Scalable real-time distributed DER control

Steven Low (PI), Caltech

Team: Robert Sherick (SCE),
    Paul DeMartini (Newport/Resnick)

Jan 14, 2015, New Orleans
Project Objectives

› AC OPF Theory
  – Math foundation for convex relaxation of OPF

› DER optimization algorithms
  – Balanced mesh networks
  – Unbalanced radial networks
  – Centralized and distributed algorithms

› Modeling
  – SCE distribution systems, feeders & secondary circuits

› Implementation & demo

› Tech-2-market
Project Objectives

- **Uniqueness**
  - Guaranteed optimality and convergence
  - New framework for algorithm design

- **Challenges**
  - Mesh networks, unbalanced networks
  - Distributed algorithms with guarantees (stab, perf)
  - Numerical stability

- **Performance metrics**
  - Distributed algorithm for unbalanced radial network
    - size (demo’ed): 2,000 buses
    - time: 5 mins
    - optimality gap: 5%

Algorithms scalable to 10K nodes and beyond
2014 accomplishments

- Convex relaxation of unbalanced network
- Distributed relaxation algorithms
- Distribution system modeling
- Implementation & demo
2014 accomplishments

- Convex relaxation for unbalanced network
  - Theory
    - Chordal relaxation (exploits sparsity)
    - Branch flow model, bus injection model
  - Algorithms
    - Extension of semidefinite algorithms to unbalanced radial networks
    - Centralized and decentralized algorithms
  - Simulations
    - Centralized alg: SCE 2,000-bus, 3 mins, 0% gap
    - Distributed alg: IEEE 123-bus, 3 mins, 0% gap
## Simulation results (Aug 2014 review)

<table>
<thead>
<tr>
<th></th>
<th>Simulation performance</th>
<th>Target performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>#instances (4 week)</td>
<td>8,064</td>
<td>8,064</td>
</tr>
<tr>
<td>#instances solved (convergence)</td>
<td>100%</td>
<td>80%</td>
</tr>
<tr>
<td>suboptimality gap (exactness)</td>
<td>0%</td>
<td>5%</td>
</tr>
<tr>
<td>solution time (per instance)</td>
<td>2 min</td>
<td>3 mins</td>
</tr>
</tbody>
</table>

- Uses generalized BFM chordal relaxation using Rossi (~2000-bus) feeder
- Much more numerically stable than BIM
- Ran on 16 servers
- Exactness (ev ratios): 16.6M ratios (= 2064 lines/instance x 8064 instances)
2014 accomplishments

- **Distributed relaxation algorithms**
  - Theory and algorithms: build on
    - Convex relaxation of OPF
    - Branch flow model
    - ADMM
  - Simulations
    - Balanced radial network: 2,000 buses, 3 mins, 0% optimality gap
    - Unbalanced radial network: ~100 buses, <4 mins, 0% optimality gap
## Computation Time

### Table

<table>
<thead>
<tr>
<th>Network size N</th>
<th>Diameters D</th>
<th>Iterations</th>
<th>Total Time</th>
<th>Avg time T</th>
</tr>
</thead>
<tbody>
<tr>
<td>2,065 buses</td>
<td>64 links</td>
<td>1,114</td>
<td>1,153 sec</td>
<td>0.56 sec</td>
</tr>
<tr>
<td>1,313</td>
<td>54</td>
<td>671</td>
<td>471</td>
<td>0.36</td>
</tr>
<tr>
<td>792</td>
<td>53</td>
<td>524</td>
<td>226</td>
<td>0.29</td>
</tr>
<tr>
<td>363</td>
<td>36</td>
<td>289</td>
<td>66</td>
<td>0.18</td>
</tr>
<tr>
<td>108</td>
<td>16</td>
<td>267</td>
<td>16</td>
<td>0.14</td>
</tr>
</tbody>
</table>

- Suboptimality gap: 0%
- Compute time in distributed execution

### Scalability Trend

Regression: \( T = 9.8 \times 10^{-7} N + 8.6 \times 10^{-3} D \)
Comparison: ADMM-based algs

Huge speedup

- Recent distributed OPF algorithms (inc ours) are ADMM-based
- All these algorithms solve the ADMM subproblems in each update iteratively
- Ours solves them in closed form

<table>
<thead>
<tr>
<th>per-bus computation time</th>
<th>x-update</th>
<th>z-update</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our algorithm</td>
<td>1.7 x 10^{-4} sec</td>
<td>5.1 x 10^{-4} sec</td>
</tr>
<tr>
<td>CVX</td>
<td>2 x 10^{-1} sec</td>
<td>3 x 10^{-1} sec</td>
</tr>
<tr>
<td>speedup</td>
<td>1,176x</td>
<td>588x</td>
</tr>
</tbody>
</table>

per-bus computation time: time to solve 1 sample ADMM iteration for Rossi circuit with 2,065 buses, divided by 2,065, for both algorithms
2014 accomplishments

- Distribution system modeling
  - SCE systems
    - 6 feeders (4KV, 12KV)
    - ~15,000 buses
    - <10% error compared with substation measurements
2014 accomplishments

Implementation & demo

- SCE Rossi feeder
  - ~2,000 buses
  - DER: inverters, HVAC, EV, pool pumps
  - Unbalanced multiphase
  - 4-week simulations

peak load reduction: 8%
energy cost reduction: 4%

baseline  
optimized
Remaining tasks

- Distributed OPF for unbalanced network
  - SCE Rossi feeder: ~2,000 buses
  - time: 5 mins
  - optimality gap: 5%

![Graph showing network size vs. time](image-url)
Tech-to-market

- **T2M objectives**
  - Validation: market, technology

- **Key activities**
  - IAB, Berkeley Haas C2M project
  - Prototype
  - Pilots, VC and strategic investments

![Diagram](Image)
Tech-to-market

Price in $/mmBtu (Real)

- Henry Hub
- US Bituminous Coal
- Brent
- LNG
- Solar

Disruption in the $15B to $50B range

Smart Grid TAM...

$15B

Utilities

$50B

$220M (2014)

Majority M2M connections = Transportation + Utilities (2014-2024)

> mins

secs

EAN’s $IoT$ Gateway

Response Decision + Update Model

Line Sensor Head End

Update Model

Line Sensor Head End

Response Decision

Cellular Network

Battery Storage

Line Sensor

Transformer

Meter

Solar PV

> mins

secs
Post ARPA-E goals

› New R&D
  – Builds on existing results for fast timescale dynamic control & optimization
  – Scalable distributed real-time control with guaranteed stability and performance

› Implementation & pilots
  – Commercial grade software for DERMS
  – Pilots with industry

› Tech-to-market
Conclusions

‣ Most important contributions
  – Math foundation for convex relaxation of OPF
  – Relaxation algorithms: unbalanced radial, distributed
  – Detailed feeder models
  – Implementation & demo

‣ Challenges
  – Numerical instability, scalability
  – Data for realistic and detailed models
  – T2M: prototype, pilots
Backup slides
GridLab-D simulator

\[ \begin{align*}
\dot{x} &= f(x(t), u(t)) \\
y &= g(x(t), u(t))
\end{align*} \]

\[ \hat{x} = \hat{f}(\hat{x}(t), u(t); e(t)) \]
\[ \hat{y} = \hat{g}(\hat{x}(t), u(t)) \]
\[ e(t) = y(t) - \hat{y}(t) \]

\[ u(t) = \text{argmin}_u \text{OPF}(u; \hat{x}(t)) \]

**DERMS applications**

- **control**
- **estimate & learn**

**estimate state in fast timescale**