ONLINE IDENTIFICATION OF DIRECTIONAL GRAPH TOPOLOGIES CAPTURING DYNAMIC AND NONLINEAR DEPENDENCIES[†]

Yanning Shen and Georgios B. Giannakis

Dept. of ECE and DTC, University of Minnesota, Minneapolis, MN 55455, USA

ABSTRACT

Linear structural vector autoregressive models (SVARMs) have well-documented merits for topology inference of directional graphs emerging in diverse applications, including gene-regulatory, brain, and social networks. Although simple and tractable, linear SVARMs cannot capture nonlinearities that are inherent to complex systems, such as the human brain, that can also vary over time. Given nodal measurements, these considerations motivate the dynamic nonlinear SVARM approach developed here to track the possibly directed and dynamic nonlinear interactions among network nodes. For slow-varying topologies, nonlinear model parameters are estimated via functional stochastic gradient descent. Numerical tests showcase the effectiveness of the novel algorithms in unveiling sparse dynamically-evolving topologies.

***Index Terms*—** Network topology inference, structural vector autoregressive models, nonlinear, dynamics

1. INTRODUCTION

Graph topology inference plays a crucial role in numerous applications, such as financial networks, brain, social, and gene-regulatory networks, to name a few. In these networks, the edges may not be available, but observations can be collected at nodes. For example, one may have access to time series of stock prices for all stocks, yet the dependency structure between them is hidden. Granger causal models, vector autoregressive (VAR) models, and structural equation models (SEMs) are widely adopted to infer directed network topologies; see e.g., [5, 7, 11], and references therein. VAR models postulate that causal relationships are captured through time-lagged dependencies between nodal time series, while SEMs are based on instantaneous interactions. On the other hand, structural vector autoregressive models (SVARMs) [3] offer a unifying approach, postulating that nodal time series result from both instantaneous and time-lagged interactions, thus adding generality relative to either VARs or SEMs.

Contemporary SVARMs rely on linear and static models, due to their simplicity and tractability. However, resorting to linear and static SVARMs is limiting, since interactions within complex systems (e.g., the human brain) are generally nonlinear, and time-varying. Recently, several variants of nonlinear SEMs and VARs have been advocated by e.g., [5, 8, 10, 12, 18, 20] and references therein.

Besides nonlinearities, network connectivity can also vary dynamically, while nodal measurements may also arrive sequentially. To cope with these challenges, online algorithms capable of tracking dynamic network topologies have been pursued. Dynamic SEMs for instance were advocated for topology identification of time-varying directed networks in [2, 19]; see also [1, 5, 6].

The present paper considerably broadens the scope of prior art, putting forth a general *online nonlinear* SVARM for identification of *directed* and *dynamic* network topologies. The developed estimator leverages kernels as an encompassing framework for nonlinear topology learning, while a stochastic gradient descent functional iteration is developed for tracking the possibly dynamic network topology, and estimating the underlying nonlinear interactions in real time.

2. PRELIMINARIES ON LINEAR SVARMS

Consider a directed graph with unknown topology, comprising N nodes, each associated with an observable time series $\{y_{it}\}_{t=1}^{T}$ measured over T slots, per node $i = 1, \ldots, N$. In the context of brain networks, each node could represent a region of interest (RoI), while the per-RoI time course is formed using standard imaging modalities, e.g., EEG or fMRI data. The network topology or edge weights is captured by the weighted adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$, whose (i, j)-th entry a_{ij} is nonzero, only if a directed (causal) influence is effected from region i to region j.

In order to unveil the hidden causal network topology, traditional linear SVARMs postulate that y_{jt} at node j is represented as a linear combination of instantaneous measurements at nodes other than j, namely $\{y_{it}\}_{i \neq j}$, and their time-lagged versions $\{\{y_{i(t-\ell)}\}_{i=1}^{N}\}_{\ell=1}^{L}$ [3]. Specifically, y_{jt} admits the following linear instantaneous plus time-lagged model

$$y_{jt} = \sum_{i \neq j} a_{ij}^{(0)} y_{it} + \sum_{i=1}^{N} \sum_{\ell=1}^{L} a_{ij}^{(\ell)} y_{j(t-\ell)} + e_{jt} \qquad (1)$$

with $a_{ij}^{(\ell)}$ capturing the causal influence of node i upon node

[†] Work was supported by NSF grants 1500713, 1514056, and 1711471.

j over a lag of ℓ time points, while $a_{ij}^{(0)}$ encodes the corresponding instantaneous causal relationship between them. The coefficients encode the causal structure of the network, that is, a causal link is present between nodes i and j only if $a_{ij}^{(0)} \neq 0$, or if there exists $a_{ij}^{(\ell)} \neq 0$ for $\ell = 1, \ldots, L$. If $a_{ij}^{(\ell)} = 0 \ \forall i, j, \ell \neq 0$ reduces (1) to a linear SEM with no exogenous inputs [11]. Defining $\mathbf{y}_t := [y_{1t}, \ldots, y_{Nt}]^\top$, $\mathbf{e}_t := [e_{1t}, \ldots, e_{Nt}]^\top$, and the time-lagged adjacency matrix $\mathbf{A}^{(\ell)} \in \mathbb{R}^{N \times N}$ with the (i,j)-th entry $\left[\mathbf{A}^{(\ell)}\right]_{ij} := a_{ij}^{(\ell)}$, one can write (1) in vector-matrix form as

$$\mathbf{y}_t = \mathbf{A}^{(0)}\mathbf{y}_t + \sum_{\ell=1}^{L} \mathbf{A}^{(\ell)}\mathbf{y}_{t-\ell} + \mathbf{e}_t \tag{2}$$

where $\mathbf{A}^{(0)}$ has zero diagonal entries $\{a_{ii}^{(0)} = 0\}_{i=1}^{N}$.

Given the multivariate time series $\{\mathbf{y}_t\}_{t=1}^{T}$, the goal is to estimate matrices $\{\mathbf{A}^{(\ell)}\}_{\ell=0}^{L}$, and consequently unveil the hidden network topology. The memory length L is prescribed, which can be determined by standard order selection methods, e.g., the Bayesian information criterion [4]. It is also worth noting that most real world networks exhibit edge sparsity, the tendency for each node to link with only a few other nodes compared to the maximal $\mathcal{O}(N)$ set of potential connections per node. This means that per j, only a few coefficients $\{a_{ij}^{(\ell)}\}$ are nonzero. In fact, several recent approaches exploiting edge sparsity have been advocated, leading to more efficient topology identification schemes; see e.g., [1, 2].

3. FROM LINEAR TO NONLINEAR SVARMS

To enhance flexibility, we introduced a nonlinear generalization of (1) in [17]. Specifically, we postulated that node j's observation at time t is a result of both instantaneous and multi-lag effects; that is [cf. (1)]

$$y_{jt} = \sum_{i \neq j} a_{ij}^{(0)} f_{ij}^{(0)}(y_{it}) + \sum_{i=1}^{N}\sum_{\ell=1}^{L} a_{ij}^{(\ell)} f_{ij}^{(\ell)}(y_{i(t-\ell)}) + e_{jt} \tag{3}$$

where similar to (1), $\{a_{ij}^{(\ell)}\}$ define the matrices $\{\mathbf{A}^{(\ell)}\}_{\ell=0}^{L}$. As before, a directed edge from node j to node i exists if the corresponding $a_{ij}^{(\ell)} \neq 0$ for $\ell = 0, 1, \ldots, L$. Note that conventional linear SVARMs in (1) assume that the functions $\{f_{ij}^{(\ell)}\}$ in (3) are linear, a limitation that we avoid by resorting to a reproducing kernel Hilbert space (RKHS) formulation to model $\{f_{ij}^{(\ell)}\}$.

Let each univariate $f_{ij}^{(\ell)}(.)$ in (3) belong to the RKHS

$$\mathcal{H}_i^{(\ell)} := \{f_{ij}^{(\ell)} | f_{ij}^{(\ell)}(y) = \sum_{t=1}^{\infty} \beta_{ijt}^{(\ell)} \kappa_i^{(\ell)}(y, y_{i(t-\ell)})\} \tag{4}$$

where $\kappa_i^{(\ell)}(y, \psi) : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ is a preselected basis (so-termed kernel) function that measures the similarity between y and ψ. Different choices of $\kappa_i^{(\ell)}$ specify their own spaces, and the linear functions can be regarded as a special case associated with $\kappa_i^{(\ell)}(y, \psi) = y\psi$. An alternative popular kernel is the Gaussian one that is given by $\kappa_i^{(\ell)}(y, \psi) := \exp[-(y - \psi)^2/(2\sigma^2)]$. Defining the inner product as $\langle \kappa_i^{(\ell)}(y, \psi_1), \kappa_i^{(\ell)}(y, \psi_2) \rangle := \sum_\tau \kappa_i^{(\ell)}(y_\tau, \psi_1)\kappa_i^{(\ell)}(y_\tau, \psi_2)$, a kernel is reproducing if it satisfies $\langle \kappa_i^{(\ell)}(y, \psi_1), \kappa_i^{(\ell)}(y, \psi_2) \rangle = \kappa_i^{(\ell)}(\psi_1, \psi_2)$, which induces the RKHS norm $\|f_{ij}^{(\ell)}\|^2_{\mathcal{H}_i^{(\ell)}} = \sum_\tau \sum_{\tau'} \beta_{ij\tau}^{(\ell)} \beta_{ij\tau'}^{(\ell)} \kappa_i^{(\ell)}(y_{i\tau}, y_{i\tau'})$ [21].

Considering the measurements per node j, with functions $f_{ij}^{(\ell)} \in \mathcal{H}_i^l$, for $i = 1, \ldots, N$ and $\ell = 0, 1, \ldots, L$, we advocate the following regularized least-squares (LS) estimates of the aforementioned functions obtained as

$$\{\hat{f}_{ij}^{(\ell)}\} = \arg \min_{\{f_{ij}^{(\ell)} \in \mathcal{H}_i^{(\ell)}\}} \frac{1}{2} \sum_{t=1}^{T} \Bigg[y_{jt} - \sum_{i \neq j} a_{ij}^{(0)} f_{ij}^{(0)}(y_{it}) - \sum_{i=1}^{N}\sum_{\ell=1}^{L} a_{ij}^{(\ell)} f_{ij}^{(\ell)}(y_{i(t-\ell)}) \Bigg]^2 + \lambda \sum_{i=1}^{N}\sum_{\ell=0}^{L} \Omega(\|a_{ij}^{(\ell)} f_{ij}^{(\ell)}\|_{\mathcal{H}^{(\ell)}}) \tag{5}$$

where typical choices for the regularization function are $\Omega(z) = |z|$, and $\Omega(z) = z^2$. The former is known to promote sparsity of edges, which is prevalent to most networks; see e.g., [14]. In principle, leveraging such prior knowledge naturally avoids overfitting in topology identification.

According to the representer theorem [9, p. 169], the optimal solution for each $f_{ij}^{(\ell)}$ in (5) is given by

$$\hat{f}_{ij}^{(\ell)}(y) = \sum_{t=1}^{T} \beta_{ijt}^{(\ell)} \kappa_i^{(\ell)}(y, y_{i(t-\ell)}). \tag{6}$$

Although the function spaces in (4) include infinite basis expansions, since the given data are finite, namely T per node, the optimal solution in (6) entails a finite basis expansion. Substituting (6) into (5), and letting $\boldsymbol{\beta}_{ij}^{(\ell)} := [\beta_{ij1}^{(\ell)}, \ldots, \beta_{ijT}^{(\ell)}]^\top$, and $\boldsymbol{\alpha}_{ij}^{(\ell)} := a_{ij}^{(\ell)} \boldsymbol{\beta}_{ij}^{(\ell)}$, the functional minimization in (5) boils down to optimizing over vectors $\{\boldsymbol{\alpha}_{ij}^{(\ell)}\}$. Specifically, (5) can be equivalently written in vector form as

$$\{\hat{\boldsymbol{\alpha}}_{ij}^{(\ell)}\} = \arg \min_{\{\boldsymbol{\alpha}_{ij}^{(\ell)}\}} \frac{1}{2} \Bigg\| \mathbf{y}_j - \sum_{i \neq j} \mathbf{K}_i^{(0)} \boldsymbol{\alpha}_{ij}^{(0)} - \sum_{i=1}^{N}\sum_{\ell=1}^{L} \mathbf{K}_i^{(\ell)} \boldsymbol{\alpha}_{ij}^{(\ell)} \Bigg\|_2^2 + \lambda \sum_{i=1}^{N}\sum_{\ell=0}^{L} \Omega\left(\sqrt{(\boldsymbol{\alpha}_{ij}^{(\ell)})^\top \mathbf{K}_i^{(\ell)} \boldsymbol{\alpha}_{ij}^{(\ell)}} \right) \tag{7}$$

where $\mathbf{y}_j := [y_{j1}, \ldots, y_{jT}]^\top$, and the $T \times T$ matrices $\{\mathbf{K}_i^{(\ell)}\}$ are formed to have entries $[\mathbf{K}_i^{(\ell)}]_{t,\tau} = \kappa_i^{(\ell)}(y_{it}, y_{i(\tau-\ell)})$.

Let $\mathbf{W}_\alpha^{(\ell)} \in \mathbb{R}^{NT \times N}$ denote the block matrix with $\boldsymbol{\alpha}_{ij}^{(\ell)}$ as its (i,j)th block. Clearly, $\mathbf{W}_\alpha^{(\ell)}$ exhibits a structure 'modulated' by the entries of $\mathbf{A}^{(\ell)}$. For instance, if $a_{ij}^{(\ell)} = 0$,

then $\boldsymbol{\alpha}_{ij}^{(\ell)} := a_{ij}^{(\ell)}\boldsymbol{\beta}_{ij}^{(\ell)}$ is an all-zero block, irrespective of $\boldsymbol{\beta}_{ij}^{(\ell)}$. Therefore, topology identification amounts to finding the nonzero blocks in $\mathbf{W}_{\alpha}^{(\ell)} \in \mathbb{R}^{NT\times N}$, for which our batch algorithm [17] applies readily.

4. ONLINE TOPOLOGY TRACKING ALGORITHM

So far, $\{\mathbf{y}_t\}_{t=1}^T$ were processed in a batch format. However, for settings these samples become available sequentially, the present section develops a novel online algorithm capable of tracking the network topologies in real time. To this end, consider the cost at node j per slot t, namely

$$c_{jt}(\{\bar{f}_{ij}^{(\ell)}\}) := \frac{1}{2}[y_{jt} - \sum_{i\neq j}\bar{f}_{ij}^{(0)}(y_{it}) - \sum_{i=1}^{N}\sum_{\ell=1}^{L}\bar{f}_{ij}^{(\ell)}(y_{i(t-\ell)})]^2$$

where $\bar{f}_{ij}^{(\ell)} := a_{ij}^{(\ell)}f_{ij}^{(\ell)} \in \mathcal{H}_i^{(\ell)}$, since the RKHS is closed with respect to scaling. Therefore, the optimization problem at time slot t can now be written as [cf. (5)]

$$\{\hat{f}_{ij}^{(\ell)}[t]\}_{i=1}^N = \arg\min_{\{\bar{f}_{ij}^{(\ell)}\in\mathcal{H}_i^{(\ell)}\}} \sum_{\tau=1}^{t} c_{j\tau}(\{\bar{f}_{ij}^{(\ell)}\}) + \lambda\sum_{i=1}^{N}\sum_{\ell=0}^{L}\Omega(\|\bar{f}_{ij}^{(\ell)}\|_{\mathcal{H}^{(\ell)}}) \quad (8)$$

which can be solved in a batch form per slot t.

The batch solver however, faces two major limitations: a) It needs to solve (7) per slot t for all nodes, which incurs complexity $\mathcal{O}(NL^3t^3)$ per node that becomes prohibitive with time [17]; and b) the kernel-based estimator combines information from all past samples, which entails prohibitive memory requirements. To alleviate these limitations, the present section develops an online nonlinear SVARM-based algorithm with scalable updates and finite memory complexity to track the dynamic graph topology 'on the fly.'

Problem statement. Given $\{\mathbf{y}_\tau \in \mathbb{R}^N\}_{\tau=1}^t$, the goal is to estimate and track the nonlinear functions $\{\bar{f}_{ij}^{(\ell)}\}$, as well as the corresponding adjacency matrices $\{\mathbf{A}^{(\ell)}\}_{\ell=0}^L$.

Upon obtaining a new data sample, the nonlinear functions will be updated by functional gradient descent. Given $\{y_{it}\}_{i=1}^N$, the chain rule yields the instantaneous functional gradient of $c_{jt}(\{\bar{f}_{ij}^{(\ell)}\})$ per i, j, ℓ, as

$$\nabla_{\bar{f}_{ij}^{(\ell)}} c_{tj}(\{\bar{f}_{ij}^{(\ell)}\}) = \frac{\partial c_{tj}(\{\bar{f}_{ij}^{(\ell)}\})}{\partial \bar{f}_{ij}^{(\ell)}(y_{i(t-\ell)})}\frac{\partial \bar{f}_{ij}^{(\ell)}(y_{i(t-\ell)})}{\partial \bar{f}}(.). \quad (9)$$

The first fraction in (9) is the gradient of $c_{jt}(\{f_{ij}^{(\ell)}\})$ with respect to its scalar argument $\bar{f}_{ij}^{(\ell)}(y_{j(t-\ell)})$, given by

$$c'_{jt}(\bar{f}_{ij}^{(\ell)}) := \partial c_{tj}(\{\bar{f}_{ij}^{(\ell)}\})/\partial \bar{f}_{ij}^{(\ell)}(y_{i(t-\ell)})$$

$$= -\left[y_{jt} - \sum_{i\neq j}\bar{f}_{ij}^{(0)}(y_{it}) - \sum_{i=1}^{N}\sum_{\ell=1}^{L}\bar{f}_{ij}^{(\ell)}(y_{i(t-\ell)})\right] \quad (10)$$

while the second fraction in (9) can be written as

$$\frac{\partial f_{ij}^{(\ell)}(y_{i(t-\ell)})}{\partial f} = \frac{\partial\langle f, \kappa^{(\ell)}(y_{i(t-\ell)},.)\rangle}{\partial f} = \kappa^{(\ell)}(y_{i(t-\ell)},.). \quad (11)$$

Using (9), our stochastic function gradient descent iteration is

$$\bar{f}_{ij}^{(\ell)}[t+1] = \bar{f}_{ij}^{(\ell)}[t] - \eta_t\nabla_{\bar{f}_{ij}^{(\ell)}} c_{tj}(\{\bar{f}_{ij}^{(\ell)}\}). \quad (12)$$

Thanks to the representer theorem, our sought function can be expressed as

$$\hat{\bar{f}}_{ij}^{(\ell)}[t](y) = \sum_{\tau=\ell+1}^{t}\alpha_{ij}^{(\ell)}[t]\kappa_i^{(\ell)}(y, y_{i(\tau-\ell)}) := \boldsymbol{\kappa}_{\bar{\mathbf{y}}_{i,\tau-\ell}}^{\top}(y_t)\boldsymbol{\alpha}_{ij}^{(\ell)}[t] \quad (13)$$

where $\bar{\mathbf{y}}_{i,t-\ell} := [y_{i1}, \ldots, y_{i,t-\ell}]^\top \in \mathbb{R}^{t-\ell}$, and $\boldsymbol{\kappa}_{\bar{\mathbf{y}}_{i,t-\ell}}(y_t) := [\kappa^{(\ell)}(y, y_{i1})\ldots\kappa^{(\ell)}(y_t, y_{i,t-\ell})]^\top$. Combining (12) with (13), we deduce that the stochastic gradient descent algorithm in the functional space can be realized via a parameter update in the sample space as follows

$$\bar{\mathbf{y}}_{i,t-\ell+1} = [\bar{\mathbf{y}}_{i,t-\ell}; y_{i,t-\ell+1}] \quad (14a)$$

$$\boldsymbol{\alpha}_{ij}^{(\ell)}[t+1] = [\boldsymbol{\alpha}_{ij}^{(\ell)}[t]; \quad -\eta_t c'_{jt}(\bar{f}_{ij}^{(\ell)})] \quad (14b)$$

where $\eta_t > 0$ is the step size at time slot t. Note that (14) only considers the loss function without the regularizer. In case when regularizer is incorporated, and $\Omega(z) = z^2$ in (8), the update in functional space can be written as

$$\bar{f}_{ij}^{(\ell)}[t+1] = (1-\eta_t\lambda)\bar{f}_{ij}^{(\ell)}[t] - \eta_t\nabla_{\bar{f}_{ij}^{(\ell)}} c_{tj}(\{\bar{f}_{ij}^{(\ell)}\}) \quad (15)$$

which results in corresponding updates in the feature space as

$$\bar{\mathbf{y}}_{i,t-\ell+1} = [\bar{\mathbf{y}}_{i,t-\ell}; y_{i,t-\ell+1}] \quad (16a)$$

$$\boldsymbol{\alpha}_{ij}^{(\ell)}[t+1] = [(1-\eta_t\lambda)\boldsymbol{\alpha}_{ij}^{(\ell)}[t]; \quad -\eta_t c'_{jt}(\bar{f}_{ij}^{(\ell)})]. \quad (16b)$$

With the sparsity-promoting $\Omega(z) = |z|_1$, the iterative soft thresholding algorithm (ISTA) can be employed, to yield

$$\bar{\mathbf{y}}_{i,t-\ell+1} = [\bar{\mathbf{y}}_{i,t-\ell}; y_{i,t-\ell+1}] \quad (17a)$$

$$\boldsymbol{\zeta}_{ij}^{(\ell)}[t+1] = [(1-\eta_t\lambda)\boldsymbol{\beta}_{ij}^{(\ell)}[t]; \quad -\eta_t c'_{jt}(\bar{f}_{ij}^{(\ell)})] \quad (17b)$$

$$\boldsymbol{\alpha}_{ij}^{(\ell)}[t+1] = \mathcal{S}_{\eta_t\lambda}(\boldsymbol{\zeta}_{ij}^{(\ell)}[t+1]) \quad (17c)$$

where $\mathcal{S}_\lambda(\mathbf{z}) := \frac{\mathbf{z}}{\|\mathbf{z}\|_2}\max(\|\mathbf{z}\|_2 - \lambda, 0)$ is the soft thresholding operator. The overall real-time scheme is summarized as in Algorithm 1. Several remarks are now in order.

Remark 1. (Memory complexity) Notice that (16) and (17) require storing all $\{\mathbf{y}_\tau\}_{\tau=1}^t$, which results in prohibitive

Algorithm 1 Online nonlinear SVARMs

Input: $\mathbf{Y}$, κ, η, λ, L.
Initialize: $\{\bar{\mathbf{y}}_{i,0} = [y_{i1}, \ldots, y_{iL}]\}_{i=1}^{N}$, $\{\boldsymbol{\alpha}_{ij}^{(\ell)}[t]\}_{\ell=1}^{L} = \mathbf{0}$.
for $t = \ell, \ldots$ **do**
  for $j = 1, \ldots, N$ (in parallel) **do**
    for $i = 1, \ldots, N$ **do**
      $\bar{\mathbf{y}}_{i,t-\ell+1} = [\bar{\mathbf{y}}_{i,t-\ell}; y_{i,t-\ell+1}]$
      Update $\{\boldsymbol{\alpha}_{ij}^{(\ell)}[t]\}_{\ell=1}^{L}$ via (16b), or (17c).
    end for
  end for
end for

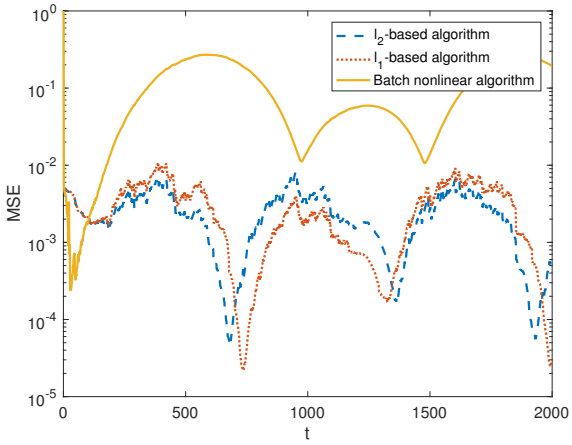

Fig. 1: Tracking MSE over time for slow-varying networks.

memory requirements as t grows. Similar to e.g., [15], this can be resolved by setting a budget B on the number of samples stored. One could possibly store the B most recent samples, along with their corresponding coefficients, while discarding all previous ones. Random feature-based alternatives can also be employed to approximate the kernel functions by inner products of a fixed number of random vectors drawn from a kernel-induced distribution [13, 16].

Remark 2. (Computational complexity) The updates in (16) and (17) incur complexity $\mathcal{O}(t)$ per (i, j, ℓ) triplet, which is markedly lower than $\mathcal{O}(t^3)$ of the batch algorithm. This complexity can be further reduced by setting a budget, or, by applying the aforementioned random-feature based approach.

Remark 3. (Choice of the kernel function) So far, the kernel κ is assumed prescribed. Instead, κ can be learned online from a preselected kernel dictionary, see e.g., [16, 17].

5. NUMERICAL TESTS

Data generation An initial 16-node graph was generated with adjacency matrices $\{\mathbf{A}^{(\ell)}[0]\}_{\ell=1,2}$ drawn from the Erdos-Renyi simulator with probability 0.3. Edge weights

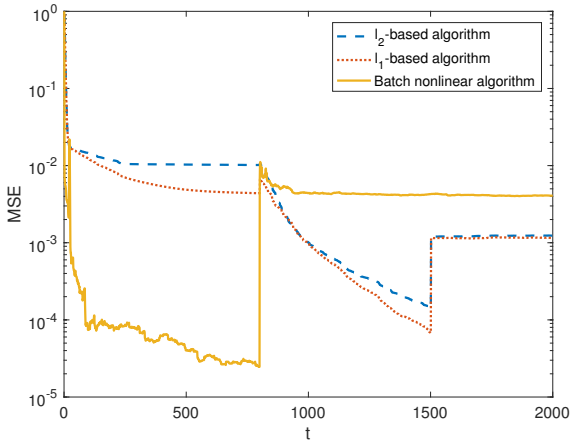

Fig. 2: Tracking MSE over time for switching networks.

in the initial non-zero support of $\mathbf{W}_{\alpha}^{(\ell)}[0]$ varied over time windows following two patterns: p1) $\boldsymbol{\alpha}_{ij}^{(\ell)}[0] = \boldsymbol{\alpha}_{ij}^{(\ell)}[t] + 0.01\sin(t/200)$, for $t = 1, \ldots$; and p2) new edges appear with probability 0.1 at the 800th slot, and edges disappear with probability 0.1 at 1500th slot. Vectors $\{\mathbf{y}_t\}$ were generated according to a nonlinear model using Gaussian kernels with $\sigma = 1$, and $\{e_{it}\}$ drawn independently from $\mathcal{N}(0, 0.01)$.

Results. Edge weights were estimated using Algorithm 1, with updates (16) (named "l_2-based approach"), or (17) ("l_1-based approach"); and also using the nonlinear batch algorithm [17]. Parameters were set at $\lambda = 0.01$, and $\eta_t = 0.01$. The performance was evaluated using the mean-square error, MSE $:= \sum_{ij\ell} \|\boldsymbol{\alpha}_{ij}^{(\ell)}[t] - \hat{\boldsymbol{\alpha}}_{ij}^{(\ell)}[t]\|_2^2/(LN(N-1))$.

Figures 1 and 2 demonstrate that the novel algorithm is capable of tracking the evolving network, while the batch algorithm fails to react fast enough, due to the fact that samples from possibly different models are utilized to estimate the topology at per time slot. At the time slot that the edge support changes, the MSE of the proposed methods increases, but gracefully returns to lower values.

6. CONCLUSIONS

This paper put forth a novel online nonlinear SVARM approach to identifying directed time-varying network topologies. Adopting a regularized LS estimator, and leveraging kernels, stochastic gradient descent iterations were developed to track the sought network topology. Tests on synthetic datasets demonstrated that the novel approach is capable of tracking also dynamic networks. Future directions include a multi-kernel learning generalization, distributed implementations suitable for large-scale networks, as well as identifiability analysis, and testing on real datasets.

7. REFERENCES

[1] D. Angelosante and G. B. Giannakis, "Sparse graphical modeling of piecewise-stationary time series," in *Proc. Intl. Conf. Acoust. Speech Signal Process.*, Prague, Czech Republic, May 2011, pp. 1960–1963.

[2] B. Baingana, G. Mateos, and G. B. Giannakis, "Proximal-gradient algorithms for tracking cascades over social networks," *IEEE J. Sel. Topics Sig. Proc.*, vol. 8, no. 4, pp. 563–575, Aug. 2014.

[3] G. Chen, D. R. Glen, Z. S. Saad, J. P. Hamilton, M. E. Thomason, I. H. Gotlib, and R. W. Cox, "Vector autoregression, structural equation modeling, and their synthesis in neuroimaging data analysis," *Computers in Biology and Medicine*, vol. 41, no. 12, pp. 1142–1155, Dec. 2011.

[4] S. Chen and P. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the Bayesian information criterion," in *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, vol. 8, Virginia, USA, 1998, pp. 127–132.

[5] G. B. Giannakis, Y. Shen, and G. V. Karanikolas, "Topology identification and learning over graphs: Accounting for nonlinearities and dynamics," *Proceedings of the IEEE*, 2018.

[6] M. Gomez Rodriguez, J. Leskovec, and B. Schölkopf, "Structure and dynamics of information pathways in online media," in *Proc. ACM Int. Conf. Web Search and Data Mining*, Rome, Italy, Dec. 2013, pp. 23–32.

[7] C. W. Granger, "Some recent development in a concept of causality," *Journal of Econometrics*, vol. 39, no. 1, pp. 199–211, Sep. 1988.

[8] J. R. Harring, B. A. Weiss, and J.-C. Hsu, "A comparison of methods for estimating quadratic effects in nonlinear structural equation models." *Psychological Methods*, vol. 17, no. 2, pp. 193–214, Jun. 2012.

[9] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer, 2001, vol. 1.

[10] K. G. Jöreskog, F. Yang, G. Marcoulides, and R. Schumacker, "Nonlinear structural equation models: The Kenny-Judd model with interaction effects," *Advanced Structural Equation Modeling: Issues and Techniques*, pp. 57–88, Jan. 1996.

[11] D. Kaplan, *Structural Equation Modeling: Foundations and Extensions*. Sage, 2009.

[12] N. Lim, F. d'Alché Buc, C. Auliac, and G. Michailidis, "Operator-valued kernel-based vector autoregressive models for network inference," *Machine Learning*, vol. 99, no. 3, pp. 489–513, Jun. 2015.

[13] A. Rahimi and B. Recht, "Random features for large-scale kernel machines," in *Proc. Advances in Neural Info. Process. Syst.*, Vancouver, Canada, Dec. 2007, pp. 1177–1184.

[14] M. Rubinov and O. Sporns, "Complex network measures of brain connectivity: Uses and interpretations," *Neuroimage*, vol. 52, no. 3, pp. 1059–1069, Sep. 2010.

[15] F. Sheikholeslami, D. Berberidis, and G. B. Giannakis, "Large-scale kernel-based feature extraction via low-rank subspace tracking on a budget," *IEEE Transactions on Signal Processing*, 2018.

[16] Y. Shen, T. Chen, and G. B. Giannakis, "Online ensemble multi-kernel learning adaptive to non-stationary and adversarial environments," in *Proc. of Intl. Conf. on Artificial Intelligence and Statistics*, Lanzarote, Canary Islands, Apr. 2018.

[17] Y. Shen, B. Baingana, and G. B. Giannakis, "Nonlinear structural vector autoregressive models for inferring effective brain network connectivity," 2016. [Online]. Available: https://arxiv.org/abs/1610.06551

[18] ——, "Kernel-based structural equation models for topology identification of directed networks," *IEEE Trans. Sig. Proc.*, vol. 65, no. 10, pp. 2503–2516, May 2017.

[19] ——, "Tensor decompositions for identifying directed graph topologies and tracking dynamic networks," *IEEE Trans. Signal Processing*, vol. 65, no. 14, pp. 3675–3687, July 2017.

[20] X. Sun, "Assessing nonlinear Granger causality from multivariate time series," in *Proc. Eur. Conf. Mach. Learn. Knowl. Disc. Databases*, Antwerp, Belgium, Sep. 2008, pp. 440–455.

[21] G. Wahba, *Spline Models for Observational Data (CBMS-NSF Regional Conference Series in Applied Mathematics)*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 1990.