Robust Graph Filter Identification and Graph Denoising from Signal Observations

Samuel Rey

King Juan Carlos University - Madrid (Spain)

In collaboration with Victor Tenorio, Antonio G. Marques

Graph Signal Processing Workshop 2023 (GSP 2023) - Oxford, United Kingdom - June 12-14, 2023

Why a robust GSP framework?

- Data is becoming heterogeneous and pervasive [Kolaczyk09][Leskovec20]
  - ⇒ Often defined over irregular domains and networks
  - ⇒ More complex structure demands more complex architectures
- **GSP**: models data structure as a graph [Shuman13][Ortega18]
  - ⇒ Leverages the graph topology to process the data

Social network

Brain network

Home automation network

Why a robust GSP framework?

- Data is becoming heterogeneous and pervasive [Kolaczyk09][Leskovec20]
  - ⇒ Often defined over irregular domains and networks
  - ⇒ More complex structure demands more complex architectures
- **GSP**: models data structure as a graph [Shuman13][Ortega18]
  - ⇒ Leverages the graph topology to process the data
- **Problem:** data is prone to errors and imperfections
  - ⇒ Noise, missing values, or outliers are pervasive in data science

Social network

Brain network

Home automation network

- Signal processing deals with perturbations on the signals
  - ⇒ Large perturbations render data useless
  - ⇒ Widely study in several fields

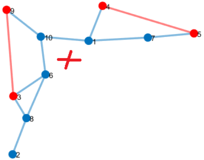
True signal

Noisy signal

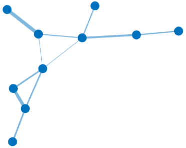
Imperfections in the graph topology

- Signal processing deals with perturbations on the signals
  - ⇒ Large perturbations render data useless
  - ⇒ Widely study in several fields
- In GSP we encounter perturbations in the graph topology
  - ⇒ Even small perturbations lead to challenging problems
  - ⇒ Most GSP methods assume the graph is perfectly known

Original graph

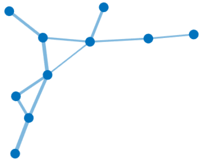


Errors in the support

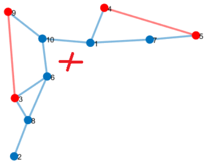


Noisy edges

- Signal processing deals with perturbations on the signals
  - ⇒ Large perturbations render data useless
  - ⇒ Widely study in several fields
- In GSP we encounter perturbations in the graph topology
  - ⇒ Even small perturbations lead to challenging problems
  - ⇒ Most GSP methods assume the graph is perfectly known



Original graph

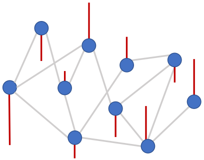


Errors in the support

Noisy edges

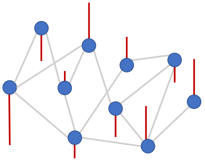
- **This work**: approach the graph FI accounting for topology imperfections

- Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with N nodes and adjacency $\mathbf{A}$
  - $\Rightarrow A_{ij}$ = Proximity between i and j



- Define a signal $\mathbf{x} \in \mathbb{R}^N$ on top of the graph
  - $\Rightarrow x_i$ = Signal value at node i

- Associated with $\mathcal{G}$ is the graph-shift operator $\mathbf{S} \in \mathbb{R}^{N \times N}$ (e.g. $\mathbf{A}$, $\mathbf{L}$)
  - $\Rightarrow S_{ij} \neq 0$ if $i = j$ or $(i, j) \in \mathcal{E}$ (local structure in $\mathcal{G}$)[Shuman13][Sandryhaila13]
  - $\Rightarrow$ Diagonalized as $\mathbf{S} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1}$



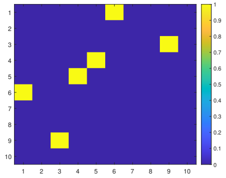
- Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with N nodes and adjacency $\mathbf{A}$
  - $\Rightarrow A_{ij}$ = Proximity between i and j
- Define a signal $\mathbf{x} \in \mathbb{R}^N$ on top of the graph
  - $\Rightarrow x_i$ = Signal value at node i
- Associated with $\mathcal{G}$ is the graph-shift operator $\mathbf{S} \in \mathbb{R}^{N \times N}$ (e.g. $\mathbf{A}$, $\mathbf{L}$)
  - $\Rightarrow S_{ij} \neq 0$ if $i = j$ or $(i, j) \in \mathcal{E}$ (local structure in $\mathcal{G}$)[Shuman13][Sandryhaila13]
  - $\Rightarrow$ Diagonalized as $\mathbf{S} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1}$
- Graph filters are defined as $\mathbf{H} = \sum_{r=0}^{R-1} h_r \mathbf{S}^r$ [Segarra17]
  - $\Rightarrow$ Diagonalized as $\mathbf{H} = \mathbf{V}\text{diag}(\tilde{\mathbf{h}})\mathbf{V}^{-1}$
  - $\Rightarrow \mathbf{S}^r$ encodes r-hop neighborhood so $\mathbf{H}\mathbf{x}$ diffuses $\mathbf{x}$ across $\mathcal{G}$

GF ID and influence of perturbations

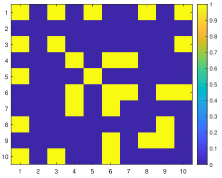
- **GF identification:** estimate the graph filter $\mathbf{H} = \sum_{r=0}^{R-1} h_r \mathbf{S}^r$
  - $\Rightarrow$ Given input/output signals $\mathbf{X}/\mathbf{Y} \in \mathbb{R}^{N \times M}$ with $\mathbf{Y} = \mathbf{H}\mathbf{X} + \mathbf{W}$
  - $\Rightarrow$ Leveraging that $\mathbf{H}$ is a polynomial of the GSO

GF ID and influence of perturbations

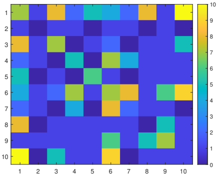
- **GF identification:** estimate the graph filter $\mathbf{H} = \sum_{r=0}^{R-1} h_r \mathbf{S}^r$
  - $\Rightarrow$ Given input/output signals $\mathbf{X}/\mathbf{Y} \in \mathbb{R}^{N \times M}$ with $\mathbf{Y} = \mathbf{H}\mathbf{X} + \mathbf{W}$
  - $\Rightarrow$ Leveraging that $\mathbf{H}$ is a polynomial of the GSO
- Due to perturbations the true $\mathbf{S}$ is **unknown**
  - $\Rightarrow$ Only perturbed $\bar{\mathbf{S}} \in \mathbb{R}^{N \times N}$ is observed
- **Q:** What if we estimate the filter as $\mathbf{H} = \sum_{r=0}^{R-1} h_r \bar{\mathbf{S}}^r$?
  - $\Rightarrow$ Error between $\mathbf{S}^r$ and $\bar{\mathbf{S}}^r$ grows with r



$|\mathbf{S} - \bar{\mathbf{S}}|$



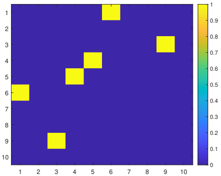
$|\mathbf{S}^2 - \bar{\mathbf{S}}^2|$



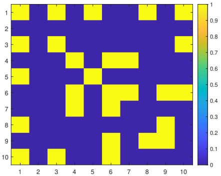
$|\mathbf{S}^4 - \bar{\mathbf{S}}^4|$

GF ID and influence of perturbations

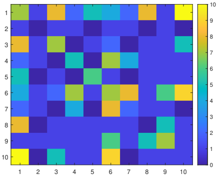
- **GF identification:** estimate the graph filter $\mathbf{H} = \sum_{r=0}^{R-1} h_r \mathbf{S}^r$
  - $\Rightarrow$ Given input/output signals $\mathbf{X}/\mathbf{Y} \in \mathbb{R}^{N \times M}$ with $\mathbf{Y} = \mathbf{HX} + \mathbf{W}$
  - $\Rightarrow$ Leveraging that $\mathbf{H}$ is a polynomial of the GSO
- Due to perturbations the true $\mathbf{S}$ is **unknown**
  - $\Rightarrow$ Only perturbed $\bar{\mathbf{S}} \in \mathbb{R}^{N \times N}$ is observed
- **Q:** What if we estimate the filter as $\mathbf{H} = \sum_{r=0}^{R-1} h_r \bar{\mathbf{S}}^r$?
  - $\Rightarrow$ Error between $\mathbf{S}^r$ and $\bar{\mathbf{S}}^r$ grows with r
- **A:** estimating $\mathbf{H}$ as polynomial of $\bar{\mathbf{S}}$ results in **high estimation error**



$|\mathbf{S} - \bar{\mathbf{S}}|$



$|\mathbf{S}^2 - \bar{\mathbf{S}}^2|$



$|\mathbf{S}^4 - \bar{\mathbf{S}}^4|$

Modeling graph perturbations

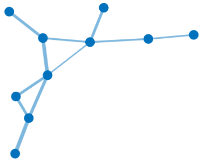
- Additive perturbation models are pervasive in SP $\Rightarrow$ In graphs $\bar{\mathbf{S}} = \mathbf{S} + \boldsymbol{\Delta}$
  - $\Rightarrow$ Structure of $\boldsymbol{\Delta} \in \mathbb{R}^{N \times N}$ depends on the type of perturbation
  - $\Rightarrow$ $\mathbf{S}$ and $\bar{\mathbf{S}}$ are close according to some metric $d(\mathbf{S}, \bar{\mathbf{S}})$

Modeling graph perturbations

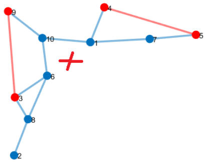
- Additive perturbation models are pervasive in SP $\Rightarrow$ In graphs $\bar{\mathbf{S}} = \mathbf{S} + \boldsymbol{\Delta}$
  - $\Rightarrow$ Structure of $\boldsymbol{\Delta} \in \mathbb{R}^{N \times N}$ depends on the type of perturbation
  - $\Rightarrow$ $\mathbf{S}$ and $\bar{\mathbf{S}}$ are close according to some metric $d(\mathbf{S}, \bar{\mathbf{S}})$

Examples of topology perturbations

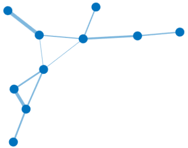
- When perturbations **create/destroy edges** $\Longrightarrow$ $d(\mathbf{S}, \bar{\mathbf{S}}) = \|\mathbf{S} - \bar{\mathbf{S}}\|_0$
  - $\Rightarrow$ $\Delta_{ij} = 1$ if $S_{ij} = 0$ and $\Delta_{ij} = -1$ if $S_{ij} = 1$
- When perturbations represent **noisy edges** $\Longrightarrow$ $d(\mathbf{S}, \bar{\mathbf{S}}) = \|\mathbf{S}_{\mathcal{E}} - \bar{\mathbf{S}}_{\mathcal{E}}\|_2^2$
  - $\Rightarrow$ $\Delta_{ij} = 0$ if $S_{ij} = 0$ and $\Delta_{ij} \sim \mathcal{N}(0, \sigma^2)$ if $S_{ij} \neq 0$



Original graph



Create/Destroy edges



Noisy edges

Traditional filter identification (FI)

- Consider formulation in either vertex or frequency domain

$$\min_{\mathbf{h}} \|\mathbf{Y} - \sum_{k=0}^{N-1} h_k \mathbf{S}^k \mathbf{X}\|_F^2 \qquad \min_{\tilde{\mathbf{h}}} \|\mathbf{Y} - \mathbf{V}\text{diag}(\tilde{\mathbf{h}})\mathbf{V}^\top \mathbf{X}\|_F^2$$

Universidad Rey Juan Carlos

Traditional filter identification (FI)

- Consider formulation in either vertex or frequency domain

$$\min_{\mathbf{h},\mathbf{S}} \|\mathbf{Y} - \sum_{k=0}^{N-1} h_k \mathbf{S}^k \mathbf{X}\|_F^2 \qquad \min_{\tilde{\mathbf{h}},\mathbf{V}} \|\mathbf{Y} - \mathbf{V}\text{diag}(\tilde{\mathbf{h}})\mathbf{V}^\top \mathbf{X}\|_F^2$$

$$\text{s. t.} \;\; \mathbf{S} \in \mathcal{S} \qquad\qquad \text{s. t.} \;\; \mathbf{V}\mathbf{V}^\top = \mathbf{I}$$

$\Rightarrow$ Modeling influence of perturbations in $\mathbf{S}^k$ and $\mathbf{V}$ is non-trivial

RFI as an optimization problem

Traditional filter identification (FI)

- Consider formulation in either vertex or frequency domain

$$\min_{\mathbf{h},\mathbf{S}} \|\mathbf{Y} - \sum_{k=0}^{N-1} h_k \mathbf{S}^k \mathbf{X}\|_F^2 \quad \text{s. t.} \quad \mathbf{S} \in \mathcal{S}$$

$$\min_{\tilde{\mathbf{h}},\mathbf{V}} \|\mathbf{Y} - \mathbf{V}\text{diag}(\tilde{\mathbf{h}})\mathbf{V}^\top \mathbf{X}\|_F^2 \quad \text{s. t.} \quad \mathbf{V}\mathbf{V}^\top = \mathbf{I}$$

$\Rightarrow$ Modeling influence of perturbations in $\mathbf{S}^k$ and $\mathbf{V}$ is non-trivial

Robust filter identification (RFI)

- Define full $\mathbf{H}$ as an optimization variable and jointly estimate $\mathbf{H}$ and $\mathbf{S}$

$$\min_{\mathbf{S}\in\mathcal{S},\mathbf{H}} \|\mathbf{Y} - \mathbf{H}\mathbf{X}\|_F^2 + \lambda d(\mathbf{S}, \bar{\mathbf{S}}) + \beta\|\mathbf{S}\|_0 \quad \text{s. t.} \quad \mathbf{S}\mathbf{H} = \mathbf{H}\mathbf{S}$$

$\Rightarrow$ The constraint captures the fact that $\mathbf{H}$ is a polynomial of $\mathbf{S}$

$\Rightarrow$ Second term promotes closeness between $\bar{\mathbf{S}}$ and $\mathbf{S}$

- Operates in vertex domains + avoids computation of high-order polynomials
- Bilinear terms and ℓ_0 render the problem non-convex

Universidad Rey Juan Carlos

Dealing with ℓ_0 norm

- We employ the ℓ_1 reweighted norm based on logarithmic penalty [Candes08]

$$\|\mathbf{Z}\|_0 \approx r_\delta(\mathbf{Z}) := \sum_{i=1}^{I}\sum_{j=1}^{J} \log(|Z_{ij}| + \delta)$$

  $\Rightarrow$ Produces sparser solutions than ℓ_1 norm

  $\Rightarrow$ Majorization-Minimization approach based on linear approximation

Dealing with ℓ_0 norm

- We employ the ℓ_1 reweighted norm based on logarithmic penalty [Candes08]

$$\|\mathbf{Z}\|_0 \approx r_\delta(\mathbf{Z}) := \sum_{i=1}^{I} \sum_{j=1}^{J} \log(|Z_{ij}| + \delta)$$

$\Rightarrow$ Produces sparser solutions than ℓ_1 norm

$\Rightarrow$ Majorization-Minimization approach based on linear approximation

Dealing with bilinear term

- Adopt an alternating-minimization approach to break the non-linearity

$\Rightarrow$ $\mathbf{H}$ and $\mathbf{S}$ are estimated in two separate iterative steps

$\Rightarrow$ Each step requires solving a convex optimization problem

Dealing with ℓ_0 norm

- We employ the ℓ_1 reweighted norm based on logarithmic penalty [Candes08]

$$\|\mathbf{Z}\|_0 \approx r_\delta(\mathbf{Z}) := \sum_{i=1}^{I}\sum_{j=1}^{J} \log(|Z_{ij}| + \delta)$$

$\Rightarrow$ Produces sparser solutions than ℓ_1 norm

$\Rightarrow$ Majorization-Minimization approach based on linear approximation

Dealing with bilinear term

- Adopt an alternating-minimization approach to break the non-linearity

$\Rightarrow$ $\mathbf{H}$ and $\mathbf{S}$ are estimated in two separate iterative steps

$\Rightarrow$ Each step requires solving a convex optimization problem

- Rewrite optimization problem as

$$\min_{\mathbf{S}\in\mathcal{S},\mathbf{H}} \|\mathbf{Y}-\mathbf{H}\mathbf{X}\|_F^2 + \lambda r_{\delta_1}(\mathbf{S}-\bar{\mathbf{S}}) + \beta r_{\delta_2}(\mathbf{S}) + \gamma\|\mathbf{S}\mathbf{H}-\mathbf{H}\mathbf{S}\|_F^2$$

$\Rightarrow$ Constraint $\mathbf{S}\mathbf{H} = \mathbf{H}\mathbf{S}$ relaxed as a regularizer

Alternating optimization algorithm

Universidad Rey Juan Carlos

- **Step 1 - GF Identification**: estimate $\mathbf{H}^{(t+1)}$ with $\mathbf{S}^{(t)}$ fixed

$$\mathbf{H}^{(t+1)} = \arg\min_{\mathbf{H}} \|\mathbf{Y} - \mathbf{H}\mathbf{X}\|_F^2 + \gamma \|\mathbf{S}^{(t)}\mathbf{H} - \mathbf{H}\mathbf{S}^{(t)}\|_F^2$$

$\Rightarrow$ LS problem with closed-form solution inverting an $N^2 \times N^2$ matrix

- **Step 2 - Graph Denoising**: estimate $\mathbf{S}^{(t+1)}$ with $\mathbf{H}^{(t+1)}$ fixed

$$\mathbf{S}^{(t+1)} = \arg\min_{\mathbf{S}\in\mathcal{S}} \sum_{i,j=1}^{N} \left(\lambda \bar{\Omega}_{ij}^{(t)} |S_{ij} - \bar{S}_{ij}| + \beta \Omega_{ij}^{(t)} |S_{ij}|\right) + \gamma \|\mathbf{S}\mathbf{H}^{(t+1)} - \mathbf{H}^{(t+1)}\mathbf{S}\|_F^2$$

$\Rightarrow$ With ℓ_1 weights $\Omega_{ij}^{(t)}, \bar{\Omega}_{ij}^{(t)}$ computed from previous GSO $\mathbf{S}^{(t)}$

- Steps 1 and 2 repeated for $t = 0, ..., t_{max} - 1$ iterations

Alternating optimization algorithm

Universidad Rey Juan Carlos

- **Step 1 - GF Identification**: estimate $\mathbf{H}^{(t+1)}$ with $\mathbf{S}^{(t)}$ fixed

$$\mathbf{H}^{(t+1)} = \arg\min_{\mathbf{H}} \|\mathbf{Y} - \mathbf{H}\mathbf{X}\|_F^2 + \gamma \|\mathbf{S}^{(t)}\mathbf{H} - \mathbf{H}\mathbf{S}^{(t)}\|_F^2$$

$\Rightarrow$ LS problem with closed-form solution inverting an $N^2 \times N^2$ matrix

- **Step 2 - Graph Denoising**: estimate $\mathbf{S}^{(t+1)}$ with $\mathbf{H}^{(t+1)}$ fixed

$$\mathbf{S}^{(t+1)} = \arg\min_{\mathbf{S} \in \mathcal{S}} \sum_{i,j=1}^{N} \left(\lambda \bar{\Omega}_{ij}{}^{(t)} |S_{ij} - \bar{S}_{ij}| + \beta \Omega_{ij}{}^{(t)} |S_{ij}| \right) + \gamma \|\mathbf{S}\mathbf{H}^{(t+1)} - \mathbf{H}^{(t+1)}\mathbf{S}\|_F^2$$

$\Rightarrow$ With ℓ_1 weights $\Omega_{ij}^{(t)}, \bar{\Omega}_{ij}^{(t)}$ computed from previous GSO $\mathbf{S}^{(t)}$

- Steps 1 and 2 repeated for $t = 0, ..., t_{max} - 1$ iterations

Theorem

The RFI algorithm **converges to an stationary** point if $\mathbf{S}$ does not have repeated eigenvalues and every row of $\tilde{\mathbf{X}} = \mathbf{V}^{-1}\mathbf{X}$ are nonzero

Joint graph filter ID

Universidad Rey Juan Carlos

- Now the goal is to estimate K GFs $\{\mathbf{H}_k\}_{k=1}^K$
  - $\Rightarrow$ For each $\mathbf{H}_k$ we have M_k input/output signals $\mathbf{X}_k/\mathbf{Y}_k$
- Several GFs show up in relevant settings [Segarra16][Isufi16]
  - $\Rightarrow$ Different network processes on a graph $\mathbf{Y}_k = \mathbf{H}_k \mathbf{X}_k + \mathbf{W}_k$
  - $\Rightarrow$ Graph-based multivariate time series $\mathbf{Y}_\kappa = \sum_{k=1}^K \mathbf{H}_k \mathbf{Y}_{\kappa-k} + \mathbf{X}_\kappa + \mathbf{W}_k$

Joint graph filter ID

Universidad Rey Juan Carlos

- Now the goal is to estimate K GFs $\{\mathbf{H}_k\}_{k=1}^K$
  - $\Rightarrow$ For each $\mathbf{H}_k$ we have M_k input/output signals $\mathbf{X}_k/\mathbf{Y}_k$
- Several GFs show up in relevant settings [Segarra16][Isufi16]
  - $\Rightarrow$ Different network processes on a graph $\mathbf{Y}_k = \mathbf{H}_k\mathbf{X}_k + \mathbf{W}_k$
  - $\Rightarrow$ Graph-based multivariate time series $\mathbf{Y}_\kappa = \sum_{k=1}^K \mathbf{H}_k\mathbf{Y}_{\kappa-k} + \mathbf{X}_\kappa + \mathbf{W}_k$
- Joint identification exploits each $\mathbf{H}_k$ being a polynomial on $\mathbf{S}$

$$\min_{\mathbf{S}\in\mathcal{S},\{\mathbf{H}_k\}_{k=1}^K} \sum_{k=1}^K \alpha_k \|\mathbf{Y}_k - \mathbf{H}_k\mathbf{X}_k\|_F^2 + \lambda r_{\delta_1}(\mathbf{S}-\bar{\mathbf{S}}) + \beta r_{\delta_2}(\mathbf{S}) + \sum_{k=1}^K \gamma \|\mathbf{S}\mathbf{H}_k - \mathbf{H}_k\mathbf{S}\|_F^2$$

  - $\Rightarrow$ K commutativity constraints improve estimation of $\mathbf{S}$
  - $\Rightarrow$ A better estimate of $\mathbf{S}$ leads to better estimates of $\mathbf{H}_k$
- Solved via 2-step alternating optimization

Efficient implementation

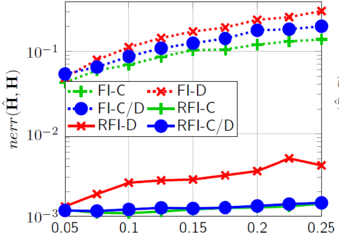
- RFI algorithm has a computational complexity of $\mathcal{O}(N^7)$
  - $\Rightarrow$ Prohibitive for large graphs
  - $\Rightarrow$ Steps 1 and 2 can be accelerated via an iterative process

Efficient implementation

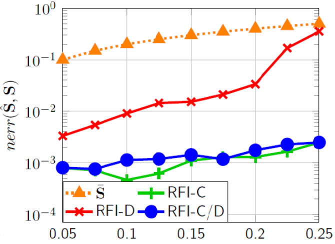
Universidad Rey Juan Carlos

- RFI algorithm has a computational complexity of $\mathcal{O}(N^7)$
  - $\Rightarrow$ Prohibitive for large graphs
  - $\Rightarrow$ Steps 1 and 2 can be accelerated via an iterative process
- **Step 1 - Efficient GF Identification**
  - $\Rightarrow$ Estimate $\mathbf{H}^{(t+1)}$ performing τ_{max_1} iterations of gradient descent
  - $\Rightarrow$ Involves multiplications of $N \times N$ matrices
- **Step 2 - Efficient Graph Denoising**
  - $\Rightarrow$ Estimate $\mathbf{S}^{(t+1)}$ via alternating optimization for τ_{max_2}
  - $\Rightarrow$ Solve N^2 scalar problems
  - $\Rightarrow$ Closed-form solution based on projected soft-thresholding
- Computational complexity reduced to $\mathcal{O}(N^3)$

- Test the estimates $\hat{\mathbf{H}}$ and $\hat{\mathbf{S}}$ with and without robust approach
  - ⇒ Graphs are sampled from the small-world random graph model
  - ⇒ We consider different types of perturbations



(a) Proportion of perturbed links



(b) Proportion of perturbed links

- RFI consistently outperforms classical FI
  - ⇒ Clear improvement in estimation of $\mathbf{S}$ with respect to $\bar{\mathbf{S}}$
- Only destroying links is the most damaging perturbation

Numerical Evaluation (II)

- **Dataset**: 5-nearest neighbor graph of weather stations in California
  - ⇒ Signals are temperature measurements
- **Goal:** Predict temperature 1 or 3 days in the future
  - ⇒ Estimate $\mathbf{H}$ using 25% or 50% of the available data
- Consider LS as a naive solution and TLS-SEM as a robust baseline

Models	1-Step		3-Step	
	TTS=0.25	TTS = 0.5	TTS=0.25	TTS = 0.5
LS	$6.9 \cdot 10^{-3}$	$3.1 \cdot 10^{-3}$	$2.1 \cdot 10^{-2}$	$9.1 \cdot 10^{-3}$
LS-GF	$3.3 \cdot 10^{-3}$	$3.3 \cdot 10^{-3}$	$8.4 \cdot 10^{-3}$	$8.5 \cdot 10^{-3}$
TLS-SEM	$4.0 \cdot 10^{1}$	$3.7 \cdot 10^{-2}$	$6.8 \cdot 10^{-1}$	$5.5 \cdot 10^{-2}$
RFI	$3.4 \cdot 10^{-3}$	$3.1 \cdot 10^{-3}$	$8.5 \cdot 10^{-3}$	$7.5 \cdot 10^{-3}$
AR(3)-RFI	$\mathbf{3.2 \cdot 10^{-3}}$	$\mathbf{2.8 \cdot 10^{-3}}$	$\mathbf{7.8 \cdot 10^{-3}}$	$\mathbf{6.9 \cdot 10^{-3}}$

- Best performance achieved by joint inference assuming AR model of order 3
  - ⇒ Follow up closely by the (separate) RFI algorithm

Closing remarks

- Proposed a general robust graph filter identification model that
  - ⇒ Simultaneously learns $\mathbf{S}$ and $\mathbf{H}$
- Problem formulated as a non-convex optimization problem
  - ⇒ Convex algorithm based on AM and MM techniques
  - ⇒ Proposed algorithm is shown to converge to a stationary point
- Generalized to joint GF identification to deal with several GFs
- Efficient algorithm to deal with graphs with large number of nodes
- Numerical evaluation over synthetic and real-world graphs
  - ⇒ Code: `https://github.com/reysam93/graph_denoising`

Questions at: **samuel.rey.escudero@urjc.es**