Grid Event Fingerprints and PMUs

Colin Ponce    David Bindel

March 14, 2014
Basic picture

- Direct state measurements (e.g. via PMU) at some buses
- Not enough for complete observability
  - Because of incomplete deployment in local grid
  - Because of interactions with neighboring parts of grid
- Goal: Check for fingerprint of significant events
Variations on a theme

- **Steady state fingerprints (now)**
  - Measurement: Change in steady-state voltage
  - Events considered: Failure of one or two lines
  - Fingerprints derived computationally

- **Transient fingerprints (next)**
  - Measurement: Time-aligned windowed PMU transient data
  - Events considered: Line failures, problems in neighbors?
  - Fingerprints derived from computation or recordings
Steady state fingerprints

Concrete case: fingerprint for line failures in the network

- State: complex bus voltage vector $\nu$
- Fast observation: subset of voltage vector $Ev$
- Fingerprint: change $E\delta\nu$

Question: Can we find a line failure ($\Delta Y$) that explains $E\delta\nu$?
Steady state fingerprint test

Line \((i,j)\) fails, \(Y \mapsto Y + \Delta Y_{ij}\). Power flow equations:

\[
A_{ij}\delta v = (A + U_{ij}CU_{ij}^T)\delta v = b_{ij} + O(\|\delta v\|^2)
\]

where \(U_{ij} \in \mathbb{R}^{n \times 4}\) and \(b_{ij} \in \mathbb{R}^n\) are simple functions of \((i,j)\).

Linearize in \(\delta v\) to get fingerprint:

- Fingerprint: \(E\delta v_{ij} = EA_{ij}^{-1}b_{ij}\).
- Fingerprint distance: \(t_{ij} = \|E\delta v_{ij} - \delta v\|\)

Computing \(t_{ij}\) may require two linear solves with \(A\).
Fast filtering

Goal: Avoid a linear solve to compute each fingerprint distance.

Start with defining equation

$$(A + U_{ij}CU_{ij}^T)\delta v_{ij} = b_{ij}$$

Rewrite as

$$\delta v_{ij} = A^{-1}(b_{ij} - U_{ij}CU_{ij}^T \delta v_{ij})$$

Bound

$$t_{ij} \equiv \|E\delta v_{ij} - E\delta v\| \geq s_{ij} \equiv \min_z \|EA^{-1}(b_{ij} - U_{ij}z) - E\delta v\|$$
Algorithm

Compute and store $EA^{-1}$.
Compute $s_{ij}$ for all $ij$.
Order $ij$’s by ascending $s_{ij}$.
for all transmission lines $i, j$ do
    Keep track of $M :=$ smallest $t_{ij}$ yet found.
    If $s_{ij} > M$, continue.
    Compute $t_{ij}$.
end for
Select smallest computed $t_{ij}$. 
### How Accurate Is It?

#### IEEE 57-Bus Test Network

<table>
<thead>
<tr>
<th># PMUs</th>
<th>% Correct</th>
<th>% In Top 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>77 %</td>
<td>81.2 %</td>
</tr>
<tr>
<td>3</td>
<td>86.5 %</td>
<td>93.2 %</td>
</tr>
<tr>
<td>Everywhere</td>
<td>94.6 %</td>
<td>94.6 %</td>
</tr>
</tbody>
</table>
Results with 3 PMUs

\[ \log \left( \frac{\|E \delta v - E \delta v_{ij}\|}{\|E \delta v\|} \right) \]
## Filter Results

### No. of Buses that Get Past Filter

<table>
<thead>
<tr>
<th></th>
<th>IEEE 57-Bus Network</th>
<th>IEEE 118-Bus Network</th>
</tr>
</thead>
<tbody>
<tr>
<td># PMUs</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>Median</td>
<td>5.0</td>
<td>4.0</td>
</tr>
<tr>
<td>Mean</td>
<td>10.1</td>
<td>13.1</td>
</tr>
<tr>
<td>Stddev</td>
<td>15.2</td>
<td>29.4</td>
</tr>
</tbody>
</table>
Two Line Failures?

- Possible failure scales quadratically!
- IEEE 57 bus: 76 lines, 2850 pairs
- Can we narrow it down with fingerprints?
Empirical Observation

When lines $i$ and $j$ fail, change in voltage $\delta v$ often looks like

$$\delta v = \delta v_i + \text{other stuff},$$

where $\delta v_i$ is voltage change if just $i$ failed.
Two Lines: Current Approach

Start by guessing at one of the failed lines.

- Compute effects of single-line failures $\delta v_i$: cheap compared to testing all pairs!

- Let

$$M = \begin{bmatrix}
\delta v_1 & \delta v_2 & \cdots & \delta v_m
\end{bmatrix}$$
Two Lines: Current Approach

Start by guessing at one of the failed lines.

- Compute effects of single-line failures $\delta v_i$: cheap compared to testing all pairs!

- Let

$$M = \begin{bmatrix} \delta v_1 & \delta v_2 & \cdots & \delta v_m \end{bmatrix}$$

- Solve

$$\min_x (E \delta v - M x)^2 + \|x\|_1 \text{ for } x \in [0, 1]^m$$

to guess at one failure.
Two Lines: Current Approach

- Resulting $x$ scores each line in network with failure likelihood.
- Take most likely single line choices.
- For that choice, run **one-line algorithm** to get the second.
- Create a list of top scoring pairs.
- Check top 20 or 40 pairs exactly.
Two Lines: Initial Results (IEEE 57)

Top 3: 73%
Top 40: 80%
Future Directions
Fingerprints for Transient Analysis

- Transients travel through network.
- PMUs can observe transients, create a fingerprint.
- Use fingerprint to analyze transient.
  - Have I seen something like this before?
  - What is its likely effect?
  - How quickly will it transit to a neighboring operator?
- Goal: Fast lookup for data streaming from GridCloud