Unveiling Anomalies in Large-scale Networks via Sparsity and Low Rank

Morteza Mardani, Gonzalo Mateos, and Georgios B. Giannakis
Dept. of ECE, University of Minnesota, Minneapolis, MN 55455, USA
Emails: {morteza,mate0058,georgios}@umn.edu

***Abstract*—In the backbone of large-scale networks, traffic flows experience abrupt unusual changes which can result in congestion, and limit the extent to which end-user quality of service requirements are met. Diagnosing such *traffic volume anomalies* is a crucial task towards engineering the traffic in the network. This is challenging however, since the available data are the superposition of *unobservable* origin-to-destination (OD) flows per link. Leveraging the low intrinsic-dimensionality of OD flows and the *sparse* nature of anomalies, a convex program is formulated to unveil anomalies across flows and time. A centralized solver is developed using the proximal gradient algorithm, which offers provable iteration complexity guarantees. An equivalent nonconvex but separable criterion enables *in-network* processing of link-load measurements, when optimized via the alternating-direction method of multipliers. The novel distributed iterations entail reduced-complexity local tasks, and affordable message passing between neighboring nodes. Interestingly, under mild conditions the distributed algorithm approaches its centralized counterpart. Numerical tests with synthetic and real network data corroborate the effectiveness of the novel scheme.**

I. Introduction

In the backbone of large-scale networks, origin-to-destination (OD) traffic flows experience abrupt unusual changes which can result in congestion, and limit the quality of service provisioning of the end users. These so-termed *traffic volume anomalies* could be due to external sources such as network failures, denial of service attacks, or, intruders which hijack the network services [12]. Unveiling such anomalies is a crucial task towards engineering network traffic. This is a challenging task however, since the available data are usually high-dimensional noisy link-load measurements, which are the superposition of *unobservable* OD flows.

Several studies have demonstrated the low intrinsic dimensionality of the traffic matrix, which is mainly due to common temporal patterns across OD flows, and periodic behaviors across time [7]. Exploiting the low-rank structure of the anomaly-free traffic matrix, a principal component analysis (PCA)-based method was put forth in [7] to identify network anomalies. PCA-based methods require knowledge of the rank of the traffic matrix, and face scalability issues [6]. A sequential detection-based approach was proposed in [4]. Most importantly, [7] and [4] have not exploited the *sparsity* of anomalies across flows and time – anomalous traffic spikes are rare, and tend to last for short periods of time relative to the measurement horizon. In a nutshell, state-of-the-art anomaly detection algorithms rely on *central processing*, are unable to *identify* anomalous flows, and do not capitalize on the sparsity present.

In this context, the fresh look advocated here permeates benefits from rank minimization and compressive sensing to unveil traffic volume anomalies in large-scale networks. Nuclear norm and ℓ_1-norm regularization have been widely adopted as convex surrogates to the rank and ℓ_0-norm, respectively, in a variety of problems pertaining to low-rank matrix completion and variable selection. Recently, these ideas were applied to split a given data matrix into its sparse and low-rank components, with remarkable performance guarantees; see e.g., [2] and [5].

Inspired by the features of OD flow traffic and anomalies as well as recent advances in compressive sensing and rank minimization, the goal of this paper is to efficiently and accurately diagnose network anomalies in a *distributed* fashion. To this end, a convex estimator is proposed which optimizes the trade-off between data fit, (low-) rank of the traffic matrix, and the sparsity of the anomalies across flows and time. A centralized solver is developed using the proximal gradient algorithm, which offers provable iteration complexity guarantees [10], [8]. An equivalent nonconvex but separable criterion enables *in-network* processing of link-load measurements, when optimized via the alternating-direction method of multipliers (AD-MoM) [1]. The novel distributed iterations entail reduced-complexity local tasks, and affordable message passing among neighboring nodes. Interestingly, under mild conditions the distributed algorithm approaches its centralized counterpart. Insightful tests with synthetic and real network data corroborate the effectiveness of the novel scheme, and their ability to outperform existing alternatives.

II. Preliminaries and Problem Statement

Consider a backbone Internet protocol (IP) network with topology represented by the directed graph $G(\mathcal{N}, \mathcal{L})$, where $\mathcal{L}$ and $\mathcal{N}$ denote the set of links and nodes (routers) of cardinality $|\mathcal{L}| = L$ and $|\mathcal{N}| = N$, respectively. In this network a set of OD traffic flows $\mathcal{F}$ with $|\mathcal{F}| = F$ traverse the links connecting different source-destination pairs. For backbone networks, the number of network layer flows is much larger than the number of physical links ($F \gg L$). Single-path routing is considered to deliver the traffic flow from a source to its intended destination. Accordingly, for a particular flow multiple links connecting the corresponding source-destination

† This work was supported by MURI (AFOSR FA9550-10-1- 0567) grant.

pair are chosen to carry the traffic. Let $\{r_{l,f}\}_{l\in\mathcal{L}}^{f\in\mathcal{F}}$ denote the flow to link assignments (routing variables), which take the value one whenever flow f goes through link l, and zero otherwise. The routing matrix $\mathbf{R} := [r_{l,f}] \in \mathbb{R}^{L\times F}$ is assumed fixed and given. Likewise, let $x_{f,t}$ denote the unknown traffic rate of flow f at time t, measured in e.g., Mbps. The traffic carried over link l is then the superposition of the flow rates routed through link l, i.e., $\sum_{f\in\mathcal{F}} r_{l,f}x_{f,t}$.

It is not uncommon for some of the OD flows to experience unusual sudden changes. Let $a_{f,t}$ denote the unknown traffic volume anomaly of flow f at time t. In the presence of anomalous flows, the measured link-layer traffic over link ℓ at time t is given by

$$y_{l,t} = \sum_{f\in\mathcal{F}} r_{l,f}(x_{f,t} + a_{f,t}) + v_{l,t} \tag{1}$$

where $v_{l,t}$ accounts for noise and unmodeled dynamics. Collecting T measurements and introducing matrices $\mathbf{Y} := [y_{l,t}], \mathbf{V} := [v_{l,t}] \in \mathbb{R}^{L\times T}$, and $\mathbf{X} := [x_{f,t}], \mathbf{A} := [a_{f,t}] \in \mathbb{R}^{F\times T}$, the matrix model across T time slots is [cf. (1)]

$$\mathbf{Y} = \mathbf{R}\,(\mathbf{X} + \mathbf{A}) + \mathbf{V}. \tag{2}$$

Common temporal patterns among the traffic flows in addition to their periodic behavior, render the traffic matrix $\mathbf{X}$ typically low-rank [7]. Anomalies are expected to occur sporadically over time, and only last for short periods relative to the (possibly long) measurement period T. In addition, only a small fraction of the flows are supposed to be anomalous at a any given time instant. This renders the anomaly matrix $\mathbf{A}$ sparse across both rows and columns. Given measurements $\mathbf{Y}$ and the binary-valued routing matrix $\mathbf{R}$, the primary goal of this paper is to accurately estimate the anomaly matrix $\mathbf{A}$, leveraging the sparsity and low-rank attributes of $\mathbf{A}$ and $\mathbf{X}$. Upon forming the estimate $\hat{\mathbf{A}}$, if $|\hat{a}_{f,t}| > 0$ the fth flow at time t is declared anomalous.

III. Centralized Approach

In a different context, the problem of recovering $\mathbf{A}$ from the observations modeled in (2) when $\mathbf{R} = \mathbf{I}_F$ ($\mathbf{I}_F$ denotes the $F \times F$ identity matrix) has been investigated in [2] and [5]. However, in the presence of the fat routing matrix $\mathbf{R}$ the recovery task becomes more challenging, since the null space of $\mathbf{R}$ compromises identifiability of $\mathbf{X}$ and $\mathbf{A}$. For instance, if any of the matrices $\mathbf{X}$ and $\mathbf{A}$ lies in the null space of $\mathbf{R}$, there is no chance of accurate estimation.

Inspired by the recent results in compressive sensing, there is hope to recover a sufficiently sparse $\mathbf{A}$; see e.g., [3]. Since the primary goal is to recover $\mathbf{A}$, define $\mathbf{X}_r := \mathbf{R}\mathbf{X}$ which inherits the low-rank property from $\mathbf{X}$, and consider [cf. (2)]

$$\mathbf{Y} = \mathbf{X}_r + \mathbf{R}\mathbf{A} + \mathbf{V}. \tag{3}$$

Notice that $\mathbf{R}\mathbf{A}$ is not necessarily sparse even though $\mathbf{A}$ is a sparse matrix. To find the estimates $(\hat{\mathbf{X}}_r, \hat{\mathbf{A}})$, the following *convex* optimization problem is formulated

$$\text{(P1)} \quad \min_{(\mathbf{X}_r,\mathbf{A})} \frac{1}{2}||\mathbf{Y} - \mathbf{X}_r - \mathbf{R}\mathbf{A}||_F^2 + \lambda_*||\mathbf{X}_r||_* + \lambda_1||\mathbf{A}||_1 \tag{4}$$

where $\|\mathbf{X}_r\|_* := \sum_i \sigma_i(\mathbf{X}_r)$ denotes the nuclear norm, and $\|\mathbf{A}\|_1 := \sum_f \sum_t |a_{f,t}|$ is the ℓ_1-norm of matrix $\mathbf{A}$. The regularization terms $||\mathbf{X}_r||_*$ and $||\mathbf{A}||_1$ promote the low-rank property of $\mathbf{X}_r$ and the sparsity of $\mathbf{A}$, respectively. The corresponding tuning parameters λ_* and λ_1 control the rank and sparsity levels in $(\hat{\mathbf{X}}_r, \hat{\mathbf{A}})$. A centralized algorithm to solve (P1) is discussed next.

A. Accelerated proximal gradient-descent algorithm

Accelerated proximal gradient (APG) algorithms were originally studied in [10], and have been recently applied to matrix-valued problems under the name (stable) principal components pursuit (PCP) [13], [8]. APG algorithms offer several attractive features, most notably a convergence rate guarantee of $O(1/\sqrt{\epsilon})$ iterations to return an $\epsilon-$optimal solution. In addition, APG algorithms are first-order methods that scale nicely to high-dimensional problems arising with large networks.

The algorithm to be developed here extends the one in [8], proposed to solve the stable PCP problem (P1) with $L = F$ and $\mathbf{R} = \mathbf{I}_L$. For the matrix $\mathbf{S} := [\mathbf{X}_r', \ \mathbf{A}']'$, define

$$f(\mathbf{S}) := \frac{1}{2}\|\mathbf{Y} - \mathbf{X}_r - \mathbf{R}\mathbf{A}\|_F^2, \quad g(\mathbf{S}) := \lambda_*\|\mathbf{X}_r\|_* + \lambda_1\|\mathbf{A}\|_1.$$

Instead of directly optimizing the cost in (4), APG algorithms minimize a sequence of overestimates of $f(\mathbf{S}) + g(\mathbf{S})$, obtained at judiciously chosen points $\mathbf{T}$. A Taylor approximation around $\mathbf{T}$ yields the following upper-bound of the cost

$$Q(\mathbf{S}, \mathbf{T}) := \frac{L_f}{2}\|\mathbf{S} - \mathbf{G}\|_F^2 + g(\mathbf{S}) + f(\mathbf{T}) - \frac{1}{2L_f}\|\nabla f(\mathbf{T})\|_F^2$$

where $\mathbf{G} := \mathbf{T} - L_f^{-1}\nabla f(\mathbf{T})$ and $L_f := \lambda_{\max}([\mathbf{I}_L \ \mathbf{R}]'[\mathbf{I}_L \ \mathbf{R}])$ is a Lipschitz constant for $\nabla f(\mathbf{S})$. With $k = 1, 2, \ldots$ indexing iterations, APG algorithms generate the sequence of iterates

$$\mathbf{Z}[k] := \arg\min_{\mathbf{S}} Q(\mathbf{S}, \mathbf{T}[k]). \tag{5}$$

There are two key aspects to the success of APG algorithms. First is the selection of points $\mathbf{T}[k]$ where the sequence of approximations $Q(\mathbf{S}, \mathbf{T}[k])$ are formed, since these strongly determine the convergence rate. The choice $\mathbf{T}[k] = \mathbf{Z}[k] + \frac{t[k-1]-1}{t[k]}(\mathbf{Z}[k] - \mathbf{Z}[k-1])$, where $t[k] = \left[1 + \sqrt{4t^2[k-1]+1}\right]/2$, considerably accelerates the algorithm [10]. The second key element stems from the possibility of efficiently solving the sequence of subproblems (5). For the particular case of (P1), note that (5) decomposes into

$$\mathbf{X}_r[k+1] := \arg\min_{\mathbf{X}_r}\left\{\frac{L_f}{2}\|\mathbf{X}_r - \mathbf{G}_X[k]\|_F^2 + \lambda_*\|\mathbf{X}_r\|_*\right\}$$

$$\mathbf{A}[k+1] := \arg\min_{\mathbf{A}}\left\{\frac{L_f}{2}\|\mathbf{A} - \mathbf{G}_A[k]\|_F^2 + \lambda_1\|\mathbf{A}\|_1\right\}$$

where $\mathbf{G}[k] := [\mathbf{G}_X'[k] \ \mathbf{G}_A'[k]]'$. Letting $\mathcal{S}_\tau(\mathbf{M}) := \text{sign}(\mathbf{M})\max(|\mathbf{M}| - \tau, \mathbf{0})$ denote the soft-thresholding operator, and $\mathbf{U}\mathbf{\Sigma}\mathbf{V}' = \text{svd}(\mathbf{G}_X[k])$ the singular value decomposition of matrix $\mathbf{G}_X[k]$, it follows that (see, e.g. [8])

$$\mathbf{X}_r[k+1] = \mathbf{U}\mathcal{S}_{\lambda_*/L_f}[\mathbf{\Sigma}]\mathbf{V}', \quad \mathbf{A}[k+1] = \mathcal{S}_{\lambda_1/L_f}[\mathbf{G}_A[k]].$$

Algorithm 1 : APG solver for (P1)

input $\mathbf{Y}, \mathbf{R}, \lambda_*, \lambda_1, L_f = \lambda_{\max}([\mathbf{I}_L \ \mathbf{R}]'[\mathbf{I}_L \ \mathbf{R}])$
initialize $\mathbf{X}_r[0] = \mathbf{X}_r[-1] = \mathbf{0}_{L\times T}$, $\mathbf{A}[0] = \mathbf{A}[-1] = \mathbf{0}_{F\times T}$, $t[0] = t[-1] = 1$, and set $k = 0$.
while not converged **do**
$\mathbf{T}_X[k] = \mathbf{X}_r[k] + \frac{t[k-1]-1}{t[k]}(\mathbf{X}_r[k] - \mathbf{X}_r[k-1])$.
$\mathbf{T}_A[k] = \mathbf{A}[k] + \frac{t[k-1]-1}{t[k]}(\mathbf{A}[k] - \mathbf{A}[k-1])$.
$\mathbf{G}_X[k] = \mathbf{T}_X[k] + \frac{1}{L_f}(\mathbf{Y} - \mathbf{T}_X[k] - \mathbf{R}\mathbf{T}_A[k])$.
$\mathbf{G}_A[k] = \mathbf{T}_A[k] + \frac{1}{L_f}\mathbf{R}'(\mathbf{Y} - \mathbf{T}_X[k] - \mathbf{R}\mathbf{T}_A[k])$.
$\mathbf{U}\mathbf{\Sigma}\mathbf{V}' = \text{svd}(\mathbf{G}_X[k])$, $\mathbf{X}_r[k+1] = \mathbf{U}\mathcal{S}_{\lambda_*/L_f}(\mathbf{\Sigma})\mathbf{V}'$.
$\mathbf{A}[k+1] = \mathcal{S}_{\lambda_1/L_f}(\mathbf{G}_A[k])$.
$t[k+1] = \left[1 + \sqrt{4t^2[k]+1}\right]/2$
$k \leftarrow k+1$
end while
return $\mathbf{X}_r[k]$, $\mathbf{A}[k]$

The APG algorithm for unveiling network anomalies is tabulated under Algorithm 1. Iterations terminate whenever the distance between the origin and the set of subgradients of the cost in (4), evaluated at $\mathbf{X}[k+1]$, is below a prescribed threshold [8].

Implementing Algorithm 1 presumes that network nodes communicate their local link traffic measurements to a central processing unit, which uses their aggregation in $\mathbf{Y}$ to determine network anomalies. Collecting all this information can be challenging though, or even impossible in e.g., wireless sensor networks operating under stringent power budget constraints. Performing the optimization in a centralized fashion raises robustness concerns as well, since the central node carrying out the specific task at hand represents an isolated point of failure. These reasons motivate devising *distributed* algorithms for unveiling anomalies in large scale networks, whereby each node carries out simple computational tasks locally, relying only on its local measurements and messages exchanged with its directly connected neighbors. This is the subject dealt with next.

IV. Distributed Approach

For node $n \in \mathcal{N}$, let $\mathcal{J}_n$ denote the set of its single-hop neighbors, where $m \in \mathcal{J}_n$ if and only if $n \in \mathcal{J}_m$. Let also $\mathcal{L}_{\text{out}}(n)$ denote the set of outgoing links, where $\mathcal{L}_{\text{out}}(n) \cap \mathcal{L}_{\text{out}}(m) = \emptyset$ for $m \neq n$, and $\mathcal{L} = \cup_{n\in\mathcal{N}}\mathcal{L}_{\text{out}}(n)$. Node $n \in \mathcal{N}$ can measure the traffic rate of its outgoing links $\mathcal{L}_{\text{out}}(n)$ over a time interval of length T. These local measurements are collected in $\mathbf{Y}_n := [y_{\ell,t}]_{\ell\in\mathcal{L}_{\text{out}}(n), t\in[1,T]}$. Likewise, node n's local routing table is denoted by $\mathbf{R}_n := [r_{\ell,f}]_{\ell\in\mathcal{L}_{\text{out}}(n), f\in\mathcal{F}}$, which indicates the end-to-end flows carried by the outgoing links in $\mathcal{L}_{\text{out}}(n)$. Observe that both $\mathbf{Y}_n$ and $\mathbf{R}_n$ correspond to a subset of the rows of the global matrices $\mathbf{Y}$ and $\mathbf{R}$, respectively. Specifically, it is possible to write $\mathbf{Y} = \mathbf{\Pi}[\mathbf{Y}_1', \ldots, \mathbf{Y}_N']'$, where $\mathbf{\Pi}$ is a suitably chosen row permutation matrix, and likewise for $\mathbf{R}$. In this context, the problem addressed in this section is: given $\mathbf{Y}_n$ and $\mathbf{R}_n$ per node $n \in \mathcal{N}$, and under the constraint of local communications within neighborhoods $\mathcal{J}_n$, how can one *efficiently* solve (P1) in a *distributed* fashion? The main issue is for each network node to form its own estimate $\hat{\mathbf{A}}_n \in \mathbb{R}^{F\times T}$ of the anomalies, across all flows and measurement time instants. Moreover, local estimates should consent on the global optimum solution of (P1), that is $\hat{\mathbf{A}} = \hat{\mathbf{A}}_1 = \ldots = \hat{\mathbf{A}}_N$.

To facilitate reducing computational complexity and memory storage requirements of the distributed algorithm sought, it is henceforth assumed that an upper bound $\text{rank}(\hat{\mathbf{X}}_r) \leq \rho$ is known a priori, where $\hat{\mathbf{X}}_r$ is the estimated traffic matrix obtained via (P1). Because $\text{rank}(\hat{\mathbf{X}}_r) \leq \rho$, (P1)'s search space is effectively reduced and one can factorize $\mathbf{X}_r = \mathbf{L}\mathbf{Q}'$, where $\mathbf{L}$ and $\mathbf{Q}$ are $L \times \rho$ and $T \times \rho$ matrices, respectively. Adopting this reparametrization of $\mathbf{X}_r$ in (P1) and defining $r_n(\mathbf{L}_n, \mathbf{Q}, \mathbf{A}) := \frac{1}{2}\|\mathbf{Y}_n - \mathbf{L}_n\mathbf{Q}' - \mathbf{R}_n\mathbf{A}\|_F^2$ one obtains the following equivalent optimization problem

$$\text{(P3)} \quad \min_{\{\mathbf{L},\mathbf{Q},\mathbf{A}\}} \sum_{n=1}^{N}\left\{r_n(\mathbf{L}_n, \mathbf{Q}, \mathbf{A}) + \frac{\lambda_*}{N}\|\mathbf{L}\mathbf{Q}'\|_* + \frac{\lambda_1}{N}\|\mathbf{A}\|_1\right\}.$$

Note that (P3) is non-convex due to the bilinear term $\mathbf{L}_n\mathbf{Q}'$, where $\mathbf{L} := \mathbf{\Pi}[\mathbf{L}_1', \ldots, \mathbf{L}_N']'$. However, the number of variables is reduced from $LT + FT$ in (P1), to $\rho(L+T) + FT$ in (P3). The savings can be significant when ρ is in the order of a few dozens and both L and T are large. Also note that the dominant FT-term in the variable count of (P3) is due to $\mathbf{A}$, which is sparse and can be efficiently handled even when both F and T are large.

A. A separable regularization

Problem (P3) is still not amenable for distributed implementation due to: (i) the non-separable nuclear norm present in the cost function; and (ii) the global variables $\mathbf{Q}$ and $\mathbf{A}$ coupling the per-node summands. To address (i), consider the following separable characterization of the nuclear norm [11]

$$\|\mathbf{X}\|_* := \min_{\{\mathbf{L},\mathbf{Q}\}} \frac{1}{2}\left\{\|\mathbf{L}\|_F^2 + \|\mathbf{Q}\|_F^2\right\}, \quad \text{s. to} \quad \mathbf{X} = \mathbf{L}\mathbf{Q}'. \quad (6)$$

The optimization (6) is over all possible bilinear factorizations of $\mathbf{X}$, so that the number of columns of $\mathbf{L}$ and $\mathbf{Q}$ is not treated as fixed. Building on (6), the following reformulation of (P3) provides an important first step towards obtaining a distributed estimator for unveiling network anomalies ($u_n(\mathbf{L}_n, \mathbf{Q}) := N\|\mathbf{L}_n\|_F^2 + \|\mathbf{Q}\|_F^2$)

$$\text{(P4)} \min_{\{\mathbf{L},\mathbf{Q},\mathbf{A}\}} \sum_{n=1}^{N}\left\{r_n(\mathbf{L}_n, \mathbf{Q}, \mathbf{A}) + \frac{\lambda_*}{2N}u_n(\mathbf{L}_n, \mathbf{Q}) + \frac{\lambda_1}{N}\|\mathbf{A}\|_1\right\}.$$

As asserted in the following lemma, adopting the separable regularization in (P4) comes with no loss of optimality, provided the upper bound ρ is chosen large enough.[1]

Lemma 1: *If $(\hat{\mathbf{X}}_r, \hat{\mathbf{A}})$ denotes the minimizer of (P1) and $\text{rank}(\hat{\mathbf{X}}_r) \leq \rho$, then (P4) is equivalent to (P1).*

Lemma 1 implies that by finding the global minimum of (P4) [which could have considerably less variables than (P1)], one can recover the optimal solution of (P1). However, since

[1]Proofs are omitted here due to space limitation; see [9].

(P4) is non-convex, it may have stationary points which need not be globally optimum. Interestingly, the next proposition establishes that under relatively mild assumptions, every stationary point of (P4) is globally optimum for (P1).

Proposition 1: *If* $(\bar{\mathbf{L}}, \bar{\mathbf{Q}}, \bar{\mathbf{A}})$ *is a stationary point of (P4), and* $\|\mathbf{Y}-\bar{\mathbf{L}}\bar{\mathbf{Q}}'-\mathbf{R}\bar{\mathbf{A}}\|_2 \leq \lambda_*/2$ *holds, then* $(\hat{\mathbf{X}} := \bar{\mathbf{L}}\bar{\mathbf{Q}}', \hat{\mathbf{A}} := \bar{\mathbf{A}})$ *is the optimal solution of (P1).*

Notice that for given λ_*, the condition $\|\mathbf{Y}-\bar{\mathbf{L}}\bar{\mathbf{Q}}'-\mathbf{R}\bar{\mathbf{A}}\|_2 \leq \lambda_*/2$ implicitly lower bounds ρ and upper bounds the noise variance. To decompose the cost in (P4), in which summands are coupled through the global variables $\mathbf{Q}$ and $\mathbf{A}$ [cf. (ii) at the beginning of this section], introduce auxiliary variables $\{\mathbf{Q}_n, \mathbf{A}_n\}_{n=1}^N$ representing local estimates of $\{\mathbf{Q}, \mathbf{A}\}$ per node n. These local estimates are utilized to form the separable *constrained* minimization problem (P5)

$$\min_{\substack{\{\mathbf{L},\mathbf{Q}_n\}\\ \{\mathbf{A}_n,\mathbf{B}_n\}}} \sum_{n=1}^{N}\left\{ r_n(\mathbf{L}_n,\mathbf{Q}_n,\mathbf{B}_n) + \frac{\lambda_*}{2N}u_n(\mathbf{L}_n,\mathbf{Q}_n) + \frac{\lambda_1}{N}\|\mathbf{A}_n\|_1 \right\}$$
$$\text{s. to} \quad \mathbf{B}_n = \mathbf{A}_n, \quad n \in \mathcal{N}$$
$$\mathbf{Q}_n = \mathbf{Q}_m, \quad \mathbf{A}_n = \mathbf{A}_m, \quad m \in \mathcal{J}_n,\ n \in \mathcal{N}.$$

As will be clear later, additional variables $\{\mathbf{B}_n\}_{n=1}^N$ were introduced to split the ℓ_2-norm fitting-error part of the cost in (P5), from the ℓ_1-norm regularization on the $\mathbf{A}_n$'s. The set of additional constraints $\mathbf{B}_n = \mathbf{A}_n$ ensures that, in this sense, nothing changes in going from (P4) to (P5). Most importantly, (P4) and (P5) are equivalent optimization problems provided the network graph $G(\mathcal{N}, \mathcal{L})$ is connected.

B. The alternating-direction method of multipliers

To minimize (P5) in a distributed fashion, a variation of the alternating-direction method of multipliers (AD-MoM) will be adopted here. Accordingly, the constraints in (P5) are dualized through the dual variables collected in $\mathcal{M} := \{\mathbf{M}_n, \mathbf{P}_n, \mathbf{O}_n\}_{n=1}^N$. The weighted ℓ_2-norms of the constraints are also added to the cost as a penalty, to form the augmented Lagrangian. The primal variables are then split in three groups $\mathcal{V}_1 := \{\mathbf{Q}_n, \mathbf{A}_n\}_{n=1}^N$, $\mathcal{V}_2 := \{\mathbf{L}_n\}_{n=1}^N$, and $\mathcal{V}_3 := \{\mathbf{B}_n\}_{n=1}^N$. The AD-MoM iteratively minimizes the augmented Lagrangian using a block-coordinate descent scheme that cycles through $\mathcal{V}_1 \rightarrow \mathcal{V}_2 \rightarrow \mathcal{V}_3$, with additional dual variable updates.

Reformulating the estimator (P1) to its equivalent form (P5) renders the augmented Lagrangian highly decomposable [9]. The separability comes in two flavors, both with respect to the variable groups $\mathcal{V}_1$, $\mathcal{V}_2$, and $\mathcal{V}_3$, as well as across the network nodes $n \in \mathcal{N}$. This in turn leads to highly parallelized, simplified recursions. Specifically, it is shown in [9] that if the multipliers are initialized to zero, one arrives at the distributed iterations tabulated as Algorithm 2.

The main computational burden stems from repeated inversions of (small) $\rho \times \rho$ matrices, and simple soft-thresholding operations. Conceivably the number of flows F can be large, thus inverting the $F \times F$ matrix $\mathbf{R}_n'\mathbf{R}_n + c\mathbf{I}_F$ to update $\mathbf{B}_n[k]$ could be computationally demanding. Fortunately, the inversion needs to be carried out once, and can be performed off-line.

Algorithm 2 : AD-MoM solver at node $n \in \mathcal{N}$

input $\mathbf{Y}_n, \mathbf{R}_n, \lambda_*, \lambda_1, c$
initialize $\mathbf{M}_n[0] = \mathbf{P}_n[0] = \mathbf{A}_n[1] = \mathbf{B}_n[1] = \mathbf{0}_{F\times T}$, $\mathbf{O}[0] = \mathbf{0}_{T\times\rho}$, and $\mathbf{L}_n[1]$, $\mathbf{Q}_n[1]$ at random.
for $k = 1, 2, \ldots$ **do**
Receive $\{\mathbf{Q}_m[k], \mathbf{A}_m[k]\}$ from neighbors $m \in \mathcal{J}_n$
Step 1) update the dual variables
$\mathbf{M}_n[k] = \mathbf{M}_n[k-1] + c(\mathbf{B}_n[k] - \mathbf{A}_n[k])$
$\mathbf{O}_n[k] = \mathbf{O}_n[k-1] + c\sum_{m\in\mathcal{J}_n}(\mathbf{Q}_n[k] - \mathbf{Q}_m[k])$
$\mathbf{P}_n[k] = \mathbf{P}_n[k-1] + c\sum_{m\in\mathcal{J}_n}(\mathbf{A}_n[k] - \mathbf{A}_m[k])$
Step 2) update the primal variables
$\mathbf{Q}_n[k+1] = [\mathbf{L}_n'[k]\mathbf{L}_n[k] + (\lambda_*/N + 2c)\mathbf{I}_\rho]^{-1}\{\mathbf{Y}'_n\mathbf{L}_n[k] - \mathbf{B}_n'[k]\mathbf{R}_n'\mathbf{L}_n[k] - \mathbf{O}_n[k] + c\sum_{m\in\mathcal{J}_n}(\mathbf{Q}_n[k] + \mathbf{Q}_m[k])\}$
$\mathbf{E}_n[k] = \mathbf{M}_n[k] + c\mathbf{B}_n[k] - \mathbf{P}_n[k] + c\sum_{m\in\mathcal{J}_m}(\mathbf{A}_n[k] + \mathbf{A}_m[k])$
$\mathbf{A}_n[k+1] = [c(1+2|\mathcal{J}_n|)]^{-1}\mathcal{S}_{\lambda_1/N}(\mathbf{E}_n[k]))$
$\mathbf{I}_n[k+1] = [\mathbf{Q}_n'[k+1]\mathbf{Q}_n[k+1] + \lambda_*\mathbf{I}_\rho]^{-1}$
$\mathbf{L}_n[k+1] = (\mathbf{Y}_n - \mathbf{R}_n\mathbf{B}_n[k])\,\mathbf{Q}_n[k+1]\mathbf{I}_n[k+1]$
$\mathbf{B}_n[k+1] = [\mathbf{R}_n'\mathbf{R}_n + c\mathbf{I}_F]^{-1} \times \{\mathbf{R}_n'(\mathbf{Y}_n - \mathbf{L}_n[k+1]\mathbf{Q}_n'[k+1]) - \mathbf{M}_n[k] + c\mathbf{A}_n[k+1]\}$
Broadcast $\{\mathbf{Q}_n[k+1], \mathbf{A}_n[k+1]\}$ to neighbors $m \in \mathcal{J}_n$
end for
return $\mathbf{A}_n, \mathbf{Q}_n, \mathbf{L}_n$

On a per iteration basis, network nodes communicate their updated local estimates $\{\mathbf{Q}_n[k], \mathbf{A}_n[k]\}$ with their neighbors, in order to carry out updates of the primal and dual variables during the next iteration. Regarding communication cost, $\mathbf{Q}_n[k]$ is a $T \times \rho$ matrix and its transmission does not incur significant overhead for small values of ρ. In addition, the $F \times T$ matrix $\mathbf{A}_n[k]$ is sparse, and can be communicated efficiently. Note that the dual variables need not be exchanged.

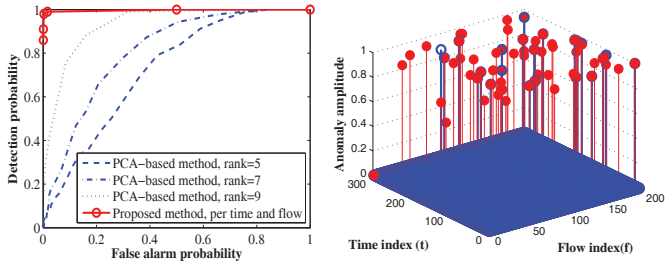
While a formal convergence analysis is beyond the scope of this work, the following proposition proved in [9] asserts that upon convergence, Algorithm 2 achieves global optimality.

Proposition 2: *If the iterates generated by Algorithm 2 converges to* $\{\bar{\mathbf{Q}}_n, \bar{\mathbf{L}}_n, \bar{\mathbf{A}}_n\}_{n\in\mathcal{N}}$, *and* $\|\mathbf{Y} - \bar{\mathbf{L}}\bar{\mathbf{Q}}_1' - \mathbf{R}\bar{\mathbf{A}}_1\|_2 \leq \lambda_*/2$ *holds, then: i)* $\bar{\mathbf{Q}}_i = \bar{\mathbf{Q}}_j$, $\bar{\mathbf{A}}_i = \bar{\mathbf{A}}_j$, $\forall i, j \in \mathcal{N}$; *and ii)* $\hat{\mathbf{A}} = \bar{\mathbf{A}}_1$ *and* $\hat{\mathbf{X}}_r = \bar{\mathbf{L}}\bar{\mathbf{Q}}_1'$, *where* $(\hat{\mathbf{A}}, \hat{\mathbf{X}}_r)$ *is the global optimum of (P1).*

V. NUMERICAL TESTS

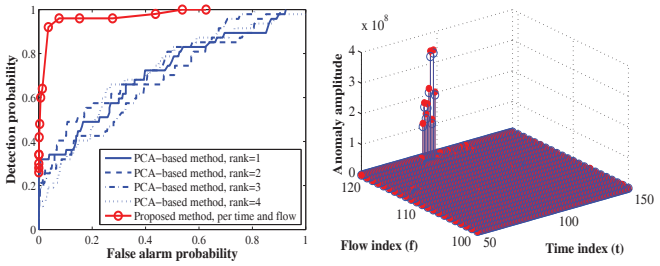
Performance of the proposed estimator is assessed in this section via numerical tests using both synthetic and real network data. To generate synthetic data, network topologies with randomly placed nodes are simulated. For each candidate OD pair, shortest-path routes are considered to form $\mathbf{R}$. The i.i.d. entries of matrix $\mathbf{V}$ are zero-mean Gaussian distributed, with variance σ^2. Low-rank matrices are generated as $\mathbf{X}_0 = \mathbf{W}_1\mathbf{W}_2'$, where $\mathbf{W}_1 \in \mathbb{R}^{F\times r}$ and $\mathbf{W}_2 \in \mathbb{R}^{T\times r}$ contain i.i.d. zero-mean Gaussian entries with variance $10\,\sigma/\sqrt{FT}$. Every entry of $\mathbf{A}_0$ is picked randomly from the set $\{-1, 0, 1\}$ with $\Pr(a_{i,j} = -1) = \Pr(a_{i,j} = 1) = \rho/2$.

Real OD flow data are collected from the operations of the Abilene network (backbone of the Internet 2 protocol) during



(a) (b)

Fig. 1. Performance for synthetic data. (a) ROC curves of the proposed versus the PCA-based method with $\rho = 0.001$, $\sigma^2 = 0.01$, $N = 20$, $L = 108$, $F = 360$, $T = 760$. (b) Amplitude of the true and estimated anomalies for $P_f = 10^{-4}$ and $P_d = 0.97$. Lines with open and filled circle markers denote the true and estimated anomalies, respectively.



(a) (b)

Fig. 2. Performance for Abilene network data. (a) ROC curves of the proposed versus the PCA-based method. (b) Amplitude of the true and estimated anomalies for $P_f = 0.04$ and $P_d = 0.93$.

Dec. 8–28, 2008 [7]. The link loads in $\mathbf{Y}$ are obtained based on (2) and the Abilene routing matrix. The available OD flows are a superposition of 'clean' and anomalous traffic, i.e., $\mathbf{X}_0 + \mathbf{A}_0$, and thus the anomalies are not recognizable. However, this data is the sum of low rank plus sparse matrices which respects the considered model in (2) when $\mathbf{R} = \mathbf{I}_F$. Therefore, the proposed algorithms are applied to find a reasonably precise estimate of the "ground-truth" traffic and anomaly matrices.

A. Comparison with a PCA-based method

To highlight the merits of the proposed anomaly detection algorithm, its performance is compared with the PCA-based approach of [7]. The crux of this method is that the anomaly-free data is expected to be low-rank, whereas the presence of anomalies considerably increases the rank of $\mathbf{Y}$. PCA requires a priori knowledge of the rank of the anomaly-free traffic matrix, and is unable to identify anomalous flows, i.e., the scope of [7] is limited to a single anomalous flow per time slot. Different from [7], the developed framework here enables identifying multiple anomalous flows per time instant. To assess performance, the detection rate will be used as figure of merit, which measures the algorithm's success in identifying anomalies across both flows and time instants.

For the synthetic data case, ROC curves are depicted in Fig. 1 (a), for different values of the rank required to run the PCA-based method. It is apparent that the proposed scheme detects accurately the anomalies, even at low false alarm rates. For the particular case of $P_f = 10^{-4}$, $P_d = 0.97$, Fig. 1 (b) illustrates the magnitude of the true and estimated anomalies across flows and time. Similar results are depicted for the Abilene data in Fig. 2, where it is also evident that proposed method markedly outperforms PCA in terms of detection performance. For an instance of $P_f = 0.03$ and $P_d = 0.92$, Fig. 2 (b) shows the effectiveness of the proposed algorithm in terms of unveiling the anomalous flows and time instants.

VI. Conclusions and Future Work

This paper introduced an efficient algorithm for in-network unveiling of anomalies present in OD flows. To this end, an estimator resulting from a convex problem was developed first using the proximal gradient method. For in-network operation a distributed algorithm was also developed based on the AD-MOM method which can afford low computational complexity. Interestingly, the distributed estimator can attain the centralized performance.

The ongoing research includes: i) studying the exact recovery performance of the proposed estimator in the absence of noise; and ii) extending the distributed approach to the general matrix completion problem.

References

[1] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, 2nd ed. Athena-Scientific, 1999.

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

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

[4] P. Casas, S. Vaton, L. Fillatre, and I. Nikiforov, "Optimal volume anomaly detection and isolation in large-scale ip networks using coarse-grained measurements," *Elsevier Computer Networks*, vol. 54, pp. 1750–1766, Aug. 2010.

[5] 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.

[6] H. Kim, S. Lee, X. Ma, and C. Wang, "Higher-order PCA for anomaly detection in large-scale networks," in *Proc. of IEEE Int. Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, Aruba, Dutch Antilles, Dec. 2009, pp. 85 – 88.

[7] A. Lakhina, M. Crovella, and C. Diot, "Diagnosing network-wide traffic anomalies," in *Proc. of ACM SIGCOMM*, Portland, OR, Aug. 2004.

[8] Z. Lin, A. Ganesh, J. Wright, L. Wu, M. Chen, and Y. Ma, "Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix," UIUC Technical Report UILU-ENG-09-2214, July 2009.

[9] M. Mardani, G. Mateos, and G. B. Giannakis, "Unveiling network anomalies across flows and time via sparsity and low rank," *IEEE Trans. Info. Theory*, 2011 (submitted).

[10] Y. Nesterov, "A method of solving a convex programming problem with convergence rate $o(1/k^2)$," *Soviet Mathematics Doklady*, vol. 27, pp. 372–376, 1983.

[11] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Rev.*, vol. 52, no. 3, pp. 471–501, 2010.

[12] M. Thottan and C. Ji, "Anomaly detection in IP networks," *IEEE Trans. Signal Process.*, vol. 51, pp. 2191–2204, Aug. 2003.

[13] Z. Zhou, X. Li, J. Wright, E. Candes, and Y. Ma, "Stable principal component pursuit," in *Proc. of Intl. Symp. on Information Theory*, Austin, TX, Jun. 2010, pp. 1518–1522.