Advanced Methods for Security Constrained Financial Transmission Rights (FTR)

Stephen Elbert, Steve.Elbert@PNNL.gov
Karan Kalsi, Kurt Glaesemann, Mark Rice, Maria Vlachopoulou, Ning Zhou
Pacific Northwest National Laboratory, Richland, WA

FERC Staff Technical Conference on
Increasing Real-Time and Day-Ahead Market Efficiency through Improved Software
(Docket No. AD10-12-003)

June 25-27, 2012
Washington, DC
Financial Transmission Rights (FTR) improve power market operation efficiency by providing financial tool to hedge price risk associated with congestion.

- Mitigate incentives for inefficient transmission investment.
- FTR auction is formulated as a linear programming optimization problem.
- FTR calculations are computationally expensive because:
  - Large number of security constraints (*N-1 contingency analysis*).
  - Many FTR variables (*obligatory and optional FTR bids*).
  - Multiple time periods (*security constraints coupled* & no. of constraints increase *exponentially* with no. of categories).

FTR computation must be finished in time to improve market efficiency.
Objectives

- Develop innovative mathematical reformulation of the FTR problem
- Compare multiple solvers for FTR computations
- Developed approaches will be able to
  - Support N-1 Simultaneous Feasibility Test (SFT) e.g. DC contingency analysis
  - Support both optional and obligatory FTR bids
  - Support multi-period FTR calculation (e.g. winter, summer and annual)
- Algorithms designed to solve FTR problem should be parallelizable to support large-scale implementation in a cloud environment
Problem Formulation

- **Power flow constraints**
  - $B$ is (singular) admittance matrix
  - $\theta_i$ are the bus voltage angles
  - $A$ is FTR location matrix

- **Thermal constraints**
  - $C$ converts voltage angles to line flows
  - $L_i$ are transmission line limits

- **Bid-in constraints**

- **Combine**

- **A dimension is**
  (constraints x bids)
Standard FTR Solvers

- **CPLEX (industry standard)**
  - **Primal simplex**; most basic LP solver method
    - Updates tableau containing objective function and constraint information at every iteration
    - Consistently slower on FTR than dual simplex
  - **Dual simplex**; fastest of the CPLEX methods
    - Similar to primal simplex method, but uses dual formulation of the LP to improve convergence time of optimization
    - Core computation is a linear solve; scales as cube of size
  - **Barrier**; an interior point method (best for large sparse problems)
    - A primal-dual logarithmic barrier algorithm that generates a sequence of strictly positive primal and dual solutions
    - Fewest iterations but each is more computationally intense
PNNL FTR solver – Parallel Adaptive Non-linear Dynamical System (NDS)

- Transform LP into coupled set of non-linear dynamical equations
- Dynamical system converges to stable states which are solutions of primal and dual LP problems respectively

**Primal**

maximize $c^T x$

subject to $Ax \leq b$ and $x \geq 0$

**Dual**

minimize $b^T y$

subject to $A^T y \geq c$ and $y \geq 0$

**Non-linear Dynamical System**

$$\frac{dx}{dt} = k_1 \left( c - A^T \left( y + k \frac{dy}{dt} \right) \right)$$

$$\frac{dy}{dt} = k_2 \left( -b + A \left( x + k \frac{dx}{dt} \right) \right)$$

$$k_1 = \frac{k}{i} \quad i = 1, 2, \ldots, M \quad k_2 = \frac{1}{k_1}$$

- Kernel is a pair of easily parallelized matrix-vector operations: scale as square of problem size (constraints x variables)
Implementation notes

- Obligatory and Optional bids sorted into separate blocks
  - Obligatory bids may use dense matrix arithmetic
  - Optional bids use sparse matrix arithmetic (up to 50% sparse)
- $A$ matrix segments communicated once at beginning
- $A$ and $A^T$ stored separately to maximize unit stride access
- Global sums for product vectors and distribution of $x$ and $y$ vectors are the only communication after initialization
- Symmetry of Obligatory bids reduces computation by two
Validation test cases

<table>
<thead>
<tr>
<th>Cases</th>
<th>Constraints</th>
<th>Bids</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a. WECC 230 single period</td>
<td>5,362</td>
<td>5,790</td>
</tr>
<tr>
<td>2a. WECC 230 single period &amp; many bids</td>
<td>5,362</td>
<td>100,000</td>
</tr>
<tr>
<td>3a. WECC 230 multi-period</td>
<td>10,724</td>
<td>17,370</td>
</tr>
<tr>
<td>4a. WECC 230 multi-period &amp; many bids</td>
<td>10,724</td>
<td>300,000</td>
</tr>
<tr>
<td>1b. WECC 100 single period</td>
<td>19,094</td>
<td>22,455</td>
</tr>
<tr>
<td>2b. WECC 100 single period &amp; many bids</td>
<td>19,094</td>
<td>100,000</td>
</tr>
<tr>
<td>3b. WECC 100 multi-period</td>
<td>38,188</td>
<td>67,365</td>
</tr>
<tr>
<td>4b. WECC 100 multi-period &amp; many bids</td>
<td>38,188</td>
<td>300,000</td>
</tr>
</tbody>
</table>

- WECC 230 & 100 model power flow on transmission lines operating at min of
  - 230 kV (1,930 buses and 2,681 branches)
  - 100 kV (7,485 buses and 9,547 branches)

- Multi-period problems have independent periods plus a coupling block

<table>
<thead>
<tr>
<th>FTR bids</th>
<th>PF (winter)</th>
<th>PF (summer)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bids (winter)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bids (summer)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bids (annual,)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Results – single period cases

- **WECC 230**
  - Primal simplex takes 20 min, dual simplex and serial NDS (1 core) takes 2 minutes
  - Parallel NDS is six times faster than dual simplex (CPLEX)

- **WECC 100**
  - At 4 hours, dual simplex not yet converged
  - Parallel NDS 46X faster than dual simplex
Results – single period & many bids (100,000)

- NDS 256-core and CPLEX comparable results (cross at 53 seconds)

- NDS 100 times faster when crossing CPLEX curve
- NDS scaling well
Results – two period (summer/winter) cases

- WECC 230
  - Serial NDS is faster than CPLEX
  - NDS 128-core is 17 times faster

- WECC 100
  - CPLEX no longer practical—time is divided by 10 and not converged
  - NDS 256-core is 185 times faster
  - 1.7 billion non-zero matrix elem.

The bigger the problem, The faster the relative performance
Results – two periods & many bids (300,000)

WECC 230

- Additional cores and code improvements → solution in under 4 hours
- 15.3 billion non-zero matrix elem.
Real-world data 1

Cplex time per iteration slows by 85x from beginning to end due to backfill
Real World Data 2

CPLEX time per iteration slows by 269x from beginning to end due to backfill
Summary

- Developed novel non-linear dynamical system based FTR solver
- Easily parallelized to solve large linear programming (LP) problems for FTR application within few hours (cloud compatible)
- Parallel NDS more computationally efficient than CPLEX for LP
  - Computational kernel of CPLEX is linear solver that scales as cube of problem size
  - NDS kernel is matrix-vector multiplication that scales as square
  - NDS avoids backfill (filling in zeros) of coupled blocks
  - Maintains numerical stability through using only original matrix
  - Uses dense algorithm for obligatory bids, sparse (50%) for optional bids
  - Half the arithmetic for obligatory bids (two inner products differ only in sign)
  - Data loaded efficiently in parallel
- Further enhancements
  - Further improve parallelization (asynchronous communication, sparse ops)
  - Refine adaptive time stepping and explore ode time stepping for faster convergence
Future

- Develop quadratic programming capability
  - Improved FTR constraints
- Explore other application needing LP and/or QP capability
  - Transmission planning
  - Locational Marginal Pricing (LMP)
  - Optimal Power Flow (OPF)
- Explore using method with discrete problems
  - Mixed Integer Programming (MIP)
  - Resource Scheduling and Commitment (RSC) (aka Unit Commitment)
  - Stochastic RSC
Acknowledgements

For providing Alstom data sets
- Xing Wang
- David Sun

Support

This work was supported by the U.S. Department of Energy’s Office of Electricity Delivery and Energy Reliability, Advanced Grid Analytics program at the Pacific Northwest National Laboratory.
Pacific Northwest National Laboratory is operated for the U.S. Department of Energy by Battelle Memorial Institute under Contract DE-AC05-76RL01830.
References
