GRAPH ERROR EFFECT IN GRAPH SIGNAL PROCESSING

Jari Miettinen, Sergiy A. Vorobyov, and Esa Ollila

Department of Signal Processing and Acoustics, Aalto University, P.O. Box 15400, FIN-00076 Aalto

ABSTRACT

The first step in any graph signal processing (GSP) task is to learn the graph signal representation, i.e., to capture the dependence structure of the data into an adjacency matrix. Indeed, the adjacency matrix is typically not known a priori and has to be learned. However, it is learned with errors. A little, if any, attention has been paid to modeling such errors in the adjacency matrix, and studying their effects on GSP tasks. Modeling errors in adjacency matrix will enable both to study the graph error effects in GSP and to develop robust GSP algorithms. In this paper, we therefore introduce practically justifiable graph error models. We also study, both analytically and in terms of simulations, the graph error effect on the performance of GSP based on the example of independent component analysis of graph signals (graph decorrelation).

Index Terms— Erdös-Rényi graphs, error effect, graph signal processing, minimum distance index, shift matrix

1. INTRODUCTION

Graph signal processing (GSP) expands the standard signal processing tools to datasets whose structures differ from those of time series [1, 2]. Such data arise from multiple fields, including for example sensor networks, brain networks, gene regulatory networks and social networks [3, 4, 5, 6]. In the first step of any GSP task, the dependence structure of the data should be captured into an adjacency matrix $\mathbf{A}$. The ijth element of $\mathbf{A}$ is nonzero if the ith and the jth data units are related, and the value $[\mathbf{A}]_{ij} = a_{ij}$ describes the strength of the relationship.

It has been argued that knowing $\mathbf{A}$ a priori is not always possible, and that precise estimation of the adjacency matrix (or the graph Laplacian matrix) is important for succesful GSP [7, 8]. However, studying the consequences of misspecifying the adjacency matrix as a result of imperfect learning is missing in the literature. In this paper, we introduce practically justifiable graph error models that aim to capture the effect of imperfectly learned signal graph adjacency matrix. They are useful for analyzing graph error effects in GSP as well as for designing robust GSP algorithms. As an example, we study the graph error effect on the performance of independent component analysis (ICA) of graph signals by quantifying the effect of adjacency matrix errors to the performance of the second-order source separation method of [6], referred hereafter as graph decorrelation (GraDe). The study is performed both analytically and in terms of simulations. To the best of our knowledge, this is the first work that introduces graph error models for GSP and studies their effect in subsequent GSP tasks.

Notation: We use boldface capital letters for matrices, boldface lowercase letters for vectors, and capital calligraphic letters for sets. The exceptions are $\mathbf{1}_N$ which is the N-dimensional vector full of ones, the $M \times N$ matrix full of ones $\mathbf{1}_{M\times N} = \mathbf{1}_M\mathbf{1}_N^\top$, and $\mathbf{1}_\mathbf{A}$ is a matrix of the same size as $\mathbf{A}$, such that $[\mathbf{1}_\mathbf{A}]_{ij} = 1$, if $a_{ij} \neq 0$ and $[\mathbf{1}_\mathbf{A}]_{ij} = 0$, if $a_{ij} = 0$. The matrix $\mathbf{I}_{N\times N}$ is the $N \times N$ identity matrix. The notations $(\cdot)^\top$, $\odot$, $\|\cdot\|$, $\mathrm{tr}\{\cdot\}$, $\mathbb{P}(\cdot)$, $\mathbb{E}\{\cdot\}$, and $\mathrm{var}(\cdot)$ stand for the transpose, Hadamard product, Euclidian norm of a vector, trace of a matrix, probability, mathematical expectation, and variance, respectively.

2. GRAPH ERROR MODELS

Let $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ be a directed graph that represents a graph signal, where $\mathcal{N}$ is the set of N nodes and $\mathcal{E}$ is the set of edges. Then the true but unknown adjacency matrix of the graph $\mathcal{G}$, denoted as $\mathbf{A}$, is a matrix that satisfies the conditions $a_{ii} = 0$ for $i = 1, \ldots, N$ and $a_{ij} \neq 0$ if and only if $(i, j) \in \mathcal{E}$, i.e., there is an edge from the jth node to the ith node.

For developing our graph error models, we will use the Erdös–Rényi model according to which a random graph is constructed by connecting nodes randomly with a constant probability [9]. The corresponding graph is denoted as $\mathcal{G} = (\mathcal{N}, \epsilon)$ and its adjacency matrix $\mathbf{\Delta}_\epsilon$ is a random $N \times N$ matrix such that $\mathbb{P}([\mathbf{\Delta}_\epsilon]_{ij} = 1) = \epsilon$ and $\mathbb{P}([\mathbf{\Delta}_\epsilon]_{ij} = 0) = 1 - \epsilon$ for all $i \neq j$, and $[\mathbf{\Delta}_\epsilon]_{ii} = 0$ for $i = 1, \ldots, N$, where each element of the matrix is generated independently from the other elements. An important characteristic of the Erdös-Rényi graph is that it does not allow for formations of communities [10], and if applied on the top of another graph, it will not change the essential structures of such graphs, which can be described, for example, in terms of other kernel-based random graphs known as graphons [10, 11, 12]. Instead, it just disturbs the spectrum of the original graph. Whether the essential structures in a graph signal contaminated by Erdös-

This research was supported in parts by the Academy of Finland grants No. 299243 and No. 298118

Rényi graph can be correctly captured will depend of course on how large the probability ϵ is.

Because of the errors in the adjacency matrix learning, the estimated adjacency matrix will deviate from the true one. Consider first unweighted graphs, for which the adjacency matrix becomes $\mathbf{A} = a\mathbf{1}_{\mathbf{A}}$ where $a > 0$ is some constant weight. Assuming that the outcome of the graph signal adjacency matrix learning is accurate enough, that is, assuming that all or sufficiently many essential structures of the graph signal are captured correctly, the learning errors can be accurately described by the Erdös-Rényi model. Then the actually available learned adjacency matrix of a graph signal can be modeled as an inaccurate version of $\mathbf{A}$, that is,

$$\mathbf{A}_\epsilon = \mathbf{A} + \mathbf{\Delta}_\epsilon \odot (a\mathbf{1}_{N\times N} - 2\mathbf{A}). \quad (1)$$

According to model (1), the true adjacency matrix of a graph signal is disturbed as a result of imperfect learning by the Erdös-Rényi model, where the level of distortion depends on the probability ϵ in the following way. Since the Erdös-Rényi graph applies on the top of another graph, an edge can be added with probability ϵ, when there is no edge in the true graph, or an edge of the true graph can be removed with the same probability. It corresponds to flipping the value from 0 to 1 or from 1 to 0 in the matrix $\mathbf{1}_{\mathbf{A}}$ in the positions corresponding to value 1 in the Erdös-Rényi adjacency matrix $\mathbf{\Delta}_\epsilon$.

The basic model (1), can be easily extended to the case where the probability of removing an edge from a graph which correctly captures a graph signal, denoted as ϵ_1, is not the same as the probability of adding an edge, which does not exist in the true graph, denoted as ϵ_2. The corresponding inaccurately learned adjacency matrix of a graph signal can then be modeled as

$$\mathbf{A}_{\epsilon_1,\epsilon_2} = \mathbf{A} - \mathbf{\Delta}_{\epsilon_1} \odot \mathbf{A} + \mathbf{\Delta}_{\epsilon_2} \odot (a\mathbf{1}_{N\times N} - \mathbf{A}). \quad (2)$$

Model (2) can be interpreted as an application of two Erdös–Rényi graphs on top of the true graph, where one Erdös–Rényi graph $\mathcal{G} = (\mathcal{N}, \epsilon_2)$ can only add edges which do not exist in the true graph, while the other Erdös-Rényi graph $\mathcal{G} = (\mathcal{N}, \epsilon_1)$ can only remove existing edges. It is easy to see that (2) is equivalent to (1) when $\epsilon_1 = \epsilon_2 = \epsilon$.

Further, model (2) can be extended to graph signals represented by weighted graphs. Let $\mathcal{A}$ be the set of nonzero elements of the true $\mathbf{A}$. The inaccurately learned weighted adjacency matrix is then modeled as

$$\begin{aligned}\mathbf{A}_{\epsilon_1,\epsilon_2,c} =& \mathbf{A} + (\mathbf{1}_{N\times N} - \mathbf{\Delta}_{\epsilon_1}) \odot \mathbf{1}_{\mathbf{A}} \odot \mathbf{\Sigma}_c - \mathbf{\Delta}_{\epsilon_1} \odot \mathbf{A} \\ &+ \mathbf{\Delta}_{\epsilon_2} \odot \mathbf{B} \odot (\mathbf{1}_{N\times N} - \mathbf{1}_{\mathbf{A}}) \end{aligned} \quad (3)$$

where $\mathbf{B}$ is an $N \times N$ matrix whose elements are drawn from $\mathcal{A}$ with replacement and $\mathbf{\Sigma}_c$ is an $N \times N$ matrix whose elements are drawn from a zero mean Gaussian distribution with variance $c\sigma^2$, where σ^2 is the sample variance of $\mathcal{A}$ and c is the variance multiplier.

Moreover, models (1)–(3) can be easily revised for undirected graphs by defining lower triangular matrices $\mathbf{\Delta}_\epsilon^l$ and $\mathbf{\Sigma}_c^l$ analogously to $\mathbf{\Delta}_\epsilon$ and $\mathbf{\Sigma}_c$, and then replacing $\mathbf{\Delta}_\epsilon$ and $\mathbf{\Sigma}_c$ by $\mathbf{\Delta}_\epsilon^l + (\mathbf{\Delta}_\epsilon^l)^\top$ and $\mathbf{\Sigma}_c^l + (\mathbf{\Sigma}_c^l)^\top$, respectively.

Models (1)–(3) all share the same additive structure, i.e., have the form

$$\mathbf{W} = \mathbf{A} + \mathbf{E} \quad (4)$$

where $\mathbf{W} \in \{\mathbf{A}_\epsilon, \mathbf{A}_{\epsilon_1,\epsilon_2}, \mathbf{A}_{\epsilon_1,\epsilon_2,c}\}$ and $\mathbf{E} \in \{\mathbf{\Delta}_\epsilon \odot (a\mathbf{1}_{N\times N} - 2\mathbf{A}), -\mathbf{\Delta}_{\epsilon_1} \odot \mathbf{A} + \mathbf{\Delta}_{\epsilon_2} \odot (a\mathbf{1}_{N\times N} - \mathbf{A}), (\mathbf{1}_{N\times N} - \mathbf{\Delta}_{\epsilon_1}) \odot \mathbf{1}_{\mathbf{A}} \odot \mathbf{\Sigma}_c - \mathbf{\Delta}_{\epsilon_1} \odot \mathbf{A} + \mathbf{\Delta}_{\epsilon_2} \odot \mathbf{B} \odot (\mathbf{1}_{N\times N} - \mathbf{1}_{\mathbf{A}})\}$. Then the error matrix $\mathbf{E}$ can be viewed as an analog of the additive error/noise component in the traditional signal processing and time series analysis. Moreover, the Erdös–Rényi graph is the basic GSP error model analogous to the Gaussian noise in the traditional signal processing.

3. GRAPH MOVING AVERAGE SIGNAL MODEL

Translating time series concepts from the traditional signal processing into the GSP context is yet a major part of GSP literature. Indeed, developments in GSP are typically based on defining/considering a certain meaningful, but not unique, shift operator, Laplacian matrix and graph filter being the most popular ones [7, 8, 13]. Here we extend the framework of designing meaningful shift operators by introducing and analyzing a basic graph moving average (GMA) signal model. Specifically, the GMA signal model of order m, GMA(m), called after the traditional time series moving average (MA) model, is given as

$$\mathbf{z} = \mathbf{y} + \sum_{l=1}^{m} \theta_l \mathbf{A}^l \mathbf{y} \quad (5)$$

where $\mathbf{y} \triangleq [y_1, \ldots, y_N]^\top$ with $y_1, \ldots, y_N \sim N(0, \sigma_y^2)$ being mutually independent Gaussian random variables with zero mean and variance σ_y^2, and $\theta_1, \ldots, \theta_m$ are MA coefficients. Then the composite shift matrix corresponding to GMA(m) model (5) and defined via the signal graph adjacency matrix $\mathbf{A}$ is $\sum_{l=1}^m \theta_l \mathbf{A}^l$. Such composite shift matrix can be decomposed and viewed as an iterative application (m times) of the basic shift matrix, the adjacency matrix $\mathbf{A}$ itself, weighted by corresponding MA coefficients $\{\theta_l\}_{l=1}^m$. To the best of our knowledge, the GMA has been considered before in [6], but was not studied. Notice that the traditional time series MA model is obtained from (5) when $\mathbf{A}$ is the cycle graph which satisfies $a_{ij} = 1$, if $j = i - 1$, and $a_{ij} = 0$, otherwise.

Let us now derive some statistics of the graph signal given by (5). For analytical tractability of the later studies and because of the space limitations, we limit our study here mostly for the GMA(1) model

$$\mathbf{z} = \mathbf{y} + \theta\mathbf{A}\mathbf{y} = \tilde{\mathbf{A}}\mathbf{y} \quad (6)$$

where $\tilde{\mathbf{A}} \triangleq \mathbf{I}_{N\times N} + \theta\mathbf{A}$. In detailed view, the value of the ith node is given by $z_i = y_i + \theta \sum_{j\in\mathcal{N}_i} a_{ij} y_j$, where $\mathcal{N}_i$ denotes

the incoming neighbors of node i. Thus, if $\theta \neq 0$, two nodes are correlated if they are neighbors, or if they have shared incoming neighbors.

The covariance matrix of the graph signal $\mathbf{z}$ given by the GMA(1) model can be found as a function of $\mathbf{A}$ as follows

$$\begin{aligned}\mathbf{C}_z(\mathbf{A}) &\triangleq \mathbb{E}\left\{\mathbf{z}\mathbf{z}^\top\right\} = \mathbb{E}\left\{\tilde{\mathbf{A}}\mathbf{y}\mathbf{y}^\top\tilde{\mathbf{A}}^\top\right\} = \tilde{\mathbf{A}}\mathbb{E}\left\{\mathbf{y}\mathbf{y}^\top\right\}\tilde{\mathbf{A}}^\top \\ &= \sigma_y^2\tilde{\mathbf{A}}\tilde{\mathbf{A}}^\top = \sigma_y^2\left(\mathbf{I}_{N\times N}+\theta\left(\mathbf{A}+\mathbf{A}^\top\right)+\theta^2\mathbf{A}\mathbf{A}^\top\right) \quad (7)\end{aligned}$$

where the last before last equality follows from the fact that $\mathbb{E}\{\mathbf{y}\mathbf{y}^\top\} = \sigma_y^2\mathbf{I}_{N\times N}$. Moreover, the covariance matrix for the general GMA(m) in (5) can be obtained by substituting the expression $\mathbf{I}_{N\times N} + \sum_{l=1}^{m}\theta_l\mathbf{A}^l$ into the last before last equality instead of $\tilde{\mathbf{A}}$.

The covariance matrix of the graph signal (6) with inaccurately learned graph adjacency matrix modeled as in (4) can be obtained as

$$\begin{aligned}&\mathbf{C}_z(\mathbf{W}) = \sigma_y^2\left(\mathbf{I}_{N\times N}+\theta\left(\mathbf{W}+\mathbf{W}^\top\right)+\theta^2\mathbf{W}\mathbf{W}^\top\right) \\ &= \mathbf{C}_z(\mathbf{A}) + \sigma_y^2\theta\left(\mathbf{E}+\mathbf{E}^\top+\theta(\mathbf{A}\mathbf{E}^\top+\mathbf{E}\mathbf{A}^\top+\mathbf{E}\mathbf{E}^\top)\right) \quad (8) \\ &= \mathbf{C}_z(\mathbf{A}) + \mathbf{C}_z(\mathbf{E}) + \sigma_y^2\left(\theta^2\left(\mathbf{A}\mathbf{E}^\top+\mathbf{E}\mathbf{A}^\top\right)-\mathbf{I}_{N\times N}\right).\end{aligned}$$

It can be seen from (8) that the covariance matrix of the graph signal (6) with inaccurately learned $\mathbf{A}$ is not only the summation of the covariance matrices of $\mathbf{A}$ and $\mathbf{E}$, but has a third component.

For subsequent GSP tasks, such as for example ICA discussed in Section 4, or even for the shift operator learning itself as in [8], the graph signal autocovariance is of a high importance. It is proportional to a weighted sum of the covariances between the nodes of a graph signal, and serves, for example, as a measure of smoothness with respect to the shift matrix. To derive an expression for the autocovariance of the graph signal given by (6), let us first define the centering matrix $\mathbf{H} \triangleq \mathbf{I}_{N\times N} - \mathbf{1}_{N\times N}/N$. The matrix $\mathbf{H}$ is symmetric and $\mathbf{H}^2 = \mathbf{H}$. The graph signal autocovariance can then be found as

$$\begin{aligned}&\mathbb{E}\left\{\frac{1}{N}(\mathbf{H}\mathbf{z})^\top\mathbf{W}\mathbf{H}\mathbf{z}\right\} = \frac{1}{N}\mathbb{E}\left\{\mathbf{z}^\top\mathbf{H}\mathbf{W}\mathbf{H}\mathbf{z}\right\} \\ &= \frac{1}{N}\mathbb{E}\left\{\mathbf{y}^\top\tilde{\mathbf{A}}^\top\mathbf{H}\mathbf{W}\mathbf{H}\tilde{\mathbf{A}}\mathbf{y}\right\} = \frac{1}{N}\mathrm{tr}\left\{\mathbf{H}\mathbf{W}\mathbf{H}\mathbf{C}_z(\mathbf{A})\right\} \\ &= \frac{\sigma_y^2}{N}\mathrm{tr}\left\{\left(\mathbf{H}+\theta\mathbf{A}^\top\mathbf{H}\right)\mathbf{W}\left(\mathbf{H}+\theta\mathbf{H}\mathbf{A}\right)\right\}. \quad (9)\end{aligned}$$

4. INDEPENDENT COMPONENT ANALYSIS

After introducing the graph signal error models and the GMA signal model, our objective is to investigate the graph error effect on the performance of GSP tasks. In this section, we take ICA of a mixture of GMA signals using the GraDe method [6] as an example of a GSP task.

Let $\mathbf{X} \in \mathbb{R}^{P\times N}$ denote P-dimensional graph signal generated as a mixture of independent components according to the model

$$\mathbf{X} = \mathbf{\Omega}\mathbf{Z} + \boldsymbol{\mu}\mathbf{1}_N^\top \quad (10)$$

where $\mathbf{\Omega} \in \mathbb{R}^{P\times P}$ is a full rank mixing matrix, $\mathbf{Z} \in \mathbb{R}^{P\times N}$ is the matrix of independent components with zero means and unit variances, and $\boldsymbol{\mu} \in \mathbb{R}^P$ is the location vector. We further assume that each of the components satisfy GMA model with unweighted and symmetric $\mathbf{A}$ such that $P(a_{ij} = a) = \alpha$ and $P(a_{ij} = 0) = 1 - \alpha$. The ICA goal is to estimate the unmixing matrix $\mathbf{\Gamma} = \mathbf{\Omega}^{-1}$ using only the data matrix $\mathbf{X}$.

Let $\mathbf{X}_w \triangleq \hat{\mathbf{S}}_0^{-1/2}\left(\mathbf{X} - \bar{\mathbf{X}}\mathbf{1}_N^\top\right)$ be the whitened data, where $\hat{\mathbf{S}}_0$ is the sample covariance matrix of $\mathbf{X}$ and $\bar{\mathbf{X}}$ is the vector of row means of $\mathbf{X}$. In GraDe, the unmixing matrix estimate is obtained by diagonalizing/jointly diagonalizing one or more graph-autocorrelation matrices given as

$$\hat{\mathbf{S}}_k(\mathbf{W}) = \frac{1}{N-k}(\mathbf{X}_w\mathbf{W}^k\mathbf{X}_w^\top), \;\; k = 1, \ldots, K \quad (11)$$

i.e., by finding the orthogonal $\mathbf{U}$ which maximizes the objective function

$$\sum_{k=1}^{K}\|\mathrm{diag}(\mathbf{U}\hat{\mathbf{S}}_k(\mathbf{W})\mathbf{U}^\top)\|^2. \quad (12)$$

The unmixing matrix estimate is then $\hat{\mathbf{\Gamma}} = \mathbf{U}\hat{\mathbf{S}}_0^{-1/2}$. A fast algorithm for the joint diagonalization is available in [14] and applicable for the case when the shift matrix $\mathbf{W}$ is chosen to be symmetric, or the graph-autocorrelation matrices are symmetrized. The unmixing matrix estimate for an inaccurately learned adjacency matrix $\mathbf{W}$ is denoted as $\hat{\mathbf{\Gamma}}(\mathbf{W})$.

We will use the following (see [15] for details) asymptotic result, derived in the context of the second-order blind identification (SOBI) estimator [16], for an unmixing matrix estimate $\hat{\mathbf{\Gamma}}$ obtained using joint diagonalization of matrices $\hat{\mathbf{S}}_1, \ldots, \hat{\mathbf{S}}_K$. When $\mathbf{\Omega} = \mathbf{I}_{P\times P}$, for $i \neq j$, we have

$$\sqrt{N}\,(\hat{\gamma}_{ii} - 1) = -\frac{1}{2}\sqrt{N}\,([\hat{\mathbf{S}}_0]_{ii} - 1) + o_P(1) \quad (13)$$

and

$$\begin{aligned}\sqrt{N}\,\hat{\gamma}_{ij} &= \frac{\sum_k(\lambda_{ki}-\lambda_{kj})(\sqrt{N}\,[\hat{\mathbf{S}}_k]_{ij} - \lambda_{ki}\sqrt{N}\,[\hat{\mathbf{S}}_0]_{ij})}{\sum_k(\lambda_{ki}-\lambda_{kj})^2} \\ &+ o_P(1), \quad (14)\end{aligned}$$

where $\lambda_{ki} \triangleq \mathbb{E}\{[\hat{\mathbf{S}}_k]_{ii}\}$, and $o_P(1)$ stands for negligible terms. The diagonal elements of $\hat{\mathbf{\Gamma}}$ do not depend asymptotically on $\hat{\mathbf{S}}_1, \ldots, \hat{\mathbf{S}}_K$, and thus, in the case of graph signals ICA, do not depend on $\mathbf{W}$. Therefore, the sum of the off-diagonal elements

$$\mathrm{SOV}(\hat{\mathbf{\Gamma}}(\mathbf{W})) = N\sum_{j\neq i}\mathrm{var}(\hat{\mathbf{\Gamma}}(\mathbf{W})_{ij}) \quad (15)$$

can be used when comparing the separation efficiencies induced by different choices of $\mathbf{W}$. We will use the ratio of the sums given as $R(\mathbf{W}_1, \mathbf{W}_2) = \text{SOV}(\hat{\mathbf{\Gamma}}(\mathbf{W}_1))/\text{SOV}(\hat{\mathbf{\Gamma}}(\mathbf{W}_2))$. We have calculated the variances for the GraDe estimate with $K = 1$, when the independent components are GMA(1) signals and the adjacency matrix is symmetric and unweighted, which corresponds to the case of undirected and unweighted graphs.

5. SIMULATION STUDY

The performance of the GraDe method is studied here when only $\mathbf{W}$ (imperfect version of $\mathbf{A}$) is known. The performance is measured using the minimum distance (MD) index [17]

$$D(\hat{\mathbf{\Gamma}}) \triangleq \frac{1}{\sqrt{P-1}} \inf_{\mathbf{C}\in\mathcal{C}} \|\mathbf{C}\hat{\mathbf{\Gamma}}\mathbf{\Omega} - \mathbf{I}_{P\times P}\| \qquad (16)$$

where $\mathcal{C} \triangleq \{\mathbf{C} \ : \ \text{each row and column of } \mathbf{C} \text{ has exactly one non-zero element}\}$. The MD index takes values between zero and one, and it is invariant with respect to the mixing matrix. Also, there is a connection between the minimum distance index and the sum of variances of the off-diagonal elements when $\mathbf{\Omega} = \mathbf{I}_{P\times P}$, given as

$$N(P-1)\mathbb{E}\{D(\hat{\mathbf{\Gamma}})^2\} \to \text{SOV}(\hat{\mathbf{\Gamma}}), \text{ as } N \to \infty. \qquad (17)$$

For two sets of estimates, $\mathbf{W}_1$ and $\mathbf{W}_2$, we define

$$\hat{R}(\mathbf{W}_1, \mathbf{W}_2) = \text{ave}\{D(\hat{\mathbf{\Gamma}}(\mathbf{W}_1))^2\}/\text{ave}\{D(\hat{\mathbf{\Gamma}}(\mathbf{W}_2))^2\}.$$

Equation (17) implies that $\hat{R}(\mathbf{W}_1, \mathbf{W}_2) \approx R(\mathbf{W}_1, \mathbf{W}_2)$ for large N.

Table 1. $R(\mathbf{A}, \mathbf{W})$ for $\mathbf{A}$ with $\alpha = 0.05$ and $\mathbf{W}$ given by $\epsilon_1 = 0, 0.1, \ldots, 0.5$ and $\epsilon_2 = 0, 0.01, \ldots, 0.05$.

$\epsilon_1 \backslash \epsilon_2$	0	0.01	0.02	0.03	0.04	0.05
0	1.00	0.81	0.68	0.58	0.51	0.45
0.1	0.88	0.71	0.58	0.49	0.43	0.38
0.2	0.77	0.60	0.49	0.41	0.35	0.30
0.3	0.66	0.50	0.40	0.33	0.28	0.24
0.4	0.56	0.41	0.31	0.26	0.21	0.18
0.5	0.46	0.31	0.24	0.19	0.15	0.13

Table 2. $\hat{R}(\mathbf{A}, \mathbf{W})$ from 1000 repetitions for $\alpha = 0.05$ and $\epsilon_1 = 0, 0.1, \ldots, 0.5$ and $\epsilon_2 = 0, 0.01, \ldots, 0.05$.

$\epsilon_1 \backslash \epsilon_2$	0	0.01	0.02	0.03	0.04	0.05
0	1.00	0.81	0.67	0.56	0.48	0.44
0.1	0.88	0.65	0.56	0.47	0.43	0.35
0.2	0.76	0.59	0.46	0.40	0.34	0.29
0.3	0.62	0.46	0.37	0.32	0.27	0.25
0.4	0.52	0.37	0.29	0.25	0.21	0.20
0.5	0.43	0.31	0.25	0.20	0.17	0.16

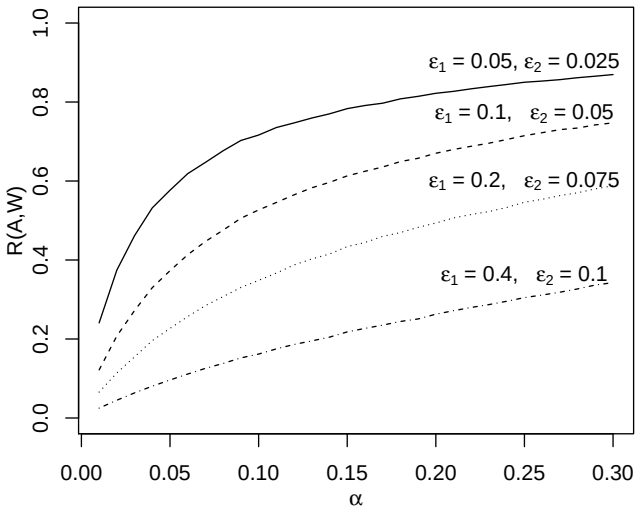

Fig. 1. Ratio of the theoretical variances as a function of α for four choices of (ϵ_1, ϵ_2).

The estimate $\hat{\mathbf{\Gamma}}(\mathbf{A})$ (with true $\mathbf{A}$) is a natural benchmark to which we compare the estimates obtained using $\mathbf{W}$. The matrix $\mathbf{W}$ is generated from $\mathbf{A}$ using the error model (2) with different values of ϵ_1 and ϵ_2. In Tables 1 and 2, the values of $R(\mathbf{A}, \mathbf{W})$ and $\hat{R}(\mathbf{A}, \mathbf{W})$ are shown, respectively, when $\mathbf{A}$ is 1000×1000 matrix with $\alpha = 0.05$ and there are $p = 4$ independent components generated from (6) with $\theta = 0,\ 0.2,\ 0.4$ and 0.6. For Table 2 we generate 1000 datasets for each pair (ϵ_1, ϵ_2) and always generate a new $\mathbf{W}$. In Table 1, the sum of variances is an average for ten $\mathbf{W}$'s, even though $\text{SOV}(\hat{\mathbf{\Gamma}}(\mathbf{W}))$ is quite stable for fixed ϵ_1 and ϵ_2. The simulation results match the theoretical values quite well. The results show that GraDe is more sensitive to adding irrelevant edges than missing the real edges.

For four selected pairs (ϵ_1, ϵ_2), Fig. 1 plots $R(\mathbf{A}, \mathbf{W})$ as a function of α that is used in creating $\mathbf{A}$. The curves display the averages of ten values given by different $\mathbf{W}$'s. As expected, the efficiency loss caused by inaccuracy in the adjacency matrix is the larger, the more sparse the graph is.

6. CONCLUSION

Error models for graph adjacency matrix learning have been introduced. The models are based on Erdös-Rényi model, which can be viewed as an analog to the Gaussian error/noise model in the traditonal signal processing, and therefore, should play a fundamental role in developing GSP algorithms. Also, graph moving average signal model have been discussed and statistically studied by deriving its covariance structure and graph autocovariance. Finally, the graph error models and the GMA signal model have been applied in a graph ICA example, where we used simulations and theoretical results to demonstrate the graph error effect.

7. REFERENCES

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

[2] 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, 2013.

[3] S. Y. Tu and A. H. Sayed, "Mobile adaptive networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 4, pp. 649–664, 2011.

[4] D. Acemoglu, G. Como, F. Fagnani, and A. Ozdaglar, "Opinion fluctuations and disagreement in social networks," *Mathematics of Operations Research*, vol. 38, no. 1, pp. 1–27, 2013.

[5] W. Huang, L. Goldsberry, N. F. Wymbs, S. T. Grafton, D. S. Bassett, and A. Ribeiro, "Graph frequency analysis of brain signals," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 7, pp. 1189–1203, 2016.

[6] F. Blöchl, A. Kowarsch, and F. J. Theis, "Second-order source separation based on prior knowledge realized in a graph model," in *"Latent Variable Analysis and Signal Separation", LNCS* (V. Vigneron, V. Zarzoso, E. Moreau, R. Gribonval, and E. Vincent, eds.), vol. 6365, (Heidelberg), pp. 434–441, Springer, 2010.

[7] X. Dong, D. Thanou, P. Frossard, and P. Vandergheynst, "Learning Laplacian matrix in smooth graph signal representations," *IEEE Transactions on Signal Processing*, vol. 64, no. 23, pp. 6160–6173, 2016.

[8] S. P. Chepuri, S. Liu, G. Leus, and A. O. Hero, "Learning sparse graphs under smoothness prior," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6508–6512, 2017.

[9] P. Erdös and A. Rényi, "On random graphs," *Publ. Math. Debrecen*, vol. 6, pp. 290–297, 1959.

[10] M. W. Morency and G. Leus, "Signal processing on kernel-based random graphs," in *European Signal Processing Conference (EUSIPCO), Kos island, Greece*, pp. 385–2389, 2017.

[11] R. Albert and A. L. Barabási, "Statistical mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, no. 1, pp. 47–97, 2002.

[12] L. Lovász, "Large networks and graph limits," *Providence: American Mathematical Society*, vol. 60, 2012.

[13] 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, 2014.

[14] D. Clarkson, "A least squares version of algorithm as 211: The f-g diagonalization algorithm," *Applied Statistics*, vol. 37, pp. 317–321, 1988.

[15] J. Miettinen, K. Illner, K. Nordhausen, H. Oja, S. Taskinen, and F. Theis, "Separation of uncorrelated stationary time series using autocovariance matrices," *Journal of Time Series Analysis*, vol. 37, no. 3, pp. 337–354, 2016.

[16] A. Belouchrani, K. Abed-Meraim, J. F. Cardoso, and E. Moulines, "A blind source separation technique using second-order statistics," *IEEE Transactions on Signal Processing*, vol. 45, no. 2, pp. 434–444, 1997.

[17] P. Ilmonen, K. Nordhausen, H. Oja, and E. Ollila, "A new performance index for ICA: Properties computation and asymptotic analysis," in *"Latent Variable Analysis and Signal Separation", LNCS* (V. Vigneron, V. Zarzoso, E. Moreau, R. Gribonval, and E. Vincent, eds.), vol. 6365, (Heidelberg), pp. 229–236, Springer, 2010.