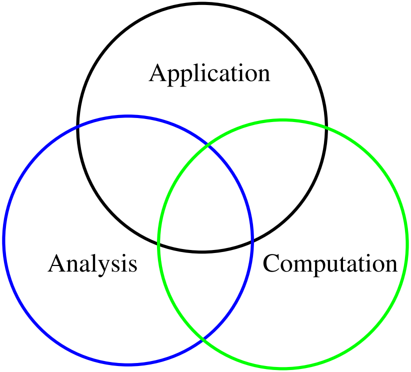
Fast Fingerprints for Power System Events

David Bindel

19 Oct 2016

The Computational Science & Engineering Picture

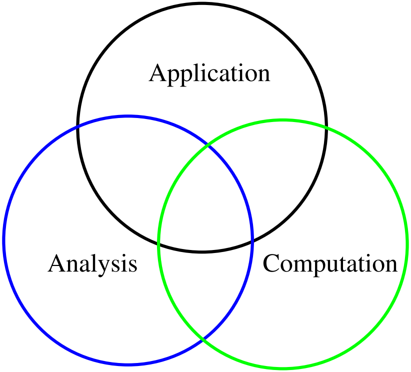


- MEMS
- Smart grids
- Networks

- Linear algebra
- Approximation theory
- Symmetry + structure

- HPC / cloud
- Simulators
- Solvers

The Computational Science & Engineering Picture

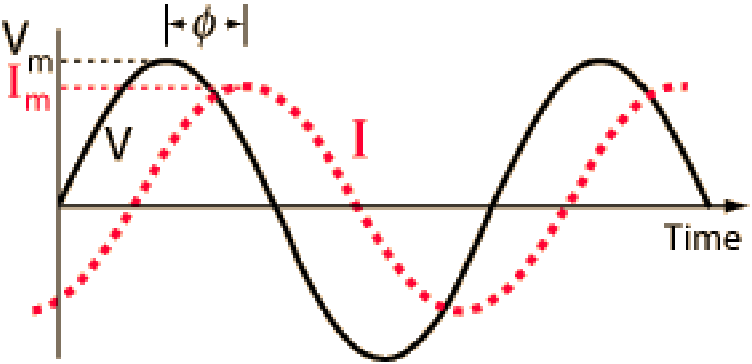


- MEMS
- **Smart grids**
- Networks

- **Linear algebra**
- Approximation theory
- Symmetry + structure

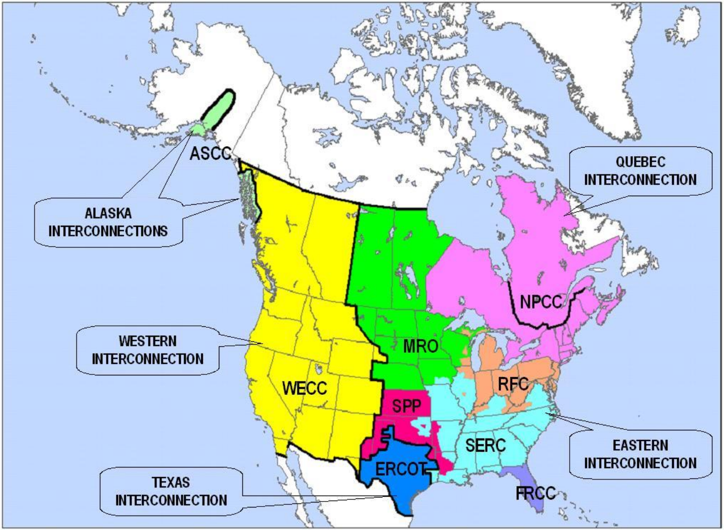
- HPC / cloud
- Simulators
- Solvers

Reminder: AC Power



Voltages	$V(t) = \sqrt{2} V_{\text{avg}} \cos(\omega t)$
Currents	$I(t) = \sqrt{2} I_{\text{avg}} \cos(\omega t + \phi)$
Power	$P_{\text{avg}} = V_{\text{avg}} I_{\text{avg}} \cos \phi$

Biggest Machine(s) in the World



SCADA: Supervisory Control and Data Acquisition

- Non-synchronized measurements every 2–10 seconds
- Report digital status and power flows
- Complete observability in transmission grid
- Voltages/currents inferred from power flows (state estimation)
- *Topology estimation* co-evolved with state estimation

Synchrophasors / Phasor Measurement Units (PMUs)

- GPS-synchronized measurements and 10-30 reports per second
- Directly report voltage and current angles/magnitudes
- Really now coming into their own
  - First commercial PMU in 1992
  - Funding from American Recovery and Reinvestment Act of 2008
  - Now enough for *partial* observability in most places
- We're still figuring out what to do with them!
  - Model free system identification approaches (PMU only)
  - Dynamic state estimation / data fusion (PMUs + SCADA)
  - Today: topology update estimation from PMUs + SCADA

Best of Both Worlds

- Combine *model-driven* state estimates with PMU observations
- Goal: Identify system (topology change) events
(line outages, substation change, generator trips, ...)
- Idea: Match PMU measurements $E\Delta v$ to model predictions δv_c
  - Need predictions for many possible changes c!
  - Each δv_c depends on current state – constantly changing.
- How can we do this fast?

Power Flow Basics

Abstract power flow equation

$$H(v; Y) - s = 0$$

In polar form:

$$\begin{bmatrix} H_\ell \\ H_{n+\ell} \end{bmatrix} = \sum_{h=1}^{n} |v_\ell||v_h| \begin{bmatrix} g_{\ell h} & b_{\ell h} \\ -b_{\ell h} & g_{\ell h} \end{bmatrix} \begin{bmatrix} \cos(\theta_{\ell h}) \\ \sin(\theta_{\ell h}) \end{bmatrix},$$

with $\theta_{\ell h} = \theta_\ell - \theta_h$ and

$$s = \begin{bmatrix} P_1 & \cdots & P_n & Q_1 & \cdots & Q_n \end{bmatrix}^T.$$

Power Flows and PMUs

Power flow equation

$$H(v; Y) - s = 0$$

- Measure Ev via PMUs ($E \in \{0, 1\}^{m \times n}$).
- Suppose Ev shifts to $Ev' = Ev + E\Delta v$.
- Goal: Map $E\Delta v$ to change in topology (e.g. ΔY).

Breaker, Breaker!

More detailed: *breaker-level* formulation

$$
\begin{aligned}
H(v; Y) + C\lambda - s &= 0 \\
C^T v &= b
\end{aligned}
$$

Equations in $C^T v = b$ have the form

$v_i - v_j = 0$	Equal voltage on bus sections
$v_i = b_k$	Specified voltage at PV node

Can eliminate constraints if desired – but we don't want to!

Breaker, Breaker!

Breaker-level formulation

$$\begin{aligned} H(v; Y) + C\lambda - s &= 0 \\ C^T v &= b \end{aligned}$$

Notation:

$$x = \begin{bmatrix} v \\ \lambda \end{bmatrix}, \quad A = \begin{bmatrix} \frac{\partial H}{\partial v}(v; Y) & C \\ C^T & 0 \end{bmatrix}, \quad \bar{E} = \begin{bmatrix} E & 0 \end{bmatrix}.$$

Augmented Systems

Consider several possible topology changes:

- Breaker closes in a substation (bus merge)
- Breaker opens in a substation (bus split)
- Load or generator trip
- Line trip

Write each as an *augmented* power flow system.

Example: Breaker Opening

Add a multiplier to "delete" previous constraint in C:

$$\begin{aligned} H(v'; Y) + C\lambda' - s &= 0 \\ C^T v' + F\gamma - b &= 0 \\ F^T \lambda' &= 0 \end{aligned}$$

where $F \in \{0,1\}^{n \times 2}$ indicates voltage for "breakaway" segment.

Example: Line Trip

After trip $Y' = Y + \Delta Y$. H linear in Y, so:

$$
\begin{aligned}
H(v'; Y) + Uw + C\lambda' - s &= 0 \\
C^T v' - b &= 0 \\
w - U^T H(v'; \Delta Y) &= 0
\end{aligned}
$$

where $U \in \{0, 1\}^{n \times 4}$ indicates equations for end-point buses.[1]

[1] Can actually use three rather than four slack variables – see paper.

Strawman Method 0

For each possible topology update (and base case):

- Compute predicted change Δv^{pred}
- Compute mismatch $\mu = \ell(E\Delta v^{\mathrm{pred}} - E\Delta v)$

Report (mismatch, update) pairs in descending order by mismatch.

Problem: Too many possible updates!

Fast Fingerprints

Post-update systems linearized about *pre-update* state look like:

$$\begin{bmatrix} A & U \\ V^T & D \end{bmatrix} \begin{bmatrix} \delta x \\ \gamma \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \end{bmatrix}$$

For each possible update, estimate change to be $\bar{E}\delta x$.

- Core matrix A does not change with updates – factor *once*
- Border matrix and RHS do depend on updates
- Solve bordered systems via a few solves with A

Strawman Method 1

- Save factorization of core matrix A
- For each possible topology update (and base case):
  - Get linearized predicted change δv^{pred}
  - Compute mismatch $\mu = \ell(E\delta v^{\mathrm{pred}} - E\Delta v)$
- Report (mismatch, update) pairs in descending order by mismatch.

Problem: Too many possible updates! And maybe approximation error?

Filtering

Care about some loss $\ell(E\delta x - E\Delta x)$, where

$$\begin{bmatrix} A & U \\ V^T & D \end{bmatrix} \begin{bmatrix} \delta x \\ \gamma \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \end{bmatrix}$$

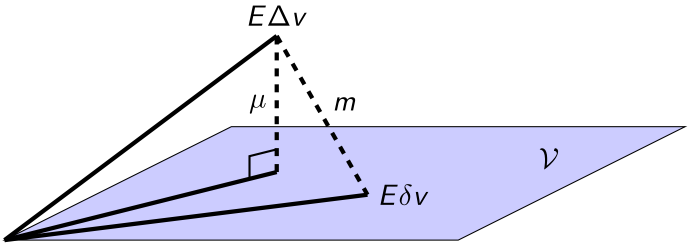
Rewrite as

$$E\delta v = \bar{E}A^{-1}(d_1 - U\gamma)$$

and

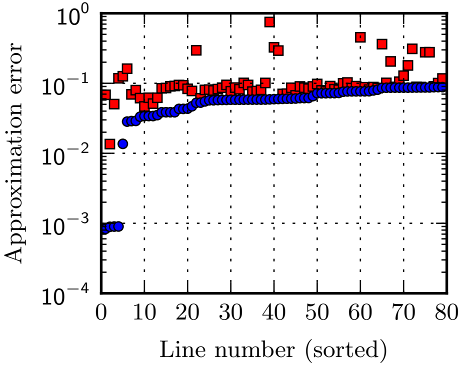
U narrow and *sparse* $\implies$
$(\bar{E}A^{-1})U$ involves only a few columns of $\bar{E}A^{-1}$

Partial Predictions



- Find *subspace* $\mathcal{V}_c$ containing predictions $E\delta v_c$
- Bound: subspace distance $\mu(c) \leq$ mismatch $m(c)$
- Sort events by ascending $\mu(c)$
- Check $c_1, \ldots, c_k$ until $\mu(c_{k+1}) \leq \min_{1 \leq j \leq k} m(c_j)$

Filter Effectiveness (IEEE 57-bus, line trips only)



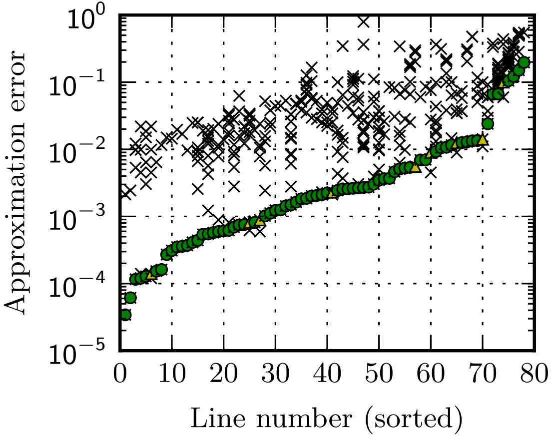
Blue squares are filter scores, red squares are actual mismatches.

But is it right?

Test scenarios (IEEE 57-bus network)

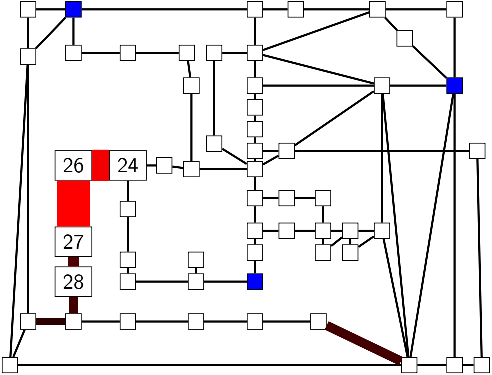
- Three PMU deployments: everywhere, sparse, and single
- With no noise or Gaussian noise ($\sigma = 0.0017$)
  - Consistent with largest phase angle error allowed by the synchrophasor standard (0.1 degree)
- Will show behavior with bad data later

Line Failure Diagnosis (sparse, no noise)



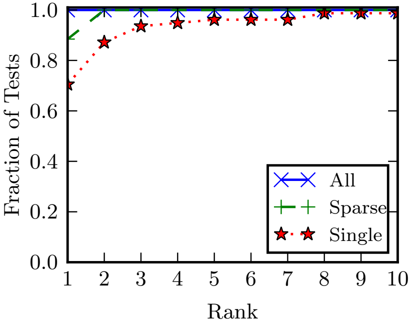
Green/yellow: scores for correctly/incorrectly diagnosed tripped lines.
Black crosses: scores for other lines.

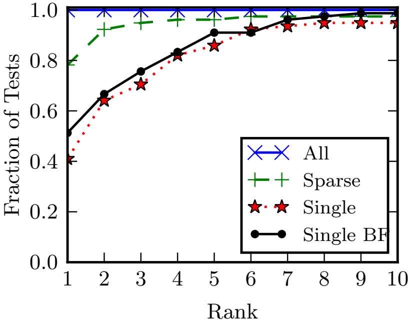
Example: Hard Case in IEEE 57-bus



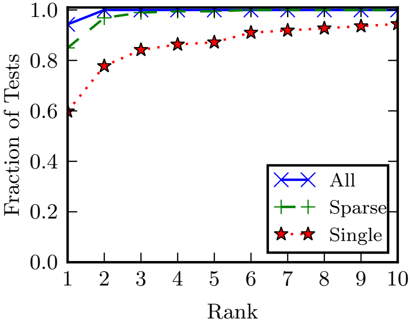
(24,26) tripped, (26,27) chosen. Color/thickness by mismatch$^{-1/2}$.
PMU locations are highlighted in blue.

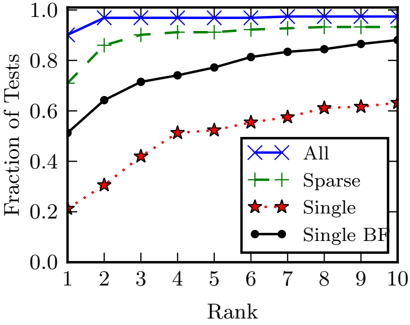
IEEE 57-bus: Line Failures



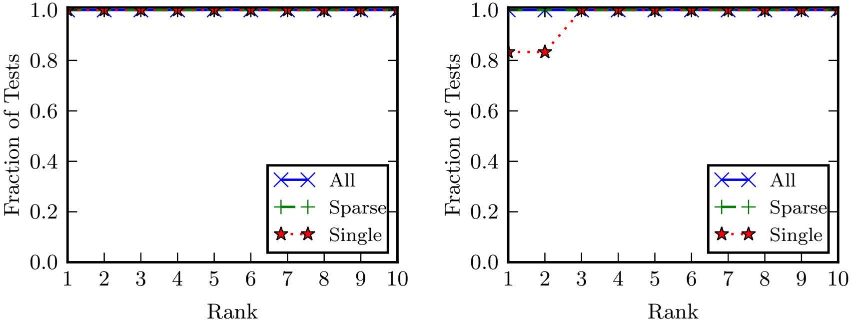


IEEE 57-bus: Substation Reconfigurations

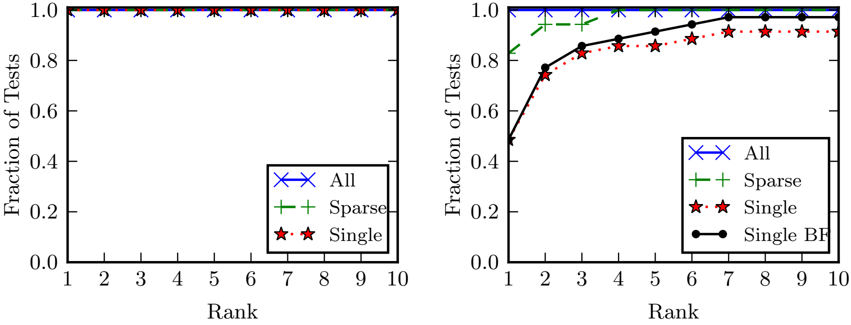




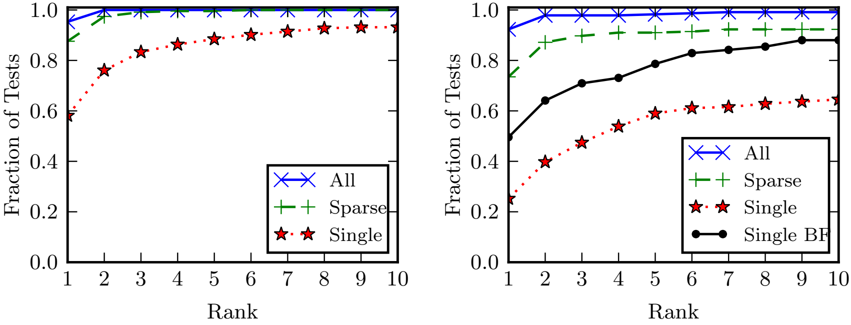
IEEE 57-bus: Generator Trips



IEEE 57-bus: Load Trips



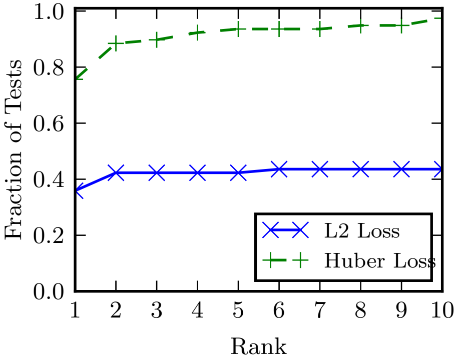
IEEE 57-bus: All Contingencies



Bad Data Robustness via Huber

- Experiments so far reflect ℓ^2 loss.
- With outliers/bad data: use a more robust loss!
- We use Huber with scale parameter $1.365 \cdot 0.0017$.

IEEE 57-bus: All Contingencies



One PMU delivers 5 degree errors (sparse PMU deployment) + Gaussian noise on all readings.

Ongoing Related Efforts

- Spectroscopic event identification (Eric Lee, Nate Rogalskyj)
  - Goal: Identify state from ringdown/ambient oscillations
  - Approach: Residual + bound generation similar to FLiER
- SECURED: Synthetic regulating reserves (Eaton, CMU, ANL, LLNL)
  - Goal: Reduce regulating reserve req'ts to offset VER
  - Approach: Fast distribution-level coordinated demand response
- GridCloud (Birman, WSU)
  - Goal: Fast, reliable cloud infrastructure to communicate PMU data
  - Approach: Replication for performance and reliability

Acknowledgements

FLiER: Practical Topology Update Detection Using Sparse PMUs.
Ponce and Bindel, arXiv:1409.6644