Decomposing Dynamic Programming equations

From global to nodal value functions

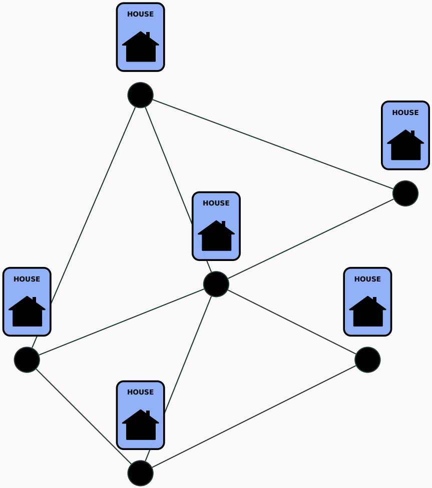
F. Pacaud — P. Carpentier — J.P. Chancelier — M. De Lara

EDF R&D, 19 juillet

ENSTA ParisTech — ENPC ParisTech — Efficacity

Motivation

We consider a *peer-to-peer* community, where different buildings exchange energy



Lecture outline

- We will formulate a large scale (stochastic) optimization problem
- We will apply decomposition algorithm on it

Optimization upper and lower bounds by decomposition

Decompose optimization problem with coupling constraints

Let, for $i \in [\![1, N]\!]$

- $\mathcal{C}^i$ be a Hilbert space
- $u^i \in \mathbb{U}^i$ be a decision variable
- $J^i : \mathbb{U}^i \to \mathbb{R}$ be a local objective
- $\Theta^i : \mathbb{U}^i \to \mathcal{C}^i$ be a mapping
- $S \subset \mathcal{C}^1 \times \cdots \times \mathcal{C}^N$ be a set

We consider the following problem

$$
\begin{aligned}
V^\sharp = \inf_{u^1, \cdots, u^N} & \sum_{i=1}^{N} J^i(u^i) \\
\text{s.t.} & \underbrace{\big(\Theta^1(u^1), \cdots, \Theta^N(u^N)\big) \in S}_{\text{coupling constraint}}
\end{aligned}
$$

Price and resource value functions provide bounds

We define for $i \in [\![1, N]\!]$

- The *local price value function*

$$\underline{V}^i[\lambda^i] = \min_{u^i} \; J^i(u^i) + \left\langle \lambda^i \,, \Theta^i(u^i) \right\rangle \,, \;\; \forall \lambda^i \in (\mathcal{C}^i)^\star$$

- The *local resource value function*

$$\overline{V}^i[r^i] = \min_{\substack{u^i \\ \Theta^i(u^i) = r^i}} \; J^i(u^i) \,, \;\; \forall r^i \in \mathcal{C}^i$$

Theorem

For any

- *admissible price* $\lambda = (\lambda^1, \cdots, \lambda^N) \in S^o = \left\{ \lambda \in \mathcal{C}^\star \mid \left\langle \lambda \,, r \right\rangle \leq 0 \,, \;\; \forall r \in \mathcal{C} \right\}$
- *admissible resource* $r = (r^1, \cdots, r^N) \in S$

$$\sum_{i=1}^{N} \underline{V}^i[\lambda^i] \leq V^\sharp \leq \sum_{i=1}^{N} \overline{V}^i[r^i]$$

Application to stochastic optimal control

We now consider the stochastic optimal control problem

$$V_0^\sharp(x_0) = \min_{\mathbf{X},\mathbf{U}} \mathbb{E}\Big[\sum_{i=1}^{N} \sum_{t=0}^{T-1} L_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_{t+1}) + K^i(\mathbf{X}_T^i)\Big]$$

$$\text{s.t. } \mathbf{X}_{t+1}^i = g_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_{t+1}) \ , \ \ \mathbf{X}_0^i = x_0^i$$

$$\sigma(\mathbf{U}_t^i) \subset \sigma(\mathbf{W}_0, \cdots, \mathbf{W}_t)$$

$$\big(\Theta_t^1(\mathbf{X}_t^1, \mathbf{U}_t^1, \mathbf{W}_{t+1}), \cdots, \Theta_t^N(\mathbf{X}_t^N, \mathbf{U}_t^N, \mathbf{W}_{t+1})\big) \in S_t$$

- $t = 0, \cdots, T$ are stages
- $\mathbf{W} = (\mathbf{W}_0, \cdots, \mathbf{W}_T)$ a global white noise process
- $\mathbf{X}^i = (\mathbf{X}_0^i, \cdots, \mathbf{X}_T^i)$ a local state process
- $\mathbf{U} = (\mathbf{U}_0^i, \cdots, \mathbf{U}_{T-1}^i)$ a local control process
- $g_t^i : \mathbb{X}_t^i \times \mathbb{U}_t^i \times \mathbb{W}_{t+1} \to \mathbb{X}_{t+1}^i$ a local dynamics
- $L_t^i : \mathbb{X}_t^i \times \mathbb{U}_t^i \times \mathbb{W}_{t+1} \to \mathbb{R}$ a local instantaneous cost

Obtaining bounds for the global problem

Theorem

For any

- *admissible price process* $\boldsymbol{\lambda} = (\boldsymbol{\lambda}^1, \cdots, \boldsymbol{\lambda}^N) \in S^o$
- *admissible resource process* $\mathbf{R} = (\mathbf{R}^1, \cdots, \mathbf{R}^N) \in S$

$$\sum_{i=1}^{N} \underline{V}_0^i[\boldsymbol{\lambda}^i](x_0^i) \leq V_0(x_0) \leq \sum_{i=1}^{N} \overline{V}_0^i[\mathbf{R}^i](x_0^i)$$

Price local value function

$$\underline{V}_0^i[\boldsymbol{\lambda}^i](x_0^i) = \min_{\mathbf{X}^i, \mathbf{U}^i} \mathbb{E}\Big[\sum_{t=0}^{T-1} L_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_{t+1}) + \big\langle \boldsymbol{\lambda}_t^i , \Theta_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_{t+1}) \big\rangle + K^i(\mathbf{X}_T^i) \Big]$$

$$\text{s.t. } \mathbf{X}_{t+1}^i = g_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_{t+1}) \; , \quad \mathbf{X}_0^i = x_0^i$$

$$\sigma(\mathbf{U}_t^i) \subset \sigma(\mathbf{W}_0, \cdots, \mathbf{W}_t)$$

Resource local value function

$$\overline{V}_0^i[\mathbf{R}^i](x_0^i) = \min_{\mathbf{X}^i, \mathbf{U}^i} \mathbb{E}\Big[\sum_{t=0}^{T-1} L_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_{t+1}) + K^i(\mathbf{X}_T^i) \Big]$$

$$\text{s.t. } \mathbf{X}_{t+1}^i = g_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_{t+1}) \; , \quad \mathbf{X}_0^i = x_0^i$$

$$\sigma(\mathbf{U}_t^i) \subset \sigma(\mathbf{W}_0, \cdots, \mathbf{W}_t)$$

$$\Theta_t^i(\mathbf{X}_t^i, \mathbf{U}_t^i, \mathbf{W}_{t+1}) = \mathbf{R}_t^i$$

Mixing price/resource and temporal decompositions

$$\sum_{i=1}^{N} \underline{V}_0^i[\boldsymbol{\lambda}^i](x_0^i) \leq V_0(x_0) \leq \sum_{i=1}^{N} \overline{V}_0^i[\mathbf{R}^i](x_0^i)$$

Price decomposition

- Fix a deterministic price $\lambda = (\lambda^1, \cdots, \lambda^N)$
- Obtain $\underline{V}_0^i[\lambda^i](x_0^i)$ by Dynamic Programming

$$\underline{V}_t^i(x_t^i) = \min_{u_t^i} \mathbb{E}\big[L_t(x_t^i, u_t^i, \mathbf{W}_{t+1}) + \langle \lambda_t^i, \Theta_t^i(x_t^i, u_t^i, \mathbf{W}_{t+1}) \rangle + \underline{V}_{t+1}^i(g_t^i(x_t^i, u_t^i, \mathbf{W}_{t+1})\big]$$

- Return the value functions $\{\underline{V}_t^i\}$

Resource decomposition

- Fix a deterministic resource $r = (r^1, \cdots, r^N)$
- Obtain $\overline{V}_0^i[r^i](x_0^i)$ by Dynamic Programming

$$\overline{V}_t^i(x_t^i) = \min_{u_t^i} \mathbb{E}\big[L_t(x_t^i, u_t^i, \mathbf{W}_{t+1}) + \overline{V}_{t+1}^i(g_t^i(x_t^i, u_t^i, \mathbf{W}_{t+1})\big]$$
$$\text{s.t. } \Theta_t^i(x_t^i, u_t^i, \mathbf{W}_{t+1}) = r_t^i$$

- Return the value functions $\{\overline{V}_t^i\}$

Deducing two control policies

Once value functions $\underline{V}_t^i$ and $\overline{V}_t^i$ computed, we define

- the global price policy

$$\underline{\pi}_t(x_t^1, \cdots, x_t^N) \in \operatorname*{arg\,min}_{u_t^1, \cdots, u_t^N} \mathbb{E}\Big[\sum_{i=1}^N L_t^i(x_t^i, u_t^i, \mathbf{W}_{t+1}) + \underline{V}_{t+1}^i(\mathbf{X}_{t+1}^i)\Big]$$

$$\text{s.t. } \mathbf{X}_{t+1}^i = g_t^i(x_t^i, u_t^i, \mathbf{W}_{t+1}) \ , \ \ \forall i \in [\![1, N]\!]$$

$$\big(\Theta_t(x_t^1, u_t^1, \mathbf{W}_{t+1}), \cdots, \Theta_t(x_t^N, u_t^N, \mathbf{W}_{t+1})\big) \in S_t$$

- the global resource policy

$$\overline{\pi}_t(x_t^1, \cdots, x_t^N) \in \operatorname*{arg\,min}_{u_t^1, \cdots, u_t^N} \mathbb{E}\Big[\sum_{i=1}^N L_t^i(x_t^i, u_t^i, \mathbf{W}_{t+1}) + \overline{V}_{t+1}^i(\mathbf{X}_{t+1}^i)\Big]$$

$$\text{s.t. } \mathbf{X}_{t+1}^i = g_t^i(x_t^i, u_t^i, \mathbf{W}_{t+1}) \ , \ \ \forall i \in [\![1, N]\!]$$

$$\big(\Theta_t(x_t^1, u_t^1, \mathbf{W}_{t+1}), \cdots, \Theta_t(x_t^N, u_t^N, \mathbf{W}_{t+1})\big) \in S_t$$

Where are we where are we heading to?

- First, we have obtained upper and lower bounds
  for global optimization problems with coupling constraints
  thanks to two spatial decomposition schemes
  - Price decomposition
  - Resource decomposition

- Second, with proper coordinating price and resource processes
  we have computed the upper and lower bounds
  by Dynamic Programming (temporal decomposition)

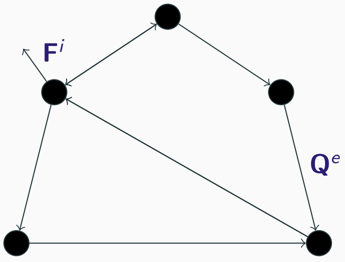
- With the upper and lower Bellman value functions,
  we have deduced two online policies

- Now, we will apply these decomposition schemes to a graph problem

Nodal decomposition of a network optimization problem

Modeling flows between nodes

Graph $G = (\mathcal{V}, \mathcal{E})$



- $\mathbf{Q}_t^e$ flow through edge e,
- $\mathbf{F}_t^i$ flow imported at node i

Let A be the *node-edge* incidence matrix

At each time $t \in [\![0, T-1]\!]$, Kirchhoff current law couples nodal and edge flows

$$A\mathbf{Q}_t + \mathbf{F}_t = 0$$

Writing down the nodal problem

We aim at minimizing the nodal costs over the nodes $i \in \mathcal{V}$

$$J^i_{\mathcal{V}}(\mathbf{F}^i) = \min_{\mathbf{X}^i, \mathbf{U}^i} \mathbb{E}\Big[\sum_{t=0}^{T-1} \underbrace{L^i_t(\mathbf{X}^i_t, \mathbf{U}^i_t, \mathbf{W}_{t+1})}_{\text{instantaneous cost}} + K^i(\mathbf{X}^i_T) \Big]$$

subject to, for all $t \in [\![0, T-1]\!]$

i) The nodal dynamics constraint (for battery and hot water tank)

$$\mathbf{X}^i_{t+1} = g^i_t(\mathbf{X}^i_t, \mathbf{U}^i_t, \mathbf{W}_{t+1})$$

ii) The non-anticipativity constraint (future remains unknown)

$$\sigma(\mathbf{U}^i_t) \subset \sigma(\mathbf{W}_0, \cdots, \mathbf{W}_t)$$

iii) The load balance equation (production + import = demand)

$$\Delta^i_t(\mathbf{X}^i_t, \mathbf{U}^i_t, \mathbf{F}^i_t, \mathbf{W}_{t+1}) = 0$$

Transportation costs are decoupled in time

At each time step $t \in [\![0, T-1]\!]$, we define the edges cost as the sum of the costs of flows $\mathbf{Q}_t^e$ through the edges e of the grid

$$J_{\mathcal{E}}^e(\mathbf{Q}) = \mathbb{E}\Big(\sum_{t=0}^{T-1} l_t^e(\mathbf{Q}_t^e) \Big)$$

Global optimization problem

The *nodal cost* $J_{\mathcal{V}}$ aggregates the costs at all nodes i

$$J_{\mathcal{V}}(\mathbf{F}) = \sum_{i \in \mathcal{V}} J_{\mathcal{V}}^{i}(\mathbf{F}^{i})$$

and the *edge cost* $J_{\mathcal{E}}$ aggregates the edges costs at all time t

$$J_{\mathcal{E}}(\mathbf{Q}) = \sum_{e \in \mathcal{E}} J_{\mathcal{E}}^{e}(\mathbf{Q}^{e})$$

The global optimization problem writes

$$V^{\sharp} = \min_{\mathbf{F},\mathbf{Q}} J_{\mathcal{V}}(\mathbf{F}) + J_{\mathcal{E}}(\mathbf{Q})$$
$$\text{s.t. } A\mathbf{Q} + \mathbf{F} = 0$$

What do we plan to do?

- We have formulated a multistage stochastic optimization problem on a graph

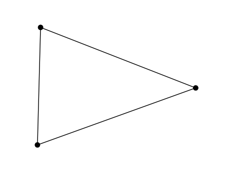
- We will handle the coupling Kirchhoff constraints by the two methods presented earlier
  - Price decomposition
  - Resource decomposition

- We will show the scalability of decomposition algorithms (We solve problems with up to 48 buildings)

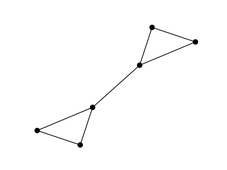
Numerical results on urban microgrids

We consider different urban configurations

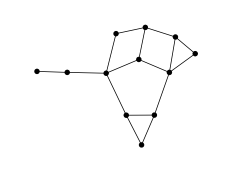
3-Nodes

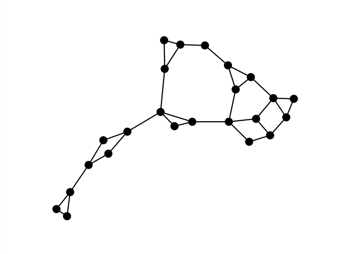


6-Nodes

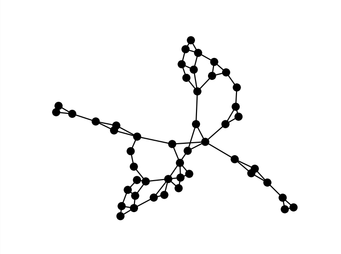


12-Nodes





24-Nodes



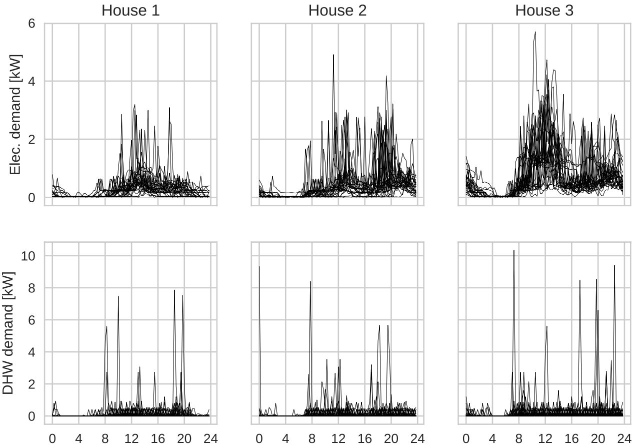
48-Nodes

Problem settings

- One day horizon at 15mn time step: $T = 96$
- Weather corresponds to a sunny day in Paris *(June 28th, 2015)*
- We mix three kind of buildings
  1. Battery + Electrical Hot Water Tank
  2. Solar Panel + Electrical Hot Water Tank
  3. Electrical Hot Water Tank

  and suppose that all consumers are commoners sharing their devices

Electrical and thermal demands are uncertain



These scenarios are generated with `StRoBE`, a generator open-sourced by KU-Leuven

Algorithms inventory

Nodal decomposition

- Encompass price and resource decompositions
- Resolution by Quasi-Newton (BFGS) gradient descent

$$\boldsymbol{\lambda}^{(k+1)} = \boldsymbol{\lambda}^{(k)} + \rho^{(k)} W^{(k)} \nabla \underline{V}(\boldsymbol{\lambda}^{(k)})$$

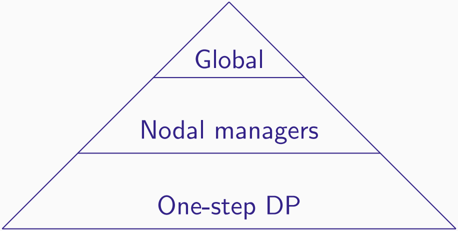
- BFGS iterates till no descent direction is found
- Each nodal subproblem solved by local SDDP (quickly converge)
- Oracle $\nabla \underline{V}(\boldsymbol{\lambda})$ estimated by Monte Carlo ($N^{scen} = 1,000$)

Global SDDP

We use as a reference the good old SDDP algorithm

- Noises $\mathbf{W}_t^1, \cdots, \mathbf{W}_t^N$ are independent node by node
  (total support size is $|supp(\mathbf{W}_t^i)|^N$.) Need to resample the support!
- Level-one cut selection algorithm (keep 100 most relevant cuts)
- Converged once gap between UB and LB is lower than 1%

Each level of hierarchy has its own algorithm



L-BFGS (IPOPT)

SDDP (StochDynamicProgramming)

QP (Gurobi)

All glue code is implemented in Julia 0.6 with JuMP 0.18

Fortunately, everything converge nicely!

Illustrating convergence for **12-Nodes** problem

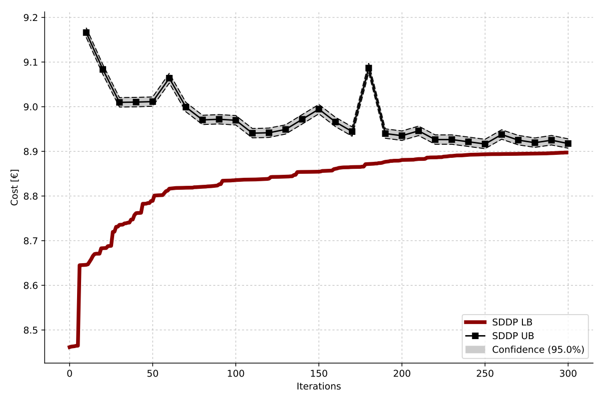


Figure 1: SDDP convergence, upper and lower bounds

Fortunately, everything converge nicely!

Illustrating convergence for **12-Nodes** problem

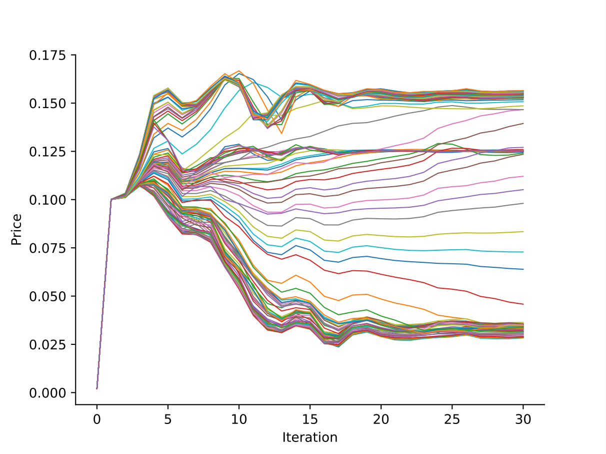


Figure 1: DADP convergence, multipliers for **Node-1**

Upper and lower bounds on the global problem

	Graph	**3-Nodes**	**6-Nodes**	**12-Nodes**	**24-Nodes**	**48-Nodes**
State dim.	$\lvert\mathbb{X}\rvert$	4	8	16	32	64
SDDP	time	1'	3'	10'	79'	453'
SDDP	LB	2.252	4.559	8.897	17.528	33.103
Price	time	6'	14'	29'	41'	128'
Price	LB	2.137	4.473	8.967	17.870	33.964
Resource	time	3'	7'	22'	49'	91'
Resource	UB	2.539	5.273	10.537	21.054	40.166

Upper and lower bounds on the global problem

	Graph	**3-Nodes**	**6-Nodes**	**12-Nodes**	**24-Nodes**	**48-Nodes**
State dim.	$\lvert\mathbb{X}\rvert$	4	8	16	32	64
SDDP	time	1'	3'	10'	79'	453'
SDDP	LB	2.252	4.559	8.897	17.528	33.103
Price	time	6'	14'	29'	41'	128'
Price	LB	2.137	4.473	8.967	17.870	33.964
Resource	time	3'	7'	22'	49'	91'
Resource	UB	2.539	5.273	10.537	21.054	40.166

- For the **24-Nodes** problem

$$\begin{array}{ccccccc} \underline{V}_0[sddp] & \leq & \underline{V}_0[price] & \leq & V^\sharp & \leq & \overline{V}_0[resource] \\ 17.528 & \leq & 17.870 & \leq & V^\sharp & \leq & 21.054 \end{array}$$

Upper and lower bounds on the global problem

	Graph	**3-Nodes**	**6-Nodes**	**12-Nodes**	**24-Nodes**	**48-Nodes**
State dim.	$\lvert\mathbb{X}\rvert$	4	8	16	32	64
SDDP	time	1'	3'	10'	79'	453'
SDDP	LB	2.252	4.559	8.897	17.528	33.103
Price	time	6'	14'	29'	41'	128'
Price	LB	2.137	4.473	8.967	17.870	33.964
Resource	time	3'	7'	22'	49'	91'
Resource	UB	2.539	5.273	10.537	21.054	40.166

- For the **24-Nodes** problem

$$\begin{array}{ccccccc} \underline{V}_0[sddp] & \leq & \underline{V}_0[price] & \leq & V^\sharp & \leq & \overline{V}_0[resource] \\ 17.528 & \leq & 17.870 & \leq & V^\sharp & \leq & 21.054 \end{array}$$

- For the biggest instance, Price Decomposition is 3.5x as fast as SDDP (and parallelization is straightforward!)

Policy evaluation by Monte Carlo simulation

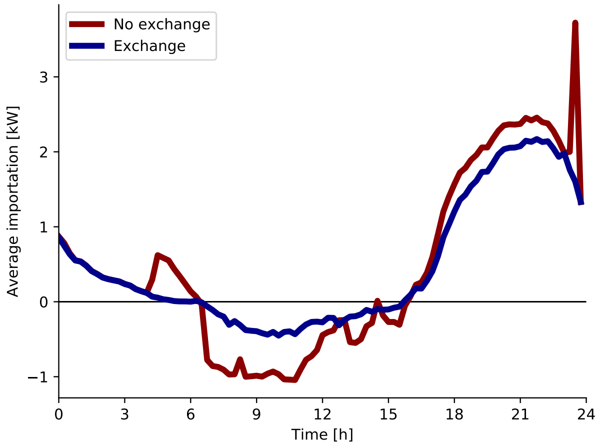
Graph	**3-Nodes**	**6-Nodes**	**12-Nodes**	**24-Nodes**	**48-Nodes**
SDDP policy	2.26 ± 0.006	4.71 ± 0.008	9.36 ± 0.011	18.59 ± 0.016	35.50 ± 0.023
Price policy	2.28 ± 0.006	4.64 ± 0.008	9.23 ± 0.012	18.39 ± 0.016	34.90 ± 0.023
Gap	-0.9 %	+1.5%	+1.4%	+1.1%	+1.7%
Resource policy	2.29 ± 0.006	4.71 ± 0.008	9.31 ± 0.011	18.56 ± 0.016	35.03 ± 0.022
Gap	-1.3 %	0.0%	+0.5%	+0.2%	+1.2%

Price policy beats numerically SDDP policy and resource policy

$$\begin{array}{ccccccc} V^\sharp & \leq & C[price] & \leq & C[resource] & \leq & C[sddp] \\ V^\sharp & \leq & 18.39 & \leq & 18.56 & \leq & 18.59 \end{array}$$

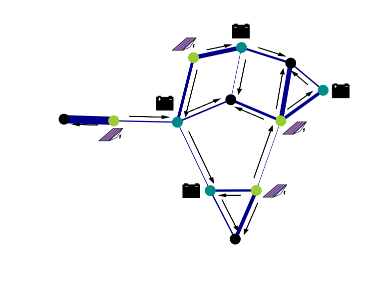
Hunting down the duck curve

Looking at the *average* global electricity importation from the external distribution grid

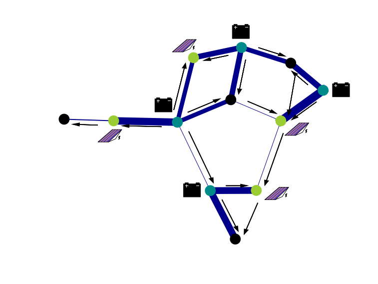


Optimal flows in simulation for 12-Nodes problem

1. We simulate price policy over 1,000 scenarios
2. We look at flows at two moments in the day

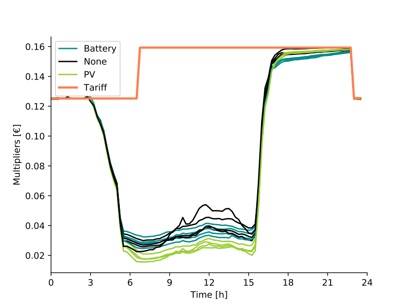


12am

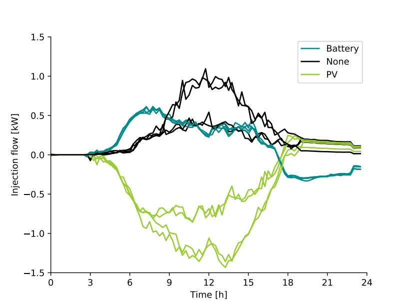


9pm

Optimal prices and flows returned by decomposition



Price



Resource

Conclusion

Conclusion

- We have presented two algorithms that decompose, spatially then temporally, a global optimization problem under coupling constraints

- On this case study, decomposition beat SDDP for large instances ($\geq$ 24 nodes)
  - In time (3.5x faster)
  - In precision ($>$ 1% better)

- Can we obtain tighter bounds?
  If we select properly the resource and price processes $\mathbf{R}$ and $\boldsymbol{\lambda}$, among Markovian ones we can obtain nodal value functions (with an extended local state)