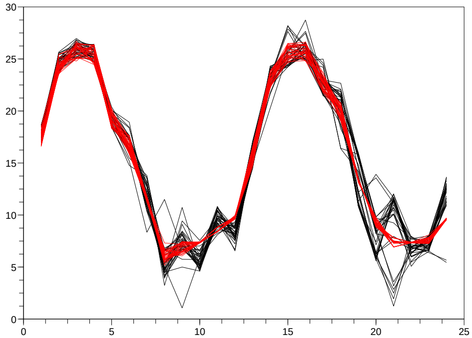
- Optimal prices (black curves) are computed via dynamic programming.
- Instabilities in this curve come from numerical differentiation of the estimated Bellman functions.

How prices converge to the optimal price
Iteration 50



- Optimal prices (black curves) are computed via dynamic programming.
- Instabilities in this curve come from numerical differentiation of the estimated Bellman functions.

How prices converge to the optimal price
Iteration 90



- Optimal prices (black curves) are computed via dynamic programming.
- Instabilities in this curve come from numerical differentiation of the estimated Bellman functions.

Conclusion

We developed a **decomposition heuristic for large-scale stochastic** optimal control problems where:

- units have independent dynamics;
- they are coupled by a static contraint;

↪ this specific kind of problem is **common in production/portfolio management**.

with which we obtained **promising experimental results**.

Much work to do:

- theoretical justification;
- how to identify the good shape for the dynamics?
- experiments on real-life problems.

Conclusion

We developed a **decomposition heuristic for large-scale stochastic** optimal control problems where:

- units have independent dynamics;
- they are coupled by a static contraint;

$\hookrightarrow$ this specific kind of problem is **common in production/portfolio management**.

with which we obtained **promising experimental results**.

Much work to do:

- theoretical justification;
- how to identify the good shape for the dynamics?
- experiments on real-life problems.

Bibliography

Barty, K., Girardeau, P., and Lahirigoyen, P. (2008).
Entre programmation dynamique et décomposition par les prix dans le cadre stochastique : une méthode hybride.
Rapport interne EDF R&D, H-R35-2008-03799-FR.

Bellman, R. (1957).
Dynamic Programming.
Princeton University Press, New Jersey.

Bertsekas, D. P. (2000).
Dynamic Programming and Optimal Control, volume 1 & 2.
Athena Scientific, 2 edition.

Higle, J. and Sen, S. (1996).
Stochastic Decomposition.
Kluwer, Dordrecht.

Shapiro, A. and Ruszczynski, A., editors (2003).
Stochastic Programming.
Elsevier, Amsterdam.

Strugarek, C. (2006).
Approches variationnelles et autres contributions en optimisation stochastique.
Thèse de doctorat, École Nationale des Ponts et Chaussées.

How do we perform the regression?

Given M samples at each time step, we would like to solve:

$$\min_{\alpha,\beta,\gamma} \quad \sum_{t=1}^{T-1} \sum_{m=1}^{M} (\mu_t^m - \lambda_t^m)^2$$

$$\text{s.t.} \quad \lambda_{t+1}^m = \alpha_t \lambda_t^m + \beta_t \xi_{t+1}^m + \gamma_t,$$

$$\lambda_1^m = \beta_0 \xi_1^m + \gamma_0.$$

For complexity reasons, we solve the following approximate regression problem:

$$\min_{\alpha,\beta,\gamma} \quad \sum_{t=1}^{T-1} \sum_{m=1}^{M} (\mu_t^m - \lambda_t^m)^2$$

$$\text{s.t.} \quad \lambda_{t+1}^m = \alpha_t \mu_t^m + \beta_t \xi_{t+1}^m + \gamma_t,$$

$$\lambda_1^m = \beta_0 \xi_1^m + \gamma_0.$$

that is separable through time.

How do we perform the regression?

Given M samples at each time step, we would like to solve:

$$\begin{aligned} \min_{\alpha,\beta,\gamma} \quad & \sum_{t=1}^{T-1} \sum_{m=1}^{M} (\mu_t^m - \lambda_t^m)^2 \\ \text{s.t.} \quad & \lambda_{t+1}^m = \alpha_t \lambda_t^m + \beta_t \xi_{t+1}^m + \gamma_t, \\ & \lambda_1^m = \beta_0 \xi_1^m + \gamma_0. \end{aligned}$$

For complexity reasons, we solve the following approximate regression problem:

$$\begin{aligned} \min_{\alpha,\beta,\gamma} \quad & \sum_{t=1}^{T-1} \sum_{m=1}^{M} (\mu_t^m - \lambda_t^m)^2 \\ \text{s.t.} \quad & \lambda_{t+1}^m = \alpha_t \mu_t^m + \beta_t \xi_{t+1}^m + \gamma_t, \\ & \lambda_1^m = \beta_0 \xi_1^m + \gamma_0. \end{aligned}$$

that is separable through time.

Dynamics we choose for the numerical experiments

- In the experiments, we have water inflows $\boldsymbol{\xi}_t^i$ and power demand $\mathbf{D}_t$.
- Control $\mathbf{U}_t$ observes $\boldsymbol{\xi}_t^i$ and $\mathbf{D}_{t-1}$.

We choose the following shape for the prices' dynamics:

$$\boldsymbol{\lambda}_{t+1} = \alpha_t \boldsymbol{\lambda}_t + \beta_t \mathbf{D}_t + \gamma_t .$$

Maybe it would be relevant to include water inflows or the previous demand.