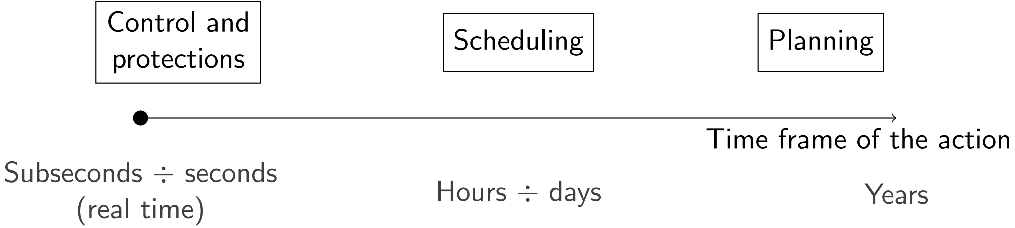
# Scheduling Energy Storage System Operations

Prof. Dr. Fabrizio Sossan (HES-SO Valais-Wallis)
fabrizio.sossan@hevs.ch

EPFL EE-466
Week 3

Slides compiled on: September 26, 2024

# Control, scheduling, and planning of energy storage

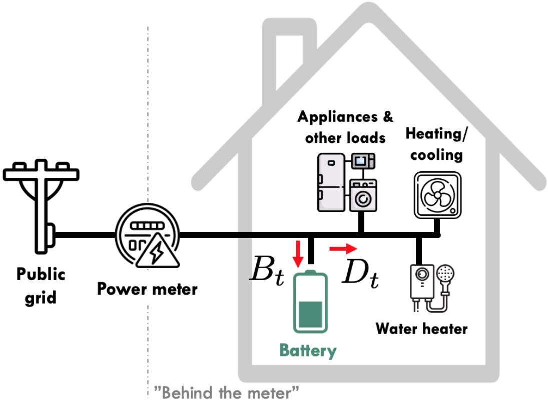


# Control, scheduling, and planning of energy storage: Examples

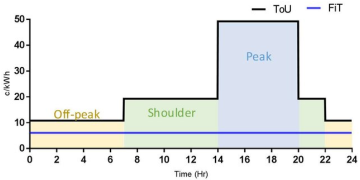
| Action | Electric Vehicle | Grid application |
|---|---|---|
| Control | Battery management system setting the maximum discharging power | Satisfy application requirements (e.g., maximize PV self-consumption) |
| Scheduling | Route planning/deciding on charging stops | Designing a suitable state-of-charge trajectory so that the system has energy when needed |
| Planning | How large of a battery one needs? | Size (power rating and energy capacity), site (if that's an option), technology (ageing) |

Because the state of charge/energy availability of an energy storage resource depends on the control history and application (the problem has time-coupling constraints), these procedures are somehow intertwined, requiring to design about these actions in an unified setting.

# Work example: Optimizing consumption with a time of use-electricity tariff




- Time-of-use tariff (ToU) $p(t)$:




- Power demand $D_t$ is inflexible and random (at best, it can be forecasted).
- Battery $B_t$ is controllable (while respecting its the physical constraints).

Problem statement: Find the battery charging/discharging power that minimizes the cost of electricity over an operational time horizon (say, the next 24 hours).

# Convex optimization problems

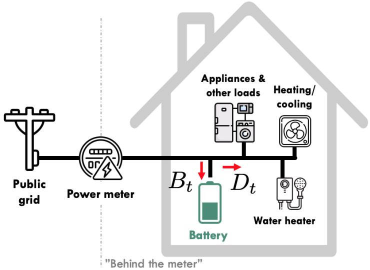
$$
\begin{aligned}
\underset{x}{\text{minimize}} \quad & f(x) \\
\text{subject to} \quad & g_i(x) = 0, \quad i = 0, \ldots, I \\
& h_j(x) \leq 0, \; j = 0, \ldots, J
\end{aligned}
$$

The problem above is convex if $f(x)$ is a convex function in $x$, all $g_i(x)$ are linear functions in $x$, and all $h_j(x)$ are convex functions in $x$. $x$ must be a real number (scalar) or a vector (an order set of scalars).

A local minimum $f(x^o)$, provided that it exists, is a global minimum.

Reference on convex optimization (pdf freely available): S. Boyd and L. Vandenberghe, Convex Optimization.

# Problem formulation




- $t$: index of the time interval, with duration $\Delta$ (in hours).
- $p(t)$: price of electricity (\$/kWh).
- $B(t)$: battery charging power (kW); when negative, the battery discharges.
- $D(t)$: power demand, assumed known from forecast (kW).

All quantities assumed piecewise constant in the time interval $\Delta$.

Problem objective: minimize electricity cost over an optimization horizon $t = 0, \ldots, T-1$, with $T$ number of intervals.

$$\text{Electricity cost over the optimization interval} = \Delta \sum_{t=0}^{T-1} p(t) \cdot (B(t) + D(t))$$

# Problem formulation (cont'd)

The constraints of the problem are:

- The charging/discharging power of the battery should not exceed the power rating $\bar{P}$ (kW):

$$|B(t)| \leq \bar{P}, \quad t = 0, \ldots, T-1. \tag{1}$$

- The battery's state of energy should be non-negative and not exceed the energy capacity $\bar{E}$ in kWh (as a first approximation; one should have customizable margins). Recalling the ideal SOE model seen previously ([1])

$$SOE(t+1) = SOE(0) + \Delta \sum_{\tau=0}^{t} B(\tau),$$

where $SOE(0)$ is given, the SOE constraints read as:

$$0 \leq SOE(t) \leq \bar{E}, \quad t = 1, \ldots, T. \tag{2}$$

Assumptions: ideal battery efficiency, constant power capability, and no reactive power. Non-ideal efficiency will be introduced next; other assumptions will be progressively replaced by more appropriate physical constraints throughout the course.

# Problem formulation (cont'd)

The problem formulation (cost + constraints) reads as:

$$
\begin{aligned}
\underset{B_0, B_1, \ldots, B_{T-1}}{\text{minimize}} \quad & \left\{ \Delta \sum_{t=0}^{T-1} p(t) \cdot (B(t) + D(t)) \right\} \\
\text{subject to} \quad & -\bar{P} \leq B(t) \leq \bar{P}, \qquad && t = 0, \ldots, T-1 \\
& 0 \leq SOE(0) + \Delta \sum_{\tau=0}^{t} B(\tau) \leq \bar{E}, \qquad && t = 0, \ldots, T-1
\end{aligned}
$$

where $SOE(0)$ is given.

Linear optimization problem (also convex) because all functions (cost, equalities, and inequalities) are linear functions in the problem variable $B(t)$ for all $t$.

One could use algebraic modelling languages (GAMS, CVX, YALMIP and others) to compile a problem understandable by a solver and solve it.

# Conversion to standard matrix form

Documentation | Examples | Functions | Apps | Videos | Answers

## linprog

Solve linear programming problems

### Syntax

```
x = linprog(f,A,b)
x = linprog(f,A,b,Aeq,beq)
x = linprog(f,A,b,Aeq,beq,lb,ub)
x = linprog(f,A,b,Aeq,beq,lb,ub,options)
x = linprog(problem)
[x,fval] = linprog(___)
[x,fval,exitflag,output] = linprog(___)
[x,fval,exitflag,output,lambda] = linprog(___)
```

### Description

Linear programming solver

Finds the minimum of a problem specified by

$$\min_x f^T x \text{ such that } \begin{cases} A \cdot x \leq b, \\ Aeq \cdot x = beq, \\ lb \leq x \leq ub. \end{cases}$$

$f$, $x$, $b$, $beq$, $lb$, and $ub$ are vectors, and $A$ and $Aeq$ are matrices.

- Optimization solvers (on the left, the doc page of `linprog` in Matlab) often accept matrices as way to specify an optimization problem.
- Matrix form is typically the way that solvers "attack" the problem.

# Conversion to matrix form: Cost function

$$\text{Electricity cost} = \Delta \sum_{t=0}^{T-1} p(t) \cdot (B(t) + D(t)) =$$

By grouping the scalar values into vectors, one can render the expression above as:

$$= \Delta \begin{bmatrix} p(0) & p(1) & p(2) & \dots p(T-1) \end{bmatrix} \begin{bmatrix} B(0) + D(0) \\ B(1) + D(1) \\ B(2) + D(2) \\ \vdots \\ B(T-1) + D(T-1) \end{bmatrix} =$$

and finally

$$= \Delta p_{[0,T]}^{\top} B_{[0,T]} + \Delta p_{[0,T]}^{\top} D_{[0,T]}. \tag{3}$$

where $^T$ denotes transpose, and the notation $x_{[0,T]}$ denotes a vector with $T$ elements, starting from time interval 0.

# Conversion to matrix form: SOE Constraints

Rendering

$$SOE(t+1) = SOE(0) + \Delta \sum_{\tau=0}^{t} B(\tau)$$

explicit for $t = 0, 1, 2, \ldots, T-1$ yields:

$$\begin{aligned}
SOE(1) &= SOE(0) + \Delta B(0) \\
SOE(2) &= SOE(0) + \Delta \left(B(0) + B(1)\right) \\
SOE(3) &= SOE(0) + \Delta \left(B(0) + B(1) + B(2)\right) \\
&\vdots \\
SOE(T) &= SOE(0) + \Delta \left(B(0) + B(1) + B(2) + \cdots + B(T-1)\right)
\end{aligned}$$

# Conversion to matrix form: SOE Constraints (cont'd)

In matrix form, the former expression becomes:

$$\begin{bmatrix} SOE(1) \\ SOE(2) \\ SOE(3) \\ \vdots \\ SOE(T) \end{bmatrix} = SOE(0) \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} + \Delta \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 1 & 1 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \dots & 1 \end{bmatrix} \begin{bmatrix} B(0) \\ B(1) \\ B(2) \\ \vdots \\ B(T-1) \end{bmatrix}$$

and more compactly as:

$$SOE_{[1,T]} = SOE(0) \cdot \mathbf{1}_T + \Delta\phi_T \cdot B_{[0,T]}.$$

where $\phi_T$ is the all-one full-rank lower triangular matrix of size $T$. Using the expression above to render the SOE constraints in (2) gives:

$$\mathbf{0}_T \preceq SOE(0) \cdot \mathbf{1}_T + \Delta\phi_T \cdot B_{[0,T]} \preceq \bar{E} \cdot \mathbf{1}_T \tag{4}$$

where $\preceq$ denotes component-wise inequalities.

# Conversion to matrix form: SOE Constraints (cont'd)

Rearranging the terms of the right-hand-side inequality constraints of (4) reads as:

$$\Delta\phi_T \cdot B_{[0,T]} \preceq (\bar{E} - SOE(0)) \cdot \mathbf{1}_T \tag{5}$$

Consider now the left-hand-side inequalities of (4). Rearranging and multiplying by -1 yield:

$$-\Delta\phi_T \cdot B_{[0,T]} \preceq SOE(0) \cdot \mathbf{1}_T \tag{6}$$

Assembling (5) and (6) together yields :

$$\begin{bmatrix} \Delta\phi_T \\ -\Delta\phi_T \end{bmatrix} B_{[0,T]} \preceq \begin{bmatrix} (\bar{E} - SOE(0)) \cdot \mathbf{1}_T \\ SOE(0) \cdot \mathbf{1}_T \end{bmatrix}$$

which is an expression in the form $Ax \preceq b$, compatible with what `linprog` expects as an input argument.

# Conversion to matrix form: Power constraints

Power constraints

$$-\bar{P} \leq B(t) \leq \bar{P}, \quad t = 0, \ldots, T-1$$

can be written in matrix form as:

$$-\bar{P} \cdot \mathbf{1}_T \preceq \begin{bmatrix} 1 & 0 & 0 & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 1 \end{bmatrix} \begin{bmatrix} B(0) \\ B(1) \\ B(2) \\ \vdots \\ B(T-1) \end{bmatrix} \preceq \bar{P} \cdot \mathbf{1}_T$$

By applying the same reasoning as before, one can derive:

$$\begin{bmatrix} 1_{T\times T} \\ -1_{T\times T} \end{bmatrix} B_{[0,T]} \preceq \begin{bmatrix} \bar{P} \cdot \mathbf{1}_T \\ \bar{P} \cdot \mathbf{1}_T \end{bmatrix},$$

where $1_{T\times T}$ is the identity matrix of size $T$.

# Formulation of the optimization problem in matrix form

The optimization problem can be written as follows ([1]):

$$
\begin{aligned}
B^{\star}_{[0,T]} = & \arg\min \left\{ p^{\top}_{[0,T]} B_{[0,T]} \right\} \\
& \text{subject to:} \\
& \begin{bmatrix} \Delta\phi_T \\ -\Delta\phi_T \\ 1_{T\times T} \\ -1_{T\times T} \end{bmatrix} B_{[0,T]} \preceq \begin{bmatrix} (\bar{E} - SOE(0)) \cdot \mathbf{1}_T \\ SOE(0) \cdot \mathbf{1}_T \\ \bar{P} \cdot \mathbf{1}_T \\ \bar{P} \cdot \mathbf{1}_T \end{bmatrix}
\end{aligned} \tag{7}
$$

$B^{\star}$ is the battery power that minimizes the cost of electricity.

[1] Compared to the cost function enunciated in (3), rescaling and translation terms can be neglected because the argument of the minimization problem is invariant under them (not the cost, though).

# Inclusion of power efficiency

# State-of-energy with efficiency

When including efficiency, assuming the efficiency is constant with the power (it is not, but in scheduling applications, that's a convenient choice to avoid further modeling complication), the model can be modified as this (as seen, already):

$$\mathsf{SOE}\,(t+1) = SOE(0) + \Delta \sum_{\tau=0}^{t} \left( \eta\,[B(\tau)]^{+} - \frac{1}{\eta}[B(\tau)]^{-} \right).$$

This function is non-convex and its inclusion in the constraints of the previous optimization problem will render the problem non-convex.

# Charging and discharging power

A modeling alternative widely adopted is to separate charging and discharging power into (non-negative) independent variables, which we call $B^+(t)$, and $B^-(t)$.

$$\mathrm{SOE}\,(t+1) = SOE(0) + \Delta \sum_{\tau=0}^{t} \left( \eta B^+(\tau) - \frac{1}{\eta} B^-(\tau) \right). \tag{8}$$

This way restores the linearity of the models in the power(s). However, because the battery cannot charge and discharge at the same time, one should ensure that these variables are mutually exclusive. Two ways of doing this are:

1. using an additional set of binary variables; and
2. penalizing the norm-2 of the sum of the variables in the cost so as to penalizing mutual activation.

We will see the first way.

# Mutually exclusive charging/discharging power with binary variables

Say $c(t) \in \{0, 1\}$ for $t = 0, \dots, T-1$ is an additional set of binary variables added to the optimization problem. Mutually exclusive charging and discharging behavior of the battery can be enforced with the following constraints:

$$
\begin{aligned}
B^+(t) &\geq 0, & t &= 0, 1, \dots, T-1 \\
B^-(t) &\geq 0, & t &= 0, 1, \dots, T-1 \\
B^+(t) &\leq c(t) \cdot \bar{P}, & t &= 0, 1, \dots, T-1 \\
B^-(t) &\leq (1 - c(t)) \cdot \bar{P}, & t &= 0, 1, \dots, T-1
\end{aligned}
$$

For example, say $c(1) = 1$, only the charging power can be positive; discharging power is bound to zero by the second and fourth constraints here above.

# 1. Mutually exclusive ... (cont'd)

When reformulating the optimization model seen above, one should now consider that the battery power is given by the sum of two (mutually exclusive) contributions to readjust cost function and constraints.

For example, the electricity cost for battery operations (the cost function) is now:

$$\sum_{t=0}^{T-1} p(t)\left(B^{+}(t) - B^{-}(t)\right) = \sum_{t=0}^{T-1} \left\{p(t)B^{+}(t) - p(t)B^{-}(t)\right\}.$$

# 1. Mutually exclusive ... implementation

The (column) vector of decision variables now consists of:

$$x = \begin{bmatrix} B^{+}_{[0,T]}{}^{\top} & B^{-}_{[0,T]}{}^{\top} & c^{\top}_{[0,T]} \end{bmatrix}^{\top}$$

where both $B^{+}_{[0,T]}$, $B^{-}_{[0,T]}$, and $c_{[0,T]}$ are column vectors of $T$ real and binary variables, respectively.

The modified optimization problem reads as:

$$\min \left\{ \begin{bmatrix} p^{\top}_{[0,T]} & -p^{\top}_{[0,T]} & \mathbf{0}^{\top}_{T} \end{bmatrix} x \right\}$$

subject to:

$$\begin{bmatrix} \Delta\phi_T & -\Delta\phi_T & 0_{T\times T} \\ -\Delta\phi_T & \Delta\phi_T & 0_{T\times T} \\ -1_{T\times T} & 0_{T\times T} & 0_{T\times T} \\ 0_{T\times T} & -1_{T\times T} & 0_{T\times T} \\ 1_{T\times T} & 0_{T\times T} & -1_{T\times T}\cdot\bar{P} \\ 0_{T\times T} & 1_{T\times T} & 1_{T\times T}\cdot\bar{P} \end{bmatrix} \underbrace{\begin{bmatrix} B^{+}_{[0,T]} \\ B^{-}_{[0,T]} \\ c_{[0,T]} \end{bmatrix}}_{x} \preceq \begin{bmatrix} (\bar{E} - SOE(0))\cdot\mathbf{1}_T \\ SOE(0)\cdot\mathbf{1}_T \\ \mathbf{0}_T \\ \mathbf{0}_T \\ \mathbf{0}_T \\ \bar{P}\cdot\mathbf{1}_T \end{bmatrix}$$

# Closing remarks

- The problem computes the battery's charging/discharging operations that minimize the cost of electricity.
- The formulation can be applied in practice by using a receding-horizon policy: at each time interval, the problem is re-computed using updated information for the new time horizon. This allows for producing a "control action" at each time interval (sent for actuation) as well as accounting for forecast errors and new prices (e.g., new electricity prices). Same spirit as model predictive control (if you have seen this in other courses).
- The control action is sent to the battery. The real-time controller will take care of implementing this on a best-effort basis given the current conditions of the battery (see initial slides).
- If battery operative limits are well modelled within the scheduling problem, probably the control action commanded by the scheduler will be feasible, and the real-time controller will allow it.
- More constraints and models can be added ..

# Exercise 3.1

1. Implement in Matlab the problem (7) to schedule charging/discharging operations of a battery system. Use input information given in the next slide and the `linprog` function.
2. Compute the electricity cost over the horizon and specify whether it is a revenue or a loss.
3. It is not a good idea to bring the battery near its state of energy limits, for various reasons. Modify the former formulation so as to have tighter bounds on the allowed SOE constraints:

$$0 \leq SOE(t) \leq \bar{E}, \text{for all } t \quad \longrightarrow \quad 0.15\bar{E}\cdot \leq SOE(t) \leq 0.85 \cdot \bar{E}, \text{for all } t$$

   and compare the electricity cost against the former case.
4. Plot the optimal charging/discharging trajectory of the battery. Do you have any concern with it? What's the fundamental (mathematical) reason for it?
5. If you spot a problem, what can you do to solve it? Try it out in Matlab and verify the result.

# Exercise 3.1 (cont'd)

```
% Copy and paste in Matlab
E_bar = 15; % Energy capacity [kW]
P_bar = 10; % Power capacity [kWh]
SOE_0 = 7; % [kWh]
T = 24; % number of samples and look-ahead time
Delta = 1; % [hours]
% ToU electricity tariff ($/kWh) - if in doubt, assume it to be valid for both consuming
% and generating power. If not in doubt, ignore this comment (to be rediscussed in class).
c(1:7) = 0.05;
c(8:10) = 0.08;
c(11:17) = 0.15;
c(18:21) = 0.35;
c(22:24) = 0.05;
stairs(c);
xlabel('Time [hours]');
ylabel('Tariff ($/kWh)')
title('A peek to the price signal')
% .. now it is on you!
```

# Exercise 3.2

You are the head of the electrical infrastructure of the company ACME. You bought and programmed a battery system to optimize electricity consumption with ToU tariffs (or spot-market electricity prices if the company is large enough to access the electricity market), saving your company many thousands of CHF in electricity bills. That was so good that ACME awarded a public holiday to all their people on your birthday. Now, ACME's process department has bought a new piece of equipment that, when activated, consumes a large amount of power for a short time interval, which could exceed the rating of the substation transformer, requiring a new expensive transformer. You think that perhaps your battery can help to mitigate this peak of power (i.e., performing so-called peak shaving), avoiding replacing the transformer.

- Modify the formulation of the scheduling problem so as to limit the power below a customizable threshold $\bar{D}$.
- Verify results using a toy example in Matlab (use the former case study as a start).
- What other input information is required for the problem formulation that was not present before?