ANNIHILATION FILTER APPROACH FOR ESTIMATING GRAPH DYNAMICS FROM DIFFUSION PROCESSES

Arun Venkitaraman and Pascal Frossard

Signal Processing Laboratory LTS4, EPFL, Switzerland

ABSTRACT

We propose an approach for estimating graph diffusion processes using annihilation filters from a finite set of observations of the diffusion process made at regular intervals. Our approach is based on the key observation that a graph diffusion process can be entirely estimated by estimating the eigenvalues and the contributions from the corresponding eigenvectors of the graph-Laplacian, that we achieve through the use of annihilation filters applied in a node-wise manner. We show that the diffusion process can be exactly estimated when the number of samples exceeds $2N+1$, where N is the number of nodes in the graph. We further show how our approach can be used to explicitly learn the underlying graph using an eigenvector proxy. We demonstrate the potential of our approach using experiments with synthesized small-world graphs and real-world network time series data.

***Index Terms*—** Graph signal processing, graph diffusion, annihilation filters, graph learning.

1. INTRODUCTION

Data or signals in a large variety of man-made and naturally occuring phenomena appear over graphs or networks, where every node or agent of the graph has an associated signal value. The edges of the graph capture the interaction between the nodes of the graph [1]. The temporal evolution of signals over the graph is intimately related to and governed by the connections between the nodes, and may be seen as the result of diffusion of information between the nodes and their neighbours. Social networks, biological networks, sensor networks, traffic-flow networks, are some of the real-world examples of such signals. While many different models can be used to mathematically describe the signal evolution over graphs, the heat diffusion model forms one of the simplest, more interpretable, yet effective models thanks to its strong physical intuition and long history of use in study of physical processes [2, 3].

Consider a graph $\mathcal{G}$ with N nodes, with a weighted adjacency matrix $\mathbf{A}=[a_{ij}]_{ij}$ such that a_{ij} denotes the strength of the edge between the ith and jth nodes. Let $\mathbf{L}\triangleq\mathbf{D}-\mathbf{A}$, denote the graph-Laplacian of $\mathcal{G}$, where $\mathbf{D}\triangleq\text{diag}(\mathbf{A1})$ is known as the degree matrix. Let $x(v,t)$ denote the value of the signal at the vth node $v\in[1,\cdots,N]$, at time t. Then, the heat diffusion over graph is governed by the following equation [4]:

$$\frac{\partial\, x(v,t)}{\partial\, t}=-\mathbf{L}\,x(v,t)\text{ such that }x(v,0)=x_0(v),$$

where $x_0(v)$ denotes the signal value at time $t=0$ or the initial state of the system at node v. Expressing the signal value at all the nodes as a vector $\mathbf{x}(t)\triangleq[x(1,t),\cdots,x(N,t)]^\top$, the solution to the diffusion process takes the closed form [4]:

$$\mathbf{x}(t)=e^{-t\,\mathbf{L}}\mathbf{x}(0). \tag{1}$$

Let us assume that we have M observations of the diffusion process $\{\mathbf{x}(nT)\}_{n=1}^{N}$, sampled regularly at T seconds. We ask the question: *Can we estimate the diffusion process exactly?* More specifically, we ask the following:

Given $\{\mathbf{x}(nT)\}_{n=1}^{M}$, *can we estimate* $\{\mathbf{x}(t),\ \forall\, t,\ \mathbf{x}(0),\ \mathbf{L}\}$*?*

We show that it is indeed possible to do so thanks to the special structure of the graph diffusion process through the use of the annihilation filters [5, 6, 7]. In particular, our contribution is an approach that:

- Estimates the graph diffusion process $\mathbf{x}(t)$, $\forall t$ *exactly* when $M>2N+1$, by estimating the eigenvalues and the contributions of the corresponding eigenvectors of the underlying graph-Laplacian matrix through *annihilation filters*.
- Exactly estimates the initial state $\mathbf{x}(0)$, and is *parallelizable* across the nodes.
- Estimates the underlying graph from a proxy of the eigenvectors
- Generalizes to diffusion processes with multiple graphs, that is, when $\mathbf{x}(t)=\sum_{i=l}^{L}e^{-t,\mathbf{L}_l}\mathbf{x}_l(0)$

1.1. Related work

Some of the early works pertaining to graph diffusion deal with diffusion wavelets and multi-resolution analysis on graphs [8, 2]. Diffusion models have since been used in a variety of applications ranging from brain networks to social networks [9, 10]. Diffusion models have been used in solving the graph inference problem in various settings such as kernel identification [11], dictionary learning [4], and graph heat mixture learning [12]. Graph diffusion has also been shown great potential in improving the performance of graph-based learning and estimation techniques, particularly in graph neural networks (GNNs) [13]. Recently, the graph diffusion model was shown to be a unified model for a large variety of graph neural networks demonstrating that GNNs are essentially space-time discretizations of the continuous diffusion equation [3]. To the best of our knowledge, we believe that estimation of graph diffusion dynamics by the use of annihilation filters has never been pursued before.

2. PRELIMINARIES

We now review the basics of graph signal processing and annihilation filters relevant to our discussion.

2.1. Graph signal processing

A graph signal on $\mathcal{G}$ is a vector $\mathbf{x}\in\mathbb{R}^N$ whose components represent the signal at the different nodes of the graph. The smoothness

This work was supported by the SNSF Sinergia project 'PEDESITE: Personalized Detection of Epileptic Seizure in the Internet of Things (IoT) Era'.

978-1-6654-0540-9/22/$31.00 ©2022 IEEE

of a graph signal is typically quantified in terms of the following quadratic form [1]:

$$\mathbf{x}^\top \mathbf{L}\mathbf{x} = \sum_i \sum_j a_{ij}(x(i) - x(j))^2,$$

where $\top$ denotes the transpose operation. The quantity $\mathbf{x}^\top \mathbf{L}\mathbf{x}$ measures the weighted difference of the signal values across connected nodes and captures the natural intuition of smoothness – more similar the signal values across neighbours, smoother that signal. The concept of graph Fourier transform, that generalizes the discrete Fourier transforms to graphs,is routinely used to analyze the smoothness of graph signals. The eigenvectors of the graph-Laplacian are used to define the high-frequency and low-frequency contents of a graph signal. Using the graph smoothness and GFT notions, it can be shown that the signals from a graph diffusion process are smooth graph signals that become smoother with time.

2.2. Annihilation filters

Annihilation filters are central in the analysis of finite-rate-of-innovation (FRI) approaches [5, 6, 7], where the goal is to estimate a signal that is a sum of Dirac pulses. Annihilation filter is used as an intermediate tool to estimate both the strength and the location of the constituent Dirac pulses.

Consider a discrete-time signal $s[n]$ that is a sum of K exponential terms, where n denotes the time-index:

$$s[n] = \sum_{i=1}^{K-1} c_i\, u_i^n, \quad n \in \mathbb{N},\ u_i, c_i \in \mathbb{C}. \tag{2}$$

In order to compute the signal $s[n]$ exactly for any n, all that need are the values of $\{c_i, u_i\}$s. Let us now consider a $(K+1)$-tap finite-impulse response (FIR) filter with impulse response $a[n]$ whose z-transform is given by $A[z] = \sum_{n=0}^{K} a[n] z^{-n} = \Pi_{i=1}^{K}(1 - z^{-1}u_i)$. Then, it can be shown that the filter $a[n]$ annihilates $s[n]$, that is, the output of the convolution of $a[n]$ and $s[n]$ is identically zero [5, 6, 7]:

$$(a * s)[n] = 0 \quad \forall\, n.$$

Owing to this annihilation property, the filter $a[n]$ is referred to as the annihilation filter in literature. Since the annihilation filter must necessarily have the zeros exactly at u_i, the exponents u_i are then found by solving for the roots of the polynomial equation $A[z] = 0$ once the coefficients $a[n]$ are obtained. The filter coefficients are obtained by solving the following linear system of equations:

$$\underbrace{\begin{bmatrix} \vdots & \vdots & \cdots & \vdots \\ s[0] & s[-1] & \cdots & s[-N] \\ \vdots & \vdots & \cdots & \vdots \\ s[K] & s[K-1] & \cdots & s[K-N] \\ \vdots & \vdots & \cdots & \vdots \end{bmatrix}}_{\mathbf{S}} \underbrace{\begin{bmatrix} a[0] \\ a[1] \\ \vdots \\ a[K+1] \end{bmatrix}}_{\mathbf{a}} = 0, \tag{3}$$

by setting $a[0] = 1$ [5, 6]. Once the exponents u_i are computed, the values of c_i are obtained by solving the following linear system of equations:

$$\underbrace{\begin{bmatrix} u_0 & u_1 & \cdots & u_{K-1} \\ \vdots & \vdots & \cdots & \vdots \\ u_0^K & u_1^K & \cdots & u_{K-1}^K \end{bmatrix}}_{\mathbf{U}} \underbrace{\begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{K-1} \end{bmatrix}}_{\mathbf{c}} = \underbrace{\begin{bmatrix} s[1] \\ s[1] \\ \vdots \\ s[K] \end{bmatrix}}_{\mathbf{s}}. \tag{4}$$

Thus, $s[n]$ is completely determined by solving two systems of linear equations given by (3) and (4). Since there are a total of $2K+1$ unknowns in these equations, the signal $s[n]$ can be exactly estimated if the number of samples of $s[n]$ is greater than $2K+1$. We note also that the absolute index of the samples does not matter in the estimation of the exponents, but only affects the values of c_is. In practice due to numerical considerations and the presence of additive noise, regularized variants of (3) and (4) are used [5, 6, 7].

3. PROPOSED APPROACH

Let us now return to the solution of the graph diffusion process given by (1). In our analysis, we make the assumption that the graph-Laplacian is diagonalizable, that is, all the *eigenvalues of the graph are distinct*. Let $\mathbf{L} = \mathbf{V}\boldsymbol{\lambda}\mathbf{V}^\top$ denote the eigenvalue decomposition of the graph-Laplacian, where $\boldsymbol{\lambda} = \mathrm{diag}(\lambda_1 \cdots, \lambda_N)$ and $\mathbf{V} = [\mathbf{v}_1, \cdots, \mathbf{v}_N]^\top$, λ_i being the ith eigenvalue of $\mathbf{L}$, and $\mathbf{v}_i$ the corresponding eigenvector. Then, (1) may be expressed as

$$\mathbf{x}(t) = e^{-t\,\mathbf{L}}\mathbf{x}(0) = \mathbf{V}e^{-t\,\boldsymbol{\lambda}}\mathbf{V}^\top\mathbf{x}(0) = \sum_{i=1}^{N} \boldsymbol{\alpha}_i e^{-\lambda_i\, t}, \tag{5}$$

where $\boldsymbol{\alpha}_i = \mathbf{v}_i^\top\mathbf{x}(0)$. The graph diffusion process takes form of a vector-weighted sum of N exponentials. In particular, the signal at the vth node takes the form:

$$x(v,t) = \sum_{i=1}^{N} \boldsymbol{\alpha}_i(v) e^{-\lambda_i\, t}.$$

Let us now assume that we have the samples $\mathbf{x}(nT)$ for some known $T > 0$. By a slight abuse of notation, let $\mathbf{x}[n]$ denote the discrete time samples of $\mathbf{x}(t)$, and similarly, let $x[v,n]$ denote $x(v,nT)$. Then, the discrete-time signal $\mathbf{x}[n]$ takes the form of a sum of discrete-time exponentials as: $\mathbf{x}[n] = \sum \boldsymbol{\alpha}_{i=1}^{N} u_i^n$, where $u_i = e^{-\lambda_i\, T}$. For the vth node, this becomes

$$\mathbf{x}[v,n] = \sum \boldsymbol{\alpha}_i(v) u_i^n. \tag{6}$$

On comparing with (2), we observe that $\mathbf{x}[v,n]$ is expressible as N discrete-time exponential signals, with $K = N$, $c_i = \boldsymbol{\alpha}_i(v)$, and $u_i = e^{-\lambda_i\, T}$. This in turn implies that the annihilation filter approach of Section 2.2 can be applied at every node to gives estimate of the eigenvalues of the graph-Laplacian as

$$\hat{\lambda}_i = -(1/T)\, \ln\, u_i,$$

and the projection of the initial state onto the corresponding eigenvectors $\boldsymbol{\alpha}_i(v)$. Since the exponents to be estimated are the same for every node, they need to be estimated only once. Once all the eigenvalues λ_is and the vector-weights $\boldsymbol{\alpha}_i$s are obtained, we get the *estimate for* $\mathbf{x}(t)$ *for any* $t \geq 0$ through (5). Consequently, we have the following proposition:

Proposition 1 *A graph diffusion on a graph with N nodes can be exactly estimated with $M > 2N+1$ regularly samples of the graph diffusion process.*

We also note that we estimate the graph diffusion process *without the explicit estimation of the graph-Laplacian*. Further, we note that since $\boldsymbol{\alpha}_i$s are the projections of the initial state $\mathbf{x}(0)$ onto different eigenvectors, we have that $\sum_{i=1}^{N} \boldsymbol{\alpha}_i = (\sum_{i=1}^{N} \mathbf{v}_i\mathbf{v}_i^\top)\mathbf{x}(0) = \mathbf{x}(0)$. In the analysis above, we have implicitly assumed that all $\boldsymbol{\alpha}_i$s are non-zero to be able to fully identify $\mathbf{L}$. This requires that $\boldsymbol{\alpha}_i(v) \neq 0$, which in turn requires that for a given $\mathbf{L}$, we must have $\mathbf{x}(0)^\top \mathbf{v}_i > 0$ for $i = 1, \cdots, N$. That is, the initial state $\mathbf{x}(0)$ is constrained to be not orthogonal to any of the eigenvectors of $\mathbf{L}$. This constraint is related to the important property of persistence of excitation of the signal $\mathbf{x}(t)$ known in the context of linear system identification [14, 15].

3.1. Graph-learning

In our analysis so far, we have not made an explicit estimation of the underlying graph, that is, of the graph-Laplacian $\mathbf{L}$. In many cases, it is also important that we are able to estimate the underlying graph topology explicitly. We note that our approach estimates the eigenvalues directly, but not the eigenvectors.We note that while the information of the eigenvectors is inherernt in the weights $\boldsymbol{\alpha}_i$s, obtaining $[\mathbf{v}_1, \cdots, \mathbf{v}_N]$ from $[\boldsymbol{\alpha}_1, \cdots, \boldsymbol{\alpha}_N]$ is not trivial in practice. This is primarily because the different $\boldsymbol{\alpha}_i$s have not been constrained to be orthogonal in the estimation. If indeed they were orthogonal, one could arrive at the eigenvectors $\mathbf{v}_i$ from $\boldsymbol{\alpha}_i$ by scaling them to have unit-norm, since we have

$$\boldsymbol{\alpha}_i = \|\mathbf{v}_i\|_2 \|\mathbf{x}(0)\|_2 \cos(\theta_i)\mathbf{v}_i = \cos(\theta_i)\|\mathbf{x}(0)\|_2 \mathbf{v}_i.$$

Since $\boldsymbol{\alpha}_i$s are not constrained to be orthogonal, the estimates of the eigenvectors obtained by scaling too will not be orthonormal in general. Indeed they typically turn out to be a highly linearly dependent set of vectors, and very dissimilar to the ground-truth eigenvectors in numerical experiments. For this reason, we pursue graph learning by assuming that a suitable proxy of the eigenvectors is available, as pursued in [16]. Once a proxy is available, we construct the estimate of the graph-Laplacian from the eigenvalues obtained using annihilation filters. We strongly believe that the orthonormal eigenvectors may be directly estimated from our approach by considering a regularized estimation where all the $\boldsymbol{\alpha}_i$s are regularized to be orthonormal in a spirit similar to the orthogonal Procrustes approach [17, 18]. This will in turn obviate the need for a proxy. We will be pursuing this in the near future.

3.2. Sequential estimation of the eigenvalues

While the annihilation filter approach works for any exponential function in theory, numerical experiments in practice often produce less accurate estimates of the eigenvalues belonging to the 'mid-frequency' range. This becomes more noticeable as the number of nodes becomes large. This is so because the eigenvalues in many of these graphs, even though distinct, tend to be similar in value to each other. It is known that the eigenvalue profiles of many well-known graphs tend to be smooth [19]. As a result, it becomes harder to resolve the eigenvalues numerically, with the added challenge posed by the exponential basis $e^{-\lambda\, t}$ being non-orthogonal for different λ.

In order to obtain improved eigenvalue estimates, we propose a sequential or block variant to our approach, by performing the annihilation filter operation over smaller subsets of size $N_b < N$, sequentially. In other words, instead of estimating all the N exponentials at once, we successively estimate N_b exponentials, subtract their contribution from the original signal, then estimate the next N_b exponentials from the residue and repeat the process till N exponentials are estimated. While theoretically this is not exactly equivalent to estimating all the exponentials at once, we observe that the sequential approach consistently estimates the eigenvalues better, especially as N_b is made small. In our experiments, we consider $N_b = N$, $N_b = N/2$, and $N_b = N/4$. However, since $N_b = N/4$ always results in the smallest error in eigenvalue estimation, we report the results only for that case in this paper.

3.3. Generalization to diffusion with multiple graphs

Let us now consider the case where the diffusion process is a resultant of diffusions coming from multiple graphs, such that $\mathbf{x}(t) = \sum_{l=1}^{L} e^{-t\,\mathbf{L}_l}\mathbf{x}_l(0)$, where $\mathbf{L}_l$ denote the graph-Laplacians of the different constituent graphs and $\mathbf{x}_l(0)$, the corresponding initial states. Such a signal may describe the case of graph mixture models [4, 12], where multiple graphs exist one the same set of nodes and the signal at any given time comes from one of the I graphs $-$ $\mathbf{x}_l(0)$ then corresponds to the effective initial state of the ith graph given by the true initial state multiplied by the prior probability of the graph. Once again, this signal can be expressed as a sum of exponentials as

$$\mathbf{x}(t) = \sum_{l=1}^{L}\sum_{i=1}^{N} \boldsymbol{\alpha}_{l,i} e^{-\lambda_{l,i}\, t}, \tag{7}$$

where $\boldsymbol{\alpha}_{l,i}$ is the vector-weight corresponding to the ith eigen-value $\lambda_{l,i}$of the lth graph. Hence, our approach is directly applicable to this setting. However, it is not trivial to map the estimated eigenvalues (and hence the corresponding vector-weights) uniquely to one of the L graphs $-$ $\mathbf{x}(t)$ can be estimated exactly from its samples, but not its constituent graphs unlike in the single graph case. This is due to the non-uniqueness of the ordering of the exponentials. In the case of single graph, the non-uniqueness does not pose a challenge as one can choose a convention for ordering the eigenvalues and the corresponding eigenvectors/spectral templates.

4. EXPERIMENTS

We apply our approach to both synthesized and real-world graph signal examples.

4.1. Synthesized small-world graph signals

We consider experiments on three different synthesized graph classes, namely, Barabási-Albert (BA), Erdős-Rényi (ER), and random geometric (RG) graphs. In each case, we randomly generate the graph and generate the diffusion process starting from a random initial state $\mathbf{x}(0)$ (drawn randomly from the multi-variate normal distribution). We consider regular samples of M points over the time interval $[10^{-1}, 10^{1}]$. The eigenvectors are obtained as the eigenvectors of the covariance matrix computed from many random time samples of diffusion processes starting from random initial states, drawn from a multivariate normal distribution. All the graphs are normalized to have the maximum eigenvalue of $\mathbf{L}$ to be equal to unity, and ordered in descending order of magnitude. We measure the performance of our approach in terms of the following metrics: (1) Mean-squared error (MSE) in eigenvalue estimation $\frac{E\|\hat{\boldsymbol{\lambda}}-\boldsymbol{\lambda}\|_2^2}{E\|\boldsymbol{\lambda}\|_2^2}$, (2) MSE in initial-state estimation: $\frac{E\|\hat{\mathbf{x}}[0]-\mathbf{x}[0]\|_2^2}{E\|\mathbf{x}[0]\|_2^2}$, and (3) F-score with respect to the ground-truth graph, where the expectation E is obtained by averaging over multiple realizations of the graphs. We compare the performance of our approach with the popular graph-learning approach in [20] (that we denote by GL). We wish to emphasize that this comparison is for illustrating the potential of our approach, and note that the GL approach is not particularly tuned to handle diffusion signals. All the adjacency matrices are sparsified by suitable thresholding (typically around 10^{-2}).

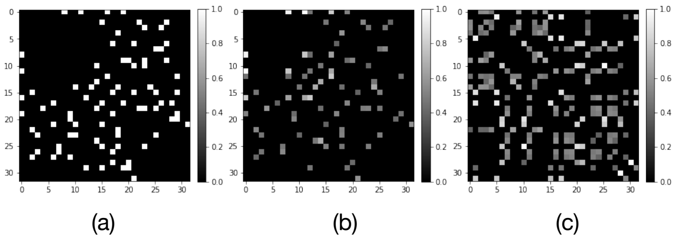

Fig. 1. The adjacency matrix for an instance of synthesized random graph. (a) Ground truth, (b) Our approach, and (c) GL approach.

Graph class	$E\|\hat{\boldsymbol{\lambda}}-\boldsymbol{\lambda}\|_2^2/E\|\boldsymbol{\lambda}\|_2^2$		F-score	
	Ours	GL	Ours	GL
ER	**0.35**	1.2	**0.89**	0.31
BA	**0.14**	0.72	**0.75**	0.34
RG	**0.16**	0.53	**0.85**	0.49

Table 1. Comparison of metrics for single graph diffusion process.

We observe from Table 1, that our approach performs well in terms of the two metrics when $M = 8N$, and for sequential blocks of $N/4$ components each. Since the values obtained for $\frac{E\|\hat{\mathbf{x}}[0]-\mathbf{x}[0]\|_2^2}{E\|\mathbf{x}[0]\|_2^2}$ are always small in the order $\mathbf{10^{-3}}$, we do not report them explicitly. The high F-score further indicates that the learnt graph almost coincides with the ground-truth. An instance of the estimated graph is shown in Figure 1. We observe that GL learns a graph that is fairly close to a fully connected graph even when the true graph is sparse. This is not surprising as the signals used for learning the graph have been obtained by successively diffusing the initial signal over time, and hence, signal values over all the nodes tend to become more similar as the diffusion progresses. We also repeat the experiment for the case of diffusion with two graphs, and report the metrics in Table 3. We observe the general trend that error in eigenvalues is primarily contributed by the eigenvalues that correspond to the 'mid-frequency' components of the graph, that tend to be close in value and hence, harder to resolve for the annihilating filter. We also observe that this error reduces as M is increased. Further, ER graphs follow a distribution of eigenvalues which is close to the semi-circular profile and smooth, unlike the BA and RG graphs that have skewed eigenvalue profiles [21, 22], possibly accounting for larger error in eigenvalue estimation.

We further consider a simple graph classification experiment: we use only the eigenvalues estimated as the feature to classify the graph as belonging to one of the three class of small-world graphs. We classify on the basis of nearest neighbour - we measure the distance of the eigenvalues of a given process with the mean of the eigenvalues obtained for the three different classes (with training 250 training and 250 test graphs per class). We found that the average test classification accuracy to be around $\mathbf{75\%}$. The same classification experiment when done with the true eigenvalues results in a 95% average test accuracy.

$E\|\hat{\boldsymbol{\lambda}}-\boldsymbol{\lambda}\|_2^2/E\|\boldsymbol{\lambda}\|_2^2$	ER	BA	Random
	0.38	0.26	0.24

Table 2. MSE in eigenvalue estimation for multiple graph diffusion process ($L = 2$).

Dataset	$E\|\hat{\boldsymbol{\lambda}}-\boldsymbol{\lambda}\|_2^2/E\|\boldsymbol{\lambda}\|_2^2$		F-score	
	Ours	GL	Ours	GL
WikiMath	**0.34**	0.72	**0.53**	0.16
Windmill Medium	**0.14**	0.86	**0.86**	0.39
Windmill Large	**0.17**	0.85	**0.73**	0.38

Table 3. Comparison of metrics for real-world datasets.

5. REAL-WORLD SPATIO-TEMPORAL NETWORK SIGNALS

We next apply our approach to three network time-series signal datasets publicly available from the PyTorch Geometric Temporal library [23]. These datasets consist of time-varying signals over static network/graph. These datasets have recently become popular as benchmarks for spatio-temporal graph-based deep learning. The datasets are as follows:

- *Wiki Maths Dataset:* The graph is made up of vital mathematics articles from Wikipedia, edges representing the links between them. The signal at the nodes denotes the daily number of users. We consider a subgraph of 50 articles and consider a symmetricized version of the originally directed graph.
- *Windmill Medium Dataset:* The graph is made up of windmills across Europe as nodes, edges representing the strength of relationships between them. The signal at the nodes denotes the hourly energy output of the windmills. This dataset consists of 26 nodes
- *Windmill Medium Dataset:* This is an extended dataset of the windmill data with over 300 windmills, of which we consider 50 in our analysis.

In every dataset, we consider set of first 200 consecutive time-signals and perform the graph estimation with our approach and with GL. We used a sequential estimation with block size of $N/4$. We consider the eigenvectors of the covariance matrix as the eigenvector proxy, where the covariance matrix is constructed using additional 300 time-samples. We observe from Table 3 that our approach results in better F-score and eigenvalue estimation than the GL approach. The relatively poorer performance on the first dataset may be owing to the undirected graph assumption that we have made, given that the original graph is directed. We wish to note that in the case of these datasets, the signals are not known to be an exact diffusion process or smooth graph signals. This explains why the absolute F-score is not very high. Nevertheless, they help evaluate the potential of our approach in diverse real-world settings.

6. CONCLUSIONS

We proposed an approach for estimating the graph diffusion process from a finite number of samples of the process using annihilation filters. Experiments on synthesized small-world graphs show great potential in terms of both learning the underlying graph and also for graph classification. Our approach was also shown to suitable in diverse real-world spatio-temporal data settings. It is however desirable to estimate the eigenvectors directly from the observations without the need for a proxy or spectral template. There is also the need to improve the estimation of mid-frequency eigenvalues by use of more stable root-finding approaches. We intend to pursue these directions in the future.

7. REFERENCES

[1] Antonio Ortega, Pascal Frossard, Jelena Kovačević, José M. F. Moura, and Pierre Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, 2018.

[2] Ronald R. Coifman and Mauro Maggioni, "Diffusion wavelets," *Applied and Computational Harmonic Analysis*, vol. 21, no. 1, pp. 53–94, 2006, Special Issue: Diffusion Maps and Wavelets.

[3] Benjamin Paul Chamberlain, James Rowbottom, Maria Gorinova, Stefan Webb, Emanuele Rossi, and Michael M. Bronstein, "Grand: Graph neural diffusion," 2021.

[4] Dorina Thanou, Xiaowen Dong, Daniel Kressner, and Pascal Frossard, "Learning heat diffusion graphs," *IEEE Trans. Signal Inf. Process. over Networks*, vol. 3, no. 3, pp. 484–499, 2017.

[5] M. Vetterli, P. Marziliano, and T. Blu, "Sampling signals with finite rate of innovation," *IEEE Transactions on Signal Processing*, vol. 50, no. 6, pp. 1417–1428, 2002.

[6] Thierry Blu, Pier-Luigi Dragotti, Martin Vetterli, Pina Marziliano, and Lionel Coulot, "Sparse sampling of signal innovations," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 31–40, 2008.

[7] I. Maravic and M. Vetterli, "Sampling and reconstruction of signals with finite rate of innovation in the presence of noise," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 2788–2805, 2005.

[8] Sridhar Mahadevan and Mauro Maggioni, "Value function approximation with diffusion wavelets and laplacian eigenfunctions," *Advances in Neural Information Processing Systems*, pp. 843–850, 2005.

[9] Weiyu Huang, Thomas A. W. Bolton, John D. Medaglia, Danielle S. Bassett, Alejandro Ribeiro, and Dimitri Van De Ville, "A graph signal processing perspective on functional brain imaging," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 868–885, 2018.

[10] Hao Ma, Haixuan Yang, Michael R. Lyu, and Irwin King, "Mining social networks using heat diffusion processes for marketing candidates selection," *Proceedings of the ACM Conference on Information and Knowledge Management*, p. 233–242, 2008.

[11] Hilmi E. Egilmez, Eduardo Pavez, and Antonio Ortega, "Graph learning from filtered signals: Graph system and diffusion kernel identification," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 5, no. 2, pp. 360–374, 2019.

[12] Hermina Petric Maretic, Mireille El Gheche, and Pascal Frossard, "Graph heat mixture model learning," *Asilomar Conference on Signals, Systems, and Computers*, pp. 1003–1007, 2018.

[13] Johannes Klicpera, Stefan Weißenberger, and Stephan Günnemann, "Diffusion improves graph learning," *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 2019.

[14] T. Söderström and P. Stoica, *System Identification*, chapter 5, Prentice-Hall, Inc., USA, 1988.

[15] Padoan et al., "A geometric characterization of the persistence of excitation condition for the solutions of autonomous systems," *IEEE Trans. Autom. Control.*, vol. 62, no. 11, pp. 5666–5677, 2017.

[16] Santiago Segarra, Antonio G. Marques, Gonzalo Mateos, and Alejandro Ribeiro, "Network topology identification from spectral templates," *2016 IEEE Statistical Signal Processing Workshop (SSP)*, pp. 1–5, 2016.

[17] J. C. Gower and G. B. Dijksterhuis, *Procrustes problems*, vol. 30 of *Oxford Statistical Science Series*, Oxford University Press, Oxford, UK, January 2004.

[18] R. Everson, "Orthogonal, but not orthonormal, procrustes problems," in *Advances in Computational Mathematics. Available from http://www.ee.ic.ac.uk/research/neural/everson*, 1997.

[19] M. E. J. Newman, "The structure and function of complex networks," *SIAM Review*, vol. 45, no. 2, pp. 167–256, Jan. 2003.

[20] Xiaowen Dong, Dorina Thanou, Pascal Frossard, and Pierre Vandergheynst, "Learning laplacian matrix in smooth graph signal representations," *IEEE Transactions on Signal Processing*, vol. 64, no. 23, pp. 6160–6173, 2016.

[21] Farkas et al., "Spectra of "real-world" graphs: Beyond the semicircle law," *Phys. Rev. E*, vol. 64, pp. 026704, Jul 2001.

[22] Dorogovtsev et al., "Spectra of complex networks," *Phys. Rev. E*, vol. 68, pp. 046109, Oct 2003.

[23] Benedek Rozemberczki, Paul Scherer, Yixuan He, George Panagopoulos, Alexander Riedel, Maria Astefanoaei, Oliver Kiss, Ferenc Beres, Guzmán López, Nicolas Collignon, and Rik Sarkar, *PyTorch Geometric Temporal: Spatiotemporal Signal Processing with Neural Machine Learning Models*, p. 4564–4573, Association for Computing Machinery, New York, NY, USA, 2021.