INVITED PAPER

# Topology Identification and Learning Over Graphs: Accounting for Nonlinearities and Dynamics

*This article focuses on the problem of learning graphs from data, in particular, to capture the nonlinear and dynamic dependencies.*

By Georgios B. Giannakis, *Fellow IEEE*, Yanning Shen, *Student Member IEEE*, and Georgios Vasileios Karanikolas, *Student Member IEEE*

**ABSTRACT** | Identifying graph topologies as well as processes evolving over graphs emerge in various applications involving gene-regulatory, brain, power, and social networks, to name a few. Key graph-aware learning tasks include regression, classification, subspace clustering, anomaly identification, interpolation, extrapolation, and dimensionality reduction. Scalable approaches to deal with such high-dimensional tasks experience a paradigm shift to address the unique modeling and computational challenges associated with data-driven sciences. Albeit simple and tractable, linear time-invariant models are limited since they are incapable of handling generally evolving topologies, as well as nonlinear and dynamic dependencies between nodal processes. To this end, the main goal of this paper is to outline overarching advances, and develop a principled framework to capture nonlinearities through kernels, which are judiciously chosen from a preselected dictionary to optimally fit the data. The framework encompasses and leverages (non) linear counterparts of partial correlation and partial Granger causality, as well as (non)linear structural equations and vector autoregressions, along with attributes such as low rank, sparsity, and smoothness to capture even directional dependencies with abrupt change points, as well as time-evolving processes over possibly time-evolving topologies. The overarching approach inherits the versatility and generality of kernel-based methods, and lends itself to batch and computationally affordable online learning algorithms, which include novel Kalman filters over graphs. Real data experiments highlight the impact of the nonlinear and dynamic models on consumer and financial networks, as well as gene-regulatory and functional connectivity brain networks, where connectivity patterns revealed exhibit discernible differences relative to existing approaches.

**KEYWORDS** | Kernel-based models; network topology inference; nonlinear modeling; time-varying networks

## I. INTRODUCTION

The science of networks and networked interactions has recently emerged as a major catalyst for understanding the behavior of complex systems [28], [67], [90], [109]. Such systems are typically described by graphs, and can be man-made or natural. For example, human interaction over the web commonly occurs over social networks such as Facebook and Twitter, while sophisticated brain functions are the result of complex physical interactions among neurons; see, e.g., [95] and references therein. Other complex networks show up in diverse fields including financial markets, genomics, proteomics, power grids, and transportation systems, to name a few.

Despite their popularity, single-layer networks may fall short in describing complex systems. For instance, modeling interactions between two individuals using a single edge weight can be an oversimplification of reality. Generalizing their single-layer counterparts, multilayer networks allow nodes to belong to different groups, termed layers [10], [66].

Manuscript received September 25, 2017; revised January 5, 2018; accepted February 2, 2018. Date of current version April 24, 2018. This work was supported by the National Science Foundation (NSF) under Grants 1514056, 1500713, 1711471, and NIH 1R01GM104975-01. *(Corresponding author: Georgios B. Giannakis.)*

The authors are with the Department of Electrical and Computer Engineering and the Digital Technology Center (DTC), University of Minnesota, Minneapolis, MN 55455 USA (e-mail: georgios@umn.edu; shenx513@umn.edu; karan029@umn.edu).

Digital Object Identifier: 10.1109/JPROC.2018.2804318

These layers could represent different views, such as temporal snapshots of the same network, distinct subnetworks (e.g., family, soccer club, or work-related subnetworks), or different units (e.g., infantry, vehicles or airborne units in tactical networks) [83]. Multilayer networks can further model systems typically impossible to represent by traditional graphs, such as heterogeneous information networks [114], [126].

When unknown, the first step in understanding network structure is identification of the underlying graph topology—a critical task in diverse setups; see [67, Ch. 7], [29], [84], [99], and references therein. Applications include the discovery of causal links between regions of interest in the brain, as well as identifying regulatory and inhibitory interactions among genes. Terrorists and fugitives can be unveiled by learning hidden links in social interactions, or telephone call graphs; see, e.g., [6, Ch. 1] for the intelligence leading to the capture of Saddam Hussein. Both undirected as well as directed links are of interest to identify. Pertinent tools for directed graph connectivity identification include Granger causality [89], vector autoregressive models (VARMs) [42], structural equation models (SEMs) [62], [76], and dynamic causal models (DCMs) [37]. The directionality of links cannot be revealed using symmetric correlations between nodal random variables; see, e.g., [36]. Such correlation-based approaches are simple and popular as they rely on tractable linear connectivity models. Linear SEMs have been widely adopted in sociometrics [43], psychometrics [79], genetics [12], and dynamically evolving social networks [5], [87], [106]. Despite their simplicity, linear models cannot capture complex nonlinear interactions that are prevalent in real networks. Here, we will outline advances on nonlinear models for graph topology inference that also subsume their linear counterparts.

Having acquired or knowing *a priori* the topology of a graph provides statistical information about relationships among nodes, and can thus be beneficial for inference of processes evolving over networks. Prevalent learning tasks include dimensionality reduction, classification, and clustering [50]. Dimensionality reduction has been extensively studied [9], [60], [93], [98], and principal component analysis (PCA) [60] is the "workhorse" method for obtaining low-dimensional representations preserving most of the variance present in high-dimensional data. Multidimensional scaling (MDS) [68] on the other hand maintains the pairwise distances between data when going from high- to low-dimensional spaces, while local linear embedding (LLE) [93] only preserves linear relationships between neighboring data. Information from nonneighboring data, however, influences the performance of ensuing tasks such as reconstruction, regression, classification, or clustering [49], [116]. It is also worth stressing that PCA, MDS, and LLE account for only linear relationships among nodal data. Generalizing PCA, kernel PCA [59] captures nonlinear relationships, while Laplacian eigenmaps [9] preserve nonlinear similarities between neighboring data. However, all aforementioned learning tools do not account for structural graph-induced information that is potentially available. Such information may be task specific, e.g., provided by some "expert" or be dictated by the physics specifying the underlying graph, or be inferred from alternative views of the data. As shown in [57], [59], [100], and [101] for PCA, graph awareness can be incorporated in the dimensionality reduction process through regularization. We will also overview in this tutorial nonlinear graph-aware dimensionality reduction approaches that build and broaden the scope of graph-regularized PCA in our era of big data analytics.

Although early graph topology identification and learning presumed static topologies, it became evident that in many domains (e.g., consumer recommendations and financial interactions, gene regulation, and brain functional connectivity) accounting for dynamics can offer valuable insights [5], [13], [52], [53], [56], [91]. These dynamics emerge when the underlying graph topologies are varying, but also when the learning tasks over graphs entail nonstationary processes. Such tasks include clustering, link prediction, and reconstruction of dynamic signals on graphs. These themes are motivated by the need of, e.g., tracking communities evolving over social networks, leveraging multiple graph snapshots obtained across different time slots for improving recommendations, as well as achieving higher reconstruction accuracy for (non) stationary signals over static or dynamic graphs. Here, we will overview learning approaches over dynamic graphs along with recent works that account for nonlinear dynamical models.

The rest of the paper is organized as follows. Section II deals with linear topology identification and learning for processes (signals) evolving over graphs. Section III outlines general kernel-based nonlinear topology identification approaches. Section IV considers generalizations of learning tasks such as dimensionality reduction and clustering to nonlinear settings. Section V overviews several methods for topology identification of time-varying graphs, whereas Section VI outlines learning tasks over such graphs. Finally, Section VII uses numerical tests on both real and synthetic data to illustrate several of the approaches considered.

*Notation:* Bold uppercase (lowercase) letters will denote matrices (column vectors), while operators $(\cdot)^\top$, $\lambda_{\max}(\cdot)$, and $\mathrm{diag}(\cdot)$ will stand for matrix transposition, maximum eigenvalue, and diagonal matrix, respectively. The identity matrix will be represented by $\mathbf{I}$, while $\mathbf{0}(\mathbf{1})$ will denote the matrix or vector of all zeros (ones), and their dimensions will be clear in context. Finally, the $\ell_p$ and Frobenius norms will be denoted by $\|\cdot\|_p$, and $\|\cdot\|_F$, respectively.

## II. PRELUDE: LINEAR AND STATIC MODELS

Consider an $N$-node network $\mathcal{G}(\mathcal{V}, \mathcal{E})$, whose topology is captured by a generally unknown graph adjacency matrix $\mathbf{A} \in \mathbb{R}^{N\times N}$, having nonzero $(i,j)$th entry only if a directed edge is present from node $i$ to node $j$; see Fig. 1. Suppose that the network represents an abstraction of a complex system with measurable input sample $\{x_{it}\}$ of node $i$ at time $t$ scaled by $b_{ii}$,

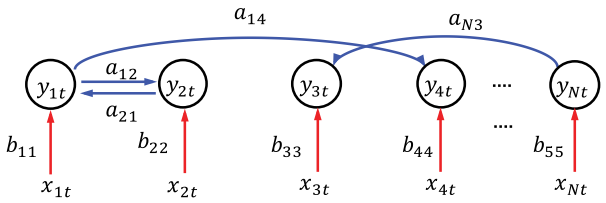


**Fig. 1.** ***Illustration of an N-node network with directed edges (in blue), and (the tth sample of) exogenous measurements per node (red arrows) [105].***

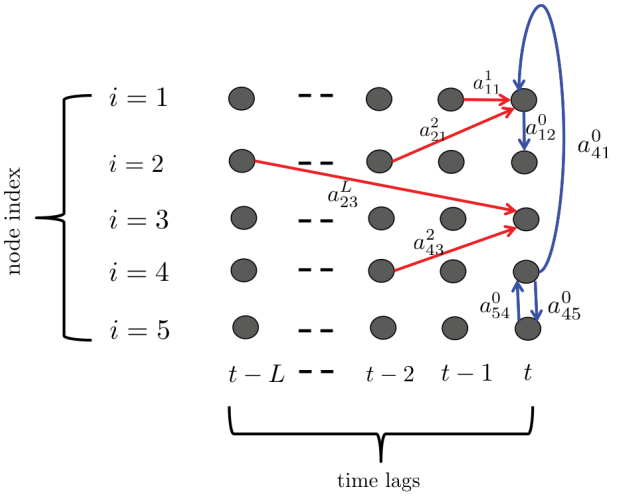


**Fig. 2.** ***SVARMs postulate that dependencies between the nodal time series may be due to instantaneous effects (blue links), and/or time-lagged effects (red links) [104].***

and corresponding output $\{y_{it}\}$ with endogenous (generally directed) links $a_{ij}$; clearly, $a_{ij} = a_{ji}$ for undirected links. In the context of brain networks, $y_{it}$ could represent the $t$-th sample from the $i$th electroencephalogram (EEG) electrode, while $x_{it}$ could be a controlled stimulus that affects a specific region of the brain. In social networks (e.g., Twitter) over which information diffuses, $y_{it}$ could represent the timestamp of user $i$ tweeting about viral story $t$, while $x_{it}$ measures the level of interest (quantified by, e.g., the page rank) of node $i$.

### A. Identifying Graph Topologies

Here we outline methods to identify $\{a_{ij}\}$ from $\{y_{it}\}$ (and $\{x_{it}\}$ if available). A common metric quantifying the adjacency $\{a_{ij}\}$ is the (Pearson) correlation coefficient estimated from $T$ nodal samples collected in vector $\mathbf{y}_i := [y_{i1} \ldots y_{iT}]^\top$, zero-mean compensated by $\overline{\mathbf{y}}_i := T^{-1}\sum_{t=1}^{T} y_{it}\mathbf{1}$, and normalized by the vector norms, to obtain

$$\rho_{ij} := \frac{(\mathbf{y}_i - \overline{\mathbf{y}}_i)^\top (\mathbf{y}_j - \overline{\mathbf{y}}_j)}{\|\mathbf{y}_i - \overline{\mathbf{y}}_i\|_2 \, \|\mathbf{y}_j - \overline{\mathbf{y}}_j\|_2}. \tag{1}$$

Given a probability of false alarms, a threshold $\mathcal{T}_{\text{fa}}$ can be specified to test whether $|\rho_{ij}| > \mathcal{T}_{\text{fa}}$, and thus assert that an edge having strength $a_{ij} = \rho_{ij}$ links nodes $(i,j)$; see, e.g., [67, Ch. 7]. The symmetry of $\rho_{ij}$ implies that it can not reveal directionality of edges. In addition, $\rho_{ij}$ can not discern mediated from unmediated dependencies between pairs of nodal variables. Indeed, consider for instance the three-node toy network $i \to k \to j$, where nodes $i$ and $j$ are mediated through node $k$. This mediation would imply correlation of variables at nodes $i$ and $j$ based on $\rho_{ij}$; thus, correlation-based connectivity can incorrectly declare presence of an $(i,j)$ edge. Fortunately, one can cope with mediation via partial correlations (PCs) that correspond to the correlation coefficients of the residual vectors $\tilde{\mathbf{y}}_i := \mathbf{y}_i - \hat{\mathbf{y}}_{i|\backslash ij}$, where $\hat{\mathbf{y}}_{i|\backslash ij} = f(\{\mathbf{y}_k \,|\, k \in \{\mathcal{V}\backslash ij\}\})$ denotes the predictor of $\mathbf{y}_i$ formed by a function $f$ of observations from all nodes but $i$ and $j$ (this set is henceforth abbreviated as $\backslash ij$). PCs regress $\mathbf{y}_k$ out of $\mathbf{y}_i$ and $\mathbf{y}_j$ to avoid the possibly spurious (due to mediation) edge $(i,j)$. The resultant hypothesis test compares with a prescribed threshold $\mathcal{T}_{\text{fa}}$ the absolute value of [cf., (1)]

$$\tilde{\rho}_{ij} := \frac{(\tilde{\mathbf{y}}_i - \overline{\tilde{\mathbf{y}}}_i)^\top (\tilde{\mathbf{y}}_j - \overline{\tilde{\mathbf{y}}}_j)}{\|\tilde{\mathbf{y}}_i - \overline{\tilde{\mathbf{y}}}_i\|_2 \, \|\tilde{\mathbf{y}}_j - \overline{\tilde{\mathbf{y}}}_j\|_2}. \tag{2}$$

PC-based inference of (un)mediated yet undirected tooplogies requires testing (2) for $\mathcal{O}(N^2)$ pairs of nodes, which can be challenging as $N$ grows. Nonetheless, PCs offer a principled means of detecting edges with constant false-alarm rate.

Interestingly, PCs relying on linear predictor functions $f$ are intimately related with the inverse covariance matrix $\mathbf{\Theta}^{-1} := \text{cov}(\boldsymbol{y})$, where $\boldsymbol{y} := [\mathrm{y}_1 \ldots \mathrm{y}_N]^\top$ collects random variables across nodes. Specifically, if $\hat{\mathrm{y}}_{i|\backslash ij} = \sum_{k \neq i,j} \beta_{kj} \mathrm{y}_k$ is the linear minimum mean-square error (LMMSE) predictor in (2), it holds that (see, e.g., [67, Ch. 7])

$$\tilde{\rho}_{ij} = -[\mathbf{\Theta}]_{ij} / \sqrt{[\mathbf{\Theta}]_{ii}[\mathbf{\Theta}]_{jj}}.$$

If $\mathbf{y}$ is also zero-mean Gaussian distributed, $\boldsymbol{y} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Theta}^{-1})$, the linear predictor is MMSE optimal, and the variables $(y_i, y_j)$ are independent conditioned on all other nodal variables, if and only if $[\mathbf{\Theta}]_{ij} = 0$; that is, e.g., [67, Ch. 7]

$$\text{cov}(\mathrm{y}_i, \mathrm{y}_j \,|\, \boldsymbol{y}_{\backslash ij}) = 0 \Leftrightarrow [\mathbf{\Theta}]_{ij} = 0.$$

This link among linear PCs in (2), conditional uncorrelatedness of nodal variables (or independence in the Gaussian case), and (non)zero entries of $\mathbf{\Theta}^{-1}$ is at the heart of the graphical Lasso approach to topology identification [36]. The latter starts with the regularized log-likelihood of temporally independent Gaussian vectors to form the Lasso criterion for inference of sparse yet undirected graphs as [36]

$$\hat{\mathbf{\Theta}} = \arg\max_{\mathbf{\Theta} \succ 0} \log(\det(\mathbf{\Theta})) - \text{tr}(\hat{\mathbf{\Sigma}}\mathbf{\Theta}) - \lambda \|\mathbf{\Theta}\|_1 \tag{3}$$

where $[\hat{\mathbf{\Sigma}}]_{ij} = T^{-1}\sum_{t=1}^{T} y_{it} y_{jt}$ is the sample covariance estimated using $T$ nodal measurements, and $\|\mathbf{\Theta}\|_1$ the $\ell_1$-norm sparsity regularizer that together with $\lambda$ tune the number of zero entries $[\mathbf{\Theta}]_{ij}$, and thus the adjacency entries $a_{ij}$.

Albeit not able to deal with directionality, the upshot of graphical Lasso and variants (e.g., [29]) is that they reveal edges simultaneously, at complexity $\mathcal{O}(N^3)$ comparable to that required by the PC-based tests.[1] An alternative that also pursues all edges simultaneously and can deal with directionality entails SEMs. Linear SEMs postulate that each $y_{jt}$ depends on two sets of variables: endogenous $\{y_{it}\}_{i\neq j}$ and exogenous $\{x_{jt}\}$, with the unknown structure identified by $\{a_{ij}, b_{jj}\}$ [62]

$$y_{jt} = \sum_{i\neq j} a_{ij} y_{it} + b_{jj} x_{jt} + e_{jt}, \qquad j = 1, \ldots, N \tag{4}$$

where $e_{jt}$ captures unmodeled dynamics. Given samples $\{y_{it}, x_{it}\}$, the topology coefficients $\{a_{ij}\}$ can be obtained using least squares (LS) estimation possibly regularized as in [12] to effect sparsity. Note that the output $y_{jt}$ of node $j$ depends only on its input $x_{jt}$, and its single-hop neighbors. Conditions for identifiability of directional edges $\{a_{ij} \neq a_{ji}\}$ can be found in [7], where the critical role played by the exogenous terms is also highlighted. Such a role in SEMs will be expanded by multilayer SEMs, and (non)linear SVARMs.

A second popular approach to identifying directed topologies relies on Granger causality (GC) [44], whereby a directed edge from node $j$ to $i$ corresponds to a causal dependence of $i$ on $j$. To assess such dependence, linear GC builds on the following two regression hypotheses (see, e.g., [47, Ch. 11]):

$$\mathcal{H}_0 : y_i[t] = \bar{\mathbf{y}}_{\backslash j}^{\top}[t]\boldsymbol{\gamma}_i + \epsilon_{i|\backslash j}[t] \tag{5a}$$

$$\mathcal{H}_1 : y_i[t] = [\bar{\mathbf{y}}_{\backslash j}^{\top}[t], y_j[t-1], \ldots, y_j[t-L]]\boldsymbol{\gamma}_i' + \epsilon_i[t] \tag{5b}$$

where $\bar{\mathbf{y}}_{\backslash j}[t] := [\mathbf{y}_{\backslash ij}^{\top}[t-1]\ldots\mathbf{y}_{\backslash ij}^{\top}[t-L], y_i[t-1]\ldots y_i[t-L]]^{\top}$; subscript $\backslash ij$ denotes all nodal measurements but $i$ and $j$; and $L$ is the model order. After estimating $\boldsymbol{\gamma}_i$ and $\boldsymbol{\gamma}_i'$ using LS, the estimated residuals can be obtained along with their scaled variances as $\hat{\mathrm{s}}_0^2 := \sum_{t=L+1}^{T} \hat{\epsilon}_{i|\backslash j}^2[t]$, and $\hat{\mathrm{s}}_1^2 := \sum_{t=L+1}^{T} \hat{\epsilon}_i^2[t]$. The test statistic $F_{ij} = (\hat{\mathrm{s}}_0^2 - \hat{\mathrm{s}}_1^2)/\hat{\mathrm{s}}_1^2$ is compared to a threshold $\mathcal{T}_{\mathrm{fa}}$ found for a prescribed false-alarm probability. If $F_{ij} > \mathcal{T}_{\mathrm{fa}}$, model $\mathcal{H}_1$ is in effect, and $\{y_j\}$ is said to "Granger cause" $\{y_i\}$. Intuitively, $\{y_j\}$ causes $\{y_i\}$ if including past values of $\{y_j[t']\}_{t'<t}$ in the regressors for predicting $y_i[t]$ lowers the prediction error variance.

Our final class of linear models for topology identification is that of SVARMs, which postulate that each $y_{jt}$ is represented as a linear combination of instantaneous measurements at the remaining nodes $\{y_{it}\}_{i\neq j}$, and their time-lagged counterparts $\{\{y_{i(t-\ell)}\}_{i=1}^{N}\}_{\ell=1}^{L}$ [16]. Specifically, $y_{jt}$ obeys the model

$$y_{jt} = \sum_{i\neq j} a_{ij}^{(0)} y_{it} + \sum_{i=1}^{N}\sum_{\ell=1}^{L} a_{ij}^{(\ell)} y_{i(t-\ell)} + e_{jt} \tag{6}$$

[1]Trading off generality for complexity, Segarra *et al.* [99] postulated smooth polynomial maps of adjacencies to correlations that are linked to diffusions, and a notion of "graph stationarity." Different from (1)–(3), [99] and [29] are not linked to connectivity-related (un)conditional correlations between nodal vectors.

where $a_{ij}^{(\ell)}$ for $\ell \neq 0$ captures the causal influence of node $i$ on node $j$ over a lag of $\ell$ slots, while $a_{ij}^{(0)}$ encodes the corresponding instantaneous relationship between the two. A link is present from node $i$ to node $j$ either when $a_{ij}^{(0)} \neq 0$, or, when there exists some $\ell \in \{1, \ldots, L\}$ for which $a_{ij}^{(\ell)} \neq 0$. Order $L$ can be determined via model selection methods such as the Bayesian information [17], or Akaike's criterion [11].

If $a_{ij}^{(\ell)} = 0 \;\; \forall i, j, \ell \neq 0$, then (6) boils down to (4) with $\mathbf{B} = \mathbf{0}$; hence, SVARMs subsume SEMs without exogenous inputs. In addition, with $a_{ij}^{(0)} = 0 \;\forall i, j$, (6) reduces to the model considered in (5b) [89].

With $\mathbf{y}_t := [y_{1t} \cdots y_{Nt}]^{\top}$, $\mathbf{e}_t := [e_{1t} \ldots e_{Nt}]^{\top}$, and the lagged adjacencies $[\mathbf{A}^{(\ell)}]_{ij} := a_{ij}^{(\ell)}$, the matrix–vector version of (6) is $\mathbf{y}_t = \mathbf{A}^{(0)}\mathbf{y}_t + \sum_{\ell=1}^{L}\mathbf{A}^{(\ell)}\mathbf{y}_{t-\ell} + \mathbf{e}_t$, where $\mathbf{A}^{(0)}$ has $\{a_{ii}^{(0)} = 0\}_{i=1}^{N}$. Broadening the scope of PC, SEM, and Granger models, SVARM unveils the sought topology of (un)directed graphs by estimating via ordinary LS [16] the matrices $\{\mathbf{A}^{(\ell)}\}_{\ell=0}^{L}$ based on the vector time series $\{\mathrm{y}_t\}_{t=1}^{T}$; see also [77] for a recent approach. Alternatively, as with PCs multiple hypotheses can be tested to detect individual links under prescribed false-alarm rates [89]; see also [67, Ch. 7.2] for approaches to predicting missing links.

Even though attractive in its simplicity, the linear time-invariant (static) SVARM falls short in capturing nonlinear dependencies inherent to complex networks. To this end, generalizations of the linear SVARMs to nonlinear kernel-based SVARMs will be considered in Sections III-A and III-B.

### B. Reducing Dimensionality via Graph Regularization

This section deals with a paradigm of learning over static graphs, namely linear dimensionality reduction, when the topology is known and can be employed as prior information. Consider $N$ vectors, each centered by subtracting $N^{-1}\sum_{n=1}^{N}\mathbf{y}_n$, and collected as columns of the $D \times N$ matrix $\mathbf{Y} := [\mathbf{y}_1 \ldots \mathbf{y}_N]$. Dimensionality reduction seeks $d \times 1$ vectors $\{\boldsymbol{\psi}_i\}_{i=1}^{N}$, with $d < D$, that preserve certain properties of the original data $\{\mathbf{y}_i\}$. MDS, for instance, aims at low-dimensional representations $\{\boldsymbol{\psi}_i\}$ that preserve the pairwise distances among $\{\mathbf{y}_i\}$ [68], while LLE maintains local linear relationships within neighborhoods [93]. It is known that all these dimensionality reduction schemes are special cases of kernel-based PCA, which will be presented in Section IV-B [39]; but first, it is instructive to outline PCA and its dual form.

Given $\mathbf{Y}$, PCA obtains the low-dimensional representations $\boldsymbol{\psi}_i = \mathbf{U}_d^{\top}\mathbf{y}_i$, where $\mathbf{U}_d$ has columns the eigenvectors of $\mathbf{Y}\mathbf{Y}^{\top} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^{\top}$ corresponding to its $d$ largest eigenvalues [50]. Matrix $\mathbf{U}$ can equivalently be obtained via the singular value decomposition (SVD) $\mathbf{Y} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\top}$, and the original vectors can be recovered as $\mathbf{y}_i = \mathbf{U}_d\boldsymbol{\psi}_i$. PCA thrives when the

data lie close to a $d$-dimensional hyperplane. Its complexity is that of eigendecomposing $\mathbf{Y}\mathbf{Y}^\top$, namely $\mathcal{O}(ND^2)$, which means that PCA is more affordable when $D \ll N$ [60].

In contrast, for small sets of high-dimensional vectors ($D \gg N$) dual PCA is more attractive. Indeed, the SVD of $\mathbf{Y}$ implies that $\mathbf{U} = \mathbf{Y}\mathbf{V}\mathbf{\Sigma}^{-1}$, which in turn yields the low-dimensional vectors as $\psi_i = \mathbf{U}_d^\top \mathbf{y}_i = \mathbf{\Sigma}_d^{-1}\mathbf{V}_d^\top \mathbf{Y}^\top \mathbf{y}_i$. It follows that $\mathbf{\Psi} = \mathbf{U}_d^\top \mathbf{Y} = \mathbf{\Sigma}_d \mathbf{V}_d^\top$, where $\mathbf{\Sigma}_d \in \mathbb{R}^{d\times d}$ is a diagonal matrix containing the $d$ leading eigenvalues of $\mathbf{Y}^\top\mathbf{Y}$, and $\mathbf{V}_d \in \mathbb{R}^{N\times d}$ is the submatrix of $\mathbf{V}$ collecting the corresponding eigenvectors of $\mathbf{Y}^\top\mathbf{Y}$. The complexity of dual PCA is $\mathcal{O}(DN^2)$; it is thus preferred over PCA when $D \gg N$. It can be readily verified that dual PCA is also the optimal solution to $\min_{\mathbf{\Psi},\mathbf{\Psi}\mathbf{\Psi}^\top=\mathbf{\Lambda}_d} \|\mathbf{Y}^\top\mathbf{Y} - \mathbf{\Psi}^\top\mathbf{\Psi}\|_F^2$, a fact that we will be used in Section IV-B; see, e.g., [108].

In some application scenarios, side information available by the graph structure can be potentially useful for dimensionality reduction. Suppose, for instance, that there is a graph $\mathcal{G}$ over which the data are smooth; that is, vectors $\{\mathbf{y}_i\}$ on connected nodes of $\mathcal{G}$ are also close to each other in Euclidean distance. The Laplacian of $\mathcal{G}$ is $\mathbf{L}_\mathcal{G} := \mathbf{D} - \mathbf{A}$, where $\mathbf{D}$ is a diagonal matrix with entries $[\mathbf{D}]_{ii} = d_{ii} = \sum_j a_{ij}$, and $[\mathbf{A}]_{ij} = a_{ij} \neq 0$ if node $i$ is connected with node $j$. Consider now the term $\mathrm{tr}(\mathbf{\Psi}\mathbf{L}_\mathcal{G}\mathbf{\Psi}^\top) = \sum_{i=1}^N \sum_{j\neq i}^N a_{ij}\|\psi_i - \psi_j\|^2$, which is a sum of the distances of pairs of $\psi_i$'s, weighted by the corresponding edge weight of the pair in $\mathcal{G}$. Invoking this term as regularizer promotes low-dimensional representations corresponding to pairs of nodes connected with large edge weights $a_{ij}$ to stay close to each other. Augmenting the PCA cost function with this regularizer yields the graph-regularized PCA [59]

$$\min_{\mathbf{U}_d,\mathbf{\Psi}} \|\mathbf{Y} - \mathbf{U}_d\mathbf{\Psi}\|_F^2 + \lambda \mathrm{tr}\left(\mathbf{\Psi}\mathbf{L}_\mathcal{G}\mathbf{\Psi}^\top\right) \tag{7}$$

where $\lambda > 0$ controls the strength of regularization. Building upon (7), robust versions of graph-regularized PCA have also been developed in, e.g., [100]and [101].

## III. NONLINEAR MODELS FOR TOPOLOGY IDENTIFICATION

Going beyond linearity, this section generalizes the linear models outlined in Section II-A to capture nonlinear dependencies among nodal variables of a graph.

### A. Undirected Graphs

The linear PC coefficient in (2) is tailored to assessing only linear mediating dependencies. To overcome this limitation, a nonlinear PC metric has been introduced recently [63], using a dictionary of known (so termed kernel) basis functions to replace the linear predictor $\hat{y}_{i|\backslash ij}[t]$ in (2) with a nonlinear function of $\mathbf{y}_{\backslash ij}[t] := \{y_{kt}|k \in \mathcal{V}\backslash ij\}$. To this end, consider the kernel-based regression model $y_i[t] = f_i(\mathbf{y}_{\backslash ij}[t]) + \epsilon_{i|j}[t]$, where $f_i \in \mathcal{H}$ is a function from the reproducing kernel Hilbert space (RKHS) $\mathcal{H} := \left\{f|\, f(\mathbf{y}_{\backslash ij}[t]) = \sum_{t'=1}^\infty \beta_{t'}\, \kappa(\mathbf{y}_{\backslash ij}[t], \mathbf{y}_{\backslash ij}[t'])\right\}$, where the kernel $\kappa$ measures the similarity between $\mathbf{y}_{\backslash ij}[t]$ and $\mathbf{y}_{\backslash ij}[t']$. The functional optimization problem of interest is

$$\hat{f}_i = \operatorname*{argmin}_{f\in\mathcal{H}} \sum_{t'=1}^T (y_i[t'] - f(\mathbf{y}_{\backslash ij}[t']))^2 + \lambda\|f\|_\mathcal{H}^2 \tag{8}$$

where $\lambda$ is a regularization scalar, and $\|\cdot\|_\mathcal{H}$ denotes the norm induced by $\mathcal{H}$. The representer theorem asserts that the solution to (8) is $\hat{f}_i(\mathbf{y}_{\backslash ij}[t]) = \sum_{t'=1}^T \beta_{it'}\, \kappa(\mathbf{y}_{\backslash ij}[t], \mathbf{y}_{\backslash ij}[t'])$ [50, p. 169], which upon substituting into (8) boils down to estimating the $T$ parameters in $\boldsymbol{\beta}_i := [\beta_{i1}, \ldots, \beta_{iT}]^\top$.

Clearly, selecting $\kappa$ specifies $\mathcal{H}$, and hence it affects critically the estimation performance. The nontrivial task of choosing $\kappa$ can be addressed using the data-driven approach known as multi-kernel learning (MKL), where an optimal linear combination of kernels from a preselected dictionary $\{\kappa_p\}_{p=1}^P$ is learned; see, e.g., [19]. That is, $\kappa = \sum_{p=1}^P \theta_p \kappa_p$ with $\theta_p \geq 0\ \forall\ p$. Since for vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ it is possible to include the linear kernel $\kappa_{\mathrm{lin}}(\mathbf{v}_1, \mathbf{v}_2) := \mathbf{v}_1^\top \mathbf{v}_2$, this RKHS-based PC approach subsumes its linear counterpart in (2).

As far as dictionary selection, it depends on the amount of prior information available, and the complexity that can be afforded by the MKL optimization that follows up. For instance, one can adopt a family of smoothness-promoting, linear, Gaussian, heat, or, diffusion kernels (over a grid of their parameters), and many more that can be available as prior information in the application at hand.

Jointly optimizing (8) over $\boldsymbol{\beta}_i$ and over the MKL parameters $\boldsymbol{\theta} := [\theta_1, \ldots, \theta_P]^\top$ turns out to be equivalent to [63]

$$\operatorname*{arg\,min}_{\boldsymbol{\theta}\in\mathcal{C}_q,\, \boldsymbol{\beta}_i\in\mathbb{R}^T} \|(1/\sqrt{\lambda})\mathbf{y}_i - \sqrt{\lambda}\boldsymbol{\beta}_i\|^2 + \sum_{p=1}^P \theta_p \boldsymbol{\beta}_i^\top \mathbf{K}_{p\backslash ij}\boldsymbol{\beta}_i \tag{9}$$

where $[\mathbf{K}_{p\backslash ij}]_{tt'} = \kappa_p(\mathbf{y}_{\backslash ij}[t], \mathbf{y}_{\backslash ij}[t'])$ and $\mathcal{C}_q := \{\boldsymbol{\theta} \geqslant \mathbf{0}, \|\boldsymbol{\theta} - \boldsymbol{\theta}_0\|_q \leq c\}$ with $c$ controlling the regularization. Solving with respect to $\boldsymbol{\beta}_i$ and eliminating it from (9) yields [125]

$$\operatorname*{arg\,min}_{\boldsymbol{\theta}\in\mathcal{C}_q} \mathbf{y}_i^\top \left(\lambda\mathbf{I} + \sum_{p=1}^P \theta_p \mathbf{K}_{p\backslash ij}\right)^{-1} \mathbf{y}_i. \tag{10}$$

This is a convex program that can be solved using the projected gradient descent iteration [20], namely $\boldsymbol{\theta}_{v+1} = \mathcal{P}_{\mathcal{C}_q}(\boldsymbol{\theta}_v - \eta\mathbf{g}(\boldsymbol{\theta}_v))$, where $\mathbf{g}(\boldsymbol{\theta})$ stands for the gradient with respect to $\boldsymbol{\theta}$ of the cost in (10), $\mathcal{P}_{\mathcal{C}_q}(\cdot)$ is a projection operator on the $q$th-norm constraint set $\mathcal{C}_q$, and $\eta$ denotes the step size. The iterative algorithm converges to the global optimum at a rate of $\mathcal{O}(1/v)$ [20], [125]; see also [102] for a recent efficient alternative.

Once the estimates $\hat{y}_{i|\backslash ij}[t] := \hat{f}_i(\mathbf{y}_{\backslash ij}[t])$ for $t = 1, \ldots, T$ are obtained (and likewise for $\hat{y}_{j|\backslash ij}[t]$), the kernel PC of $i$ and $j$ with respect to the rest of the nodes can readily be found by substituting into (2). A hypothesis test is then performed to decide whether an $(i, j)$ edge is present as described in Section II-A. Even though the kernel-based PC captures (un)mediated nonlinear interactions, it does so pairwise; thus, it is more computationally attractive for predicting

only a few edges. In addition, it cannot reveal directionality, which motivates the nonlinear SVARMs considered next.

### B. Directed Graphs

Recognizing the limitations of linear methods for modeling nonlinear dependencies, several nonlinear variants of SEMs have emerged; see, e.g., [48], [58], [61], [65], [69] [104], and [121]. Unfortunately, these works assume that the graph topology is known *a priori*, and the algorithms developed only estimate the unknown edge weights. On the other hand, several variants of nonlinear GC and VARMs have well-documented merits in unveiling links that often remain undiscovered by traditional linear models; see, e.g., [72], [74], [75], and [113]. Linear and nonlinear GC metrics on the other hand entail multiple pairwise tests. These considerations motivate the ensuing approach that jointly identifies edges by leveraging sparse nonlinear SVARMs.

Consider the multivariate nonlinear regression [cf., (6)] $\mathbf{y}_t = \bar{\mathrm{f}}(\mathbf{y}_{\backslash jt}, \{\mathbf{y}_{t-\ell}\}_{\ell=1}^L) + \mathbf{e}_t$, and its entry-wise form, $y_{jt} = \bar{f}_j(\mathbf{y}_{\backslash jt}, \{\mathbf{y}_{t-\ell}\}_{\ell=1}^L) + e_{jt}, \quad j = 1, \ldots, N$.

To circumvent the "curse of dimensionality" in estimating a $[(L+1)N-1]$-variate function, we will confine our multivariate function $\bar{f}_j$ to be separable with respect to each of its $(L+1)N-1$ variables. Such a simplification amounts to adopting a generalized additive model [50, Ch. 9], here of the form $\bar{f}_j(\mathbf{y}_{\backslash jt}, \{\mathbf{y}_{t-\ell}\}_{\ell=1}^L) = \sum_{i\neq j} \bar{f}_{ij}^0(y_{it}) + \sum_{i=1}^N \sum_{\ell=1}^L \bar{f}_{ij}^{(\ell)}(y_{i(t-\ell)})$, where $\{\bar{f}_{ij}^{(\ell)}\}$ will be specified later. With $\bar{f}_{ij}^{(\ell)}(y) := a_{ij}^{(\ell)} f_{ij}^{(\ell)}(y)$, and postulating that the node $j$ measurement at $t$ depends on instantaneous spatial and time lagged effects, one arrives at [cf., (6)]

$$y_{jt} = \sum_{i\neq j} a_{ij}^{(0)} f_{ij}^{(0)}(y_{it}) + \sum_{i=1}^N \sum_{\ell=1}^L a_{ij}^{(\ell)} f_{ij}^{(\ell)}(y_{i(t-\ell)}) + e_{jt} \tag{11}$$

where similar to (6), $\{a_{ij}^{(\ell)}\}$ specify the lag-adjacency matrices $\{\mathbf{A}^{(\ell)}\}_{\ell=0}^L$. Rather than the $[(L+1)N-1]$-variate $\bar{f}_j$, (11) requires estimating $(L+1)N-1$ univariate functions $\{f_{ij}^{(\ell)}\}$.

The linear SVARM in (6) assumes that $\{f_{ij}^{(\ell)}\}$ in (11) are linear, what can be generalized by resorting again to an RKHS model of the nonlinear $\{f_{ij}^{(\ell)}\}$ [105]. Let each univariate $f_{ij}^{(\ell)}(.)$ in (11) belong to the RKHS $\mathcal{H}_i^{(\ell)} := \big\{f_{ij}^{(\ell)} \,|\, f_{ij}^{(\ell)}(y) = \sum_{t=1}^\infty \beta_{ijt}^{(\ell)} \kappa_i^{(\ell)}(y, y_{i(t-\ell)})\big\}$. Considering the measurements at node $j$, and $f_{ij}^{(\ell)} \in \mathcal{H}_i^{(\ell)}$, for $i = 1, \ldots, N$ and $\ell = 0, 1, \ldots, L$, the regularized LS estimates of these functions are

$$\Big\{\hat{f}_{ij}^{(\ell)}\Big\} = \arg \min_{\{f_{ij}^{(\ell)} \in \mathcal{H}_i^{(\ell)}\}} \frac{1}{2} \sum_{t=1}^T \Big[y_{jt} - \sum_{i\neq j} a_{ij}^{(0)} f_{ij}^{(0)}(y_{it}) - \sum_{i=1}^N \sum_{\ell=1}^L a_{ij}^{(\ell)} f_{ij}^{(\ell)}(y_{i(t-\ell)})\Big]^2 + \lambda \sum_{i=1}^N \sum_{\ell=0}^L \Omega\Big(\| a_{ij}^{(\ell)} f_{ij}^{(\ell)} \|_{\mathcal{H}^{(\ell)}}\Big) \tag{12}$$

where the regularizer $\Omega(\mathrm{z})$ can be chosen to effect different attributes, such as sparsity using the $\Omega(\zeta) = \|\zeta\|_1$ surrogate of the $\ell_0$-norm [26]. Invoking again the representer theorem [50, p. 169], the optimal $\hat{f}_{ij}^{(\ell)}(y) = \sum_{t=1}^T \beta_{ijt}^{(\ell)} \kappa_i^{(\ell)}(y, y_{i(t-\ell)})$ can be substituted into (12), and with $\boldsymbol{\beta}_{ij}^{(\ell)} := [\beta_{ij1}^{(\ell)}, \ldots, \beta_{ijT}^{(\ell)}]^\top$, $\boldsymbol{\alpha}_{ij}^{(\ell)} := a_{ij}^{(\ell)} \boldsymbol{\beta}_{ij}^{(\ell)}$, the functional minimization in (12) boils down to optimizing over vectors $\{\alpha_{ij}^{(\ell)}\}$ to find

$$\Big\{\hat{\boldsymbol{\alpha}}_{ij}^{(\ell)}\Big\} = \arg \min_{\{\alpha_{ij}^{(\ell)}\}} \frac{1}{2} \mathrm{z} \Big\| \mathrm{yj} - \sum_{i\neq j} \mathbf{K}_i^{(0)} \alpha_{ij}^{(0)} - \sum_{i=1}^N \sum_{\ell=1}^L \mathbf{K}_i^{(\ell)} \alpha_{ij}^{(\ell)} \Big\|_2^2 + \lambda \sum_{i=1}^N \sum_{\ell=0}^L \Omega\Big(\sqrt{\big(\alpha_{ij}^{(\ell)}\big)^\top \mathrm{K}_i^{(\ell)} \alpha_{ij}^{(\ell)}}\Big) \tag{13}$$

where the $T \times T$ matrices $\{\mathbf{K}_i^{(\ell)}\}$ have entries $[\mathbf{K}_i^{(\ell)}]_{t,t'} = \kappa_i^{(\ell)}(y_{it}, y_{i(t'-\ell)})$. The nonzero $a_{ij}^{(\ell)}$ specifying the topology can be found as the solution of (13) using the alternating direction method of multipliers (ADMM); see, e.g., [40].

As with the kernel-based PC, rather than preselecting $\{\kappa_i^{(\ell)}\}$ a data-driven MKL alternative applies here as well [105]. Consider just for notational simplicity that $\kappa_i^{(\ell)} = \kappa \in \mathcal{K}$, for $\ell = 0, 1, \ldots, L$ and $i = 1, \ldots, N$ in (12); and thus, $\mathcal{H}_i^{(\ell)} = \mathcal{H}^{(\kappa)}$. With $\mathcal{H}_p$ denoting the RKHS induced by $\kappa_p$, the optimal $\{\hat{f}_{ij}^{(\ell)}\}$ is expressible in a separable form as $\hat{f}_{ij}^{(\ell)}(y) := \sum_{p=1}^P f_{ij}^{(\ell,p)}(y)$, where $f_{ij}^{(\ell,p)}$ belongs to RKHS $\mathcal{H}_p$, for $p = 1, \ldots, P$ [8], [78]. Hence, (12) with data-driven kernel selection reduces to

$$\Big\{\hat{f}_{ij}^{(\ell)}\Big\} = \arg \min_{\{f_{ij}^{(\ell,p)} \in \mathcal{H}_p\}} \frac{1}{2} \sum_{t=1}^T \Big[y_{jt} - \sum_{i\neq j} \sum_{p=1}^P a_{ij}^{(0)} f_{ij}^{(0,p)}(y_{it}) - \sum_{i=1}^N \sum_{\ell=1}^L \sum_{p=1}^P a_{ij}^{(\ell)} f_{ij}^{(\ell,p)}(y_{it})\Big]^2 + \lambda \sum_{i=1}^N \sum_{\ell=0}^L \sum_{p=1}^P \Omega\Big(\| a_{ij}^{(\ell)} f_{ij}^{(\ell,p)} \|_{\mathcal{H}_p}\Big). \tag{14}$$

As (14) and (12) are only different in the extra summation over $P$ kernels, (14) can also afford an efficient solver [105].

The kernel-based SVARM outlined here can identify the topology of directed graphs. By simply including linear kernels in the dictionary, it subsumes also linear SVARMs. It can further account for nonlinear interactions, as well as sparsity and low rank of adjacency matrices, while at the same time it scales well with the number of data and the graph size. In a nutshell, the MKL-based RKHS methodology offers a principled overarching approach to topology identification.

### C. Multilayer Graphs

While single-layer graphs are useful for modeling various networks, additional structural information may be revealed if certain networks are modeled via multilayer graphs. Take social networks as an example, where each layer represents a network constructed based on connections on either Facebook, LinkedIn, or Twitter. Nodes in different layers may be related when they correspond to accounts belonging to the same person. This motivates well the focus of this section on modeling and topology identification of multilayer networks.

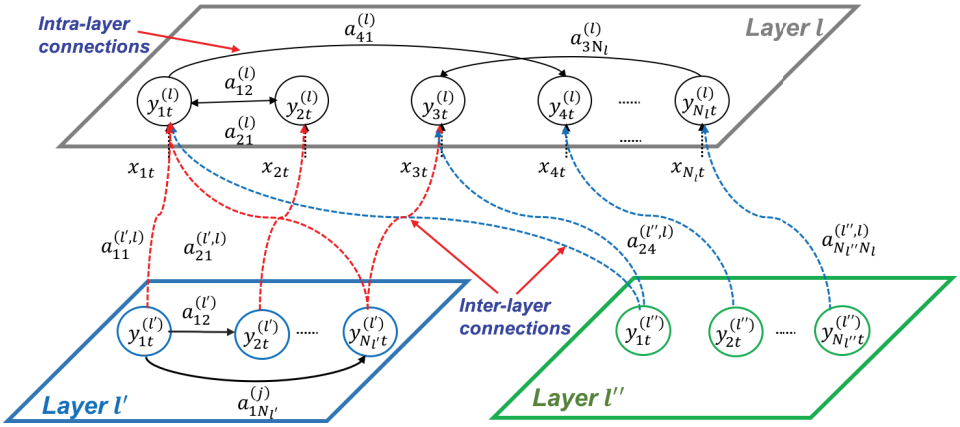


**Fig. 3.** ***Example of a multilayer network.***

Consider an $L$-layer network, consisting of $N=\sum_{\ell=1}^{L}N_\ell$ nodes with $N_\ell$ nodes per layer $\ell$; see also Fig. 3. Each layer $\ell$ can be modeled as a graph $\mathcal{G}^{(\ell)}(\mathcal{V}^{(\ell)},\mathcal{E}^{(\ell)})$, where $\mathcal{V}^{(\ell)}$ is the set of nodes, and $\mathcal{E}^{(\ell)}$ the set of edges. Each $\mathcal{G}^{(\ell)}$ can be described using its adjacency $\mathbf{A}^{(\ell)}$ whose $(i,j)$-th entry $a_{ij}^{(\ell)}\neq 0$ if $(i,j)\in\mathcal{E}^{(\ell)}$; hence, $a_{ij}^{(\ell)}$ is nonzero if there is a directed edge from node $j$ to node $i$ of the $\ell$-th layer. To capture connectivity between nodes belonging to different layers, say $\ell$ and $\ell'$, consider the $N_\ell\times N_{\ell'}$ matrix $\mathbf{A}^{(\ell,\ell')}$, with entries $\alpha_{ij}^{(\ell,\ell')}\neq 0$ if $(i,j)\in\mathcal{E}^{(\ell,\ell')}$, where $\mathcal{E}^{(\ell,\ell')}$ is the set of edges between layers $\ell$ and $\ell'$.

Consider a process observed over the entire network with $y_{it}^{(\ell)}$ denoting the $t$th observation at node $i$ of the $\ell$-th layer. For $L>1$, the so-termed multilayer (mule)-SEM is [cf., (4)]

$$y_{it}^{(\ell)}=\sum_{j\neq i}\alpha_{ij}^{(\ell)}y_{jt}^{(\ell)}+\sum_{\ell'\neq\ell}\sum_{k=1}^{N_{\ell'}}\alpha_{ik}^{(\ell',\ell)}y_{kt}^{(\ell')}+e_{it}^{(\ell)} \tag{15}$$

where the intralayer term $\sum_{j\neq i}\alpha_{ij}^{(\ell)}y_{jt}^{(\ell)}$ captures the influence from same-layer neighboring nodes, while the interlayer term $\sum_{\ell'\neq\ell}\sum_{k=1}^{N_{\ell'}}\alpha_{ik}^{(\ell',\ell)}y_{kt}^{(\ell')}$ models the influence of neighboring nodes from different layers. Nodes are not allowed to connect with themselves per layer. Defining the $T\times 1$ vectors $\mathbf{y}_i^{(\ell)}:=[y_{i1}^{(\ell)},\ldots,y_{iT}^{(\ell)}]^\top$, and the $T\times N_\ell$ matrices $\mathbf{Y}^{(\ell)}:=[\mathbf{y}_1^{(\ell)}\ldots\mathbf{y}_{N_\ell}^{(\ell)}]$ for $\ell=1,\ldots,L$, the matrix mule-SEM is $\mathbf{Y}^{(\ell)}=\mathbf{Y}^{(\ell)}\mathbf{A}^{(\ell)}+\sum_{\ell'\neq\ell}\mathbf{Y}^{(\ell')}\mathbf{A}^{(\ell',\ell)}+\mathbf{E}^{(\ell)}$, $\ell=1,\ldots,L$, where $\mathbf{E}^{(\ell)}$ collects all noise variables for layer $\ell$.

Given $\{\mathbf{Y}^{(\ell)}\}_{\ell=1}^{L}$, topology identification here seeks the unknown $\{\mathbf{A}^{(\ell)}\}_{\ell=1}^{L}$, as well as the interlayer connectivity matrices $\{\mathbf{A}^{(\ell,\ell')}\}_{\ell'=1,\ell'\neq\ell}^{L}$. Since many real-world networks are sparse, $\{\mathbf{A}^{(\ell)}\}_{\ell=1}^{L}$ and $\{\mathbf{A}^{(\ell,\ell')}\}_{\ell'=1,\ell'\neq\ell}^{L}$ are clearly also expected to be sparse. Leveraging this attribute, the topology of multilayer graphs can be estimated via [119]

$$\min_{\substack{A^{(\ell)},\\ \{A^{(\ell,\ell')}\}}}\ \frac{1}{2}\Big\|\mathbf{Y}^{(\ell)}-\mathbf{Y}^{(\ell)}\mathbf{A}^{(\ell)}-\sum_{\ell'\neq\ell}\mathbf{Y}^{(\ell')}\mathbf{A}^{(\ell',\ell)}\Big\|_F^2 +\lambda_1^{(\ell)}\big\|\mathbf{A}^{(\ell)}\big\|_1+\lambda_2^{(\ell)}\sum_{\ell'\neq\ell}\big\|\mathbf{A}^{(\ell',\ell)}\big\|_1\ \text{ s. to } \operatorname{diag}\big(\mathbf{A}^{(\ell)}\big)=\mathbf{0} \tag{16}$$

where $\|\mathbf{Z}\|_1$ denotes the sum of the absolute values of the entries of matrix $\mathbf{Z}$. Problem (16) is convex, and can be solved efficiently in a distributed fashion using ADMM [40], [97].

The nonlinear SVARM approach of the previous section can be readily adapted to multilayer networks by introducing an additional summation over the layers [cf., (11) and (15)].

At this point, it is also worth reflecting on the role of exogenous variables in linear SEMs that are known to aid identifiability of single-layer topologies [7]. This role can be played by mule-SEMs/SVARMs, where multiple layers can represent lagged terms in (non)linear SVARMs or snapshots of dynamic networks across time, as will be seen in Section V.

## IV. NONLINEAR MODELS FOR GRAPH-AWARE LEARNING

With the adjacency matrices at hand, this section studies how various learning tasks can benefit from incorporating dependence information conveyed by graphs. Although graph-aware (semi)supervised classification has been also actively pursued [15], [111], due to space limitations, the ensuing sections will touch on graph-aware nonlinear reconstruction, dimensionality reduction, and clustering approaches.

### A. Nonparametric Regression for Signal Reconstruction

Various applications involve inference of a function defined over a graph naturally, or, as a result of encoding probabilistic dependence among variables viewed as "signals" taking values over the nodes of a graph [32]. Depending on the application, one may have available only limited nodal measurements. In social networks, for instance, individuals may be reluctant to share private information. Such settings could benefit from inference methods that estimate

the nodal features based on samples observed at a subset of the nodes.

A real-valued function (or signal) on a graph is a map $y : \mathcal{V} \to \mathbb{R}$, where $\mathcal{V}$ is the set of vertices. The value $y(v)$ represents a feature of $v \in \mathcal{V}$, e.g., age, political alignment, or annual income of person $v$ in a social network. Suppose that a collection of noisy samples $\{z_m = y(v_m) + e_m\}_{m=1}^M$ is available, where $e_m$ models noise, and $M \leq N$ is the number of measurements. Given $\{z_m\}_{m=1}^M$, and assuming that the graph topology is known, the goal is to estimate $y$, and thus reconstruct the graph signal at unobserved vertices. Letting $\mathbf{z} := [z_1, \ldots, z_M]^\top$, the observation vector obeys $\mathbf{z} = \mathbf{M}\mathbf{y} + \mathbf{e}$, where $\mathbf{y} := [y(v_1), \ldots, y(v_N)]^\top$, $\mathbf{e} := [e_1, \ldots, e_M]^\top$, and $\mathbf{M} \in \{0,1\}^{M \times N}$ is a sampling matrix with binary entries $[\mathbf{M}]_{m,v_m} = 1$ for $m = 1, \ldots, M$, and 0 elsewhere.

Permeating our overarching RKHS approach from topology identification to signal reconstruction, consider $\mathcal{H}_y := \left\{ y |\ y(v) = \sum_{n=1}^N \alpha_n \kappa(v, v_n), \alpha_n \in \mathbb{R} \right\}$ defined over the graph of $N$ nodes. If $y \in \mathcal{H}_y$, it can always be represented as $\mathbf{y} = \mathbf{K}\boldsymbol{\alpha}$, where $[\mathbf{K}]_{ij} := \kappa(v_i, v_j)$, and $\boldsymbol{\alpha} := [\alpha_1, \ldots, \alpha_N]^\top$. Given $\mathbf{z}$, RKHS-based function estimators are found as

$$\hat{\mathbf{y}} = \arg\min_{y \in \mathcal{H}_y} \mathcal{L}(\mathbf{z}, \mathbf{y}) + \lambda \Omega\left( \|\mathbf{y}\|_{\mathcal{H}_y} \right) \tag{17}$$

where $\mathcal{L}$ (e.g., the quadratic loss in LS) measures how the estimated function values at the observed vertices $\{v_m\}_{m=1}^M$, fit the data $\mathbf{z}$; while $\|\mathbf{y}\|_{\mathcal{H}_y}^2 := \boldsymbol{\alpha}^\top \mathbf{K}\boldsymbol{\alpha}$; and, the regularizer $\Omega(.)$ can be chosen to promote desired properties, e.g., smoothness with $\Omega(\zeta) = \zeta^2$. Appealing again to the representer theorem, the solution of (17) is $\hat{y}(v) = \sum_{m=1}^M \check{\alpha}_m \kappa(v, v_m)$, where $\kappa$ is a graph-aware kernel, e.g., representing edge weights. With $\check{\boldsymbol{\alpha}} := [\check{\alpha}_1, \ldots, \check{\alpha}_M]^\top$, and $\alpha := \mathrm{M}^\top \check{\alpha}$, it follows that $\mathrm{y} = \mathrm{K}\alpha = \mathrm{K}\mathrm{M}^\top \check{\alpha}$ [53], [92]. Substituting into (17), and with $\mathcal{L}$ selected as the LS loss, one finds

$$\hat{\check{\alpha}} = \arg\min_{\check{\alpha}} \|\mathbf{z} - \check{\mathbf{K}}\check{\boldsymbol{\alpha}}\|_2^2 + \lambda \Omega\left( \left( \check{\alpha}^\top \check{\mathbf{K}} \check{\alpha} \right)^{1/2} \right) \tag{18}$$

where $\check{\mathbf{K}} := \mathbf{M}\mathbf{K}\mathbf{M}^\top$. With $\hat{\check{\alpha}}$ available, the reconstructed signal is $\hat{\mathbf{y}} = \mathbf{K}\mathbf{M}^\top \hat{\check{\alpha}}$. Generalizing (18), an MKL scheme can be developed by letting $\mathbf{K} = \sum_{p=1}^P \theta_p \mathbf{K}^{(p)}$, where $\{\mathbf{K}^{(p)}\}_{p=1}^P$ is a dictionary of graph kernels. To this end, $\{\theta_q\}_{p=1}^P$ can be incorporated as variables over which to optimize in (18) in order to find the best kernel combination as in Section III-A [92].

### B. Graph-Aware Dimensionality Reduction

To deal with large-scale graphs and high-dimensional data in the learning tasks discussed so far, a task of paramount importance is dimensionality reduction, typically handled by PCA as outlined in Section II-B. While PCA performs well for data close to a hyperplane, this may not hold for many data sets [59]. In such cases, one may resort to kernel (K)PCA, which first "lifts" $\{\mathrm{y}_i\}$ using a nonlinear mapping $\varphi$, onto a higher (possibly infinite) dimensional space. The premise is that with an appropriate $\varphi$ the data will lie on or near a hyperplane in the latter space. KPCA then finds the low-dimensional representations $\{\psi_i\}$, by solving

$$\min_{\boldsymbol{\Psi}: \boldsymbol{\Psi}\boldsymbol{\Psi}^\top = \boldsymbol{\Lambda}_d} \mathrm{tr}\left( \boldsymbol{\Psi} \mathbf{K}_y^{-1} \boldsymbol{\Psi}^\top \right) \tag{19}$$

where $[\mathbf{K}_y]_{i,j} = \kappa(\mathbf{y}_i, \mathbf{y}_j) = \langle \varphi(\mathbf{y}_i), \varphi(\mathbf{y}_j) \rangle$ is the prescribed kernel [46], and $\boldsymbol{\Lambda}_d$ a diagonal matrix containing the $d$ largest eigenvalues of $\mathbf{K}_y$. If a linear kernel is adopted, (19) is equivalent to the dual PCA approach reviewed in Section II-B.

While $\mathbf{K}_y$ in (19) depends only on $\mathbf{Y}$, extra dependencies conveyed by graphs, potentially available, can be accounted for in the dimensionality reduction task. Toward that end, (19) can be regularized by a graph-aware term [cf. (7)]

$$\min_{\boldsymbol{\Psi}: \boldsymbol{\Psi}\boldsymbol{\Psi}^\top = \boldsymbol{\Lambda}_d} \mathrm{tr}\left( \boldsymbol{\Psi} \mathrm{K}_y^{-1} \boldsymbol{\Psi}^\top \right) + \lambda \mathrm{tr}\left( \boldsymbol{\Psi} \mathrm{L}_{\mathcal{G}} \boldsymbol{\Psi}^\top \right) \tag{20}$$

where $\lambda$ is a positive scalar, and $\boldsymbol{\Lambda}_d$ collects the $d$ smallest eigenvalues of $\mathbf{K}_y^{-1} + \lambda \mathbf{L}_{\mathcal{G}} = \overline{\mathbf{V}} \boldsymbol{\Lambda} \overline{\mathbf{V}}^\top$. Combining the Laplacian regularizer with the KPCA cost, (20) is capable of finding $\{\psi_i\}$ that preserve the "lifted" covariance captured by $\mathbf{K}_y$, while at the same time exhibiting smoothness over the graph $\mathcal{G}$. Problem (20) admits the closed-form solution $\boldsymbol{\Psi} = \boldsymbol{\Lambda}_d^{1/2} \overline{\mathbf{V}}_d^\top$, where $\overline{\mathbf{V}}_d$ denotes the submatrix of $\overline{\mathbf{V}}$ formed with columns the eigenvectors corresponding to the eigenvalues in $\boldsymbol{\Lambda}_d$.

When $\kappa$ is not prescribed, once again data-driven MKL approaches can be developed along the lines of Section III-B. In addition, instead of directly using $\mathbf{L}_{\mathcal{G}}$, a family of graph kernels $r(\mathbf{L}_{\mathcal{G}}) := \mathbf{U}_{\mathcal{G}} r(\boldsymbol{\Lambda}) \mathrm{U}_{\mathcal{G}}^\top$ can be employed, where $r(.)$ is a scalar function of the eigenvalues of $\mathbf{L}_{\mathcal{G}}$. By properly selecting $r(.)$, different properties of signals evolving over graphs can be accounted for. As an example, when $r(.)$ sets eigenvalues above a certain threshold to 0, it acts as a sort of "low pass" filter over the graph; see also [53]and [92]. Incorporating $r(.)$ in (20) yields

$$\widehat{\boldsymbol{\Psi}} = \arg\min_{\boldsymbol{\Psi}: \boldsymbol{\Psi}\boldsymbol{\Psi}^\top = \boldsymbol{\Lambda}_d} \mathrm{tr}\left( \boldsymbol{\Psi} r\left( \mathrm{K}_y^{-1} \right) \boldsymbol{\Psi}^\top \right) + \lambda \mathrm{tr}\left( \boldsymbol{\Psi} r(\mathrm{L}_{\mathcal{G}}) \boldsymbol{\Psi}^\top \right). \tag{21}$$

Even though only a single graph regularizer is introduced in (20), this scheme has the flexibility to include multiple graph regularizers based on different graphs [108].

### C. Graph-Aware Subspace Clustering

Using either $\widehat{\boldsymbol{\Psi}}$ or $\mathbf{Y}$, this section will deal with unsupervised learning when data are constrained by a graph model. The focus will be on generalizing subspace clustering, which is known to subsume ordinary clustering (e.g., $K$-means), to account for nonlinear manifolds. In the absence of exogenous inputs ($x_{it} = 0$), (4) bears remarkable resemblance to sparse subspace clustering (SSC) [30], [118], whose goal is to cluster high-dimensional data belonging to a union of low-dimensional subspaces. In particular, given $\{\mathbf{y}_i \in \mathbb{R}^D\}_{i=1}^N$ sampled from the union of $d$-dimensional subspaces embedded in $\mathbb{R}^D$, with $d \ll D$, SSC postulates that $\mathbf{y}_i = \sum_j a_{ij} \mathbf{y}_j + \epsilon_i$, where $a_{ij} \neq 0$ only

if $i$ and $j$ belong to the same subspace, while $\epsilon_i$ captures noise and unmodeled dynamics. SSC seeks a sparsity-promoting LS estimator for $\{a_{ij}\}$ by solving

$$\min_{\{a_{ij},\, j\neq i\}} \left\| \mathrm{y}_i - \sum_j a_{ij}\mathbf{y}_j \right\|_2^2 + \lambda \sum_j |a_{ij}|$$
$$\text{s.t.} \sum_{j=1}^{N} a_{ij} = 1, \quad \forall i = 1, \ldots, N \tag{22}$$

which promotes only a few nonzero coefficients $\{a_{ij}\}$ per $i$. Given $[\mathbf{A}]_{ij} = a_{ij}$, spectral clustering is performed, followed by PCA to identify the constituent linear subspaces [30].

Clearly, estimating SSC weights is reminiscent of identifying $\{a_{ij}\}$ in linear SEMs [cf., (4)]. Viewing SSC as an approximate linear approach to manifold learning (compare also with LLE in, e.g., [93]), the kernel-based SEM advocated in [105] could also be adopted in the first SSC step to estimate $\{a_{ij}\}$, with the goal of exploiting nonlinear relationships between data samples, and thus improving clustering accuracy.

### D. Graph-Aware Recommender Systems

In Section IV-A, a graph with known topology was leveraged to reconstruct missing nodal samples. Here, we will pursue a similar imputation task for recommender systems, where the topology is generally unknown, but can be estimated from the limited available data. To this end, a popular approach known as sparse linear method (SLIM) for top-$N_r$ recommendations starts by representing ratings of each item as a linear combination of ratings of other items with weights $\{a_{ij}\}$ [81]. With vector $\mathrm{r}_i$ collecting ratings of the $i$-th item by all users (those not rating the $i$th item enter 0 ratings), SLIM solves the following problem per item $i$:

$$\min_{\{a_{ii'}\}} \left\| \mathrm{r}_i - \sum_{i'} a_{ii'}\mathbf{r}_{i'} \right\|_2^2 + \lambda \sum_{i'} |a_{ii'}|$$
$$\text{s.t. } a_{ii} = 0, \quad a_{ii'} \geq 0 \ \ \forall i'. \tag{23}$$

Upon obtaining the $\{a_{ii'}\}$, the estimated rating of item $i$ by user $u$ is found as $\hat{r}_{ui} = \sum_{i'} a_{ii'} r_{ui'}$. A top $N_r$ list for user $u$ can then be created by the rank-ordered collection $\{\hat{r}_{ui}\}_{i=1}^{N_r}$. Again, estimating $a_{ii'}$ in (23) is similar to identifying linear SEM coefficients in (4). Thus, a sparse nonlinear method (SNLM) can henceforth be developed along the lines of nonlinear SEMs to improve the accuracy of recommendations.

### E. Joint Inference of Signals and Graphs

So far, the tasks of topology identification and learning signals over graphs were accomplished by solving two separate yet related subproblems. Indeed, they are related because topology identification relied on measurements at all nodes, while signal learning relied on knowing the graph topology.

Here, joint inference of signals and graphs is pursued using limited data $\mathbf{z}_l = \mathbf{M}_l \mathbf{y}_l$, with $\mathbf{M}_l$ denoting the measurement matrix at slot $l$. Given $\mathrm{z}_l$ and $\mathbf{M}_l$, the goal is to find the missing features $\mathbf{y}_l$ and the graph adjacency via [55]

$$\min_{\mathbf{A},\mathbf{Y}|} \sum_{l=1}^{L} \left\| \mathbf{y}_l - \mathrm{A}\mathbf{y}_l \right\|_2^2 + \sum_{l=1}^{L} \left\| \mathbf{z}_l - \mathbf{M}_l \mathbf{y}_l \right\|_2^2 + \lambda_1 \|\mathbf{A}\|_1 + \lambda_2 \|\mathbf{A}\|_F^2 \tag{24}$$

where $\lambda_1$ and $\lambda_2$ are positive constants. Block coordinate descent can be used to solve (24) with guaranteed convergence to a stationary point. Thus, one can jointly estimate the topology and reconstruct unobserved nodal samples, just based on a subset of observations.

The tacit assumption so far is that the graph topology remains invariant over the observation interval. If this is violated, topology identification methods will yield an "average topology," whereas the associated learning tasks will perform suboptimally since they do not fully leverage the information provided by the temporal dimension. This motivates the methods outlined in the ensuing sections that can cope with dynamic topologies, as well as with dynamic nodal processes.

## V. DYNAMIC MODELS FOR TOPOLOGY IDENTIFICATION

In this section, several methods will be outlined for time-varying (TV) graph topology identification, each specified by the model describing the topology per time slot. A TV graph in this context is defined as $\mathcal{G}_t := \{\mathcal{V}, \mathcal{E}_t\}$, with $\mathcal{E}_t$ denoting the set of (possibly directed) edges present at time $t$.

### A. Graphical Lasso-Based Methods

Here, we review how the static graphical Lasso in Section II-A can be adapted to TV topologies [36]. The time-dependent counterpart of the cost in (3) becomes $\mathcal{C}_t(\mathbf{\Theta}_t, \hat{\mathbf{\Sigma}}_t) := \log(\det(\mathbf{\Theta}_t)) - \mathrm{tr}(\hat{\mathbf{\Sigma}}_t \mathbf{\Theta}_t) - \lambda \|\mathbf{\Theta}_t\|_1$, where $\lambda \|\mathbf{\Theta}_t\|_1$ adjusts the sparsity of the sought topology. Matrices $\{\mathbf{\Theta}_t\}$ across slots are estimated as [cf., (3)]

$$\{\hat{\mathbf{\Theta}}_t\} = \arg\max_{\mathbf{\Theta}_t \succ 0} \sum_{t=1}^{T} \mathcal{C}_t(\mathbf{\Theta}_t, \hat{\mathbf{\Sigma}}_t) - \mu \sum_{t=2}^{T} R(\mathbf{\Theta}_t, \mathbf{\Theta}_{t-1}) \tag{25}$$

where $\hat{\mathbf{\Sigma}}_t$ denotes a (possibly weighted) estimate of the covariance matrix; $R(\cdot)$ is an optional term promoting similarity between temporally adjacent topologies; and $\mu, \lambda$ control the respective strength of regularization. Clearly, edge $(i, j)$ is deemed present at slot $t$, if $[\hat{\mathbf{\Theta}}_t]_{ij} \neq 0$.

The relevant dynamic graphical Lasso schemes either assume that $\mathbf{\Theta}_t$ is continuously (albeit slowly) changing, or, it exhibits switching behavior, meaning $\mathbf{\Theta}_1 = \ldots = \mathbf{\Theta}_{\tau_1 - 1} \neq \mathbf{\Theta}_{\tau_1} = \mathbf{\Theta}_{\tau_1 + 1} \ldots \mathbf{\Theta}_{\tau_k - 1} \neq \mathbf{\Theta}_{\tau_k} = \mathbf{\Theta}_{\tau_k + 1} = \ldots = \mathbf{\Theta}_T$, for change points $\tau_1, \ldots, \tau_k$ of the dynamic topology.

The first subclass of methods appeals to smooth topology variations. Among these, e.g., [127], entails $R = 0$, and $\widehat{\Sigma}_t = \sum_\tau \kappa(|t-\tau|)\mathbf{y}_\tau \mathbf{y}_\tau^\top / \sum_\tau \kappa(|t-\tau|)$, where $\kappa(\cdot)$ is a symmetric nonnegative kernel. Alternatively, [41] adopts $R(\boldsymbol{\Theta}_t, \boldsymbol{\Theta}_{t-1}) = \| \boldsymbol{\Theta}_t - \boldsymbol{\Theta}_{t-1} \|_1$ with the aforementioned goal of explicitly promoting smooth evolution of the graph topology; see also [45] for additional choices of $R(\cdot)$, each effecting a topology evolution with different characteristics.

In the second subclass of methods, [3] estimates optimally the change points, as well as the corresponding topologies between pairs of change points, using dynamic programming. Per segment, the approach in [3] adopts $R = 0$, and relies on $\widehat{\Sigma}_t = (\tau_{k+1} - \tau_k)^{-1} \sum_{\tau=\tau_k}^{\tau_{k+1}-1} \mathbf{y}_\tau \mathbf{y}_\tau^\top$ for $t \in [\tau_k, \tau_{k+1} - 1]$.

Graphical Lasso-based approaches can identify the topologies of dynamic, but only undirected graphs. For dynamic directed graphs, one can resort to the methods presented next.

### B. SEM-Based Methods

In order to identify dynamic directional connectivity, approaches here adopt a static linear SEM per slot. The switched dynamic SEM in [4] postulates that the adjacency and input scaling matrices $\{\mathbf{A}, \mathbf{B}\}$ jump among $S$ states $\{\mathbf{A}^s, \mathbf{B}^s\}_{s=1}^S$. Let $\sigma(t)$ denote the state per slot with indicator $\chi_{ts} := \mathbf{1}\{\sigma(t) = s\}$, and suppose that $\check{L}$-variate (instead of univariate) observations $\{y_{it}^{(l)}\}_{l=1}^{\check{L}}$ are available per node $i$ at slot $t$. Given data $\mathbf{Y}_t := [\mathbf{y}_t^{(1)} \ldots \mathbf{y}_t^{(\check{L})}]$, the change points and states are obtained by solving the following problem:

$$\min_{\substack{\{\mathbf{A}^s, \mathbf{B}^s\}_{s=1}^S \\ \{\chi_{ts}\}_{t,s=1}^{T,S}}} \sum_{t=1}^{T} \sum_{s=1}^{S} \chi_{ts} \| \mathbf{Y}_t - \mathbf{A}^s \mathbf{Y}_t - \mathbf{B}^s \mathbf{X} \|_F^2 + \sum_{s=1}^{S} \lambda_s \| \mathbf{A}^s \|_1$$
$$\text{s.t. } \mathbf{A}_{ii}^s = 0, \mathbf{B}_{ij}^s = 0, \quad \forall s, i \neq j, \ \sum_{s=1}^{S} \chi_{ts} = 1 \ \forall t.$$

With $\{\mathbf{Y}_t\}$ acquired sequentially, this NP-hard mixed integer program can be relaxed and solved with a two-step alternating scheme. Using the most recent $\{\widehat{\mathbf{A}}^s, \widehat{\mathbf{B}}^s\}_{s=1}^S$, the state is estimated as $\hat{\sigma}(t) = \arg\min_{s=1,\ldots,S} \| \mathbf{Y}_t - \widehat{\mathbf{A}}_s \mathbf{Y}_t - \widehat{\mathbf{B}}_s \mathbf{X}_t \|_F^2$. Having $\hat{\sigma}(t)$ (and thus $\{\hat{\chi}_{t's}\}_{t',s=1}^{t,S}$) available, solve decoupled problems per $t'$ and $s$ to update $\{\widehat{\mathbf{A}}^s, \widehat{\mathbf{B}}^s\}_{s=1}^S$.

In domains where a slow-varying topology is deemed more plausible than an abruptly switching one, the exponentially weighted LS estimator can be used instead as detailed in [5].

Regarding generalizations, since SEMs are memoryless, one is prompted to pursue dynamic SVARMs and Bayesian models that account for lagged observations.

### C. SVARM-Based Methods

In matrix–vector form, the TV counterpart of the (S)VARM in Section II-A obeys the relationship

$$\mathbf{y}_t = \sum_{l=0}^{L} \mathbf{A}_t^{(l)} \mathbf{y}_{t-l} + \boldsymbol{\epsilon}_t \tag{26}$$

where $[\mathbf{A}_t^{(l)}]_{ij}$ captures the link of $y_{it}$ with $y_{jt-l}$, $L$ is the order, and $\boldsymbol{\epsilon}_t$ accounts for noise and modeling inaccuracies.

The primary differentiation between alternatives comes from the inference process involved. For instance, Fox *et al.* [34] assume that $\mathbf{A}_t^{(l)} \in \{\mathbf{A}^{(ls)}\}_{s=1}^S \ \forall l, t$ with $\mathbf{A}_t^{(l)} = \mathbf{A}^{(ls)}$ if $\sigma(t) = s$. The state is assumed to follow a hierarchical Dirichlet process hidden Markov model (HDP-HMM), while the state sequence is inferred using a Gibbs sampler.

### D. Bayesian-Network-Based Methods

Given the broad scope of dynamic Bayesian networks (DBNs) a multitude of methods are available in this category. The resultant algorithms produce directed graphs with edge directionality assuming a causality interpretation.

For instance, Robinson and Hartemink [88] consider that the transitions between temporally adjacent graphs are restricted to changes from a predefined "move set" that comprises, e.g., the introduction or removal of edges. The Bayesian–Drichlet equivalent metric is taken as the likelihood $p(\mathbf{y}_t | \mathcal{G})$ of the observations $\mathbf{y}_t$ given a particular graph topology $\mathcal{G}$. A Markov chain Monte Carlo (MCMC) sampler is then employed to sample from the posterior of the sequence of graphs and corresponding change points, conditioned on $\mathbf{Y} := [\mathbf{y}_1^\top, \ldots, \mathbf{y}_T^\top]^\top$. HMMs can also be considered as a special case of DBNs. In HMM-based methods, the network structure is assumed to be dictated by a hidden state. The hierarchical Dirichlet process HMM is adopted by [117] to model the distribution of the hidden states. Conditioned on the state, the observations are then assumed to follow a Gaussian Bayesian network model. Inference is performed using an MCMC sampling based algorithm.

### E. Graphical-Regression-Based Approaches

Since the task of inferring the topology of a graph is tantamount to obtaining the neighborhood of each node, a class of methods have emerged that operate on a per-node basis, following this principle.

Logistic regression can be employed to model the (binary) observation(s) at node $i$ and slot $t$ as a function of the observations at the rest of the nodes at $t$ [2]. The logistic regression cost is augmented with an $\ell_1$-norm regularizer and a fusion penalty, to respectively promote sparsity for each $\mathcal{G}_t$, and smooth temporal evolution of the sequence $\{\mathcal{G}_1, \ldots, \mathcal{G}_T\}$.

### F. Tensor-Based Methods

Here the graph is postulated to have a piecewise-constant topology, modeled by a sequence of unknown adjacency matrices $\{\mathbf{A}_m \in \mathbb{R}^{N \times N}, \ t \in [\tau_m, \tau_{m+1} - 1]\}_{m=1}^M$, over $M$ time segments. The $(i,j)$-th entry $[\mathbf{A}_m]_{ij} = a_{ij}^m$ is nonzero only if a directed edge links node $i$ to $j$. The observations obey time-varying SEMs; that is, $y_{jt} = \sum_{i \neq j} a_{ij}^m y_{it} + b_{jj}^m x_{jt} + e_{jt}$ for $t \in [\tau_m, \tau_{m+1} - 1]$ per segment $m = 1, \ldots, M$, with $e_{jt}$

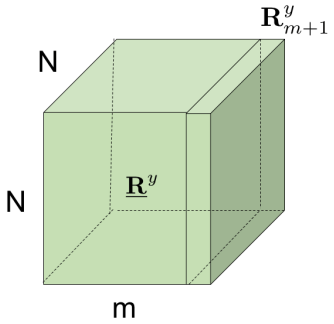


**Fig. 4.** *Tensor $\underline{\mathbf{R}}^{y}$ with frontal slices $\{\mathbf{R}^{y}_{m}\}^{M}_{m=1}$.*

capturing unmodeled dynamics, while coefficients $\{a^{m}_{ij}\}$ and $\{b^{m}_{jj}\}$ are unknown. The noise-free matrix–vector SEM is then $\mathbf{y}_t = \mathbf{A}_m \mathbf{y}_t + \mathbf{B}_m \mathrm{x}_t$, where $[\mathbf{A}_m]_{ij} = a^{m}_{ij}$ and $\mathrm{B}_m := \mathrm{Diag}(b^{m}_{11}, \ldots, b^{m}_{NN})$. Suppose also that the exogenous inputs $\{\mathrm{x}^{(m)}_t\}$ are piecewise stationary over segments $t \in [\tau_m, \tau_{m+1} - 1]$, $m = 1, \ldots, M+1$, each with a fixed correlation matrix $\{\mathbf{R}^{x}_{m} := \mathbb{E}[\mathrm{x}^{(m)}_t (\mathrm{x}^{(m)}_t)^{\top}]\}^{M}_{m=1}$. Under these conditions, an online algorithm can be developed for tracking $\{\mathbf{A}_m, \mathbf{B}_m\}^{M}_{m=1}$ using measured endogenous variables, and the correlation matrices $\{\mathbf{R}^{x}_{m}\}^{M}_{m=1}$ [103], [106].

To this end, let $\boldsymbol{\mathcal{A}}_m := (\mathbf{I} - \mathbf{A}_m)^{-1}\mathbf{B}_m$, and consider a tensor $\underline{\mathbf{R}}^{y}$ with its $m$-th slice $\mathbf{R}^{y}_{m} = \boldsymbol{\mathcal{A}}_m \mathbf{R}^{x}_{m} \boldsymbol{\mathcal{A}}^{\top}_{m}$, $t \in [\tau_m, \tau_{m+1} - 1]$ sequentially appended at $t = \tau_{m+1}$, for $m = 1, \ldots, M$. If $\mathbb{E}\{x_{it} x_{jt}\} = 0, \forall i \neq j$, the $m$th slice can be expressed as a weighted sum of rank-one matrices

$$\mathbf{R}^{y}_{m} = \boldsymbol{\mathcal{A}}_m \mathrm{Diag}(\boldsymbol{\rho}^{x}_{m}) \boldsymbol{\mathcal{A}}^{\top}_{m} \tag{27}$$

where $\boldsymbol{\rho}^{x}_{m} := [\rho^{x}_{m1} \ldots \rho^{x}_{mN}]^{\top}$, with $\rho^{x}_{mi} := \mathbb{E}(x^{2}_{it})$, for $t \in [\tau_m, \tau_{m+1} - 1]$; see also Fig. 4.

Allowing $\underline{\mathbf{R}}^{y}$ to grow sequentially along one mode is well motivated for real-time operation, where data may be acquired in a streaming manner. In this case, unveiling the evolving topology calls for approaches capable of tracking tensor factors $\boldsymbol{\mathcal{A}}_m$. Given the tensor $\underline{\mathbf{R}}^{y}$, and possibly $\mathbf{R}_x$, algorithms for tracking dynamic tensor factors, e.g., PARAFAC via recursive least-squares tracking (PARAFAC-RLST), can be employed; see, e.g., [82], [103], and [106] for details. Once $\widehat{\boldsymbol{\mathcal{A}}}_m$ is obtained, $\mathbf{A}_m$ can be estimated on the fly as $\widehat{\mathbf{A}}_m = \mathbf{I} - \left(\mathrm{Diag}(\widehat{\boldsymbol{\mathcal{A}}}^{-1}_{m})\right)^{-1} \widehat{\boldsymbol{\mathcal{A}}}^{-1}_{m}$ [106].

Tensor-based topology identification along these lines applies to both dynamic and static graphs, so long as (even a subset of) second-order statistics of the exogenous inputs are available, and change across segments; see [106], and [107] where identifiability is studied under low-rank and sparsity constraints on the adjacency matrix. Thus, piecewise input stationary correlations play a role analogous to multiple layers, time-lagged and nonlinear terms in SVARMs, or, the exogenous variables themselves in linear SEMs—what can be critical for identifability when inputs cannot be available (e.g., due to privacy concerns), but their statistics can be measured.

### G. Change Point Detection Methods

Methods in this class typically rely on the likelihood of the observations $p_{\boldsymbol{\Phi}_t}(\mathbf{y}_t)$, $t = 1, \ldots, T$, to detect changes in the topology-specifying parameters $\boldsymbol{\Phi}_t$ per slot $t$. An example of such parameters is the inverse covariance matrix $\boldsymbol{\Theta}_t$ for multivariate Gaussian observations. At their core, these test whether the constancy hypothesis $\boldsymbol{\Phi}_1 = \cdots = \boldsymbol{\Phi}_T$ is broken at certain change points, and estimate their locations.

One specific approach assumes that the graphs at each instance come from the distribution defined by a generalized hierarchical random graph model [85]. A (bootstrapped) hypothesis test for constancy of $\{\boldsymbol{\Phi}_t\}^{T}_{t=1}$ is performed to detect change points. Alternatives model the per-slot topology as a Markov random field [94], or rely on the graphical Lasso [22].

A multi-kernel approach to detecting changes in the generally nonlinear relationships among nodal samples is outlined next [64]. Per node $i$ and for $t \in [\tau_{m-1}, \tau_m - 1]$, suppose samples obey $y_{it} = f^{(m)}_i(\check{\mathbf{y}}_{\backslash it}) + \epsilon^{(m)}_{it}$, where $f^{(m)}_i$ is a nonlinear function, and $\check{\mathbf{y}}_{\backslash it} := [\mathbf{y}^{\top}_{\backslash it}, \mathbf{y}^{\top}_{t-1}, \ldots, \mathbf{y}^{\top}_{t-L}]^{\top}$, with the usual notational convention on the subscripts and memory $L$. Function $f^{(m)}_i$ can be estimated using MKL-based ridge regression [cf. (8)–(10)]. With such estimates available, the residuals $\hat{\epsilon}^{(m)}_{it} := y_{it} - \hat{f}^{(m)}_i(\check{\mathbf{y}}_{\backslash it})$ can be used to infer the presence, and estimate the locations of change points.

Consider first the base case of having at most one change point, and let $\mathbf{f}^{(m)} := [f^{(m)}_1 \ldots f^{(m)}_N]^{\top}$ collect the functions characterizing the segment $m$ across nodes. Here $m \in \{0, 1, 2\}$, with $m = 0$ corresponding to the whole data record $[1, T]$. Deciding whether a change point is present amounts to performing the composite hypothesis test

$$\mathcal{H}_0 \colon \mathbf{f}^{(1)} = \mathbf{f}^{(2)} := \mathbf{f}^{(0)} \qquad \mathcal{H}_1 \colon \mathbf{f}^{(1)} \neq \mathbf{f}^{(2)} \tag{28}$$

where according to $\mathcal{H}_1$ there is a change point $\tau_1$ (the location of which is to be estimated) when the vector functions differ for segments $[1, \tau_1 - 1]$ and $[\tau_1, T]$. According to $\mathcal{H}_0$, no such $\tau_1$ is present. Toward specifying a test statistic for (28), let $G(\epsilon; \mu, \sigma^2)$ denote the probability density function of a Gaussian variable $\epsilon$ with mean $\mu$ and variance $\sigma^2$. The approximate likelihood with $\{\hat{\epsilon}^{(m)}_{it}\}$ Gaussian, under $\mathcal{H}_0$ is $p(\mathbf{Y}; \mathcal{H}_0) \approx \prod^{N}_{i=1} \prod^{T}_{t=1} G(\hat{\epsilon}^{(0)}_{it}; 0, \hat{\sigma}^{2(0)}_i)$; and under $\mathcal{H}_1$, $p(\mathbf{Y}; \tau, \mathcal{H}_1) \approx \prod^{N}_{i=1} \left[\prod^{\tau-1}_{t=\tau} G\left(\hat{\epsilon}^{(1)}_{it}; 0, \hat{\sigma}^{2(1,2)}_i\right) \prod^{T}_{t=\tau} G\left(\hat{\epsilon}^{(2)}_{it}; 0, \hat{\sigma}^{2(1,2)}_i\right)\right]$, with $\hat{\sigma}^{2(0)}_i = T^{-1} \sum^{T}_{t=1} \hat{\epsilon}^{2(0)}_{it}$, and $\hat{\sigma}^{2(1,2)}_i(\tau) = T^{-1}\left(\sum^{\tau-1}_{t=1} \hat{\epsilon}^{2(1)}_{it} + \sum^{T}_{t=\tau} \hat{\epsilon}^{2(2)}_{it}\right)$. The corresponding approximate generalized likelihood ratio test statistic of change point $\hat{\tau}$ is

$$\begin{aligned} \Lambda(\mathbf{Y}; \hat{\tau}) &:= \max_{\tau \in (1,T)} \log p(\mathbf{Y}; \tau, \mathcal{H}_1) / p(\mathbf{Y}; \mathcal{H}_0) \\ &= \max_{\tau \in (1,T)} (T/2) \log \sum^{N}_{i=1} \hat{\sigma}^{2(0)}_i / \hat{\sigma}^{2(1,2)}_i(\tau) \,. \end{aligned} \tag{29}$$

We decide that $\mathcal{H}_1$ is in effect if $\Lambda$ exceeds a certain threshold, which for a given probability of false alarms is

obtained from the distribution of $\Lambda$ under $\mathcal{H}_0$. This distribution is estimated using a model-based bootstrap, as detailed in [64].

The case of an unknown number of change points can be tackled using a variant of the binary segmentation approach of [120] that builds on (28) and (29) [64]. At the beginning of iteration $k$, the interval $[1,T]$ is split into $k$ segments with $\Lambda_1,\ldots,\Lambda_k$ denoting the corresponding test statistics. Let the maximum over these statistics correspond to the segment $n^*:=\arg\max_{n=1,\ldots,k}\ \Lambda_n$. A hypothesis test is then conducted over this segment to assess whether a change point lies therein. If $\mathcal{H}_1$ is accepted, the proposed point is appended to the discovered change points and the process moves on to iteration $k+1$; otherwise it stops, with $k$ segments discovered.

With the change points and corresponding segments available, any of the static topology identification methods of Section II-A or Section III can be applied per segment.

## VI. DYNAMIC MODELS FOR LEARNING OVER GRAPHS

In this section, a sample of learning tasks over dynamic graphs will be reviewed. Some of the methods can afford online implementation allowing nodes to (dis)appear as time progresses, which explains the $t$-dependent notation $\mathcal{V}_t$ for the set of nodes. Most methods further assume that the (generally varying) topology of the graph is either known, or, it has been acquired using the methods outlined in the previous section.

### A. Dynamic Graph-Aware Link Prediction

Temporal link or edge prediction amounts to inferring the (dis)appearance of edges ahead of time by leveraging currently available graph snapshots. Let $\mathscr{A}_t:=\{\mathbf{A}_1,\ldots,\mathbf{A}_t\}$ denote this set of snapshots with $[\mathbf{A}_t]_{ij}\in\{0,1\}\ \forall\ i,j,t$, and $[\mathbf{A}_t]_{ij}=1$ if the edge $i\to j$ is present at time $t$. This is in contrast to the ordinary link prediction setup, where a single snapshot is used [71], thereby ignoring temporal patterns potentially present in the data.

Given $\mathscr{A}_t$, the goal is to predict $\mathbf{A}_{t+\Delta t}$ for $\Delta t\geq 1$. Supposing $\Delta t=1$ for brevity, we will rely on an $N\times N$ matrix $\check{\mathbf{R}}^{t+1}$ comprising "edge scores," based on which link $i\to j$ will be deemed present in $\mathbf{A}_{t+1}$, if $\left[\check{\mathbf{R}}^{t+1}\right]_{ij}$ exceeds a certain threshold. An early work combined per-snapshot spatial predictors with temporal predictors across snapshots [51]. Specifically, Huang and Lin [51] postulate that the time series $\{[\mathbf{A}_1]_{ij},\ldots,[\mathbf{A}_t]_{ij}\}$ per $(i,j)$ obeys an autoregressive integrated moving average (ARIMA) model. Predicting $[\mathbf{A}_{t+1}]_{ij}$ is thus converted to a score $\left[\check{\mathbf{R}}^{t+1}_{\text{ARIMA}}\right]_{ij}$, and the overall score is obtained as $\left[\check{\mathbf{R}}^{t+1}\right]_{ij}:=\left[\check{\mathbf{R}}^{t+1}_{\text{ARIMA}}\right]_{ij}\left[\check{\mathbf{R}}^{t+1}_{\text{static}}\right]_{ij}$, where the second factor is found after applying a static link predictor to $\overline{\mathbf{A}}_t:=\sum_{\tau=1}^{t}\mathbf{A}_\tau$.

Matrix and tensor factorization methods have also been considered for dynamic link prediction [27]. Matrix $\check{\mathbf{R}}^{t+1}$ is found as a low-rank approximation to the weighted average adjacency $\overline{\mathbf{A}}_t^{(w)}:=\sum_{\tau=1}^{t}w^{t-\tau}\mathbf{A}_t$, with $w\in(0,1)$ being the forgetting factor. Alternatively, one can form a three-way tensor with entries $[\mathbf{A}_t]_{ij}$, and invoke its low $K$-component CANDECOMP/PARAFAC approximant through the decomposition $\sum_{k=1}^{K}\eta_k\boldsymbol{\alpha}_k\circ\boldsymbol{\beta}_k\circ\boldsymbol{\gamma}_k$, where $\eta_k>0$; $\boldsymbol{\alpha}_k$, $\boldsymbol{\beta}_k,\boldsymbol{\gamma}_k$ denote the factors; and $\circ$ is the Khatri–Rao product [27]. The score matrix is then $\check{\mathbf{R}}^{t+1}=\sum_{k=1}^{K}\delta_k\eta_k\boldsymbol{\alpha}_k\boldsymbol{\beta}_k^\top$ with $\delta_k=T_0^{-1}\sum_{\tau=t-T_0+1}^{t}\boldsymbol{\gamma}_k(\tau)$, where $T_0$ represents the moving window size over which the entries of the temporal profiles $\boldsymbol{\gamma}_k$ are averaged. In other words, $\left[\check{\mathbf{R}}^{t+1}\right]_{ij}$ is a weighted sum of the relationships between the pair $(i,j)$ across the $K$ components $\left\{\left[\boldsymbol{\alpha}_k\boldsymbol{\beta}_k^\top\right]_{ij}\right\}_{k=1}^{K}$ with the contribution of component $k$ weighted by the recent "average activity" $\delta_k$ thereof.

A provably consistent nonparametric temporal link predictor is developed in [96] by assuming that $[\mathbf{A}_{t+1}]_{ji}|\mathscr{A}_t\sim\text{Bernoulli}(g(\boldsymbol{\varphi}_t(i,j)))$, where $\boldsymbol{\varphi}_t(i,j)$ comprises features specific to the $(i,j)$ pair (e.g., number of common neighbors shared by the pair and the time instance of the last appearance of the edge), as well as to the local neighborhood of node $i$. A kernel-based estimator $\hat{g}$ is developed, using which the score of the edge $i\to j$ is obtained as $[\check{\mathbf{R}}^{t+1}]_{ij}=\hat{g}(\boldsymbol{\varphi}_t(j,i))$ [96]. Recently, a deep learning approach to temporal link prediction has been also developed [70].

### B. Dynamic Graph-Aware Clustering

Paralleling Section IV-C, this section deals with estimation of dynamically evolving clusters that exhibit TV graph-encoded similarities between nodal objects. Many of the methods presented hereafter are geared toward discovering communities, that is clusters of nodes exhibiting dense intra-cluster connectivity—a key task in network science, e.g., [33].

With $\mathbf{y}_{it}$ denoting the observation vector (object) of node $i$ at slot $t$, the corresponding set of objects for the same slot is $\mathcal{Y}_t:=\{\mathbf{y}_{it}\}_{i=1}^{N_t}$ or in matrix form $\mathbf{Y}_t:=[\mathbf{y}_{1t}\ldots\mathbf{y}_{N_t t}]$. The TV (dis)similarity between $\mathbf{y}_{it}$ and $\mathbf{y}_{jt}$ is encoded in the TV edge weights $[\mathbf{A}_t]_{ij}\in\mathbb{R}$. If pairwise dissimilarities are measured by Euclidean distances, as in the $K$-means clustering algorithm, then $[\mathbf{A}_t]_{ij}=\|\mathbf{y}_{it}-\mathbf{y}_{jt}\|_2^2$. Suppose that $K_t=K$ clusters are formed $\forall t$, and let $\boldsymbol{\Pi}_t:=[\pi_{1t}\ldots\pi_{Kt}]$ denote the $N_t\times K$ membership matrix at slot $t$ with binary entries $[\boldsymbol{\Pi}_t]_{ik}=1$, if node $i$ belongs to cluster $k$ at slot $t$, and $[\boldsymbol{\Pi}_t]_{ik'}=0\ \forall\ k'\neq k$. This corresponds to hard clustering, but soft (or probabilistic) clustering [50, Ch. 14] can be also accommodated with $\sum_{k=1}^{K}[\boldsymbol{\Pi}_t]_{ik}=1$ for $i=1,\ldots,N_t$.

A class of dynamic graph-aware clustering approaches relies on two functions of $(\mathbf{A}_t,\boldsymbol{\Pi}_t)$, namely the snapshot quality and the history cost [14]. The snapshot quality $\text{sq}(\boldsymbol{\Pi}_t,\mathbf{A}_t)$ measures how well the objects $\mathcal{Y}_t$ are represented by the clustering $\boldsymbol{\Pi}_t$, with respect to the (dis)similarities given by $\mathbf{A}_t$. In $K$-means, for example, $\text{sq}(\boldsymbol{\Pi}_t,\mathbf{A}_t)=\sum_{k=1}^{K}\sum_{i=1}^{N_t}[\boldsymbol{\Pi}_t]_{ik}\big(1-\|$

$\mathbf{y}_{it} - \left(\pi_{kt}^\top \mathbf{1}\right)^{-1} \mathbf{Y}_t \pi_{kt}\|_2^2\big)$, where nodal vectors are normalized so that $\|\mathbf{y}_{it}\|_2^2 = 1 \ \forall\ i,t$. Here, $\mathrm{sq}(\mathbf{\Pi}_t, \mathbf{A}_t)$ is proportional to the negative of the sum of distances of each $\{\mathbf{y}_{it}\}$ from the cluster centroid it is assigned to. The history cost $\mathrm{hc}(\mathbf{\Pi}_t, \mathbf{\Pi}_{t-1})$ is a measure of the distance between $\mathbf{\Pi}_t$ and $\mathbf{\Pi}_{t-1}$, thereby promoting similarity between temporally adjacent clusterings. For $K$-means, $\mathrm{hc}(\mathbf{\Pi}_t, \mathbf{\Pi}_{t-1}) = \min_{f:\{1\ldots K\}\to\{1\ldots K\}} \sum_{k=1}^K \|\left(\pi_{kt}^\top \mathbf{1}\right)^{-1} \mathbf{Y}_t \pi_{kt} - \left(\pi_{f(k)(t-1)}^\top \mathbf{1}\right)^{-1} \mathbf{Y}_{t-1} \pi_{f(k)(t-1)}\|_2^2$. In words, $\mathrm{hc}(\mathbf{\Pi}_t, \mathbf{\Pi}_{t-1})$ is the sum of the Euclidean distances of each centroid at time $t$ to the corresponding one at $t-1$. Combining the two metrics, the optimal clustering at slot $t$ is

$$\mathbf{\Pi}_t^* = \arg\max_{\mathbf{\Pi}_t} \alpha\ \mathrm{sq}(\mathbf{\Pi}_t, \mathbf{A}_t) - (1-\alpha)\,\mathrm{hc}(\mathbf{\Pi}_t, \mathbf{\Pi}_{t-1}) \quad (30)$$

where $\alpha \in [0,1]$ controls the weight between the two costs. Note that (30) entails a criterion naturally suited for streaming graphs, since $\mathbf{\Pi}_t$ does not depend on $\mathbf{A}_\tau$ with $\tau > t$. This class of methods is typically referred to as "evolutionary clustering."

The approach in (30) detailed for dynamic $K$-means, generalizes to clustering algorithms that correspond to different choices of $\mathrm{sq}(\cdot)$. In particular, nonnegative matrix factorization [73], and several spectral clustering costs have been considered in, e.g., [18]. As far as tuning $\alpha$, it depends on the dynamic scenario at hand, but Folino and Pizzuti [31] have advocated data-driven genetic algorithms to optimize its value.

Instead of adjusting clustering algorithms to graph dynamics, one can rely on generative models to describe the evolution of clusters, and infer the associated model parameters. Examples include Dirichlet process mixture models [115], as well variants of the stochastic blockmodel (SBM) possibly augmented with a state evolution model [38], [122]. In particular, Xu and Hero [122] assume that the SBM parameters are dictated by the system state whose evolution is governed by a stochastic dynamical system. Inference is then performed using a variant of the extended Kalman filter, and the cluster memberships are estimated using label-switching methods. Detectability limits of dynamic SBMs along with belief propagation algorithms that attain these limits have been also reported [38].

One last model postulates that the community structure is piecewise constant, that is $\mathbf{\Pi}_1 = \ldots = \mathbf{\Pi}_{\tau_1 - 1} \neq \mathbf{\Pi}_{\tau_1} = \mathbf{\Pi}_{\tau_1+1} \ldots \mathbf{\Pi}_{\tau_k - 1} \neq \mathbf{\Pi}_{\tau_k} = \mathbf{\Pi}_{\tau_k+1} = \ldots = \mathbf{\Pi}_T$ for some change points $\tau_1, \ldots, \tau_k$. Such an approach based on the minimum description length (MDL) criterion is developed in [112]. The community structure is obtained by minimizing the model complexity with respect to the cluster assignments, as assessed by the MDL criterion. A change point $\tau_l$ is declared if assigning the current graph snapshot $\mathcal{G}_t$ to the current segment $[\tau_{l-1}, t-1]$, does not reduce the overall model complexity.

The parallelism between multilayer and dynamic SEMs mentioned by the end of Section III-C can permeate benefits of multilayer clustering [24] to dynamic clustering [123].

### C. Dynamic Graph-Aware Reconstruction

Reconstructing signals on TV graphs amounts to estimating a function defined over the nodes, based on observations from a subset of nodes collected at possibly different time instances. The associated methods rely on function properties across nodes (e.g., smoothness or band-limitedness), and leverage the TV graph topology to perform the reconstruction task (also known as interpolation or imputation).

Specifically, let $\mathbf{z}_t = \mathbf{M}_t \mathbf{y}_t + \epsilon_t$ denote a noisy subset of nodal measurements at slot $t$ collected at nodes specified by the binary $M_t \times N$ wide ($M_t < N$) matrix $\mathbf{M}_t$. For the dynamics of state $\mathbf{y}_t$ affected by the known graph transition adjacency matrix $\mathbf{A}_{t,t-1}$, consider the superimposed model

$$\mathbf{y}_t = \mathbf{y}_t^{(\nu)} + \mathbf{y}_t^{(\chi)}, \qquad \mathbf{y}_t^{(\chi)} = \mathbf{A}_{t,t-1}\mathbf{y}_{t-1}^{(\chi)} + \boldsymbol{\eta}_t \quad (31)$$

where the space-only component $\mathbf{y}_t^{(\nu)}$ is temporally uncorrelated to account for "fast" dynamics across slots; while the spatio-temporal correlated VARM component $\mathbf{y}_t^{(\chi)}$ captures the "slow" dynamics (also known as trend). The state evolution model in (31) is suitable for capturing variations in, e.g., packet delays, stock prices, and temperature, measured respectively at Internet, financial markets, and sensor networks [54], [87].

Using $\{\mathbf{z}_{t'}\}$, (31), as well as graph kernels $\{\mathbf{K}_{t'}^{(\eta)}\}$ and $\{\mathbf{K}_{t'}^{(\nu)}\}$, a space-time approach to reconstructing $\{\mathbf{y}_{t'}\}$ is

$$\begin{aligned}\arg\min_{\{\mathbf{y}_{t'}^{(\chi)}, \mathbf{y}_{t'}^{(\nu)}\}_{t'=1}^t} & \sum_{t'=1}^t \frac{1}{M_{t'}} \left\|\mathbf{z}_{t'} - \mathbf{M}_{t'}\mathbf{y}_{t'}^{(\chi)} - \mathbf{M}_{t'}\mathbf{y}_{t'}^{(\nu)}\right\|_2^2 \\ & + \mu_1 \sum_{t'=1}^t \left\|\mathbf{y}_{t'}^{(\chi)} - \mathbf{A}_{t',t'-1}\mathbf{y}_{t'-1}^{(\chi)}\right\|_{\mathbf{K}_{t'}^{(\eta)}}^2 + \mu_2 \sum_{t'=1}^t \left\|\mathbf{y}_{t'}^{(\nu)}\right\|_{\mathbf{K}_{t'}^{(\nu)}}^2\end{aligned} \quad (32)$$

where the first sum is an LS measurement error; the second sum is a spatiotemporal graph weighted LS state transition error; and the last sum is a spatial graph weighted (so-termed kriging) regularizer. Although (32) can be solved in batch form, its complexity grows prohibitively with the time horizon. However, it has been shown that the sequence of estimates $\{\hat{\mathbf{y}}_{t'|t'}^{(\chi)}, \hat{\mathbf{y}}_{t'|t'}^{(\nu)}\}_{t'=1}^t$ can be obtained online using what is called in [54] kernel kriged Kalman filtering; see also [91] and [23].

## VII. NUMERICAL TESTS

This section presents numerical tests conducted on both synthetic and real data to demonstrate the effectiveness of some of the approaches considered.

### A. Synthetic Tests for SVARMs

We first test the performance of the approach in Section III-B using synthetic data. Setting $L = 1$ in (11), samples were generated via a random Erdős–Rényi graph having $N = 20$ nodes, with probability of edge presence set to 0.4. Nodal samples were generated using linear and nonlinear models. For several values of $T$, entries of $\mathbf{Y} \in \mathbb{R}^{N\times T}$ were

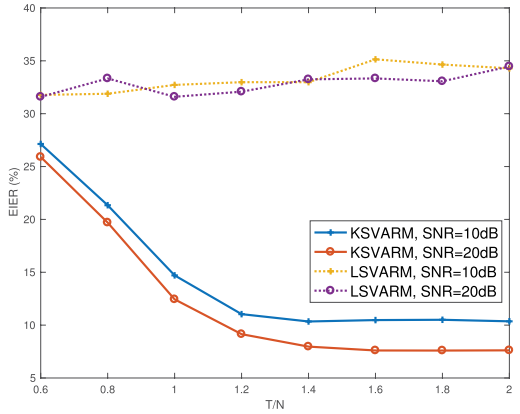


**Fig. 5.** ***EIER versus measurements-to-nodes ratio (T/N) for simulated data generated using polynomial kernel of order P = 2. K-SVARMs consistently outperform linear (L)SVARMs.***

randomly drawn from the standardized normal distribution, that is $y_{it} \sim \mathcal{N}(0,1)$. Matrices $\{\mathbf{K}_t^{(\ell)}\}$ were formed with entries $\left[\mathbf{K}_t^{(\ell)}\right]_{ij} = \kappa(y_{it}, y_{jt})$ for some kernel $\kappa$. The entries of $\alpha_{ij} \in \mathbb{R}^T$ were drawn independently from $\mathcal{N}(0,1)$, while noise terms were generated independent identically distributed (i.i.d.) as $e_{ij} \sim \mathcal{N}(0, \sigma_e^2)$. For all tests, error plots were averaged over 100 independent runs.

We evaluated performance using the edge identification error rate (EIER) defined as EIER $:= \left(\|\mathrm{A} - \widehat{\mathrm{A}}\|_0\right) / \left(N(N-1)\right) \times 100$, where $\|\cdot\|_0$ denotes the number of nonzero entries of its argument. Fig. 5 plots EIER against the measurements-to-nodes ratio ($T/N$) under variable signal-to-noise ratios (SNRs), for a polynomial kernel. The synthetic graph was generated with edge probability 0.3. Fig. 5 plots the EIER when data are generated by (11), using a polynomial kernel of order $P = 2$. It is clear that nonlinear SVARMs exhibit markedly improved performance relative to linear SVARMs. This corroborates the effectiveness of the former in identifying the network topology when the dependencies among nodes are nonlinear.

In order to assess the edge detection performance, receiver operating characteristic (ROC) curves are plotted under different modeling assumptions in Fig. 6. With $P_D$ denoting the probability of detection, and $P_{FA}$ the probability of false alarms, each point on the ROC corresponds to a pair $(P_{FA}, P_D)$ for a prescribed threshold. Fig. 6(a) is obtained from tests run on data generated by Gaussian kernels with $\sigma^2 = 1$, while Fig. 6(b) corresponds to polynomial kernels of order $P = 2$. Using the area under the curve (AUC) as the edge-detection performance metric, Fig. 6(a) and (b) illustrates the benefits of accounting for nonlinearities. In both plots, kernel-based approaches result in higher AUC metrics as compared to approaches relying on linear SVARMs.

Fig. 6(c) depicts ROC curves parameterized by $\lambda$ for linear and kernel-based SVARMs, with simulated data generated using a linear SVARM. Not surprisingly, kernel-based SVARMs with polynomial kernels underperform the linear SVARM, due to the inherently present model mismatch. However, the kernel SVARM endowed with a multikernel learning scheme (MK-SVARM) is shown to attain comparable performance to the linear SVARM when the prescribed dictionary comprises both linear and polynomial kernels.

### B. Real Gene Expression Data

This section tests the performance of kernel-based SEMs, which can be viewed as a special case of the kernel-based SVARM in Section III-B. The experiments were carried out on gene regulatory data collected form 69 unrelated Nigerian individuals, under the International HapMap project [35]. From the 929 identified genes, expression levels and the genotypes of the expression quantitative trait loci (eQTLs) of 39 immune-related genes were selected and normalized; see [12] and [86] for detailed descriptions. Genotypes of eQTLs were adopted as exogenous inputs $\mathbf{X}$, and gene expression levels were treated as the endogenous variables $\mathbf{Y}$.

The underlying gene regulatory network topology was inferred by adopting both linear and nonlinear SEMs. For each algorithm, $\lambda$ was selected by fivefold cross validation. Fig. 7 shows the identified topologies, with the nodes annotated by their corresponding gene IDs. Fig. 7(a) depicts the resulting network based on a linear SEM, while Fig. 7(b) and (c) results from nonlinear SEMs based on a polynomial

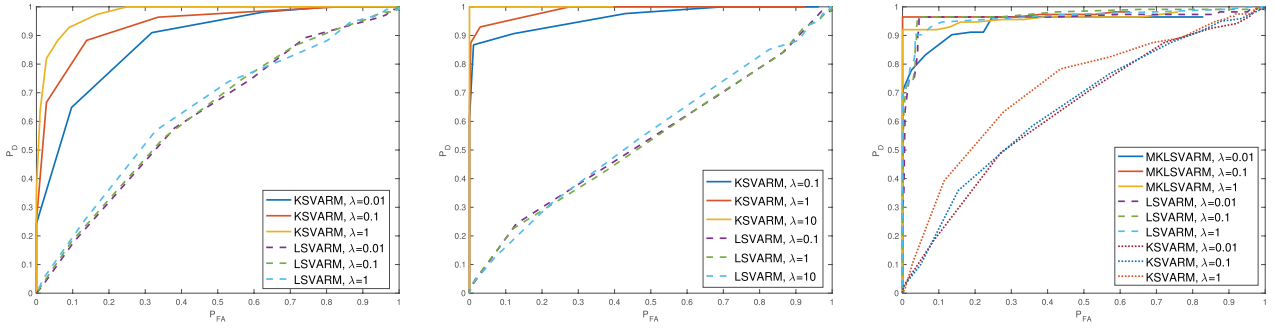


**Fig. 6.** ***ROC curves for data generated under different modeling assumptions: (a) K-SVARM based on a Gaussian kernel with $\sigma^2 = 1$; (b) K-SVARM based on a polynomial kernel of order P = 2; and (c) Linear SVARM.***

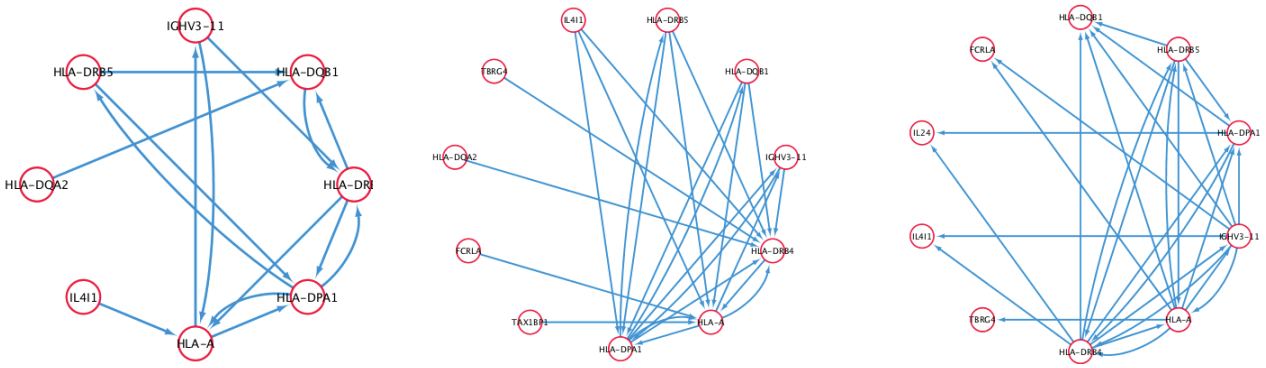


**Fig. 7.** ***Inferred gene regulatory networks for 39 immune-related genes based on gene expression data of T = 69 individuals using: (a) a linear SEM; (b) a kernel-based SEM using polynomial kernels of order 2; and (c) a kernel-based SEM using Gaussian kernels with $\sigma^2 = 1$ [105].***

kernel of order 2, and a Gaussian kernel with $\sigma^2 = 1$. In all cases, the identified networks are very sparse, and the nonlinear approaches unveil all edges identified by the linear SEMs, alongside with a number of additional edges. Clearly, considering the possibility that interactions among genes may be driven by nonlinear dynamics, nonlinear frameworks encompass linear approaches, and facilitate discovery of causal patterns not captured by linear SEMs.

### C. Inference of Real Temperature Data

This section tests the functional learning over graphs discussed in Section IV-A on a real data set. The data set comprises 24 signals corresponding to the average temperature per month in the intervals 1961–1990 and 1981–2010 measured by 89 stations in Switzerland [1]. The training set contains the first 12 signals, corresponding to the interval 1961–1990, whereas the test set contains the remaining 12. Each station is represented by a vertex and the graph was constructed using the algorithm in [25] based on the training signals. Given samples on a test signal on a randomly chosen subset of $M$ vertices, the values at the remaining $N - M$ vertices were estimated. NMSE is averaged across the test signals [92]. Fig. 8 compares the performance of the MKL schemes, along with single-kernel ridge regression (KRR), and estimators for bandlimited signals (BL) with different bandwidth $B$; see, e.g., [80]. The MKL adopts a dictionary consisting of ten diffusion kernels with parameter $\sigma^2$ uniformly spaced between 1 and 20. Single-kernel ridge regression uses diffusion kernels for different values of $\sigma^2$. It is clear from Fig. 8 that the MKL approach outperforms all other approaches.

### D. Recommendations for MovieLens

Here we evaluate the performance of the algorithm in Section IV-D on MovieLens 1M data set, which contains 3706 users, 6040 movies, and 1M ratings. The training and testing procedure follows that in [21]. Specifically, the data set is randomly split into training and testing sets, and a probe set is then formed by collecting only the five-star ratings in the testing set. The nonlinear model is then trained based on the training set, and for each user a list with top-$N_r$ recommended items is provided by the model. Performance of the novel sparse nonlinear method (SNLM) is compared with PureSVD with 50 and 100 leading eigenvectors [21], as well as SLIM [81], in terms of the recall and precision metrics defined, respectively, as recall($N_r$) := #hits/#probe, and precision($N_r$) := $N_r^{-1}$recall($N_r$), where #hits counts the number of ratings in the probe set that also appear in the recommendation list; and #probe is the number of ratings in the probe set. It can be readily observed from Fig. 9 and (10) that the nonlinear approach outperforms other approaches in both recall and precision.

### E. Resting State fMRI Synthetics—Static Topology

The MKL-based PC approach in Section III-A was evaluated on DCM-based synthetics [110]; see [63] for the choice of DCM parameters and a short description of the model. The upper triangular ground truth adjacency matrix

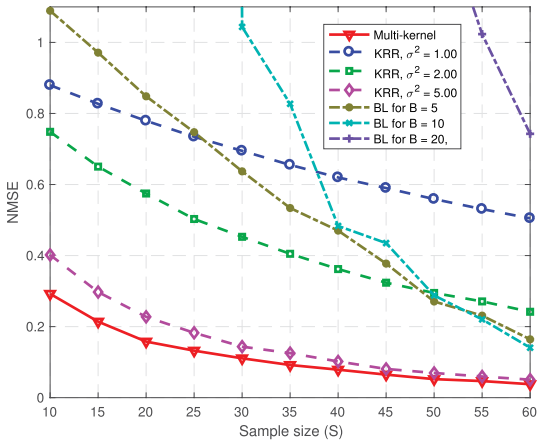


**Fig. 8.** ***NMSE obtained on the temperature data set.***

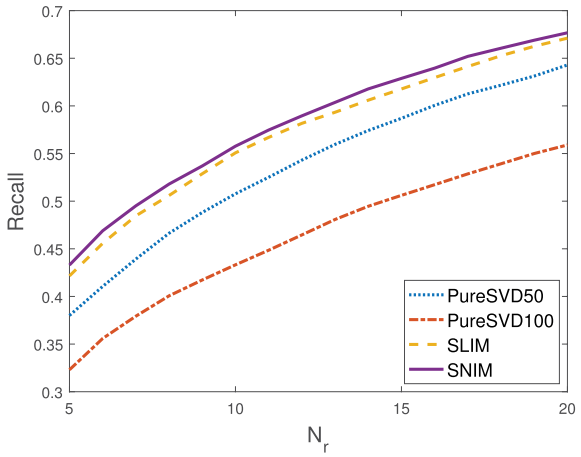


**Fig. 9.** ***Recall versus top $N_r$.***

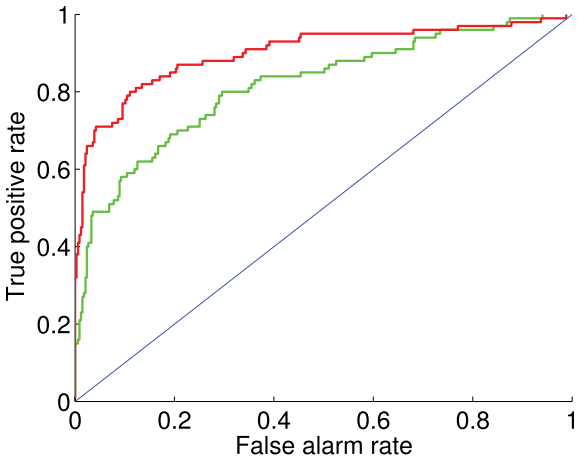


**Fig. 11.** ***ROC curves obtained on synthetic resting state fMRI data. The red curve corresponds to multi-kernel based partial correlation whereas the green one stands for ordinary PC [63].***

$\mathbf{A} \in \mathbb{R}^{30\times 30}$ contains 100 non-zero entries, each drawn as $[\mathbf{A}]_{ij} \sim \mathcal{U}(0.25, 0.6)$ with $\mathcal{U}$ denoting the uniform distribution, and placed at a random entry of the matrix. The resulting DCM-based time courses were of length $T = 200$.

The kernel dictionary consisted of a single linear kernel and 19 Gaussian kernels with variances belonging to $[10^{-6}, 1]$. The optimal regularization parameters were chosen via five-fold cross validation. ROC curves highlighting the performance gains over linear PC can be found in Fig. 11.

### F. Topology Identification for Meshed Power Grids

The MKL-based PC approach in Section III-A was further evaluated on the task of inferring meshed power grid topologies from voltage angle data [124]. The ground truth topology was that of the IEEE 14-bus benchmark system, whereas the (real) load data utilized were obtained from the 2012 Global Energy Forecasting Competition [124]. The voltage angles per bus, which in this context constitute the observations $\{y_{it}\}$, were computed via the alternating current (AC) power flow equations. The performance improvement achieved by the MKL-based PC approach over existing approaches is evident in Fig. 14.

### G. Resting State fMRI Synthetics—Time-Varying Topology

The performance of the MKL-based change point detection approach in Section V-G was assessed on DCM-based synthetics similar to the ones in Section VII-E, but here with a time-varying topology. It is assumed that $\mathbf{A} \in \mathbb{R}^{10\times 10}$ is replaced by its time-varying counterpart $\mathbf{A}_t$, with $\mathbf{A}_t = \mathbf{A}^{(m)}$ for $t \in [\tau_{m-1}, \tau_m - 1]$ and $m = 1, \ldots, N_m$, where $N_m$ denotes the number of segments; that is, $\mathbf{A}_t$ exhibits a switching behavior. The kernel dictionary used consisted of ten Gaussian kernels and a linear one. Letting the vector of change points $\boldsymbol{\tau} := [\tau_0 \ldots \tau_{N_m}]$ with $\tau_0 = 1$ and $\tau_{N_m} = T$, the following five temporal configurations were

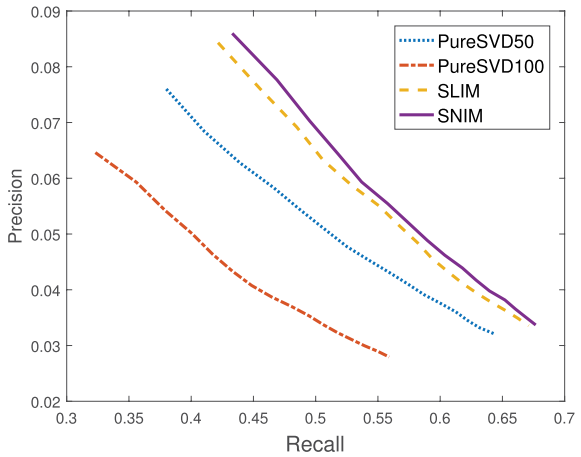


**Fig. 10.** ***Precision versus recall.***

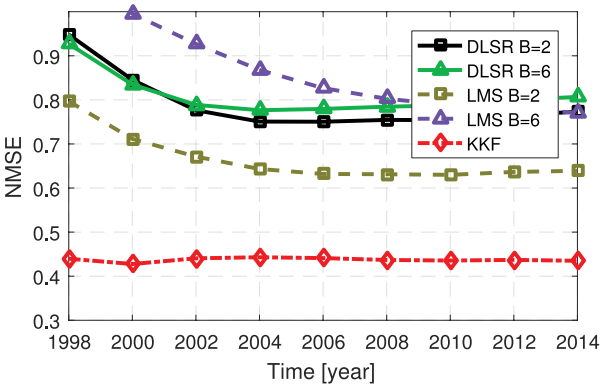


**Fig. 12.** ***NMSE obtained on the economic sectors data set [91].***

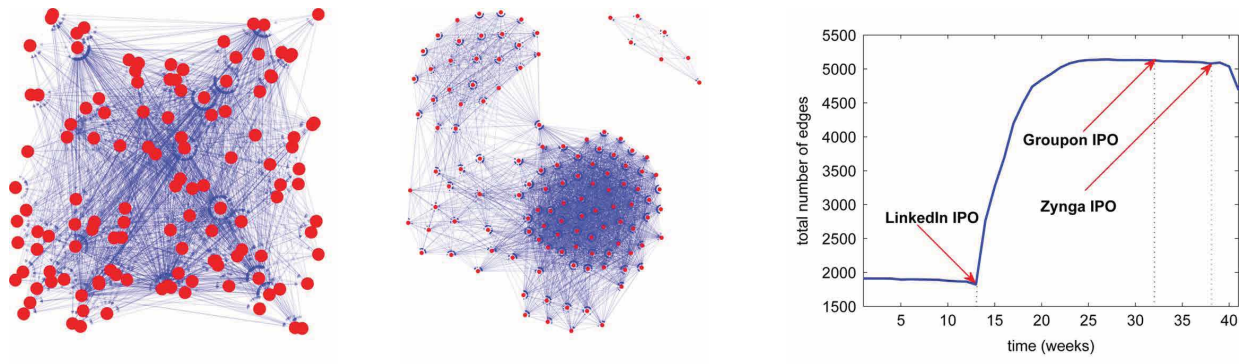


**Fig. 13.** ***Graphs of "Reid Hoffman" memes at t = 5 (left) t = 30 weeks (center); inferred edges per week (right) [5].***

considered: $\boldsymbol{\tau} = [1, 60, 110]$, $[1, 40, 70, 180]$, $[1, 60, 175, 250]$, $[1, 50, 90, 130, 180]$, $[1, 60, 120, 180, 240]$. For each configuration, five data sets were generated, and the topology corresponding to each segment was randomly drawn as in Section VII-E. The number of change points $N_m - 1$ was also inferred by the method. Finally, the minimum allowed segment length was set equal to 25.

The MKL-based change detection approach assigned 83% of the time points to the correct segment, achieving both accurate change point presence detection (avg. $|\hat{N}_m - N_m|$ of 0.4) and change point location estimation (avg. $|\hat{\tau} - \tau|$ of 13 time points).

### H. Inferring the Production of Economy Sectors

The reconstruction approach in [91] was evaluated on a data set provided by the Bureau of Economic Analysis of the U.S. Department of Commerce. In particular, given a set of graph snapshots $\{\mathbf{A}[t]\}$ with $[\mathbf{A}[t]]_{nm}$ denoting the investment in trillions of U.S. dollars between sectors $n$ and $m$ during the year $1995 + 2t$ for $t = 1, \ldots, 9$, the goal is to track the total production of sector $n'$, call it $f_{n'}[t]$, during the year $1996 + 2t$ for $t = 1, \ldots, 9$. The results are depicted in Fig. 12 that plots the normalized mean-square error for each year $1996 + 2t$ for the approach in [91] (KKF). For comparison purposes, the performance achieved by distributed least-squares reconstruction (DLSR) scheme, as well as by the least mean-squares algorithm (LMS) when applied to the temporally averaged graph $\overline{\mathbf{A}} = (1/9)\sum_{t=1}^{9}\mathbf{A}[t]$ (since these methods cannot handle TV topologies) is also provided. Note that KKF successfully leverages the TV topology, and significantly outperforms the rest of the approaches considered.

### I. Time-Varying Topology Inference From Information Cascades

The data set examined in [5] consists of memes (popular text phrases), and the time of their appearance on certain websites from March 2011 to February 2012. The subset of memes associated with the keyword "Reid Hoffman," the cofounder of LinkedIn, was considered. The observation $y_{it}^{(c)}$ corresponds to the timestamp of the appearance of meme $c$ on website $i$, if such an appearance occurred during the $t$-th week; otherwise, $y_{it}^{(c)}$ is set to a fixed large value. It is seen from Fig. 13 that there is an increase in the number of edges in the estimated networks coinciding with the initial public offering (IPO) of LinkedIn in May 2011. As time progresses, with other technology company IPOs emerging, this trend stops.

## VIII. CONCLUDING SUMMARY AND RESEARCH OUTLOOK

This paper outlined approaches to inferring connectivity of graphs and learning signals over graphs, while taking into account nonlinear and dynamic effects present. A reproducing-kernel Hilbert space framework for topology identification based on nodal observations was presented first to encompass a number of existing topology inference methods, and markedly broaden their scope. With the graph

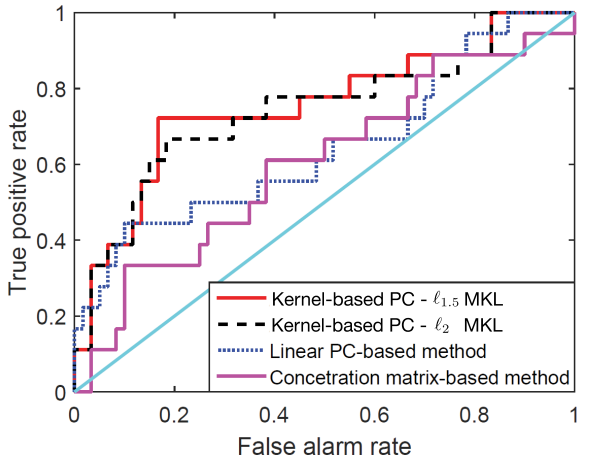


**Fig. 14.** ***ROC curves for topology inference of a meshed power grid from voltage angle data (adapted from [124]).***

topologies in hand, key learning tasks over graphs were considered to argue that accounting for nonlinear dependencies of signals residing on graphs generalizes existing learning methods and further improves their performance. Moreover, connectivity and inference over dynamic graphs was considered.

The vantage point of this overview opens up a number of exciting directions for future research, including: 1) broadening the scope of the nonlinear approach to dynamic settings; 2) establishing identifiability of the novel nonlinear and dynamic models for separate, as well as joint inference of signals and graphs; 3) exploring more efficient nonlinear inference algorithms via, e.g., online, parallel, and distributed implementations that are well motivated for large-scale networks; 4) analyzing the performance of nonlinear models for recommender systems; and 5) developing approaches for graph-aware detection, classification, and subspace clustering that account for nonlinearities and dynamics, both in semisupervised and unsupervised settings. ■

## REFERENCES


[1] *Meteorology and Climatology Meteoswiss*. [Online]. Available: http://www.meteoswiss.admin.ch/home/climate/past/climate-normals/climate-diagrams-and-normal-values-per-station.html

[2] A. Ahmed and E. P. Xing, "Recovering time-varying networks of dependencies in social and biological studies," *Proc. Nat. Acad. Sci. USA*, vol. 106, no. 29, pp. 11878–11883, Jul. 2009.

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

[4] B. Baingana and G. B. Giannakis, "Tracking switched dynamic network topologies from information cascades," *IEEE Trans. Signal Process.*, vol. 65, no. 4, pp. 985–997, Feb. 2017.

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

[6] A.-L. Barabási. (2012). *Network Science*. [Online]. Available: http://barabasi.com/networksciencebook

[7] J. A. Bazerque, B. Baingana, and G. B. Giannakis, "Identifiability of sparse structural equation models for directed and cyclic networks," in *Proc. Global Conf. Signal Inf. Process.*, Austin, TX, USA, Dec. 2013, pp. 839–842.

[8] J. A. Bazerque and G. B. Giannakis, "Nonparametric basis pursuit via sparse kernel-based learning: A unifying view with advances in blind methods," *IEEE Signal Process. Mag.*, vol. 30, no. 4, pp. 112–125, Jul. 2013.

[9] M. Belkin and P. Niyogi, "Laplacian Eigenmaps for dimensionality reduction and data representation," *Neural Comput.*, vol. 15, no. 6, pp. 1373–1396, 2003.

[10] S. Boccaletti, "The structure and dynamics of multilayer networks," *Phys. Rep.*, vol. 544, no. 1, pp. 1–122, Nov. 2014.

[11] H. Bozdogan, "Model selection and Akaike's information criterion (AIC): The general theory and its analytical extensions," *Psychometrika*, vol. 52, no. 3, pp. 345–370, 1987.

[12] X. Cai, J. A. Bazerque, and G. B. Giannakis, "Inference of gene regulatory networks with sparse structural equation models exploiting genetic perturbations," *PLoS Comput. Biol.*, vol. 9, p. e1003068, May 2013.

[13] V. D. Calhoun, R. Miller, G. Pearlson, and T. Adali, "The chronnectome: Time-varying connectivity networks as the next frontier in fMRI data discovery," *Neuron*, vol. 84, no. 2, pp. 262–274, Oct. 2014.

[14] D. Chakrabarti, R. Kumar, and A. Tomkins, "Evolutionary clustering," in *Proc. ACM SIGKDD Int. Conf. Know. Discovery Data Mining*, Philadelphia, PA, USA, Aug. 2006, pp. 554–560.

[15] O. Chapelle, B. Scholkopf, and A. Zien, *Semi-Supervised Learning*. Cambridge, MA, USA: MIT Press, 2006.

[16] G. Chen, "Vector autoregression, structural equation modeling, and their synthesis in neuroimaging data analysis," *Comput. Biol. Med.*, vol. 41, no. 12, pp. 1142–1155, Dec. 2011.

[17] S. Chen and P. S. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the Bayesian information Criterion," in *Proc. DARPA Broadcast News Transcription Understand.* Workshop, VA, USA, Aug. 1998, pp. 127–132.

[18] Y. Chi, X. Song, D. Zhou, K. Hino, and B. Tseng, "On evolutionary spectral clustering," *Trans. Know. Discovery Data*, vol. 3, no. 4, 2009, Art. no. 17.

[19] C. Cortes, M. Mohri, and A. Rostamizadeh, "$L_2$ regularization for learning kernels," in *Proc. Conf. Uncertainty Artif. Intell.*, Arlington, VA, USA, 2009, pp. 109–116.

[20] C. Cortes, M. Mohri, and A. Rostamizadeh, "Learning non-linear combinations of kernels," in *Proc. Adv. Neural Inf. Process. Sys.*, Vancouver, BC, Canada, 2009, pp. 396–404.

[21] P. Cremonesi, Y. Koren, and R. Turrin, "Performance of recommender algorithms on top-n recommendation tasks," in *Proc. ACM Conf. Rec. Syst.*, Barcelona, Spain, Sep. 2010, pp. 39–46.

[22] I. Cribben, R. Haraldsdottir, L. Y. Atlas, T. D. Wager, and M. A. Lindquist, "Dynamic connectivity regression: Determining state-related changes in brain connectivity," *NeuroImage*, vol. 61, no. 4, pp. 907–920, Jul. 2012.

[23] P. Di Lorenzo, S. Barbarossa, P. Banelli, and S. Sardellitti, "Adaptive least mean squares estimation of graph signals," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 2, no. 4, pp. 555–568, Dec. 2016.

[24] X. Dong, P. Frossard, P. Vandergheynst, and N. Nefedov, "Clustering with multi-layer graphs: A spectral perspective," *IEEE Trans. Signal Process.*, vol. 60, no. 11, pp. 5820–5831, Nov. 2012.

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

[26] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal $\ell^1$-norm solution is also the sparsest solution," *Commun. Pure Appl. Math.*, vol. 59, no. 6, pp. 797–829, 2006.

[27] D. M. Dunlavy, T. G. Kolda, and E. Acar, "Temporal link prediction using matrix and tensor factorizations," *ACM Trans. Knowl. Discovery Data*, vol. 5, no. 2, pp. 10:1–10:27, Feb. 2011.

[28] D. Easley and J. Kleinberg, *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. New York, NY, USA: Cambridge Univ. Press, 2010.

[29] H. E. Egilmez, E. Pavez, and A. Ortega, "Graph learning from data under Laplacian and structural constraints," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 6, pp. 825–841, Sep. 2017.

[30] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, Miami, FL, USA, Jun. 2009, pp. 2790–2797.

[31] F. Folino and C. Pizzuti, "An evolutionary multiobjective approach for community discovery in dynamic networks," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 8, pp. 1838–1852, Aug. 2014.

[32] P. A. Forero, K. Rajawat, and G. B. Giannakis, "Prediction of partially observed dynamical processes over networks via dictionary learning," *IEEE Trans. Signal Process.*, vol. 62, no. 13, pp. 3305–3320, Sep. 2014.

[33] S. Fortunato, "Community detection in graphs," *Phys. Rep.*, vol. 486, nos. 3–5, pp. 75–174, 2010.

[34] E. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky, "Nonparametric Bayesian learning of switching linear dynamical systems," in *Proc. 21st Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2008, pp. 457–464.

[35] K. A. Frazer, D. G. Ballinger, D. R. Cox, D. A. Hinds, and L. L. Stuve, "A second generation human haplotype map of over 3.1 million SNPs," *Nature*, vol. 449, no. 7164, pp. 851–861, 2007.

[36] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432–441, Jul. 2008.

[37] K. J. Friston, L. Harrison, and W. Penny, "Dynamic causal modelling," *NeuroImage*, vol. 19, no. 4, pp. 1273–1302, Aug. 2003.

[38] A. Ghasemian, P. Zhang, A. Clauset, C. Moore, and L. Peel, "Detectability thresholds and optimal algorithms for community structure in dynamic networks," *Phys. Rev. X*, vol. 6, p. 031005, Jul. 2016.

[39] A. Ghodsi, "Dimensionality reduction—A short tutorial," *Dept. Stat. Actuarial Sci., Univ. Waterloo, Waterloo, ON, Canada*, 2006, vol. 37.

[40] G. B. Giannakis, Q. Ling, G. Mateos, I. D. Schizas, and H. Zhu, "Decentralized learning for wireless communications and networking," in *Splitting Methods in*

*Communication, Imaging, Science, and Engineering*, R. Glowinski, S. Osher, and W. Yin, Eds. New York, NY, USA: Springer-Verlag, 2016.

[41] A. J. Gibberd and J. D. B. Nelson, "High dimensional changepoint detection with a dynamic graphical lasso," in *Proc. Int. Conf. Acoust. Speech Signal Process.*, Florence, Italy, May 2014, pp. 2684–2688.

[42] R. Goebel, A. Roebroeck, D.-S. Kim, and E. Formisano, "Investigating directed cortical interactions in time-resolved fMRI data using vector autoregressive modeling and Granger causality mapping," *Magn. Reson. Imag.*, vol. 21, no. 10, pp. 1251–1261, Dec. 2003.

[43] A. S. Goldberger, "Structural equation methods in the social sciences," *Econometrica*, vol. 40, no. 6, pp. 979–1001, Nov. 1972.

[44] C. W. J. Granger, "Investigating causal relations by econometric models and cross-spectral methods," *Econometrica*, vol. 37, no. 3, pp. 424–438, Aug. 1969.

[45] D. Hallac, Y. Park, S. Boyd, and J. Leskovec, "Network inference via the time-varying graphical lasso," in *Proc. ACM Int. Conf. Knowl. Discovery Data Mining*, Halifax, NS, Canada, 2017, pp. 205–213.

[46] J. Ham, D. D. Lee, S. Mika, and B. Schölkopf, "A kernel view of the dimensionality reduction of manifolds," in *Proc. Int. Conf. Mach. Learn.*, Alberta, AB, Canada, Jul. 2004, p. 47.

[47] J. D. Hamilton, *Time Series Analysis*. Princeton, NJ, USA: Princeton Univ. Press, 1994.

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

[49] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A k-means clustering algorithm," *Appl. Stat.*, vol. 28, no. 1, pp. 100–108, 1979.

[50] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. New York, NY, USA: Springer-Verlag, 2009.

[51] Z. Huang and D. K. Lin, "The time-series link prediction problem with applications in communication surveillance," *INFORMS J. Comput.*, vol. 21, no. 2, pp. 286–303, 2009.

[52] R. M. Hutchison, "Dynamic functional connectivity: Promise, issues, and interpretations," *NeuroImage*, vol. 80, pp. 360–378, Oct. 2013.

[53] V. N. Ioannidis, M. Ma, A. Nikolakopoulos, and G. B. Giannakis, "Kernel-based inference of functions on graphs," in *Adaptive Learning Methods for Nonlinear System Modeling*, D. Comminiello and J. Principe, Eds. Amsterdam, The Netherlands: Elsevier, 2018.

[54] V. N. Ioannidis, D. Romero, and G. B. Giannakis, "Inference of spatiotemporal processes over graphs via kernel kriged Kalman filtering," in *Proc. Eur. Signal Process. Conf.*, Kos, Greece, Aug./Sep. 2017, pp. 1679–1683.

[55] V. N. Ioannidis, Y. Shen, and G. B. Giannakis, "Semi-blind inference of topologies and signals over graphs," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, Calgary, AB, Canada, 2018.

[56] E. Isufi, A. Loukas, A. Simonetto, and G. Leus, "Autoregressive moving average graph filtering," *IEEE Trans. Signal Process.*, vol. 65, no. 2, pp. 274–288, Jan. 2017.

[57] B. Jiang, C. Ding, B. Luo, and J. Tang, "Graph-Laplacian PCA: Closed-form solution and robustness," in *Proc. IEEE Conf. Comput. Version Pattern Recognit. (CVPR)*, Jun. 2013, pp. 3492–3498.

[58] X. Jiang, S. Mahadevan, and A. Urbina, "Bayesian nonlinear structural equation modeling for hierarchical validation of dynamical systems," *Mech. Syst. Signal. Process.*, vol. 24, no. 4, pp. 957–975, Apr. 2010.

[59] T. Jin, J. Yu, J. You, K. Zeng, C. Li, and Z. Yu, "Low-rank matrix factorization with multiple Hypergraph regularizer," *Pattern Recognit.*, vol. 48, no. 3, pp. 1011–1022, Mar. 2015.

[60] I. Jolliffe, *Principal Component Analysis*. Hoboken, NJ, USA: Wiley, 2002.

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

[62] D. W. Kaplan, *Structural Equation Modeling: Foundations and Extensions*, 2nd ed. Newbury Park, CA, USA: Sage, 2009.

[63] G. V. Karanikolas, G. B. Giannakis, K. Slavakis, and R. M. Leahy, "Multi-kernel based nonlinear models for connectivity identification of brain networks," in *Proc. Int. Conf. Acoust. Speech Signal Process.*, Shanghai, China, Mar. 2016, pp. 6315–6319.

[64] G. V. Karanikolas, O. Sporns, and G. B. Giannakis, "Multi-kernel change detection for dynamic functional connectivity graphs," in *Proc. Asilomar Conf.*, Pacific Grove, CA, USA, Oct. 2017.

[65] A. Kelava, B. Nagengast, and H. Brandt, "A nonlinear structural equation mixture modeling approach for nonnormally distributed latent predictor variables," *Struct. Equ. Model., Multidiscipl. J.*, vol. 21, no. 3, pp. 468–481, Jun. 2014.

[66] M. Kivelä, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, "Multilayer networks," *J. Complex Netw.*, vol. 2, no. 3, pp. 203–271, 2014.

[67] E. D. Kolaczyk, *Statistical Analysis of Network Data: Methods and Models*. New York, NY, USA: Springer-Verlag, 2009.

[68] J. B. Kruskal and M. Wish, *Multidimensional Scaling*. Newbury Park, CA, USA: Sage, 1978.

[69] S.-Y. Lee and X.-Y. Song, "Model comparison of nonlinear structural equation models with fixed covariates," *Psychometrika*, vol. 68, no. 1, pp. 27–47, Mar. 2003.

[70] X. Li, N. Du, H. Li, K. Li, J. Gao, and A. Zhang, "A deep learning approach to link prediction in dynamic networks," in *Proc. SIAM Int. Conf. Data Mining*, Philadelphia, PA, USA, 2014, pp. 289–297.

[71] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," *J. Amer. Soc. Inf. Sci. Technol.*, vol. 58, no. 7, pp. 1019–1031, 2007.

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

[73] Y.-R. Lin, Y. Chi, S. Zhu, H. Sundaram, and B. L. Tseng, "Facetnet: A framework for analyzing communities and their evolutions in dynamic networks," in *Proc. Int. Conf. World Wide Web*, Beijing, China, Apr. 2008, pp. 685–694.

[74] D. Marinazzo, M. Pellicoro, and S. Stramaglia, "Kernel-Granger causality and the analysis of dynamical networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 77, no. 5, p. 056215, May 2008.

[75] D. Marinazzo, M. Pellicoro, and S. Stramaglia, "Kernel method for nonlinear Granger causality," *Phys. Rev. Lett.*, vol. 100, no. 14, p. 144103, Apr. 2008.

[76] A. Mclntosh and F. Gonzalez-Lima, "Structural equation modeling and its application to network analysis in functional brain imaging," *Human Brain Mapping*, vol. 2, nos. 1–2, pp. 2–22, Oct. 1994.

[77] J. Mei and J. M. F. Moura, "Signal processing on graphs: Causal modeling of unstructured data," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 2077–2092, Apr. 2017.

[78] C. A. Micchelli and M. Pontil, "Learning the kernel function via regularization," *J. Mach. Learn. Res.*, vol. 6, pp. 1099–1125, 2005.

[79] B. Muthén, "A general structural equation model with dichotomous, ordered categorical, and continuous latent variable indicators," *Psychometrika*, vol. 49, no. 1, pp. 115–132, Mar. 1984.

[80] S. K. Narang, A. Gadde, and A. Ortega, "Signal processing techniques for interpolation in graph structured data," in *Proc. Int. Conf. Acoust. Speech Signal Process.*, Vancouver, BC, Canada, May 2013, pp. 5445–5449.

[81] X. Ning and G. Karypis, "SLIM: Sparse linear methods for top-N recommender systems," in *Proc. Int. Conf. Data Mining*, Vancouver, BC, Canada, Dec. 2011, pp. 497–506.

[82] D. Nion and N. D. Sidiropoulos, "Adaptive algorithms to track the PARAFAC decomposition of a third-order tensor," *IEEE Trans. Signal Process.*, vol. 57, no. 6, pp. 2299–2310, Jun. 2009.

[83] D. Papakostas, P. Basaras, D. Katsaros, and L. Tassiulas, "Backbone formation in military multi-layer ad hoc networks using complex network concepts," in *Proc. Military Commun. Conf.*, Baltimore, MD, USA, Nov. 2016, pp. 842–848.

[84] E. Pavez and A. Ortega, "Generalized Laplacian precision matrix estimation for graph signal processing," in *Proc. Int. Conf. Acoust. Speech Signal Process.*, Shanghai, China, Mar. 2016, pp. 6350–6354.

[85] L. Peel and A. Clauset, "Detecting change points in the large-scale structure of evolving networks," in *Proc. AAAI Conf. Artif. Intell.*, Austin, TX, USA, Jan. 2015, pp. 2914–2920.

[86] J. K. Pickrell, "Understanding mechanisms underlying human gene expression variation with RNA sequencing," *Nature*, vol. 464, pp. 768–772, Apr. 2010.

[87] K. Rajawat, E. Dall'Anese, and G. B. Giannakis, "Dynamic network delay cartography," *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2910–2920, May 2014.

[88] J. W. Robinson and A. J. Hartemink, "Learning non-stationary dynamic Bayesian networks," *J. Mach. Learn. Res.*, vol. 11, pp. 3647–3680, Dec. 2010.

[89] A. Roebroeck, E. Formisano, and R. Goebel, "Mapping directed influence over the brain using Granger causality and fMRI," *NeuroImage*, vol. 25, no. 1, pp. 230–242, Mar. 2005.

[90] E. M. Rogers, *Diffusion of Innovations*. Washington, DC, USA: Free Press, 1995.

[91] D. Romero, V. N. Ioannidis, and G. B. Giannakis, "Kernel-based reconstruction of space-time functions on dynamic graphs," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 6, pp. 856–869, Sep. 2017.

[92] D. Romero, M. Ma, and G. B. Giannakis, "Kernel-based reconstruction of graph signals," *IEEE Trans. Signal Process.*, vol. 65, no. 3, pp. 764–778, Feb. 2017.

[93] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.

[94] S. Roy, Y. Atchadé, and G. Michailidis, "Change point estimation in high dimensional Markov random-field models," *J. Roy. Statist. Soc. B, Stat. Methodol.*, vol. 79, no. 4, pp. 1187–1206, Sep. 2017.

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

[96] P. Sarkar, D. Chakrabarti, and M. I. Jordan, "Nonparametric link prediction in dynamic networks," in *Proc. Int. Conf. Mach. Learn.*, Edinburgh, Scotland, Jul. 2012, pp. 1687–1694.

[97] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, "Consensus in ad hoc WSNs with noisy links—Part I: Distributed estimation of deterministic signals," *IEEE Trans. Signal Process.*, vol. 56, no. 1, pp. 350–364, Jan. 2008.

[98] B. Schölkopf, A. Smola, and K.-R. Müller, "Kernel principal component analysis," in *Proc. Int. Conf. Artif. Neural Netw.*, Lausanne, Switzerland, Oct. 1997, pp. 583–588.

[99] S. Segarra, A. G. Marques, G. Mateos, and A. Ribeiro, "Network topology inference from spectral templates," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 3, no. 3, pp. 467–483, Sep. 2017.

[100] N. Shahid, N. Perraudin, V. Kalofolias, G. Puy, and P. Vandergheynst, "Fast robust PCA on graphs," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 4, pp. 740–756, Jun. 2016.

[101] F. Shang, L. C. Jiao, and F. Wang, "Graph dual regularization non-negative matrix factorization for co-clustering," *Pattern Recognit.*, vol. 45, no. 6, pp. 2237–2250, Jun. 2012.

[102] Y. Shen, T. Chen, and G. B. Giannakis, "Online ensemble multi-kernel learning adaptive to non-stationary and adversarial environments," in *Proc. Int. Conf. Artif. Intell. Stat.*, Lanzarote, Spain, Apr. 2018.

[103] Y. Shen, B. Baingana, and G. B. Giannakis, "Inferring directed network topologies via tensor factorization," in *Proc. Asilomar Conf.*, Pacific Grove, CA, USA, Nov. 2016, pp. 1739–1743.

[104] Y. Shen, B. Baingana, and G. B. Giannakis, "Nonlinear structural vector autoregressive models for inferring effective brain network connectivity," *IEEE Trans. Med. Imag.*, to be published, doi: https://arxiv.org/abs/1610.06551

[105] Y. Shen, B. Baingana, and G. B. Giannakis, "Kernel-based structural equation models for topology identification of directed networks," *IEEE Trans. Signal Process.*, vol. 65, no. 10, pp. 2503–2516, May 2017.

[106] Y. Shen, B. Baingana, and G. B. Giannakis, "Tensor decompositions for identifying directed graph topologies and tracking dynamic networks," *IEEE Trans. Signal Process.*, vol. 65, no. 14, pp. 3675–3687, Jul. 2017.

[107] Y. Shen, X. Fu, G. B. Giannakis, and N. D. Sidiropoulos, "Inferring directed network topologies via joint diagonalization," in *Proc. Asilomar Conf.*, Pacific Grove, CA, USA, Nov. 2017.

[108] Y. Shen, P. A. Traganitis, and G. B. Giannakis, "Nonlinear dimensionality reduction on graphs," in *Proc. CAMSAP*, Dec. 2017.

[109] 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 Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2013.

[110] S. M. Smith, "Network modelling methods for FMRI," *NeuroImage*, vol. 54, no. 2, pp. 875–891, Jan. 2011.

[111] A. J. Smola and R. I. Kondor, "Kernels and regularization on graphs," in *Learning Theory and Kernel Machines*. Berlin, Germany: Springer-Verlag, 2003, pp. 144–158.

[112] J. Sun, C. Faloutsos, S. Papadimitriou, and P. S. Yu, "Graphscope: Parameter-free mining of large time-evolving graphs," in *Proc. ACM SIGKDD Intel. Conf. Knowledge Discovery Data Mining*, San Jose, CA, USA, May 2007, pp. 687–696.

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

[114] Y. Sun and J. Han, "Mining heterogeneous information networks: A structural analysis approach," *ACM SIGKDD Explorations Newslett.*, vol. 14, no. 2, pp. 20–28, 2013.

[115] Y. Sun, J. Tang, J. Han, M. Gupta, and B. Zhao, "Community evolution detection in dynamic heterogeneous information networks," in *Proc. Mining Learn. Graphs*, Washington, DC, USA, 2010, pp. 137–146.

[116] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Process. Lett.*, vol. 9, no. 3, pp. 293–300, Jun. 1999.

[117] T. Thorne and M. P. H. Stumpf, "Inference of temporally varying Bayesian networks," *Bioinformatics*, vol. 28, no. 24, pp. 3298–3305, 2012.

[118] P. A. Traganitis and G. B. Giannakis, "Sketched subspace clustering," *IEEE Trans. Signal Process.*, to be published.

[119] P. Traganitis, Y. Shen, and G. B. Giannakis, "Topology inference for multilayer networks," in *Proc. Int. Workshop Netw. Sci. Commun.*, Atlanta, GA, USA, May 2017.

[120] L. J. Vostrikova, "Detecting 'disorder' in multidimensional random processes," *Soviet Math., Doklady*, vol. 24, pp. 55–59, 1981.

[121] M. Wall and Y. Amemiya, "Estimation for polynomial structural equation models," *J. Amer. Stat. Assoc.*, vol. 95, no. 451, pp. 929–940, Sep. 2000.

[122] K. S. Xu and A. O. Hero, "Dynamic stochastic blockmodels for time-evolving social networks," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 4, pp. 552–562, Aug. 2014.

[123] O. Zamir and O. Etzioni, "Grouper: A dynamic clustering interface to Web search results," *Comput. Netw.*, vol. 31, nos. 11–16, pp. 1361–1374, 1999.

[124] L. Zhang, G. Wang, and G. B. Giannakis, "Going beyond linear dependencies to unveil connectivity of meshed grids," in *Proc. CAMSAP*, Curacao, Dutch Antilles, Dec. 2017.

[125] L. Zhang, D. Romero, and G. B. Giannakis, "Fast convergent algorithms for multi-kernel regression," in *Proc. IEEE Workshop Stat. Signal Process.*, Palma de Mallorca, Spain, Jun. 2016, pp. 1–4.

[126] D. Zhou, S. A. Orshanskiy, H. Zha, and C. L. Giles, "Co-ranking authors and documents in a heterogeneous network," in *Proc. Int. Conf. Data Mining*, Omaha, NE, USA, Oct. 2007, pp. 739–744.

[127] S. Zhou, J. Lafferty, and L. Wasserman, "Time varying undirected graphs," *Mach. Learn.*, vol. 80, nos. 2–3, pp. 295–319, Sep. 2010.

## ABOUT THE AUTHORS

**Georgios B. Giannakis** (Fellow, IEEE) received the Diploma in electrical engineering from the National Technical University of Athens, Athens, Greece, 1981 and the MSc. degree in electrical engineering, the M.Sc. degree in mathematics, and the Ph.D. degree in electrical engineering from the University of Southern California (USC), Los Angeles, CA, USA, in 1983, 1086, and 1986, respectively.

He was with the University of Virginia, Charlottesville, VA, USA, from 1987 to 1998, and since 1999, he has been a Professor with the University of Minnesota, Twin Cities, MN, USA, where he holds an Endowed Chair in Wireless Telecommunications, a University of Minnesota McKnight Presidential Chair in ECE, and serves as Director of the Digital Technology Center. His general interests span the areas of communications, networking and statistical signal processing, subjects on which he has published more than 400 journal papers, 700 conference papers, 25 book chapters, two edited books and two research monographs (h-index 130). His current research focuses on learning from big data, wireless cognitive radios, and network science with applications to social, brain, and power networks with renewables.

Prof. Giannakis is the (co)inventor of 32 patents issued, and the (co)recipient of nine best paper awards from the IEEE Signal Processing (SP) and Communications Societies, including the G. Marconi Prize

Paper Award in Wireless Communications. He also received Technical Achievement Awards from the SP Society (2000), from EURASIP (2005), a Young Faculty Teaching Award, the G. W. Taylor Award for Distinguished Research from the University of Minnesota, and the IEEE Fourier Technical Field Award (inaugural recipient in 2015). He is a Fellow of EURASIP, and has served the IEEE in a number of posts, including that of a Distinguished Lecturer for the IEEE-SP Society.

**Yanning Shen** (Student Member, IEEE) received the B.Sc. and M.Sc degrees in electrical engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2011 and 2014, respectively. Currently, she is working toward the Ph.D. degree at the Department of Electrical and Computer Engineering, University of Minnesota, Twin Cities, MN, USA.

Her general research interests include signal processing, network science and machine learning.

**Georgios Vasileios Karanikolas** (Student Member, IEEE) received the Diploma (valedictorian) in electrical and computer engineering from the University of Patras, Patras, Greece, in 2014 and the M.Sc. degree in electrical and computer engineering from the University of Minnesota, Twin Cities, MN, USA, in 2016, where he is currently working toward the Ph.D. degree at the Department of Electrical and Computer Engineering.

His research interests lie in the broad areas of machine learning, signal processing and network science, with emphasis on networks of the brain.