Multiperiod OPF, inventory control, and storage

Josh Taylor

Section 4.1 in *Convex Optimization of Power Systems*

1 Basic storage modeling

Parameters:

- Energy capacity, $\bar{C} > 0$
- Power capacity, $\bar{T} > 0$
- Leakage, $0 \leq \alpha \leq 1$
- Injection and extraction losses, $0 \leq \eta_{\text{in}} \leq 1$, $\eta_{\text{out}} \geq 1$

Variables:

- State of charge, S^t
- Grid side power in/out, $U_{\text{in}}^t, U_{\text{out}}^t$ (Energy: $E^t = \Delta U^t$)

Model:

- Dynamics:

$$S^{t+1} = \alpha S^t + \eta_{\text{in}} U_{\text{in}}^t + \eta_{\text{out}} U_{\text{out}}^t$$

- Constraints:

$$0 \leq S^t \leq \bar{C}, \quad 0 \leq U_{\text{in}}^t \leq \bar{T}, \quad -\bar{T} \leq U_{\text{out}}^t \leq 0$$

- Apparent power limit instead of real power limit:

$$(U_{\text{in}}^t + U_{\text{out}}^t)^2 + Q^{t2} \leq \bar{T}^2$$

- Injection extraction complementarity:

$$U_{\text{in}}^t U_{\text{out}}^t = 0$$

Smarter way - disjunctive constraint:

$$0 \leq U^t_{\text{in}} \leq \sigma\bar{T}, \quad -\bar{T}(1-\sigma) \leq U^t_{\text{out}} \leq 0, \quad \sigma \in \{0,1\}$$

Interpretation

- When would simultaneous injection/extraction be useful? Negative nodal prices.
- Over a time period: inject first half, extract second.
- Rational for barring: can be detrimental to storage health, which might be unmodeled in dispatch routine.

What can we do with this?

- Convex optimization: MP-OPF, trajectory
- Dynamic programming: Inventory control, policy
- Also: LQR, state-space control

2 Multiperiod optimal power flow

- Optimal power flow is solved every 5 minutes (real-time dispatch) or faster
- Storage is a dynamic constraint between time periods
- Storage couples constraints across periods

SDP relaxation version:

$$\begin{aligned}
\min_{P^t,Q^t,V^t} \quad & \sum_{i,t} f_i^t(P_i^t) \\
\text{s.t.} \quad & P_{ij}^t + jQ_{ij}^t = (W_{ii}^t - W_{ij}^t)y_{ij}^* \\
& P_i^t + jQ_i^t = U_{i,\text{in}}^t + U_{i,\text{out}}^t + \sum_j P_{ij}^t + jQ_{ij}^t \\
& \underline{P}_i^t \le P_i^t \le \overline{P}_i^t \\
& \underline{Q}_i^t \le Q_i^t \le \overline{Q}_i^t \\
& \underline{V}_i^{t2} \le W_{ii}^t \le \overline{V}_i^{t2} \\
& P_{ij}^{t2} + Q_{ij}^{t2} \le \overline{S}_{ij}^{t2} \\
& W^t \succeq 0
\end{aligned}$$

Storage constraints:

$$\begin{aligned}
\textit{Linear} \quad & S^{t+1} = \alpha S_i^t + \eta_{i,\text{in}} U_{i,\text{in}}^t + \eta_{i,\text{out}} U_{i,\text{out}}^t \\
\textit{Linear} \quad & 0 \le S_i^t \le C_i \\
\textit{Linear} \quad & 0 \le U_{i,\text{in}}^t \le \bar{T}_i, \quad -\bar{T}_i \le U_{i,\text{out}}^t \le 0, \quad \text{or} \\
\textit{Convex quadratic} \quad & (U_{i,\text{in}}^t + U_{i,\text{out}}^t)^2 + Q_i^{t2} \le \overline{T}_i^2
\end{aligned}$$

Additional dynamics:

- Ramp constraints: $|P_i^{t+1} - P_i^t| \le \bar{R}_i$
- Rate of change-based costs: $f_i^t(P_i^t, P_i^{t+1})$

What does this capture?

- Reactive power support
- Load-shifting
- Shifting non-dispatchable supply for feasibility (duck curve)

2.1 Load-shifting

Intuition: simple example

- Power generated over two time periods: $P^1 + P^2 = D$

- Cost: $a(P^1)^2 + a(P^2)^2 + b(P^1 + P^2)$
- Optimum: $P^1 = P^2 = D/2$
- Flatter load yields more efficient generation.

Draw picture of load shifting
Multiperiod OPF gives us

- Optimal load shifting and reactive power support
- Does not capture: stability (regulation), power balancing (reserves) - require uncertainty modeling

3 Dynamic programming

Discrete-time dynamic system:

$$x_{t+1} = f_t(x_t, u_t, d_t), \quad t = 0, ..., N$$

(switch to subscript t)
Cost/value function that's additive over time:

$$J_0(x_0) = \min_{u \in U} \mathbb{E}\left[g_N(x_N) + \sum_{t=0}^{N-1} g_t(x_t, u_t, d_t)\right]$$

Components:

State: x_t

Control: u_t

Random input: d_t

Dynamics: f_t

Stage cost: g_t

A policy: $u_t = \sigma_t(x_t)$ - an instruction for any state.

- More flexible than trajectories (MOPF), valid under uncertainty
- Harder to obtain

- DP is a general formalism

Cost/value function from k onwards:

$$J_k(x_k) = \min_{u_k,\ldots,u_{N-1}\in U} \mathbb{E}\left[g_N(x_N) + \sum_{t=k}^{N-1} g_t(x_t, u_t, d_t)\right]$$

- Principle of optimality: Tail policy is optimal for the tail problem.
- First solve $N-1$, then $N-2$, so on. – less work than solving all at once.

(Informally) observe:

$$J_k(x_k) = \min_{u_k\in U} \mathbb{E}\left[g_k(x_k, u_k, d_k) + J_{k+1}(f_k(x_k, u_k, d_k))\right]$$

- The DP recursion
- Since we solve for u_k for all x_k, it is a policy $u_k = \sigma_k(x_k)$.

DP:

- Very general, but often intractable
- Often leads to analytical insights when they exist
- Popular starting point for computational approximations

4 DP for a single storage

Simplifications

- Neglect inject/extract inefficiencies (leakage ok)
- No power (ramp) constraints

Recall model:

$$S_{t+1} = \alpha S_t + U_t, \quad 0 \le S_t \le \bar{C}$$

Uncertainty:

- D_t - random (possibly non-Gaussian) energy, e.g. power imbalance
- Assume U_t chosen before D_t known
- D_t limited by capacity.

Augmented (nonlinear!) dynamics:

$$S_{t+1} = [\alpha S_t + U_t + D_t]_0^{\bar{C}} = \max\{\min\{\alpha S_t + U_t + D_t, \bar{C}\}, 0\}.$$

DP recursion

$$J_k(S_k) = \min_{U_k:\, 0 \le \alpha S_k + U_k \le \bar{C}} \mathbb{E}\left[g_k(S_k, U_k, D_k) + J_{k+1}\left([\alpha S_k + U_k + D_k]_0^{\bar{C}} \right) \right]$$

What is the cost?

- Arbitrage: $\sum_t \lambda_t U_t$, λ_t is price. Similar to load-shifting, why?
- Spillover: $\left| D_t - [D_t]_{-\alpha S_t - U_t}^{\bar{C} - \alpha S_t - U_t} \right|$.

Solution trick: 1-to-1 substitution.

$$Y_t \longleftrightarrow \alpha S_t + U_t$$

DP becomes:

$$J_k(S_k) = \min_{Y_k:\, 0 \le Y_k \le \bar{C}} \mathbb{E}\left[\lambda_k (Y_k - \alpha S_k) + \left| D_k - [D_k]_{-Y_k}^{\bar{C} - Y_k} \right| + J_{k+1}\left([Y_k + D_k]_0^{\bar{C}} \right) \right]$$

State only appears in one place, rewrite:

$$J_k(S_k) = -\lambda_k \alpha S_k + \min_{Y_k:\, 0 \le Y_k \le \bar{C}} \mathbb{E}\left[\lambda_k Y_k + \left| D_k - [D_k]_{-Y_k}^{\bar{C} - Y_k} \right| + J_{k+1}\left([Y_k + D_k]_0^{\bar{C}} \right) \right]$$

Observe: minimization doesn't depend on S_k. Define:

$$\begin{aligned} G_k(Y_k) &= \mathbb{E}\left[\lambda_k Y_k + \left| D_k - [D_k]_{-Y_k}^{\bar{C} - Y_k} \right| + J_{k+1}\left([Y_k + D_k]_0^{\bar{C}} \right) \right] \\ Z_k &= \operatorname*{argmin}_{Y_k:\, 0 \le Y_k \le \bar{C}} G_k(Y_k) \end{aligned}$$

Then

$$J_k(S_k) = -\lambda_k \alpha S_k + G_k(Z_k)$$

Optimal policy: reverse substitution:

$$U_k^* = \sigma(S_k) = Z_k - \alpha S_k$$

Properties:

- Optimal policy is affine in state: setpoint interpretation
- Optimal value fn. in each period affine - can solve backwards for Z with no enumeration over S_k:

- Heuristic basis for more detailed models

Same structure as classic inventory control

- A store/warehouse buys inventory (U_k)
- Random demand each day (D_k)
- Limited storage capacity ($0 \leq S_k \leq \bar{C}$)
- Unsold inventory, S_k, is stored (dynamics).

References