A Synergistic Combination of Surrogate Lagrangian Relaxation and Branch-and-Cut for MIP Problems in Power Systems

*University of Connecticut
**Southern California Edison

February, 2014

This work was supported in part by grants from Southern California Edison and by the National Science Foundation under grant ECCS-1028870
Motivation

- Unit commitment and economic dispatch (UCED) is to commit units and decide their generation levels to satisfy demand and reserve requirements.
- It is modeled as MIP and is computationally intensive especially when:
  - considering combined cycle units
  - considering uncertainties introduced by renewables.
How to formulate the problem?

\[
\min \{ p_i(t) \} \quad \text{with} \quad J = \sum_{t=1}^{T} \sum_{i=1}^{I} \left\{ C_i(p_i(t), t) + S_i(t) \right\}
\]

s.t.:

- System demand \( \sum_{i=1}^{I} p_i(t) = P_d(t) \) and max/min power level constraints
- Reserve requirements and transmission constraints are ignored for simplicity

In view of separability, the problem can be decomposed by Lagrangian relaxation.

Since \( C_i \) can be converted to a linear function, start-up cost is linear and constraints are linear, the problem can also be solved by branch-and-cut.
Solution by Lagrangian Relaxation

- Lagrangian relaxation

\[ L(\lambda, p) \equiv \sum_{t=1}^{T} \sum_{i=1}^{I} \left\{ C_i(p_i(t), t) + S_i(t) \right\} \]

\[ + \sum_{t=1}^{T} \left\{ \lambda(t) \left( P_d(t) - \sum_{i=1}^{I} p_i(t) \right) \right\} \]

\[ \Rightarrow \]

\[ L = \sum_{i=1}^{I} \left\{ \sum_{t=1}^{T} \left( C_i(p_i(t), t) + S_i(t) - \lambda(t)p_i(t) \right) \right\} \]

\[ + \sum_{t=1}^{T} \lambda(t) P_d(t) \]

- Lagrangian relaxation decomposes \( L \) into \( I \) subproblems, and updates \( \lambda \) based on levels of constraints violation

\[ \lambda^{k+1}(t) = \lambda^k(t) + c^k g(p^k), \quad c^k < \frac{q^* - L(\lambda^k, p^k)}{||g(p^k)||^2} \]

- Standard subgradient methods require \( L \) to be fully optimized with respect to all subproblems
  - \( L \) is difficult to fully optimize
  - \( \lambda \) can suffer from zigzagging
  - Convergence proof requires the optimal dual value \( q^* \)

\[ g(p^k) \]

\[ (p^k) \]

\[ (p^k) \]

\[ (p^k) \]

\[ (p^k) \]
Overview of the Presentation

- Motivation: Unit commitment and economic dispatch
- Surrogate Lagrangian relaxation (SLR)
  - Lagrangian relaxation with surrogate subgradient methods
  - Surrogate Lagrangian relaxation
- Synergistic combination of SLR and Branch-and-cut
  - A brief introduction of branch-and-cut (B&C)
  - Synergistic combination of SLR and Branch-and-cut
- Numerical examples
  - A small illustrative example
  - Large-scale generalized assignment problems
  - UCED with combined cycle units
- Conclusion
The Surrogate Subgradient Method (1999)

- The surrogate subgradient method allows approximate optimization of $L$ s.t. the surrogate optimality condition guaranteeing an acute angle with the direction toward $\lambda^*$:

$$L(\lambda^{k+1}, x^{k+1}) < L(\lambda^{k+1}, x^k)$$

- For UCED, it is sufficient to solve one subproblem:

$$\min \left\{ \sum_{t=1}^{T} \left( C_i(p_i(t), t) + S_i(t) - \lambda(t) p_i(t) \right) \right\}$$

- Multipliers are updated:

$$\lambda^{k+1}(t) = \lambda^k(t) + c^k \tilde{g}_{\lambda}^k$$

$$\tilde{g}_{\lambda}^k = (P_d(t) - \sum_{i=1}^{I} p_i^k(t))$$

- Surrogate directions are smooth because now we change only one $p_i(t)$

- Critically needs $q^*$

Directions are smoother as compared to those of the subgradient methods
Surrogate Lagrangian Relaxation (SLR)

- **Main Contribution:** Develop a new method, prove convergence, and guarantee practical implementability
  - Without fully optimizing the relaxed problem (s.t. the surrogate optimality condition)
  - Without requiring $q^*$

- **Main Idea 1:** Decrease distances between multipliers at consecutive iterations ($||\lambda^{k+1} - \lambda^k||$ decreases)

- $||\lambda^{k+1} - \lambda^k||$ decreases $\Rightarrow$ fixed-point mapping $\Rightarrow \lambda^k \rightarrow \bar{\lambda}$

![Graph showing convergence of multipliers](image)
Surrogate Lagrangian Relaxation

\[
\lambda^{k+1} = \lambda^k + c^k \tilde{g}(x^k)
\]

- Parameters \( \alpha_k \) should satisfy

\[
\begin{align*}
\left\| \lambda^{k+1} - \lambda^k \right\| &= \alpha_k \left\| \lambda^k - \lambda^{k-1} \right\|, \quad 0 < \alpha_k < 1, \\
\left\| c^k \tilde{g}(x^k) \right\| &= \alpha_k \left\| c^{k-1} \tilde{g}(x^{k-1}) \right\|
\end{align*}
\]

\[
\Rightarrow \quad c^k = \alpha_k \frac{c^{k-1} \left\| \tilde{g}(x^{k-1}) \right\|}{\left\| \tilde{g}(x^k) \right\|}
\]

- If \( \alpha_k \) are small, \( c^k \to 0 \) too fast \( \Rightarrow \) premature convergence

- Parameters \( \alpha_k \) should satisfy \( c^k \sim \prod_{i=1}^{k} \alpha_i \to 0 \) \( \) (2)

- Main Idea 2:
  - To avoid premature convergence, \( c_k \) should not decrease too fast
  - This can be achieved by keeping \( \alpha_k \) sufficiently close to 1

\[
\lim_{k \to \infty} \frac{1 - \alpha_k}{c^k} = 0
\]
Main Theorem

- Multipliers converge to the optimum $\lambda^*$ without requiring $q^*$ provided $\alpha_k$ satisfy:

1) $c^k \sim \prod_{i=1}^{k} \alpha_i \to 0$ (Main idea 1)
2) $\lim_{k \to \infty} \frac{1 - \alpha_k}{c^k} = 0$ (Main idea 2)

Without requiring $q^*$!

- One possible example of $\alpha_k$ that satisfies conditions 1) and 2): $\alpha_k = 1 - \frac{1}{M \cdot k^p}, 0 < p < 1, M > 1, k = 1, 2, ...$

- At convergence, the surrogate dual value approaches the (optimal) dual value $q^* \sim$ valid lower bound on the feasible cost
  - Lower bound is guaranteed before convergence by fully optimizing the relaxed problem to obtain a dual value
Schematic Flow chart of SLR

Original problem

Relax system demand by using Lagrange multipliers

Update multipliers

Solve one or a few subproblems, until the surrogate optimization condition is satisfied

Meet Stopping criteria?

No

Construct a feasible solution every few iterations

Yes

Construct a feasible solution, and compare with the best one

How to solve subproblems?

Surrogate optimization
Difficulties of Standard Branch-and-Cut

- Branch-and-cut (B&C) can suffer from slow convergence because
  - Facet-defining cuts and even valid inequalities that cut areas outside the convex hull are problem-dependent and are frequently difficult to obtain
  - When facet-defining cuts are not available, a large number of branching operations will be performed
  - No “local” concept ⇒ Constraints associated with one subproblem are treated as global constraints and affect the entire problem
Synergistic Combination with Branch-and-cut

- To overcome difficulties, SLR relaxation and B&C are synergistically combined to simultaneously exploit problem separability and linearity:
  - Relax coupling constraints (e.g., system demand)
  - Solve each subproblem by using branch-and-cut with warm start
    - The complexity of each subproblem is much lower than the complexity of the original problem
  - Updating multiplies by using SLR – convergence without requiring $q^*$

- Why is the new method effective?
  - Cuts for subproblems are more effective as compared to cuts for the original problem
  - Feasible solutions can be effectively obtained
  - The overall algorithm is efficient
Synergistic Combination with Branch-and-cut

- Why cuts for subproblems are more effective?
  - An example of important cuts is Gomory cuts
    - Constraints are linearly combined into one constraint
    - Cuts are then generated by retaining fractional parts of the coefficients (Gomory’s fractional cut)
  - Cuts for subproblems aggregate much fewer constraints → cut larger regions as compared to regions obtained by original Gomory cuts
  - Other cuts using aggregation follow similar logic
  - Cuts that require no aggregation (e.g., clique and cover cuts) are as efficient for solving subproblems

- Can feasible solutions be efficiently obtained?
  - Linearity of coupling constraints can be exploited to obtain feasible solutions
Example Illustrating the Combination of SLR and Branch-and-cut

- Consider a toy problem:
  
  \[
  \text{min}(79x_1 + 70x_2 + 108x_3 + 41x_4)
  \]
  
  \[
  \text{s.t. } 35x_1 + 51x_2 \leq 64 \quad (1), \quad 3x_3 + 65x_4 \leq 64 \quad (2)
  \]
  
  \[
  x_1 + x_3 = 1, \quad x_2 + x_4 = 1
  \]
  
  \[
  x_j \in \{0,1\}, \quad j = 1, 2, 3, 4
  \]

- Gomory cut: \[7x_1 + 10x_2 + 13x_4 \leq 26\]

- Consider subproblems:
  
  \[
  \text{min}(79x_1 + 70x_2 - \lambda_1 x_1 - \lambda_2 x_2),
  \]
  
  \[
  \text{s.t. } 35x_1 + 51x_2 \leq 64, \quad x_i \in \{0,1\}, \quad i = 1, 2
  \]

  \[
  \text{min}(108x_3 + 41x_4 - \lambda_1 x_3 - \lambda_2 x_4)
  \]
  
  \[
  \text{s.t. } 3x_3 + 65x_4 \leq 64, \quad x_i \in \{0,1\}, \quad i = 3, 4
  \]

- Gomory cuts:
  
  \[
  7x_1 + 10x_2 \leq 12
  \]
  
  \[
  13x_4 \leq 12
  \]
Flow-Chart of the Synergistic Approach

Relax coupling constraints to exploit separability (e.g., separate into subproblems)

Use branch-and-cut + warm start to solve each subproblem

Guarantee convergence without requiring $q^*$ by updating stepsizes using Main Theorem:

$$c^k = \alpha_k \frac{c^{k-1} \| \tilde{g}(x^{k-1}) \|}{\| \tilde{g}(x^k) \|}$$

Update multipliers by using (1) and smooth surrogate subgradients

Are stopping criteria satisfied?

No

Cuts generated for subproblem cut off large areas of sub-convex hull

Zigzagging is alleviated thereby reducing the number of iterations required for convergence

Yes

Search for feasible solutions

Stop

Convergence without requiring $q^*$ enables the synergistic combination
Consider the Generalized Assignment Problem:

\[
\min \sum_{i=1}^{I} \sum_{j=1}^{J} c_{i,j} x_{i,j} \quad \text{(Cost of assigning } I \text{ jobs to } J \text{ machines)}
\]

\[
s.t. \sum_{i=1}^{I} a_{i,j} x_{i,j} \leq b_{j}, j = 1, \ldots, J \quad \text{(1) (Time required by the jobs does not exceed the machine’s time available)}
\]

\[
\sum_{j=1}^{J} x_{i,j} = 1, i = 1, \ldots, I \quad \text{(2) (Each job is to be performed on one and one machine only)}
\]

Constraints (2) can be viewed as constraints coupling “machine subproblems”
## Results on Generalized Assignment Problems

<table>
<thead>
<tr>
<th>Problem Size</th>
<th>Synergistic Combination of SLR and B&amp;C</th>
<th>Branch-and-cut</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LB</td>
<td>Feasible Cost</td>
</tr>
<tr>
<td>15 machines, 900 jobs</td>
<td>55403</td>
<td>55411</td>
</tr>
<tr>
<td>40 machines, 400 jobs</td>
<td>24348</td>
<td>24394</td>
</tr>
<tr>
<td>20 machines, 1600 jobs</td>
<td>97823</td>
<td>97834</td>
</tr>
</tbody>
</table>
Combined cycle (CC) units are efficient because

- Heat from gas turbines (GT) is not wasted but is used to for steam turbines (ST)

However, UCED problem with CC units is difficult

- ST cannot be turned on if there is not enough heat from GT
  ⇒ Complicated state transitions causing major challenges
Multi-Stage Combined Cycle Units

- **Difficulty:**
  - Constraints modeling transitions between configurations of generators are logical and complex
  - Complex transitions in one such unit affect the entire problem
  - Corresponding convex hull is difficult to obtain

- **By using our new method:**
  - Transitions of a combined cycle unit are handled locally and no longer affect the entire problem
  - Certain cuts generated for subproblems cut off large areas outside the sub-convex hull
  - Branch-and-cut efficiently optimizes subproblems
  - SLR efficiently coordinate subproblem solutions
To demonstrate the efficiency of surrogate Lagrangian relaxation, a problem with 10 CC plants and 300 conventional units is considered.

<table>
<thead>
<tr>
<th>Method</th>
<th>Feasible Cost</th>
<th>Lower Bound</th>
<th>Gap (%)</th>
<th>CPU Time (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Branch-and-cut</td>
<td>50,260,500</td>
<td>45,305,200</td>
<td>9.859</td>
<td>30</td>
</tr>
<tr>
<td>Our new method</td>
<td>49,894,806</td>
<td>49,879,027</td>
<td>0.032</td>
<td>5</td>
</tr>
</tbody>
</table>

Branch-and-cut

Surrogate Lagrangian relaxation

![Graph showing feasible cost and CPU time for Branch-and-cut method and Surrogate Lagrangian relaxation method]
Conclusion

- Major theoretical result: Within the surrogate Lagrangian relaxation framework, multipliers converge to the optimum without requiring $q^*$

- SLR has been synergistically combined with B&C to solve mixed-integer programming problems efficiently
  - Subproblem constraints no longer affect the entire problem
  - Gomory cuts generated for subproblems cut off large areas outside the sub-convex hull

- Numerical results demonstrate that the innovative approach is powerful and efficient for solving mixed-integer programming problems

- Broad Impact: The novel methodology opens new directions to efficiently solve mixed-integer programming problems such as Stochastic Unit Commitment and beyond
Related Publications


