Stochastic optimization methods for efficient energy management under stochasticity

François Pacaud

March 14th, 2019

About me

- **2015-2018:** PhD student at CERMICS (ENPC)
  Supervised by Michel De Lara and Pierre Carpentier
  Stochastic optimization methods for efficient energy management under stochasticity

- **Now:** Optimization engineer at Artelys

Mixing optimization and probability

"Statistics and machine learning make best use of the past.
Mathematical optimization makes best use of the future".

Random quote from the Internet

Taking decisions facing uncertainty: mixing best of both worlds

Mixing optimization and probability

"Statistics and machine learning make best use of the past.
Mathematical optimization makes best use of the future".

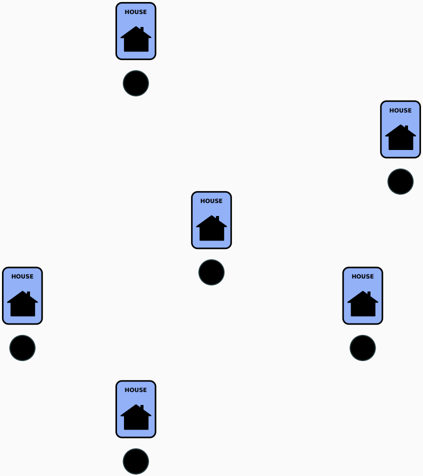
Random quote from the Internet

Taking decisions facing uncertainty: mixing best of both worlds

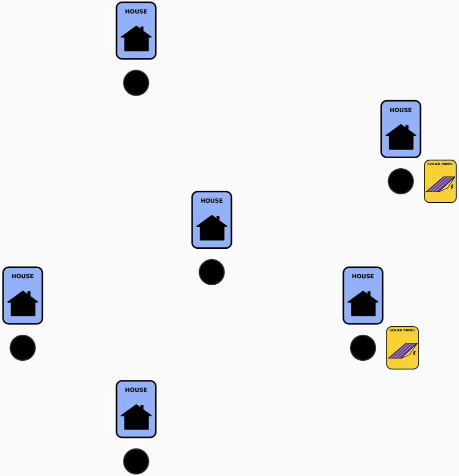
Optimization and Energy

- Unit-commitment problems
- Optimal power flow
- Energy flows optimal dispatch

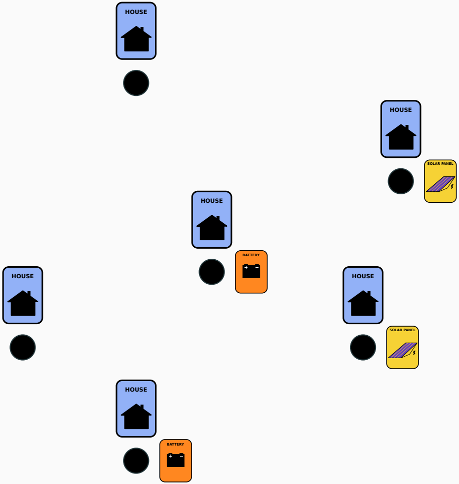
My thesis in one picture



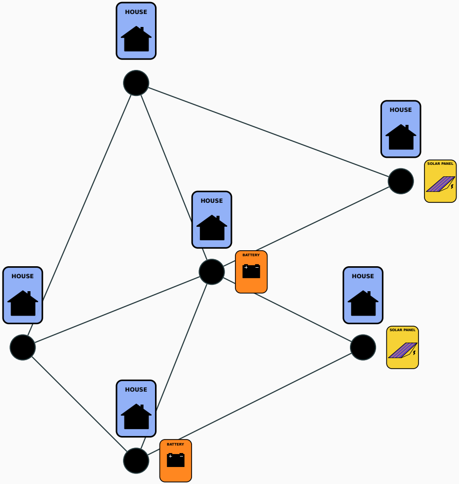
My thesis in one picture



My thesis in one picture

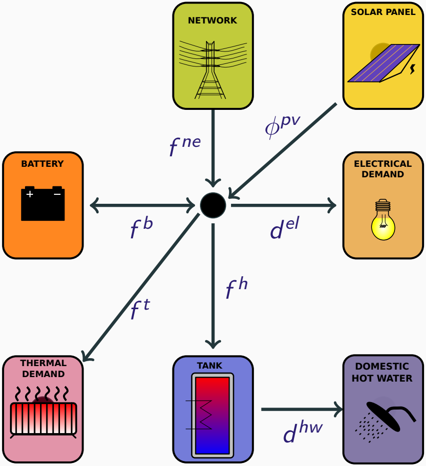


My thesis in one picture



First example: the one building problem

The building we consider



Diving into the theory of optimal control

In the next, we distinct three variables

- A **state** $x \in \mathbb{X} = \mathbb{R}^n$ *(here: the energy in stocks*
- A **control** $u \in \mathbb{U} = \mathbb{R}^m$ *(here: the controllable energy flows)*
- An exogeneous **noise** (aka uncertainty) $w \in \mathbb{W} = \mathbb{R}^l$ *(here: the energy demands)*

Diving into the theory of optimal control

In the next, we distinct three variables

- A **state** $x \in \mathbb{X} = \mathbb{R}^n$ *(here: the energy in stocks*
- A **control** $u \in \mathbb{U} = \mathbb{R}^m$ *(here: the controllable energy flows)*
- An exogeneous **noise** (aka uncertainty) $w \in \mathbb{W} = \mathbb{R}^l$
  (here: the energy demands)

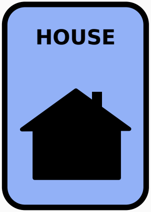
These three variables are linked together
via the continuous state equation

$$\dot{x} = f(x, u, w)$$

or equivalently, in discrete time,

$$x_{t+1} = f_t(x_t, u_t, w_t)$$

Introducing the notation of stochastic optimal control



Let $\{0, 1, \cdots, T-1, T\}$ be a discrete-time span
(here we consider a horizon $T = 24h$ and $\Delta t = 15mn$)

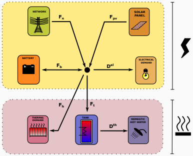
Objective

- We frame a discrete-time optimal control problem and consider at all time $t \in \{0, 1, \cdots, T-1\}$
  - An uncertainty $w_t \in \mathbb{W}_t$ *(occuring between $t-1$ and t)*
  - A control $u_t \in \mathbb{U}_t$ *(leveraging the system)*
  - A state $x_t \in \mathbb{X}_t$ *(outlining the energy stocks)*
- We model the uncertainties as random variables thus rendering the optimization problem stochastic
- We look at policies

$$\pi_t : \mathbb{X}_t \to \mathbb{U}_t$$

to compute decision online for all time $t \in \{0, \cdots, T-1\}$

Introducing noises, controls and states

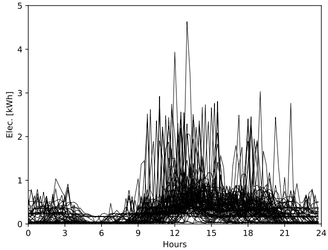


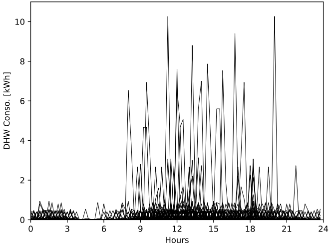
- **Uncertainties** $\mathbf{W}_t = \left(\mathbf{D}^{el}_t, \mathbf{D}^{hw}_t\right)$
  - $\mathbf{D}^{el}_t$, electrical demand (kW)
  - $\mathbf{D}^{hw}_t$, domestic hot water demand (kW)

- **Control variables** $\mathbf{U}_t = \left(\mathbf{F}_{\mathbf{B},t}, \mathbf{F}_{T,t}, \mathbf{F}_{\mathbf{H},t}\right)$
  - $\mathbf{F}_{\mathbf{B},t}$, energy exchange with the battery (kW)
  - $\mathbf{F}_{T,t}$, energy used to heat the hot water tank (kW)
  - $\mathbf{F}_{\mathbf{H},t}$, thermal heating (kW)

- **Stock variables** $\mathbf{X}_t = \left(\mathbf{B}_t, \mathbf{H}_t, \boldsymbol{\theta}^i_t, \boldsymbol{\theta}^w_t\right)$
  - $\mathbf{B}_t$, battery level (kWh)
  - $\mathbf{H}_t$, hot water storage (kWh)
  - $\boldsymbol{\theta}^i_t$, inner temperature ($^\circ C$)
  - $\boldsymbol{\theta}^w_t$, wall's temperature ($^\circ C$)

Looking at scenarios of demands during one day





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

Modeling energy flows

We use (simplified) physical models to model the energy exchanges

Energy flow balance

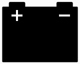
$$\text{Stock}_{t+1} = \text{Stock}_t + \text{Flow}_t^{in} - \text{Flow}_t^{out}$$

where **Flow** is either

- A control (e.g. battery charge/discharge)
- An uncertainty (e.g. electricity demand)

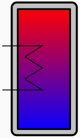
Discrete time state equations

We have the four state equations (all linear), describing the evolution over time of the stocks:



$$\mathbf{B}_{t+1} = \alpha_{\mathbf{B}} \mathbf{B}_t + \Delta T \Big(\rho_c \mathbf{F}^{+}_{\mathbf{B},t} - \frac{1}{\rho_d} \mathbf{F}^{-}_{\mathbf{B},t} \Big)$$

$$\mathbf{H}_{t+1} = \alpha_{\mathbf{H}} \mathbf{H}_t + \Delta T \big[\mathbf{F}_{T,t} - \mathbf{D}^{hw}_{t+1} \big]$$



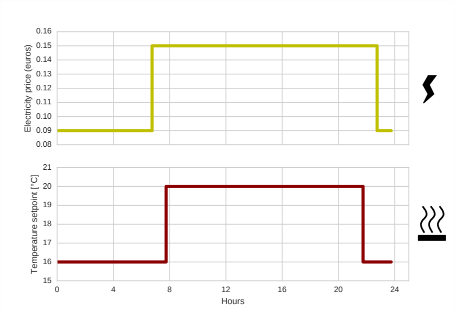
$$\boldsymbol{\theta}^{w}_{t+1} = \boldsymbol{\theta}^{w}_t + \frac{\Delta T}{c_m} \left[\frac{\boldsymbol{\theta}^{i}_t - \boldsymbol{\theta}^{w}_t}{R_i + R_s} + \frac{\boldsymbol{\theta}^{e}_t - \boldsymbol{\theta}^{w}_t}{R_m + R_e} + \gamma \mathbf{F}_{\mathbf{H},t} + \frac{R_i}{R_i + R_s} P^{int}_t + \frac{R_e}{R_e + R_m} P^{ext}_t \right]$$

$$\boldsymbol{\theta}^{i}_{t+1} = \boldsymbol{\theta}^{i}_t + \frac{\Delta T}{c_i} \left[\frac{\boldsymbol{\theta}^{w}_t - \boldsymbol{\theta}^{i}_t}{R_i + R_s} + \frac{\boldsymbol{\theta}^{e}_t - \boldsymbol{\theta}^{i}_t}{R_v} + \frac{\boldsymbol{\theta}^{e}_t - \boldsymbol{\theta}^{i}_t}{R_f} + (1 - \gamma) \mathbf{F}_{\mathbf{H},t} + \frac{R_s}{R_i + R_s} P^{int}_t \right]$$

which will be denoted

$$\boxed{\mathbf{X}_{t+1} = f_t(\mathbf{X}_t, \mathbf{U}_t, \mathbf{W}_{t+1})}$$

Prices and temperature setpoints vary along time



- $T = 24$h, $\Delta T = 15$mn
- Electricity peak and off-peak hours
  $p_t^E = 0.09$ or 0.15 euros/kWh
- Temperature set-point
  $\bar{\theta}_t^i = 16^\circ C$ or $20^\circ C$

The costs we have to pay

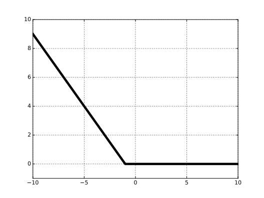
- Cost to import electricity from the network

$$p_t^E \times \max\{0, \mathbf{F}_{NE,t+1}\}$$

where we define the recourse variable (electricity balance):

$$\underbrace{\mathbf{F}_{NE,t+1}}_{Network} = \underbrace{\mathbf{D}^{el}_{t+1}}_{Demand} + \underbrace{\mathbf{F}_{\mathbf{B},t}}_{Battery} + \underbrace{\mathbf{F}_{\mathbf{H},t}}_{Heating} + \underbrace{\mathbf{F}_{T,t}}_{Tank} - \underbrace{\mathbf{F}_{pv,t}}_{Solar\ \ panel}$$

- Virtual Cost of thermal discomfort: $\kappa_{th}(\underbrace{\boldsymbol{\theta}_t^i - \bar{\boldsymbol{\theta}}_t^i}_{\text{deviation from setpoint}})$



κ_{th}
Piecewise linear cost which penalizes temperature if below given setpoint

Instantaneous and final costs for a single house

- The instantaneous convex costs are

$$L_t(\mathbf{X}_t, \mathbf{U}_t, \mathbf{W}_{t+1}) = \underbrace{p_t^E \max\{0, \mathbf{F}_{NE,t+1}\}}_{bill} + \underbrace{\kappa_{th}(\theta_t^i - \bar{\theta}_t^i)}_{discomfort}$$

- We add a final linear cost

$$K(\mathbf{X}_T) = -p^{\mathbf{H}}\mathbf{H}_T - p^{\mathbf{B}}\mathbf{B}_T$$

to avoid empty stocks at the final horizon T

Writing the stochastic optimization problem

We now write the optimization problem

$$\min_{\mathbf{X},\mathbf{U}} \mathbb{E}\left[\sum_{t=0}^{T-1} L_t(\mathbf{X}_t, \mathbf{U}_t, \mathbf{W}_{t+1}) + K(\mathbf{X}_T)\right]$$

$$\text{s.t. } \mathbf{X}_{t+1} = f_t(\mathbf{X}_t, \mathbf{U}_t, \mathbf{W}_{t+1}) \,, \quad \mathbf{X}_0 = x_0 \quad \text{(Dynamic)}$$

$$x^\flat \leq \mathbf{X}_t \leq x^\sharp \quad \text{(Bounds)}$$

$$\mathbf{U}_t = \pi(\mathbf{W}_1, \dots, \mathbf{W}_t)$$

We aim at minimizing the expected value of the sum of the operational costs

We look at solution as state feedback policies $\pi_t : \mathbb{X}_t \to \mathbb{U}_t$

$$\mathbf{U}_t = \pi_t(\mathbf{X}_t)$$

First example: the one building problem

Numerical results

Modeling the future uncertainties

Some triviality ahead:

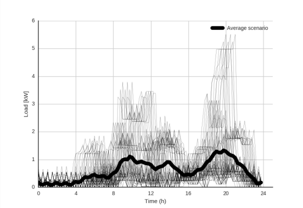
- We do not know the future
- We can only rely on *models*
- Probabilities do **not** exist by themselves
  (see [Matheron, *Estimating and choosing*])

Two algorithms using two different modelings

Model Predictive Control: Views the future as
a *deterministic forecast*

Stochastic Dynamic Programming: Views the future as
time independent random variables

Model Predictive Control (MPC)



Procedure

Input

- A deterministic forecast scenario $(\overline{w}_{t+1}, \cdots, \overline{w}_T)$ (possibly updated)

Output

- A policy $\pi_t^{mpc} : \mathbb{X}_t \to \mathbb{U}_t$ that maps the current state x_t to a decision u_t

The MPC policy $\pi_t^{mpc} : \mathbb{X}_t \to \mathbb{U}_t$ writes, for all time $t \in \{0, \cdots, T-1\}$

$$\pi_t^{mpc}(x_t) \in \arg\min_{x,u} \sum_{s=t}^{T-1} L_s(x_s, u_s, \overline{w}_{s+1}) + K(x_T)$$

$$\text{s.t. } x_{s+1} = f_s(x_s, u_s, \overline{w}_{s+1})$$

and corresponds to solve a deterministic optimization problem

Brief recalls on Dynamic Programming

You may have heard of Dynamic Programming already

Backward computation of a collection of value functions $\{V_t\}_{t=0,\cdots,T}$

$$V_t(x_t) = \min_u \; l_t(x_t, u_t) + \underbrace{V_{t+1}(f_t(x_t, u_t))}_{\text{future cost}}$$

Brief recalls on Dynamic Programming

You may have heard of Dynamic Programming already

Backward computation of a collection of value functions $\{V_t\}_{t=0,\cdots,T}$

$$V_t(x_t) = \min_u \; \mathbb{E}_{w_t}\Big[L_t(x_t, u_t, w_{t+1}) + \underbrace{V_{t+1}(f_t(x_t, u_t, w_{t+1}))}_{\text{future cost}}\Big]$$

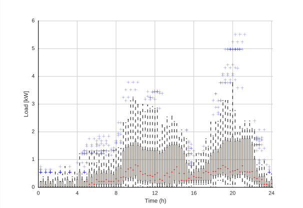
Brief recalls on Dynamic Programming

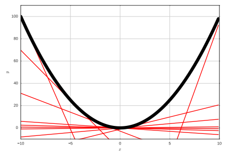
You may have heard of Dynamic Programming already

Backward computation of a collection of value functions $\{V_t\}_{t=0,\cdots,T}$

$$V_t(x_t) = \min_u \sum_{i=1}^{n} \pi_i \big[L_t(x_t, u_t, w_{t+1}^i) + \underbrace{V_{t+1}(f_t(x_t, u_t, w_{t+1}^i))}_{\text{future cost}} \big]$$

Stochastic Dual Dynamic Programming





Procedure

Input

- A family of discrete marginal distributions $\mu_{t+1}(\cdot) = \sum_{s=1}^{S} \pi_s \delta_{w_{t+1}^s}(\cdot)$ with $\sum_{s=1}^{S} \pi_s = 1$

Output

- Value functions $\underline{V}_t : \mathbb{X}_t \to \mathbb{R}$ approximating the original Bellman functions $V_t : \mathbb{X}_t \to \mathbb{R}$ as a supremum of affine functions

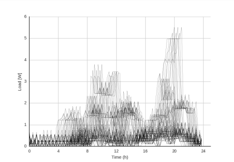
$$\underline{V}_t(x) = \max_{1 \le k \le K} \{\lambda_t^k x + \beta_t^k\} \le V_t(x)$$

- A policy $\pi_t^{sddp} : \mathbb{X}_t \to \mathbb{U}_t$ that maps the current state x_t to a decision u_t

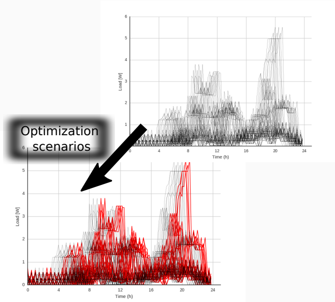
The SDDP policy $\pi^{sddp} : \mathbb{X}_t \to \mathbb{U}_t$ writes, for all time $t \in \{0, \cdots, T-1\}$

$$\pi_t^{sddp}(x_t) \in \arg\min_{u_t \in \mathbb{U}_t} \sum_{s=1}^{S} \pi_s \left[L_t(x_t, u_t, w_{t+1}^s) + \underline{V}_{t+1}\big(f_t(x_t, u_t, w_{t+1}^s)\big) \right]$$

How to assess MPC and SDDP strategies?



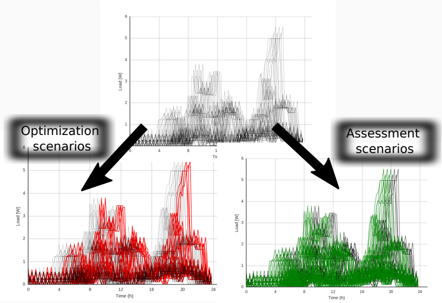
How to assess MPC and SDDP strategies?



Optimization scenarios

- Dedicated to fit probability laws or forecasts
- All algorithms have access to the same set of scenarios
- Algorithms do not have access to assessment scenarios

How to assess MPC and SDDP strategies?



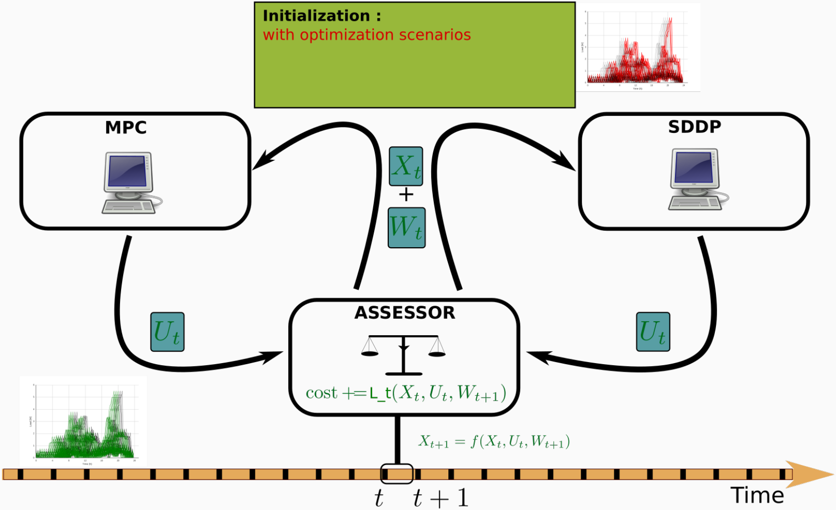
Optimization scenarios

- Dedicated to fit probability laws or forecasts
- All algorithms have access to the same set of scenarios
- Algorithms do not have access to assessment scenarios

Assessment scenarios

- Dedicated to assess the performance of the different policies
- We simulate each policy along each assessment scenario

Assessment procedure



All glue code is implemented in Julia 0.6 with JuMP 0.18

A language perfectly suited for optimization use!

Comparison of MPC and SDDP

We compare MPC and SDDP over 1,000 assessment scenarios

	SDDP	MPC	Heuristic
Electricity bill (€)			
Winter day	4.38 ± 0.02	4.59 ± 0.02	5.55 ± 0.02
Spring day	1.46 ± 0.01	1.45 ± 0.01	2.83 ± 0.01
Summer day	0.10 ± 0.01	0.18 ± 0.01	0.33 ± 0.02

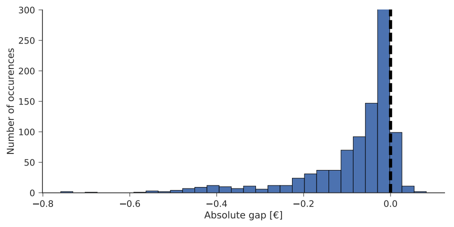
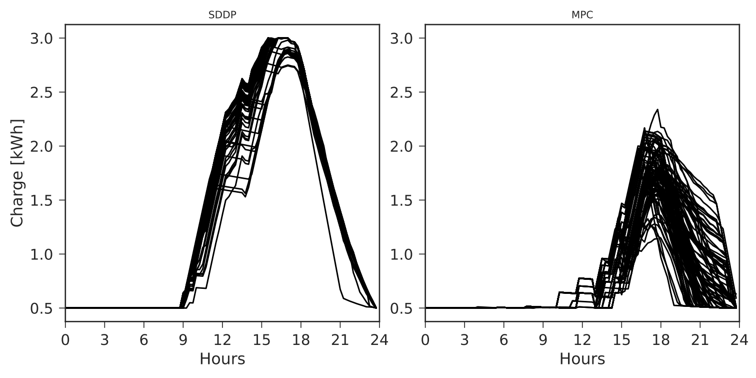
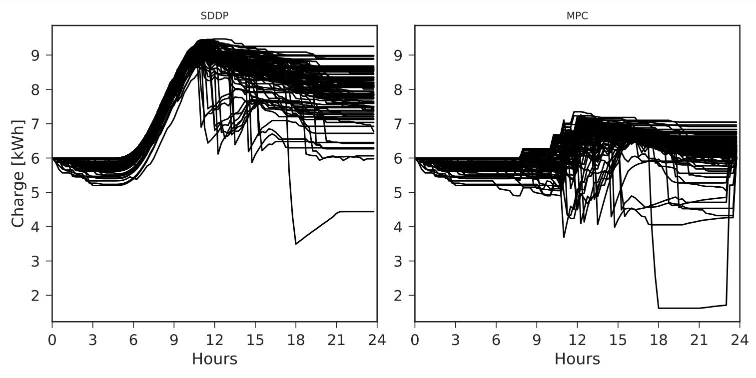


Figure 1: Absolute gap savings between MPC and SDDP during *Summer day*

Looking at optimal trajectories



Looking at optimal trajectories



Conclusion for the single house problem

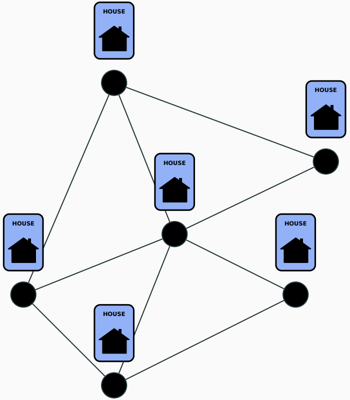
Contributions

- We begin the study by a simple example
- We have formulated a stochastic optimization problem for domestic energy management system
- We have compared two resolution algorithms (MPC and SDDP)
- On this particular example, SDDP gives better performance than MPC

Mixing time and spatial decomposition in large-scale optimization problems

The challenge is now to be able to tackle larger problems

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

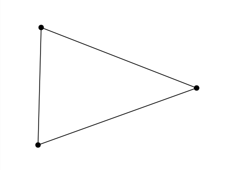


Objective

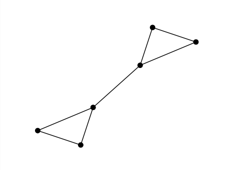
- We will formulate a large scale (stochastic) optimization problem
- We will apply decomposition algorithm on it

We consider different urban configurations

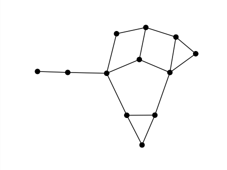
3-Nodes

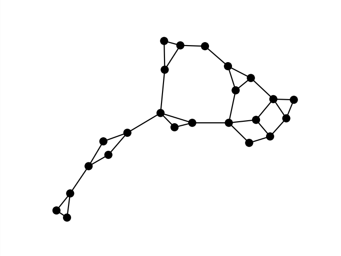


6-Nodes

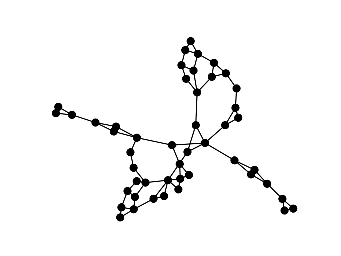


12-Nodes





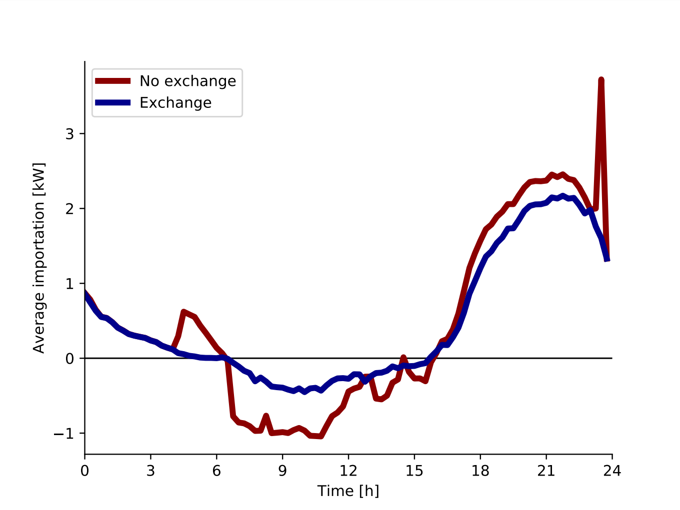
24-Nodes



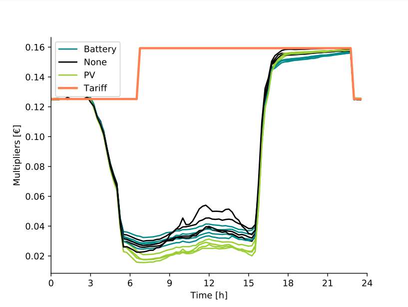
48-Nodes

Hunting down the duck curve

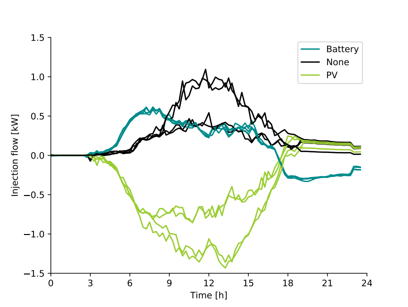
Looking at the *average* global electricity importation from the external distribution grid



Optimal prices and flows returned by decomposition



Price



Resource

Conclusion

Ok then...

- This is just the beginning!
- Rising interests in integrating cutting edge optimization methods in machine learning
- **Artelys is hiring!**

francois.pacaud@artelys.com