[Gonzalo Mateos and Ketan Rajawat]

Dynamic Network Cartography

[Advances in network health monitoring]

Communication networks have evolved from specialized research and tactical transmission systems to large-scale and highly complex interconnections of intelligent devices, increasingly becoming more commercial, consumer oriented, and heterogeneous. Propelled by emergent social networking services and high-definition streaming platforms, network traffic has grown explosively thanks to the advances in processing speed and storage capacity of state-of-the-art communication technologies. As "netizens" demand a seamless networking experience that entails not only higher speeds but also resilience and robustness to failures and malicious cyberattacks, ample opportunities for signal processing (SP) research arise. The vision is for ubiquitous smart network devices to enable data-driven statistical learning algorithms for distributed, robust, and online network operation and management, adaptable to the dynamically evolving network landscape with minimal need for human intervention. This article aims to delineate the analytical background and the relevance of SP tools to dynamic network monitoring, introducing the SP readership to the concept of dynamic network cartography—a framework to construct maps of the dynamic network state in an efficient and scalable manner tailored to large-scale heterogeneous networks.

Digital Object Identifier 10.1109/MSP.2012.2232355
Date of publication: 5 April 2013

INTRODUCTION

The emergence of multimedia-enriched social networking services and Internet-friendly portable devices is multiplying network traffic volume day by day [53]. Wireless connectivity under the envisioned dynamic spectrum paradigm [30] relies on mobile networks of diverse nodes, which are nevertheless united by unparalleled cognition capabilities, adaptability, and decision-making attributes. Moreover, the advent of networks of intelligent devices such as those deployed to monitor the smart power grid, transportation networks, medical information networks, and cognitive radio (CR) networks, will transform the communication infrastructure to an even more complex and

heterogeneous one. Thus, ensuring compliance to service-level agreements and quality-of-service (QoS) guarantees necessitates breakthrough management and monitoring tools providing operators with a comprehensive view of the network landscape. Situational awareness provided by such tools will be the key enabler for effective information dissemination, routing and congestion control, network health management, risk analysis, and security assurance.

But this great promise comes with great challenges. Acquiring network-wide performance and utilization metrics for large networks is no easy task. Suppose, for instance, that traffic volumes are of interest, not only for gauging instantaneous network health but also for more complex network management tasks such as intrusion detection, capacity provisioning, and network planning [56]. While traffic volumes on links (also called link counts) are readily acquired using off-the-shelf tools such as the simple network management protocol (SNMP), missing link-count measurements may still skew the network operator's perspective. SNMP packets may be dropped, for instance, if some links become congested, rendering link-count information for those links more important as well as less available [48], [50]. Classical approaches relying either on simple time-series interpolation or on regularized least-squares (LS) formulations for predicting the missing link counts [51] have not been able to fully capture the complexity of the Internet traffic. This is evidenced by the recent upsurge of efforts toward advanced network tomography [14] and spatiotemporal traffic estimation algorithms for network monitoring [27], [50], [56].

SITUATIONAL AWARENESS PROVIDED BY SUCH TOOLS WILL BE THE KEY ENABLER FOR EFFECTIVE INFORMATION DISSEMINATION, ROUTING AND CONGESTION CONTROL, NETWORK HEALTH MANAGEMENT, RISK ANALYSIS, AND SECURITY ASSURANCE.

Similarly, path metrics such as end-to-end delays are of great interest to service providers because they directly affect the end-user experience. The challenge here is that the number of paths grows very fast as the number of nodes increases. Probing exhaustively all origin-destination (OD) pairs is impractical and wasteful of resources even for moderate-size networks [18], [49]. Accurate prediction of missing delays based on the inherent, e.g., topology-induced correlation or smoothness traits among link and path quantities is therefore crucial for statistical analysis and monitoring tasks [33]. While the prevailing operational paradigm adopted in current networks entails nodes continuously communicating their link measurements to a central monitoring station, in-network distributed cooperation through local interactions is preferred for scalability and robustness considerations [39].

Conventional network monitoring tools entail a couple of additional limitations. First, they are typically resource heavy and tend to overload network operators with crude, unrefined data, without enough processing to separate the "data wheat from the chaff"; see, e.g., [20] and references therein. It is thus of paramount importance to construct parsimonious descriptors of the network state, for the purpose of modeling, monitoring, and management of complex interconnected systems. Due to the diversity of modern networks, the network state can incorporate typical quantities such as traffic volumes and end-to-end delays, as well as latent social metrics such as hierarchy, reputation, and vulnerability. Second, malicious activities intended to undermine network functionality or compromise secrecy of data have grown in sophistication, thus rendering traditional signature-based intrusion detection schemes increasingly obsolete. Intrusion attempts and malicious attacks manifest themselves as abrupt changes in network states [6], and such anomalous patterns are oftentimes hidden within the raw high-dimensional network data [55]. For these reasons, unveiling network anomalies in a reliable and computationally efficient manner is a challenging yet essential goal [34], [39], [55].

All in all, accurate network diagnosis and statistical analysis tools are instrumental for maintaining seamless end-user experience in dynamic environments as well as for ensuring network security and stability. In this direction, this tutorial advocates the concept of dynamic network cartography as a tool for statistical modeling, monitoring, and management of complex networks. Focus will be placed on two complementary aspects of network cartography, specifically, online construction of global network state maps using only a few measurements and the unveiling of network anomalies across network flows and time. The surveyed cartography algorithms leverage recent advances in machine learning and statistical SP methods, including sparsity-cognizant learning, kriged Kalman filtering of dynamical processes over networks, nuclear norm minimization for low-rank matrix completion, semisupervised dictionary learning (DL), and in-network optimization via the alternating-directions method of multipliers. Through a unifying treatment that revolves around network cartography, this article demonstrates how benefits from foundational SP methods can permeate to dynamic network monitoring and collectively enable inference of global network health, thus leading to enhanced network robustness and QoS.

GLOBAL PERFORMANCE PREDICTION VIA DYNAMICAL NETWORK CARTOGRAPHY

This section deals with the problem of mapping the network state from incomplete sets of measurements and touches upon two application domains. A DL algorithm is introduced first to efficiently impute missing link traffic volumes, using measurements from a wide class of (possibly nonstationary) traffic patterns [27]. Subsequently, the problem of tracking and predicting end-to-end network delay is considered, and the dynamic network kriging approach of [46] is described.

SEMISUPERVISED DICTIONARY LEARNING FOR TRAFFIC MAPS

Consider an Internet protocol (IP) network comprising N nodes and L links, carrying the traffic of F OD flows (network connections). Let $x_{l,t}$ denote the traffic volume (in bytes or packets) passing through link $l \in \{1, \ldots, L\}$ over a fixed interval of time $(t, t + \Delta t)$. Link counts across the entire network are collected in the vector $\mathbf{x}_t \in \mathbb{R}^L$, e.g., using the ubiquitous SNMP protocol. Since measured link counts are both unreliable and incomplete due to hardware or software malfunctioning, jitter, and communication errors [56], [48], they are expressed as noisy versions of a subset of $S < L$ links

$$\mathbf{y}_t = \mathbf{S}_t \mathbf{x}_t + \epsilon_t, \quad t = 1, 2, \ldots \tag{1}$$

where $\mathbf{S}_t$ is an $S \times L$ selection matrix with 0–1 entries whose rows correspond to rows of the identity matrix of size L, and ϵ_t is an $S \times 1$ zero-mean noise term with constant variance accounting for measurement and synchronization errors. Given $\mathbf{y}_t$ the aim is to form an estimate $\hat{\mathbf{x}}_t$ of the full vector of link counts $\mathbf{x}_t$, which in this case defines the network state.

A simple approach implemented in measurement-processing software, such as RRDtool [44], is to ignore the noise term and rely on one-dimensional interpolation for the time series $\{x_{l,t}\}$ per link l. The applicability and accuracy of this scheme is, however, limited since it tacitly assumes that the entries of $\mathbf{x}_t$ are uncorrelated; missing entries $x_{l,t}$ are few and do not occur in bursts; and the time series $\{\mathbf{x}_t\}$ is stationary. Nevertheless, none of these assumptions holds true in real networks [48].

The reliance on stationarity and availability of measurements from contiguous time intervals can be foregone if estimation of $\mathbf{x}_t$ is performed for each t individually. In principle, $\hat{\mathbf{x}}_t$ can be obtained if the volumes of OD traffic flows $\mathbf{z}_t \in \mathbb{R}^F$ are available, since they are related through

$$\mathbf{x}_t = \mathbf{R}\mathbf{z}_t, \tag{2}$$

where the so-termed routing matrix $\mathbf{R} := [r_{l,f}] \in \{0, 1\}^{L \times F}$ is such that $r_{l,f} = 1$ if link l carries the flow f, and zero otherwise. However, measuring $\mathbf{z}_t$ is even more difficult and in practice $\mathbf{z}_t$ is itself estimated from $\{\mathbf{x}_t\}$ through tomographic traffic inference [14], [33], where given $\mathbf{R}$ and noisy link counts, the goal is to estimate the OD flows as the solution of a linear inverse problem. Since the inverse problem is highly under-determined $[F = \mathcal{O}(N^2) \gg L = \mathcal{O}(N)]$, early approaches relied on prior knowledge in the form of statistical models for the OD flows (such as the Poisson, Gaussian, logit-choice, or gravity models), that ultimately serve as complexity-controlling (that is regularization) mechanisms [33, Ch. 9]. Among these, the state-of-the-art traffic matrix estimation algorithm uses an entropy-based regularizer and has been shown to be fast, accurate, robust, and flexible [54]. Time-series analysis-based approaches (such as the Kalman filter in [51]) have also been proposed for scenarios where link-count measurements are available over contiguous time slots.

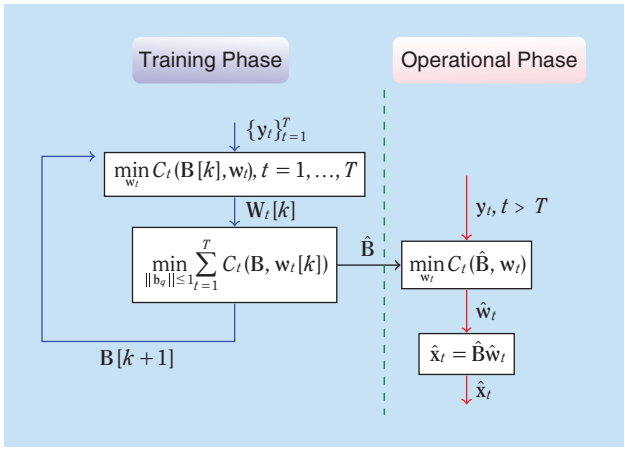
Recently, a link-count prediction algorithm was put forth in [27], where missing entries of $\mathbf{x}_t$ are estimated from historical measurements in $\mathcal{T}_S := \{\mathbf{y}_t\}_{t=1}^T$ by leveraging the structural regularity of $\mathbf{R}$ through a semisupervised DL approach. Under the DL framework, data-driven dictionaries for sparse signal representation are adopted as a versatile means of capturing parsimonious signal structures; see, e.g., [52] for a tutorial treatment. Propelled by the success of compressive sampling (CS) [24], sparse signal modeling has led to major advances in several machine learning, audio, and image processing tasks [52], [28]. Motivated by these ideas, it is postulated in [27] that link counts can be represented as a linear combination $\mathbf{x}_t = \mathbf{B}\mathbf{w}_t$ of a few ($\ll Q$) columns of an overcomplete dictionary (basis) matrix $\mathbf{B} := [\mathbf{b}_1, \ldots, \mathbf{b}_Q] \in \mathbb{R}^{L \times Q}$, where $\mathbf{w}_t \in \mathbb{R}^Q$ is a sparse vector of expansion coefficients. Many signals including speech and natural images admit sparse representations even under generic predefined dictionaries, such as those based on the Fourier and the wavelet bases, respectively [52]. Like audio and natural images, link counts can exhibit strong correlations as evidenced from the structure of $\mathbf{R}$ [cf. (2)]. For instance, the traffic volumes on links i and j are highly correlated if they both carry common flows. DL schemes are attractive due to their flexibility, since they utilize training data to *learn* an appropriate overcomplete basis customized for the data at hand. However, the use of DL for modeling network data is well motivated but so far relatively unexplored.

ACCURATE NETWORK DIAGNOSIS AND STATISTICAL ANALYSIS TOOLS ARE INSTRUMENTAL FOR MAINTAINING SEAMLESS END-USER EXPERIENCE IN DYNAMIC ENVIRONMENTS AS WELL AS FOR ENSURING NETWORK SECURITY AND STABILITY.

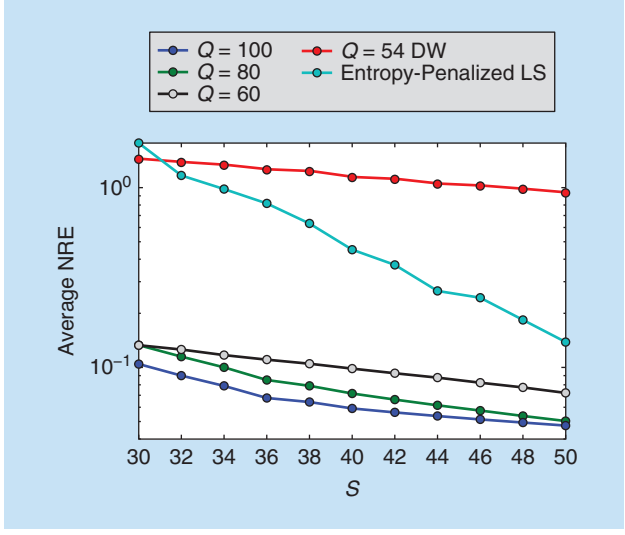
PREDICTION OF LINK COUNTS

Suppose for now that either a learned, or, a suitable prespecified dictionary $\mathbf{B}$ is available and consider predicting the missing link counts. Data-driven learning of dictionaries from historical data will be addressed in the ensuing subsection. Given $\mathbf{R}$ and the link count measurements $\mathbf{y}_t$, contemporary tools developed in the area of CS and semisupervised learning can be used to form $\hat{\mathbf{x}}_t$, which includes estimates for the missing $L - S$ link counts [9], [28], [24]. The spatial regularity of the link counts is captured through the auxiliary weighted graph $\mathcal{G}$ with L vertices, one for each link in the network. The edge weights for all edges in $\mathcal{G}$ are subsumed by the off-diagonal entries of the Gram matrix $\mathbf{G} = [g_{i,j}] := \mathbf{R}\mathbf{R}' \in \mathbb{R}^{L \times L}$, where $(\cdot)'$ denotes transposition. The off-diagonal entries $g_{i,j}$ count the number of OD flows that are common to both links i and j. Main diagonal entries of $\mathbf{G}$ count the number of OD flows that use the corresponding links.

Given a snapshot of incomplete link counts $\mathbf{y}_t$ during the operational phase (where a suitable basis $\mathbf{B}$ is



[FIG1] Training and operational phases of the semisupervised DL approach for link-traffic cartography in [27], where $C_t(B,w)$ denotes the tth summand from the cost in (4) and $k = 1, 2, \ldots$ indicate iterations of the BCD solver.



[FIG2] Link-traffic cartography of Internet-2 data [1]. Comparison of NRE for different values of S [27]. (Figure used with permission from [27].)

available), the sparse basis expansion coefficient vector $\mathbf{w}_t$ is estimated as

$$\hat{\mathbf{w}}_t := \arg\min_{\mathbf{w}_t} \|\mathbf{y}_t - \mathbf{S}_t\mathbf{B}\mathbf{w}_t\|_2^2 + \lambda_w\|\mathbf{w}_t\|_1 + \lambda_g\mathbf{w}_t'\mathbf{B}'\mathbf{L}\mathbf{B}\mathbf{w}_t, \quad (3)$$

where $\mathbf{L} := \mathrm{diag}(\mathbf{G}\mathbf{1}_L) - \mathbf{G}$ denotes the Laplacian matrix of $\mathcal{G}$; $\lambda_w, \lambda_g > 0$ are tunable regularization parameters; and $\mathbf{1}_L$ is the $L \times 1$ vector of all ones. The criterion in (3) consists of an LS error between the observed and postulated link counts, along with two regularizers. The ℓ_1-norm $\|\mathbf{w}_t\|_1$ encourages sparsity in the coefficient vector $\hat{\mathbf{w}}_t$ [24], [28]. With $\mathbf{x}_t := [x_{1,t}, \ldots, x_{L,t}]'$ given by $\mathbf{x}_t = \mathbf{B}\mathbf{w}_t$, the Laplacian regularization can be explicitly written as $\mathbf{w}_t'\mathbf{B}'\mathbf{L}\mathbf{B}\mathbf{w}_t = (1/2)\sum_{i=1}^L \sum_{j=1}^L g_{i,j}(x_{i,t} - x_{j,t})^2$. It is thus apparent that $\mathbf{w}_t'\mathbf{B}'\mathbf{L}\mathbf{B}\mathbf{w}_t$ encourages the link counts to be close if their corresponding vertices are connected in $\mathcal{G}$. Each summand is weighted according to the number of OD flows common to links i and j. Typically adopted for semisupervised learning, such a regularization term encourages $\mathbf{B}\mathbf{w}_t$ to lie on a smooth manifold approximated by $\mathcal{G}$, which constrains how the measured link counts relate to $\mathbf{x}_t$ [9], [45]. It is also common to use normalized variants of the Laplacian instead of $\mathbf{L}$ [33, p. 46].

The cost in (3) is convex but nonsmooth, and customized solvers developed for ℓ_1-norm regularized optimization can be employed here as well, e.g., [28, p. 92]. Once $\hat{\mathbf{w}}_t$ is available, an estimate of the full vector of link counts is readily obtained as $\hat{\mathbf{x}}_t := \mathbf{B}\hat{\mathbf{w}}_t$. It is apparent that the quality of the imputation depends on the chosen $\mathbf{B}$, and DL from historical network data in $\mathcal{T}_S$ is described next.

DATA-DRIVEN DL

In its canonical form, DL seeks a (typically fat) dictionary $\mathbf{B}$ so that training data $\mathcal{T}_L := \{\mathbf{x}_t\}_{t=1}^T$ are well approximated as $\mathbf{x}_t \approx \mathbf{B}\mathbf{w}_t$, $t = 1, \ldots, T$, for some sparse vectors $\mathbf{w}_t$ of expansion coefficients [52]. Standard DL algorithms cannot, however, be directly applied to learn $\mathbf{B}$ since they rely on the entire vector $\mathbf{x}_t$. To learn the dictionary in the training phase using incomplete link counts $\mathcal{T}_S$ instead of $\mathcal{T}_L$, the idea is to capitalize on the structure in $\mathbf{x}_t$, of which $\mathcal{G}$ is an abstraction [27]. To this end, one can adopt a similar cost function as in the operational phase [cf. (3)], yielding the data-driven basis and the corresponding sparse representation

$$\{\hat{\mathbf{W}}, \hat{\mathbf{B}}\} := \arg\min_{\mathbf{W}, \mathbf{B}:\{\|\mathbf{b}_q\|_2 \le 1\}_{q=1}^Q} \sum_{t=1}^T [\|\mathbf{y}_t - \mathbf{S}_t\mathbf{B}\mathbf{w}_t\|_2^2 + \lambda_w\|\mathbf{w}_t\|_1 + \lambda_g\mathbf{w}_t'\mathbf{B}'\mathbf{L}\mathbf{B}\mathbf{w}_t], \quad (4)$$

where $\hat{\mathbf{W}} := [\hat{\mathbf{w}}_1, \ldots, \hat{\mathbf{w}}_T] \in \mathbb{R}^{Q \times T}$. The constraints $\{\|\mathbf{b}_q\|_2 \le 1\}_{q=1}^Q$ remove the scaling ambiguity in the products $\mathbf{B}\mathbf{w}_t$ and prevent the entries in $\mathbf{B}$ from growing unbounded. Again, the combined regularization terms in (4) promote both sparsity in $\mathbf{w}_t$ through the ℓ_1-norm, and smoothness across the entries of $\mathbf{B}\mathbf{w}_t$ via the Laplacian $\mathbf{L}$. The regularization parameters λ_w and λ_g are typically cross-validated [28, Ch. 7]. Although (4) is nonconvex, a block coordinate-descent (BCD) solver still guarantees convergence to a stationary point [10]. The BCD updates involve solving for $\mathbf{B}$ and $\mathbf{W}$ in an alternating fashion, both doable efficiently via convex programming [27]. Alternatively, the online DL algorithm in [37] offers enhanced scalability by sequentially processing the data in $\mathcal{T}_S$. The training and operational (prediction) phases are summarized in Figure 1, where $C_t(\mathbf{B}, \mathbf{w})$ denotes the tth summand from the cost in (4), and $k = 1, 2, \ldots$ indicate iterations of the BCD solver employed during the training phase.

The explicit need for Laplacian regularization is apparent from (4). Indeed, if measurements from a certain link are not present in $\mathcal{T}_S$, the corresponding row of $\mathbf{B}$ may still be estimated with reasonable accuracy because of the third term in $C_t(\mathbf{B}, \mathbf{w})$. On top of that, it is because of Laplacian regularization that the prediction performance degrades gracefully as the number of

missing entries in $\mathbf{y}_t$ increases; see also Figure 2. It is worth stressing that the time series $\{\mathbf{y}_t\}$ need not be stationary or even contiguous in time. The link-traffic cartography approach described so far can also be adapted to accommodate time-varying network topologies or routing matrices, using a time-dependent Laplacian $\mathbf{L}_t$. A word of caution is due, however, since drastic changes in either $\mathbf{L}_t$ or in the statistical properties of the underlying OD flows $\mathbf{z}_t$, will necessitate retraining $\mathbf{B}$ to attain satisfactory performance. Finally, note that DL techniques incur a complexity at least cubic in the size of the network and are better suited for monitoring of backbone wide-area networks which are typically not very large.

A PROMISING APPROACH IN THIS CONTEXT HAS BEEN THE APPLICATION OF KRIGING, A TOOL FOR SPATIAL PREDICTION POPULAR IN GEOSTATISTICS AND ENVIRONMENTAL SCIENCES.

Next, a numerical test on link count data from the Internet-2 measurement archive [1] is outlined. The data consists of link counts, sampled at five-minute intervals, collected over several weeks. For the purposes of comparison, the training phase consisted of 2,000 time slots, with a random subset of 50 links measured (out of $L = 54$ per time slot). The performance of the learned dictionary is then assessed over the next $T_0 = 2{,}000$ time slots. Each test vector $\mathbf{y}_t$ is constructed by randomly selecting S entries of the full link count vector $\mathbf{x}_t$. The tuning parameters are chosen via cross-validation ($\lambda_s = 0.1$ and $\lambda_g = 10^{-5}$). Figure 2 shows the normalized reconstruction error (NRE), evaluated as $(LT_0)^{-1}\sum_{t=1}^{T_0}\|\mathbf{y}_t - \hat{\mathbf{x}}_t\|^2$ for different values of Q and S. For comparison, the prediction performance with a fixed diffusion wavelet matrix [19] (instead of the data-trained dictionary), as well as that of the entropy-penalized LS method [54] is also shown. The latter approach solves a LS problem augmented with a specific entropy-based regularizer that encourages the traffic volumes at the source/destination pairs to be stochastically independent. The DL-based method markedly outperforms the competing approaches, especially for low values of S. Furthermore, note how performance degrades gracefully as S decreases. Remarkably, the predictions are close to the actual traffic even when using only 30 link counts during the prediction phase.

DELAY CARTOGRAPHY VIA DYNAMIC NETWORK KRIGING

Instead of link counts, consider now the problem of monitoring delays $d_{p,t}$ on a set of multihop paths $p \in \mathcal{P}$, that connect $P := |\mathcal{P}|$ source-destination pairs in an IP network. Path delays are important metrics required by network operators for assessment, planning, and fault diagnosis [18], [33], [46]. However, monitoring path metrics is challenging primarily because P generally grows as the square of the number of nodes in the network. Therefore, at any time t delays can only be measured on a subset of paths $\mathcal{S}_t \subset \mathcal{P}$, collected in the vector $\mathbf{d}_t^s$. Based on the partial current and past measurements $\mathcal{H}_t := \{\mathbf{d}_\tau^s\}_{\tau=1}^t$, delay cartography amounts to predicting the remaining path delays $\mathbf{d}_t^{\bar{s}} := \{d_{p,t}\}_{p\in\mathcal{P}\setminus\mathcal{S}}$.

A promising approach in this context has been the application of kriging, a tool for spatial prediction popular in geostatistics and environmental sciences [22]. A network kriging scheme was developed in [18], which advocates prediction of network-wide path delays using measurements on a fixed subset of paths. The class of linear predictors introduced therein leverages network topology information to model the covariance among path delays. Building on these ideas, a dynamic network kriging approach capable of real-time spatiotemporal delay predictions was put forth in [46]. Specifically, a kriged Kalman filter (KKF) is employed to explicitly capture temporal variations due to queuing delays, while retaining the topology-based spatial kriging predictor. The per-path delay $d_{p,t}$ comprises several independent components due to contributions from each intermediate link and router and is modeled in [46] as

$$d_{p,t} = \chi_{p,t} + \nu_{p,t} + \varepsilon_{p,t}. \tag{5}$$

The queuing delay $\chi_{p,t}$ (collected in $\boldsymbol{\chi}_t \in \mathbb{R}^P$) depends on the traffic and exhibits spatiotemporal correlation, periodic behavior as well as occasional bursts, prompting the following random walk model

$$\boldsymbol{\chi}_t = \boldsymbol{\chi}_{t-1} + \boldsymbol{\eta}_t, \tag{6}$$

where the driving noise $\boldsymbol{\eta}_t$ has zero mean and covariance matrix $\mathbf{C}_\eta$. The second term in (5), collected in the vector $\boldsymbol{\nu}_t$, combines the processing, transmission, and propagation delays and is temporally white but spatially correlated, owing to the overlap between paths. Similar to [18], the correlation between two paths is modeled as being proportional to the number of links they share, so that the covariance matrix $\mathbf{C}_\nu = \alpha\mathbf{U}\mathbf{U}'$, where α is a parameter to be estimated from training path-delay data; while $u_{p,l} = 1$ if path p contains link l, and $u_{p,l} = 0$ otherwise. Finally, the noise term $\epsilon_{p,t}$ is zero mean independent and identically distributed (i.i.d.) with known variance σ^2. Defining the $S \times P$ path selection matrix as in the section "Semisupervised Dictionary Learning for Traffic Maps," the measurement equation can be written as (introduce $\boldsymbol{\nu}_t^s := \mathbf{S}_t\boldsymbol{\nu}_t$ and likewise $\boldsymbol{\epsilon}_t^s$)

$$\mathbf{d}_t^s = \mathbf{S}_t\boldsymbol{\chi}_t + \boldsymbol{\nu}_t^s + \boldsymbol{\epsilon}_t^s. \tag{7}$$

In the absence of $\mathbf{S}_t$, the spatiotemporal model in (6) and (7) is widely employed in geostatistics, where $\boldsymbol{\chi}_t$ is generally referred to as trend, and $\boldsymbol{\nu}_t$ captures the random fluctuations around $\boldsymbol{\chi}_t$; see, e.g. [41]. Similar models have been employed in [31] to describe the dynamics of wireless propagation channels, and in [21] for spatiotemporal random field estimation. For a static selection matrix, i.e., $\mathbf{S}_t := \mathbf{S}$ for all t, the network kriging approach [18] entails the following two-step procedure: Step 1) treat $\boldsymbol{\nu}_t^s$ as noise, and estimate $\boldsymbol{\chi}_t$ using the generalized LS criterion; and Step 2) use the aforesaid estimate to find the linear

minimum mean-square error (LMMSE) estimator (denoted by $\mathbb{E}^*$) for $\boldsymbol{\nu}_t^s$, specifically

$$\mathbb{E}^*[\boldsymbol{\nu}_t^s \,|\, \boldsymbol{\chi}_t] = \mathbf{S}\mathbf{C}_\nu \mathbf{S}'(\mathbf{S}\mathbf{C}_\nu \mathbf{S}' + \sigma^2 \mathbf{I}_S)^{-1}[\mathbf{d}_t^s - \mathbf{S}_t \boldsymbol{\chi}_t]. \tag{8}$$

Recently, a CS-based approach has also been reported for predicting network-wide performance metrics [19]. For instance, diffusion wavelets were utilized in [19] to obtain a compressible representation of the delays and account for spatial and temporal correlations. Although this allows for enhanced prediction accuracy relative to [18], it requires batch processing of measurements, which does not scale well to large networks for real-time operation. Pictorially, the performance of different algorithms can be assessed through the delay maps shown in Figure 3.

The spatiotemporal model set forth earlier can provide a better estimate of $\boldsymbol{\chi}_t$ by efficiently processing both present and past measurements jointly. Towards this end, a Kalman filter is employed in [46], which at time t yields the following update equations:

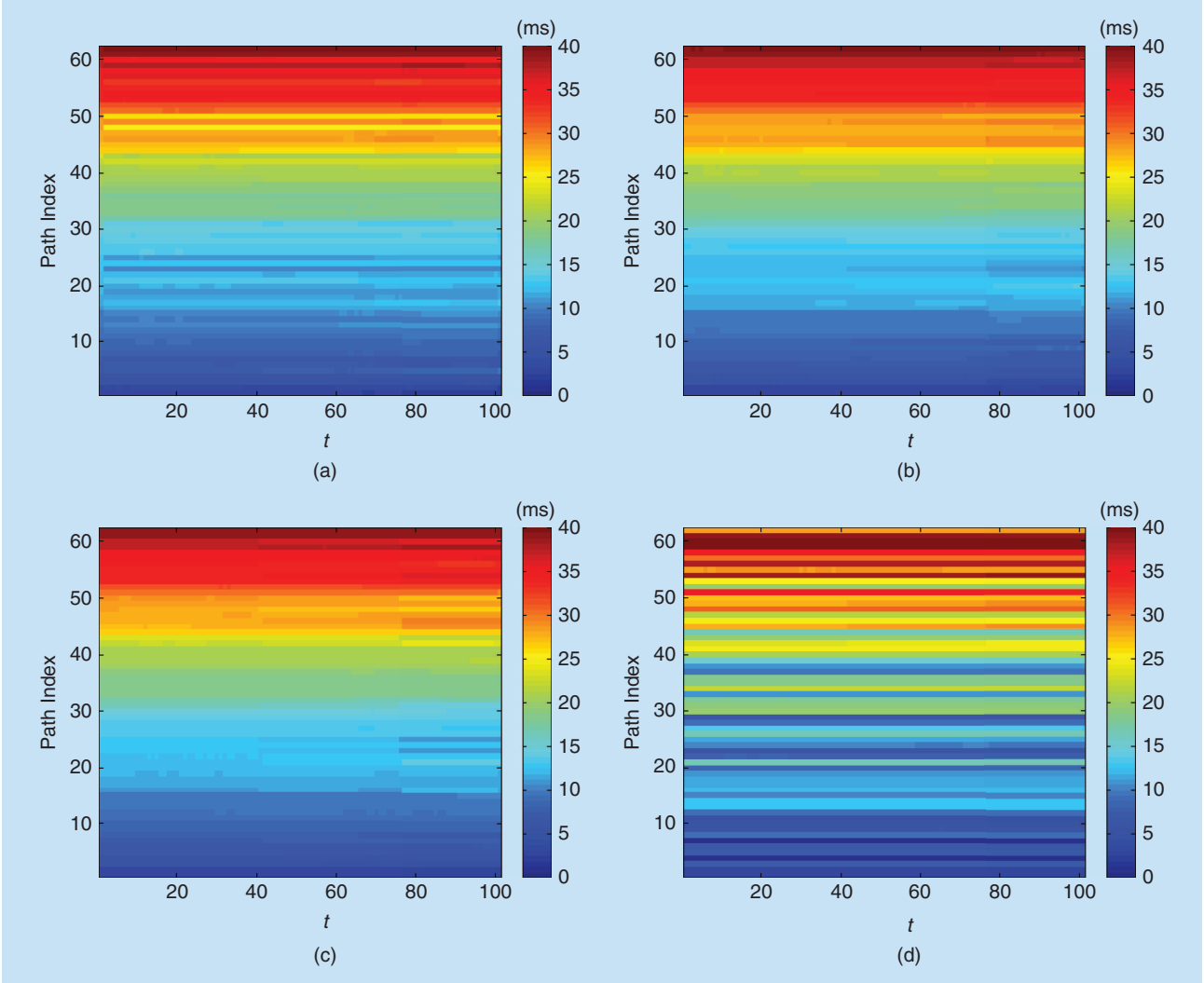
$$\hat{\boldsymbol{\chi}}_t := \mathbb{E}^*[\boldsymbol{\chi}_t | \mathcal{H}_t] = \hat{\boldsymbol{\chi}}_{t-1} + \mathbf{K}_t(\mathbf{d}_t^s - \mathbf{S}_t \hat{\boldsymbol{\chi}}_{t-1})$$
$$\mathbf{M}_t := \mathbb{E}[(\boldsymbol{\chi}_t - \hat{\boldsymbol{\chi}}_t)(\boldsymbol{\chi}_t - \hat{\boldsymbol{\chi}}_t)'] = (\mathbf{I}_P - \mathbf{K}_t \mathbf{S}_t)(\mathbf{M}_{t-1} + \mathbf{C}_\nu),$$

where $\mathbf{K}_t := (\mathbf{M}_{t-1} + \mathbf{C}_\nu)\mathbf{S}_t'[\mathbf{S}_t(\mathbf{C}_\nu + \mathbf{C}_\eta + \mathbf{M}_{t-1})\mathbf{S}_t' + \sigma^2 \mathbf{I}_S]^{-1}$ is the so-termed Kalman gain. The final predictor, referred also as the KKF, is given by

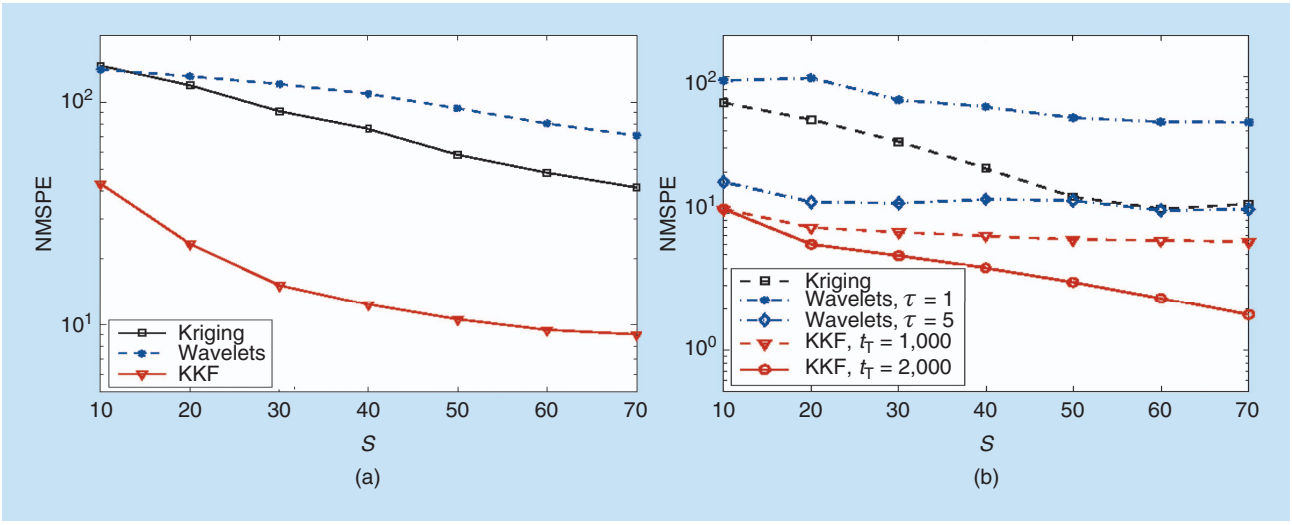
$$\hat{\mathbf{d}}_t^{\bar{s}} := \bar{\mathbf{S}}_t \hat{\boldsymbol{\chi}}_t + \bar{\mathbf{S}}_t \mathbf{C}_\nu \mathbf{S}_t'(\mathbf{S}_t \mathbf{C}_\nu \mathbf{S}_t' + \sigma^2 \mathbf{I}_S)^{-1}[\mathbf{d}_t^s - \mathbf{S}_t \hat{\boldsymbol{\chi}}_t]$$

and the prediction error covariance matrix is

$$\mathbf{M}_t^{\bar{s}} := \mathbb{E}[(\mathbf{d}_t^{\bar{s}} - \hat{\mathbf{d}}_t^{\bar{s}})(\mathbf{d}_t^{\bar{s}} - \hat{\mathbf{d}}_t^{\bar{s}})']$$
$$= \sigma^2 \mathbf{I}_S + \bar{\mathbf{S}}_t \left[(\mathbf{M}_{t-1} + \mathbf{C}_\nu + \mathbf{C}_\eta)^{-1} + \frac{1}{\sigma^2}\mathbf{S}_t'\mathbf{S}_t\right]^{-1} \bar{\mathbf{S}}_t'.$$



[FIG3] True and predicted delay map for 62 paths in the Internet-2 data set [1] over an interval of 100 min. (a) True delays. (b) Network kriging [18]. (c) Difussion wavelets [19]. (d) KKF [46]. Delays of several paths change slightly around $t = 80$, but this change is only discernible from the delay predictions offered by KKF. Delay maps summarize the network state and are useful tools aiding operational decision in network monitoring and control stations [46]. (Figure used with permission from [46].)



[FIG4] Delay cartography using the NZ-AMP data set [2], which includes path delays collected over a month for an IP network where $P = 186$ and $N = 30$ [46]. Normalized mean-square prediction error (NMSPE) as a function of S. (a) Random path selection. (b) "Optimal" path selection, that is, using heuristic or approximate algorithms specified for each algorithm. Observe further that the performance of the KKF improves as the length of the training interval t_T increases. (Figure used with permission from [46].)

The KKF framework for dynamic network delay cartography has several attractive features. First, the KKF yields the LMMSE estimate even for non-Gaussian distributed noise. The Kalman filter step also allows for a τ-step prediction given by $\hat{\mathbf{d}}_{t+\tau} = \hat{\boldsymbol{\chi}}_t$, which can be useful for preemptive routing and congestion control algorithms, as well as for extrapolating missing measurements. Second, the KKF framework provides a metric, particularly the error covariance matrix $\mathbf{M}_t^s$, for choosing the paths to be measured at each t, which define the selection matrix $\mathbf{S}_t$. In the present setting, it turns out that the D-optimal design metric $\log\det \mathbf{M}_t^s$ is monotonic and supermodular with respect to the set $\mathcal{S}$ [46]. Thus, a simple greedy algorithm with complexity $\mathcal{O}(PS^3)$ can be employed to find the set of paths that are at least 63% optimal [43]; see Figure 4. Consequently, the technique can be readily applied to large-scale networks since the complexity increases only linearly with P. The framework also admits related problem formulations such as selecting the best set of monitors (nodes) capable of measuring delay on all its outgoing paths. This represents a significant departure from state-of-the-art delay prediction/tracking methods [18], [19], where path selection is heuristic. Note that training is required to estimate the model parameters $\mathbf{C}_\eta$ and α. To this end, empirical estimation techniques similar to those in [42] can be adapted to the present case.

TRAFFIC VOLUME ANOMALIES ARE TYPICALLY DUE TO NETWORK EQUIPMENT MISCONFIGURATION OR OUTRIGHT FAILURE, UNFORESEEN BEHAVIORS FOLLOWING ROUTING POLICY CHANGES, OR CYBERATTACKS.

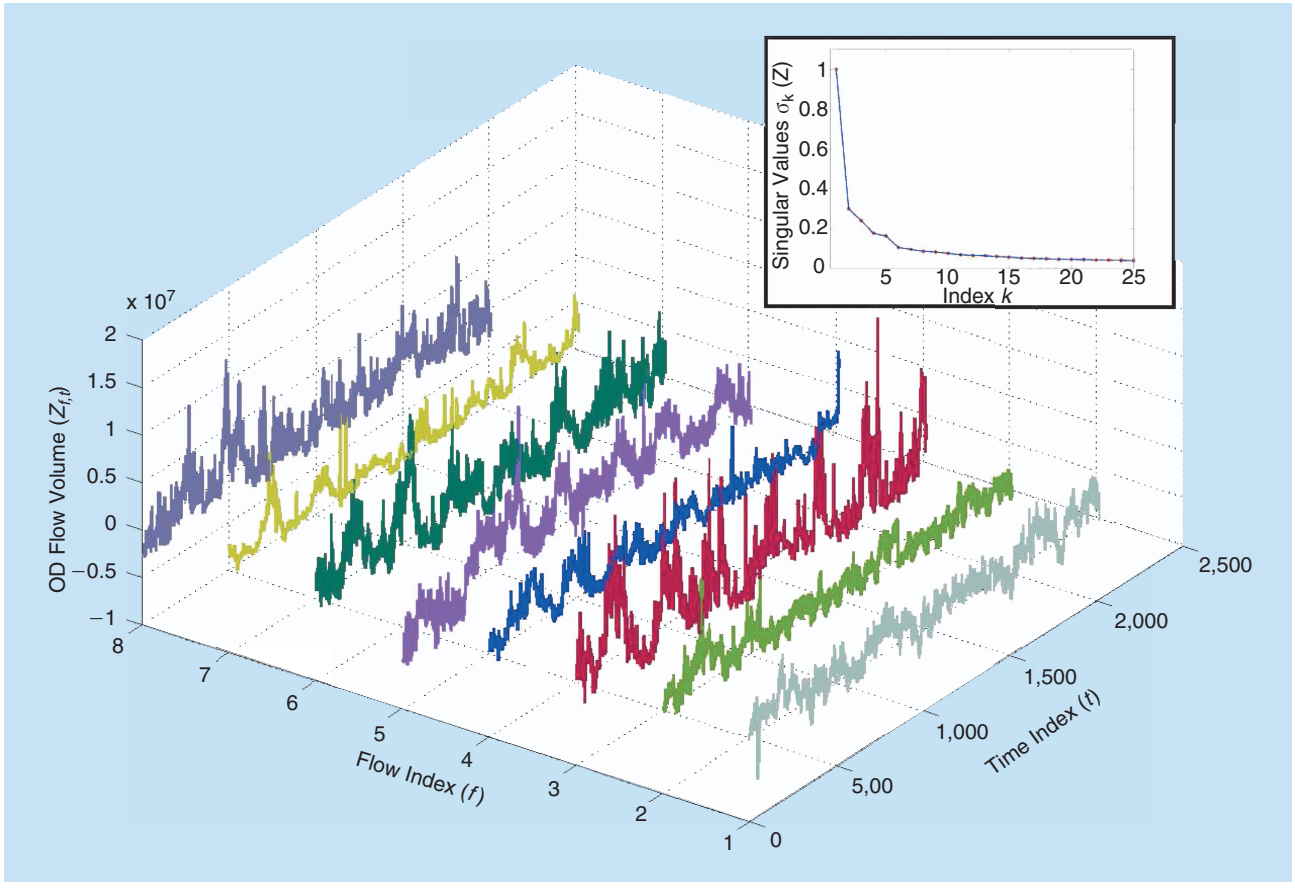
DYNAMIC ANOMALOGRAPHY

This section switches gears to anomalography, the problem of unveiling and mapping out network traffic anomalies across flows and time given link-level traffic measurements. This is a crucial monitoring task toward engineering network traffic since anomalies can result in congestion and limit QoS provisioning.

TRAFFIC MODELING

Consider a backbone IP network where $\mathcal{N}$ and $\mathcal{L}$ denote the sets of nodes (routers) and physical links of cardinality $|\mathcal{N}| = N$ and $|\mathcal{L}| = L$, respectively. The operational goal of the network is to transport a set of OD traffic flows $\mathcal{F}$ (with $|\mathcal{F}| = F$) associated with specific OD (ingress-egress router) pairs. Single-path routing is adopted here, meaning a given flow's traffic is carried through multiple links connecting the corresponding source-destination pair along a single path. Accordingly, over a discrete time horizon $t \in [1, T]$ the measured link counts $\mathbf{X} := [x_{l,t}] \in \mathbb{R}^{L\times T}$ and (unobservable) OD flow traffic matrix $\mathbf{Z} := [z_{f,t}] \in \mathbb{R}^{F\times T}$ are thus related through $\mathbf{X} = \mathbf{RZ}$ [cf. (2)]. Unless otherwise stated, the routing matrix $\mathbf{R}$ is assumed given since it can be otherwise estimated using traceroute or topology inference algorithms [25]. It is also fat, as for backbone networks the number of OD flows is much larger than the number of physical links ($F \gg L$). A cardinal property of the traffic matrix is noteworthy. Common temporal patterns across OD traffic flows in addition to their almost periodic behavior, render most rows (respectively columns) of the traffic matrix linearly dependent, and thus $\mathbf{Z}$ typically has low rank. This intuitive property has been extensively validated with real network data; see Figure 5 and [34].

It is not uncommon for some of the OD flow rates to experience unexpected abrupt changes. These so-termed traffic



[FIG5] Volumes of six representative (out of 121 total) OD flows, taken from the operation of Internet-2 during a seven-day period [1]. Temporal periodicities and correlations across flows are apparent. As expected, in this case, Z can be well approximated by a low-rank matrix, since its normalized singular values decay rapidly to zero.

volume anomalies are typically due to (unintentional) network equipment misconfiguration or outright failure, unforeseen behaviors following routing policy modifications, or cyberattacks (e.g., denial-of-service attacks) that aim at compromising the services offered by the network [34], [55]. Let $a_{f,t}$ denote the unknown amount of anomalous traffic in flow f at time t, which one wishes to estimate. Explicitly accounting for the presence of anomalous flows, the measured traffic carried by link l is then given by $y_{l,t} = \sum_{f \in \mathcal{F}} r_{l,f}(z_{f,t} + a_{f,t}) + \epsilon_{l,t}$, $t = 1, ..., T$, where the noise variables $\epsilon_{l,t}$ capture measurement errors and unmodeled dynamics. Traffic volume anomalies are (unsigned) sudden changes in OD flow's traffic, and as such their effect can span multiple links in the network. A key difficulty in unveiling anomalies from link-level measurements only is that oftentimes, clearly discernible anomalous spikes in the flow traffic can be masked through "destructive interference" of the superimposed OD flows [34]. An additional challenge stems from missing link-level measurements $y_{l,t}$, an unavoidable operational reality affecting most traffic engineering tasks that rely on (indirect) measurement of traffic matrices [48], [56]. To model missing link measurements, collect the tuples (l,t) associated with the available observations $y_{l,t}$ in the set $\Omega \subseteq [1, 2, ..., L] \times [1, 2, ..., T]$. Introducing the matrices $\mathbf{Y} := [y_{l,t}], \mathbf{E} := [\epsilon_{l,t}] \in \mathbb{R}^{L \times T}$, and $\mathbf{A} := [a_{f,t}] \in \mathbb{R}^{F \times T}$, the (possibly incomplete) set of link-traffic measurements can be expressed in compact matrix form as

$$\mathcal{P}_\Omega(\mathbf{Y}) = \mathcal{P}_\Omega(\mathbf{X} + \mathbf{R}\mathbf{A} + \mathbf{E}), \tag{9}$$

where the sampling operator $\mathcal{P}_\Omega(.)$ sets the entries of its matrix argument not in Ω to zero and keeps the rest unchanged. Since the objective here is not to estimate the OD flow traffic matrix $\mathbf{Z}$, (9) is expressed in terms of the nominal (anomaly free) link-level traffic rates $\mathbf{X}$, which inherits the low-rank property of $\mathbf{Z}$. Anomalies in $\mathbf{A}$ are expected to occur sporadically over time and last for a short time relative to the (possibly long) measurement interval $[1, T]$. In addition, only a small fraction of the flows is supposed to be anomalous at a any given time instant. This renders the anomaly traffic matrix $\mathbf{A}$ sparse across both rows (flows) and columns (time).

UNVEILING ANOMALIES VIA SPARSITY AND LOW RANK

Given link-level traffic measurements $\mathcal{P}_\Omega(\mathbf{Y})$ adhering to (9), dynamic anomalography is a critical network monitoring task that aims at accurately estimating the anomaly matrix $\mathbf{A}$.

As argued next, capitalizing on the sparsity of $\mathbf{A}$ and the low-rank property of $\mathbf{X}$ will be instrumental in achieving this ambitious goal. From a network cartography vantage point, the resultant estimated map $\hat{\mathbf{A}}$ offers a depiction of the network's "health state" along both the flow and time dimensions. If $|\hat{a}_{f,t}| > 0$, the fth flow at time t is deemed anomalous, otherwise it is healthy. This joint estimation-detection task not only allows one to identify the time of the anomaly in addition to the affected flows but also to estimate its magnitude, which hints to the importance of the anomaly event. By examining $\mathbf{R}$, the network operator can immediately determine the links carrying the anomalous flows. Subsequently, planned contingency measures involving traffic-engineering algorithms can be implemented to address network congestion.

The low-rank property of the traffic matrix $\mathbf{Z}$ (and $\mathbf{X}$) is at the heart of the seminal network anomaly detection approach in [34]. In the absence of missing data, the method therein adopts principal component analysis (PCA) to decompose the link traffic $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_T]$ into nominal and anomalous components (also known as modeled and residual traffic). For instance, if most of the variance in $\mathbf{Y}$ is captured by $r \ll \min(L, T)$ dominant principal components, then by construction the nominal subspace $\mathcal{S}_n$ is spanned by the r dominant right singular vectors of $\mathbf{Y}'$ (cf. the low-rank assumption). Naturally, the anomalous subspace $\mathcal{S}_a$ corresponds to the orthogonal complement, i.e., $\mathcal{S}_a := \mathcal{S}_n^{\perp}$. In the operational phase, an anomaly is declared at time t when $\|\mathbf{P}_{\mathcal{S}_a}\mathbf{y}_t\|_2^2$ exceeds a given threshold, where $\mathbf{P}_{\mathcal{S}_a}$ is an orthogonal projection matrix onto $\mathcal{S}_a$. Subsequently, a single anomalous flow is identified after running a greedy algorithm, and an estimate of the amount of anomalous traffic is obtained as a byproduct. Notice that for large networks and reduced number of measurements ($L \gg T$), one should resort to "high-dimensional" variants of PCA to obtain satisfactory performance; see, e.g., [4], [28], and references therein.

Likewise, the spatial approach within the network anomography framework [55] forms the matrix $\mathbf{P}_{\mathcal{S}_a}\mathbf{Y}$ of link anomalies, thus exploiting the correlation between traffic across different links. Temporal approaches obtain link anomalies as $\mathbf{YT}$ instead, where $\mathbf{T}$ is a linear operator which judiciously filters the traffic time series per link (implementing an "anomaly pass" filter). Several choices for $\mathbf{T}$ are proposed to this end, based on different forms of temporal analysis including autoregressive integrated moving average (ARIMA), wavelets, and fast Fourier transform (FFT). Different from [34], the inference algorithm in [55] capitalizes on the sparsity of $\mathbf{A}$ to estimate the anomaly map by, e.g., solving in the spatial case

$$\hat{\mathbf{A}} := \arg\min_{\mathbf{A}} \|\mathbf{A}\|_1, \quad \text{s.t.} \quad \mathbf{P}_{\mathcal{S}_a}\mathbf{Y} = \mathbf{R}\mathbf{A}.$$

Network anomography algorithms can be extended to accommodate routing changes across time; see [55] for further details and comprehensive performance tests.

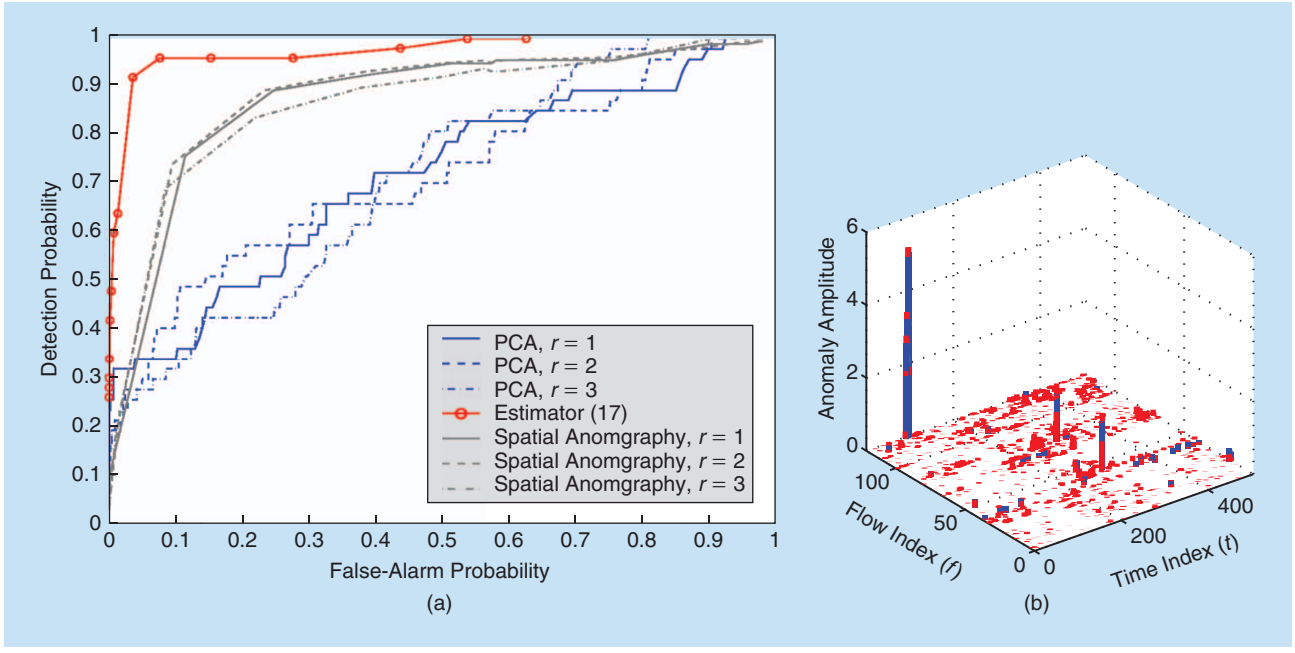
Recently, a natural estimator leveraging the low rank property of $\mathbf{X}$ and the sparsity of $\mathbf{A}$ was put forth in [39], which can be found at the crossroads of CS [24] and timely low-rank plus sparse matrix decompositions [11], [15]. The idea is to fit the incomplete data $\mathcal{P}_{\Omega}(\mathbf{Y})$ to the model $\mathbf{X} + \mathbf{R}\mathbf{A}$ [cf. (9)] in the LS error sense, as well as minimize the rank of $\mathbf{X}$, and the number of nonzero entries of $\mathbf{A}$ measured by its ℓ_0 -(pseudo) norm. Unfortunately, albeit natural both rank and ℓ_0-norm criteria are in general NP-hard to optimize. Typically, the nuclear norm $\|\mathbf{X}\|_* := \sum_k \sigma_k(\mathbf{X})$ ($\sigma_k(\mathbf{X})$ denotes the kth singular value of $\mathbf{X}$) and the ℓ_1-norm $\|\mathbf{A}\|_1$ are adopted as surrogates [12], [26], since they are the closest convex approximants to $\text{rank}(\mathbf{X})$ and $\|\mathbf{A}\|_0$, respectively. Accordingly, one solves

$$\min_{\{\mathbf{X},\,\mathbf{A}\}} \|\mathcal{P}_{\Omega}(\mathbf{Y} - \mathbf{X} - \mathbf{R}\mathbf{A})\|_F^2 + \lambda_*\|\mathbf{X}\|_* + \lambda_1\|\mathbf{A}\|_1, \tag{10}$$

where $\lambda_*, \lambda_1 \geq 0$ are rank- and sparsity-controlling parameters. While a nonsmooth optimization problem, being convex (10) is appealing. An efficient accelerated proximal gradient algorithm with quantifiable iteration complexity was developed to unveil network anomalies [40]. Interestingly, (10) also offers a cleansed estimate of the link-level traffic $\hat{\mathbf{X}}$ that could be subsequently utilized for network tomography tasks. In addition, (10) jointly exploits the spatiotemporal correlations in the link traffic as well as the sparsity of the anomalies, through an optimal single-shot estimation-detection procedure that has been shown to outperform the algorithms in [34] and [55] (that decouple the estimation and detection steps); see Figure 6.

THIS JOINT ESTIMATION DETECTION TASK NOT ONLY ALLOWS ONE TO IDENTIFY THE TIME OF THE ANOMALY IN ADDITION TO THE AFFECTED FLOWS BUT ALSO TO ESTIMATE ITS MAGNITUDE, WHICH HINTS TO THE IMPORTANCE OF THE ANOMALY EVENT.

Before moving on to distributed implementations, it is instructive to elaborate on the generality of (10). When there is no missing data and $\mathbf{X} = \mathbf{0}_{L\times T}$, one is left with an underdetermined sparse signal recovery problem typically encountered with CS; see, e.g., [24]. The decomposition $\mathbf{Y} = \mathbf{X} + \mathbf{A}$ corresponds to principal component pursuit (PCP), also referred to as robust PCA [11], [15]. For the idealized noise-free setting ($\mathbf{E} = \mathbf{0}_{L\times T}$), sufficient conditions for exact recovery of the unknowns are available for both of the aforementioned special cases [11], [12], [15]. However, the superposition of a low-rank plus a compressed sparse matrix in (9) further challenges identifiability of $\{\mathbf{X}, \mathbf{A}\}$; see [40] for early results. Going back to the CS paradigm, even when $\mathbf{X}$ is nonzero one could envision a variant where the measurements are corrupted with correlated (low-rank) noise [16]. Last but not least, when $\mathbf{A} = \mathbf{0}_{F\times T}$ and $\mathbf{Y}$ is noisy, the recovery of $\mathbf{X}$ subject to a rank constraint is nothing but PCA—arguably, the workhorse of high-dimensional data analytics. This same formulation is



[FIG6] Unveiling anomalies from Internet-2 data [1]. (a) ROC curve comparison between (10) and the PCA methods in [34] and [55], for different values of $r := \dim(\mathcal{S}_n)$. Leveraging sparsity and low rank jointly leads to improved performance. (b) In red, the estimated anomaly map $\hat{\mathbf{A}}$ obtained via (10) superimposed to the "true" anomalies shown in blue [38]. (Figure used with permission from [38].)

adopted for low-rank matrix completion, to impute the missing entries of a low-rank matrix observed in noise, i.e., $\mathcal{P}_\Omega(\mathbf{Y}) = \mathcal{P}_\Omega(\mathbf{X}+\mathbf{E})$ [13].

IN-NETWORK DISTRIBUTED PROCESSING

Implementing (10) presumes that network nodes continuously communicate their link traffic measurements to a central monitoring station, which uses their aggregation in $\mathcal{P}_\Omega(\mathbf{Y})$ to unveil anomalies. While for the most part this is the prevailing operational paradigm adopted in current networks, it is fair to say there are limitations associated with this architecture. For instance, fusing all this information may entail excessive protocol overheads. Moreover, minimizing the exchanges of raw measurements may be desirable to reduce unavoidable communication errors that translate to missing data. Solving (10) centrally raises robustness concerns as well, since the central monitoring station represents an isolated point of failure.

These reasons motivate well devising fully distributed iterative algorithms for dynamic anomalography, embedding the network anomaly detection functionality to the routers. In a nutshell, per iteration nodes $n \in \mathcal{N}$ carry out simple computational tasks locally, relying on their own link count measurements (a submatrix $\mathbf{Y}_n$ within $\mathbf{Y} = [\mathbf{Y}_1', \ldots, \mathbf{Y}_N']'$ corresponding to router n's links). Subsequently, local estimates are refined after exchanging messages only with directly connected neighbors, which facilitates percolation of local information to the whole network. The end goal is for network nodes to consent on a global map of network anomalies $\hat{\mathbf{A}}$, and attain (or at least come close to) the estimation performance of the centralized counterpart (10) which has all data $\mathcal{P}_\Omega(\mathbf{Y})$ available.

Equation (10) is not amenable for distributed implementation due to the nonseparable nuclear norm present in the cost function. If an upper bound $\text{rank}(\hat{\mathbf{X}}) \leq \rho$ is a priori available [recall $\hat{\mathbf{X}}$ is the estimated link-level traffic obtained via (10)], (10)'s search space is effectively reduced and one can factorize the decision variable as $\mathbf{X} = \mathbf{P}\mathbf{Q}'$, where $\mathbf{P}$ and $\mathbf{Q}$ are $L \times \rho$ and $T \times \rho$ matrices, respectively. Again, it is possible to interpret the columns of $\mathbf{X}$ (viewed as points in $\mathbb{R}^L$) as belonging to a low-rank nominal subspace $\mathcal{S}_n$, spanned by the columns of $\mathbf{P}$. The rows of $\mathbf{Q}$ are thus the projections of the columns of $\mathbf{X}$ onto $\mathcal{S}_n$. Next, consider the following alternative characterization of the nuclear norm (see, e.g., [47])

$$\|\mathbf{X}\|_* := \min_{\{\mathbf{P},\mathbf{Q}\}} \frac{1}{2}(\|\mathbf{P}\|_F^2 + \|\mathbf{Q}\|_F^2), \quad \text{s.t.} \quad \mathbf{X} = \mathbf{P}\mathbf{Q}', \tag{11}$$

where the optimization is over all possible bilinear factorizations of $\mathbf{X}$, so that the number of columns ρ of $\mathbf{P}$ and $\mathbf{Q}$ is also a variable. Leveraging (11), the following reformulation of (10) provides an important first step towards obtaining a distributed anomalography algorithm

$$\min_{\{\mathbf{P},\mathbf{Q},\mathbf{A}\}} \sum_{n=1}^{N} \Big[\|\mathcal{P}_{\Omega_n}(\mathbf{Y}_n - \mathbf{P}_n\mathbf{Q}' - \mathbf{R}_n\mathbf{A})\|_F^2 + \frac{\lambda_*}{2N}(N\|\mathbf{P}_n\|_F^2 + \|\mathbf{Q}\|_F^2) + \frac{\lambda_1}{N}\|\mathbf{A}\|_1 \Big], \tag{12}$$

which is nonconvex due to the bilinear terms $\mathbf{P}_n\mathbf{Q}'$, and where $\mathbf{R} := [\mathbf{R}_1', \ldots, \mathbf{R}_N']'$ is partitioned into local routing tables available per router n. Adopting the separable Frobenius-norm

regularization in (12) comes with no loss of optimality relative to (10), provided $\text{rank}(\hat{\mathbf{X}}) \leq \rho$. By finding the global minimum of (12) [which could have considerably less variables than (10)], one can recover the optimal solution of (10). But since (12) is nonconvex, it may have stationary points which need not be globally optimum. As asserted in [39, Prop. 1] however, if a stationary point $\{\bar{\mathbf{P}}, \bar{\mathbf{Q}}, \bar{\mathbf{A}}\}$ of (12) satisfies $\|\mathcal{P}_\Omega(\mathbf{Y} - \bar{\mathbf{P}}\bar{\mathbf{Q}}' - \bar{\mathbf{A}})\| < \lambda_*$, then $\{\hat{\mathbf{X}} := \bar{\mathbf{P}}\bar{\mathbf{Q}}', \hat{\mathbf{A}} := \bar{\mathbf{A}}\}$ is the globally optimal solution of (10). Note that for sufficiently small ρ the residual $\|\mathcal{P}_\Omega(\mathbf{Y} - \bar{\mathbf{P}}\bar{\mathbf{Q}}' - \bar{\mathbf{A}})\|$ becomes large, and the qualification inequality is violated [unless λ_* is large enough, in which case a sufficiently low-rank solution to (10) is expected].

NONSTATIONARITIES DUE TO ROUTING CHANGES AND MISSING DATA FURTHER CHALLENGE IDENTIFIABILITY OF ANOMALIES.

To decompose the cost in (12), in which summands inside the square brackets are coupled through the global variables $\{\mathbf{Q}, \mathbf{A}\}$, introduce auxiliary copies $\{\mathbf{Q}_n, \mathbf{A}_n\}_{n=1}^N$ representing local estimates of $\{\mathbf{Q}, \mathbf{A}\}$, one per node n. These local copies along with consensus constraints yield the distributed estimator

$$\min_{\{\mathbf{P}_n, \mathbf{Q}_n, \mathbf{A}_n\}} \sum_{n=1}^{N} \Big[\|\mathcal{P}_{\Omega_n}(\mathbf{Y}_n - \mathbf{P}_n\mathbf{Q}_n' - \mathbf{R}_n\mathbf{A}_n)\|_F^2 + \frac{\lambda_*}{2N}(N\|\mathbf{P}_n\|_F^2 + \|Q_n\|_F^2) + \frac{\lambda_1}{N}\|\mathbf{A}_n\|_1 \Big]$$
$$\text{s.t.} \quad \mathbf{Q}_n = \mathbf{Q}_m, \mathbf{A}_n = \mathbf{A}_m \quad m \text{ linked with } n \in \mathcal{N}, \qquad (13)$$

which is equivalent to (12) provided the network topology graph is connected. Even though consensus is a fortiori imposed within neighborhoods, it extends to the whole (connected) network and local estimates agree on the global solution of (12). Exploiting the separable structure of (13), a general framework for in-network sparsity-regularized rank minimization was put forth in [39]. Specifically, distributed iterations were obtained after adopting the alternating-direction method of multipliers (ADMM), an iterative Lagrangian method well-suited for parallel processing [10]. In a nutshell, local tasks per iteration $k = 1, 2, \ldots$ entail solving small unconstrained quadratic programs to refine the normal subspace $\mathbf{P}_n[k]$, in addition to soft-thresholding operations to update the anomaly maps $\mathbf{A}_n[k]$ per router. Each iteration, routers exchange their estimates $\{\mathbf{Q}_n[k], \mathbf{A}_n[k]\}$ only with directly connected neighbors. This way the communication overhead remains affordable, and independent of the network size N.

When employed to solve nonconvex problems such as (13), so far ADMM offers no convergence guarantees. However, there is ample experimental evidence in the literature that supports empirical convergence of ADMM, especially when the nonconvex problem at hand exhibits "favorable" structure. For instance, (13) is a linearly constrained biconvex problem with potentially good convergence properties—extensive numerical tests in [39] demonstrate that this is indeed the case. While establishing convergence remains an open problem, one can still prove that upon convergence the distributed iterations attain consensus and global optimality, offering the desirable centralized performance guarantees [39].

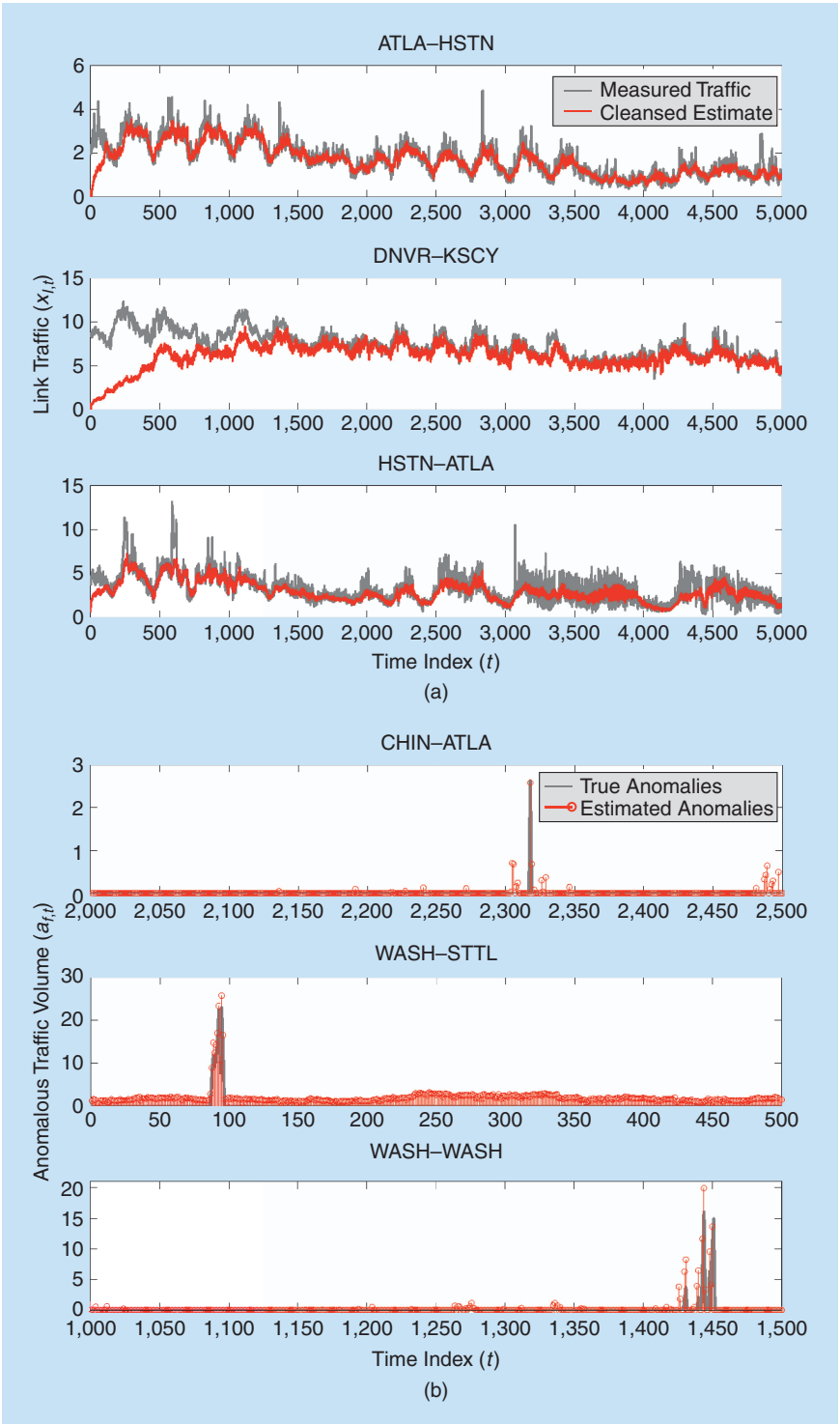
REAL-TIME ANOMALY TRACKERS

Monitoring large-scale IP networks necessitates massive recollection of data which far outweigh the ability of modern computers to store and analyze them in real time. In addition, nonstationarities due to routing changes and missing data further challenge identification of anomalies. In dynamic networks routing tables are constantly readjusted to effect traffic load balancing and avoid congestion caused by, e.g., traffic anomalies. To account for slowly time-varying routing tables, let $\mathbf{R}_t \in \mathbb{R}^{L \times F}$ denote the routing matrix at time t. In this dynamic setting, the partially observed link counts at time t adhere to $\mathcal{P}_{\Omega_t}(\mathbf{y}_t) = \mathcal{P}_{\Omega_t}(\mathbf{x}_t + \mathbf{R}_t\mathbf{a}_t + \epsilon_t)$, $t = 1, 2, \ldots$, where the link-level traffic $\mathbf{x}_t := \mathbf{R}_t\mathbf{z}_t$. In general, routing changes may alter a link load considerably by, e.g.,routing traffic completely away from a specific link. Therefore, even though the OD flow vectors $\{\mathbf{z}_t\}$ live in a low-dimensional subspace, the same may not be true for the $\{\mathbf{x}_t\}$ when the routing updates are major and frequent. In backbone networks however, routing changes are sporadic relative to the time-scale of data acquisition used for network monitoring tasks. For example, data collected from the operation of Internet-2 network reveals that only a few rows of $\mathbf{R}_t$ change per week [1]. It is thus safe to assume that $\{\mathbf{x}_t\}$ still lies in a low-dimensional subspace, and exploit the spatiotemporal correlations of the observations to identify the anomalies in real time.

On top of the previous arguments, in practice link measurements are acquired sequentially in time, which motivates updating previously obtained estimates rather than recomputing new ones from scratch each time a new datum becomes available. The goal is then to recursively estimate $\{\hat{\mathbf{x}}_t, \hat{\mathbf{a}}_t\}$ at time t from historical observations $\{\mathcal{P}_{\Omega_\tau}(\mathbf{y}_\tau)\}_{\tau=1}^t$, naturally placing more importance on recent measurements. To this end, one possible adaptive counterpart to (12) is the exponentially weighted LS estimator found by minimizing the empirical cost [38]

$$\min_{\{\mathbf{P}, \mathbf{Q}, \mathbf{A}\}} \sum_{\tau=1}^{t} \beta^{t-\tau} \Big[\|\mathcal{P}_{\Omega_\tau}(\mathbf{y}_\tau - \mathbf{P}\mathbf{q}_\tau - \mathbf{R}_\tau\mathbf{a}_\tau)\|_2^2 + \frac{\lambda_*}{2\sum_{u=1}^{t}\beta^{t-u}}\|\mathbf{P}\|_F^2 + \frac{\lambda_*}{2}\|\mathbf{q}_\tau\|_2^2 + \lambda_1\|\mathbf{a}_\tau\|_1 \Big] \qquad (14)$$

in which $0 < \beta \leq 1$ is the so-termed forgetting factor. When $\beta < 1$ data in the distant past are exponentially downweighted, which facilitates tracking network anomalies in nonstationary environments. For static routing ($\mathbf{R}_t = \mathbf{R}$) and infinite memory ($\beta = 1$), the formulation (14) coincides with the batch estimator (12). A provably convergent online algorithm for dynamic anomalography is developed in [38], based on alternating



[FIG7] Unveiling anomalies in real time from Internet-2 data [1]. (a) Measured link traffic and cleansed estimates for three representative links. (b) Three rows of the estimated anomaly map $\hat{\mathbf{A}}$ corresponding to three anomalous flows [38]. (Figure used with permission from [38].)

minimization of (14); see Figure 7. Each time a new datum is acquired, anomaly estimates are formed via the Lasso [28, p. 68], and the low-rank nominal traffic subspace is refined using recursive LS. For situations were reducing computational complexity is critical, an online stochastic gradient algorithm based on Nesterov's acceleration technique is developed as well [38].

Algorithms in [38] are closely related to timely robust subspace trackers, which aim at estimating a low-rank subspace $\mathbf{P}$ from grossly corrupted and possibly incomplete data, particularly $\mathcal{P}_{\Omega_t}(\mathbf{y}_t) = \mathcal{P}_{\Omega_t}(\mathbf{P}\mathbf{q}_t + \mathbf{a}_t + \epsilon_t)$, $t = 1, 2, \ldots$. In the absence of sparse "outliers" $\{\mathbf{a}_t\}_{t=1}^{\infty}$, an online algorithm based on incremental gradient descent on the Grassmannian manifold of subspaces was put forth in [5]. The second-order RLS-type algorithm in [17] extends the seminal projection approximation subspace tracking (PAST) algorithm to handle missing data. When outliers are present, robust counterparts can be found in [16], [29]. Relative to all aforementioned works, the estimation problem (14) is more challenging due to the presence of the (compression) routing matrix $\mathbf{R}_t$; see [40] for fundamental identifiability issues related to the model (9).

BROADENING THE NETWORK ATLAS

Additional cartography instances are outlined in this section, including anomalography from flow measurements and network distance prediction. To exemplify the development of sensing infrastructure for situational awareness at the physical layer of wireless CR networks, the notion of radio-frequency (RF) cartography is introduced as well. All these problems can be tackled through SP methods subsumed by (10), particularly PCP [15], low-rank matrix completion [13], the Lasso [28, p. 68], and nonparametric versions of basis pursuit [8].

UNVEILING ANOMALIES FROM FLOW DATA

Since some networks today collect OD flow (not link-level) measurements $z_{f,t} + a_{f,t}$ for at least part of their network (using, e.g., the Netflow protocol), anomalies can be detected using temporal decomposition and standard change-detection approaches per flow. Leveraging the low-rank property of the traffic matrix and the sparsity of anomalies, anomalography from OD flow measurements was

formulated as the PCP matrix decomposition problem and solved centrally in [3]; see also [39] for a distributed implementation of the PCP estimator aimed at scalable monitoring of networks.

NETWORK DISTANCE PREDICTION

End-to-end network distance information is critical towards enhancing QoS in Internet applications such as content distribution and peer-to-peer file sharing systems. Clients naturally prefer to establish connections with "closer" network resources or servers that are likely to respond faster. There are different metrics to quantify the distance between a pair of network nodes. The most common choices are defined in terms of latency (one-way delay and the so-termed round-trip time) or router hop-counts. Unfortunately, either probing or passively measuring all pairwise distances becomes infeasible in large-scale networks. Given those few affordable distance measurements, the problem of network distance prediction is to impute (that is interpolate) the missing entries in a highly incomplete matrix of end-to-end distances.

THE UNCEASING DEMAND FOR CONTINUOUS SITUATIONAL AWARENESS CALLS FOR INNOVATIVE AND LARGE-SCALE DISTRIBUTED SP ALGORITHMS.

If one collects the end-to-end latencies $d_{i,j}$ of source-sink pairs (i, j) in a delay matrix $\mathbf{D} := [d_{i,j}] \in \mathbb{R}^{N \times N}$, strong dependencies among path delays render $\mathbf{D}$ low rank; see, e.g., [36] for an experimental validation with multiple data sets. Intuitively, correlations among rows and columns of $\mathbf{D}$ emerge because nearby nodes (e.g., those belonging to a common subnetwork) are connected to every other node through paths with significant overlap, possibly sharing common bottleneck links. The low-rank property of $\mathbf{D}$ along with the distributed-processing requirements of large-scale networks, motivated decentralized matrix-factorization [36], and nuclear-norm minimization [39] algorithms for network distance prediction. Different from schemes based on Euclidean embedding via multidimensional scaling [23], low-rank modeling does not require distances in $\mathbf{D}$ to be symmetric and satisfy the triangle inequality—properties that are oftentimes violated by network-related distances [35].

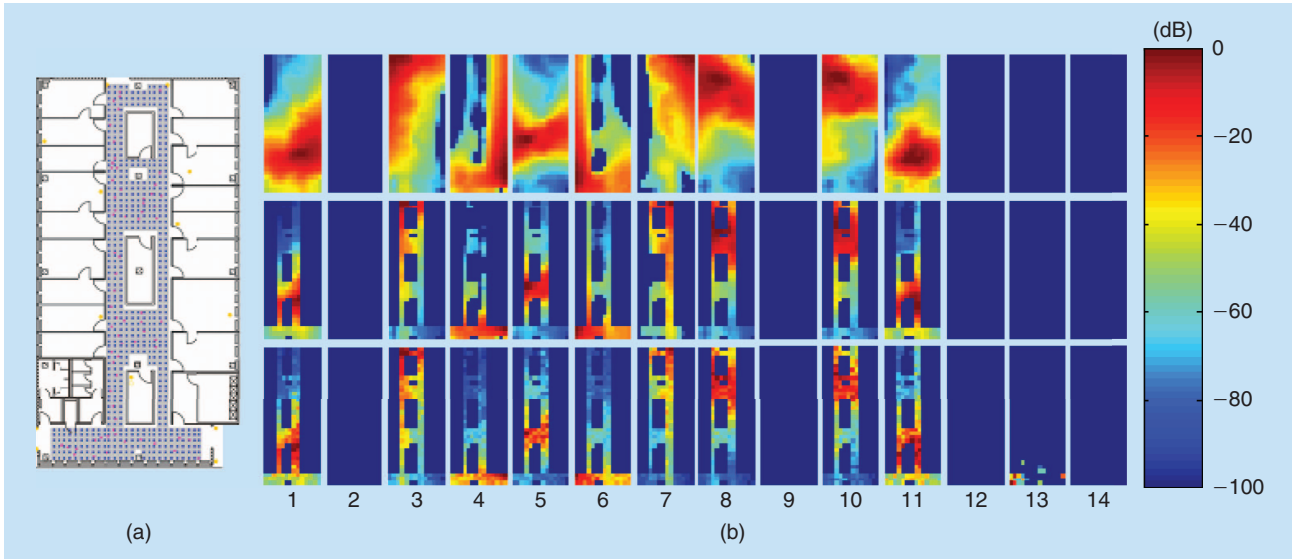
To avoid the excessive overhead of active probing mechanisms, one can leverage network monitors that passively observe router hop-counts from traffic traversing those monitored links; see, e.g., [25] and references therein. Collect these hop-count measurements in the matrix $\mathbf{H} := [h_{m,n}] \in \mathbb{N}^{M \times N}$, where M is the number of monitors, and N ($\gg M$) the total hosts observed. Because monitor m only observes a fraction of the total network traffic, $\mathbf{H}$ will be depleted with missing entries. Despite typically having $\text{rank}(\mathbf{H}) = M$, $\mathbf{H}$ consists of low-rank column blocks, each corresponding to a subnetwork with access to the Internet core through a single border router. Recognizing this structure, a high-rank matrix completion algorithm that performs subspace clustering of incomplete hop-count data was put forth in [25], and shown to attain good performance both in theory and practice.

Different from the dynamic network delay cartography problem considered in the section "Delay Cartography via Dynamic Network Kriging," network distance prediction approaches do not account for the temporal variations in the delays and typically rely on batch imputation of the distance matrix of interest. The techniques used in the section "Delay Cartography via Dynamic Network Kriging" do not apply in this context either, since some path delays are never observed, and thus it is impossible to estimate the spatial covariance matrices (such as $\mathbf{C}_\eta$ and $\mathbf{C}_\nu$) completely.

RF CARTOGRAPHY

In the domain of spectrum sensing for CR networks, RF cartography amounts to constructing in a distributed fashion: 1) global power spectral density (PSD) maps capturing the distribution of radiated power across space, time, and frequency and 2) local channel gain (CG) maps offering the propagation medium per frequency from each node to any point in space. These maps enable identification of opportunistically available spectrum bands for reuse and handoff operation as well as localization, transmit-power estimation, and tracking of primary user activities. While the focus here is on the construction of PSD maps, the interested reader is referred to [30] for a tutorial treatment on CG cartography.

A cooperative approach to RF cartography was introduced in [7] that builds on a basis expansion model of the PSD map $\Phi(\mathbf{x}, f)$ across space $\mathbf{x} \in \mathbb{R}^2$, and frequency f. Spatially distributed CRs collect smoothed periodogram samples of the received signal at given sampling frequencies, based on which they want to determine the unknown expansion coefficients. Introducing a virtual spatial grid of candidate source locations, the estimation task can be cast as a linear LS problem with an augmented vector of unknown parameters. Still, the problem complexity (or effective degrees of freedom) can be controlled by capitalizing on two forms of sparsity: the first one introduced by the narrowband nature of transmit-PSDs relative to the broad swaths of usable spectrum and the second one emerging from sparsely located active radios in the operational space (due to the grid artifact). Nonzero entries in the parameter vector sought correspond to spatial location-frequency band pairs corresponding to active transmissions. All in all, estimating the PSD map and locating the active transmitters as a byproduct boils down to a variable selection problem. This motivates well employment of the Lasso for distributed sparse linear regression [39], an estimator also subsumed by (10) when $\mathbf{X} = \mathbf{0}_{L \times T}$, $T = 1$, and the regression matrix $\mathbf{R}$ has a specific structure that depends on the chosen bases and path-loss propagation model.



[FIG8] Spline-based RF cartography using the data set [32]. (a) Detailed floor plan schematic including the location of $N = 166$ sensing radios. The lower panel of (b) shows original measurements spanning 14 frequency subbands while the center panel of (b) shows the estimated maps over the surveyed area. The top panel of (b) shows extrapolated maps. The proposed estimator is capable of recovering the nine (out of 14 total) center frequencies that are being utilized for transmission. It accurately recovers the power levels in the surveyed area with a smooth extrapolation to zones where there are no measurements and suggests possible locations for the transmitters [8]. (Figure used with permission from [8] and [32].)

Sparse total LS variants are also available to cope with uncertainty in the regression matrix, arising due to inaccurate channel estimation and grid-mismatch effects [30]. Nonparametric spline-based PSD map estimators [8] have been also shown effective in capturing general propagation characteristics including both shadowing and fading; see also Figure 8 for an actual PSD atlas spanning 14 frequency subbands.

CONCLUSIONS

In this tutorial, the concept of dynamic network cartography is introduced as a framework to construct maps of the dynamically evolving network state, in an efficient and scalable manner even for large-scale heterogeneous networks. Here the focus is placed on key tasks geared to obtaining full yet succinct representation of network state metrics such as link traffic and path delays as well as prompt and accurate identification of network anomalies from possibly partial and corrupted measurement data.

Looking forward, the unceasing demand for continuous situational awareness calls for innovative and large-scale distributed SP algorithms, complemented by collaborative and adaptive monitoring platforms to accomplish the objectives of network management and control. Avenues where significant impact can be made include: 1) judicious design of critical cognition infrastructure to sense, learn, and adapt to the environment where networks operate; 2) development of scalable tools for distilling, summarizing, and tracking the network state for the purpose of network management; 3) ensuring robustness in the face of missing and grossly corrupted network data, in addition to possibly malicious attacks; and 4) developing effective network adaptation techniques based on global network inference, further impacting protocol designs, network taxonomy, and categorization.

ACKNOWLEDGMENTS

The work in this article was supported by the NSF-ECCS grant 1202135. The authors would like to thank Prof. G.B. Giannakis (University of Minnesota) for his invaluable help as Ph.D. advisor.

AUTHORS

Gonzalo Mateos (mate0058@umn.edu) received his B.Sc. degree in electrical engineering from Universidad de la Republica, Uruguay, in 2005 and the M.Sc. and Ph.D. degrees in electrical and computer engineering from the University of Minnesota, in 2009 and 2012. Since 2012, he has been a postdoctoral research associate with the Department of Electrical and Computer Engineering, University of Minnesota. From 2003 to 2006, he worked as a systems engineer at ABB, Uruguay. His research interests lie in the areas of SP, statistical learning, and networking. His current research focuses on distributed and sparsity-aware SP for network health monitoring, spectrum sensing for wireless CR networks, and robust learning for social data analysis.

Ketan Rajawat (ketan@iitk.ac.in) received his B.Tech and M.Tech degrees in electrical engineering from the Indian Institute of Technology (IIT) Kanpur, in 2007, and his Ph.D. degree in electrical and computer engineering from the University of Minnesota in 2012. Currently, he is an assistant professor in the Department of Electrical Engineering, IIT Kanpur. His research interests lie in the areas of SP and communication networks. His current research focuses on cross-layer network optimization and dynamic network monitoring.

REFERENCES

[1] (2012). The Internet2 Observatory Data Collections. [Online]. Available: http://www.internet2.edu/observatory/archive/data-collections.html

[2] Wand Network Research Group. (2013). New Zealand Active Measurement Project. [Online]. Available: http://erg.wand.net.nz

[3] A. Abdelkefi, Y. Jiang, W. Wang, A. Aslebo, and O. Kvittem, "Robust traffic anomaly detection with principal component pursuit," in *Proc. ACM CoNEXT Student Workshop*, Philadelphia, PA, article no. 10, Nov. 2010.

[4] A. A. Amini, "High-dimensional principal component analysis," Ph.D. dissertation, Dept. Elect. Eng. Comput. Sci., Univ. California, Berkeley, 2011.

[5] L. Balzano, R. Nowak, and B. Recht, "Online identification and tracking of subspaces from highly incomplete information," in *Proc. Allerton Conf. Communication, Control, and Computing*, Monticello, IL, 2010, pp. 704–711.

[6] V. W. Bandara and A. P. Jayasumana, "Extracting baseline patterns in Internet traffic using robust principal components," in *Proc. IEEE Int. Conf. Local Computer Networks*, Bonn, Germany, 2011, pp. 407–415.

[7] J. A. Bazerque and G. B. Giannakis, "Distributed spectrum sensing for cognitive radio networks by exploiting sparsity," *IEEE Trans. Signal Processing*, vol. 58, pp. 1847–1862, Mar. 2010.

[8] J. A. Bazerque, G. Mateos, and G. B. Giannakis, "Group Lasso on splines for spectrum cartography," *IEEE Trans. Signal Processing*, vol. 59, pp. 4648–4663, Oct. 2011.

[9] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *J. Mach. Learn. Res.*, vol. 7, pp. 2399–2434, Dec. 2006.

[10] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Belmont, MA: Athena Scientific, 1999.

[11] E. J. Candes, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 1, pp. 1–37, 2011.

[12] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Trans. Inform. Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[13] E. Candes and Y. Plan, "Matrix completion with noise," *Proc. IEEE*, vol. 98, no. 6, pp. 925–936, 2009.

[14] R. Castro, M. Coates, G. Liang, R. Nowak, and B. Yu, "Network tomography: Recent developments," *Stat. Sci.*, vol. 19, no. 3, pp. 499–517, 2004.

[15] V. Chandrasekaran, S. Sanghavi, P. R. Parrilo, and A. S. Willsky, "Rank-sparsity incoherence for matrix decomposition," *SIAM J. Optim.*, vol. 21, no. 2, pp. 572–596, 2011.

[16] Q. Chenlu and N. Vaswani, "Recursive sparse recovery in large but correlated noise," in *Proc. Allerton Conf. Communication, Control, and Computing*, Monticello, IL, 2011, pp. 752–759.

[17] Y. Chi, Y. C. Eldar, and R. Calderbank, "Petrels: Subspace estimation and tracking from partial observations," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, Kyoto, Japan, Mar. 2012, pp. 3301–3304.

[18] D. Chua, E. Kolaczyk, and M. Crovella, "Network kriging," *IEEE J. Select. Areas Commun.*, vol. 24, no. 12, pp. 2263–2272, 2006.

[19] M. Coates, Y. Pointurier, and M. Rabbat, "Compressed network monitoring for IP and all-optical networks," in *Proc. ACM Internet Measurement Conf.*, San Diego, CA, Oct. 2007.

[20] G. Conti, *Security Data Visualization: Graphical Techniques for Network Analysis*. San Francisco, CA: No Starch Press, 2007.

[21] J. Cortés, "Distributed kriged Kalman filter for spatial estimation," *IEEE Trans. Autom. Contr.*, vol. 54, no. 12, pp. 2816–2827, Dec. 2009.

[22] N. Cressie, "The origins of kriging," *Math. Geol.*, vol. 22, no. 3, pp. 239–252, 1990.

[23] F. Dabek, R. Cox, F. Kaashoek, and R. Morris, "Vivaldi: A decentralized network coordinate system," in *Proc. ACM SIGCOMM*, Portland, OR, Aug. 2004, pp. 15–26.

[24] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inform. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[25] B. Eriksson, L. Balzano, and R. Nowak, "High-rank matrix completion," in *Proc. Int. Conf. Artificial Intelligence and Statistics*, La Palma, Canary Islands, Apr. 2012, pp. 373–381.

[26] M. Fazel, "Matrix rank minimization with applications," Ph.D. dissertation, Dept. Elect. Eng., Stanford University, CA, 2002.

[27] P. A. Forero, K. Rajawat, and G. B. Giannakis, "Semi-supervised dictionary learning for network-wide link load prediction," in *Proc. Cognitive Information Processing Workshop*, Baiona, Spain, May 2012, pp. 1–5.

[28] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, 2nd ed. New York: Springer, 2009.

[29] J. He, L. Balzano, and A. Szlam, "Incremental gradient on the Grassmannian for online foreground and background separation in subsampled video," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Providence, RI, June 2012, pp. 1568–1575.

[30] S.-J. Kim, E. Dall'Anese, J. A. Bazerque, K. Rajawat, and G. B. Giannakis, "Advances in spectrum sensing and cross-layer design for cognitive radio networks," *E-Reference Signal Processing*, 2013, to be published.

[31] S.-J. Kim, E. Dall'Anese, and G. B. Giannakis, "Cooperative spectrum sensing for cognitive radios using Kriged Kalman filtering," *IEEE J. Select. Topics Signal Processing*, vol. 5, no. 1, pp. 24–36, Feb. 2011.

[32] T. King, S. Kopf, T. Haenselmann, C. Lubberger, and W. Effelsberg. (2008, Apr.). CRAWDAD data set mannheim/compass (v. 2008-04-11). [Online]. Available: http://crawdad.cs.dartmouth.edu/mannheim/compass

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

[34] A. Lakhina, M. Crovella, and C. Diot, "Diagnosing network-wide traffic anomalies," in *Proc. ACM SIGCOMM*, Portland, OR, Aug. 2004, pp. 219–230.

[35] S. Lee, Z. Zhang, S. Sahu, and D. Saha, "On suitability of Euclidean embedding Internet hosts," in *Proc. ACM SIGMETRICS*, Saint Malo, France, June 2006, pp. 157–168.

[36] Y. Liao, P. Geurts, and G. Leduc, "Network distance prediction based on decentralized matrix factorization," in *Proc. IFIP Networking Conf.*, Chennai, India, May 2010, pp. 15–26.

[37] J. Mairal, J. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *J. Mach. Learn. Res.*, vol. 11, pp. 19–60, Jan. 2010.

[38] M. Mardani, G. Mateos, and G. B. Giannakis, "Dynamic anomalography: Tracking network anomalies via sparsity and low rank," *IEEE J. Select. Topics Signal Processing*, vol. 7, no. 1, pp. 50–66, 2013.

[39] M. Mardani, G. Mateos, and G. B. Giannakis. (2012, March 8). "In-network sparsity-regularized rank minimization: Applications and algorithms," *IEEE Trans. Signal Processing*. [Online]. Available: http://arxiv.org/pdf/1203.1570v1.pdf

[40] M. Mardani, G. Mateos, and G. B. Giannakis, "Exact recovery of low-rank plus compressed sparse matrices," in *Proc. IEEE Statistical Signal Processing Workshop*, Ann Arbor, MI, Aug. 2012, pp. 49–52.

[41] K. V. Mardia, C. Goodall, E. J. Redfern, and F. J. Alonso, "The kriged Kalman filter," *Test*, vol. 7, no. 2, pp. 217–285, Dec. 1998.

[42] K. Myers and B. Tapley, "Adaptive sequential estimation with unknown noise statistics," *IEEE Trans. Autom. Contr.*, vol. 21, no. 4, pp. 520–523, Aug. 1976.

[43] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions: I," *Math. Program.*, vol. 14, no. 1, pp. 265–294, Dec. 1978.

[44] T. Oetiker. (2013). About RRDtool. [Online]. Available: http://oss.oetiker.ch/rrdtool/

[45] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng, "Self-taught learning: Transfer learning from unlabeled data," in *Proc. 24th Int. Conf. Machine Learning (ICML '07)*, pp. 759–766.

[46] K. Rajawat, E. Dall'Anese, and G. B. Giannakis, "Dynamic network kriging," in *Proc. IEEE Statistical Signal Processing Workshop*, Ann Arbor, MI, Aug. 2012, pp. 121–124.

[47] F. Niu, B. Recht, C. Re, and S. J, Wright, "HOGWILD!: A lock-free approach to parallelizing stochastic gradient descent," in *Proc. Advances in Neural Information Processing Systems*, Granada, Spain, pp. 1–7, Dec. 2011.

[48] M. Roughan, "A case study of the accuracy of SNMP measurements," *J. Electr. Comput. Eng.*, vol. 2010, paper 812979, pp. 1–7, 2010.

[49] Y. Shavitt, X. Sun, A. Wool, and B. Yener, "Computing the unmeasured: An algebraic approach to Internet mapping," *IEEE J. Select. Areas in Commun.*, vol. 22, no. 1, pp. 67–78, Jan. 2004.

[50] A. Soule, A. Lakhina, N. Taft, K. Papagiannaki, K. Salamatian, A. Nucci, M. Crovella, and C. Diot, "Traffic matrices: Balancing measurements, inference and modeling," in *Proc. ACM SIGMETRICS*, Banff, AB, Canada, June 2005.

[51] A. Soule, K. Salamatian, A. Nucci, and N. Taft, "Traffic matrix tracking using Kalman filters," *SIGMETRICS Perform. Eval. Rev.*, vol. 33, no. 3, pp. 24–31, Dec. 2005.

[52] I. Tošić and P. Frossard, "Dictionary learning," *IEEE Signal Processing Mag.*, vol. 28, pp. 27–38, Mar. 2010.

[53] X. Wu, K. Yu, and X. Wang, "On the growth of Internet application flows: A complex network perspective," in *Proc. IEEE Int. Conf. Computer Communications*, Shangai, China, June 2011, pp. 2096–2104.

[54] Y. Zhang, M. Roughan, C. Lund, and D. L. Donoho, "Estimating point-to-point and point-to-multipoint traffic matrices: An information-theoretic approach," *IEEE/ACM Trans. Networking*, vol. 13, no. 5, pp. 947–960, Oct. 2005.

[55] Y. Zhang, Z. Ge, A. Greenberg, and M. Roughan, "Network anomography," in *Proc. ACM SIGCOM Conf. Interent Measurements*, Berkeley, CA, Oct. 2005, pp. 317–330.

[56] M. Roughan, Y. Zhang, W. Willinger, and L. Qiu, "Spatio-temporal compressive sensing and Internet traffic matrices," *IEEE/ACM Trans. on Netw.*, vol. 20, no. 3, pp. 662–676, June 2012.

[SP]