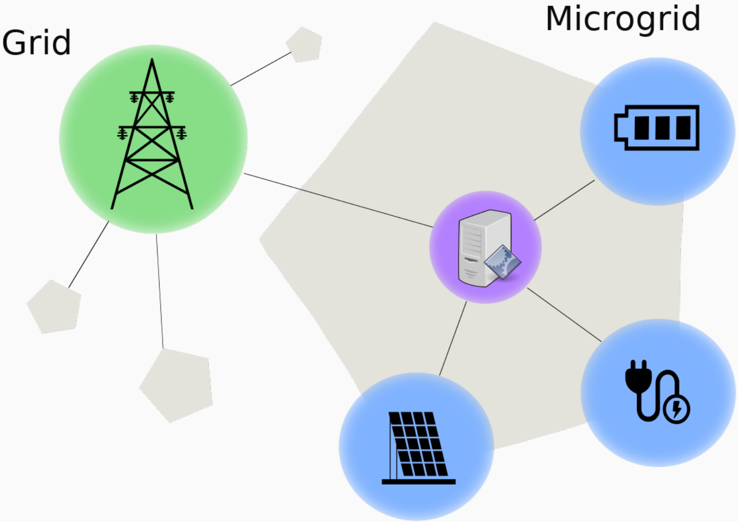
# EMSx: an Numerical Benchmark for Energy Management Systems

SMAI Mode, September 2020

Adrien Le Franc[1,2], Pierre Carpentier[3], Jean-Philippe Chancelier[1], Michel De Lara[1]

CERMICS ENPC[1], Efficacity[2], UMA ENSTA Paris[3]


Grid
Microgrid

# Problem context

- **Question**

  How to evaluate an **Energy Management System** (EMS)
  for operating a microgrid at **least expected cost** ?

- **Our contribution**

  We introduce EMSx, a **microgrid controller benchmark**
  to compare EMS techniques on an **open** and **diversified** testbed

# Outline of the presentation

1. **The EMSx dataset**
2. **The EMSx mathematical framework**
3. **The EMSx software**
4. **Numerical examples**
5. **Conclusion**

# Outline of the presentation

1. **The EMSx dataset**
2. **The EMSx mathematical framework**
3. **The EMSx software**
4. **Numerical examples**
5. **Conclusion**

# Examples of daily scenarios from EMSx

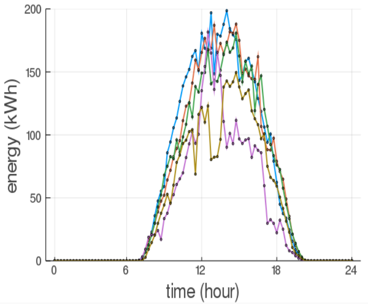



**Figure 1:** Examples of daily photovoltaic profiles

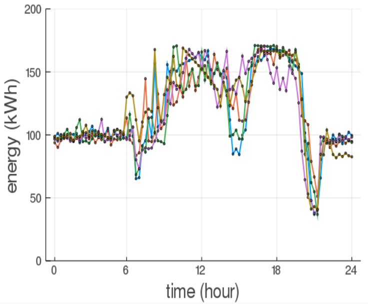



**Figure 2:** Examples of daily load profiles

(data collected by Schneider Electric on real industrial sites)

# Detailed description of the dataset

- Over 1 year of historical data for **70 industrial sites**
- 15 minutes sampled **historical observations**
- 15 minutes actualized 24h ahead **historical forecasts**
- Publicly available

# Our data reflect a large diversity of microgrids

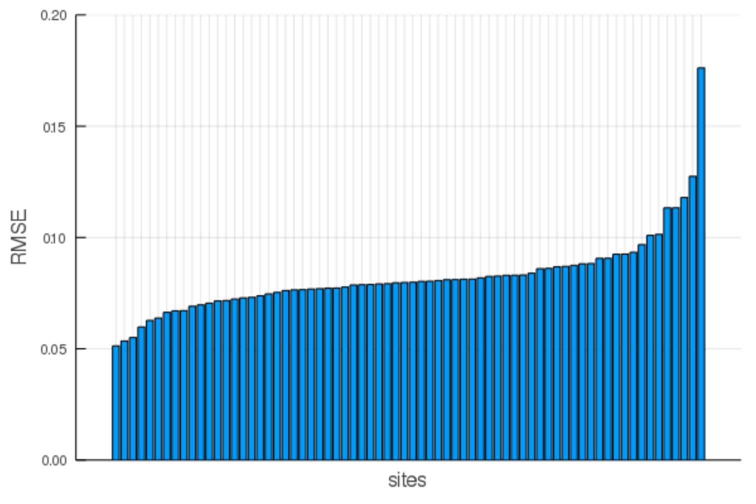



**Figure 3:** RMSE of the net demand forecasts for each of the 70 sites

# Outline of the presentation

1. The EMSx dataset

2. **The EMSx mathematical framework**

3. The EMSx software

4. Numerical examples

5. Conclusion

# Outline of the section

**2. The EMSx mathematical framework**

Microgrid control simulation

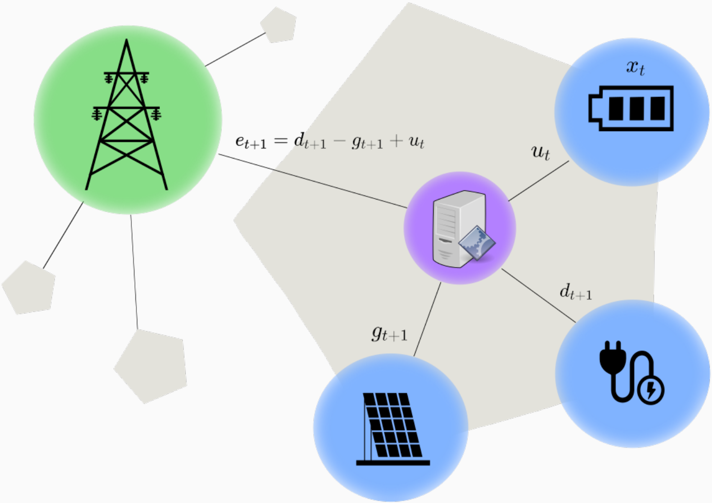
What is a controller ?

Assessing a controller design technique

# Time scale and variables

We manage a microgrid over time steps $t \in \{0, 1, \dots, T\}$ , $\Delta_t = 15$ min

- $x_t \in [0, 1]$ state of charge of the battery
- $u_t \in [\underline{u}, \overline{u}]$ energy charged $(u_t \geq 0)$
  or discharged $(u_t \leq 0)$ over $[t, t+1]$
- $w_{t+1} = (g_{t+1}, d_{t+1})$ generation and demand
  historical data over $[t, t+1]$
- $\hat{w}_{t,t+k} = (\hat{g}_{t,t+k}, \hat{d}_{t,t+k})$ , $k \in \{1, \dots, 96\}$
  generation and demand historical forecast at time $t$
  over $[t + k - 1, t + k]$


$e_{t+1} = d_{t+1} - g_{t+1} + u_t$
$u_t$
$x_t$
$d_{t+1}$
$g_{t+1}$

# Our microgrid management model

- state of charge ruled by the **dynamics**

$$x_{t+1} = f(x_t, u_t) = x_t + \frac{\rho_{\mathrm{c}}}{c} u_t^+ - \frac{1}{\rho_{\mathrm{d}} c} u_t^-$$

- controls restricted to the **admissibility set**

$$\mathcal{U}(x_t) = \{u_t \in \mathbb{R} \mid \underline{u} \leq u_t \leq \overline{u} \text{ and } 0 \leq f(x_t, u_t) \leq 1\}$$

- energy exchanges induce a **cost**

$$L_t(u_t, w_{t+1}) = p_t^{\text{buy}} \cdot (d_{t+1} - g_{t+1} + u_t)^+ - p_t^{\text{sell}} \cdot (d_{t+1} - g_{t+1} + u_t)^-$$

# Outline of the section

2. **The EMSx mathematical framework**

   Microgrid control simulation

   What is a controller ?

   Assessing a controller design technique

# Online information for the management problem

- The **online information** contains
  - 24h of historical observation
  - 24h of historical forecasts

- At time $t \in \{0, \dots, T-1\}$, we have access to

$$h_t = \begin{pmatrix} w_t, w_{t-1}, \dots, w_{t-95} \\ \hat{w}_{t,t+1}, \dots, \hat{w}_{t,t+96} \end{pmatrix} \in \mathbb{H} = \mathbb{R}^{2\times 96} \times \mathbb{R}^{2\times 96}$$

- The sequence $\{0, \dots, T-1\}$ is characterized by the **partial chronicle**

$$h = (h_0, \dots, h_{T-1}) \in \mathbb{H}^T$$

# A generic controller definition

A **controller** is a sequence of mappings $\phi = (\phi_0, \dots, \phi_{T-1})$ such that

$$\begin{array}{rl} \phi_t & : [0,1] \times \mathbb{H} \to \mathbb{R} \\ & (x_t, h_t) \mapsto \phi_t(x_t, h_t) \in \mathcal{U}(x_t) \end{array}, \quad \forall t \in \{0, \dots, T-1\}$$

- decisions are **non anticipative**
- this **generic** definition covers a large class of controllers

# Outline of the section

**2. The EMSx mathematical framework**

Microgrid control simulation

What is a controller ?

Assessing a controller design technique

# Management cost of a controller

For a site $i$ in the total pool of sites $I = \{1, \ldots, 70\}$,
the application of a controller $\phi^i$ along a partial chronicle $h \in \mathbb{H}^T$
yields a **management cost**

$$J^i(\phi^i, h) = \sum_{t=0}^{T-1} L_t^i(u_t, w_{t+1})$$

$$x_0 = 0$$

$$x_{t+1} = f^i(x_t, u_t)$$

$$u_t = \phi_t^i(x_t, h_t)$$

# Two baseline management costs

We compare the management cost of a controller $\phi^i$ with

- the cost of a **dummy controller** $\phi^{\mathrm{d}} = 0$
  which does not use the battery
- the **anticipative cost** $\underline{J}^i(h) \leq J^i(\phi^i, h)$
  obtained by managing the microgrid
  with an **idealistic** full knowledge of the future

# Gain of a controller $\phi^i$ on site $i \in I$

We have a pool of **simulation chronicles** $\mathscr{S}^i$

- We define the **gain** of $\phi^i$ as

$$G^i(\phi^i) = \frac{1}{|\mathscr{S}^i|} \sum_{h \in \mathscr{S}^i} J^i(\phi^{\mathrm{d}}, h) - J^i(\phi^i, h)$$

- We define the **anticipative gain** as

$$\overline{G}^i = \frac{1}{|\mathscr{S}^i|} \sum_{h \in \mathscr{S}^i} J^i(\phi^{\mathrm{d}}, h) - \underline{J}^i(h)$$

# Normalized score of a control technique $(\phi^i)_{i \in I}$

- We define the **normalized gain** of a controller $\phi_i$ as

$$\mathcal{G}^i(\phi^i) = \frac{G^i(\phi^i)}{\overline{G}^i} = \frac{\text{average gain of } \phi^i \text{ vs } \phi^{\mathsf{d}}}{\text{average anticipative gain vs } \phi^{\mathsf{d}}}$$

- We define the **normalized score** of a control technique $(\phi^i)_{i \in I}$ as

$$\mathcal{G}\big(\{\phi^i\}_{i \in I}\big) = \frac{1}{|I|} \sum_{i \in I} \mathcal{G}^i(\phi^i)$$

# Outline of the presentation

1. The EMSx dataset
2. The EMSx mathematical framework
3. **The EMSx software**
4. Numerical examples
5. Conclusion

# A Julia package: EMSx.jl

```julia
using EMSx

mutable struct DummyController <: EMSx.AbstractController end

EMSx.compute_control(controller::DummyController,
    information::EMSx.Information) = 0.

const controller = DummyController()

EMSx.simulate_sites(controller,
    "home/xxx/path_to_save_folder",
    "home/xxx/path_to_price",
    "home/xxx/path_to_metadata",
    "home/xxx/path_to_simulation_data")
```

**Figure 4:** Example of the implementation and simulation of a dummy controller with the EMSx.jl package

# Outline of the presentation

1. The EMSx dataset
2. The EMSx mathematical framework
3. The EMSx software
4. **Numerical examples**
5. Conclusion

# Outline of the section

4. **Numerical examples**

   Controller design techniques

   Numerical results

# Model Predictive Control (MPC)

At time $t \in \{0, \dots, T-1\}$,

$$
\left\{
\begin{array}{ll}
u_t^\star \in \mathop{\arg\min}\limits_{u_t} \min\limits_{(u_{t+1}, \dots, u_{t+H-1})} & \displaystyle\sum_{s=t}^{t+H-1} L_s(u_s, \hat{w}_{t,s+1}) \\
& x_{s+1} = f(x_s, u_s)\ , \ \ \forall s \in \{t, \dots, t+H-1\} \\
& u_s \in \mathcal{U}(x_s)\ , \ \ \forall s \in \{t, \dots, t+H-1\} \\
\phi_t^{\mathrm{MPC}}(x_t, h_t) = u_t^\star &
\end{array}
\right.
$$

# Open Loop Feedback Control (OLFC)

At time $t \in \{0, \dots, T-1\}$,

$$\left\{ \begin{array}{ll} u_t^\star \in \arg\min_{u_t} \min_{(u_{t+1}, \dots, u_{t+H-1})} & \sum_{\sigma \in \mathbb{S}} \pi_t^\sigma \Big( \sum_{s=t}^{t+H-1} L_s(u_s, \hat{w}_{t,s+1}^\sigma) \Big) \\ & x_{s+1} = f(x_s, u_s)\ , \ \ \forall s \in \{t, \dots, t+H-1\} \\ & u_s \in \mathcal{U}(x_s)\ , \ \ \forall s \in \{t, \dots, t+H-1\} \\ \phi_t^{\mathrm{OLFC}}(x_t, h_t) = u_t^\star & \end{array} \right.$$

# Stochastic Dynamic Programming (SDP)

- We compute value functions **offline**

$$
\begin{aligned}
V_T(x) &= 0 \\
V_t(x) &= \min_{u \in \mathcal{U}(x)} \sum_{\sigma \in \mathbb{S}^{\text{off}}} \pi_{t+1}^{\text{off},\sigma} \Big( L_t(u, w_{t+1}^{\text{off},\sigma}) + V_{t+1}\big(f(x,u)\big) \Big)
\end{aligned}
$$

- We use value functions to compute **online** controls at time $t \in \{0, \dots, T-1\}$

$$
\begin{cases}
u_t^\star \in \arg\min_{u} \sum_{\sigma \in \mathbb{S}^{\text{on}}} \pi_{t+1}^{\text{on},\sigma} \Big( L_t(u, w_{t+1}^{\text{on},\sigma}) + V_{t+1}\big(f(x,u)\big) \Big) \\
\phi_t^{\text{SDP}}(x_t, h_t) = u_t^\star \; .
\end{cases}
$$

(If uncertainties $\mathbf{W}_1, \dots, \mathbf{W}_T$ are stagewise independent SDP gives an optimal solution)

# Modeling uncertainties with an AR($k$) process

- We model the net demand $z_t = d_t - g_t$ with an AR($k$) process

$$\mathbf{Z}_{t+1} = \sum_{j=0,\dots,k-1} \alpha_t^j \mathbf{Z}_{t-j} + \beta_t + \boldsymbol{\epsilon}_{t+1} \ , \ \ \forall t \in \{0, \dots, T-1\}$$

- We **extend the state** to $\widetilde{x}_t = (x_t, z_t, \dots, z_{t-k+1}) \in [0,1] \times \mathbb{R}^k$ with a new dynamics

$$\widetilde{f}_t(\widetilde{x}_t, u_t, \epsilon_{t+1}) = \begin{pmatrix} f(x_t, u_t) \\ \sum_{j=0,\dots,k-1} \alpha_t^j z_{t-j} + \beta_t + \epsilon_{t+1} \\ z_t, \dots, z_{t-k+2} \end{pmatrix}$$

# SDP-AR($k$)

Similarly, we compute value functions $\widetilde{V}_t$ **offline** and use them for computing **online** controls at time $t \in \{0, \dots, T-1\}$

$$\begin{cases} u_t^\star \in \arg\min_u \sum_{\sigma \in \mathbb{S}^{\text{on}}} \pi_{t+1}^{\text{on},\sigma} \Big( \widetilde{L}_t(\widetilde{x}_t, u, \epsilon_{t+1}^{\text{on},\sigma}) + \widetilde{V}_{t+1}\big(\widetilde{f}_t(\widetilde{x}_t, u, \epsilon_{t+1}^{\text{on},\sigma})\big)\Big) \\ \phi_t^{\text{SDP-AR}}(x_t, h_t) = u_t^\star \; . \end{cases}$$

(If uncertainties $\boldsymbol{\epsilon}_1, \dots, \boldsymbol{\epsilon}_T$ are stagewise independent SDP-AR($k$) gives an optimal solution)

# Outline of the section

4. **Numerical examples**

   Controller design techniques

   Numerical results

# Experimental setting

- We use EDF energy tariff
- For each site $i \in \{1, \dots, 70\}$, we designed battery parameters $(c^i, \overline{u}^i, \rho_c^i, \rho_d^i)$ with Schneider Electric
- We split the data into **chronicles of 1 week**
  - 60% of **calibration** (training) data
  - 40% of **simulation** (testing) data
    (giving a total of 2474 simulation chronicles)

# Normalized score per design technique

| | Normalized score | Offline time (seconds) | Online time (seconds) |
|---|---|---|---|
| **MPC** | 0.487 | 0.00 | $9.82\ 10^{-4}$ |
| **OLFC-10** | 0.506 | 0.00 | $1.14\ 10^{-2}$ |
| **OLFC-50** | 0.513 | 0.00 | $8.62\ 10^{-2}$ |
| **OLFC-100** | 0.510 | 0.00 | $1.87\ 10^{-1}$ |
| **SDP** | 0.691 | 2.67 | $3.09\ 10^{-4}$ |
| **SDP-AR(1)** | 0.794 | 38.1 | $4.44\ 10^{-4}$ |
| **SDP-AR(2)** | 0.795 | 468 | $5.55\ 10^{-4}$ |
| **Upper bound** | **1.0** | - | - |

# Detailed gain over the 70 sites

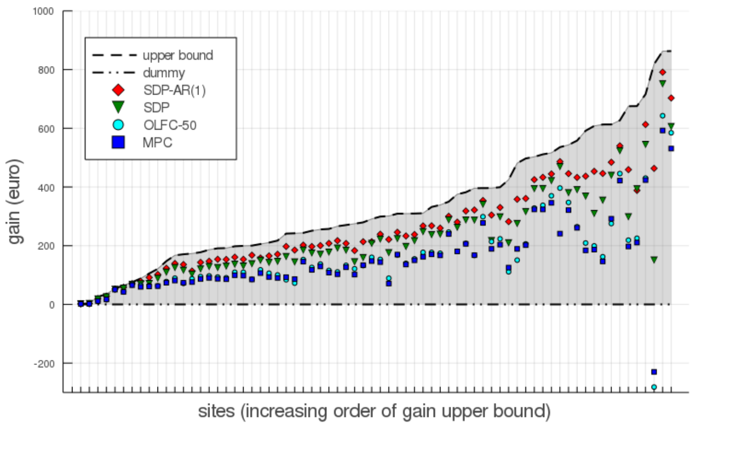



**Figure 5:** Gain $G^i(\phi^i)$ per sites $i \in I$ of controller design techniques MPC, OLFC-50, SDP and SDPAR(1), with anticipative gain $\underline{G}^i$ (dashed line) and gain of a dummy controller (dashed and dotted line)

# Outline of the presentation

# Conclusion

- EMSx provides a **dataset**, a **mathematical framework** and a **software** to compare microgrid controller techniques
- All components of EMSx are **publicly available** [1]
- EMSx makes it easy to implement and evaluate a large class of microgrid control techniques
- Further details are available in our submitted paper [2]

[1] https://github.com/adrien-le-franc/EMSx.jl
[2] https://hal.archives-ouvertes.fr/hal-02425913/document