SIGNAL PROCESSING ON GRAPHS: PERFORMANCE OF GRAPH STRUCTURE ESTIMATION

Jonathan Mei and José M. F. Moura

Carnegie Mellon University
Department of Electrical and Computer Engineering
Pittsburgh, PA 15213

ABSTRACT

A class of models for describing sets of time series generated by interacting agents using directed, weighted graphs is introduced. A computationally tractable algorithm for estimating the graph adjacency matrix of this model from observed time series data is presented. The performance guarantees of this algorithm for prediction are outlined under several assumptions on the properties of the dynamics of the system of agents and on the true values of the parameters. These guarantees are tested empirically through simulation studies using several random graph models.

***Index Terms*—** Graph Signal Processing, Sparse Estimation, Adjacency Matrix, Time Series, Statistical Performance

1. INTRODUCTION

Networks play an increasingly central role in modern society. From the study of science [1] to socializing online [2], networks are at the heart of many complex systems of interacting entities. Graphs are useful mathematical objects that can be used to describe networks, and there is growing interest in using graphs to process signals that come from networks [3]. Most of these techniques assume the graph structure is given (as either the adjacency matrix $\mathbf{A}$ or the Laplacian $\mathbf{L}$); however, often in practice, the graph underlying networked data is not known and must be estimated in order to perform further analysis.

In this paper, we estimate the weighted, directed adjacency matrix $\mathbf{A}$ of a graph for a set of time series observed from dynamically interacting agents in a network. We also present performance guarantees for our estimation procedure and the setting under which these guarantees hold. We adopt the Discrete Signal Processing (DSP_G) framework [4] in associating the estimated graph with a "shift" operation that describes dependencies among agents or network effects.

We quickly overview the DSP_G framework and describe the time series model in section 2. Then we formulate our problem and present the estimation algorithm in section 3. Next, we describe the theoretical performance of the algorithm in section 4. We show empirical result for several random network models in section 5 and finally conclude in section 6.

2. RELATION TO PRIOR WORK

2.1. Discrete Signal Processing on Graphs

Discrete Signal Processing on Graphs (DSP_G) [5] provides a framework with which to analyze data with N elements for which relational information between elements is known.

Partially supported by NSF grants CCF 1011903 and CCF 1513936.

2.1.1. Graph Signals

Consider a graph $G = (\mathbf{V}, \mathbf{A})$ where $\mathbf{A}$ is the weighted adjacency matrix of the graph and the vertex set $\mathbf{V} = \{v_0, \ldots, v_{N-1}\}$. Each data element corresponds to a node v_n, and the weight A_{nm} is assigned to a directed edge from v_m to v_n. Graph signals can be written as N length vectors,

$$\mathbf{x} = \begin{pmatrix} x_0 & x_1 & \ldots & x_{N-1} \end{pmatrix}^T \in \mathbb{C}^N.$$

2.1.2. Graph Shift

A graph shift in DSP_G is a local operation with a graph signal $\mathbf{x}$ as input and another graph signal $\widetilde{\mathbf{x}}$ as output with entries

$$\widetilde{x}_i = \sum_{j \in \mathcal{N}_i} A_{ij} x_j$$

where $\mathcal{N}_i$ is the neighborhood of node v_i. In matrix notation, we have the shifted signal given by the product of the input signal with the adjacency matrix $\widetilde{\mathbf{x}} = \mathbf{A}\mathbf{x}$. These graph shifts can be seen as corresponding to "spatial" dimension.

2.2. Causal Graph Processes

We now detail the model for these graph processes (see [6, 7] for more details). We assume we are given N time series of length T, $\{x_n[k]\}$, with $n = 1, \ldots, N$ and $k = 0, 1, \ldots, T$, where n indexes the agent and k the time index. Our goal is to fit a dynamical model that captures both the "spatial" (across agents) and temporal dependencies of the data. The spatial dependencies are captured in our model by a graph $G(\mathbf{V}, \mathbf{A})$ where node v_n represents agent n, and $\mathbf{A}$ is the weighted, directed adjacency matrix of the graph. The value of A_{mn} represents the strength of the dependency between of the signal $\{x_m[k]\}$ on the past values of $\{x_n[k]\}$. The temporal dependencies are modeled by an autoregressive (AR) model relating the time series $\{\mathbf{x}[k]\}$ to its immediate past M values, where the vector $\mathbf{x}[k] = \begin{pmatrix} x_0[k] & x_1[k] & \ldots & x_{N-1}[k] \end{pmatrix}^T \in \mathbb{C}^N$ represents the graph signal at time sample k.

We consider a Causal Graph Process (CGP) to be a time series $\mathbf{x}[k]$ on a graph $G = (\mathbf{V}, \mathbf{A})$ of the following form:

$$\begin{aligned} \mathbf{x}[k] &= \mathbf{w}[k] + \sum_{i=1}^{M} P_i(\mathbf{A}, \mathbf{c}) \mathbf{x}[k-i] \\ &= \mathbf{w}[k] + \sum_{i=1}^{M} \Big(\sum_{j=0}^{i} c_{ij} \mathbf{A}^j \Big) \mathbf{x}[k-i] \\ &= \mathbf{w}[k] + (c_{10}\mathbf{I} + c_{11}\mathbf{A}) \mathbf{x}[k-1] \\ &\qquad + \left(c_{20}\mathbf{I} + c_{21}\mathbf{A} + c_{22}\mathbf{A}^2\right) \mathbf{x}[k-2] + \ldots \\ &\qquad + \Big(c_{M0}\mathbf{I} + \ldots + c_{MM}\mathbf{A}^M\Big) \mathbf{x}[k-M] \\ &= \mathbf{w}[k] + f(\mathbf{A}, \mathbf{c}, \mathbf{X}_{k-1}) \end{aligned} \tag{1}$$

where $P_i(\mathbf{A}, \mathbf{c})$ is a matrix polynomial in $\mathbf{A}$, c_{ij} are scalar polynomial coefficients, $\mathbf{c}$ is a vector containing c_{ij} as entries,

$$\mathbf{X}_{k-1} = \begin{pmatrix} \mathbf{x}[k-1]^\top & \mathbf{x}[k-2]^\top & \dots & \mathbf{x}[k-M]^\top \end{pmatrix}^\top$$

is a data vector collecting the immediate past M time samples, the function $f(\mathbf{A}, \mathbf{c}, \mathbf{X}_{k-1}) = \sum_{i=1}^{M} P_i(\mathbf{A}, \mathbf{c})\mathbf{x}[k-i]$, and $\mathbf{w}[k]$ is statistical noise. The matrix polynomials of the CGP AR(M) model $P_i(\mathbf{A}, \mathbf{c})$ are in the DSP$_\mathrm{G}$ framework causal graph filters. In this sense, "causal" refers both to the fact that we are only using information from past observations of the process to model current and future observations and to the fact that the $\mathbf{A}$ matrix is directed, and can model non-reciprocal effects. However, this is not determining true philosophical causation, but rather providing an increase in predictive power, similar to the notion described by Granger [8].

This model allows a signal on a node at the current time index to be affected through network effects by signals on other nodes at past times. This can model delays and difference processes in the network, which is crucial for modeling dynamics in many real world applications. The matrix polynomial $P_i(\mathbf{A}, \mathbf{c})$ is at most of order $\min(i, N_\mathbf{A})$, reflecting that $\mathbf{x}[k]$ cannot be influenced by more than i^{th} order network effects and in addition is mathematically limited by $N_\mathbf{A}$, the degree of the minimum polynomial of $\mathbf{A}$. This can be interpreted as some corresponding speed for information to travel through the graph. Typically, we take the model order $M < N_\mathbf{A}$.

3. ESTIMATING THE GRAPH STRUCTURE

The problem of interest in this paper is to estimate the graph adjacency matrix $\mathbf{A}$ that describes the spatial dependencies among the N time series, since in many applications, this matrix $\mathbf{A}$ is unknown and needs to be uncovered from the time series $\{x_n[k]\}$. We consider the problem within the scope of a restricted version of the model, which yields a computationally efficient estimation algorithm with theoretical properties that can be analyzed tractably. We present results for the restricted model, noting that the analysis for the full model is more involved and will be detailed in forthcoming work.

3.1. Restricted CGP (rCGP) Model

We focus our analysis to a particular subset of the CGP model class, which we will denote by rCGP. In this model, we restrict the matrix polynomials $P_i(\mathbf{A}, \mathbf{c})$ to be linear, corresponding to setting in the matrix polynomial coefficients $P_i(\mathbf{A}, \mathbf{c})$ of the AR model the coefficients $c_{ij} = 0$ for $j \geq 2$ and all i, i.e., (1) is now reduced to

$$\begin{aligned} \mathbf{x}[k] &= \mathbf{w}[k] + \sum_{i=1}^{M} P_i(\mathbf{A}, \mathbf{c})\mathbf{x}[k-i] \\ &= \mathbf{w}[k] + (c_{10}\mathbf{I} + c_{11}\mathbf{A})\mathbf{x}[k-1] \\ &\qquad + (c_{20}\mathbf{I} + c_{21}\mathbf{A})\,\mathbf{x}[k-2] + \dots \\ &\qquad + (c_{M0}\mathbf{I} + c_{M1}\mathbf{A})\,\mathbf{x}[k-M] \\ &= \mathbf{w}[k] + f(\mathbf{A}, \mathbf{c}, \mathbf{X}_{k-1}) \end{aligned} \tag{2}$$

With this parameterization as written, there are issues with identifiability of the model (2). In order to avoid these issues, we assume that $P_1(\mathbf{A}, \mathbf{c}) \neq \alpha\mathbf{I}$ for any $\alpha \in \mathbb{R}$. Then without further loss of generality, we can take a reduced parameterization with $c_{10} = 0$ and $c_{11} = 1$ so that $P_1(\mathbf{A}, \mathbf{c}) = \mathbf{A}$.

To see this, consider the full parameterization using $(\mathbf{A}', \mathbf{c}')$. We show that we can use the reduced parameterization $(\mathbf{A}, \mathbf{c})$ with $P_1(\mathbf{A}, \mathbf{c}) = \mathbf{A}$ to represent the same process. We have $P_1(\mathbf{A}', \mathbf{c}') = c'_{10}\mathbf{I} + c'_{11}\mathbf{A}'$ so $\mathbf{A} = c'_{10}\mathbf{I} + c'_{11}\mathbf{A}'$. Then $P_i(\mathbf{A}, \mathbf{c}) = c_{i0}\mathbf{I} + c_{i1}\mathbf{A} = (c_{i0} + c'_{10}c_{i1})\mathbf{I} + c_{i1}c'_{11}\mathbf{A}'$. By assumption, $c'_{11} \neq 0$, so we can set $c_{i1} = c'_{i1}/c'_{11}$ and $c_{i0} = c'_{i0} - c'_{10}c'_{i1}/c'_{11}$. Then $P_i(\mathbf{A}, \mathbf{c}) = c'_{i0}\mathbf{I} + c'_{i1}\mathbf{A}' = P_i(\mathbf{A}', \mathbf{c}')$. In the remainder of this paper, we use the reduced parameterization.

3.2. Formulation and Algorithm

Here we describe our algorithm to estimate the graph. We formulate this as the optimization problem

$$\begin{aligned} (\widehat{\mathbf{A}}, \widehat{\mathbf{c}}) = \underset{\mathbf{A},\mathbf{c}}{\operatorname{argmin}} & \sum_{k=M}^{T-1} \left\| \mathbf{x}[k] - f(\mathbf{A}, \mathbf{c}, \mathbf{X}_{k-1}) \right\|_2^2 \| + \lambda \|\mathbf{A}\|_1 \\ \text{s.t.} \quad & \|\mathbf{c}\|_1 \leq \rho \end{aligned} \tag{3}$$

where $\|\mathbf{A}\|_1 = \sum_{i,j} |a_{ij}|$ is an ℓ_1 norm on matrix $\mathbf{A}$ and $\|\mathbf{c}\|_1 = \sum_{i,j} |c_{ij}|$ is the usual ℓ_1 vector norm. The overall problem (3) is nonconvex due to the multiplicative form of $P_i(\mathbf{A}, \mathbf{c})$, so we break the problem down into convex steps:

1. Estimate $\mathbf{R}_i \approx \mathbf{P}_i(\mathbf{A}, \mathbf{c})$ and set $\widehat{\mathbf{A}}^{(1)} = \widehat{\mathbf{R}}_1$.
2. Repeat for $t = 1, \dots, t_{\max}$ or until convergence:
   (a) Estimate $\widehat{\mathbf{c}}^{(t)}$ with fixed $\widehat{\mathbf{A}}^{(t)}$.
   (b) Estimate $\widehat{\mathbf{A}}^{(t+1)}$ with fixed $\widehat{\mathbf{c}}^{(t)}$ and increment t.

In breaking down the problem into separate steps, we aim to gain computational tractability at the potential expense of optimality from solving the original problem. However, we later show that the procedure does not sacrifice too much in performance, and that the solution we find approaches the optimal solution with high probability.

More specifically, to estimate $\{\mathbf{R}_i\}$ in step 1, we solve the following optimization:

$$\{\widehat{\mathbf{R}}_i\} = \underset{\{\mathbf{R}_i\}}{\operatorname{argmin}} \sum_{k=M}^{T-1} \left\| \mathbf{x}[k] - \sum_{i=1}^{M} \mathbf{R}_i \mathbf{x}[k-i] \right\|_2^2 + \lambda \sum_{i=1}^{M} \|\mathbf{R}_i\|_1 \tag{4}$$

Here, instead of directly estimating $\mathbf{A}$, we first estimate the value of the matrix polynomials $P_i(\mathbf{A}, \mathbf{c})$. The first term is a model-following term, and the ℓ_1 penalty term encourages sparsity in our estimates $\mathbf{R}_i$ of the polynomials. This is a convex problem in $\mathbf{R}_i$ and can be solved using conventional penalized least-squared techniques.

To solve step 2a, we solve another optimization problem:

$$\begin{aligned} \widehat{\mathbf{c}}^{(t)} = \underset{\mathbf{c}}{\operatorname{argmin}} & \sum_{k=M}^{T-1} \left\| \mathbf{x}[k] - f(\widehat{\mathbf{A}}^{(t)}, \mathbf{c}, \mathbf{X}_{k-1}) \right\|_2^2 \\ \text{s.t.} \quad & \|\mathbf{c}\|_1 \leq \rho \end{aligned} \tag{5}$$

The first term is again a model-following term, and the ℓ_1 constraint on $\mathbf{c}$ encourages terms of c_{ij} towards 0. This constraint can be seen as allowing for automatic model order selection, as we may expect the coefficients for high order lags to fall off in magnitude. Note that this problem is convex in $\mathbf{c}$ since $\mathbf{A}^{(t)}$ is fixed, and can be solved by standard constrained least-squares methods.

Finally, we solve step 2b using the optimization problem:

$$\widehat{\mathbf{A}}^{(t+1)} = \underset{\mathbf{A}}{\operatorname{argmin}} \sum_{k=M}^{T-1} \left\| \mathbf{x}[k] - f(\mathbf{A}, \widehat{\mathbf{c}}^{(t)}, \mathbf{X}_{k-1}) \right\|_2^2 + \lambda \|\mathbf{A}\|_1 \tag{6}$$

The first term is still the model-following term, and the ℓ_1 penalty on $\mathbf{A}$ now encourages the graph to be sparse. This corresponds to our desire to have an interpretable yet still interesting and descriptive model of the inter-agent interactions. This is also convex in $\mathbf{A}$ for fixed $\mathbf{c}^{(t)}$, and can be solved with sparse least-squares procedures.

To obtain an estimate from this algorithm, we can terminate at any point after obtaining the initial estimate $(\widehat{\mathbf{A}}^{(1)}, \widehat{\mathbf{c}}^{(1)})$, i.e., after completing step 2a for the first time.

4. PERFORMANCE GUARANTEES

Here we outline the theoretical performance guarantees of the estimate $(\widetilde{\mathbf{A}}, \widetilde{\mathbf{c}}) = (\widehat{\mathbf{A}}^{(1)}, \widehat{\mathbf{c}}^{(1)})$. Our error metric of interest will be:

$$\begin{aligned}\epsilon^{(t)} =& \mathbb{E}\left[\frac{1}{N}\left\|\mathbf{x}[k] - f(\widehat{\mathbf{A}}^{(t)}, \widehat{\mathbf{c}}^{(t)}, \mathbf{X}_{k-1})\right\|_2^2\right] \\ &- \mathbb{E}\left[\frac{1}{N}\left\|\mathbf{x}[k] - f(\mathbf{A}, \mathbf{c}, \mathbf{X}_{k-1})\right\|_2^2\right]\end{aligned} \tag{7}$$

which is the average excess prediction risk. In words, under the setting of predicting new samples, this quantity is the per-node error of the estimator above the intrinsic randomness of the process. Here the expectation is taken over a new samples $\mathbf{z}_k$ drawn independently of the samples used to estimate $(\widehat{\mathbf{A}}^{(t)}, \widehat{\mathbf{c}}^{(t)})$.

4.1. Assumptions

We lay out the specific assumptions about the process we make in order to derive our guarantees:

(A1) The model class is accurate:
$\mathbb{E}[\mathbf{x}[k] \,|\, \mathbf{X}_{k-1}] = f(\mathbf{A}, \mathbf{c}, \mathbf{X}_{k-1})$.

(A2) The noise is uncorrelated with the process and itself:
$\mathbb{E}[\mathbf{x}[j]\mathbf{w}[k]^\top] = \mathbf{0}$ and $\mathbb{E}[\mathbf{w}[j]\mathbf{w}[k]^\top] = \mathbf{0}$ for $j \leq k$. In addition, the noise is multivariate Gaussian with distribution $\mathbf{w}[k] \sim \mathcal{N}(\mathbf{0}, \Sigma_{\mathbf{w}})$, with $0 < \sigma_\ell \leq \|\Sigma_{\mathbf{w}}\| \leq \sigma_u$ being lower and upper bounded.

(A3) The process is stationary and is in steady state when we begin our observation. Under this assumption, the marginals distributions and expectations can be meaningfully defined, $\mathbb{E}[\mathbf{x}[k]] = \mathbf{0}$, $\Sigma_0 = \mathbb{E}[\mathbf{x}[k]\ \mathbf{x}[k]^\top]$, and $\Sigma = \mathbb{E}[\mathbf{z}_k\mathbf{z}_k^\top]$ where $\mathbf{z}_k = (\mathbf{x}[k]^\top\ \mathbf{X}_{k-1}^\top)^\top$.

(A4) The stationary correlation matrices are absolutely summable:

$$\sum_{i=-\infty}^{\infty} \|\mathbb{E}[\mathbf{x}[k]\mathbf{x}[k-i]^\top]\| = G.$$

This is a slightly stronger condition than stationarity.

(A5) The true adjacency matrix and filter coefficients are sparse and bounded: $\|\mathbf{A}\|_1 \leq S_N \ll N^2$, $\|\mathbf{A}\|_0 \leq s_N \ll N^2$, and $\|\mathbf{A}\| \leq L$; $\|\mathbf{c}\|_2 \leq \|\mathbf{c}\|_1 \leq \rho$. Also, $1 < Q = (1 + L)(1 + \rho) \leq 2$. This is also a slightly stronger condition than stationarity. The quantities with subscripts may grow with N.

(A6) The sample size is large enough relative to the "stability" of the process:

$$K = T - M \geq C\omega^2\, s_N\, (\log M + \log N)$$

for some constant $C > 0$ and $\omega = \frac{\sigma_u Q^2}{\sigma_\ell [2-Q]^2}$. Here ω is related to measures of "stability" of the process [9].

4.2. Theoretical Performance

Here, we present the main result and a brief sketch for its proof. We note that the lemmas of intermediate results and the full proof take several pages and are omitted due to lack of space.

Theorem 1 (Main result)**.** *Under Assumptions **(A1)**–**(A6)**, for some constants $d_i > 0$ and for all $0 \leq \beta \leq 1$, with probability at least*

$$1 - \left(\frac{2}{e}\left(\frac{2e}{NT}\right)^{\frac{NT}{2}} + 2\exp\left(-\frac{3}{64}(NT)^\beta\right) + d_2\exp\left(-\frac{d_3}{\omega^2}K\right)\right)$$

the error satisfies

$$\begin{aligned}\epsilon^{(1)} \leq& d_1(g(Q)\sigma_u)^2\left(\frac{\log M + \log N}{NK}\right)\frac{2s_N Q^2}{\sigma_\ell}tr(\Sigma_0) \\ &+ \frac{2GQ^2T}{K(NT)^{(1-\beta)/2}}\end{aligned}$$

Proof Sketch. Following similar algebra as [10], we can show that

$$\begin{aligned}\epsilon^{(1)} &= \frac{1}{N}\mathbb{E}\left[\left\|f(\widetilde{\mathbf{A}}, \mathbf{c}, \mathbf{X}_{k-1}) - f(\mathbf{A}, \mathbf{c}, \mathbf{X}_{k-1})\right\|_2^2\right] \\ &+ \frac{1}{N}\Big(\mathbb{E}\Big[\left\|\mathbf{x}[k] - f(\widetilde{\mathbf{A}}, \widetilde{\mathbf{c}}, \mathbf{X}_{k-1})\right\|_2^2 - \left\|\mathbf{x}[k] - f(\widetilde{\mathbf{A}}, \mathbf{c}, \mathbf{X}_{k-1})\right\|_2^2\Big]\Big)\end{aligned}$$

We can bound the first term using a result from AR estimation [9]. With a bit of additional algebra and Gaussian concentration results [11], we can bound the second term (in the parentheses) by its empirical version with high probability. □

5. EXPERIMENTS

We ran several Monte-Carlo experiments to study the empirical behavior of the algorithm, varying the number of nodes $N = 50, 100, 200$, the number of observed time samples $T = pN$ with $p = 0.5, 0.75, 1, 1.25, 1.5, 2$, and the network topologies. For each topology, we first generated a weighted adjacency matrix $\mathbf{A}$ and corresponding coefficients $\mathbf{c}$ to make a stable system of order $M = 3$ for each pair of (N, T). Then with that fixed $(\mathbf{A}, \mathbf{c})$, we generated 20 independent sets of multivariate time series data from an rCGP process with parameters $(\mathbf{A}, \mathbf{c})$. Finally, we found estimates $(\widetilde{\mathbf{A}}^{(i)}, \widetilde{\mathbf{c}}^{(i)})$ with $i = 1, 2 \ldots, 20$ for each of the 20 sets. Next, we computed the empirical error in $\mathbf{A}$ as

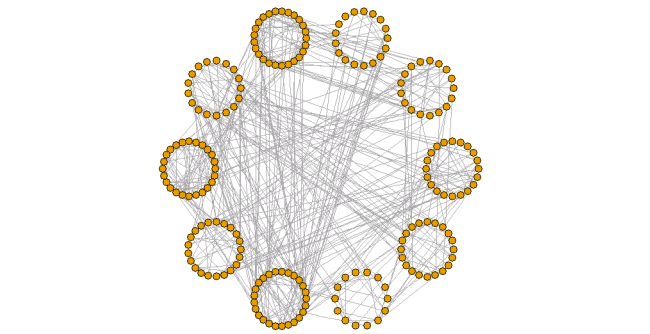
$$\widehat{\epsilon}_{\mathbf{A}} = \frac{1}{20N}\sum_{i=1}^{20}\|\widetilde{\mathbf{A}}^{(i)} - \mathbf{A}\|_F^2$$

and the empirical excess prediction error

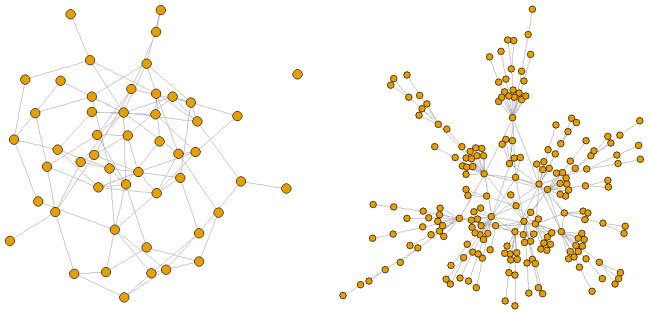
$$\widehat{\epsilon}^{(1)} = \frac{1}{20N}\sum_{i=1}^{20}\sum_{k=M}^{T-1}\frac{1}{K}\|\mathbf{x}^{(i)}[k] - f(\widetilde{\mathbf{A}}^{(i)}, \widetilde{\mathbf{c}}, \mathbf{X}_{k-1}^{(i)})\|_2^2.$$

The random graphs were generated with 3 different topologies: Stochastic Block Model (SBM) [12], Erdös-Renyi (ER) [13], and Power Law (PL). Examples of these topologies can be seen in figure 1.

The SBM graph was generated by generating 10 clusters with each node having uniform probability of belonging to a cluster. Edges between nodes were generated according to assigned intra- and inter- cluster probabilities. The edges generated were assigned



(a) Stochastic Block Model, $N = 200$



(b) Erdös-Renyi, $N = 50$ (c) Power Law, $N = 200$

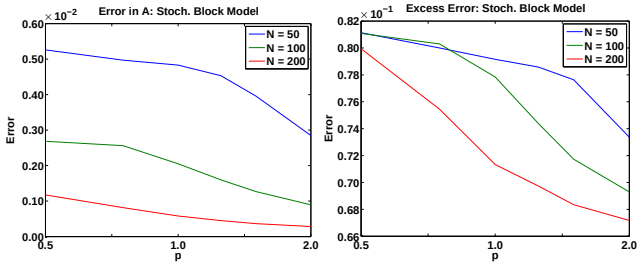
Fig. 1. Example graph topologies used in experiments

weights from a Laplacian distribution with rate $\lambda_e = 2$. Finally, the matrix was normalized by 1.1 times its largest singular value.

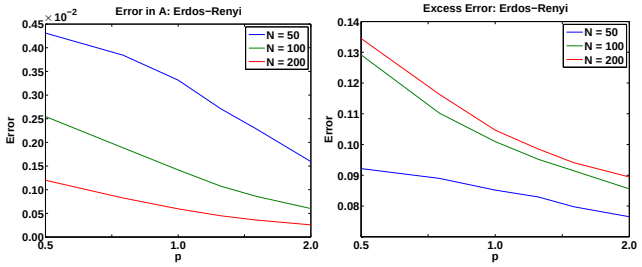
The ER graph was generated by taking edges from a standard normal $\mathcal{N}(0, 1)$ distribution and then thresholding edges to be between 1.6 and 1.8 in absolute to yield an effective probability of an edge $p_{ER} \approx 0.04$. The edges were soft thresholded by 1.5 to be between 0.1 and 0.3 in magnitude. Finally, the matrix was normalized by 1.5 times its largest eigenvalue.

The PL graph was generated by starting with a 15 node ER graph with connection probability $3/4$. New nodes were connected by one new edge of weight 1 to an existing node according to a modified preferential attachment scheme [14]. The direction of the edge was determined by a binary random variable. The probability of the new node connecting to an existing node was proportional to the existing node's out-degree (in-degree) if the edge was pointing toward (away from) the new node. The diagonal was set to $-1/2$. Lastly, the matrix was normalized by 1.5 times its largest singular value.

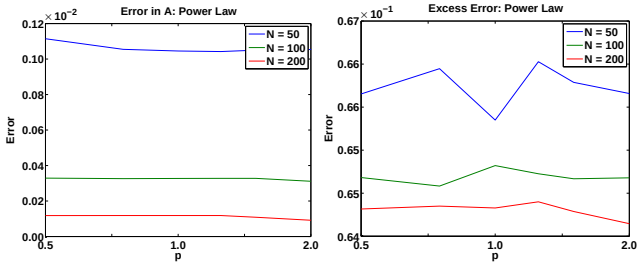
In figure 2, we see three different behaviors. For the SBM, both $\widehat{\epsilon}_{\mathbf{A}}$ and $\widehat{\epsilon}^{(1)}$ decrease with both N and T. However in the ER model, $\widehat{\epsilon}_{\mathbf{A}}$ decreases with both N and T while $\widehat{\epsilon}^{(1)}$ only decreases with T. This may be partly due to the fact that the sparsity of the ER model scales as $s_N \sim N^2$, which is fairly fast, while the sparsity in the SBM scales as $s_N \sim N$. This may be slow enough that even for increasing N we observe decreasing error. Note that in the PL model, $\widehat{\epsilon}_{\mathbf{A}}$ decays with N but does not decay significantly with T. This may be partially due to the fact that the matrix $\mathbf{A}$ is somewhat poorly conditioned, since most nodes may have no incoming or no outgoing edges, while a few nodes have many edges. Consequently, $\widehat{\epsilon}^{(1)}$ also does not show significant decay with T, although there is still decay with increasing N.



(a) Errors for Stochastic Block Model topology



(b) Errors for Erdös-Renyi topology



(c) Errors for Power Law topology

Fig. 2. Average errors in $\mathbf{A}$ and excess prediction errors for 3 different random graph topologies

6. CONCLUSION

We presented a DSP_G based model that uses directed, weighted adjacency matrices $\mathbf{A}$ to describe the dependencies among time series generated by systems of interacting agents. We outlined a computationally tractable algorithm for estimating the matrix $\mathbf{A}$ and matrix polynomial coefficients $\mathbf{c}$. We sketched the proof for performance guarantees on prediction under assumptions on the properties of the dynamic process and true parameter values. Finally, we observed that the algorithm performance is consistent with the error expression for several random graph models.

Our future work will relate the estimation and prediction performance to the network topology and consider the full CGP model rather than the rCGP model studied in this paper.

7. REFERENCES

[1] Karen Sachs, Omar Perez, Dana Pe'er, Douglas A. Lauffenburger, and Garry P. Nolan, "Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data," *Science*, vol. 308, no. 5721, pp. 523–529, Apr. 2005.

[2] J. Leskovec, M. McGlohon, C. Faloutsos, N. Glance, and M. Hurst, "Patterns of Cascading Behavior in Large Blog Graphs," in *Proceedings of the 2007 SIAM International Conference on Data Mining*, Proceedings, pp. 551–556. Society for Industrial and Applied Mathematics, Apr. 2007.

[3] D.I. Shuman, S.K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, May 2013.

[4] A. Sandryhaila and J. M. F. Moura, "Discrete Signal Processing on Graphs," *IEEE Transactions on Signal Processing*, vol. 61, no. 7, pp. 1644–1656, Apr. 2013.

[5] A. Sandryhaila and J. M. F. Moura, "Discrete Signal Processing on Graphs: Frequency Analysis," *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3042–3054, June 2014.

[6] Jonathan Mei and José M. F. Moura, "Signal Processing on Graphs: Modeling (Causal) Relations in Big Data," *arXiv:1503.00173 [cs, math, stat]*, Feb. 2015, arXiv: 1503.00173.

[7] J. Mei and J. M. F. Moura, "Signal processing on graphs: Estimating the structure of a graph," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2015, pp. 5495–5499.

[8] C. W. J. Granger, "Investigating Causal Relations by Econometric Models and Cross-spectral Methods," *Econometrica*, vol. 37, no. 3, pp. 424–438, Aug. 1969.

[9] Sumanta Basu and George Michailidis, "Regularized estimation in sparse high-dimensional time series models," *The Annals of Statistics*, vol. 43, no. 4, pp. 1535–1567, Aug. 2015.

[10] Ravi Ganti, Nikhil Rao, Rebecca M. Willett, and Robert Nowak, "Learning Single Index Models in High Dimensions," *arXiv:1506.08910 [cs, stat]*, June 2015, arXiv: 1506.08910.

[11] Iain M. Johnstone, "Chi-square oracle inequalities," in *Institute of Mathematical Statistics Lecture Notes - Monograph Series*, pp. 399–418. Institute of Mathematical Statistics, Beachwood, OH, 2001.

[12] Brian Karrer and M. E. J. Newman, "Stochastic blockmodels and community structure in networks," *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics*, vol. 83, no. 1 Pt 2, pp. 016107, Jan. 2011.

[13] P. Erdös and A. Rényi, "On the Evolution of Random Graphs," in *Publication of the Mathematical Institute of the Hungarian Academy of Sciences*, 1960, pp. 17–61.

[14] Albert-László Barabási and Réka Albert, "Emergence of Scaling in Random Networks," *Science*, vol. 286, no. 5439, pp. 509–512, Oct. 1999.