Programme Gaspard Monge pour l'Optimisation

Large-scale Stochastic Optimization Problems, Space Decomposition and DADP Algorithm

Jean-Christophe Alais

◇

StochDec Project
J.-C. Alais - P. Carpentier - J.-P. Chancelier - S. Charousset
A. Dallagi - M. De Lara - V. Leclère - N. Oudjane

CERMICS (ENPC) - OSIRIS (EDF R&D) - UMA (ENSTA)

March 27, 2013

Motivation


LEGENDE
Usines télécommandées depuis le PCC
(S) Retenue saisonnière
(H) Retenue hebdomadaire
(J) Retenue journalière
Canal usinier
0 10 20 30 km
N
Guisane
Durance
Briançon
Fournel
Embrun
Gap
Serre-Ponçon (S)
Serre-Ponçon
Ubaye
Espinasses (J)
Curbans
Pt Buech
La Saulce
Lazer
Buech
Sisteron
Sisteron
St Lazare (J)
Salignac
Bléone
Digne
Verdon
L'Escale (J)
Oraison
Castellane
La Brillanne
Le Largue
Manosque
Manosque
PCC de Ste Tulle
Ste Tulle
Ste Croix
Ste Croix (S)
Quinson (H)
Beaumont
Cadarache (J)
Vinon
Gréoux (H)
Jouques
Avignon
Rhône
Durance
Mallemort
St Estève Janson
Salon-de-P.
Salon
Arles
St Chamas
Etang de Berre
Aix en Provence
Marseille

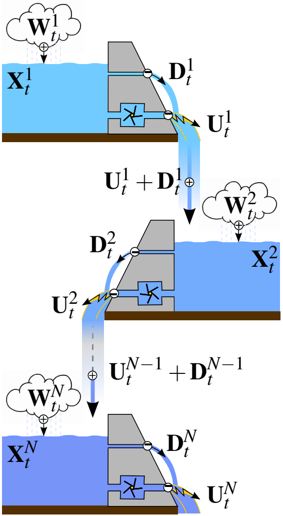
Summary

1. Introductory example: the dams cascade
   - The dams cascade problem
   - Standard ways to solve the problem
2. Decomposition over space and Dual Approximate DP
   - Decomposition/coordination methods, an overview
   - Dual Approximate Dynamic Programming
3. The three dams cascade toy problem
   - Modeling of the three dams cascade problem
   - Algorithm issues and numerical results
4. Conclusion and perspectives

Summary

1. Introductory example: the dams cascade
    - The dams cascade problem
    - Standard ways to solve the problem
2. Decomposition over space and Dual Approximate DP
3. The three dams cascade toy problem
4. Conclusion and perspectives

The dams cascade problem



Optimal management of a dams cascade hydroelectric production by means of a Discrete Time Stochastic Optimal Control Problem: let $1 \leq i \leq N$ and $0 \leq t \leq T$

- state $\mathbf{X}_t^i$: storage level
- noise $\mathbf{W}_t^i$: exogeneous inflows
- control $\mathbf{U}_t^i$: turbinated water
- $\mathbf{D}_t^i$: spilled water surplus

$\longrightarrow$ N state and N control variables

straightforward Dynamic Programming:
untractable as soon as N > 4…

Standard ways to solve the problem: approximate solving

- **Stochastic Programming:**
  model the problem using the scenario tree
  - pros availability of efficient algorithms from Mathematical Programming
  - cons difficulty to discretize the uncertainty as a tractable scenarios tree, no strategy (decisions are attached to the nodes of the tree)
- **Approximate Dynamic Programming**
  - **Aggregation methods**
  - **Stochastic Dual Dynamic Programming:** Bellman function approximation by cuts that are computed iteratively
    - pros efficient for cascade and production - demand equilibrium type problems up to $N = 12$ units
    - cons quite strong assumptions (convexity, linearity) over the cost and the dynamics functions
- **Decomposition/coordination methods**

Summary

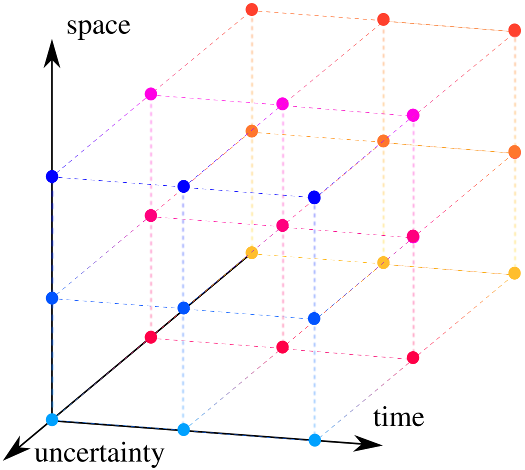
1 Introductory example: the dams cascade

2 Decomposition over space and Dual Approximate DP
- Decomposition/coordination methods, an overview
- Dual Approximate Dynamic Programming

3 The three dams cascade toy problem

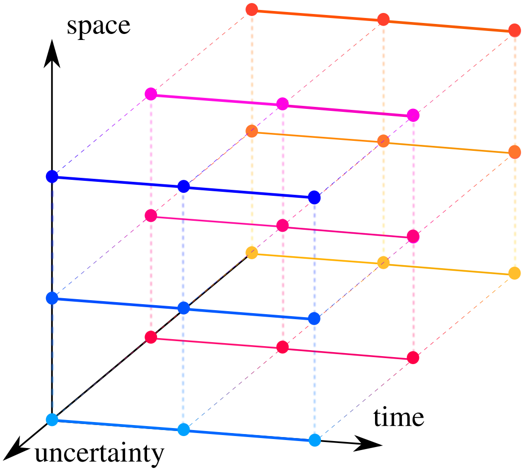
4 Conclusion and perspectives

Decomposition/coordination methods, an overview



$$\min_{\mathbf{X},\mathbf{U}} \sum_{\omega} \sum_{i=1}^{N} \sum_{t=0}^{T} L_t^i\big(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t\big)$$

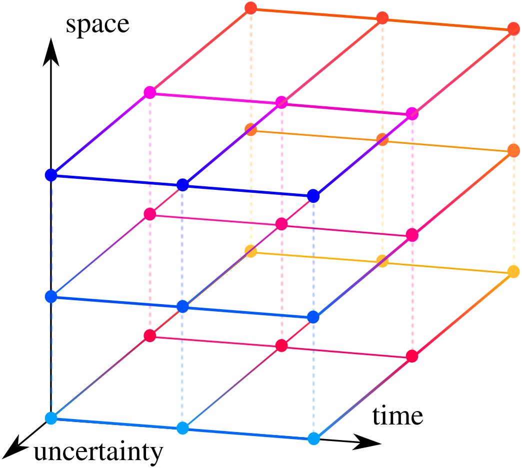
Decomposition/coordination methods, an overview



$$\min_{\mathbf{X},\mathbf{U}} \sum_{\omega} \sum_{i=1}^{N} \sum_{t=0}^{T} L_t^i\big(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t\big)$$

$$\text{s.t.} \quad \mathbf{X}_{t+1}^i = f_t^i\big(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t\big)$$

Decomposition/coordination methods, an overview

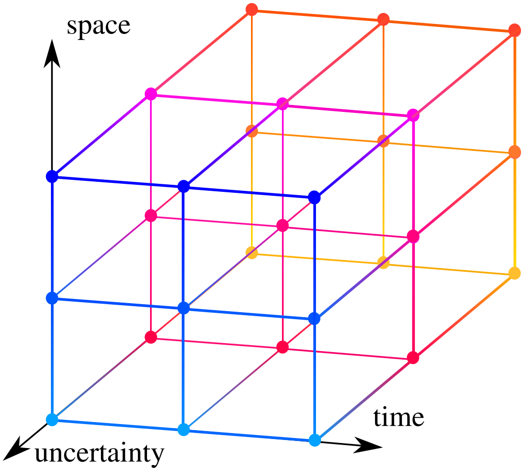


$$\min_{\mathbf{X},\mathbf{U}} \sum_{\omega} \sum_{i=1}^{N} \sum_{t=0}^{T} L_t^i\big(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t\big)$$

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

$$\mathbf{U}_t^i \preceq \mathcal{F}_t$$

Decomposition/coordination methods, an overview



$$
\begin{aligned}
\min_{\mathbf{X},\mathbf{U}} \ & \sum_{\omega} \sum_{i=1}^{N} \sum_{t=0}^{T} L_t^i\big(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t\big) \\
\text{s.t.} \ & \mathbf{X}_{t+1}^i = f_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t) \\
& \mathbf{U}_t^i \preceq \mathcal{F}_t \\
& \sum_{i=1}^{N} \theta_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t) = 0
\end{aligned}
$$

Decomposition/coordination methods, an overview

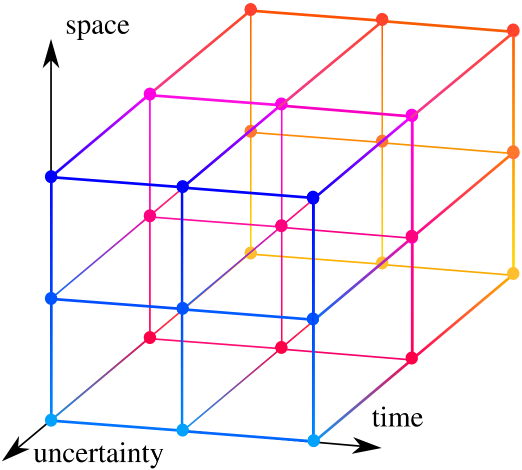
Main idea:

1. decompose a large scale problem into smaller subproblems we are able to solve independently by efficient algorithms
2. coordinate the subproblems for the concatenation of their solutions to form the initial problem solution

How to decompose the problem:

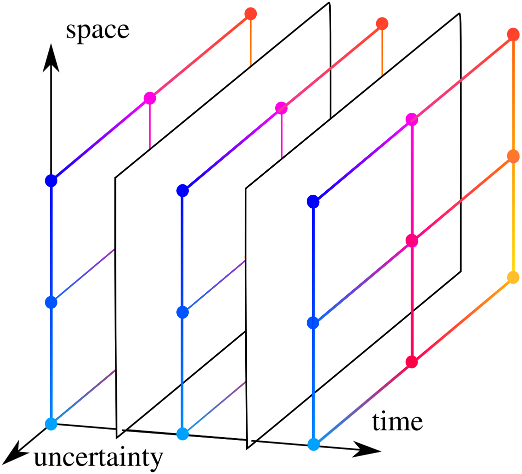
1. identify the coupling dimensions of the problem: *time*, *space* or *uncertainty*
2. dualize the coupling constraints linked to the dimension over which the problem is to be decomposed
3. split the problem into the resulting subproblems

Decomposition/coordination methods, an overview



$$
\begin{aligned}
\min_{\mathbf{X},\mathbf{U}} \ & \mathbb{E}\left(\sum_{i=1}^{N} \sum_{t=0}^{T} L_t^i\left(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t\right)\right) \\
\text{s.t.} \ & \mathbf{X}_{t+1}^i = f_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t) \\
& \mathbf{U}_t^i \preceq \mathcal{F}_t \\
& \sum_{i=1}^{N} \theta_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t) = 0
\end{aligned}
$$

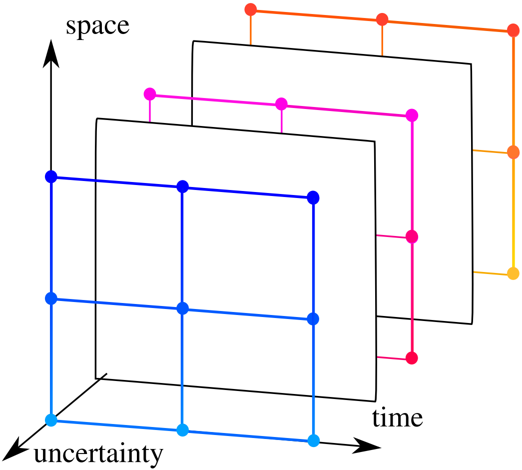
Decomposition/coordination methods, an overview



$$
\begin{aligned}
\min_{\mathbf{X},\mathbf{U}} \; & \mathbb{E}\Big(\sum_{i=1}^{N} \sum_{t=0}^{T} L_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t) \Big) \\
\text{s.t.} \; & \mathbf{X}_{t+1}^i = f_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t) \\
& \mathbf{U}_t^i \preceq \mathcal{F}_t \\
& \sum_{i=1}^{N} \theta_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t) = 0
\end{aligned}
$$

[Stochastic Pontryagin]

Decomposition/coordination methods, an overview



$$\min_{\mathbf{X},\mathbf{U}} \mathbb{E}\left(\sum_{i=1}^{N}\sum_{t=0}^{T} L_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t)\right)$$

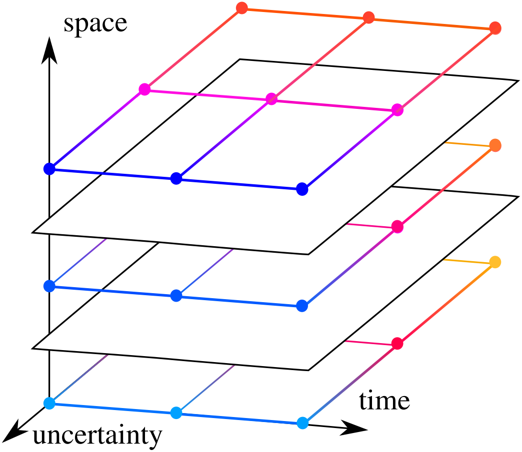
$$\text{s.t.} \quad \mathbf{X}_{t+1}^i = f_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t)$$

$$\mathbf{U}_t^i \preceq \mathcal{F}_t$$

$$\sum_{i=1}^{N} \theta_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t) = 0$$

[Progressive Hedging]

Decomposition/coordination methods, an overview



$$\min_{\mathbf{X},\mathbf{U}} \mathbb{E}\left(\sum_{i=1}^{N} \sum_{t=0}^{T} L_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t) \right)$$

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

$$\mathbf{U}_t^i \preceq \mathcal{F}_t$$

$$\sum_{i=1}^{N} \theta_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t) = 0$$

[Purpose of the talk]

Decomposition/coordination methods, an overview

The SOC problem we are interested in:

$$
\begin{aligned}
\min_{\mathbf{X},\mathbf{U}} \ & \mathbb{E}\left(\sum_{i=1}^{N}\sum_{t=0}^{T} L_t^i\big(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t\big)\right) \\
\text{s.t.} \ & \mathbf{X}_{t+1}^i = f_t^i\big(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t\big) \ \mathbb{P}-a.s. \\
& \mathbf{U}_t^i \preceq \mathcal{F}_t \\
& \sum_{i=1}^{N} \theta_t^i\big(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t\big) = 0 \ \mathbb{P}-a.s.
\end{aligned}
$$

where $\mathcal{F}_t := \sigma\Big(\big(\mathbf{X}_0^i\big)_{i=1,\dots,N}, \dots, \mathbf{W}_0, \dots, \mathbf{W}_t\Big)$

Decomposition/coordination methods, an overview

Let $(\boldsymbol{\lambda}^i)_{i\in\{1,\dots,N\}}$ be a $\mathcal{F}_t$ -adapted process of the coupling constraints multipliers. Problem $(\mathcal{P})$ may read, by dualization:

$$\min_{\substack{\mathbf{X},\,\mathbf{U}\\ \mathbf{U}_t \preceq \mathcal{F}_t}} \max_{\boldsymbol{\lambda}} \quad \mathbb{E}\left(\sum_{i=1}^{N}\sum_{t=0}^{T} L_t^i\big(\mathbf{X}_t^i,\,\mathbf{U}_t^i,\,\mathbf{W}_t\big) + \langle \boldsymbol{\lambda}_t^i,\, \theta_t^i(\mathbf{X}_t^i,\,\mathbf{U}_t^i,\,\mathbf{W}_t)\rangle\right)$$

$$\text{s.t.} \quad \mathbf{X}_{t+1}^i = f_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t), \ \forall i \ \ \mathbb{P}-a.s.$$

Assuming the existence of a saddle point, we can exchange the min and max operators:

$$\max_{\boldsymbol{\lambda}} \quad \sum_{i=1}^{N} \quad \min_{\substack{\mathbf{X}^i,\,\mathbf{U}^i\\ \mathbf{U}_t^i \preceq \mathcal{F}_t}} \quad \mathbb{E}\left(\sum_{t=0}^{T} L_t^i\big(\mathbf{X}_t^i,\,\mathbf{U}_t^i,\,\mathbf{W}_t\big) + \langle \boldsymbol{\lambda}_t^i,\, \theta_t^i(\mathbf{X}_t^i,\,\mathbf{U}_t^i,\,\mathbf{W}_t)\rangle\right)$$

$$\text{s.t.} \quad \mathbf{X}_{t+1}^i = f_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t) \ \ \mathbb{P}-a.s.$$

Decomposition/coordination methods, an overview

Uzawa type algorithm: at step k and for a given $(\boldsymbol{\lambda})^{(k)}$,

1. we solve N problems $(\mathcal{P}_i)$ that are

$$\min_{\substack{\mathbf{X}^i,\mathbf{U}^i\\ \mathbf{U}^i_t \preceq \mathcal{F}_t}} \quad \mathbb{E}\left(\sum_{t=0}^{T} L^i_t(\mathbf{X}^i_t, \mathbf{U}^i_t, \mathbf{W}_t) + \langle \boldsymbol{\lambda}^i_t, \theta^i_t(\mathbf{X}^i_t, \mathbf{U}^i_t, \mathbf{W}_t)\rangle\right)$$

$$\text{s.t.} \quad \mathbf{X}^i_{t+1} = f^i_t(\mathbf{X}^i_t, \mathbf{U}^i_t, \mathbf{W}_t), \ \forall i \ \ \mathbb{P}-a.s.$$

2. we update the multipliers by a gradient-like method

$$(\boldsymbol{\lambda}^i_t)^{(k+1)} = (\boldsymbol{\lambda}^i_t)^{(k)} + \rho \sum_{j=1}^{N} \theta^i_t\left(\left(\mathbf{X}^i_t, \mathbf{U}^i_t, \mathbf{W}_t\right)^{(k+1)}\right), \ \forall i$$

Dual Approximate Dynamic Programming

The subproblems $(\mathcal{P}_i)$:

- are small size standard SOC problems
- involve state variables that follow Markovian dynamics

their solutions should be computable by Dynamic Programming

But:

- the randomness in $(\mathcal{P}_i)$ is generated by both $\mathbf{W}$ and $(\boldsymbol{\lambda})^{(k+1)}$
- $(\boldsymbol{\lambda})^{(k+1)}$ has no reason to be white nor Markovian

we can't solve $(\mathcal{P}_i)$ by Dynamic Programming using the state $\mathbf{X}^i$.

Dual Approximate Dynamic Programming

The idea of DADP: replacing the multipliers by their conditional expectations w.r.t. chosen information variables $\mathbf{Y}_t^i$, namely

$$\mathbb{E}\left((\boldsymbol{\lambda}_t^i)^{(k)} \middle| \mathbf{Y}_t^i \right) .$$

$\rightarrow$ We transfer the measurability problem of a given variable $((\boldsymbol{\lambda})^{(k)})$ to the measurability issue of a chosen additional variable $(\mathbf{Y}_t^i)$. It is shown to be equivalent to replace the space coupling constraints by

$$\mathbb{E}\left(\sum_{i=1}^{N} \theta_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t) \middle| \mathbf{Y}_t^i \right) = 0, \quad \forall i \quad \mathbb{P}-a.s.$$

Dual Approximate Dynamic Programming

The information variable is of the user choice. It will have a great influence on the efficiency of the DADP algorithm. In practice, $\mathbf{Y}_t^i$ is a short-memory process. Possible choices for $\mathbf{Y}_t^i$ are:

(1) $\mathbf{Y}_t^i \equiv$ cste: we deal with the constraint in expectation
(2) $\mathbf{Y}_t^i = \varphi_t^i(\mathbf{W}_t)$: we incorporate a noise
(3) $\mathbf{Y}_{t+1}^i = \tilde{f}_t^i(\mathbf{Y}_t^i, \varphi(\mathbf{W}_t))$: we incorporate a new state variable in the problem

Dual Approximate Dynamic Programming

(1) $\mathbf{Y}_t^i \equiv$ cste: we deal with the constraint in expectation

The DP equation for $(\mathcal{P}_i)$ reads: no additional state variable

$$V_T^i(x) = \mathbb{E}\left[\min_u \; L_T^i(x, u, \mathbf{W}_T) + \left\langle \mathbb{E}\big((\boldsymbol{\lambda}_T^i)^{(k)}\big), \theta_T^i(x, u, \mathbf{W}_T) \right\rangle\right]$$

$$V_t^i(x) = \mathbb{E}\left[\min_u \; \left\{ \begin{array}{l} L_t^i(x, u, \mathbf{W}_t) + V_{t+1}^i\big(f_t^i(x, u, \mathbf{W}_t)\big) \\ + \left\langle \mathbb{E}\big((\boldsymbol{\lambda}_t^i)^{(k)}\big), \theta_t^i(x, u, \mathbf{W}_t) \right\rangle \end{array} \right\}\right]$$

Dual Approximate Dynamic Programming

(2) $\mathbf{Y}_t^i = \varphi_t^i(\mathbf{W}_t)$: we incorporate a noise

The DP equation for $(\mathcal{P}_i)$ reads: no additional state variable

$$V_T^i(x) = \mathbb{E}\left[\min_u \; L_T^i\left(x, u, \mathbf{W}_T^i\right) + \left\langle \mathbb{E}\big((\boldsymbol{\lambda}_T^i)^{(k)} | \varphi_T^i(\mathbf{W}_T)\big), \theta_T^i(x, u, \mathbf{W}_T) \right\rangle\right]$$

$$V_t^i(x) = \mathbb{E}\left[\min_u \; \left\{ \begin{array}{l} L_t^i(x, u, \mathbf{W}_t) + V_{t+1}^i\left(f_t^i(x, u, \mathbf{W}_t)\right) \\ + \left\langle \mathbb{E}\big((\boldsymbol{\lambda}_t^i)^{(k)} | \varphi_t^i(\mathbf{W}_t)\big), \theta_t^i(x, u, \mathbf{W}_t) \right\rangle \end{array} \right\}\right]$$

Dual Approximate Dynamic Programming

(3) $\mathbf{Y}^i_{t+1} = \tilde{f}^i_t(\mathbf{Y}^i_t, \mathbf{W}_t)$: we add a non controled variable to the state

The DP equation for $(\mathcal{P}_i)$ reads: additional state variable

$$V^i_T(x,y) = \mathbb{E}\left[\min_u \; L^i_T(x,u,\mathbf{W}_T) + \left\langle \mathbb{E}\big((\boldsymbol{\lambda}^i_T)^{(k)} \,|\, \mathbf{Y}^i_T = y\big), \theta^i_T(x,u,\mathbf{W}_T) \right\rangle\right]$$

$$V^i_t(x,y) = \mathbb{E}\left[\min_u \left\{ \begin{array}{l} L^i_t(x,u,\mathbf{W}_t) + V^i_{t+1}\big(f^i_t(x,u,\mathbf{W}_t), \tilde{f}^i_t(y,\mathbf{W}_t)\big) \\ + \left\langle \mathbb{E}\big((\boldsymbol{\lambda}^i_t)^{(k)} \,|\, \mathbf{Y}^i_t = y\big), \theta^i_t(x,u,\mathbf{W}_t) \right\rangle \end{array} \right\}\right]$$

Dual Approximate Dynamic Programming

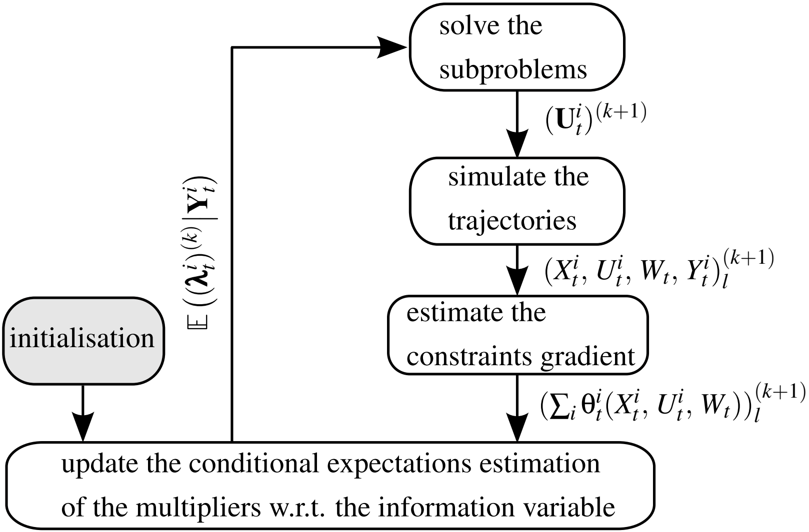
Update of the conditional expectation of the multipliers w.r.t. $\mathbf{Y}_t^i$.

- save the strategies computed at i for the fixed $(\boldsymbol{\lambda}_t^i)^{(k)}$;
- use these strategies to simulate the trajectories $(X_t^i, U_t^i, W_t, Y_t^i)_l^{(k+1)}$ over L scenarios;
- we update the multipliers possibly by the gradient method

$$(\boldsymbol{\lambda}_t^i)^{(k+1)} = (\boldsymbol{\lambda}_t^i)^{(k)} + \rho \sum_i \theta_t^i(X_t^i, U_t^i, W_t);$$

- estimate the conditional expectation of $(\boldsymbol{\lambda}_t^i)^{(k+1)}$ w.r.t. $\mathbf{Y}_t^i$ by an interpolation method

Dual Approximate Dynamic Programming



Open questions

At this point, the algorithm is used to solve:

$$\min_{\substack{\mathbf{X},\mathbf{U}\\ \mathbf{U}_t \preceq \mathcal{F}_t\\ \mathbf{X}^i_{t+1}=f^i_t(\mathbf{X}^i_t,\mathbf{U}^i_t,\mathbf{W}_t)}} \mathbb{E}\left(\sum_{i=1}^{N}\sum_{t=0}^{T} L^i_t\big(\mathbf{X}^i_t,\mathbf{U}^i_t,\mathbf{W}^i_t\big)\right) \text{ s.t. } \mathbb{E}\left(\sum_{i=1}^{N}\theta^i_t(\mathbf{X}^i_t,\mathbf{U}^i_t,\mathbf{W}^i_t)\,\middle|\,\mathbf{Y}^i_t\right)=0$$

but not the initial problem:

$$\min_{\substack{\mathbf{X},\mathbf{U}\\ \mathbf{U}_t \preceq \mathcal{F}_t\\ \mathbf{X}^i_{t+1}=f^i_t(\mathbf{X}^i_t,\mathbf{U}^i_t,\mathbf{W}_t)}} \mathbb{E}\left(\sum_{i=1}^{N}\sum_{t=0}^{T} L^i_t\big(\mathbf{X}^i_t,\mathbf{U}^i_t,\mathbf{W}^i_t\big)\right) \text{ s.t. } \sum_{i=1}^{N}\theta^i_t(\mathbf{X}^i_t,\mathbf{U}^i_t,\mathbf{W}^i_t)=0$$

Questions:

- Does the algorithm converge to the approximate solution?
- Does this solution converge to the initial problem solution?
- How shall we use the approximate solution to obtain a feasible solution?

Open questions

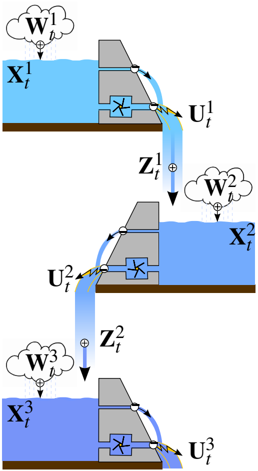
- Does the algorithm converge to the approximate solution?
  the convergence proof of Uzawa is granted, provided that:
  - the problem is posed in Hilbert spaces;
  - it exists a saddle point;

  $\rightarrow$ seem natural to place ourselves in a Hilbert space.
  But: it is known (since work of Rockafellar and Wets) that such a saddle point doesn't exist in Hilbert spaces.
- Does this solution converge to the initial problem solution?
  by an epiconvergence result but epiconvergence raises technical problems when adressed to stochastic optimization problems
- How shall we use the approximate solution to obtain a feasible solution? use a heuristic

Summary

1. Introductory example: the dams cascade
2. Decomposition over space and Dual Approximate DP
3. The three dams cascade toy problem
   - Modeling of the three dams cascade problem
   - Algorithm issues and numerical results
4. Conclusion and perspectives

Modeling of the three dams cascade problem



Criteria:

$$\mathbb{E}\Big(\sum_{i=1}^{N}\sum_{t=0}^{T-1} -C_t\eta_i\mathbf{U}_t^i + \frac{\varepsilon}{2}\mathbf{U}_t^{i\,2} + \frac{\alpha_i}{2}(\mathbf{X}_T^i - x_0^i)^2\Big)$$

Dynamics:

$$f_t^i:\ \mathbf{X}_{t+1}^i = \min\left\{\mathbf{X}_t^i + \mathbf{W}_t^i - \mathbf{U}_t^i + \mathbf{Z}_t^i, \overline{x}_{t+1}^i\right\}$$

Controls: $(\mathbf{U}_t^i, \mathbf{Z}_t^i)$

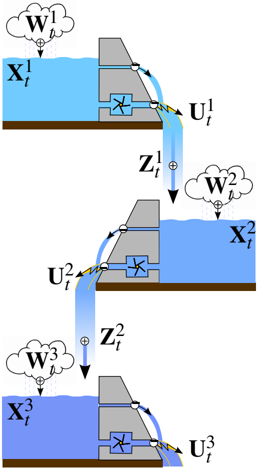
$$\mathcal{F}_t = \sigma\left(\left(\mathbf{W}_\tau^i\right)_{0\leq\tau\leq t}^{1\leq i\leq N}\right),\ \mathbf{U}_t^i \leq \overline{u}_t^i \text{ and } \mathbf{Z}_t^0 \equiv 0$$

$$g_t^i:\ \mathbf{Z}_t^{i+1} = \max\left\{\mathbf{X}_t^i + \mathbf{W}_t^i + \mathbf{Z}_t^i - \overline{x}_{t+1}^i, \mathbf{U}_t^i\right\}$$

$$\sum_{i=1}^{N}\theta_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_i) = 0:$$

$$\begin{cases} \mathbf{Z}_t^2 - g_t^1\left(\mathbf{X}_t^1, \mathbf{U}_t^1, \mathbf{W}_t^1, \mathbf{Z}_t^1\right) = 0 \\ \mathbf{Z}_t^3 - g_t^2\left(\mathbf{X}_t^2, \mathbf{U}_t^2, \mathbf{W}_t^2, \mathbf{Z}_t^2\right) = 0 \end{cases}$$

Modeling of the three dams cascade problem



Information variables:

(1) $\mathbf{Y}_t^i \equiv$ cste: we deal with the constraint in expectation

(2) $\mathbf{Y}_t^i = \mathbf{W}_t^{i-1}$: we incorporate the downstream exogeneous inflows

(3) $\mathbf{Y}_{t+1}^i = \tilde{f}_t^1(\mathbf{Y}_t^i, \mathbf{W}_t^1)$: we mimic the first dam storage level. We assume $\tilde{f}_t^1$ to be given by an oracle and fixed all over the iterations k

Algorithm issues

Dynamic Programming equation:

$$
\begin{aligned}
&V_T^i(x) = L_T^i(x) \\
&V_t^i(x,y) = \\
&\mathbb{E}\left[\min_{u,z}\left\{\begin{array}{l} L_t^i(x,u) + V_{t+1}^i\left(f_t^i\left(x,u,\mathbf{W}_t^i,z\right),\tilde{f}_t^1\left(y,\mathbf{W}_t^1\right)\right) \\ -\mathbb{E}\big((\boldsymbol{\lambda}_t^{i+1})^{(k)}\big|\mathbf{Y}_t^i = y\big) \times g_t^i(x,u,\mathbf{W}_t^i,z) \\ +\mathbb{E}\big((\boldsymbol{\lambda}_t^{i})^{(k)}\big|\mathbf{Y}_t^{i-1} = y\big) \times z \end{array}\right\}\right]
\end{aligned}
$$

Multipliers update: gradient method

$$
(\boldsymbol{\lambda}_t^{i+1})^{(k+1)} = (\boldsymbol{\lambda}_t^{i+1})^{(k)} + \rho\left((\mathbf{Z}_t^{i+1})^{(k+1)} - g_t^i((\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_t^i, \mathbf{Z}_t^i)^{(k)})\right)
$$

Numerical results

horizon: $T = 12$

The three dams share the same following discretized characteristics.

state:

$$\mathbf{X}_t^i(\omega) \in \{0, 2, \ldots, 80\}, \forall(i, t)$$

control:

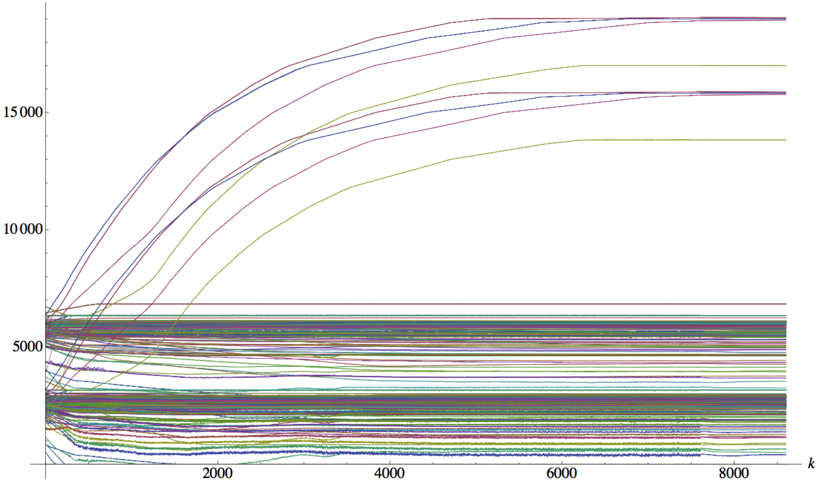
$$\mathbf{U}_t^i \in \{0, 8, \ldots, 40\}, \forall(i, t)$$
$$\mathbf{Z}_t^2 \in \{0, 2, \ldots, 40\} \text{ and } \mathbf{Z}_t^3 \in \{0, 2, \ldots, 80\}, \forall t$$

noise:

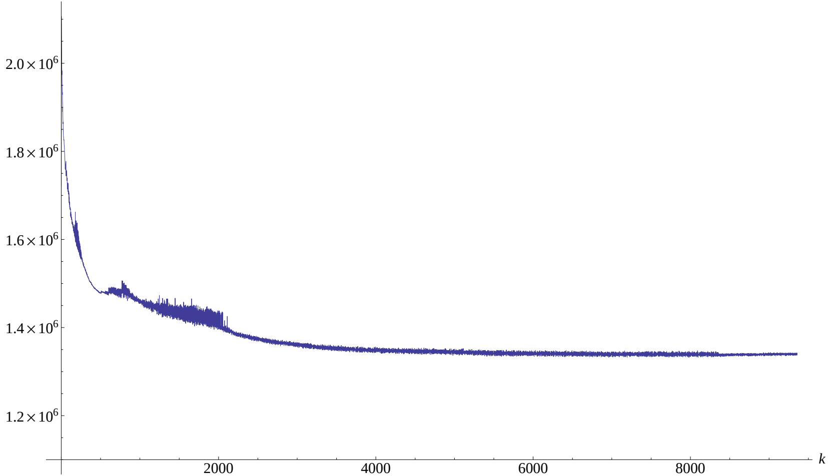
$$\mathbf{W}_t^i \in \{0, 2, \ldots, 32\}, \forall(i, t)$$

L = 10000 equiprobable scenarios to compute the multipliers update

Numerical results

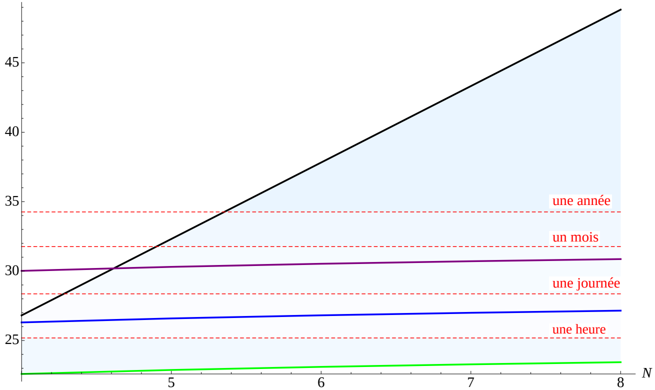


Numerical results



Numerical results

	DADP$_{(1)}$	**DADP**$_{(2)}$	**DADP**$_{(3)}$
loss	+3.3%	+2.5%	+2%
run time	$\times 2.5$	$\times 7$	$\times 375$



Summary

1. Introductory example: the dams cascade
2. Decomposition over space and Dual Approximate DP
3. The three dams cascade toy problem
4. Conclusion and perspectives

Conclusion and perspectives

Conclusion:

- encouraging results:
  - numerical convergence of the algorithm
  - satisfactory numerical results
- the add of information seems to improve the results
- first use of a dynamic information variable in DADP
- the convergence is quite slow

Conclusion and perspectives

Perspectives:

- larger dams cascade problems
- improvement of the multipliers update method (conjugate gradient, quasi-Newton...)
- theoritical study (Uzawa convergence proof in ($L\infty$, $L1$), epiconvergence...)
- comparison with standard methods
- more complex network topologies (Y, smart grids...)

Conclusion and perspectives

Thank you for your attention! =)