Robust Network Topology Inference

Santiago Segarra
Institute for Data, Systems, and Society
Massachusetts Institute of Technology
segarra@mit.edu
http://www.mit.edu/~segarra/

Co-authors: Antonio G. Marques, Gonzalo Mateos, and Alejandro Ribeiro

ICASSP, March 9, 2017
Desiderata: Process, analyze and learn from network data [Kolaczyk’09]
Desiderata: Process, analyze and learn from network data [Kolaczyk’09]

Network as graph $G = (\mathcal{V}, \mathcal{E})$: encode pairwise relationships

Interest here not in $G$ itself, but in data associated with nodes in $\mathcal{V}$

$\Rightarrow$ Object of study is a graph signal

$\Rightarrow$ As.: Signal properties related to topology of $G$ (e.g., smoothness)
Graph signal processing (GSP)

- Undirected $G$ with adjacency matrix $A$
  $\Rightarrow A_{ij} =$ Proximity between $i$ and $j$
- Define a signal $x$ on top of the graph
  $\Rightarrow x_i =$ Signal value at node $i$
Graph signal processing (GSP)

- Undirected $G$ with adjacency matrix $A$
  \[ A_{ij} = \text{Proximity between } i \text{ and } j \]
- Define a signal $x$ on top of the graph
  \[ x_i = \text{Signal value at node } i \]

- Associated with $G$ is the graph-shift operator $S = V\Lambda V^T \in \mathcal{M}^N$
  \[ S_{ij} = 0 \text{ for } i \neq j \text{ and } (i, j) \notin E \text{ (local structure in } G) \]
  \[ \text{Ex: } A, \text{ degree } D \text{ and Laplacian } L = D - A \text{ matrices} \]
Graph signal processing (GSP)

- Undirected $G$ with adjacency matrix $A$
  \[ A_{ij} = \text{Proximity between } i \text{ and } j \]
- Define a signal $x$ on top of the graph
  \[ x_i = \text{Signal value at node } i \]

- Associated with $G$ is the graph-shift operator $S = V\Lambda V^T \in \mathcal{M}^N$
  \[ S_{ij} = 0 \text{ for } i \neq j \text{ and } (i, j) \notin \mathcal{E} \text{ (local structure in } G) \]
  \[ \text{Ex: } A, \text{ degree } D \text{ and Laplacian } L = D - A \text{ matrices} \]

- Graph Signal Processing $\rightarrow$ Exploit structure encoded in $S$ to process $x$
  \[ \text{Our view: GSP well suited to study (network) diffusion processes} \]
Motivation and context

- Network topology inference from nodal observations [Kolaczyk’09]
  - Approaches use Pearson correlations to construct graphs [Brovelli04]
  - Partial correlations and conditional dependence [Friedman08, Karanikolas16]

- Key in neuroscience [Sporns’10]
  - Functional net inferred from activity
Network topology inference from nodal observations [Kolaczyk’09]
⇒ Approaches use Pearson correlations to construct graphs [Brovelli04]
⇒ Partial correlations and conditional dependence [Friedman08, Karanikolas16]

Key in neuroscience [Sporns’10]
⇒ Functional net inferred from activity

Most GSP works: How known graph $S$ affects signals and filters

Here, reverse path: How to use GSP to infer the graph topology?
⇒ Gaussian graphical models [Egilmez16]
⇒ Smooth signals [Dong15], [Kalofolias16]
⇒ Stationary signals [Segarra16], [Pasdeloup16]
⇒ Directed graphs [Mei-Moura15], [Shen16]

Today’s talk: Guarantees of robustness in topology inference
Our approach for topology inference

- We propose a two-step approach for graph topology identification

**STEP 1:**
Identify the eigenvectors of the shift

**STEP 2:**
Identify eigenvalues to obtain a suitable shift

- Alternative sources for spectral templates $V$
  - Design of graph filters [Segarra et al’15]
  - Graph sparsification and Network deconvolution [Feizi et al’13]

- Small number of $\{x_i\}$ or specific signal features
  $\Rightarrow$ May lead to noisy or incomplete eigenvectors $\hat{V}$

- How good is the recovery of $S$ when $\hat{V}$ (instead of $V$) is available?
Step 1: Obtaining the eigenvectors

- \( \mathbf{x} \) is a \textbf{stationary process} on the unknown graph \( \mathbf{S} \)
  - \( \Rightarrow \) Observed \( \{\mathbf{x}_i\} \) are random realizations of \( \mathbf{x} \)
  - \( \Rightarrow \) Eigenvectors \( \mathbf{V} \) can be recovered from covariance \( \mathbf{C}_\mathbf{x} \)

- Signal \( \mathbf{x} \) is the response of a linear diffusion process to a white input

\[
\mathbf{x} = \alpha_0 \prod_{l=1}^{\infty} (\mathbf{I} - \alpha_l \mathbf{S}) \mathbf{w} = \sum_{l=0}^{\infty} \beta_l \mathbf{S}^l \mathbf{w} = \left( \sum_{l=0}^{N-1} h_l \mathbf{S}^l \right) \mathbf{w} := \mathbf{H} \mathbf{w}
\]

- Common generative model. Heat diffusion if \( \alpha_k \) constant
- \( \mathbf{H} \) is a \textbf{graph filter} on the unknown graph
- \( \mathbf{H} \) diagonalized by the eigenvectors \( \mathbf{V} \) of the shift operator \( \mathbf{S} \)
Step 1: Obtaining the eigenvectors

- The covariance matrix of the signal $x$ is
  \[ C_x = \mathbb{E} \left[ (Hw(Hw)^H) \right] = H\mathbb{E} \left[ (ww^H) \right] H^H = HH^H \]

- Since $H$ is diagonalized by $V$, so is the covariance $C_x$
  \[ C_x = V \left| \sum_{l=0}^{L-1} h_l \Lambda^l \right|^2 V^H \]

- Any shift with eigenvectors $V$ can explain $x$
  $\Rightarrow$ $G$ and its specific eigenvalues have been obscured by diffusion

Observations

(a) Identifying $S \rightarrow$ Identifying the eigenvalues
(b) Correlation methods $\rightarrow$ Eigenvalues are kept unchanged
(c) Precision methods $\rightarrow$ Eigenvalues are inverted
Step 2: Obtaining the eigenvalues

- We can use extra knowledge/assumptions to choose one graph
  ⇒ Of all graphs, select one that is optimal in some sense

\[
S_*^0 := \arg\min_{S, \lambda} \|S\|_0 \quad \text{s. to} \quad S = \sum_{k=1}^{N} \lambda_k v_k v_k^H, \quad S \in S
\]

- Set \( S \) contains all admissible scaled adjacency matrices

\[
S := \{S \mid S_{ij} \geq 0, \quad S \in \mathcal{M}^N, \quad S_{ii} = 0, \quad \sum_j S_{1j} = 1\}
\]
Step 2: Obtaining the eigenvalues

- We can use extra knowledge/assumptions to choose one graph
  \[ \Rightarrow \text{Of all graphs, select one that is \textit{optimal} in some sense} \]

\[ S_0^* := \arg\min_{S,\lambda} \|S\|_0 \quad \text{s. to} \quad S = \sum_{k=1}^{N} \lambda_k v_k v_k^H, \quad S \in S \]

- Set \( S \) contains all admissible scaled \textit{adjacency} matrices

\[ S := \{S \mid S_{ij} \geq 0, \quad S \in \mathcal{M}^N, \quad S_{ii} = 0, \quad \sum_j S_{1j} = 1\} \]

- Non-convex problem, relax to \( \ell_1 \)-norm minimization, e.g., [Tropp’06]

\[ S_1^* := \arg\min_{S,\lambda} \|S\|_1 \quad \text{s. to} \quad S = \sum_{k=1}^{N} \lambda_k v_k v_k^H, \quad S \in S \]

- What if \( V \) is not available? \[ \Rightarrow \text{Noisy and/or incomplete } \hat{V} \]
Robust shift identification

- Two-step algorithm based on perfect **spectral templates**
  - However, perfect knowledge of $V$ may not be available
  - Robust designs?

**Q1:** How to modify the optimization in step 2?
  - Distance for noise, orthogonal subspace for incomplete

**Q2:** Recovery guarantees?
Incomplete spectral templates

- Partial access to $V \Rightarrow$ Only $K$ known eigenvectors $[v_1, \ldots, v_K]$

$$\min_{\{S, S_{\tilde{K}}, \lambda\}} \|S\|_1 \text{ s. to } S = S_{\tilde{K}} + \sum_{k=1}^{K} \lambda_k v_k v_k^H, \quad S \in S, \quad S_{\tilde{K}} v_k = 0$$

- How does the (partial) knowledge of $V_K$ affect the recovery?
Incomplete spectral templates

- Partial access to $\mathbf{V} \Rightarrow$ Only $K$ known eigenvectors $[v_1, \ldots, v_K]$

$$\min_{\{S, S_{\bar{K}}, \lambda\}} \|S\|_1 \text{ s. to } S = S_{\bar{K}} + \sum_{k=1}^{K} \lambda_k v_k v_k^H, \quad S \in S, \quad S_{\bar{K}} v_k = 0$$

- How does the (partial) knowledge of $\mathbf{V}_K$ affect the recovery?

- Define $\mathbf{P} := [P_1, P_2]$ in terms of $\mathbf{V}_K$, and $\Upsilon := [I_{N^2}, 0_{N^2 \times N^2}]$

$\Rightarrow$ Goal is to reformulate problem as $\min_t \|\Upsilon t\|_1 \text{ s.to} \mathbf{P}^T t = \mathbf{b}$

S$^*$ and $S_0^*$ coincide if the two following conditions are satisfied:
1) $\text{rank}([P_1^T, P_2^T]) = |\mathcal{K}| + N^2$; and
2) There exists a constant $\delta > 0$ such that

$$\eta_{\mathbf{P}} := \|\Upsilon_{\mathcal{K}c}(\delta^{-2}\mathbf{P}\mathbf{P}^T + \Upsilon_{\mathcal{K}c}^T \Upsilon_{\mathcal{K}c})^{-1}\Upsilon_{\mathcal{K}c}^T\|_\infty < 1.$$  

- Cond. 1) ensures uniqueness of solution $S^*$

- Cond. 2) guarantees existence of a dual certificate for $\ell_0$ optimality
We might have access to \( \hat{V} \), a noisy version of the spectral templates.

\[
\Rightarrow \text{With } d(\cdot, \cdot) \text{ denoting a (convex) distance between matrices }
\]

\[
\min_{\{S, \lambda, \hat{S}\}} \|S\|_1 \quad \text{s. to } \hat{S} = \sum_{k=1}^{N} \lambda_k \hat{v}_k \hat{v}_k^H, \quad S \in S, \quad d(S, \hat{S}) \leq \epsilon
\]

How does the noise in \( \hat{V} \) affect the recovery?
We might have access to $\hat{V}$, a noisy version of the spectral templates. With $d(\cdot, \cdot)$ denoting a (convex) distance between matrices,

$$\min_{\{S, \lambda, \hat{S}\}} \|S\|_1 \quad \text{s. to} \quad \hat{S} = \sum_{k=1}^N \lambda_k \hat{v}_k \hat{v}_k^H, \quad S \in S, \quad d(S, \hat{S}) \leq \epsilon$$

How does the noise in $\hat{V}$ affect the recovery?

Stable recovery can be established $\Rightarrow$ depends on noise level

$\Rightarrow$ Reformulate problem as $\min_t \|t\|_1 \quad \text{s. to} \quad \|R^T t - b\|_2 \leq \epsilon$

Conditions 1) and 2) but based on $R$, guaranteed $d(S^*, S_0^*) \leq C\epsilon$

$\Rightarrow \epsilon$ large enough to guarantee feasibility of $S_0^*$

$\Rightarrow$ Constant $C$ depends on $\hat{V}$ and the support $K$
Social graphs from imperfect templates

- Identification of multiple social networks $N = 32$
  - Defined on the same node set of students from Ljubljana
  - Synthetic signals from diffusion processes in the graphs
- Recovery for incomplete (left) and noisy (right) spectral templates

- Error (left) decreases with increasing nr. of spectral templates
- Error (right) decreases with increasing number of observed signals
Performance comparisons

- Comparison with graphical lasso and sparse correlation methods
  - Evaluated on 100 realizations of ER graphs with $N = 20$ and $p = 0.2$

Graphical lasso implicitly assumes a filter $H_1 = (\rho I + S)^{-1/2}$

$\Rightarrow$ For this filter spectral templates work, but not as well

- For general diffusion filters $H_2$ spectral templates still work fine
Inferring direct relations

- Our method can be used to sparsify a given network
  - Keep direct and important edges or relations
  - Discard indirect relations that can be explained by direct ones

- Use eigenvectors $\hat{V}$ of given network as noisy templates

Ex: Infer contact between amino-acid residues in BPT1 BOVIN
  - Use mutual information of amino-acid covariation as input

- Network deconvolution assumes a specific filter model [Feizi et al’13]
  - We achieve better performance by being agnostic to this
Network topology inference cornerstone problem in Network Science
- Most GSP works analyze how $S$ affect signals and filters
- Here, reverse path: How to use GSP to infer the graph topology?

Our GSP approach to network topology inference
- Two step approach: i) Obtain $V$; ii) Estimate $S$ given $V$
Network topology inference cornerstone problem in Network Science
- Most GSP works analyze how $S$ affect signals and filters
- Here, reverse path: How to use GSP to infer the graph topology?

Our GSP approach to network topology inference
- Two step approach: i) Obtain $V$; ii) Estimate $S$ given $V$

How to obtain the spectral templates $V$
- Based on covariance of stationary signals
- Other sources: network operators, network deconvolution
Network topology inference cornerstone problem in Network Science
- Most GSP works analyze how $S$ affect signals and filters
- Here, reverse path: How to use GSP to infer the graph topology?

Our GSP approach to network topology inference
⇒ Two step approach: i) Obtain $V$; ii) Estimate $S$ given $V$

How to obtain the spectral templates $V$
⇒ Based on covariance of stationary signals
⇒ Other sources: network operators, network deconvolution

Infer $S$ via convex optimization
⇒ Objectives promotes desirable properties
⇒ Constraints encode structure a priori info and structure
⇒ Formulations for noisy and incomplete templates