Structure Learning in Infrastructure Networks

Rajasekhar Anguluri

Department of Computer Science and Electrical Engineering

University of Maryland, Baltimore County (UMBC)

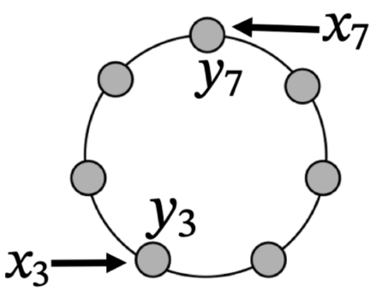
rajangul@umbc.edu

slides template inspiration: Joel Tropp, Caltech

https://rajanguluri.github.io

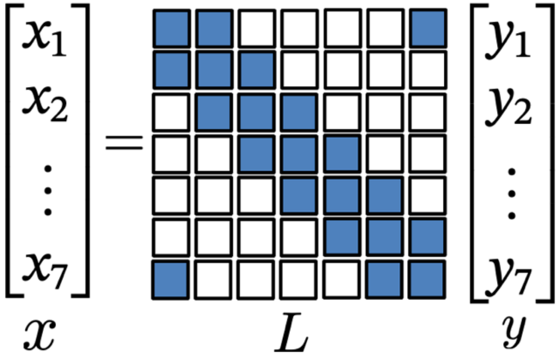
Structure Learning Problems: Recap

Network Structure = Laplacian's Sparsity Pattern



infrastructure network

sparsity (zero & non-zero) of L captures network connections



- ***measurables:*** p-dim vectors x and y
- ***full coverage:*** access x or/and y
- ***partial coverage:*** sub-vectors of x or/and y

- ***linear model:***

$$\mathrm{Vec}(X) = H(Y)\,\mathrm{Ve}(L) + \mathrm{Vec}(E) \quad \text{full coverage}$$

- ***covariance models:***

$$\Omega = L\Omega_x L \qquad \text{full coverage}$$

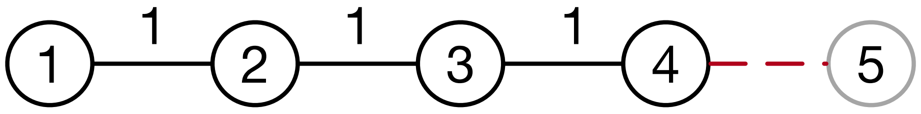
$$\Omega_{OO} = K_{OO} - K_{OH}K_{HH}^{-1}K_{HO} \qquad \text{partial}$$

- ***Estimation:***

1. estimate the vector $\mathrm{Ve}(L)$ from data
2. estimate matrices Ω *and* Ω_{OO} from data

Infrastructure Network → Graphical Model

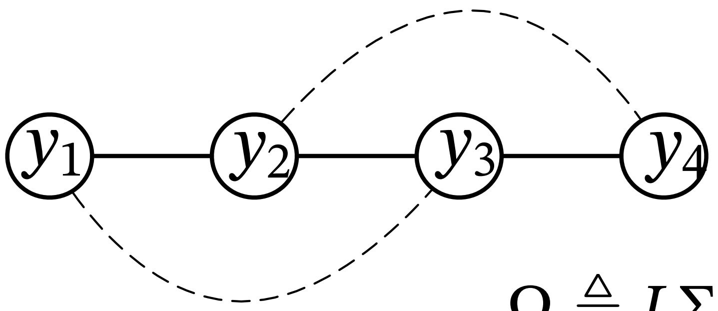
infrastructure network



$$L_{\text{org}} = \left[\begin{array}{cccc|c} 1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ \hline 0 & 0 & 0 & -1 & 1 \end{array}\right]$$

$$L = \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{bmatrix}$$

(two-hop) graphical model



$$\Omega \triangleq L\Sigma_x^{-1}L$$

$$\Omega = L^2 = \begin{bmatrix} 2 & -3 & 1 & 0 \\ -3 & 6 & -4 & 1 \\ 1 & -4 & 6 & -4 \\ 0 & 1 & -4 & 5 \end{bmatrix}$$

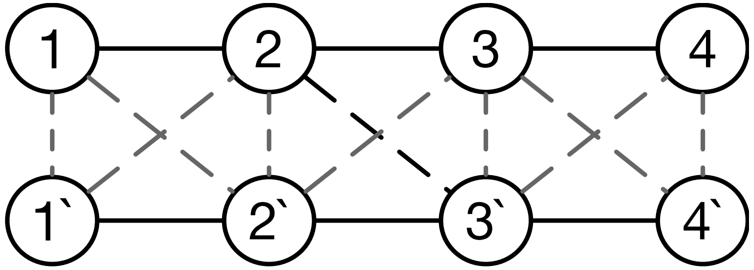
❧ +/-ve sign pattern is crucial for reconstruction

Exercises: Prove the following

- **Theorem 1** (real-case [1]) Graphical model for node potentials in an infrastructure network includes edges between potentials at neighbors and two-hop neighbors.

- **Theorem 2** (complex-lifted-case [1]) Graphical model for real and imaginary potentials in infrastructure network includes edges between real and imaginary potentials (i) at the self nodes; (ii) neighbors; and (iii) two-hop neighbors

$$\begin{bmatrix} y_R(k) \\ y_I(k) \end{bmatrix} = \underbrace{\begin{bmatrix} L_{RR} & L_{RI} \\ L_{IR} & L_{II} \end{bmatrix}}_{L_{\text{real-imag}}}^{-1} \begin{bmatrix} x_R(k) \\ x_I(k) \end{bmatrix}$$



- edge-connectivity by $L_{\text{real-imag}}$: for every node associate an imaginary node (e.g., 1’ for 1) – this is not a graphical model
- draw the connections based on the sparsity pattern of $L_{\text{real-imag}}$ (try this for line graph)

[1] D. Deka et. al. (2023) Graphical models in meshed distribution grids: topology estimation, change detection, and limitati ons, IEEE Trans on Smart Grid, 11 (5)

Graphical Model → Infrastructure Network

- *sign-based algorithms:* consider the infrastructure network $\mathcal{G}$ with *minimum cycle length* greater than three. Then show that algorithms below work (exercise)

network reconstruction : real-case [1]

input: inverse covariance Ω and threshold $\tau > 0$

output: graph $\mathcal{G}$

1. for all nodes i and j do
2. if $\Omega(i,j) < -\tau$ then
3. insert edge (i,j) in $\mathcal{G}$
4. end if
5. end for

network reconstruction : complex-lift-case [1]

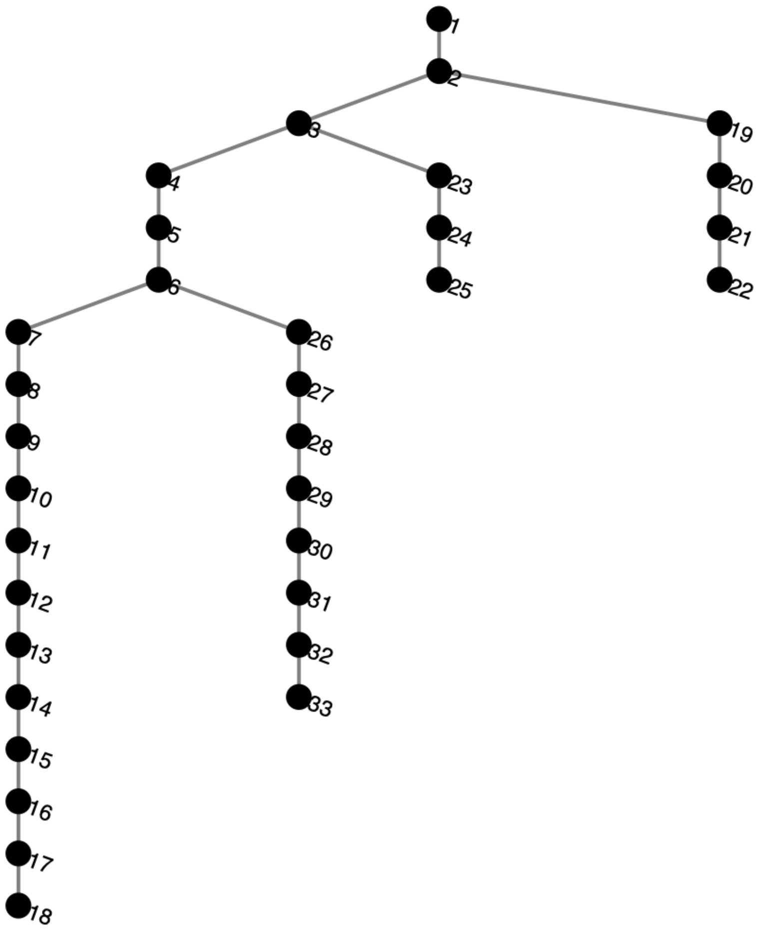
input: block matrix $\Omega_{(\mathrm{R,I})} = \begin{bmatrix} J_{RR} & J_{RI} \\ J_{IR} & J_{II} \end{bmatrix}$ and $\tau > 0$,

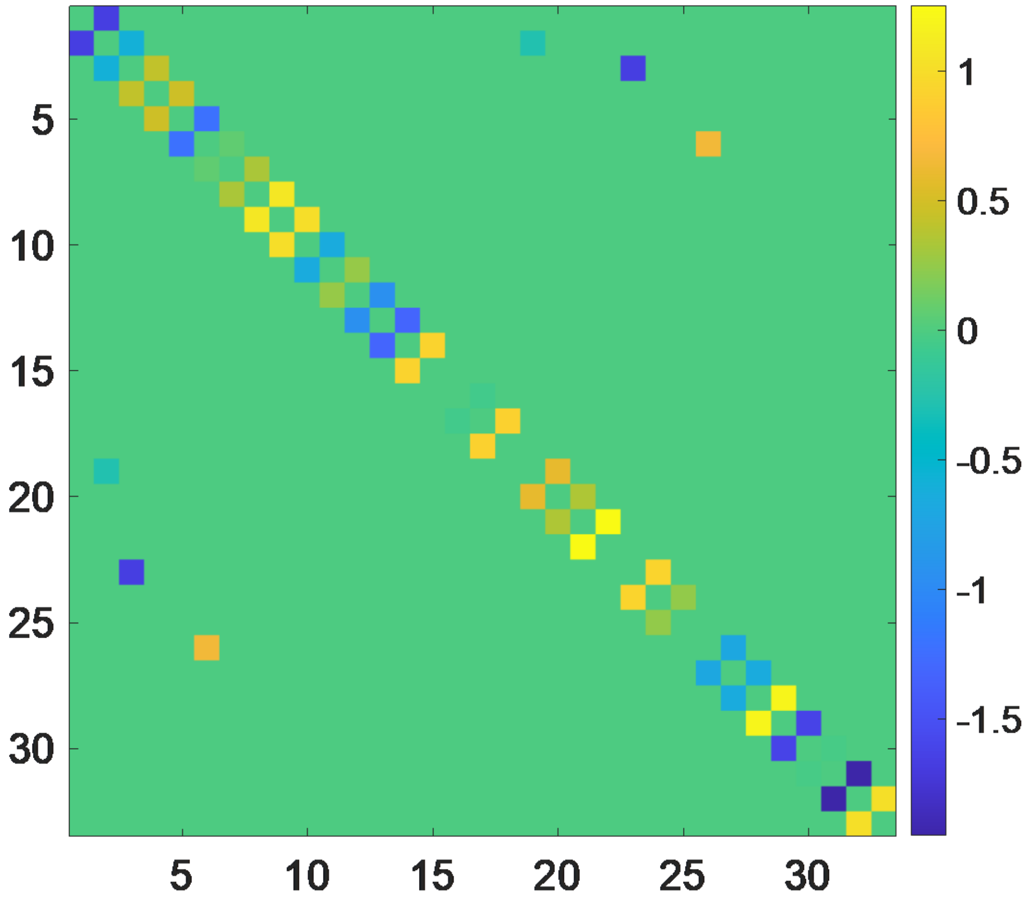
output: graph $\mathcal{G}$

1. for all nodes i and j do
2. if $\mathrm{J}_{RR}(i,j) + \mathrm{J}_{II}(i,j) < -\tau$ then
3. insert edge (i,j) in $\mathcal{G}$
4. end if
5. end for

[1] D. Deka et. al. (2023) Graphical models in meshed distribution grids: topology estimation, change detection, and limitati ons, IEEE Trans on Smart Grid, 11 (5)

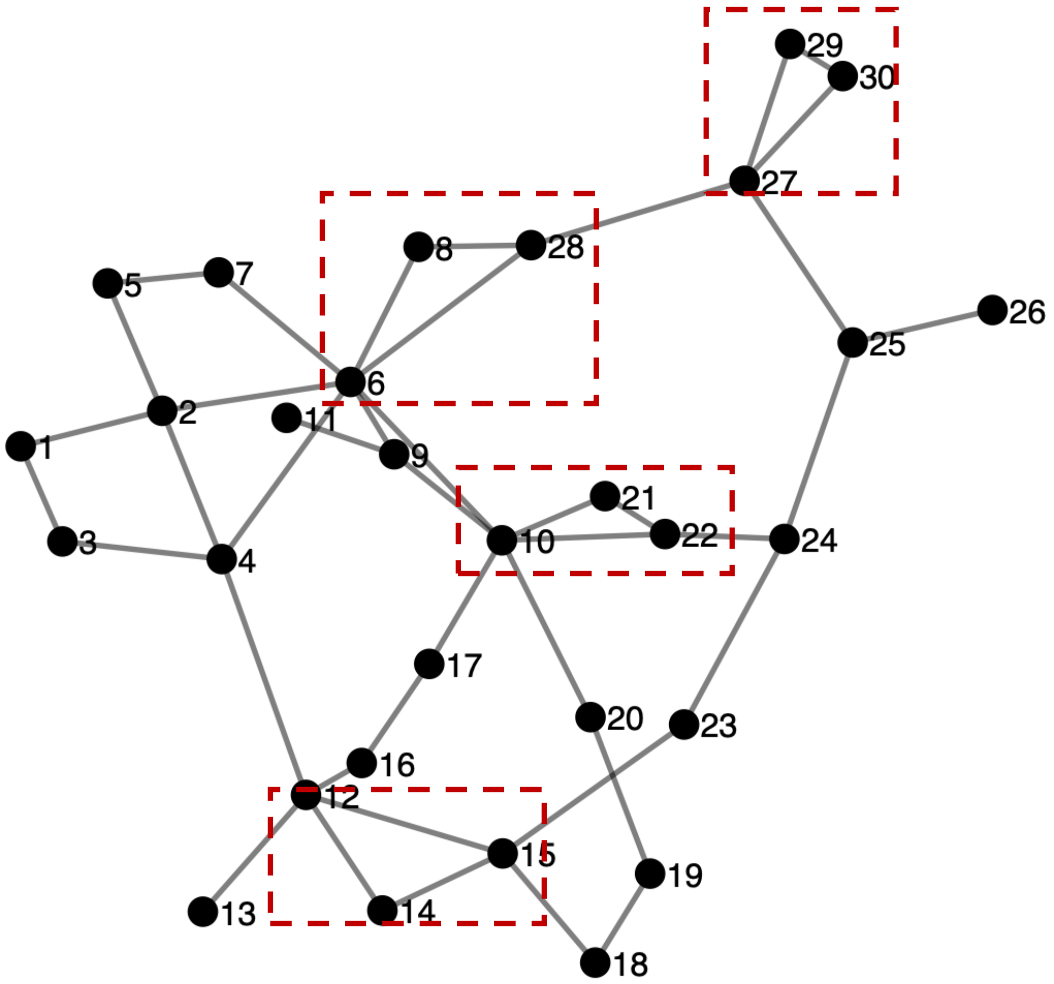
IEEE 33 BUS Distribution N/W Schematic

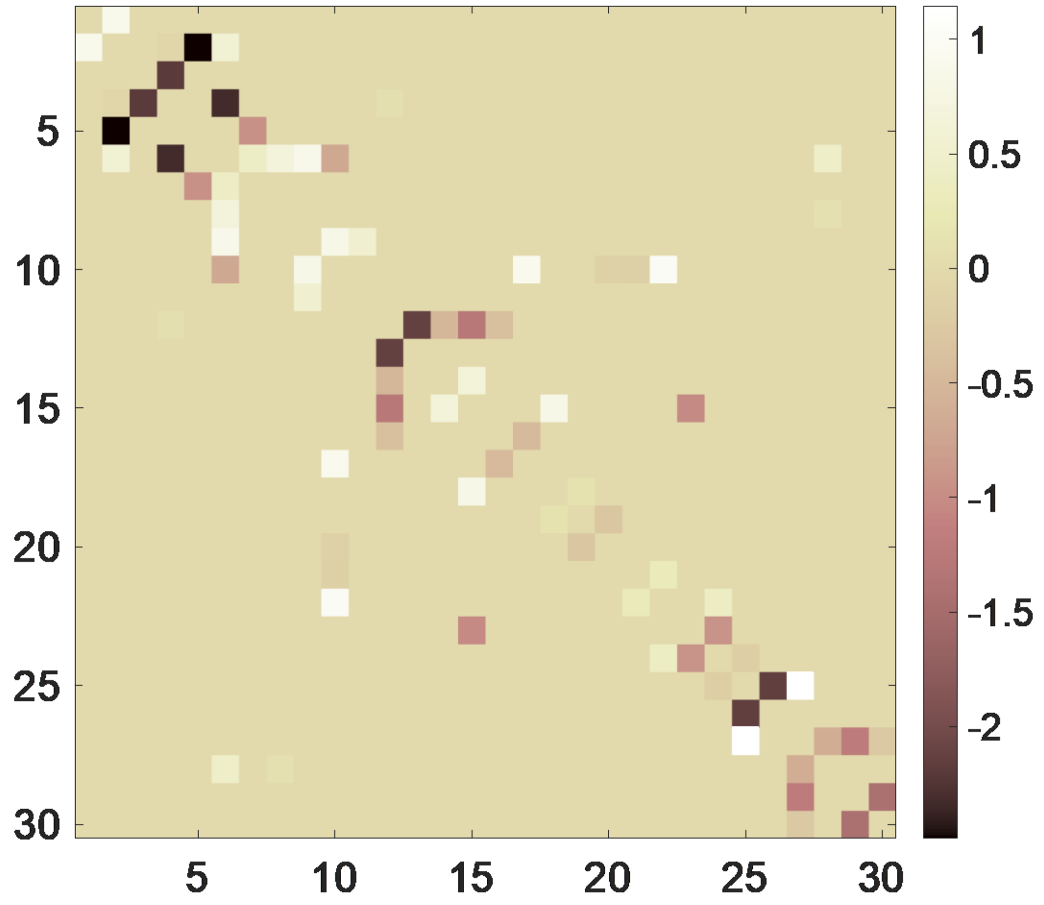




edge weights are scaled for plotting

IEEE 30 BUS Transmission N/W Schematic





edge weights are scaled for plotting

ADMM: General Recipe (vector variable)

- general problem form (with f, g convex):

$$\begin{aligned} \text{minimize}_{x,z} \quad & f(x) + g(z) \\ \text{subject to} \quad & Ax + Bz = c \end{aligned}$$

- $L_\rho(x, y, z) = f(x) + g(z) + v^T(Ax + Bz - c) + (\rho/2)||Ax + Bz - c||_2^2$

- ADMM:

$$x^{k+1} := \text{argmin}_x L_\rho\left(x, z^k, y^k\right)$$ // x- minimization

$$z^{k+1} := \text{argmin}_z L_\rho\left(x^{k+1}, z, y^k\right)$$ // z- minimization

$$v^{k+1} := v^k + \rho\left(Ax^{k+1} + Bz^{k+1} - c\right)$$ // multiplier update

Source: https://stanford.edu/~boyd/admm.html

ADMM for Sparse Inverse Covariance Matrix

- MLE (minimization) problem:

$$\text{minimize} \quad \operatorname{Tr}(S\Omega) - \log\det(\Omega) + \lambda\|\Omega\|_1$$

- ADMM form:

$$\begin{array}{ll} \text{minimize} & \operatorname{Tr}(S\Omega) - \log\det(\Omega) + \lambda\|Z\|_1 \\ \text{subject to} & \Omega - Z = 0 \end{array}$$

- ADMM (scaled):

$$\Omega^{k+1} := \underset{\Omega}{\operatorname{argmin}} \left(\operatorname{Tr}(S\Omega) - \log\det\Omega + (\rho/2)\left\|\Omega - Z^k + U^k\right\|_F^2 \right)$$ // X- minimization

$$Z^{k+1} := S\left(\Omega^{k+1} + U^k, \lambda/\rho\right)$$ // soft thresholding

$$U^{k+1} := U^k + (\Omega^{k+1} - Z^{k+1})$$ // multiplier update

Source: https://stanford.edu/~boyd/admm.html