Graph learning for power system state estimation

Tongxin Li **Lucien Werner** Steven Low

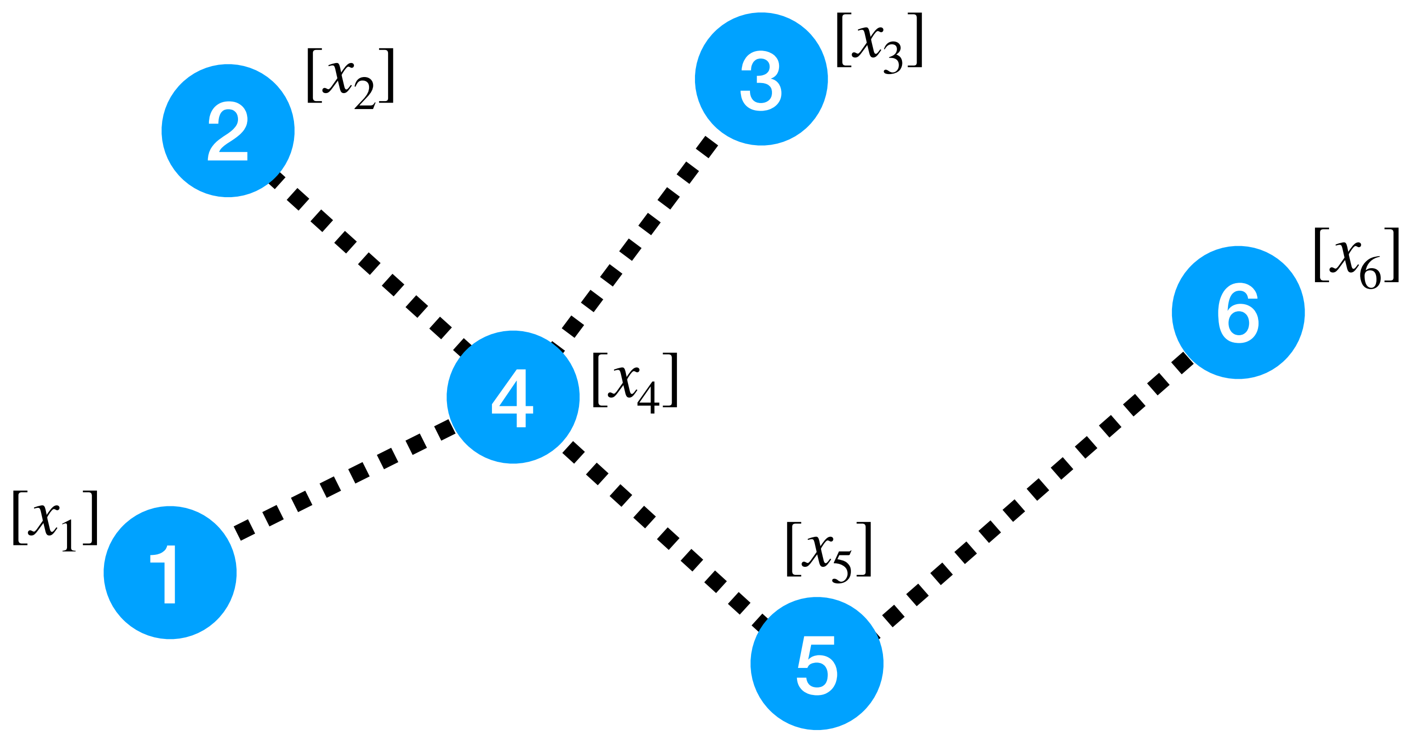
California Institute of Technology

13 December, 2019

IEEE CDC 2019, Nice FR

Graph learning problem

- Consider the graph G with $\mathscr{V}$ given and $\mathscr{E} = ?$
- G generates measurements x_i at each node according to $f(\mathrm{G})$



- Learning objective: given $\{x_i\}$ and $f(\mathrm{G})$, infer $\mathscr{E}$
- Motivating application: connectivity and line impedances in power networks may be unknown (e.g., topology changes due to breakers tripping)

Our contribution

Model-free learning (statistical inference)

- Markov random fields
- Gaussian graphical models
- Bayesian networks

Topology & parameter identification for power systems

- Chow-Liu algorithm
- Compressed sensing
- Tree structure learning

This talk: algorithm for model-based graph learning with linear measurements

- Exploit structural properties of the graph Laplacian
- Low sample complexity, even in graphs with high-degree nodes
- Power system application: joint recovery of topology and parameters

Graph matrix

Given $G = (\mathscr{V}, \mathscr{E})$, a *symmetric* matrix $Y(G) \in \mathbb{S}^{n\times n}$ is called a **graph matrix** if the following conditions hold:

$$Y_{i,j}(G) = \begin{cases} \neq 0 & \text{if } i \neq j \text{ and } (i,j) \in \mathscr{E} \\ 0 & \text{if } i \neq j \text{ and } (i,j) \notin \mathscr{E} \\ \text{arbitrary} & \text{otherwise} \end{cases}.$$

- e.g., Laplacian, adjacency matrix, nodal admittance matrix

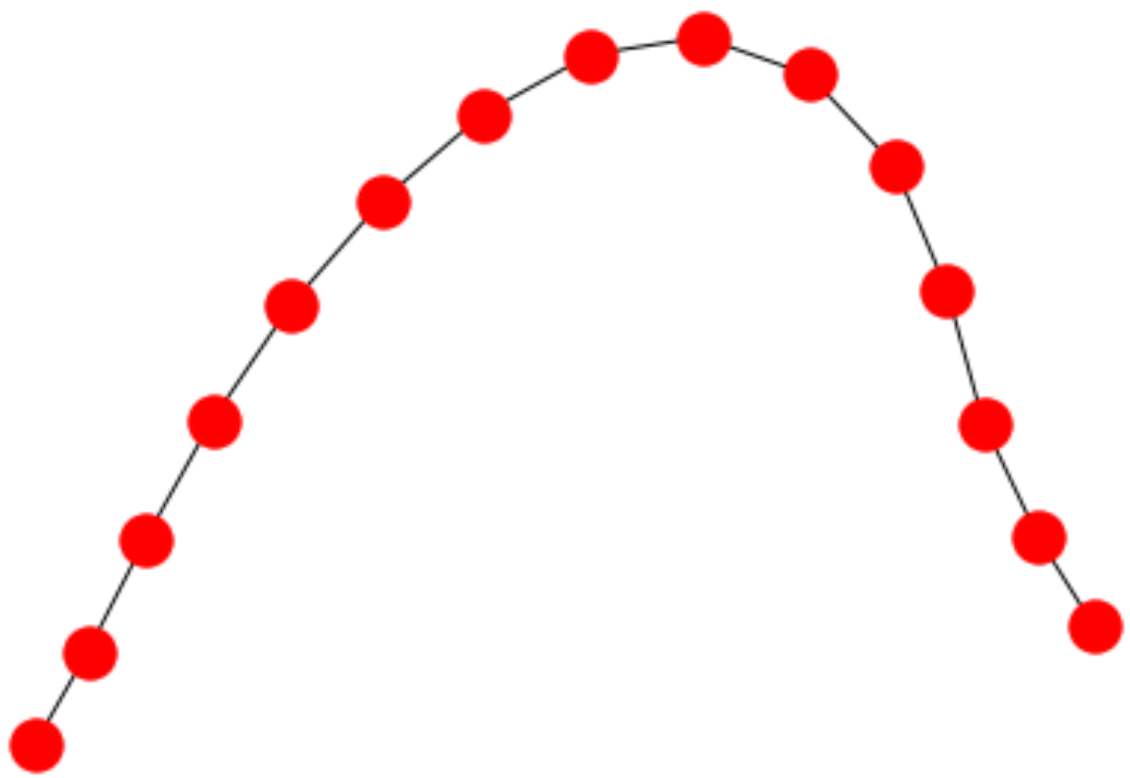
Our setting: Nodal admittance matrix $Y \in \mathbb{C}^{n\times n}$ encodes the topology, impedance, and susceptance properties of an electrical network

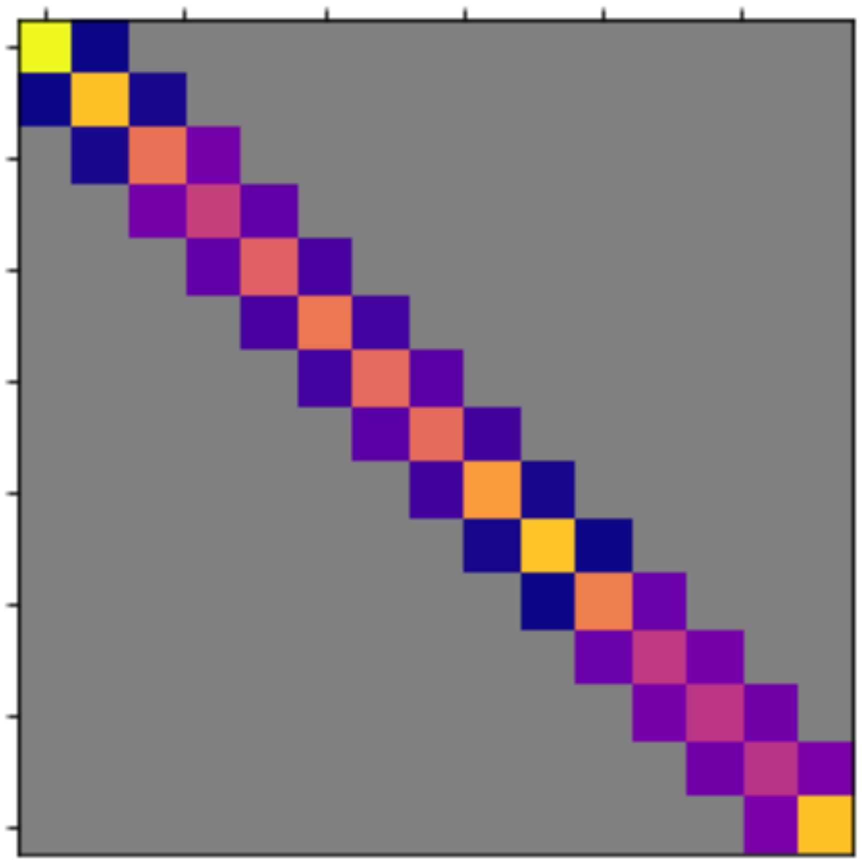
- Sparse, symmetric, off-diagonal entries must be entry-wise ≥ 0 or ≤ 0
- Physical meaning: Let $I \in \mathbb{C}^n$ nodal current injections, $V \in \mathbb{C}^n$ nodal voltages. Then Kirchhoff's Laws state that

$$I = YV$$

Admittance matrix structure

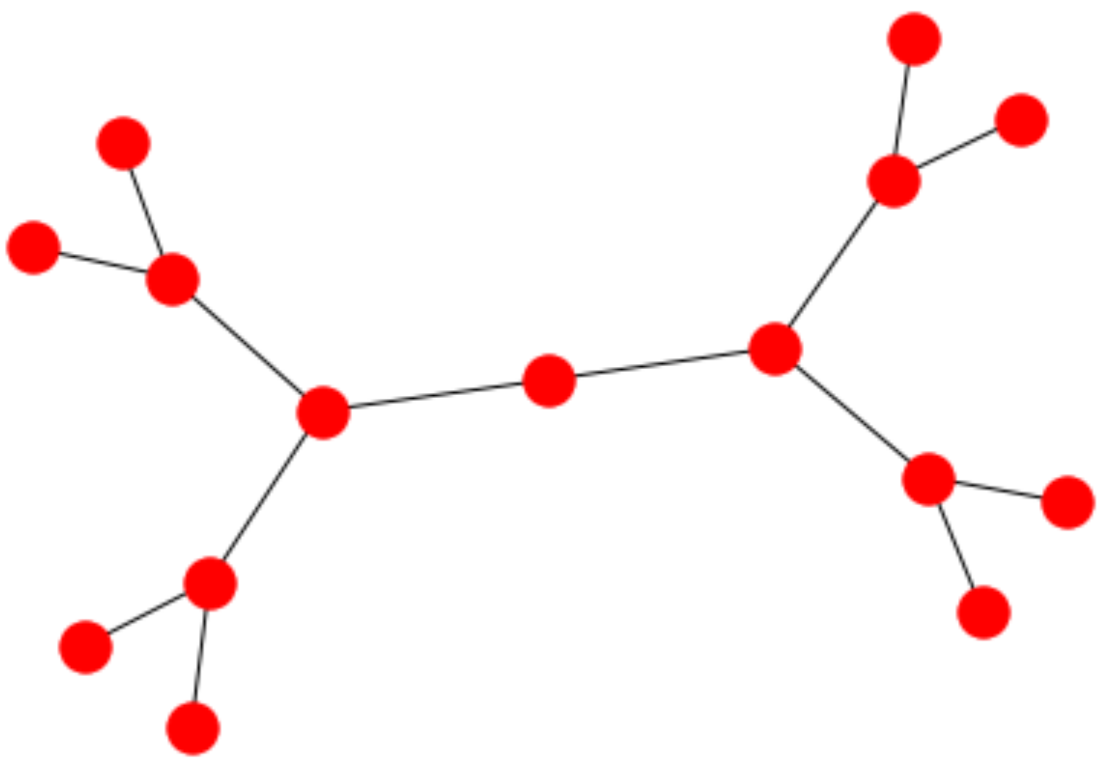
Chain topology

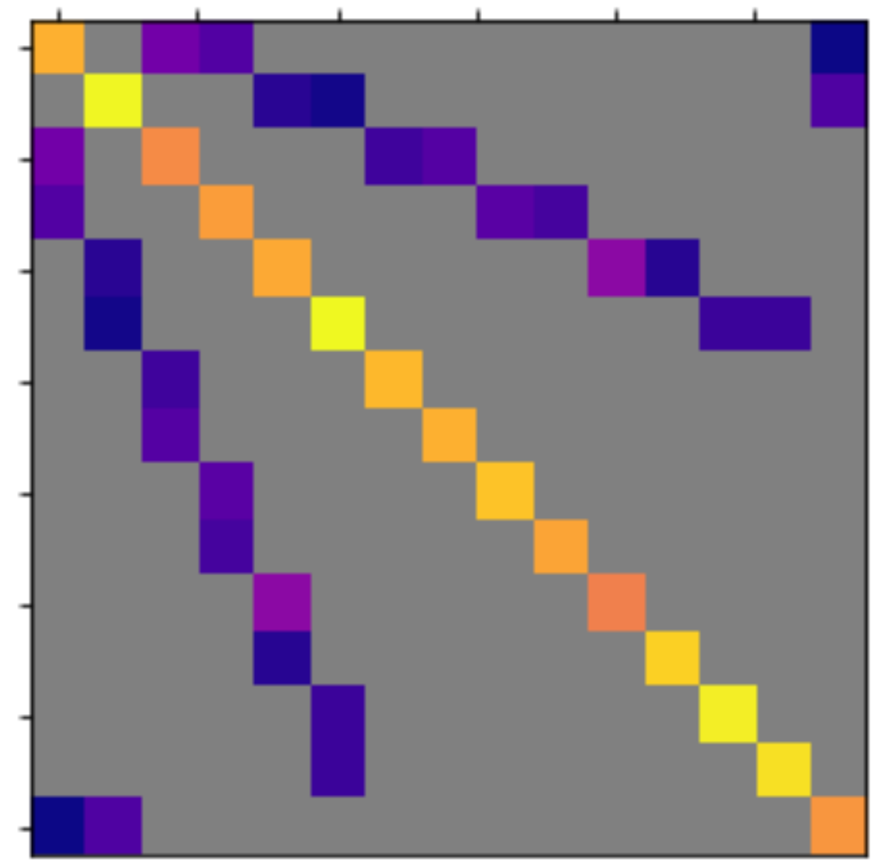




Admittance matrix structure

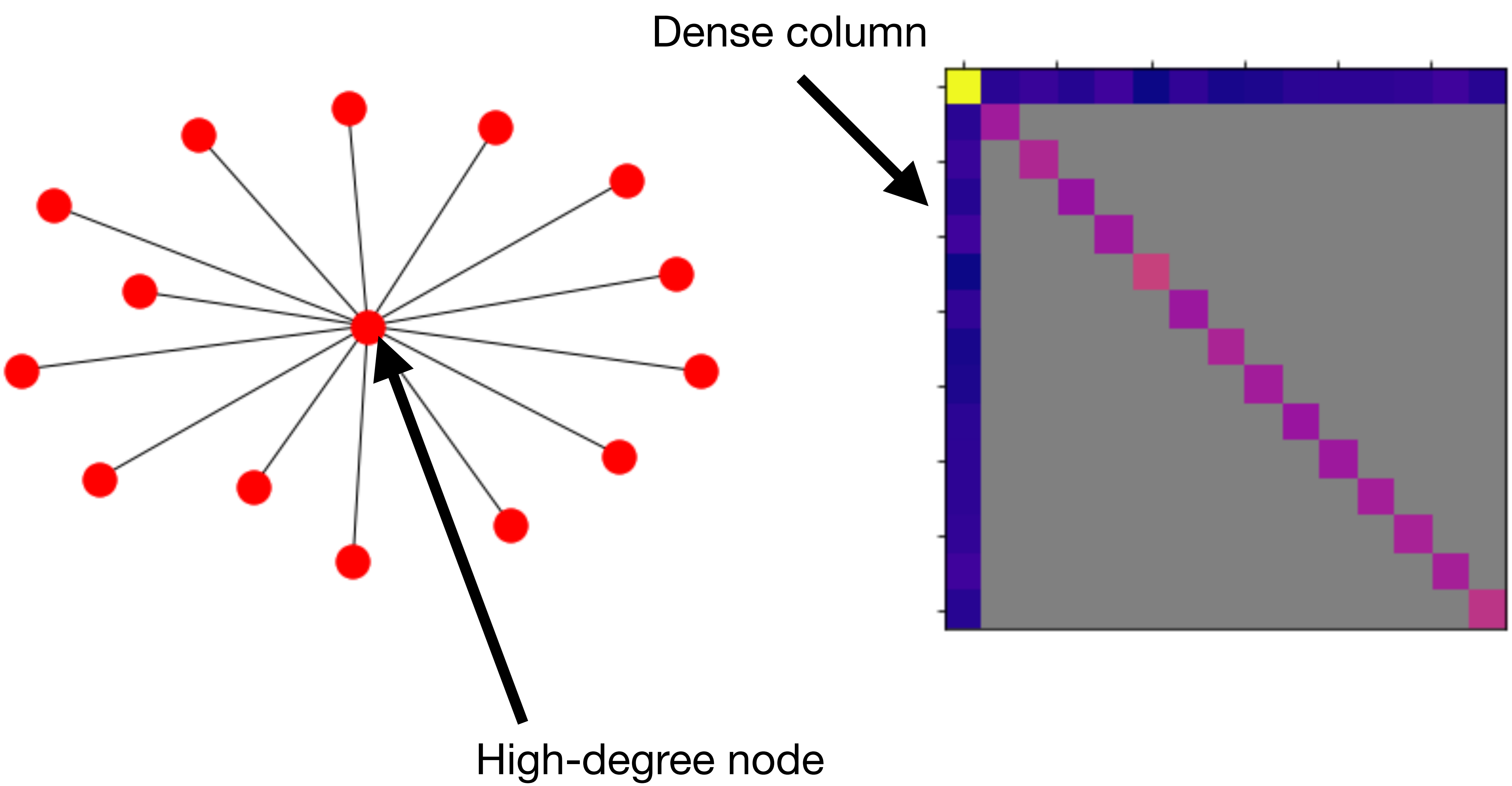
Tree topology





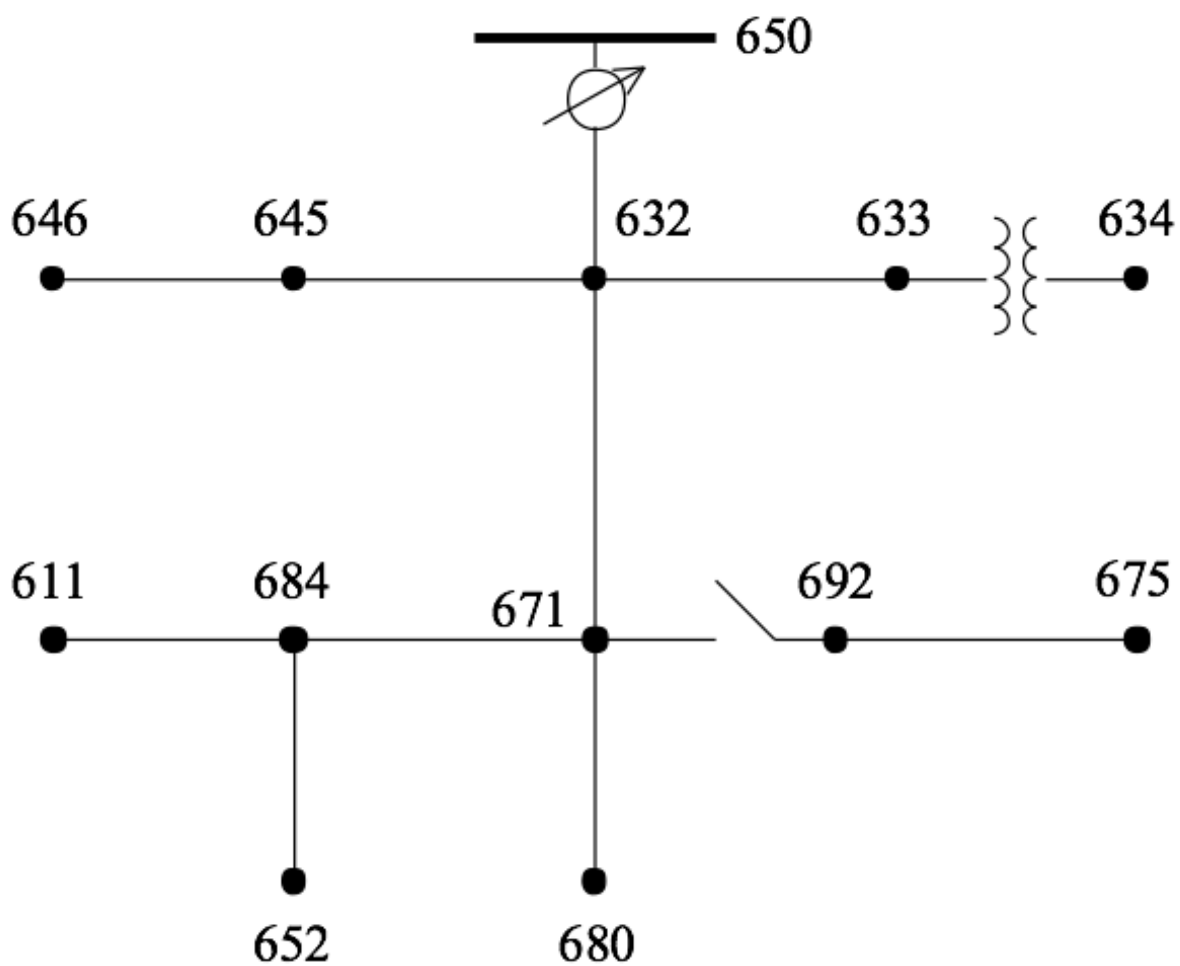
Admittance matrix structure

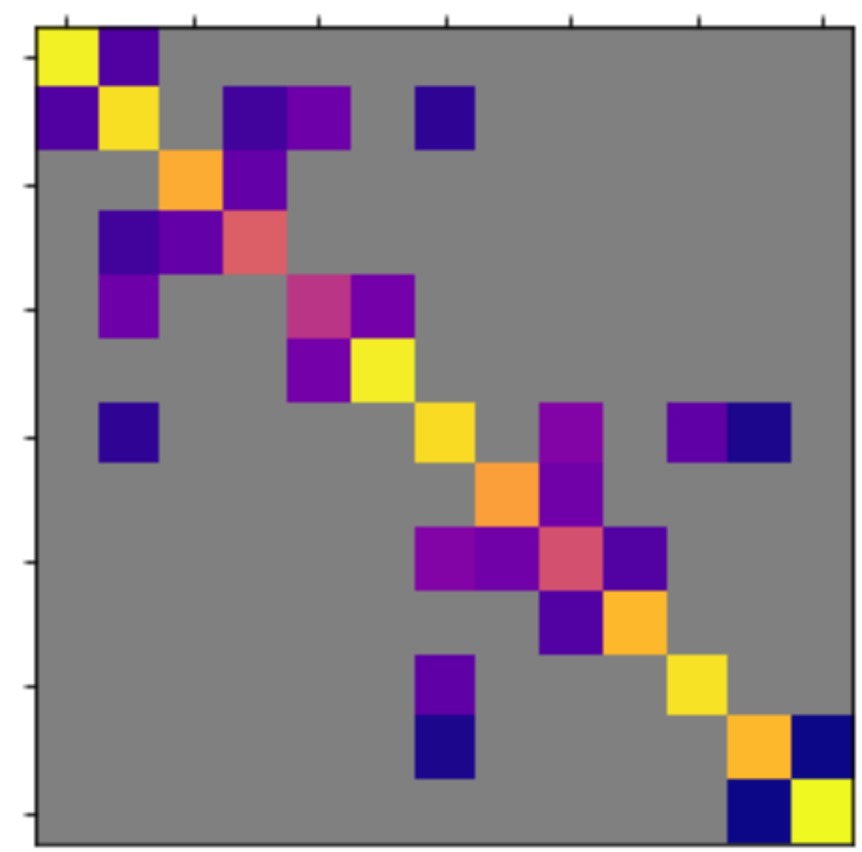
Star topology



Admittance matrix structure

IEEE 13-bus radial feeder

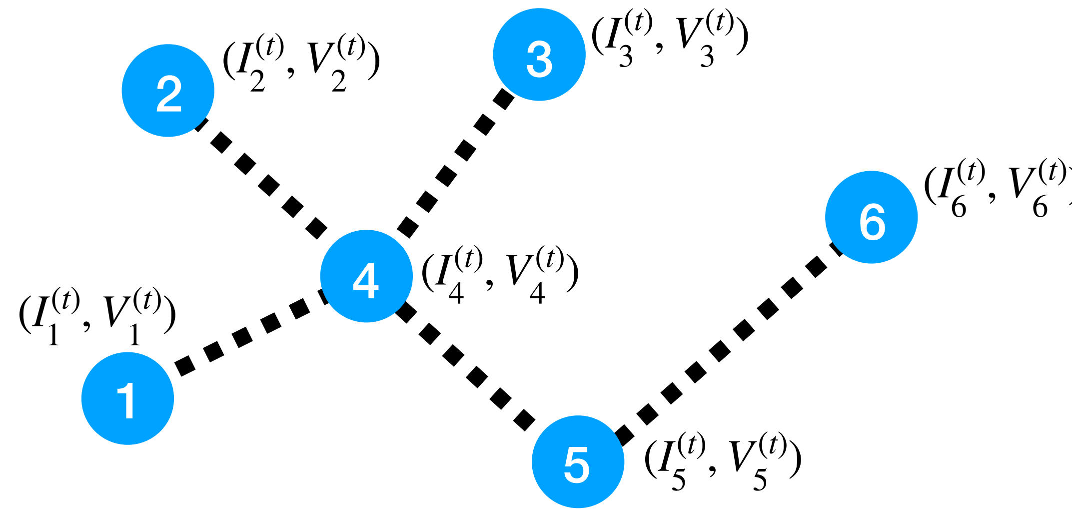




Linear measurements

- For each $t = 1, \ldots, m$, assume graph G generates $I^{(t)} \in \mathbb{C}^n$ and $V^{(t)} \in \mathbb{C}^n$
- Define matrices $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{m \times n}$

$$\mathbf{A} := \begin{bmatrix} - & I^{(1)\top} & - \\ - & I^{(2)\top} & - \\ \vdots & \vdots & \vdots \\ - & I^{(m)\top} & - \end{bmatrix} \qquad \mathbf{B} := \begin{bmatrix} - & V^{(1)\top} & - \\ - & V^{(2)\top} & - \\ \vdots & \vdots & \vdots \\ - & V^{(m)\top} & - \end{bmatrix}$$



- $f(\mathrm{G})$: Kirchhoff's Law with additive noise

$$\mathbf{A} = \mathbf{B}Y(G) + \mathbf{Z}$$

Assumptions:
- $\mathbf{B}$ is a random matrix
- $\mathbf{Z} \in \mathbb{K}^{m \times n}$ is random noise

- **Goal: recover estimate X of $Y(\mathrm{G})$ using measurements $\mathbf{A}, \mathbf{B}$ and assumptions on the structure of $Y(\mathrm{G})$**

Fundamental limits

Error criterion for topology identification

$$\varepsilon_T := \mathbb{P}\left(\exists\ i \neq j \ \middle|\ \mathrm{sign}(X_{i,j}) \neq \mathrm{sign}\left(Y_{i,j}(G)\right)\right)$$

Theorem

Let $G \sim \mathscr{G}$. The *probability of error for topology identification* ε_T is bounded from below as

$$\varepsilon_T \geq 1 - \frac{\mathbb{H}\left(\mathbf{A}\right) - \mathbb{H}\left(\mathbf{Z}\right) + \ln 2}{\mathbb{H}\left(\mathscr{G}\right)}$$

where $\mathbb{H}\left(\mathbf{A}\right)$, $\mathbb{H}\left(\mathbf{Z}\right)$ and $\mathbb{H}\left(\mathscr{G}\right)$ are differential entropy (in base e) functions of the random variables $\mathbf{A}$, $\mathbf{Z}$, and probability distribution $\mathscr{G}$, respectively.

Proof idea: Generalized Fano's Inequality.

Naive sparse recovery doesn't work

$$\mathbf{A} = \mathbf{B}Y$$

sparse matrix

- With $m = n$ measurements, problem is straightforward

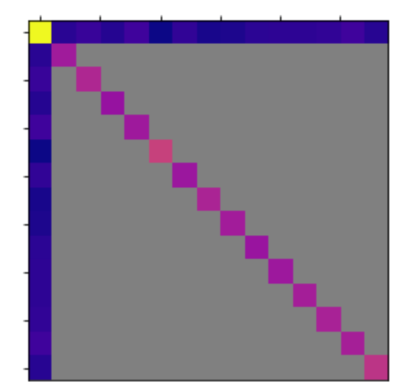
- If we want $m \ll n$, vectorize and apply techniques from sparse recovery. Catch: requires $\Theta(1)$ non-zeros in Y.

$$\underbrace{\begin{bmatrix} \quad \mathbf{A} \quad \end{bmatrix}}_{m \times n} = \underbrace{\begin{bmatrix} \quad \mathbf{B} \quad \end{bmatrix}}_{m \times n} \underbrace{\begin{bmatrix} \quad Y \quad \end{bmatrix}}_{n \times n} \rightarrow \underbrace{\begin{bmatrix} \mathrm{vec}(\mathbf{A}) \end{bmatrix}}_{mn \times 1} = \underbrace{\begin{bmatrix} \mathbf{B} & & & \\ & \mathbf{B} & & \\ & & \ddots & \\ & & & \mathbf{B} \end{bmatrix}}_{mn \times n^2} \underbrace{\begin{bmatrix} \mathrm{vec}(Y) \end{bmatrix}}_{n^2 \times 1}$$

- When might this not work? G may be a sparse graph *on average* yet still contain some completely dense columns/rows

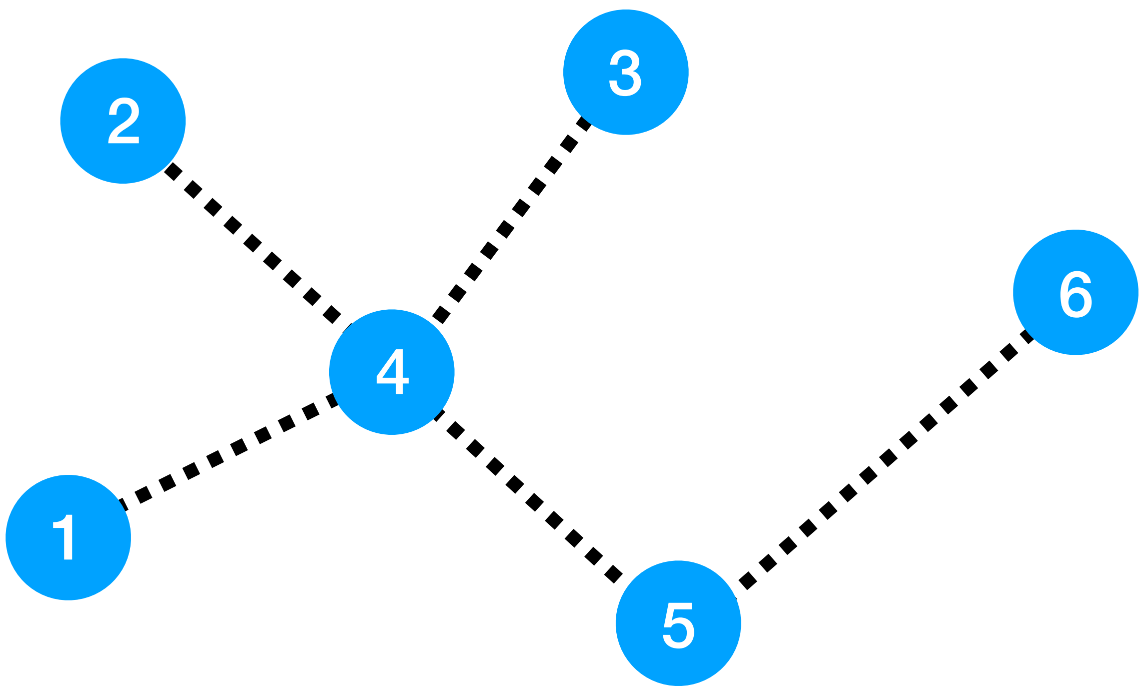
  - For some j, still have to solve $A_j = \mathbf{B}Y_j$ for Y_j dense

  - $\Longrightarrow$ cannot apply sparse recovery techniques off-the-shelf



Two-stage recovery scheme

- Consider a 6-node graph:

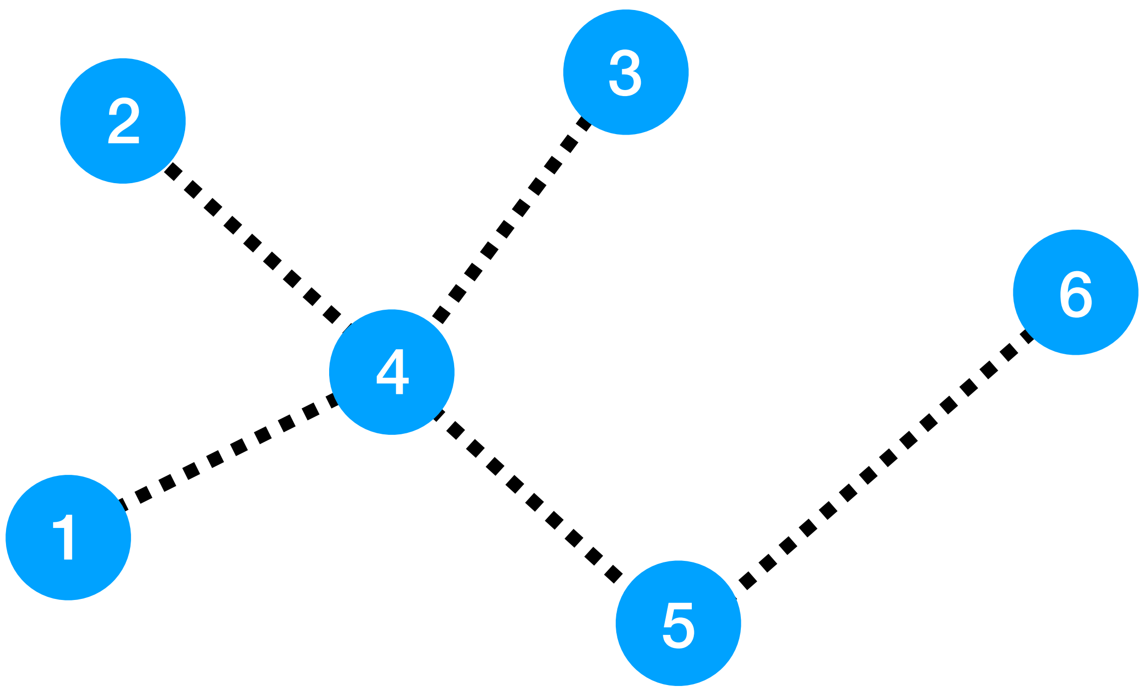


$$\begin{array}{c} \begin{bmatrix} Y_{1,1} & & Y_{1,3} & Y_{1,4} & & \\ & Y_{2,2} & & Y_{2,4} & & \\ Y_{3,1} & & Y_{3,3} & Y_{3,4} & & \\ Y_{4,1} & Y_{4,2} & Y_{4,3} & Y_{4,4} & Y_{4,5} & \\ & & & Y_{5,4} & Y_{5,5} & Y_{5,6} \\ & & & & Y_{6,5} & Y_{6,6} \end{bmatrix} \\ (\star) \end{array}$$

- $(\star)$ is the only non-“sparse” column/row

Two-stage recovery scheme

- Consider a 6-node graph:

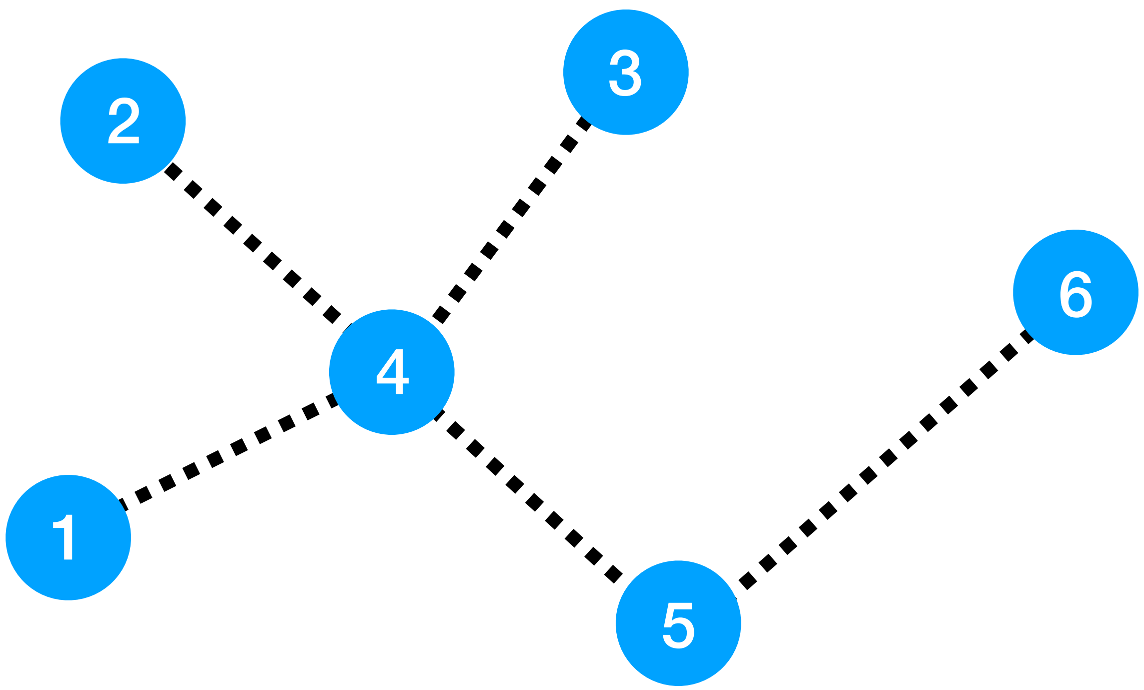


$$
\begin{array}{c}
\left[\begin{array}{cccccc}
\mathbf{Y}_{1,1} & & \mathbf{Y}_{1,3} & Y_{1,4} & & \\
& \mathbf{Y}_{2,2} & & Y_{2,4} & & \\
\mathbf{Y}_{3,1} & & \mathbf{Y}_{3,3} & Y_{3,4} & & \\
\mathbf{Y}_{4,1} & \mathbf{Y}_{4,2} & \mathbf{Y}_{4,3} & Y_{4,4} & \mathbf{Y}_{4,5} & \\
& & & Y_{5,4} & \mathbf{Y}_{5,5} & \mathbf{Y}_{5,6} \\
& & & & \mathbf{Y}_{6,5} & \mathbf{Y}_{6,6}
\end{array}\right] \\
\begin{array}{cccccc}
\phantom{\mathbf{Y}_{1,1}} & \phantom{\mathbf{Y}_{2,2}} & \phantom{\mathbf{Y}_{3,3}} & (\star) & \phantom{\mathbf{Y}_{4,5}} & \phantom{\mathbf{Y}_{5,6}}
\end{array}
\end{array}
$$

- $(\star)$ is the only non-“sparse” column/row
- Solve each “sparse” column independently

Two-stage recovery scheme

- Consider a 6-node graph:

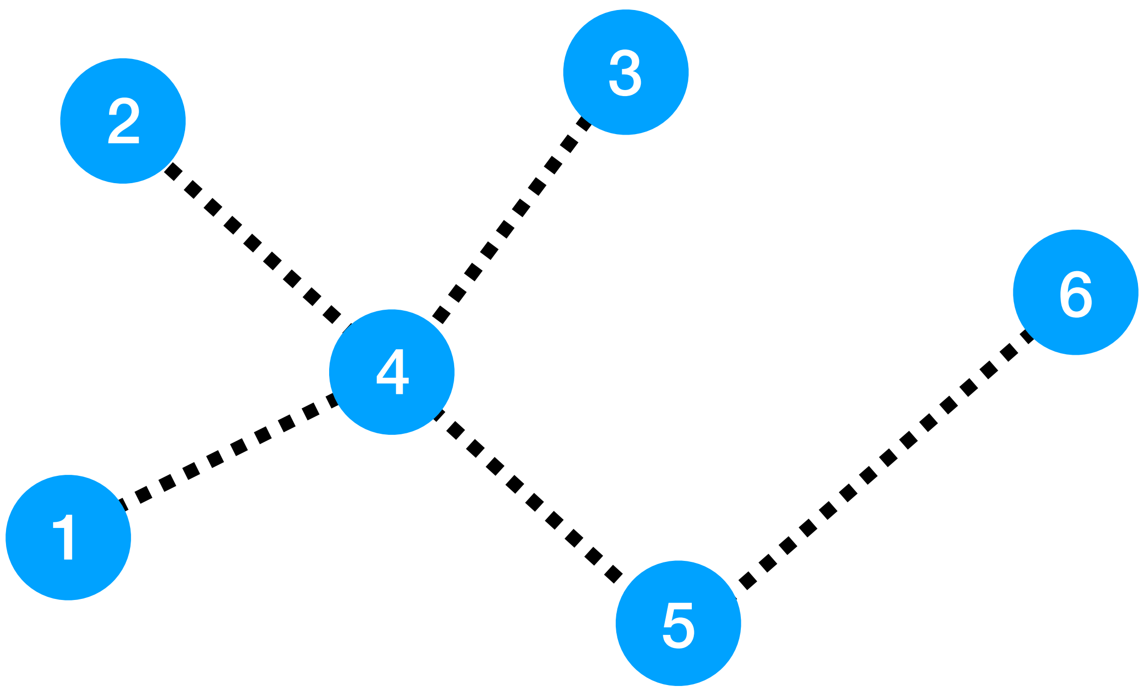


$$\begin{array}{c} \begin{bmatrix} \mathbf{Y}_{1,1} & & \mathbf{Y}_{3,1} & \mathbf{Y}_{1,4} & & \\ & \mathbf{Y}_{2,2} & & \mathbf{Y}_{2,4} & & \\ \mathbf{Y}_{3,1} & & \mathbf{Y}_{3,3} & \mathbf{Y}_{3,4} & & \\ \mathbf{Y}_{4,1} & \mathbf{Y}_{4,2} & \mathbf{Y}_{4,3} & Y_{4,4} & \mathbf{Y}_{4,5} & \\ & & & \mathbf{Y}_{5,4} & \mathbf{Y}_{5,5} & \mathbf{Y}_{5,6} \\ & & & & \mathbf{Y}_{6,5} & \mathbf{Y}_{6,6} \end{bmatrix} \\ (\star) \end{array}$$

- $(\star)$ is the only non-“sparse” column/row
- Solve each “sparse” column independently
- By symmetry, we know most of the entries of the dense column

Two-stage recovery scheme

- Consider a 6-node graph:



$$\begin{array}{c}\begin{bmatrix} \mathbf{Y}_{1,1} & & \mathbf{Y}_{1,3} & \mathbf{Y}_{1,4} & & \\ & \mathbf{Y}_{2,2} & & \mathbf{Y}_{2,4} & & \\ \mathbf{Y}_{3,1} & & \mathbf{Y}_{3,3} & \mathbf{Y}_{3,4} & & \\ \mathbf{Y}_{4,1} & \mathbf{Y}_{4,2} & \mathbf{Y}_{4,3} & \mathbf{Y}_{4,4} & \mathbf{Y}_{4,5} & \\ & & & \mathbf{Y}_{5,4} & \mathbf{Y}_{5,5} & \mathbf{Y}_{5,6} \\ & & & & \mathbf{Y}_{6,5} & \mathbf{Y}_{6,6} \end{bmatrix} \\ \begin{array}{cccccc} \phantom{\mathbf{Y}_{1,1}} & \phantom{\mathbf{Y}_{1,1}} & \phantom{\mathbf{Y}_{1,1}} & (\star) & \phantom{\mathbf{Y}_{1,1}} & \phantom{\mathbf{Y}_{1,1}} \end{array}\end{array}$$

- $(\star)$ is the only non-"sparse" column/row
- Solve each "sparse" column independently
- By symmetry, we know most of the entries of the dense column
- Solve for the remaining unknown entries (i.e., $Y_{4,4}$)

Two-stage recovery scheme

Stage 1: Independently recover columns

$$\begin{aligned} \min \quad & ||X_j||_1 \\ \textbf{s.t.} \quad & \mathbf{B}X_j = A_j \\ X_j \in \mathbb{F}^n & \end{aligned} \qquad \forall j = 1, \ldots, n$$

Stage 2: Check consistency of columns

1. Select a subset $\mathcal{S}$ from $\{1, \ldots, n\}$
2. Check $X_{i,j} = X_{j,i}$ for all $i, j \in \mathcal{S}$
3. If so, fix these entries as “correct” and reduce system order
4. Otherwise, select a different subset from $\mathcal{S}$ and repeat from step 2
5. Solve remaining entries of Y by iterating from step 1

Analysis of two-stage scheme

We want to solve for an estimate X of Y in the linear system

$$\mathbf{A} = \mathbf{B}Y + \mathbf{Z}$$

Error criterion for accurate recovery of Y

$$\varepsilon_P = \sup_Y \mathbb{P}(X \neq Y)$$

- $\mu \in [0,n-2]$ = largest number of non-zeros in a column of Y
- $K \in [0,n-1]$ = number of columns in Y with $> \mu$ non-zeros

<u>Theorem</u> (Worst-case sample complexity)

1. Suppose that $\mathbf{B}$ has Gaussian IID entries and $\mathbf{Z} = \mathbf{0}$
2. Suppose $\mu < n^{-3/\mu}(n-K)$ and $K = o(n)$.

Then for all (μ, K)-sparse sequences of distributions, the two-stage algorithm guarantees that $\lim_{n\to\infty} \varepsilon_P = 0$ with $m = O(\mu \log(n/\mu) + K)$ measurements.

Tight bounds

For certain distributions of graphs, bounds are tight

Corollary (Trees)

Suppose that $\mathbf{B}$ has Gaussian IID entries with zero mean and variance 1. Assume that Y is a graph matrix for a tree. Then $m = \Theta(\log n)$.

Corollary (Erdos-Renyi graphs)

Suppose that $\mathbf{B}$ has Gaussian IID entries with zero mean and variance 1. Assume that Y is a graph matrix for a graph G sampled from the Erdos-Renyi distribution $\mathcal{G}_{\text{ER}}(n, p)$ with $1/n \leq p \leq 1 - 1/n$. Then $m = \Theta(\log nh(p))$.

Practical algorithm

- How to determine *in practice* which of the columns of Y has been "correctly" recovered?

$$
\begin{aligned}
\min \quad & ||X_j||_1 \\
\textbf{s.t.} \quad & \mathbf{B}X_j = A_j \\
X_j \in \mathbb{F}^n &
\end{aligned}
$$

$$
\begin{bmatrix}
Y_{1,1} & & Y_{1,3} & Y_{1,4} & & \\
& Y_{2,2} & & Y_{2,4} & & \\
Y_{3,1} & & Y_{3,3} & Y_{3,4} & & \\
Y_{4,1} & Y_{4,2} & Y_{4,3} & Y_{4,4} & Y_{4,5} & \\
& & & Y_{5,4} & Y_{5,5} & Y_{5,6} \\
& & & & Y_{6,5} & Y_{6,6}
\end{bmatrix}
$$

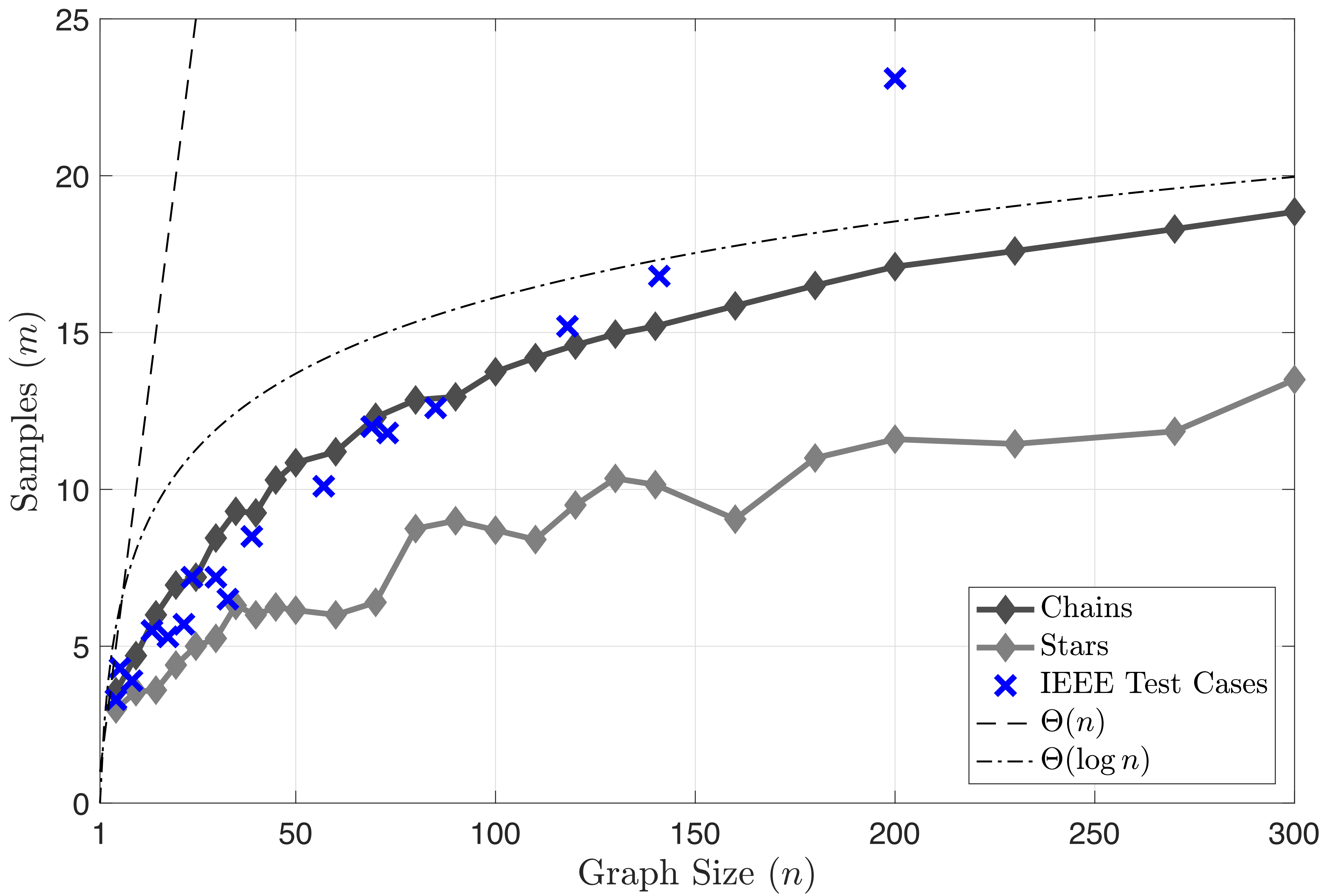
- Score each column

$$
\text{score}_j = \sum_{i=1}^{n} |X_{i,j} - X_{j,i}|
$$

- Sort columns in increasing score order, pick first $|\mathcal{S}|$ as being correct.

- Fix correct columns in X, reduce system order, iterate.

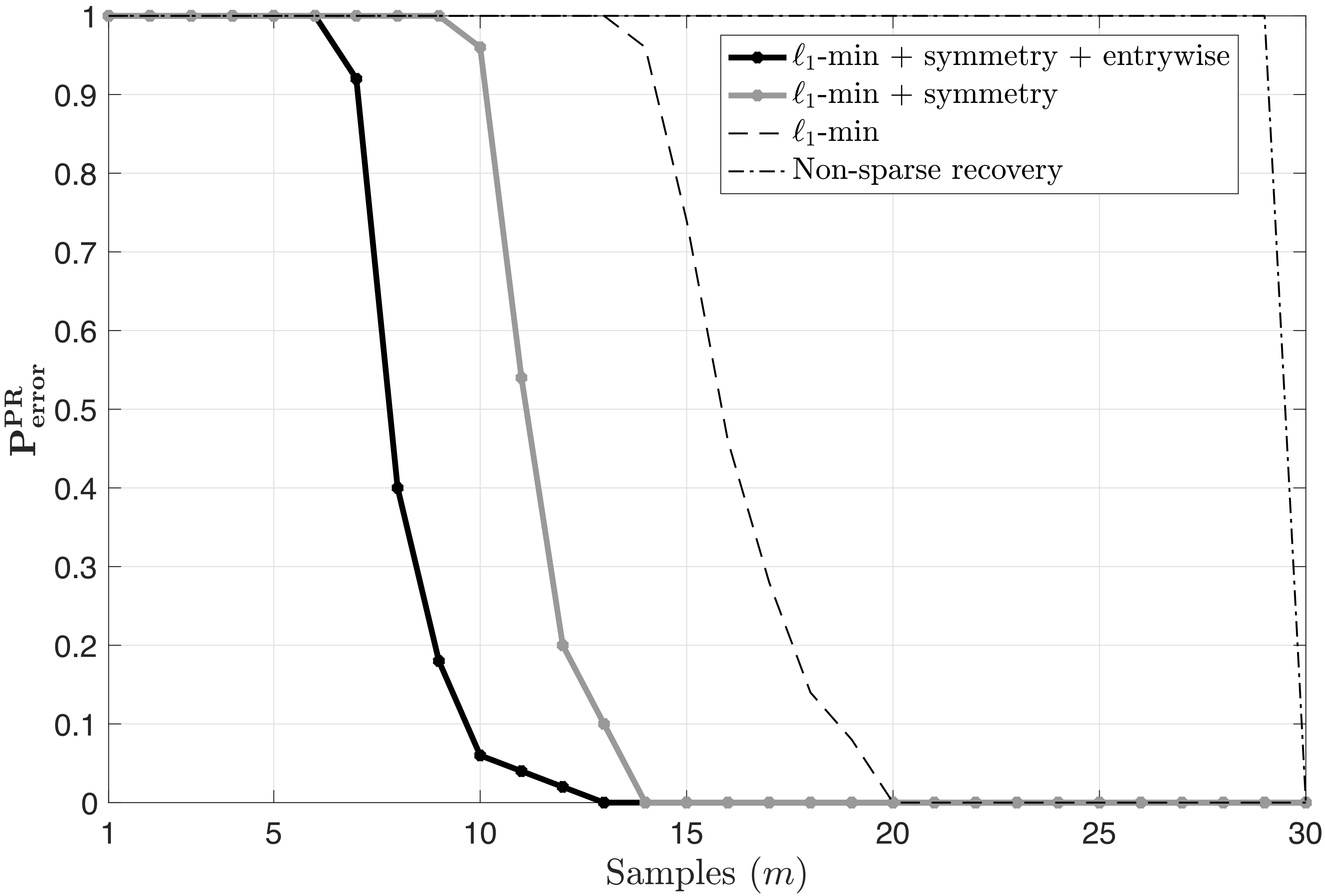
Results



25
20
15
10
5
0
1
50
100
150
200
250
300
Samples (m)
Graph Size (n)
Chains
Stars
IEEE Test Cases
$\Theta(n)$
$\Theta(\log n)$

Results

- 30-bus IEEE test network



Conclusion

- Learn graph structure and parameters from linear measurements
- Sub-linear sample complexity, even from graphs with high-degree nodes (dense columns in Laplacian)
- Application to power system topology/parameter identification problem
- Further work:
  - Robustness to noise
  - Partial measurements (only at a subset of nodes)
- Merci!