Convex Formulation of the Optimal Transmission Switching Problem

Salar Fattahi

Joint work with: Javad Lavaei, Alper Atamturk

FERC 2017

Department of Industrial Engineering and Operations Research
University of California, Berkeley
Outline

1. Introduction and problem formulation.

2. Tightening the linearized relaxation.
   - Effect on practical problems with linear objective.

3. Convex model of the problem with quadratic objective.
   - Different conic relaxations
   - Performance of the convex model in practical problems.

4. Boosting the speed using heuristic rounding schemes.
Power Systems

- **Power system:**
  - A large-scale system consisting of generators, loads, lines, etc.
  - Used for generating, transporting and distributing electricity.

**Optimal Transmission Switching Problem:** Find an optimal topology of the power network that minimizes the operational cost, subject to energy demand and operating constraints.
OTS Problem

• In plain language: Pick a subset of lines in the network that maximizes the performance.

• Intuition: Adding more lines increases the capacity of the network.
• Does the problem correspond to a conventional network flow problem? Min cost network flow or max flow problem.
• Answer: No!
• Key difference: We have more variables and constraints on the edges.
OTS Problem

- In plain language: Pick a subset of lines in the network that maximizes the performance.

- Intuition: Adding more lines increases the capacity of the network.
- Does the problem correspond to a conventional network flow problem? Min cost network flow or max flow problem.
- Answer: No!
- Key difference: We have more variables and constraints on the edges.

Max capacity = 100 MW

\[ f_{12}^{(1)} = B_{12}^{(1)} (\theta_1 - \theta_2) \]

\[ f_{12}^{(2)} = B_{12}^{(2)} (\theta_1 - \theta_2) \]

Max capacity = 300 MW
OTS Problem

- In plain language: Pick a subset of lines in the network that maximizes the performance.

- Intuition: Adding more lines increases the capacity of the network.
- Does the problem correspond to a conventional network flow problem? Min cost network flow or max flow problem.
- Answer: No!
- Key difference: We have more variables and constraints on the edges.

Max capacity = 100 MW

\[ f_{12}^{(1)} = B_{12}^{(1)}(\theta_1 - \theta_2) \]

Max capacity = 300 MW

\[ f_{12}^{(2)} = B_{12}^{(2)}(\theta_1 - \theta_2) \]
Problem Formulation

- **Decision variables:** \( \mathbf{x} \triangleq [x_1, \ldots, x_{n_s}]^T, \theta \triangleq [\theta_1, \ldots, \theta_{n_b}]^T, \mathbf{f} \triangleq [f_1, \ldots, f_{n_l}], \mathbf{p} \triangleq [p_1, \ldots, p_{n_g}]^T, \)

- **Generation costs:**
  - Power generation costs:
    \[
    \sum_{i=1}^{n_g} a_i \times (p_i)^2 + b_i \times p_i \sum_{i=1}^{n_g} b_i \times p_i
    \]

- **Constraints:**
  - Switch status:
    \( x_i \in \{0, 1\} \)
  - Generator limits:
    \( p_i; \min \leq p_i \leq p_i; \max \)
  - Flow limits for inflexible lines:
    \( -f_{ij}; \max \leq f_{ij} \leq f_{ij}; \max \ \forall (i, j) \in \mathcal{L}\setminus\mathcal{S} \)
  - Flow limits for flexible lines:
    \( -f_{ij}; \max \times x_{ij} \leq f_{ij} \leq f_{ij}; \max \times x_{ij} \ \forall (i, j) \in \mathcal{S} \)
  - Physical constraints for inflexible lines:
    \( f_{ij} = B_{ij}(\theta_i - \theta_j) \ (i, j) \in \mathcal{L}\setminus\mathcal{S} \)
  - Physical constraints for flexible lines:
    \( f_{ij} = B_{ij}(\theta_i - \theta_j)x_{ij} \ (i, j) \in \mathcal{S} \)
  - Conservation of flows:
    \( p_k - d_i = \sum_{j \in \mathcal{N}_i^+(i)} f_{ij} - \sum_{j \in \mathcal{N}_i^-(i)} f_{ji} \)
  - Cardinality constraint:
    \( \sum_{(i,j) \in \mathcal{S}} x_{ij} \geq |\mathcal{L}| - r \)
Linearization

- We can also add time horizon, security constraints, etc.
- OTS is an NP-hard problem.
- Nonlinear and nonconvex constraints:

\[ f_{ij} = B_{ij}(\theta_i - \theta_j)x_{ij} \quad (i, j) \in S \]
Linearization

- We can also add time horizon, security constraints, etc.
- OTS is an NP-hard problem.
- Nonlinear and nonconvex constraints:

\[
 f_{ij} = B_{ij}(\theta_i - \theta_j)x_{ij} \quad (i, j) \in S 
\]

Exact Linearization!

\[
 B_{ij}(\theta_i - \theta_j) - M_{ij}(1 - x_{ij}) \leq f_{ij} \leq B_{ij}(\theta_i - \theta_j) + M_{ij}(1 - x_{ij}) \\
 -f_{ij;max} \times x_{ij} \leq f_{ij} \leq f_{ij;max} \times x_{ij} 
\]
Linearization

- We can also add time horizon, security constraints, etc.
- OTS is an NP-hard problem.
- Nonlinear and nonconvex constraints:

\[
\begin{align*}
    f_{ij} &= B_{ij}(\theta_i - \theta_j)x_{ij}, \quad (i, j) \in S \\
    B_{ij}(\theta_i - \theta_j) - M_{ij}(1 - x_{ij}) &\leq f_{ij} \leq B_{ij}(\theta_i - \theta_j) + M_{ij}(1 - x_{ij}) \\
    -f_{ij,\max} \times x_{ij} &\leq f_{ij} \leq f_{ij,\max} \times x_{ij}
\end{align*}
\]

**Exact Linearization!**

**Question:** How large should \( M_{ij} \) be?

1. Small values for \( M_{ij} \) can lead to tighter relaxations and hence, fewer iterations.
2. Large values for \( M_{ij} \) can lead to numerical issues.

**Definition:** \( M_{ij} \) is called feasible if it results in exact linearization.

**Theorem:** For a flexible line \((i, j)\):
1. There is no efficient algorithm to find the smallest feasible \( M_{ij} \).
2. There is no efficient approximation algorithm to find the smallest feasible \( M_{ij} \).

---

Linearization

- Small value for $M_{ij}$ is highly desirable.
- Trivial upper bounds for feasible $M_{ij}$.
- Can we go from trivial to nontrivial values?
- Common practice: add upper bounds on the absolute value of angles.
- May significantly shrink the feasible region.

Observation: Only a small subset of lines are considered as flexible.

Nontrivial upper bounds can be found if there is a connected sub-network with no switches.
Simulation Results

- We consider the IEEE 118-bus system. This system has 118 nodes and 185 lines.
- The objective is assumed to be linear.
- We fix a randomly generated connected sub-graph of the system with 117 lines.
Simulation Results

- We consider the IEEE 118-bus system. This system has 118 nodes and 185 lines.
- The objective is assumed to be linear.
- We fix a randomly generated connected sub-graph of the system with 117 lines.
- Variables:
  - 68 binary variables corresponding to switches.
  - 118 continuous variables corresponding to angle of each node.
  - 185 continuous variables corresponding to the flow of each line.
  - 54 continuous variables corresponding to the generation of different generators.
  - Lower bound on the number of ON switches: 45
Simulation Results

- We consider the IEEE 118-bus system. This system has 118 nodes and 185 lines.
- The objective is assumed to be linear.
- We fix a randomly generated connected sub-graph of the system with 117 lines.
- Variables:
  - 68 binary variables corresponding to switches.
  - 118 continuous variables corresponding to angle of each node.
  - 185 continuous variables corresponding to the flow of each line.
  - 54 continuous variables corresponding to the generation of different generators.
  - Lower bound on the number of ON switches: 45

- Intel Core i7 quad-core 2.50 GHz CPU and 16GB RAM.
- Serial implementation in MATLAB using the CVX framework and Gurobi solver.
- The optimality threshold is set to 0.01.

<table>
<thead>
<tr>
<th>With designed upper bounds on $M_{ij}$</th>
<th>Without designed upper bounds on $M_{ij}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 min</td>
<td>46 min</td>
</tr>
</tbody>
</table>
Simulation Results

Performance with respect to cardinality lower bound

![Graph showing performance with respect to cardinality lower bound]

Performance with respect to different load factors

![Graph showing performance with respect to different load factors]
Simulation Results

Performance with respect to cardinality lower bound

Performance with respect to different load factors

- If cardinality lower bound is zero, < 1 sec for Gurobi to solve the problem.
- Linear objective.
- If the objective function is quadratic, Gurobi finds the optimal solution after 72 min!
Convex Model

• Existing methods are based on branch and bound, cutting plane, dynamic programming, or line rankings.

• **Goal:** Find a convex model of the problem.
  • Useful for *convex hull pricing*. [Gribik 07]
  • Can be adopted to solve OTS problem for AC systems.
Convex Model

- Existing methods are based on branch and bound, cutting plane, dynamic programming, or line rankings.
- **Goal:** Find a convex model of the problem.
  - Useful for *convex hull pricing*. [Gribik 07]
  - Can be adopted to solve OTS problem for AC systems.

\[
\mathbf{w} \triangleq [\mathbf{x}^\top, \mathbf{p}^\top, \mathbf{f}^\top, \theta^\top]^\top
\]

\[
\min_{\mathbf{w} \in \mathbb{R}^{n_s+n_g+n_l+n_b}} \quad c(\mathbf{w})
\]

subject to \( \mathbf{Mw} \geq \mathbf{m}, \)

\[
\quad w_k(w_k - 1) = 0, \quad k = 1, 2, \ldots, n_s,
\]

\[
\min_{\mathbf{w} \in \mathbb{R}^{n_s+n_g+n_l+n_b}} \quad c_r(\mathbf{w})
\]

subject to \( \mathbf{Mw} \geq \mathbf{m}, \)

\[
\quad W_{kk} - w_k = 0, \quad k = 1, 2, \ldots, n_s,
\]

\[
\mathbf{W} = \mathbf{ww}^\top,
\]
Convex Model

- Existing methods are based on branch and bound, cutting plane, dynamic programming, or line rankings.
- **Goal:** Find a convex model of the problem.
  - Useful for *convex hull pricing*. [Gribik 07]
  - Can be adopted to solve OTS problem for AC systems.

\[ \mathbf{w} \triangleq [\mathbf{x}^\top, \mathbf{p}^\top, \mathbf{f}^\top, \mathbf{\theta}^\top]^\top \]

\[
\begin{align*}
\text{minimize} & \quad c(\mathbf{w}) \\
\text{subject to} & \quad \mathbf{Mw} \geq \mathbf{m}, \\
& \quad w_k(w_k - 1) = 0, \quad k = 1, 2, \ldots, n_s,
\end{align*}
\]

\[
\begin{align*}
\text{minimize} & \quad c_r(\mathbf{w}) \\
\text{subject to} & \quad \mathbf{Mw} \geq \mathbf{m}, \\
& \quad W_{kk} - w_k = 0, \quad k = 1, 2, \ldots, n_s, \\
& \quad \mathbf{W} \succeq \mathbf{ww}^\top
\end{align*}
\]
Convex Model

- Existing methods are based on branch and bound, cutting plane, dynamic programming, or line rankings.
- **Goal:** Find a convex model of the problem.
  - Useful for *convex hull pricing*. [Gribik 07]
  - Can be adopted to solve OTS problem for AC systems.

\[
\mathbf{w} \triangleq [\mathbf{x}^\top, \mathbf{p}^\top, \mathbf{f}^\top, \mathbf{\theta}^\top]^\top
\]

\[
\begin{align*}
\text{minimize} & \quad c(\mathbf{w}) \\
\text{subject to} & \quad \mathbf{Mw} \geq \mathbf{m}, \\
& \quad w_k(w_k - 1) = 0, \quad k = 1, 2, \ldots, n_s,
\end{align*}
\]

\[
\begin{align*}
\text{minimize} & \quad c_R(\mathbf{w}) \\
\text{subject to} & \quad \mathbf{Mw} \geq \mathbf{m}, \\
& \quad \mathbf{W}_{kk} - w_k = 0, \quad k = 1, 2, \ldots, n_s, \\
& \quad \mathbf{W} \succeq \mathbf{ww}^\top
\end{align*}
\]

- If the optimal \(\mathbf{W}\) has rank-1, the relaxation is exact.

**Bad news:** SDP relaxation is almost as bad as QP.

**Theorem:** For generic load profiles, the SDP will work with probability 0.

- The optimality gap is 3%-80% in IEEE test cases.
- Need to strengthen the formulation by adding valid inequalities.
Valid Inequalities
Valid Inequalities
Valid Inequalities

\[ u^T w - z_1 \geq 0 \quad v^T w - z_2 \geq 0 \]

Nonlinear constraint
Valid Inequalities

\[ u^T w - z_1 \geq 0 \]
\[ v^T w - z_2 \geq 0 \]

Nonlinear constraint

\[ u^T w w^T v - (v^T z_1 + u^T z_2)w + z_1 z_2 \geq 0 \]

Linearize!

\[ u^T Wv - (v^T z_1 + u^T z_2)w + z_1 z_2 \geq 0 \]
Valid Inequalities

\[ \mathbf{u}^T \mathbf{w} - z_1 \geq 0 \quad \mathbf{v}^T \mathbf{w} - z_2 \geq 0 \]

Nonlinear constraint

\[ \mathbf{u}^T \mathbf{w} - (\mathbf{v}^T z_1 + \mathbf{u}^T z_2) \mathbf{w} + z_1 z_2 \geq 0 \]

Linearize!

Based on Sherali-Adams’ RLT relaxation.

Theorem: This relaxation is exact for large loads and/or small line ratings.

\[
\begin{align*}
\text{minimize} & \quad c_T(\mathbf{w}, \mathbf{W}) \\
\text{subject to} & \quad \mathbf{Mw} \geq \mathbf{m}, \\
& \quad \mathbf{MWM}^T - \mathbf{mw}^T \mathbf{M}^T - \mathbf{Mm}^T + \mathbf{mm}^T \geq 0, \\
& \quad W_{kk} - w_k = 0, \quad k = 1, 2, \ldots, n_s, \\
\mathbf{W} & \succeq \mathbf{ww}^T.
\end{align*}
\]
Simulation Results

- IEEE 14-bus with 12 load scenarios and 5 switches:

- IEEE 30-bus with 9 load scenarios and 7 switches:
Simulation Results

- IEEE 14-bus with 12 load scenarios and 5 switches:

- IEEE 30-bus with 9 load scenarios and 7 switches:
Simulation Results

- IEEE 57-bus with 8 load scenarios and 10 switches:

  - Optimal cost vs. Load factor
  - Optimality gap vs. Load factor

- IEEE 118-bus with 10 load scenarios and 20 switches and lower bound equal to 10:

  - Optimal cost vs. Load factor
  - Optimality gap vs. Load factor
Boosting the Speed: Heuristic Rounding

- The SSDP is extremely time-consuming to solve.
- 10 min to solve IEEE 118-bus system.
- It finds the optimal objective, but not the feasible binary variables.
- We can resort to RLT with heuristic rounding.

- We consider the IEEE 118-bus system with same settings.
- Number of variables in RLT relaxation: 90,525
- Number of constraints in RLT relaxation: 376,333

- Implementing RLT relaxation in Gurobi was extremely inefficient.
  - Instead, we used MOSEK solver with CVX framework in MATLAB.

- Only 2 levels of RLT were needed in our simulations.
- Number of rounded binary variables after first round of RLT: 50/65
Numerical Results

Performance with respect to cardinality lower bound

Performance with respect to different load factors
Extension to Unit Commitment

- Binary variables for generators (ON/OFF).
- Longer time-horizon.
- Ramping constraints, minimum up- and down-time constraints.
Extension to Unit Commitment

- Binary variables for generators (ON/OFF).
- Longer time-horizon.
- Ramping constraints, minimum up- and down-time constraints.

IEEE 118-bus with 54 generators:

- Cost vs. Load

IEEE 14-bus over 24 hours:

- Total load vs. Time
- SDP relaxation: 162600
- Reduced-strengthened SDP relaxation: 205838
Conclusions

- OTS problem with linear and quadratic objectives.
- Finding a good MILP formulation of OTS problem may be hard.
- The MILP formulation can be tightened if some part of the topology is fixed.
- Convex model for OTS problem with quadratic objective.
- Strong valid inequalities.
- Rounding heuristics in order to boost the running time.
- Extension to Unit Commitment problem.
Thank you!