Optimization Project in Energy ENT306

Elise Grosjean
Ensta-Paris

Ensta-Paris
Institut Polytechnique de Paris

1 Introduction

2 Deterministic model

Introduction

- **Main goal:** programming numerical methods (Energy Management System) for a very simple model of a microgrid.

- **Microgrid:** a set consisting of the following elements:
  - a small-size **electrical load** (a building, a few houses)
  - a source of **renewable energy** (solar panels, a wind turbine)
  - a **storage** device (Energy Storage System, a battery)
  - a **macrogrid** (an access to a large-scale energy network).

Introduction

- **Time scale:** the decisions are taken **every hour** during the **day**.
- **Optimization variables** (at each time step) :
  - the amount of energy to be **stored** or **withdrawn** from the battery
  - the amount of energy **bought** or **sold** to the network.
- **Contraints:**
  - nonnegativity of variables
  - evolution of the battery
  - storage capacity of the battery.
- **Cost function:**
  - Cost of the energy bought on the network...
  - minus the cost of the energy sold on the network.

Introduction

- **Main challenge:**
  - The electrical load and the production of renewable energy are **random**.
  - No available probabilistic model, instead the history of electrical load and solar production.
- **Tools and mathematical concepts:**
  - Dynamic programming (temporal aspect for decision process).
  - Autoregressive processes.
- **Optimization is useful for...**
  - minimizing the management costs
  - modelling random processes
  - describing some functions, for which no analytical expression is available (interpolation).

Introduction

- **Philosophy:**
  - **Compromise** between the complexity of stochastic modelling and solvability of the problem.
  - Emphasis on the **mathematical approach**. Maths concepts useful in other application contexts.
  - We will work with models of **increasing** complexity.
- **Warning:**
  - (Very) simplified models, with artificial data.
  - The purpose is not to conclude on the relevance of the introduction of such or such technology.

Introduction

- **References :**

  - Le Franc, Carpentier, Chancelier, de Lara. EMSx: an Energy Management System numerical benchmark. Energy System, 2021.
  - Hafiz, Awal, de Queiroz, Husain. Real-time Stochastic Optimization of Energy Storage Management using Rolling Horizon Forecasts for Residential PV Applications, 2019.
  - Olivares et al. Trends in microgrid control. IEEE Transactions on smart grid, 2014.

Introduction

Organisation:

- 6 units: January 17, January 07, January 14, January 21, January 28, February 4.
- Work in pairs (please form the groups by next week).
- Programming with Python.
- Evaluation: programming exercises to solve + work in class.

1 Introduction

2 Deterministic model

Deterministic model

- Horizon: 24 hours, stepsize: 1 hour.
  Optimization over $T = 24$ intervals.
- Optimisation variable :
  - $x(s)$: state of charge of the battery at time s, $s = 1, ..., T + 1$
  - $a(s)$: amount of electricity bought on the network ($s = 1, ..., T$).
  - $v(s)$: amount of energy sold on the network ($s = 1, ..., T$).
- Parameters:
  - $d(s)$: net demand of energy (load minus solar production) at time s, $s = 1, ..., T$.
  - $P_a(s)$: unitary buying price of energy at time s
  - $P_v(s)$: unitary selling price of energy at time s
  - $x_{\max}$: storage capacity of the battery.

Remark: the demand is supposed to be deterministic (that is to say, known in advance), for the moment.

Deterministic model

- Contraints:
  - $x(s+1) = x(s) - d(s) + a(s) - v(s), \forall s = 1, ..., T$
  - $x(1) = 0$
  - $a(s) \geq 0, \forall s = 1, ..., T$
  - $v(s) \geq 0, \forall s = 1, ..., T$
  - $0 \leq x(s) \leq x_{\max}, \forall s = 1, \ldots T+1.$

- Cost function to be minimized:

$$J(x, a, v) = \sum_{s=1}^{T} \Big(P_a(s)a(s) - P_v(s)v(s) \Big).$$

The buying and selling prices P_a and P_v depend on time. It holds: $P_a(s) > P_v(s)$, so that it is useless to try to buy and sell electricity on the network at the same time!

Deterministic model

Exercise 1

1. Write and solve the optimization problem in a form that is compatible with the function `linprog` of Python.

Deterministic model

Main idea behind **dynamic programming:**

- We **parametrize** the problem to be solved $\rightsquigarrow$ a sequence of problems of increasing complexity.
- We look for a **relation** ("dynamic programming principle") between the optimal values of the different problems.

Parameters :

- Initial time $t \in \{1, ..., T+1\}$.
- Initial state-of-charge of the battery $y \in [0, x_{\max}]$.

We are interested in the problem with $t = 1$ and $y = 0$.

Deterministic model

Parameterized problem:

$$V(t,y) = \inf_{\substack{x(t),x(t+1),\dots,x(T+1)\\ a(t),a(t+1),\dots,a(T)\\ v(t),v(t+1),\dots,v(T)}} \sum_{s=t}^{T} P_a(s)a(s) - P_v(s)v(s) \qquad (P(t,y))$$

under the constraints:

- $x(s+1) = x(s) - d(s) + a(s) - v(s),\ \forall s = t, \dots, T$
- $x(t) = y$
- $a(s) \geq 0,\ \forall s = t, \dots, T$
- $v(s) \geq 0,\ \forall s = t, \dots, T$
- $0 \leq x(s) \leq x_{\max},\ \forall s = t, \dots T+1.$

The function V is called **value function**; it plays a crucial role, in particular in the treatment of the stochastic version of the problem.

Deterministic model

Parameterized problem:

$$V(t,y) = \inf_{\substack{x(t),x(t+1),\dots,x(T+1)\\ a(t),a(t+1),\dots,a(T)\\ v(t),v(t+1),\dots,v(T)}} \sum_{s=t}^{T} P_a(s)a(s) - P_v(s)v(s) \qquad (P(t,y))$$

under the constraints:

- $x(s+1) = x(s) - d(s) + a(s) - v(s),\ \forall s = t, \dots, T$
- $x(t) = y$
- $a(s) \geq 0,\ \forall s = t, \dots, T$
- $v(s) \geq 0,\ \forall s = t, \dots, T$
- $0 \leq x(s) \leq x_{\max},\ \forall s = t, \dots T+1.$

The function V is called **value function**; it plays a crucial role, in particular in the treatment of the stochastic version of the problem.

Deterministic model and dynamic programming

- Dynamic programming = Temporal decomposition with smaller sub-problems
- Dynamic programming principle = Recursive relation
  - process ensuring that each subproblem is solved optimally, taking into account future implications and not the previous decisionst.
- Interests of the decomposition:
  - reducing complexity
  - optimality of the global solution thanks to Bellman optimality principle
  - clearer implementation
  - ...

Deterministic model and dynamic programming

Theorem [Dynamic programming principle]

The following holds true:

$$V(t,y) = \inf_{(z,a,v)\in\mathbb{R}^3} \quad P_a(t)a - P_v(t)v + V(t+1,z),$$

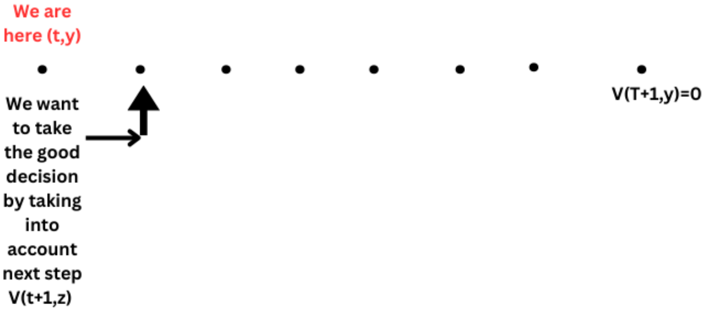
$$\text{s.t.:} \quad \begin{cases} a \geq 0 \\ v \geq 0 \\ z = y - d(t) + a - v \\ 0 \leq z \leq x_{\max} \end{cases} \qquad (DP(t,y))$$

for all $t \in \{1, \ldots T\}$ and $y \in [0, x_{\max}]$ and

$$V(T+1, y) = 0.$$

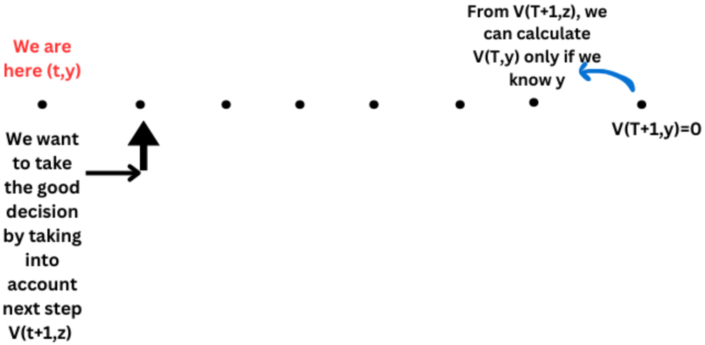
Deterministic model and dynamic programming

$$V(t,y) = \inf_{(z,a,v)\in\mathbb{R}^3} P_a(t)a - P_v(t)v + V(t+1,z)$$



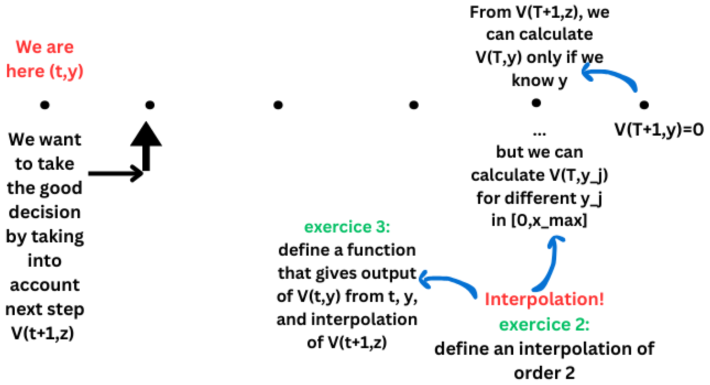
Deterministic model and dynamic programming

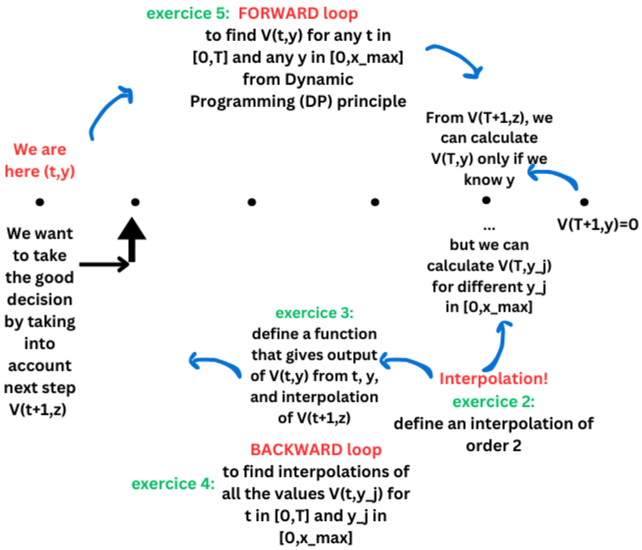
$$V(t,y) = \inf_{(z,a,v)\in\mathbb{R}^3} P_a(t)a - P_v(t)v + V(t+1,z)$$



Deterministic model and dynamic programming

$$V(t,y) = \inf_{(z,a,v)\in\mathbb{R}^3} P_a(t)a - P_v(t)v + V(t+1,z)$$




exercice 5: FORWARD loop
to find V(t,y) for any t in [0,T] and any y in [0,x_max] from Dynamic Programming (DP) principle
From V(T+1,z), we can calculate V(T,y) only if we know y
We are here (t,y)
V(T+1,y)=0
We want to take the good decision by taking into account next step V(t+1,z)
...
but we can calculate V(T,y_j) for different y_j in [0,x_max]
exercice 3:
define a function that gives output of V(t,y) from t, y, and interpolation of V(t+1,z)
Interpolation!
exercice 2:
define an interpolation of order 2
BACKWARD loop
exercice 4:
to find interpolations of all the values V(t,y_j) for t in [0,T] and y_j in [0,x_max]

Deterministic model

Computation of the value function

- Let us suppose that the function $z \mapsto V(t+1, z)$ is known, with $t \in \{1, ..., T\}$. By solving $DP(t, y)$ for all possible values of y, we can compute $y \mapsto V(t, y)$.
- In practice, we can only solve $DP(t, y)$ for a sample of values of $y \in [0, x_{\max}]$. We evaluate $V(t, y)$ for those values of y; then we interpolate.
- In that way, we can compute (an approximation of) the value function, in a recursive and backward fashion (from $T+1$ to 1).

Deterministic model

Solving the initial problem.

- Let (z, a, v) denote a solution of $DP(1, 0)$.
  Let us set: $\bar{x}(2) = z$, $\bar{a}(1) = a$, $\bar{v}(1) = v$.
- Let (z, a, v) be a solution of $DP(2, \bar{x}(2))$.
  Let us set: $\bar{x}(3) = z$, $\bar{a}(2) = a$, $\bar{v}(2) = v$.
- Let (z, a, v) be a solution of $DP(3, \bar{x}(3))$.
  Let us set: $\bar{x}(4) = z$, $\bar{a}(3) = a$, $\bar{v}(3) = v$.
- ... and so on, until the resolution of $DP(T, \bar{x}(T))$.

Deterministic model

- Let $y \in \mathbb{R}^J$ and let $val = V(t, y) \in \mathbb{R}^J$. We consider a function f such that $val_j = f(y_j)$, for all$j = 1, ..., J$.
- We interpolate f with a second-order polynomial, by solving

$$\inf_{\alpha \in \mathbb{R}^3} \sum_{j=1}^{J} \left(\alpha_1 + \alpha_2 y_j + \alpha_3 y_j^2 - val_j\right)^2.$$

- Let $\bar{\alpha}$ be the solution. We get the approximation:

$$f(y) \approx \bar{\alpha}_1 + \bar{\alpha}_2 y + \bar{\alpha}_3 y^2.$$

Exercise 2

Write a function `interpolate` implementing the interpolation with a second-order polynomial.
Inputs: J, $y \in \mathbb{R}^J$, $val_j \in \mathbb{R}^J$. Outputs: $\alpha \in \mathbb{R}^3$.

Deterministic model

Exercise 3

Write a function DP_solve with output the solution (z, a, v) of problem $DP(t, y)$.

We assume that the function $V(t+1, \cdot)$ is approximated by a second-order polynomial, described by a coefficient $\alpha \in \mathbb{R}^3$.

Input: $t \in \{1, ..., T\}$, $y \in [0, x_{\max}]$, $\alpha \in \mathbb{R}^3$.

Deterministic model

Exercise 4

Write a function DP_backward which compute a polynomial approximation of V, in the form of a matrix $\alpha \in \mathbb{R}^{(T+1)\times 3}$, that is to say:

$$V(t, y) \approx \alpha_{t,1} + \alpha_{t,2} y + \alpha_{t,3} y^2.$$

We will proceed in a recursive fashion:

- Given $\alpha_{t+1,1}$, $\alpha_{t+1,2}$, $\alpha_{t+1,3}$, evaluate $V(t, y_j)$ for all $j = 1, ..., J$, where $y_j = (j-1)/(J-1)x_{\max}$.
- Calculate $\alpha_{t,1}$, $\alpha_{t,2}$, and α_{t_3} by interpolating the values of $V(t, \cdot)$.

Deterministic model

Exercise 5

Write a function `DP_forward` computing the solution to the original problem. We will first make use of `DP_backward` to get an approximation of the value function.