A PROXIMAL GRADIENT ALGORITHM FOR TRACKING CASCADES OVER NETWORKS

Brian Baingana, Gonzalo Mateos, and Georgios B. Giannakis

Dept. of ECE and Digital Technology Center, University of Minnesota, Minneapolis, MN

ABSTRACT

Many real-world processes evolve in cascades over networks, whose topologies are often unobservable and change over time. However, the so-termed adoption times when for instance blogs mention popular news items are typically known, and are implicitly dependent on the underlying network. To infer the network topology, a *dynamic* structural equation model is adopted to capture the relationship between observed adoption times and the unknown edge weights, while accounting also for external (non-topological) perturbations. Assuming a slowly time-varying topology and leveraging the sparse connectivity inherent to social networks, edge weights are estimated by minimizing a sparsity-regularized exponentially-weighted least-squares criterion. To this end, a solver is developed by leveraging (pseudo) real-time sparsity-promoting proximal gradient iterations. Numerical tests with real cascades of online media demonstrate the effectiveness of the novel algorithm in unveiling sparse dynamically-evolving topologies.

Index Terms— Social network, structural equation model, cascade, topology inference, convex optimization.

1. INTRODUCTION

Networks arising in natural and man-made settings provide the backbone for the propagation of *contagions* such as the spread of popular news stories, the adoption of buying trends among consumers, and the spread of infectious diseases [1, 2]. For example, a terrorist attack may be reported within minutes on mainstream news websites. An information cascade emerges because these websites' readership typically includes bloggers who write about the attack as well, influencing their own readers in turn to do the same. Although the times when "nodes" get infected are often observable, the underlying network topologies over which cascades propagate are typically unknown and dynamic. Knowledge of the topology plays a crucial role for several reasons e.g., when social media advertisers select a small set of initiators so that an online campaign can go viral, or when healthcare initiatives wish to infer hidden needle-sharing networks of injecting drug users.

The propagation of a contagion is tantamount to *causal* effects or interactions being exerted among entities such as news portals and blogs, consumers, or people susceptible to being infected with a contagious disease. Acknowledging this viewpoint, *structural equation models* (SEMs) provide a general statistical modeling technique to estimate causal relationships among traits; see e.g., [3, 4]. These directional effects are often not revealed by standard linear models involving symmetric associations between random variables, such as those represented by covariances or correlations, [5], [6], [7]. SEMs are attractive because of their simplicity and ability to capture edge directionalities. They have been widely adopted in many fields, such as economics, psychometrics [8], social sciences [9], and recently in genetics for *static* gene regulatory network inference; see e.g., [10, 11] and references therein. However, SEMs have not been utilized to track the dynamics of causal effects among interacting nodes. In this context, the present paper proposes a *dynamic* SEM to account for time-varying directed networks over which contagions propagate, and describes how node infection times depend on both topological and external influences. Accounting for external influences is well motivated by drawing upon examples from online media, where established news websites depend more on on-site reporting than blog references. External influence data is also useful for model identifiability, and has been shown necessary to resolve directional ambiguities [12].

Supposing the network varies slowly with time, parameters in the proposed dynamic SEM are estimated adaptively by minimizing a sparsity-promoting exponentially-weighted least-squares (LS) criterion (Section 2). To account for the inherently sparse connectivity of social networks, an ℓ_1-norm regularization term that promotes sparsity on the entries of the network adjacency matrix is incorporated in the cost function; see also [13, 14, 15]. A novel algorithm to jointly track the network's adjacency matrix and the weights capturing the level of external influences is developed in Section 3, which minimizes the resulting non-differentiable cost function via a proximal-gradient (PG) solver; see e.g., [16, 17, 18]. The resulting dynamic iterative shrinkage-thresholding algorithm (ISTA) is provably convergent, and offers parallel, closed-form, and sparsity-promoting updates per iteration. Corroborating experiments in Section 4 involve real temporal traces of popular global events that propagated on news websites and blogs in 2011 [19].

Related work. Inference of networks using temporal traces of infection events has recently become an active area of research. According to the taxonomy in [20, Ch. 7], this can be viewed as a problem involving inference of *association* networks. Several prior approaches postulate probabilistic models and rely on maximum likelihood estimation (MLE) to infer edge weights as pairwise transmission rates between nodes [21], [22]. However, these methods assume that the network does not change over time. A dynamic algorithm has been recently proposed to infer time-varying diffusion networks by solving an MLE problem via stochastic gradient descent iterations [23]. Although successful experiments on large-scale web data reliably uncover information pathways, the estimator in [23] does not explicitly account for edge sparsity prevalent in social and information networks. Moreover, most prior approaches only attribute node infection events to the network topology, and do not account for the influence of external sources such as a ground crew for a mainstream media website.

2. NETWORK MODEL AND PROBLEM STATEMENT

Consider a dynamic network with N nodes observed over time intervals $t = 1, \dots, T$, whose abstraction is a graph with topology de-

Work in this paper was supported by the NSF ECCS Grants No. 1202135 and No. 1343248, and the NSF AST Grant No. 1247885.

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

scribed by an unknown, time-varying, and weighted adjacency matrix $\mathbf{A}^t \in \mathbb{R}^{N\times N}$. Entry (i,j) of $\mathbf{A}^t$ (henceforth denoted by a^t_{ij}) is nonzero only if a directed edge connects nodes i and j (pointing from j to i) during the time interval t. As a result, one in general has $a^t_{ij} \neq a^t_{ji}$, i.e., matrix $\mathbf{A}^t$ is generally non-symmetric, which is suitable to model directed networks. Note that the model tacitly assumes that the network topology remains fixed during any given time interval t, but can change across time intervals.

Suppose C contagions propagate over the network, and the infection time of node i by contagion c is denoted by y^t_{ic}. In online media, y^t_{ic} can be obtained by recording the time when website i mentions news item c. For uninfected nodes at slot t, y^t_{ic} is set to an arbitrarily large number. Assume that the susceptibility x_{ic} of node i to external (non-topological) infection by contagion c is known and time invariant over the observation interval. In the web context, x_{ic} can be set to the search engine rank of website i with respect to (w.r.t.) keywords associated with c.

The infection time of node i during interval t is modeled according to the following *dynamic* structural equation model (SEM)

$$y^t_{ic} = \sum_{j\neq i} a^t_{ij} y^t_{jc} + b^t_{ii} x_{ic} + e^t_{ic} \tag{1}$$

where b^t_{ii} captures the time-varying level of influence of external sources, and e^t_{ic} accounts for measurement errors and unmodeled dynamics. It follows from (1) that if $a^t_{ij} \neq 0$, then y^t_{ic} is affected by the value of y^t_{jc}. With diagonal $\mathbf{B}^t := \text{diag}(b_{11}, \ldots, b_{NN})$, collecting observations for the entire network and all C contagions yields the dynamic matrix SEM

$$\mathbf{Y}^t = \mathbf{A}^t\mathbf{Y}^t + \mathbf{B}^t\mathbf{X} + \mathbf{E}^t \tag{2}$$

where $\mathbf{Y}^t := [y^t_{ic}]$, $\mathbf{X} := [x_{ic}]$, and $\mathbf{E}^t := [e^t_{ic}]$ are all $N \times C$ matrices. A single network topology $\mathbf{A}^t$ is adopted for all contagions, which is suitable e.g., when information cascades are formed around a common meme or trending (news) topic in the Internet; see also the data in Section 4.

Problem statement. Given $\{\mathbf{Y}^t\}_{t=1}^T$ and $\mathbf{X}$ adhering to (2), the goal is to track the underlying network topology $\{\mathbf{A}^t\}_{t=1}^T$ and the effect of external influences $\{\mathbf{B}^t\}_{t=1}^T$.

To this end, the idea is to leverage the inherent sparsity of edges that is typical of social networks, and assume that the network topology is slowly time-varying. For $t = 1, \ldots, T$, consider the sparsity-regularized exponentially-weighted LS estimator (EWLSE)

$$\{\hat{\mathbf{A}}^t, \hat{\mathbf{B}}^t\} = \arg\min_{\mathbf{A},\mathbf{B}} \frac{1}{2}\sum_{\tau=1}^{t} \beta^{t-\tau}\|\mathbf{Y}^\tau - \mathbf{A}\mathbf{Y}^\tau - \mathbf{B}\mathbf{X}\|_F^2 + \lambda_t\|\mathbf{A}\|_1$$
$$\text{s. to } a_{ii} = 0,\ b_{ij} = 0,\ \forall i \neq j \tag{3}$$

where $\beta \in (0,1]$ is the forgetting factor that forms estimates $\{\hat{\mathbf{A}}^t, \hat{\mathbf{B}}^t\}$ using all measurements acquired until time t. Whenever $\beta < 1$, past data are exponentially discarded thus enabling tracking of dynamic network topologies. Moreover, $\|\mathbf{A}\|_1 := \sum_{i,j}|a_{ij}|$ is a sparsity-promoting regularization, and $\lambda_t > 0$ controls the sparsity level of $\hat{\mathbf{A}}$. In the experiments of Section 4, a time-invariant value of λ is adopted and typically chosen via trial and error to optimize the performance; see also the extended journal version of this paper for a more in-depth discussion on the tuning of λ_t [24, Remark 3]. Absence of a self-loop at node i is enforced by the constraint $a_{ii} = 0$, while having $b_{ij} = 0$, $\forall i \neq j$, ensures that $\hat{\mathbf{B}}$ is diagonal as in (2).

3. TOPOLOGY TRACKING ALGORITHM

Proximal gradient (PG) algorithms have been popularized for ℓ_1-norm regularized linear regression problems, through the class of iterative shrinkage-thresholding algorithms (ISTA); see e.g., [17] and [16] for a comprehensive tutorial treatment. The main advantage of ISTA over off-the-shelf interior point methods is its computational simplicity. Iterations boil down to matrix-vector multiplications involving the regression matrix, followed by a soft-thresholding operation [25, p. 93].

In the sequel, an ISTA algorithm is developed for the sparsity regularized dynamic SEM formulation (3) at time t. Based on this module, a (pseudo) real-time algorithm for tracking the dynamically-evolving network topology over the horizon $t = 1, \ldots, T$ is obtained as well. The algorithm's memory storage requirement and computational cost per sample $\{\mathbf{Y}^t, \mathbf{X}\}$ does not grow with t.

Solving (3) for a single time interval t. Introducing the optimization variable $\mathbf{V} := [\mathbf{A}\ \mathbf{B}]$, it follows that the gradient of $f(\mathbf{V}) := \frac{1}{2}\sum_{\tau=1}^{t}\beta^{t-\tau}\|\mathbf{Y}^\tau - \mathbf{A}\mathbf{Y}^\tau - \mathbf{B}\mathbf{X}\|_F^2$ is Lipschitz continuous, i.e., $\|\nabla f(\mathbf{V}_1) - \nabla f(\mathbf{V}_2)\| \leq L_f\|\mathbf{V}_1 - \mathbf{V}_2\|, \forall\, \mathbf{V}_1, \mathbf{V}_2$ in the domain of f. The Lipschitz constant L_f is time varying, but the dependency on t is kept implicit for notational convenience. Instead of directly optimizing the cost in (3), PG algorithms minimize a sequence of overestimators evaluated at judiciously chosen points (typically the current iterate, or a linear combination of the two previous iterates).

With $k = 1, 2, \ldots$ denoting iterations and upon defining $g(\mathbf{V}) := \lambda_t\|\mathbf{A}\|_1$, PG algorithms iterate

$$\mathbf{V}[k] := \arg\min_{\mathbf{V}}\left\{\frac{L_f}{2}\|\mathbf{V} - \mathbf{G}(\mathbf{V}[k-1])\|_F^2 + g(\mathbf{V})\right\} \tag{4}$$

where $\mathbf{G}(\mathbf{V}[k-1]) := \mathbf{V}[k-1] - (1/L_f)\nabla f(\mathbf{V}[k-1])$ corresponds to a gradient-descent step taken from $\mathbf{V}[k-1]$, with step-size equal to $1/L_f$. The optimization problem (4) is known as the *proximal operator* of the function g/L_f evaluated at $\mathbf{G}(\mathbf{V}[k-1])$, and is denoted as $\text{prox}_{g/L_f}(\mathbf{G}(\mathbf{V}[k-1]))$. Henceforth adopting the notation $\mathbf{G}[k-1] := \mathbf{G}(\mathbf{V}[k-1])$ for convenience, the PG iterations can be compactly rewritten as $\mathbf{V}[k] = \text{prox}_{g/L_f}(\mathbf{G}[k-1])$.

A key element to the success of PG algorithms stems from the possibility of efficiently evaluating the proximal operator (cf. (4)). Specializing to (3), note that (4) decomposes into

$$\mathbf{A}[k] := \arg\min_{\mathbf{A}}\left\{\frac{L_f}{2}\|\mathbf{A} - \mathbf{G}_A[k-1]\|_F^2 + \lambda_t\|\mathbf{A}\|_1\right\}$$
$$= \mathcal{S}_{\lambda_t/L_f}(\mathbf{G}_A[k-1]) \tag{5}$$
$$\mathbf{B}[k] := \arg\min_{\mathbf{B}}\left\{\|\mathbf{B} - \mathbf{G}_B[k-1]\|_F^2\right\} = \mathbf{G}_B[k-1] \tag{6}$$

subject to the constraints in (3) which so far have been left implicit, and $\mathbf{G} := [\mathbf{G}_A\ \mathbf{G}_B]$. Letting $\mathcal{S}_\mu(\mathbf{M})$ with (i,j)-th entry given by $\text{sign}(m_{ij})\max(|m_{ij}| - \mu, 0)$ denote the soft-thresholding operator, it follows that $\text{prox}_{\lambda_t\|\cdot\|_1/L_f}(\cdot) = \mathcal{S}_{\lambda_t/L_f}(\cdot)$, e.g., [17, 25]. Because there is no regularization on the matrix $\mathbf{B}$, the corresponding update (6) boils-down to a simple gradient-descent step.

What remains now is to obtain expressions for the gradient of $f(\mathbf{V})$ with respect to $\mathbf{A}$ and $\mathbf{B}$, which are required to form the matrices $\mathbf{G}_A$ and $\mathbf{G}_B$. To this end, note that by incorporating the constraints $a_{ii} = 0$ and $b_{ij} = 0$, $\forall j \neq i, i = 1, \ldots N$, one can simplify the expression of $f(\mathbf{V})$ as

$$f(\mathbf{V}) := \frac{1}{2}\sum_{\tau=1}^{t}\sum_{i=1}^{N}\beta^{t-\tau}\|(\mathbf{y}_i^\tau)^\top - \mathbf{a}_{-i}^\top\mathbf{Y}_{-i}^\tau - b_{ii}\mathbf{x}_i^\top\|_F^2 \tag{7}$$

where $(\mathbf{y}_i^\tau)^\top$ and $\mathbf{x}_i^\top$ denote the i-th row of $\mathbf{Y}^\tau$ and $\mathbf{X}$, respectively; while $\mathbf{a}_{-i}^\top$ denotes the $1 \times (N-1)$ vector obtained by removing entry i from the i-th row of $\mathbf{A}$, and likewise $\mathbf{Y}_{-i}^\tau$ is the $(N-1) \times C$ matrix obtained by removing row i from $\mathbf{Y}^\tau$. It is apparent from (7) that $f(\mathbf{V})$ is separable across the trimmed row vectors $\mathbf{a}_{-i}^\top$, and the diagonal entries b_{ii}, $i = 1, \dots, N$. The sought gradients are

$$\nabla_{\mathbf{a}_{-i}} f(\mathbf{V}) = \mathbf{\Sigma}_{-i}^t \mathbf{a}_{-i} + \bar{\mathbf{Y}}_{-i}^t \mathbf{x}_i b_{ii} - \boldsymbol{\sigma}_{-i}^t \tag{8}$$

$$\nabla_{b_{ii}} f(\mathbf{V}) = \mathbf{a}_{-i}^\top \bar{\mathbf{Y}}_{-i}^t \mathbf{x}_i + \frac{1-\beta^t}{1-\beta} b_{ii} \|\mathbf{x}_i\|_2^2 - (\bar{\mathbf{y}}_i^\tau)^\top \mathbf{x}_i. \tag{9}$$

where $(\bar{\mathbf{y}}_i^t)^\top$ denotes the i-th row of $\bar{\mathbf{Y}}^t := \sum_{\tau=1}^t \beta^{t-\tau}\mathbf{Y}^\tau$, and $\bar{\mathbf{Y}}_{-i}^t := \sum_{\tau=1}^t \beta^{t-\tau}\mathbf{Y}_{-i}^\tau$. Similarly, $\boldsymbol{\sigma}_{-i}^t := \sum_{\tau=1}^t \beta^{t-\tau}\mathbf{Y}_{-i}^\tau \mathbf{y}_i^\tau$ and $\mathbf{\Sigma}_{-i}^t$ is obtained by removing the i-th row and i-th column from $\mathbf{\Sigma}^t := \sum_{\tau=1}^t \beta^{t-\tau}\mathbf{Y}^\tau(\mathbf{Y}^\tau)^\top$. From (5)-(6) and (8)-(9), the parallel ISTA iterations

$$\nabla_{\mathbf{a}_{-i}} f[k] = \mathbf{\Sigma}_{-i}^t \mathbf{a}_{-i}[k] + \bar{\mathbf{Y}}_{-i}^t \mathbf{x}_i b_{ii}[k] - \boldsymbol{\sigma}_{-i}^t \tag{10}$$

$$\nabla_{b_{ii}} f[k] = \mathbf{a}_{-i}^\top[k] \bar{\mathbf{Y}}_{-i}^t \mathbf{x}_i + \frac{(1-\beta^t)}{1-\beta} b_{ii}[k] \|\mathbf{x}_i\|_2^2 - (\bar{\mathbf{y}}_i^t)^\top \mathbf{x}_i \tag{11}$$

$$\mathbf{a}_{-i}[k+1] = \mathcal{S}_{\lambda_t/L_f}\left(\mathbf{a}_{-i}[k] - (1/L_f)\nabla_{\mathbf{a}_{-i}} f[k]\right) \tag{12}$$

$$b_{ii}[k+1] = b_{ii}[k] - (1/L_f)\nabla_{b_{ii}} f[k] \tag{13}$$

are provably convergent to the globally optimal solution $\{\hat{\mathbf{A}}^t, \hat{\mathbf{B}}^t\}$ of (3), as per the general convergence results available for PG methods and ISTA in particular [17, 16].

Computation of the gradients in (10)-(11) requires one matrix-vector mutiplication by $\mathbf{\Sigma}_{-i}^t$ and one by $\bar{\mathbf{Y}}_{-i}^t$, in addition to three vector inner-products, plus a few (negligibly complex) scalar and vector additions. Both the update of $b_{ii}[k+1]$ as well as the soft-thresholding operation in (12) entail negligible computational complexity. All in all, the simplicity of the resulting iterations should be apparent. Per iteration, the actual rows of the adjacency matrix are obtained by zero-padding the updated $\mathbf{a}_{-i}[k]$, namely setting

$$\mathbf{a}_i^\top[k] = [a_{-i,1}[k] \dots a_{-i,i-1}[k]\; 0\; a_{-i,i}[k] \dots a_{-i,N}[k]]. \tag{14}$$

This way, the desired SEM parameter estimates at time t are given by $\hat{\mathbf{A}}^t = [\mathbf{a}_1^\top[k], \dots, \mathbf{a}_N^\top[k]]^\top$ and $\hat{\mathbf{B}}^t = \text{diag}(b_{11}[k], \dots, b_{NN}[k])$, for k large enough so that convergence has been attained.

Solving (3) **over the entire time horizon** $t = 1, \dots, T$**.** To track the dynamically-evolving network topology, one can go ahead and solve (3) sequentially for each $t = 1, \dots, T$ as data arrive, using (10)-(13). (The procedure can also be adopted in a batch setting, when all $\{\mathbf{Y}^t\}_{t=1}^T$ are available in memory.) Because the network is assumed to vary slowly across time, it is convenient to warm-restart the ISTA iterations, that is, at time t initialize $\{\mathbf{A}[0], \mathbf{B}[0]\}$ with the solution $\{\hat{\mathbf{A}}^{t-1}, \hat{\mathbf{B}}^{t-1}\}$. This way, for smooth network variations one expects convergence to be attained after few iterations.

To obtain the new SEM parameter estimates via (10)-(13), it suffices to update (possibly) λ_t and the Lipschitz constant L_f, as well as the data-dependent EWMAs $\mathbf{\Sigma}^t$, and $\bar{\mathbf{Y}}^t$. Interestingly, the potential growing-memory problem in storing the entire history of data $\{\mathbf{Y}^t\}_{t=1}^T$ can be avoided by performing the recursive updates

$$\mathbf{\Sigma}^t = \beta\mathbf{\Sigma}^{t-1} + \mathbf{Y}^t(\mathbf{Y}^t)^\top, \quad \bar{\mathbf{Y}}^t = \beta\bar{\mathbf{Y}}^{t-1} + \mathbf{Y}^t. \tag{15}$$

The complexity in evaluating the Gram matrix $\mathbf{Y}^t(\mathbf{Y}^t)^\top$ dominates the per-iteration computational cost of the algorithm. To circumvent the need of recomputing the Lipschitz constant per time interval, the step-size $1/L_f$ in (12)-(13) can be selected by a line search [16]. One choice is the backtracking step-size rule [18], for which convergence to $\{\hat{\mathbf{A}}^t, \hat{\mathbf{B}}^t\}$ can be established as well.

Algorithm 1 Pseudo real-time ISTA for topology tracking

```
Require: $\{\mathbf{Y}^t\}_{t=1}^T$, $\mathbf{X}$, $\beta$.
Initialize $\hat{\mathbf{A}}^0 = \mathbf{0}_{N\times N}, \hat{\mathbf{B}}^0 = \mathbf{\Sigma}^0 = \mathbf{I}_N, \bar{\mathbf{Y}}^0 = \mathbf{0}_{N\times C}, \lambda_0$.
for $t = 1, \dots, T$ do
    Update $\lambda_t$, $L_f$ and $\mathbf{\Sigma}^t$, $\bar{\mathbf{Y}}^t$ via (15).
    Initialize $\mathbf{A}[0] = \hat{\mathbf{A}}^{t-1}$, $\mathbf{B}[0] = \hat{\mathbf{B}}^{t-1}$, and set $k = 0$.
    while not converged do
        for $i = 1 \dots N$ (in parallel) do
            Compute $\mathbf{\Sigma}_{-i}^t$ and $\bar{\mathbf{Y}}_{-i}^t$.
            Form gradients at $\mathbf{a}_{-i}[k]$ and $b_{ii}[k]$ via (10)-(11).
            Update $\mathbf{a}_{-i}[k+1]$ via (12).
            Update $b_{ii}[k+1]$ via (13).
            Update $\mathbf{a}_i[k+1]$ via (14).
        end for
        $k = k + 1$.
    end while
    return $\hat{\mathbf{A}}^t = \mathbf{A}[k], \hat{\mathbf{B}}^t = \mathbf{B}[k]$.
end for
```

Algorithm 1 summarizes the steps outlined in this section for tracking the dynamic network topology, given temporal traces of infection events $\{\mathbf{Y}^t\}_{t=1}^T$ and susceptibilities $\mathbf{X}$. It is termed *pseudo real-time* ISTA, since in principle one needs to run multiple (inner) ISTA iterations till convergence per time interval $t = 1, \dots, T$. This will in turn incur an associated delay, that may (or may not) be tolerable depending on the specific network inference problem at hand. Nevertheless, numerical tests indicate that in practice 5-10 inner iterations suffice for convergence; see also the extended journal version of this paper [24].

4. NUMERICAL TESTS

The effectiveness of Algorithm 1 is corroborated in this section via simulations using real traces of information cascades collected from the web [19]. A comprehensive performance assessment using synthetically-generated network data can be found in [24], and is omitted here due to lack of space.

Real dataset description. The real data used was collected during a prior study by monitoring blog posts and news articles for memes (popular textual phrases) appearing within a set of over 3.3 million websites [23]. Traces of information cascades were recorded over a period of one year, from March 2011 till February 2012; the data is publicly available from [19]. The time when each website mentioned a specific news item was recorded as a Unix timestamp in hours (i.e., the number of hours since midnight on January 1, 1970). Specific globally-popular topics during this period were identified and cascade data for the top $5,000$ websites that mentioned memes associated with them were retained.

The real-data tests that follow focus on two keywords: i) "Kim Jong-un" the current leader of North Korea whose popularity rose after the death of his father and predecessor, during the observation period; and ii) "Reid Hoffman" the founder of the professional online social network *LinkedIn*, that went public during the observation period. Data was first pre-processed and filtered so that only (significant) cascades that propagated to at least 7 websites were retained. This reduced the dataset significantly to the 360 most relevant websites over which 466 cascades related to "Kim Jong-un"

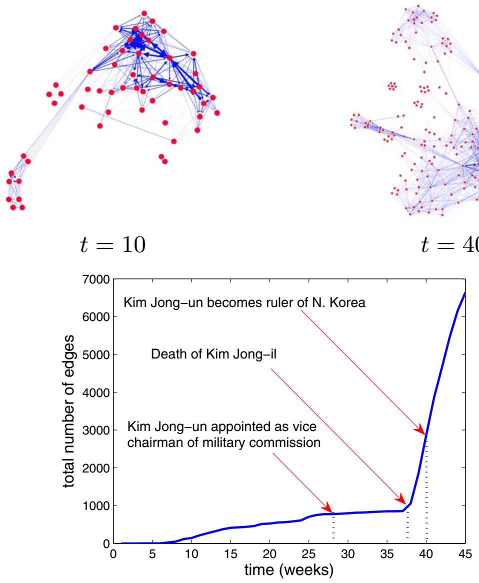

Fig. 1. (Top) Visualization of estimated networks from information cascades related to the topic "Kim Jong-un" at $t = 10$ and $t = 40$ weeks. (Bottom) Evolution of total number of inferred edges.

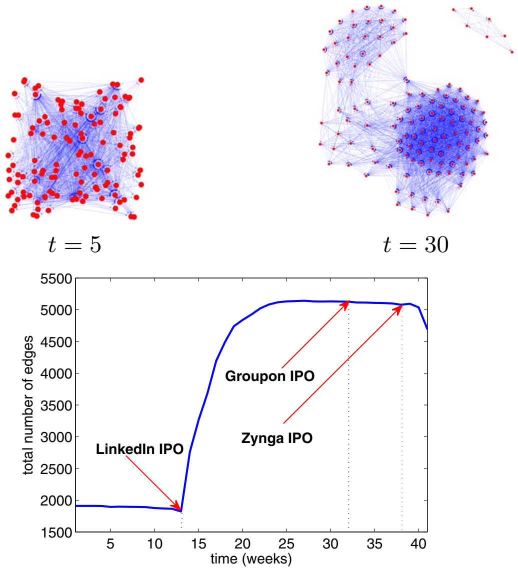

Fig. 2. (Top) Visualization of estimated networks obtained by tracking "Reid Hoffman" cascades at $t = 5$ and $t = 30$ weeks. (Bottom) Evolution of total number of inferred edges.

propagated during a 45 week period. Similarly, 125 websites were retained for propagation of 85 cascades related to "Reid Hoffman" over 41 weeks. Each time interval was set to one week and the observation time-scale was adjusted to start at the beginning of the earliest cascades.

In both cases, matrix $\mathbf{Y}^t$ was constructed by setting y^t_{ic} to the time when website i mentioned phrase c if this occurred during the span of week t. Otherwise y^t_{ic} was set to a large number, $100t_{\max}$, where $t_{\max}$ denotes the largest timestamp in the dataset. Typically the entries of matrix $\mathbf{X}$ capture prior knowledge about the susceptibility of each node to each contagion. For instance, the entry $\mathbf{x}_{ic}$ could denote the online search rank of website i for a search keyword associated with contagion c. In the absence of such real data, the entries of $\mathbf{X}$ were generated randomly from a uniform distribution over the interval $[0, 0.01]$.

Experimental results. Algorithm 1 was run on both datasets with $\beta = 0.9$ and $\lambda_t = 100$. Fig. 1 (top) depicts drawings of the inferred network for Kim Jong-un at $t = 10$ and $t = 40$ weeks. Speculation about the possible successor of the dying North Korean ruler, Kim Jong-il, rose until his death on December 17, 2011 (week 38). He was succeeded by Kim Jong-un on December 30, 2011 (week 40). The visualizations show an increasing number of edges over the 45 weeks, illustrating the growing interest of international news websites and blogs in the new ruler, about whom little was known in the first 10 weeks. Unfortunately, the observation horizon does not go beyond $T = 45$ weeks. A longer span of data would have been useful to investigate the rate at which global news coverage on the topic eventually subsided.

Fig. 1 (bottom) depicts the time evolution of the total number of edges in the inferred dynamic network. Of particular interest are the weeks during which: i) Kim Jong-un was appointed as the vice chairman of the North Korean military commission; ii) Kim Jong-il died; and iii) Kim Jong-un became the ruler of North Korea. These events were the topics of many online news articles and political blogs, an observation that is reinforced by the experimental results shown in the plot.

The results of running Algorithm 1 on the second dataset are shown in Fig. 2. Although Reid Hoffman was already popular in technology media coverage, his visibility in popular news and blogs increased tremendously following the highly successful initial public offering (IPO) of LinkedIn on May 19, 2011. Towards the end of 2011, a number of other successful technology companies like Groupon and Zynga went public, possibly stabilizing the amount of media coverage on Reid Hoffman. In fact the drop in the number of edges towards week 41 could be attributed to the captivation of media attention by the IPOs that occurred later in the year.

5. CONCLUDING SUMMARY

A dynamic SEM was proposed in this paper for network topology inference, using timestamp data for propagation of contagions typically observed in social networks. The model explicitly captures both topological influences and external sources of information diffusion over the unknown network. Exploiting the inherent edge sparsity typical of large networks, a computationally-efficient proximal gradient algorithm with well-appreciated convergence properties was developed to minimize a suitable sparsity-regularized exponentially-weighted LS estimator. A number of experiments conducted on real data demonstrated the effectiveness of the proposed algorithms in tracking dynamic and sparse networks.

The present work opens up multiple directions for exciting follow-up research. Future and ongoing research includes: i) investigating the conditions for identifiability of sparse and dynamic SEMs, as well as their statistical consistency properties tied to the selection of λ_t; ii) generalizing the SEM using kernels or suitable graph similarity measures to capture nonlinear dependencies and also enable network topology forecasting; and iii) exploiting the algorithm's parallel structure to devise MapReduce/Hadoop implementations scalable to million-node graphs.

6. REFERENCES

[1] E. M. Rogers, *Diffusion of Innovations*, Free Press, fourth edition, 1995.

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

[3] D. Kaplan, *Structural Equation Modeling: Foundations and Extensions*, Sage Publications, second edition, 2009.

[4] J. Pearl, *Causality: Models, Reasoning, and Inference*, Cambridge University Press, second edition, 2009.

[5] N. Meinshausen and P. Buhlmann, "High-dimensional graphs and variable selection with the lasso," *Ann. Statist.*, vol. 34, pp. 1436–1462, 2006.

[6] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, pp. 432–441, Dec. 2007.

[7] M. Kolar, L. Song, A. Ahmed, and E. P. Xing, "Estimating time-varying networks," *Ann. Appl. Statist.*, vol. 4, pp. 94–123, 2010.

[8] B. Muthén, "A general structural equation model with dichotomous, ordered categorical, and continuous latent variable indicators," *Pyschometrika*, vol. 49, pp. 115–132, Mar. 1984.

[9] A. S. Goldberger, "Structural equation methods in the social sciences," *Econometrica*, vol. 40, pp. 979–1001, Nov. 1972.

[10] X. Cai, J. A. Bazerque, and G. B. Giannakis, "Gene network inference via sparse structural equation modeling with genetic perturbations," *PLoS Comp. Biology*, vol. 9, May 2013, e1003068 doi:10.1371/journal.pcbi.1003068.

[11] B. A. Logsdon and J. Mezey, "Gene expression network reconstruction by convex feature selection when incorporating genetic perturbations," *PLoS Comp. Biology*, vol. 6, Dec. 2010, e1001014. doi:10.1371/journal.pcbi.1001014.

[12] J. A. Bazerque, B. Baingana, and G. B. Giannakis, "Identifiability of sparse structural equation models for directed and cyclic networks," in *Proc. of Global Conf. on Signal and Info. Processing*, Austin, TX, Dec. 2013.

[13] Y. Chen, Y. Gu, and A. O. Hero III, "Sparse LMS for system identification," in *Proc. of Intern. Conf. on Acoustics, Speech and Signal Processing*, Taipei, Taiwan, Apr. 2009.

[14] Y. Kopsinis, K. Slavakis, and S. Theodoridis, "Online sparse system identification and signal reconstruction using projections onto weighted ℓ_1 balls," *IEEE Trans. Signal Process.*, vol. 59, pp. 936–952, Mar. 2011.

[15] D. Angelosante, J. A. Bazerque, and G. B. Giannakis, "Online adaptive estimation of sparse signals: where RLS meets the ℓ_1-norm," *IEEE Trans. Signal Process.*, vol. 58, pp. 3436–3447, July 2010.

[16] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends Optimization*, vol. 1, pp. 123–231, 2013.

[17] I. Daubechies, M. Defrise, and C. D. Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Comm. Pure Appl. Math.*, vol. 57, pp. 1413–1457, Aug. 2004.

[18] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, pp. 183–202, Jan. 2009.

[19] Jure Leskovec, "Web and blog datasets," *Stanford Network Analysis Project*, 2011.

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

[21] M. G. Rodriguez, D. Balduzzi, and B. Scholkopf, "Uncovering the temporal dynamics of diffusion networks," in *Proc. of 28th Intern. Conf. Machine Learning*, Bellevue, WA, July 2011.

[22] S. Meyers and J. Leskovec, "On the convexity of latent social network inference," in *Proc. of Neural Information Proc. Sys. Conf.*, Vancouver, Canada, Feb. 2013.

[23] M. G. Rodriguez, J. Leskovec, and B. Scholkopf, "Structure and dynamics of information pathways in online media," in *Proc. of 6th ACM Intern. Conf. on Web Search and Data Mining*, Rome, Italy, Dec. 2010.

[24] B. Baingana, G. Mateos, and G. B. Giannakis, "Dynamic structural equation models for social network topology inference," *IEEE J. Selected Topics in Signal Process.*, Aug. 2013, (submitted; see also arXiv:1309.6683v2 [cs.SI]).

[25] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Springer, second edition, 2009.