Leveraging Sparsity and Low Rank for Large-Scale Networks and Data Science

---

A DISSERTATION

SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL

OF THE UNIVERSITY OF MINNESOTA

BY

Morteza Mardani

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

Professor Georgios B. Giannakis, Advisor

May 2015

Acknowledgments

First and foremost, my deepest gratitude goes to my Ph. D. advisor Prof. Georgios B. Giannakis. I would like to thank him for giving me the opportunity to embark on this journey as a graduate student, a privilege for which I am really honored. This thesis would not have been possible without all his insightful suggestions. He has been a true teacher to me, and with his guidance and constant training, now I feel not only am I a better researcher, but also a better person with higher professional skills. His passion and diligence in research propelled me whenever I felt stuck into a problem. His aptitude for order and discipline made me more organized, and his intuitive mind taught me how to find a good research problem. I would also like to extend my sincerest appreciation for true understanding of my concerns during my graduate studies, and many other reasons which cannot be expressed in the space provided.

Due thanks go to Profs. Mostafa Kaveh, Kamil Ugurbil, Nikos Sidiropoulos, Jarvis Haupt, and Gilad Lerman for agreeing to serve on my committee. I have received invaluable comments and feedback from all of them which resulted in great improvement of the quality and the presentation of my research. Prof. Kaveh has been exceptionally kind and supportive from the early days I came to Minnesota. I got to know Prof. Ugurbil more recently since I started working on medical-imaging related themes, and I appreciate his interests and valuable feedbacks to my work. I am honored to know Prof. Sidiropoulos since the first years at University of Minnesota, where I learned a lot from his personality. I also would like to thank Prof. Haupt, whose attitude makes him a supportive friend to me rather than a faculty member. I am also thankful to Prof. Lerman for his kindness and our fruitful discussions.

Throughout my graduate studies, I had the opportunity to collaborate with several individuals and I greatly benefited from their vision, ideas, and insights. I would like to extend my gratitude to Prof. Seung-Jun Kim who was patient enough to train me the first couple of years, and Prof. Gonzalo Mateos, with whom we had a great research collaboration. I would also like to acknowledge the grants that support financially our research. My sincerest thanks go to Prof. Michael Mahoney, who provided me with the opportunity to visit the International Computer Science Institute, and the AMPLab at the University of California, Berkeley. It was a great opportunity that gave me the chance to learn new exciting ideas in graph signal processing, and get to know the machine learning pioneers in the AMPlab closely. I am also thankful of my M. Sc. advisor, Prof. Farshad

Lahouti, for introducing me to the world of academic research.

The material in this thesis benefited from discussions with current and former members of SPiNCOM: Prof. Kostas Slavakis, Dr. Daniele Angelosante, Brian Baingana, Dr. Alfonso Cano, Dr. Emiliano Dall'Anese, Dr. Pedro Forero, Prof. Nikos Gatsis, Guobing Li, Prof. Geert Leus, Prof. Antonio G. Marques, Dr. Eric Msechu, Prof. Ketan Rajawat, Dr. Nasim Yahya Soltani, Yu Zhang, Dr. Juan-Andrés Bazerque, Dr. Shahrokh Farahmand, Dr. Vassilis Kekatos, Prof. Yannis Schizas, Prof. Hao Zhu, Donghoon Lee, Paniotis Traganitis, Tianyi Chen, Gang Wang, Daniel Romero, Luis Miguel Lopez, Liang Zhang, Georgios Karanikolas, Fateme Sheikholeslami, Yanning Shen, Dimitris Berberidis, and Eric Blake.

I am not forgetting my friends, some of which I have already mentioned above, both the ones in Minnesota, and my long-term friends from Iran, particularly: Meisam Razaviyayn, Maziar Sanjabi, Mojtaba Kadkhodaei, Ali Ghoreyshi, Armin Zare, Hossein Zendehdel, Aboozar Ghaffari, Maral Mousavi, Behnaz Forootaninia, Farnaz Forootaninia, Mehdi Lamee, Pardees Azodanloo, Shayesteh Kiaei, Abbas Sohrabpour, Karen Khatamifard, Sepideh Hassanmoghadam, Hamed Samavat, Sepehr Salehi Mashaei, Ameer Kian, and Saber Taghvaeeyan.

Finally, gift of a family is incomparable. They are the source of my strength, motivation, and sustenance. A special feeling of gratitude to my loving parents Maheen and Majeed, whose heart I know is sick of my long distance. Special thanks also goes to my lovely sisters and brothers with their lovely kids who never left my side.

Morteza Mardani, Berkeley, California, April 21 2015.

Dedication

This dissertation is dedicated to my family for their endless love and support.

Abstract

We live in an era of "data deluge," with pervasive sensors collecting massive amounts of information on every bit of our lives, churning out enormous streams of raw data in a wide variety of formats. While big data may bring "big blessings," there are formidable challenges in dealing with large-scale datasets. The *sheer volume* of data makes it often impossible to run analytics using central processors and storage units. Network data are also often *geographically spread*, and collecting the data might be infeasible due to communication costs or privacy concerns. Disparate origin of data also makes the datasets often *incomplete*, and thus a sizable portion of entries are missing. Moreover, large-scale data are prone to contain *corrupted measurements*, communication errors, and even suffer from *anomalies* due to cyberattacks. Moreover, as many sources continuously generate data in *real time*, analytics must often be performed online as well as without an opportunity to revisit past data. Last but not least, due to *variety*, data is typically indexed by multiple dimensions.

Towards our vision to facilitate learning, this thesis contributes to cope with these challenges via leveraging the *low intrinsic-dimensionality* of data by means of sparsity and low rank. To build a versatile model capturing various data irregularities, the present thesis focuses first on a low-rank plus compressed-sparse matrix model, which proves successful in unveiling traffic anomalies in backbone networks. Leveraging the nuclear and ℓ_1-norm, exact reconstruction guarantees are established for a convex estimator of the unknowns. Inspired by the crucial task of network traffic monitoring, the scope of this model and recovery task is broaden to a tomographic task of jointly mapping out nominal and anomalous traffic from undersampled linear measurements.

Despite the success of nuclear-norm minimization in capturing the data low-dimensionality, it scales very poorly with the data size mainly due to its entangled nature. This indeed hinders decentralized and streaming analytics. To mitigate this computational challenge, this thesis puts forth a neat framework which permeates benefits from a bilinear characterization of nuclear-norm to bring separability at the expense of nonconvexity. Notwithstanding, it is proven that under certain conditions stationary points of nonconvex program coincide with the optimum of the convex counterpart. Using this idea along with theory of alternating minimization we develop lightweight algorithms with low communication-overhead for in-network processing; and provably convergent online ones suitable for streaming analytics. All in all, the major innovative claim is that even with the budget of distributed computation and sequential acquisition one can hope to achieve accurate reconstruction guarantees offered by the batch nuclear-norm minimization.

Finally, inspired by the k-space data interpolation task appearing in dynamic magnetic resonance imaging, a novel tensor subspace learning framework is introduced to handle streaming multidimensional data. It capitalizes on the PARAFAC decomposition and effects low tensor rank by means of the Tykhonov regularization, that enjoys separability and offers real-time MRI reconstruction tailoring e.g., image-guided radiation therapy applications.

Contents

List of Figures

List of Tables

Chapter 1

Learning from 'Big Data'

1.1 Motivation and Context

We live in an era of data deluge. Pervasive sensors collect massive amounts of information on every bit of our lives, churning out enormous streams of raw data in a wide variety of formats. A large volume of this data attributes to network data, which can represent a wide range of physical, biological, and social phenomena. For instance, we as the users of the Facebook social network happily feed 10 billion messages per day, click the "like" button 4.5 billion times and upload 350 million new pictures each and every day. Learning from these large volumes of data is expected to bring significant science and engineering advances along with consequent improvements in quality of life. For instance, federal information technology officials have reported that real-time analytics of healthcare data can help the government cut at least 10% annually from the federal budget, or about $\$1,200$ per American, by simply detecting improper payments before they occur [1].

While big data may bring "big blessings," there are formidable challenges in dealing with large-scale datasets. The *sheer volume* of data makes it often impossible to run analytics using central processors and storage units. Network data are also *geographically spread*, and collecting the data might be infeasible due to communication costs or privacy concerns. In addition, disparate origin of the data makes the datasets often *incomplete*, and thus a sizable portion of entries are missing. Moreover, large-scale data are prone to contain *corrupted*

measurements, communication errors, and even suffer from *anomalies* such as cyberattacks. Furthermore, as many sources continuously generate data in *real time*, analytics must often be performed online as well as without an opportunity to revisit past data.

1.1.1 Our vision

In order to draw inference from 'Big Data' with the aforementioned challenges we leverage the groundbreaking advances in machine learning, signal processing, and optimization theory to exploit the wealth of the encoded structures in data. In fact, the low *intrinsic-dimensionality* or the so-termed *parsimonious* nature of data plays a pivotal role towards inference. This parsimony typically emanates from the spatiotemporal correlations present in practical signals. In large-scale networks, the complex network structures including social, temporal, and spatial dimensions, render the network data highly correlated. For instance, the origin-to-destination (OD) traffic flows in the backbone of Internet Protocol (IP) networks exhibit dependencies mainly due to traffic generation patterns [74], which can facilitate network monitoring tasks such as identifying the anomalies occurred due to cyberattacks. Another instance appears in the context magnetic resonance imaging (MRI), where the cardiac snapshots besides spatial smoothness, share a big portion of the heart organ that renders them temporally correlated.

In this context, our "vision" is to leverage the data parsimony to provide tangible answers, both theoretical and practical, to the following intriguing questions:

- What is an *encompassing generative model* capturing the spatiotemporal correlations present in data, that is useful e.g., to create a holistic map of network traffic in the backbone of IP networks, as a pivotal input for proactive security and network management tasks?

- What are the enabling technologies for the *scalable* execution of inference tasks with sheer volume of *streaming* data which are possibly *distributed*?

- How to handle *multidimensional* streaming data-arrays appearing for instance in high-resolution imaging via MRI to accelerate the acquisition process?

Toward these goals, this dissertation capitalizes on the sparsity and low rank as key means for developing *effective* and *low-complexity* data analytics.

1.1.2 Sparsity and low lank

Leveraging sparsity by means of convex optimization has been widely accepted as a computationally efficient technique for model selection and parameter estimation [29, 30]. The premise is that practical signals typically (under certain transformations) exhibit a few dominant components. For instance, MRI images are known to consist of only a few dominant Fourier coefficients that render them sparse in the Fourier domain. The prior sparsity information about the unknown is then imposed through the convex ℓ_1-nrom requlatizer $\|\mathbf{x}\|_1 := \sum_i |x_i|$, a tight surrogate for the nonconvex ℓ_0-norm, namely $\|\mathbf{x}\|_0 := |\text{supp}(\mathbf{x})|$. The ℓ_1-norm regularizer has been proven successful attaining exact/stable reconstruction guarantees in various recovery tasks including sparse linear regression (a.k.a. compressive sampling) and dictionary learning; see e.g., [29, 30, 44]. Considering a sparse $\mathbf{x} \in \mathbb{R}^N$ with the sparsity level $\|\mathbf{x}\|_0 = s$, compressive sampling (CS) asserts that one can accurately reconstruct $\mathbf{x}$ from only a small $m = \mathcal{O}(s \log(N/s))$ *proper* linear measurements $\mathbf{y} = \mathbf{A}\mathbf{x}$ upon solving the convex program

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x} \in \mathbb{R}^N} \|\mathbf{x}\|_1 \quad \text{s. to} \qquad \mathbf{y} = \mathbf{A}\mathbf{x}, \tag{1.1}$$

that is known as least absolute shrinkage and selection operator (LASSO) [138].

Another way to effect the inherent data low-dimensionality is through matrix rank. The spatiotemporal correlation of data collected in a matrix render the resulting matrix exhibit low rank. For the matrix $\mathbf{X} \in \mathbb{R}^{N \times N}$, nuclear norm $\|\mathbf{X}\|_* := \sum_i \sigma_i(\mathbf{X})$, ($\sigma_i(\cdot)$ signifies i-th singular value), is adopted as a convex surrogate for the nonconvex and combinatorial matrix rank. In essence, nuclear-norm can be envisioned as the ℓ_1-norm of the matrix singular values that promotes sparsity in the *spectrum* domain. Similar to ℓ_1-norm, the nuclear norm is known to attain stable/exact reconstruction guarantees in numerous recovery tasks such as matrix completion [26, 27], and low-rank-plus-sparse matrix decomposition [25, 33]. Matrix completion is particularly inspired by the NetFlix movie recommendation system,

where each user rates a small fraction of possibly random collection of movies, and the common trends among movies as well as users can be utilized to predict/recommend movies of possible interest to users. More precisely, given a *low rank* matrix $\mathbf{X}$ with only a small subset of its entries indexed by Ω ($|\Omega| \ll N^2$) accessible, one can *impute* the missing entries via solving the convex program

$$\hat{\mathbf{X}} = \arg \min_{\mathbf{X} \in \mathbb{R}^{N \times N}} \|\mathbf{X}\|_* \quad \text{s. to} \quad [\mathbf{X}]_{m,n} = [\mathbf{M}]_{m,n}, \quad (m,n) \in \Omega \tag{1.2}$$

It is shown that if the energy of $\mathbf{X}$ is sufficiently *spread out*, which can be fulfilled for instance when the singular vectors are non-spiky, then with only $\mathcal{O}(Nr\log^2(N))$ uniformly chosen matrix entries, one can accurately recover the misses [26, 27].

More recently, the problem of decomposing a matrix into low-rank and sparse components has become popular with numerous applications ranging from computer vision to graphical models [25, 33]. This is a useful model since many practical signals naturally are superposition of background and foreground components, that can be well represented by low-rank and sparse matrices, respectively. Formally speaking, for the data matrix $\mathbf{M} = \mathbf{X} + \mathbf{A}$, the underlying low-rank $\mathbf{X}$ and sparse $\mathbf{A}$ can be estimated as

$$\{\hat{\mathbf{X}}, \hat{\mathbf{A}}\} = \arg \min_{\mathbf{X}, \mathbf{A} \in \mathbb{R}^{N \times N}} \|\mathbf{X}\|_* + \lambda \|\mathbf{A}\|_1 \quad \text{s. to} \quad \mathbf{M} = \mathbf{X} + \mathbf{A}, \tag{1.3}$$

where the tunning parameter $\lambda > 0$ controls the trade-off between rank and sparsity level. The key to success is the *uncertainty principal* [33] asserting that as long as the energy of $\mathbf{X}$ is sufficiently *spread out* across the entires (satisfied for matrices with non-spiky singular vectors), and the support of $\mathbf{A}$ is sufficiently *sporadic*, accurate reconstruction becomes possible [25, 33]. Of course, on top of these assumptions $\mathbf{A}$ and $\mathbf{X}$ should be sufficiently sparse and low rank. The literature contains various recovery results under both deterministic and random modeling assumptions. For example, the results in [25] adopt a random orthonormal model for $\mathbf{X}$ when the support of $\mathbf{A}$ is uniformly random, and consequently it asserts that even with $\|\mathbf{A}\|_0 = \mathcal{O}(N^2)$ and $r = \mathcal{O}(N(\log(N))^{-2})$, one can hope for accurate estimation.

Before moving on, it is useful to introdue a neat property of the nuclear-norm that proves instrumental throughput this dissertation. Being appealing as a convex surrogate

of rank, nuclear-norm lacks separability across rows and columns of the matrix, thus challenging streaming and decentralized data analytics. To mitigate this challenge, we adopt a charactrization of the nuclear-norm which builds on a bilinear factorization $\mathbf{X} = \mathbf{L}\mathbf{Q}'$ with the factor matrices $\mathbf{L} \in \mathbb{R}^{M\times\rho}$ and $\mathbf{Q} \in \mathbb{R}^{N\times\rho}$. The value of ρ is chosen sufficiently large to overestimate $rank(\mathbf{X})$. It should be noted that such factors always exist e.g., via singular value decomposition (SVD). Nuclear-norm can then be alternatively expressed as [117,134]

$$\|\mathbf{X}\|_* := \min_{\{\mathbf{L},\mathbf{Q}\}} \quad \frac{1}{2}\left\{\|\mathbf{L}\|_F^2 + \|\mathbf{Q}\|_F^2\right\}, \quad \text{s. to} \quad \mathbf{X} = \mathbf{L}\mathbf{Q}'. \tag{1.4}$$

Adopting this characterization typically comes at the expense of nonconvexity for the corresponding recovery task. However, as it will be seen in the ensuing chapters for the considered recovery tasks, under certain conditions this adoption comes with no loss of optimality.

1.2 Motivating Application Domains

While the "Big Data" analytics put forth in this dissertation cover a wide range of applications, this dissertation predominantly focuses on the important tasks of network traffic monitoring and accelerating the acquisition process in MRI. The former is critical for network management and proactive security purposes to provide seamless and secure communication over IP networks. Needless to say, the latter is also crucial to reduce the breath-holding times for patients and to provide physicians with artifact-free images of organs for diseases diagnosis.

1.2.1 Network traffic monitoring

Consider a backbone Internet protocol (IP) network naturally modeled as a directed graph $G(\mathcal{N}, \mathcal{L})$, 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 flows $\mathcal{F}$ (with $|\mathcal{F}| = F$) associated with specific origin-destination (OD) 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 adopted here, that is, a given flow's traffic is carried through multiple links connecting the corresponding OD pair along a single path.

Let $r_{l,f}$, $l \in \mathcal{L}, f \in \mathcal{F}$, denote the flow to link assignments (routing), which take the value one whenever flow f is carried over link l, and zero otherwise. Unless otherwise stated, the routing matrix $\mathbf{R} := [r_{l,f}] \in \{0,1\}^{L \times F}$ is assumed fixed and given. Likewise, let $z_{f,t}$ denote the unknown traffic rate of flow f at time t, measured in e.g., Mbps. At any given time instant t, the traffic carried over link l is then the superposition of the flow rates routed through link l, i.e., $\sum_{f \in \mathcal{F}} r_{l,f} z_{f,t}$.

It is not uncommon for some of the flow rates to experience unusual abrupt changes. These so-termed *traffic volume anomalies* are typically due to unexpected network failures, or cyberattacks (e.g., denial of service attacks) which aim at compromising the services offered by the network [137]. Let $a_{f,t}$ denote the unknown traffic volume anomaly of flow f at time t. In the presence of anomalous flows, the measured traffic carried by link l over a time horizon $t \in [1, T]$, is then given by

$$y_{l,t} = \sum_{f \in \mathcal{F}} r_{l,f}(x_{f,t} + a_{f,t}) + v_{l,t}, \quad t = 1, ..., T \tag{1.5}$$

where the noise variables $v_{l,t}$ account for measurement errors and unmodeled dynamics.

In IP networks, $y_{l,t}$ is readily measured via SNMP, supported by most routers. Missing entries in the link-level measurements $y_{l,t}$ may however 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 [119]. To model missing link measurements, collect the tuples (l, t) associated with the available observations $y_{l,t}$ in the set $\Omega \in [1, 2, ..., L] \times [1, 2, ..., T]$. Introducing the 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 (possibly incomplete) set of measurements in (1.5) can be expressed in compact matrix form as

$$\mathcal{P}_\Omega(\mathbf{Y}) = \mathcal{P}_\Omega(\mathbf{R}\,(\mathbf{X} + \mathbf{A}) + \mathbf{V}) \tag{1.6}$$

where the sampling operator $\mathcal{P}_\Omega(\cdot)$ sets the entries of its matrix argument not in Ω to zero, and keeps the rest unchanged.

In addition to link counts that are expressed as in (1.6), the traffic-related inference tasks may possibly exploit other data sources to enhance estimation accuracy. A useful

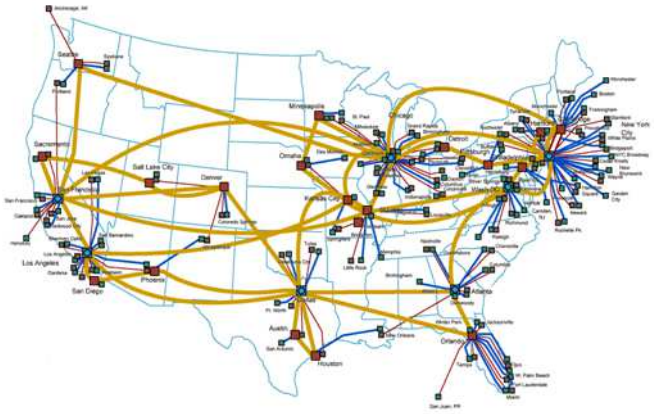


Figure 1.1: Internet-2 backbone network across USA.

such source is the direct flow-level measurements expressed per flow f as

$$u_{f,t} = x_{f,t} + a_{f,t} + w_{f,t}, \quad t = 1, ..., T \tag{1.7}$$

where $w_{f,t}$ accounts for measurement and modeling errors. The flow traffic (1.7) is sampled via NetFlow [74] at each origin node. However, due to the high cost of deploying NetFlow one can acquire only a limited number of $\{u_{f,t}\}$ samples [74]. Similar to link counts, to account for missing flow-level data, collect the available pairs (f, t) in the set $\Pi \in [F] \times [T]$, and introduce the noise matrix $\mathbf{W} := [w_{f,t}] \in \mathbb{R}^{F \times T}$. The flow counts in (1.7) can then be compactly written as

$$\mathcal{P}_{\Pi}(\mathbf{U}) = \mathcal{P}_{\Pi}(\mathbf{X} + \mathbf{A} + \mathbf{W}). \tag{1.8}$$

Matrix $\mathbf{X}$ contains the nominal traffic flows over the time horizon of interest. Common temporal traffic flow patterns in addition to their periodic behavior, render most rows (respectively columns) of $\mathbf{X}$ linearly dependent, and thus $\mathbf{X}$ typically has low rank. This intuitive property has been extensively validated with real network data; see e.g., [74]. Anomalies in $\mathbf{A}$ are expected to occur sporadically over time, and last for a small fraction of the (possibly long) 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).

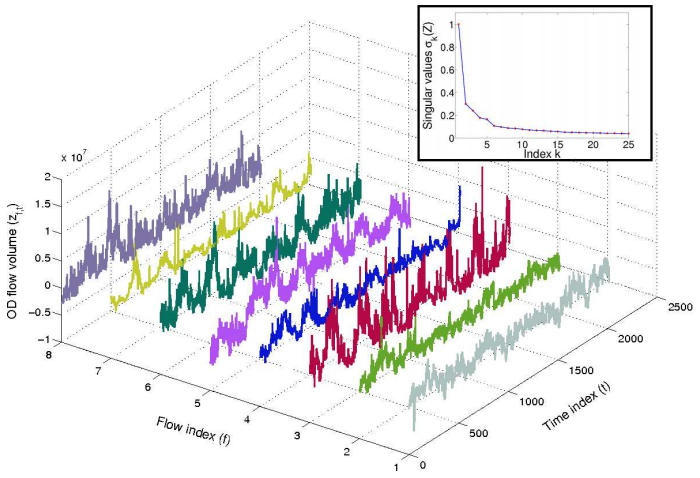


Figure 1.2: Volumes of representative (out of 121 total) OD flows, taken from the operation of Internet-2. Temporal periodicities and correlations across flows are apparent.

Given the full link count matrix $\mathbf{Y}$ and knowledge of the routing matrix $\mathbf{R}$, many inference techniques have been developed over the years to identify the network anomalies in $\mathbf{A}$; see e.g., [67, 72, 74, 158] and references therein. More recently, principal component analysis (PCA)-based methods, exploiting the low intrinsic-dimensionality of the nominal traffic, have been proven very successful [72, 158]. The accuracy of this estimator depends on knowing or estimating accurately rank of the underlying nominal traffic subspace, which in most networks, where link-load measurements are available for sequential time instants, the rank to be retained by PCA is unknown.

When it comes to estimating the nominal traffic matrix $\mathbf{X}$ inference from link counts alone becomes seriously ill-posed. Typically, certain prior information about the flow-level traffic $\mathbf{X}$ are assumed to render the sought traffic estimation well-posed. In this direction, ample research has been carried out in different contexts including transportation science and traffic analysis of computer networks [32, 67, 162], which are mainly derived from the principles of least-squares and Gaussian models [32, 162], Poisson models [146], and entropy minimization [166]. However, existing techniques either lack robustness to potential anomalies or ignore the temporal correlations. Furthermore, they typically assume that full link and flow counts are available.

1.2.2 Dynamic magnetic resonance imaging

MRI nowadays serves as a major imaging modality for noninvasive diagnosis of diseases in clinical practice [52]. However, the slow acquisition speeds introduce motions causing image artifacts, that hinder imaging of moving objects such as the heart, and the contrast-changing objects such as the flowing blood in difussion MRI for angiography. Dynamic MRI aspires to cope with these challenges by acquiring a low-spatial yet high-temporal resolution sequence of images [52]. This renders a possibly sizable portion of k-space data per snapshot inaccurate or missing, but the high spatiotemporal correlation of images can be leveraged to interpolate misses; see e.g., [81, 84, 112] as a few noteworthy representative.

To be more precise, consider a temporal ground-truth sequence $\{\mathbf{L}_t\}_{t=1}^T$ of $M \times N$ images of possible interest. The undersampled k-space data acquired by the MR machine at t-th snapshot, say $\{y_t^{(\ell)}\}_{\ell=1}^{L_t}$, then adhere to

$$y_t^{(\ell)} = [\mathcal{F}(\mathbf{L}_t)]_{i_\ell, j_\ell} + v_t^{(\ell)}, \quad (i_\ell, j_\ell) \in \Omega_t \tag{1.9}$$

where $\mathcal{F}(\cdot)$ denotes the two-dimensional discrete Fourier transform (DFT) operator, and the set $\Omega_t \subset [M] \times [N]$ indexes the acquired k-space data. The sampling trajectory Ω_t cannot take arbitrary shapes due to physical and physiological constraints. Various types of trajectories including rectilinear (also called Cartesian), radial, spiral, and zig-zag scanning constitute the most popular ones in clinical practice [78]. Non-cartesian sampling is an emerging area with great potential in a variety of applications. Many desired sampling trajectories can be implemented by appropriately designing spin-echo excitation sequence for the gradient coils. Note that in the present tensor model the sampling set Ω_t permits any arbitrary trajectory. The observations are complex-valued, meaning $y_t^{(\ell)} = \mathcal{R}\{y_t^{(\ell)}\} + j\mathcal{I}\{y_t^{(\ell)}\}$ counts for two real-valued observations. The acquisition time is clearly proportional to the sample count $\sum_{\tau=1}^t |\Omega_\tau|$, and it is desired to be as small as possible.

While the plain MRI has been widely used in clinical practice, relative to other medical imaging techniques such as computerized tomography (CT) it suffers from long acquisition times to collect the data needed for creating artifact-free images. Certain types of scans may take several minutes to acquire the necessary data. Parallel imaging has recently emerged

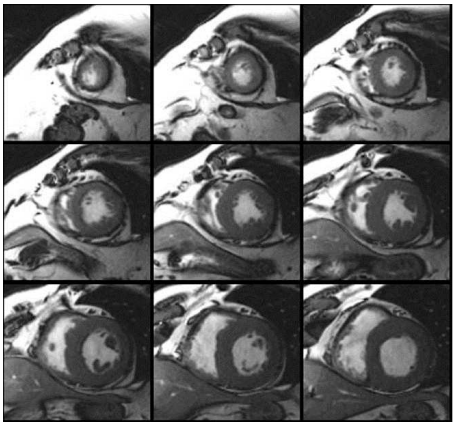


Figure 1.3: Temporal sequence of cardiac snapshots acquired via cine MRI.

as a robust means to accelerate the MRI acquisition process. Parallel MRI uses a phased array of coils, each one sensitive to signals returned from a limited spatial region of the imaged object [41]. The receiver coils are arranged in a way that their sensitivity profiles cover the desired field of view. Parallel imaging techniques are adapted to properly combine the images acquired across various coils. GRAPPA [41] and SENSE [41] are two commonly used techniques to combine MR images in the clinical practice.

1.3 Thesis Outline and Contributions

The research in this thesis contributes to the advancement of statistical learning theory from "Big Data." In accordance with our vision, outlined in Section 1.1.1, in order to cope with the "Big Data" challenges we put forth an encompassing low-rank plus compressed-sparse matrix model that nicely fits the network anomaly identification task, and puts the existing CS as well matrix imputation and decomposition tasks under the same umbrella. In light of spread nature of data, and the need for parallel processing, a novel algorithmic framework is put forth for decentralized nuclear-norm minimization under sparsity regularization, that is the first of its type to date. Along the same line, for streaming and large volume of data, a framework is introduced for online nuclear-norm minimization that results in lightweight data processing algorithms. Last but not least, to handle the multidimensional data struc-

tures, a novel framework is introduced for tensor subspace learning that is again the first of its type in the literature, and proves successful for acclerating the MRI acquisition process. In this direction, the contributions of this thesis is centered around the following five major intertwined thrusts:

[T1] Low-rank plus compressed-sparse matrix recovery

[T2] Tomographic low-rank and sparse matrix recovery: Applications to network traffic monitoring

[T3] Decentralized rank minimization and sparsity regularization

[T4] Online sparsity-regularized rank minimization: Applications to tracking network anomalies

[T5] Big data tensor subspace learning: Applications to Dynamic MRI

To gauge the effectiveness of the novel methods, extensive examinations with computer generated data are reported throughout the thesis. These are important since they provide a ground truth, against which performance can be assessed by evaluating suitable figures of merit. Nevertheless, no effort of this kind can have impact without thorough testing, experimentation, and validation with real data. To this end, various tests on real Internet traffic traces, and clinically-acquired MRI images are included to compile a comprehensive validation package. Elaborate discussion of [T1]-[T5] follows next along with a succinct literature review per thrust. Moreover, contributions of this thesis in each case are pointed out.

1.3.1 Low-rank plus compressed-sparse matrix recovery.

This research thrust was motivated by the important task of unveiling network traffic anomalies. Consider the Internet backbone network (see e.g., Fig. 1.1), where OD traffic flows, e.g., the information flow delivered from Chicago to Loss Angles, experience unusual changes which can result in congestion, and limit quality of service provisioning of the end users. These so-termed traffic-volume anomalies could be due to e.g., unexpected failures in networking equipment or cyberattacks [11,72,137]. Unveiling such anomalies in a promptly manner is a crucial monitoring task toward engineering network traffic. This is challenging

however, since the available data are usually high-dimensional, noisy and possibly incomplete link traffic $\mathbf{Y}$ (cf. (1.5)), which are the superposition of *unobservant* OD flows. More precisely, upon defining $\mathbf{X}_R := \mathbf{R}\mathbf{X}$, based on (1.5) the link counts admits the compact matrix form $\mathbf{Y} = \mathbf{X}_R + \mathbf{R}\mathbf{A} + \mathbf{V}$, where the *fat* matrix $\mathbf{R}$ corresponds to the routing information. As discussed in Section 1.2, the spatiotemporal correlations of nominal traffic and sporadic nature of anomalies render matrices $\mathbf{X}$ and $\mathbf{A}$ low rank and sparse, respectively. As a result, the anomaly detection task boils down to the low-rank plus compressed-sparse matrix decomposition

$$(\hat{\mathbf{X}}_R, \hat{\mathbf{A}}) = \arg\min_{\mathbf{X}_R, \mathbf{A}} \frac{1}{2}\|\mathbf{Y} - \mathbf{X}_R - \mathbf{R}\mathbf{A}\|_F^2 + \lambda_*\|\mathbf{X}_R\|_* + \lambda_1\|\mathbf{A}\|_1 \tag{1.10}$$

where the parameters λ_* and λ_1 control the rank and sparsity levels. It is worth noting that the scope of low-rank plus compressed-sparse matrix recovery goes beyond the anomaly detection, and, in general, it can be applied to any inference task dealing with decomposition of a signal into background and foreground components. A few pertinent application domains are outlined in Chapter 2. It is also important to recognize that the low-rank plus sparse decomposition or the so-termed robust PCA task in [25, 33] assume a *uniformly* sparse foreground, that can be easily violated in practical settings where the foreground is typically structured. In contrast with [25, 33], we however model the foreground as being sparse across a *proper* dictionary, which offers more flexibility to account for the structures. In general, the dictionary can be learned from training data, and for certain scenarios dealing with e.g., computer vision and imaging one can appeal to known sparsifying dictionaries such as Wavelet or DCT. In this respect, our main contributions, reported in Chapter 2, include: (i) establishing exact recovery conditions for the true low-rank and sparse matrices $(\mathbf{X}_R, \mathbf{A})$ in the absence of noise ($\|\mathbf{V}\|_F = 0$); (ii) developing batch iterative solvers based upon alternating-direction method-of-multipliers (ADMM) and approximate proximal gradient (APG) to procure the estimates; and (iii) evaluations on real Internet-2 traffic traces to identify anomaly maps across time instants and flows.

1.3.2 Tomographic low-rank and sparse matrix recovery: Applications to network traffic monitoring

This research thrust is also motivated by the important task of monitoring network traffic over operational IP networks. While the previous thrust primarily focuses on discovering anomaly maps, summarized in the matrix $\mathbf{A}$, a more ambitious objective is to identify the underlying nominal OD traffic flows as well, i.e., $(\mathbf{X}, \mathbf{A})$. This provides an atlas of the network traffic state, based on which the network operator can e.g., trigger proactive network security tasks upon observing any abnormal behaviors. This in turn necessitates additional information about OD flows besides the link counts available in the previous thrust. However, there are a huge number of OD flows in large-scale networks, and they are potentially subject to anomalies due to e.g., cyberattacks. To overcome this hurdle, we have proposed a novel approach which exploits the low dimensionality of the nominal traffic, and uses only a small fraction of OD flows together with the link counts to create an atlas of network traffic state across time and flows.

More formally, our input data consists of the link counts $\mathbf{Y}$ and the partial flow counts $\mathbf{Z}_\Pi$, which relate to the underlying nominal and anomalous traffic through $\mathbf{Y} = \mathbf{R}(\mathbf{X} + \mathbf{A}) + \mathbf{V}$, and $\mathbf{Z}_\Pi = \mathcal{P}_\Pi(\mathbf{X} + \mathbf{A} + \mathbf{W})$, elaborated in Section 1.2. A natural estimate of the unknowns can be obtained via the following convex formalism

$$(\hat{\mathbf{X}}, \hat{\mathbf{A}}) = \arg\min_{\mathbf{X},\mathbf{A}} \frac{1}{2}\|\mathbf{Y} - \mathbf{R}(\mathbf{X} + \mathbf{A})\|_F^2 + \frac{1}{2}\|\mathcal{P}_\Pi(\mathbf{Z} - \mathbf{X} - \mathbf{A})\|_F^2 + \lambda_*\|\mathbf{X}\|_* + \lambda_1\|\mathbf{A}\|_1, \tag{1.11}$$

where λ_* and λ_1 control the trade-off between traffic correlations and the sparsity of anomalies. In this respect, the major novelties of this dissertation, as reported in Chapter 3, include: (i) establishing exact recovery guarantees for the low-rank and sparse matrices $(\mathbf{X}, \mathbf{A})$ for the noise-free setting from (1.11); (ii) developing batch solvers for (1.11) via ADMM and alternating minimization; (iii) incorporating structural patterns of traffic and anomalies by means of a bilinear characterization of nuclear norm as detailed in Section 1.1.2; and (iv) evaluations with real Internet-2 traffic data traces.

1.3.3 Decentralized rank minimization and sparsity regularization.

Decentralized algorithms are clearly attractive for alleviating the computational and communication overhead associated with the collection of measurements at a central processing unit. Moreover, distributed algorithms are also desirable because of their scalability with regards to power requirements, network size, and robustness to isolated points of failure. For instance, a simple decentralized solution to the anomaly identification problem formulated in Section 1.2, is to adopt consensus-based iterations similar to those studied in e.g., [127]. However, this approach would still incur high computational cost for singular value decomposition (SVD) computations of the high-dimensional matrices during primal variable updates. The idea here is to exploit the fact that minimizing the nuclear-norm $\|\mathbf{X}\|_*$ is tantamount to minimizing $\frac{1}{2}\left\{\|\mathbf{L}\|_F^2+\|\mathbf{Q}\|_F^2\right\}$, where $\mathbf{X}=\mathbf{L}\mathbf{Q}'$ is a suitable decomposition of the clean traffic matrix $\mathbf{X}$. Leveraging this observation, our contributions in Chapter 4 are as follows: (i) we will formulate the distributed anomalography task as

$$\min_{\{\mathbf{L},\mathbf{Q},\mathbf{A}\}}\|\mathbf{Y}-\mathbf{L}\mathbf{Q}'-\mathbf{R}\mathbf{A}\|_F^2+\frac{\lambda_*}{2}\left\{\|\mathbf{L}\|_F^2+\|\mathbf{Q}\|_F^2\right\}+\lambda_1\|\mathbf{A}\|_1 \tag{1.12}$$

and seek a *low-complexity* distributed solution via ADMM after introducing auxiliary variables that act as local copies of variables associated with the neighbors at each node; see also [126, 127, 164]. We will further investigate conditions under which (P1) and (1.12) can be rendered equivalent. Due to nonconvexity of (1.12), it may exhibit stationary points which are not necessarily global optima. (ii) We will seek conditions on the low rank component and the noise variance under which every stationary point of (1.12) achieves the globally optimal solution of (1.10). (iii) As a means of offering additional engineering design insights, we delineate different performance tradeoffs that emerge as the network size increases, with emphasis on comparisons in terms of robustness and the required communication cost with respect to centralized processing alternatives. (iv) Finally, tests with real-world datasets confirm the efficacy of the novel decentralized algorithms toward identifying anomalies and estimating end-to-end packet delays in Internet-2 backbone network. Once more, the proposed framework is applicable to other inference tasks concerned with distributed rank minimization under sparsity regularization.

1.3.4 Online sparsity-regularized rank minimization: Applications to tracking network anomalies

Monitoring of large-scale networks necessitates massive collection 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 adjusted to effect traffic load balancing and avoid congestion caused by e.g., traffic anomalies. To account for *slowly time-varing* 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 $\mathcal{P}_{\Omega_t}(\mathbf{y}_t) = \mathcal{P}_{\Omega_t}(\mathbf{x}_t + \mathbf{R}_t\mathbf{a}_t + \mathbf{v}_t,\ t = 1, 2, \ldots$, where the link-level traffic $\mathbf{x}_t := \mathbf{R}_t\mathbf{z}_t$ lies in a low-dimensional subspace.

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 and exploiting the spatiotemporal correlations of the observations to identify the anomalies in real-time. An adaptive counterpart of (1.12) is the exponentially-weighted LS estimator found by minimizing the empirical cost

$$\min_{\{\mathbf{L},\mathbf{Q},\mathbf{A}\}} \sum_{\tau=1}^{t} \beta^{t-\tau} \left[\|\mathcal{P}_{\Omega_\tau}(\mathbf{y}_\tau - \mathbf{L}\mathbf{q}_\tau - \mathbf{R}_\tau\mathbf{a}_\tau)\|_2^2 + \frac{\lambda_*}{2\sum_{u=1}^t \beta^{t-u}}\|\mathbf{L}\|_F^2 + \frac{\lambda_*}{2}\|\mathbf{q}_\tau\|_2^2 + \lambda_1\|\mathbf{a}_\tau\|_1 \right] \tag{1.13}$$

in which $0 < \beta \leq 1$ is the so-termed forgetting factor. For $\beta < 1$, data in the distant past are exponentially downweighted, which facilitates tracking network anomalies in nonstationary environments. In this regards, our main contributions in Chapter 3 are: (i) an online algorithm for *dynamic anomalography* can be obtained by resorting to alternating minimization of (1.13). Each time a new datum is acquired, anomaly estimates are formed via the Lasso [138], and the low-rank nominal traffic subspace $\mathbf{L}$ is refined using recursive LS. For situations were reducing computational complexity is critical, we will develop an

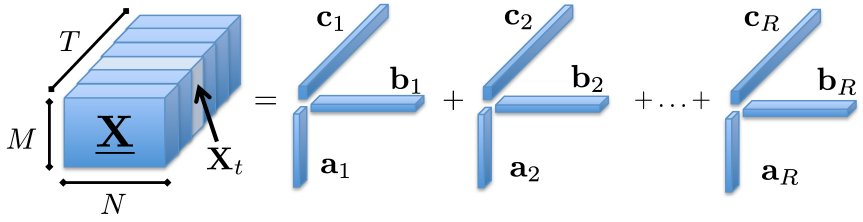


Figure 1.4: Tensor PARAFAC decomposition.

online stochastic gradient algorithm based on Nesterov's acceleration technique as well. (ii) convergence and optimality of the aforementioned algorithm is established be resorting to the theory of martingale sequences. Last but not least, (iii) simulated tests are examined that demonstrate the efficacy of the novel schemes in real-time detection of Internet-2 network anomalies.

It is worth noting that the envisioned algorithms to tackle (1.13) are closely related to robust subspace tracking algorithms [65, 70, 107, 152]. Different from existing works, the estimation problem (1.13) is more challenging due to the presence of the (compression) routing matrix $\mathbf{R}_t$, which challenges identifiability of the anomalies.

1.3.5 Big data tensor subspace learning: Applications to dynamic MRI

Although the matrix models considered in the previous research thrusts are quite versatile and can subsume a variety of important frameworks as special cases, the particular planar arrangement of data poses limitations in capturing available structures that can be crucial for effective interpolation. Imagine for instance the MRI image associated with different heart snapshots, where the matrix models can be readily used by 'unfolding' the two-dimensional image pixels. However, such an unfolding destroys the structure that one looks for. Such structure can be explicitly account for by arranging the pixels in a three-dimensional array (x, y, t) or *tensor*. A rank-one three-way array is the outer product of three vectors, namely $\mathbf{a} \circ \mathbf{b} \circ \mathbf{c}$, with the (m, n, t)-th element $a_m b_n c_t$, where a_m, b_n, and c_t capture variation across x-coordinate, y-coordinate, and time, respectively, in the previous example.

The rank of a tensor is the smallest number of rank-one tensors that sum up to generate the given tensor, as illustrated in Fig. 1.4. Notwithstanding, this is not an incremental extension from low-rank matrices to low-rank tensors, since even computing the tensor rank is an NP-hard problem in itself. Defining a convex surrogate for the rank penalty such as the nuclear norm for matrices is not obvious either, since singular values when applicable, e.g., in the Tucker model, are not related to the rank [15, 68]. Low-rank tensor approximation is relatively mature in linear algebra and factor analysis, where it is usually called parallel factor analysis (PARAFAC) or canonical decomposition (CANDECOMP). It expresses a three-way tensor as $\underline{\mathbf{X}} = \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$, where the tensor rank is the minimum value for R for which the identity holds. PARAFAC essentially asserts one can summarize the tensor $\underline{\mathbf{X}}$ with three factor matrices $\mathbf{A} \in \mathbb{R}^{M \times R}$, $\mathbf{B} \in \mathbb{R}^{N \times R}$, and $\mathbf{C} \in \mathbb{R}^{T \times R}$. Unlike the matrix case, low-rank tensor decomposition can be unique. There is deep theory behind this result [69], and algorithms recovering the rank-one factors [69]. However, various computational and big data-related challenges remain. Missing data have been handled in rather *ad hoc* ways [5, 50, 82, 129]. Parallel and decentralized implementations have not also been thoroughly addressed.

In this context, our fresh idea in this dissertation is to capitalize on the low rank of tensor to solve ill-posed inverse problems appearing in various inference tasks such as missing data interpolation. To this end, we build on the Tykhonov rank regularization introduced in [13] which is a natural extension of the decomposable rank regularizer (1.4). In particular, for the tensor interpolation task, given the available features $\{y_{m,n,t}\}_{(m,n,t)\in\Omega}$, adoption of the rank regularizer yields the formalism

$$\begin{aligned} \hat{\underline{\mathbf{X}}} := \arg \min_{\{\underline{\mathbf{X}},\mathbf{A},\mathbf{B},\mathbf{C}\}} & \|\mathcal{P}_\Omega(\underline{\mathbf{Y}} - \underline{\mathbf{X}})\|_F^2 + \frac{\lambda}{2}\big(\|\mathbf{A}\|_F^2 + \|\mathbf{B}\|_F^2 + \|\mathbf{C}\|_F^2\big) \\ \text{s. to } & \mathbf{X}_t = \mathbf{A}\text{diag}\big(\mathbf{e}_t'\mathbf{C}\big)\mathbf{B}', \quad t = 1, \ldots, T \end{aligned} \tag{1.14}$$

Different from the matrix case, it is unclear whether the regularization in (1.14) bears any relation with the tensor rank. Interestingly, the analysis in [13] reveals that (1.14) provably yields a low-rank $\hat{\underline{\mathbf{X}}}$ for sufficiently large λ.

In parallel to the matrix case motivated in Chapter 5, processing large-scale tensors

arising from Big Data applications such as high-resolution brain MRI [84] requires low-complexity algorithms implementable with compact storage. A viable approach to this end is to develop recursive inference algorithms for big data tensors, which has not been addressed to date when the tensor slices are only partially available. With the batch formulation (1.14) in mind, and building on the separable structure of the tensor rank regularizer, we can end up with the following exponentially weighted least-squares formulation

$$\min_{\{\mathbf{A},\mathbf{B},\mathbf{C}\}} \sum_{\tau=1}^{t} \beta^{t-\tau} \Big[\|\mathcal{P}_{\Omega_t}(\mathbf{Y}_t - \mathbf{A}\mathrm{diag}\big(\mathbf{e}_t'\mathbf{C}\big)\mathbf{B}')\|_F^2 + \frac{\mu}{2\sum_{\tau=1}^{t}\beta^{t-\tau}}(\|\mathbf{A}\|_F^2 + \|\mathbf{B}\|_F^2) + \mu\|\mathbf{e}_t'\mathbf{C}\|^2 \Big]. \tag{1.15}$$

With this is mind, our main contributions in Chapter 5 are listed as follows: (i) an encompassing generative tensor model is adopted where the observations are linear projections of a low-rank plus sparse tensor. For this model, light-weight first-order algorithms based on stochastic gradient-descent methods are devised to estimate the unknowns. (ii) Applications of the considered model in dynamic and parallel MRI is also throughly investigated, where recursive solvers are developed to reconstruct the MRI images from highly-undersampled k-space data. Finally, (iii) simulated tests with real cardiac MRI are examined to show the merits of the novel real-time MR reconstruction schemes.

1.4 Published Results

The present Ph.D. work in Chapters 2-5 has resulted in the publication of 5 journals papers in the Association for Computing Machinery (ACM) and the Institute of Electrical and Electronic Engineers (IEEE) Transactions on Information Theory [97], Networking [93], and Signal Processing [95, 98] as well as the Journal of Selected topics in Signal Processing [96]. More recent results in Chapter 6 are intended for submission to the Journal of Magnetic Resonance in Medicine, and IEEE Transactions on Medical Imaging. The work has also been disseminated at pertinent conferences, where a total of 15 articles have been either presented, or, accepted for presentation [9, 47, 47, 85, 86, 91, 92, 99–104, 153].

1.5 Notational Convenience

The following notational conventions will be adopted throughout the subsequent chapters. Bold uppercase letters will denote matrices, whereas bold lowercase letters will stand for column vectors. Multidimensional arrays (tensors) are denoted by underscored bold uppercase letters, and caligraphic letters will be used for sets. Whenever the context makes it sufficiently clear, $[.]_{ij}$ will be used for a matrix to denote block matrix partitioning. Operators $\otimes$, $\oplus$, $\odot$, $\circ$, $(\cdot)'$, $(\cdot)^\dagger$, $\lambda_{\max}(\cdot)$, $\lambda_{\min}(\cdot)$, $\sigma_{\max}(\cdot)$, $\sigma_{\min}(\cdot)$, $\exp(\cdot)$, $\mathrm{tr}(\cdot)$, $\mathbb{E}[\cdot]$, $\mathrm{vec}[\cdot]$, $\mathrm{med}(\cdot)$ will denote Kronecker product, direct sum, Hadamard product, outer product, transposition, matrix pseudo-inverse, spectral radius, minimum eigenvalue, maximum singular values, minimum singular value, exponential function, matrix trace, expectation, matrix vectorization, and median, respectively. Vector $\mathrm{diag}(\mathbf{M})$ collects the diagonal entries of $\mathbf{M}$, whereas the diagonal matrix $\mathrm{diag}(\mathbf{v})$ has the entries of $\mathbf{v}$ on its diagonal. The ℓ_q-(pseudo) norm of vector $\mathbf{x} \in \mathbb{R}^p$ is $\|\mathbf{x}\|_q := (\sum_{i=1}^p |x_i|^q)^{1/q}$ for $q > 0$. Also, $|\cdot|$ will be used for the cardinality of a set and a magnitude of a scalar. For matrices $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{m\times n}$ define the trace inner product $\langle \mathbf{A}, \mathbf{B}\rangle := \mathrm{tr}(\mathbf{A}'\mathbf{B})$. Also, recall that $\|\mathbf{A}\|_F := \sqrt{\mathrm{tr}\,(\mathbf{A}\mathbf{A}')}$ is the Frobenious norm, $\|\mathbf{A}\|_1 := \sum_{i,j} |a_{ij}|$ is the ℓ_1-norm, $\|\mathbf{A}\|_\infty := \max_{i,j} |a_{ij}|$ is the ℓ_∞-norm, and $\|\mathbf{A}\|_* := \sum_i \sigma_i(\mathbf{A})$ is the nuclear norm. In addition, $\|\mathbf{A}\|_{1,1} := \max_{\|\mathbf{x}\|_1=1} \|\mathbf{A}\mathbf{x}\|_1 = \max_i \|\mathbf{e}_i'\mathbf{A}\|_1$ denotes the induced ℓ_1-norm, and likewise for the induced ℓ_∞-norm $\|\mathbf{A}\|_{\infty,\infty} := \max_{\|\mathbf{x}\|_\infty=1} \|\mathbf{A}\mathbf{x}\|_\infty = \max_i \|\mathbf{A}\mathbf{e}_i\|_1$. For the linear operator $\mathcal{A}$, define the operator norm $\|\mathcal{A}\| := \max_{\|\mathbf{X}\|_F=1} \|\mathcal{A}(\mathbf{X})\|_F$, which subsumes the spectral norm $\|\mathbf{A}\| := \max_{\|\mathbf{x}\|=1} \|\mathbf{A}\mathbf{x}\|$. Positive-definite matrices will be denoted by $\mathbf{A} \succ \mathbf{0}$. The $m \times m$ identity matrix will be represented by $\mathbf{I}_m$, while $\mathbf{0}_m$ will denote the $m \times 1$ vector of all zeros, and $\mathbf{0}_{m\times n} := \mathbf{0}_m\mathbf{0}_n'$. Similar notation will be adopted for vectors (matrices) of all ones. The i-th vector of the canonical basis in $\mathbb{R}^n$ will be denoted by $\mathbf{e}_{n,i}$, $i = 1, \ldots, n$. Define also the support set $\mathrm{supp}(\mathbf{A}) := \{(i,j) : a_{ij} \neq 0\}$; the indicator function $\mathbb{1}_{\{a=b\}}$ that equals one when $a = b$, and zero otherwise; and $[n] := \{1, 2, \ldots, n\}$.

Chapter 2

Low-Rank Plus Compressed-Sparse Matrix Recovery

2.1 Introduction

Let $\mathbf{X}_0 \in \mathbb{R}^{L\times T}$ be a low-rank matrix [$r := \text{rank}(\mathbf{X}_0) \ll \min(L,T)$], and let $\mathbf{A}_0 \in \mathbb{R}^{F\times T}$ be sparse ($s := \|\mathbf{A}_0\|_0 \ll FT$, $\|\cdot\|_0$ counts the nonzero entries of its matrix argument). Given a compression matrix $\mathbf{R} \in \mathbb{R}^{L\times F}$ with $L \leq F$, and observations

$$\mathbf{Y} = \mathbf{X}_0 + \mathbf{R}\mathbf{A}_0 \tag{2.1}$$

the present paper deals with the recovery of $\{\mathbf{X}_0, \mathbf{A}_0\}$. This task is of interest e.g., to unveil anomalous flows in backbone networks [72, 94, 158], to reduce the data acquisition time in cardiac magnetic resonance imaging (MRI) [52, 53], or, to separate singing voice from its music accompaniment [59,132]; see also Section 2.2 on motivating applications. In addition, this fundamental problem is met at the crossroads of compressive sampling (CS), and the timely low-rank-plus-sparse matrix decompositions.

In the absence of the low-rank component ($\mathbf{X}_0 = \mathbf{0}_{L\times T}$), one is left with an underdetermined sparse signal recovery problem; see e.g., [29, 114] and the tutorial account [30]. When $\mathbf{Y} = \mathbf{X}_0 + \mathbf{A}_0$, the formulation boils down to principal component pursuit (PCP), also referred to as robust principal component analysis (PCA) [25, 33, 34, 43]. For this

idealized noise-free setting, sufficient conditions for exact recovery are available for both of the aforementioned special cases; see also [34] for state-of-the-art PCP recovery guarantees, even valid when only a subset of $\mathbf{Y}$'s entries are observed. However, the superposition of a low-rank and a *compressed* sparse matrix in (2.1) further challenges identifiability of $\{\mathbf{X}_0, \mathbf{A}_0\}$. Along these lines, the *compressive* PCP formulation in [148] aims at recovering a target matrix that is a superposition of low-rank and sparse components, from a (small) set of linear measurements; see also [7] for a related approach. In the presence of 'dense' noise, stable reconstruction of the low-rank and sparse matrix components is possible via PCP [151, 163]. Earlier efforts dealing with the recovery of sparse vectors in noise led to similar performance guarantees; see e.g., [19] and references therein. Even when $\mathbf{X}_0$ is nonzero, one could envision a CS variant where the measurements are corrupted with correlated (low-rank) noise [36]. Last but not least, when $\mathbf{A}_0 = \mathbf{0}_{F\times T}$ and $\mathbf{Y}$ is noisy, the recovery of $\mathbf{X}_0$ subject to a rank constraint is nothing else than PCA – arguably, the workhorse of high-dimensional data analysis [64].

In this respect, our main contribution is to establish that given $\mathbf{Y}$ and $\mathbf{R}$ in (2.1), for small enough r and s one can *exactly* recover $\{\mathbf{X}_0, \mathbf{A}_0\}$ by solving the nonsmooth *convex* optimization problem

$$\text{(P1)} \qquad \min_{\{\mathbf{X},\mathbf{A}\}} \quad \|\mathbf{X}\|_* + \lambda\|\mathbf{A}\|_1, \quad \text{s. to} \quad \mathbf{Y} = \mathbf{X} + \mathbf{R}\mathbf{A}$$

where $\lambda \geq 0$ is a tuning parameter; $\|\mathbf{X}\|_* := \sum_i \sigma_i(\mathbf{X})$ is the nuclear norm of $\mathbf{X}$ (σ_i stands for the i-th singular value); and, $\|\mathbf{X}\|_1 := \sum_{i,j} |x_{ij}|$ denotes the ℓ_1-norm. The aforementioned norms are convex surrogates to the rank and ℓ_0-norm, respectively, which albeit natural as criteria they are NP-hard to optimize [37,109]. Recently, a greedy algorithm for recovering low-rank and sparse matrices from compressive measurements was put forth in [147]. However, convergence of the algorithm and its error performance are only assessed via numerical simulations. A recursive online algorithm can be found in [36], which attains good performance in practice but does not offer theoretical guarantees; see also [96].

A *deterministic* approach along the lines of [33] is adopted first to derive conditions under which (2.1) is locally identifiable (Section 2.3). Introducing a notion of incoherence between the additive components $\mathbf{X}_0$ and $\mathbf{R}\mathbf{A}_0$, and resorting to the restricted isometry

constants of $\mathbf{R}$ [29], sufficient conditions are obtained to ensure that (P1) succeeds in exactly recovering the unknowns (Section 2.4.1). Intuitively, the results here assert that if r and s are sufficiently small, the nonzero entries of $\mathbf{A}_0$ are sufficiently spread out, and subsets of columns of $\mathbf{R}$ behave as isometries, then (P1) exactly recovers $\{\mathbf{X}_0, \mathbf{A}_0\}$. As a byproduct, recovery results for PCP and CS are also obtained by specializing the aforesaid conditions accordingly (Section However, these induced recovery guarantees are weaker than those recently obtained for PCP and CS by relying on state-of-the-art analysis techniques tailored to these specific problems; see e.g., [34, 114], and references therein. The proof of the main result builds on Lagrangian duality theory [17, 22], to first derive conditions under which $\{\mathbf{X}_0, \mathbf{A}_0\}$ is the *unique* optimal solution of (P1) (Section 2.5.1). In a nutshell, satisfaction of the optimality conditions is tantamount to the existence of a valid dual certificate. Stemming from the unique challenges introduced by $\mathbf{R}$, the dual certificate construction procedure of Section 2.5.2 is markedly distinct from the direct sum approach in [33], and the (random) golfing scheme of [25]. Section 2.6 shows that low-rank, sparse, and compression matrices drawn from certain random ensembles satisfy the sufficient conditions for exact recovery with high probability.

Two batch iterative algorithms for solving (P1) are developed in Section 2.7, based on the accelerated proximal grandient (APG) method [14, 80, 110, 111], and the alternating-direction method of multipliers (AD-MoM) [18, 22]. Decentralized and online algorithms were put forth in the companion papers [95] and [96]. These are useful when rows of $\mathbf{Y}$ are distributed over a network, and for real-time processing of streaming data (columns of $\mathbf{Y}$), respectively. Numerical tests corroborate the exact recovery claims, and the effectiveness of (P1) in unveiling traffic volume anomalies from real network data (Section 2.8). While the obtained sufficient conditions for exact recovery may be violated in the anomaly detection context of Section 2.8.2, the encouraging results obtained in Section 2.8.2 suggest that there is room for improving these conditions. Section 2.9 concludes the paper with a summary and a discussion of limitations, possible extensions, and interesting future directions. Technical details are deferred to the Appendix.

2.2 Applications

This section outlines several application domains that involve decomposing a data matrix as in (2.1).

2.2.1 Unveiling network anomalies via sparsity and low rank

In the backbone of large-scale networks, origin-to-destination (OD) traffic flows experience abrupt changes which can result in congestion, and limit the quality of service provisioning of the end users. These so-termed *traffic volume anomalies* can be due to external sources such as network failures, denial of service attacks, or, intruders hijacking the network services [137], [72], [158]. 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 comprise the superposition of *unobservable* OD flows as explained next.

Consider a backbone 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. The network transports F end-to-end flows associated with specific OD pairs. For backbone networks, the number of network layer flows is typically much larger than the number of physical links ($F \gg L$). Single-path routing is considered here to send the traffic flow from a source to its intended destination. Accordingly, for a particular flow multiple links connecting the corresponding OD pair are chosen to carry the traffic. Sparing details that can be found in [94], the traffic $\mathbf{Y} := [y_{l,t}] \in \mathbb{R}^{L \times T}$ carried over links $l \in \mathcal{L}$ and measured at time instants $t \in [1, T]$, can be compactly expressed as

$$\mathbf{Y} = \mathbf{R}\,(\mathbf{Z} + \mathbf{A}) + \mathbf{E} \tag{2.2}$$

where the fat routing matrix $\mathbf{R} := [r_{\ell,f}] \in \{0, 1\}^{L \times F}$ is fixed and given, $\mathbf{Z} := [z_{f,t}]$ denotes the unknown 'clean' traffic flows over the time horizon of interest, $\mathbf{A} := [a_{f,t}]$ collects the traffic volume anomalies across flows and time, and $\mathbf{E} := [e_{l,t}]$ captures measurement errors.

Common temporal patterns among the traffic flows in addition to their periodic behavior, render most rows (respectively columns) of $\mathbf{Z}$ linearly dependent, and thus $\mathbf{Z}$ typically

has low rank [72, 120]. Anomalies are expected to occur sporadically over time, and only last for short periods relative to the (possibly long) measurement interval $[1, T]$. In addition, only a small fraction of the flows are supposed to be anomalous at any given time instant. This renders the anomaly matrix $\mathbf{A}$ sparse across rows and columns. Given link measurements $\mathbf{Y}$ and the routing matrix $\mathbf{R}$, the goal is to estimate $\mathbf{A}$ by capitalizing on the sparsity of $\mathbf{A}$ and the low-rank property of $\mathbf{Z}$. Since the primary goal is to recover $\mathbf{A}$, define $\mathbf{X} := \mathbf{RZ}$ which inherits the low-rank property from $\mathbf{Z}$, and consider

$$\mathbf{Y} = \mathbf{X} + \mathbf{RA} + \mathbf{E} \tag{2.3}$$

which is identical to (2.1) modulo small measurement errors in $\mathbf{E} \in \mathbb{R}^{L \times T}$. If $\mathbf{E} = \mathbf{0}_{L \times T}$, then (P1) can be used to unveil network anomalies, whereas the algorithm outlined in Section 2.7.1 is more suitable for the noisy setting.

By adopting the model (2.3), one is neglecting the structure $\mathbf{X} := \mathbf{RZ}$. However, it is otherwise not clear how could one efficiently estimate $\mathbf{Z}$ and $\mathbf{A}$ from measurements as in (2.2), which is a more difficult problem. The compressive PCP approach [148] deals with recovery of $\{\mathbf{Z}, \mathbf{A}\}$ from measurements $\tilde{\mathbf{Y}} = \mathcal{P}_Q(\mathbf{Z} + \mathbf{A}) \in \mathbb{R}^{F \times T}$, where $\mathcal{P}_Q(\cdot)$ denotes orthogonal projection onto a linear subspace $Q \subseteq \mathbb{R}^{F \times T}$. Note that compressive PCP cannot be adopted here since it requires $\{\mathbf{Z}, \mathbf{A}\}$ to be sufficiently incoherent with the orthogonal subspace $Q^{\perp}$, a condition which is violated in (2.2) since $Q^{\perp}$ is the nullspace of the fat compression matrix $\mathbf{R}$.

2.2.2 Dynamic magnetic resonance imagery

As a result of the existing limitations in magnetic resonance imaging (MRI) data-acquisition time, respiratory motions can severely degrade the quality of MRI. Consequently, this can result in e.g., dose-delivery errors for patients subjected to radiation therapy [150]. *Dynamic* MRI aims at resolving the variations of the imaged object by reconstructing a temporal series of 'ground truth' images [87]. As an illustrative example, consider cardiac MRI which nowadays serves as a major imaging modality for noninvasive diagnosis of heart diseases in clinic practice [49]. A critical specification of cardiac MRI is the simultaneous realization

of higher spatial and temporal resolution. This in turn necessitates longer data-acquisition periods, which are however limited by the patient's breath-holding time. Inspired by the low intrinsic-dimensionality of (cardiac) MRI images [52], devising efficient techniques to reduce the acquisition time for a prescribed image quality becomes an important issue.

Consider each 'ground truth' cardiac snapshot as a piecewise-constantly discretized image of P pixels. Each image can be modeled as a superposition of a *background* component and a *motion* component [52, 53]. The background component refers to the temporally stationary or slowly-varying part of the acquired images. Moreover, the motion component captures the rapidly-changing pixels due to heart beating. The spatial structure of the heart has motivated the adoption of models involving a (possibly learnt and overcomplete) dictionary, under which the motion component admits a sparse representation based on few atoms (columns) of this dictionary [52, 53]. Let $\mathbf{x}_t \in \mathbb{R}^P$ denote the background component of the dynamic MRI frame acquired at time t, and let $\mathbf{D}\mathbf{a}_t \in \mathbb{R}^P$ denote the motion component, where $\mathbf{D} \in \mathbb{R}^{P\times F}$ is a given overcomplete dictionary, and $\mathbf{a}_t$ a sparse vector of coefficients. The MRI acquisition procedure entails measuring Fourier coefficients of the image, and only a subset of size $L \leq P$ of Fourier coefficients is sampled to reduce the data acquisition time. Accordingly, the partial FFT matrix $\mathbf{\Psi} \in \mathbb{R}^{L\times P}$ containing a row-subset of cardinality L of the full FFT $P \times P$ matrix, maps the image to a subset of its Fourier coefficients. The scanned temporal sequence of images in the frequency domain can thus be modeled as

$$\mathbf{y}_t = \mathbf{\Psi}(\mathbf{z}_t + \mathbf{D}\mathbf{a}_t) + \mathbf{v}_t, \quad t = 1, \ldots, T \tag{2.4}$$

where $\mathbf{v}_t$ accounts for modeling and measurement errors. Collect the components $\{\mathbf{x}_t := \mathbf{\Psi}\mathbf{z}_t\}_{t=1}^T$ and $\{\mathbf{a}_t\}_{t=1}^T$ as columns of the matrices $\mathbf{X}$ and $\mathbf{A}$, respectively; and recognize that (2.4) boils down to (2.1) upon defining $\mathbf{R} := \mathbf{\Psi}\mathbf{D}$. Notice that it suffices to estimate $\mathbf{X}$ (rather that $\mathbf{Z}$) since in cardiac MRI the main objective is to reconstruct the motion component $\mathbf{DA}$, which offers valuable information to physicians about possible heart diseases. By the very definition of background component, the sought matrix $\mathbf{X}$ is low rank. Also, $\mathbf{A}$ is sparse by construction of the dictionary $\mathbf{D}$. All in all, adopting (P1) to recover $\mathbf{A}$ and subsequently the motion component $\mathbf{DA}$ is well motivated.

2.2.3 Face recognition

Accurately estimating the low-dimensional subspace of a human's facial images is an important task in computer vision, with application to face recognition [12]. In this context, a robust approach is needed since facial images in the training set tend to be exposed to different illuminations, and typically suffer from specularities as well as self-shadowing (e.g., around the nose and eyes' areas). Similar to the dynamic MRI setup, a reasonable model represents each image as the superposition of a background (shadow-free face) component $\mathbf{X}$ which has low rank, and the error (shadow) component which is highly structured and localized. Model (2.1) is naturally aligned with this decomposition, upon learning a (possibly overcomplete) dictionary $\mathbf{D} := \mathbf{R}$ under which the error component $\mathbf{A}$ is sparsely represented. While PCP has been adopted in [25] to remove shadows and specularities from face images, (2.1) offers a more general alternative. This is because PCP presumes the sparse errors are independently scattered across the face image. However, this assumption neglects the fact that shadows and specularities usually contain certain spatial structure, which can be better modeled via a suitably learned dictionary of atoms.

2.2.4 Separation of singing voice from its music accompaniment

Separation of singing voice from its music accompaniment has wide applicability in areas such as automatic lyrics recognition and alignment, singer identification, and music information retrieval [76]. Even though this is an effortless task for the human auditory system, it is difficult for machines [59]. Let $\mathbf{Y}$ denote the spectrogram of a given song, which can be naturally modeled as the superposition of music plus singing-voice components. Due to the repetitious nature of music accompaniment, the music component $\mathbf{X}$ has low rank [59, 132]. In contrast, the singing voice exhibits higher variability, but as it is customary for speech signals [61], it can be reasonably assumed sparsely expressible over a proper dictionary $\mathbf{D} := \mathbf{R}$ of sounds. In a nutshell, (P1) can be adopted to carry out this decomposition task, while incorporating non-negativity constraints on the matrix components is a natural extension since the spectrogram is inherently non-negative [132].

2.3 Local Identifiability

The first issue to address is model identifiability, meaning that there are *unique* low-rank and sparse matrices satisfying (2.1). If there exist multiple decompositions of $\mathbf{Y}$ into $\mathbf{X}+\mathbf{RA}$ with low-rank $\mathbf{X}$ and sparse $\mathbf{A}$, there is no hope of recovering $\{\mathbf{X}_0, \mathbf{A}_0\}$ from the data. For instance, if the null space of the fat matrix $\mathbf{R}$ contains sparse matrices, there may exist a sparse perturbation $\mathbf{H}$ such that $\mathbf{A}_0 + \mathbf{H}$ is still sparse and $\{\mathbf{X}_0, \mathbf{A}_0 + \mathbf{H}\}$ is a legitimate solution. Another problematic case arises when there is a sparse perturbation $\mathbf{H}$ such that $\mathbf{RH}$ is spanned by the row or column spaces of $\mathbf{X}_0$. Then, $\mathbf{X}_0 + \mathbf{RH}$ has the same rank as $\mathbf{X}_0$ and $\mathbf{A}_0 - \mathbf{H}$ may still be sparse. As a result, one may pick $\{\mathbf{X}_0 + \mathbf{RH}, \mathbf{A}_0 - \mathbf{H}\}$ as another valid solution. Dealing with such identifiability issues is the subject of this section.

Let $\mathbf{U\Sigma V'}$ denote the singular value decomposition (SVD) of $\mathbf{X}_0$, and consider the subspaces: s1) $\Phi(\mathbf{X}_0) := \{\mathbf{Z} \in \mathbb{R}^{L\times T} : \mathbf{Z} = \mathbf{UW}_1' + \mathbf{W}_2\mathbf{V}', \ \mathbf{W}_1 \in \mathbb{R}^{T\times r}, \ \mathbf{W}_2 \in \mathbb{R}^{L\times r}\}$ of the span of all matrices with either the same column space or row space as $\mathbf{X}_0$; s2) $\Omega(\mathbf{A}_0) := \{\mathbf{H} \in \mathbb{R}^{F\times T} : \text{supp}(\mathbf{H}) \subseteq \text{supp}(\mathbf{A}_0)\}$ of matrices in $\mathbb{R}^{F\times T}$ with support contained in the support of $\mathbf{A}_0$; and s3) $\Omega_R(\mathbf{A}_0) := \{\mathbf{Z} \in \mathbb{R}^{L\times T} : \mathbf{Z} = \mathbf{RH}, \ \mathbf{H} \in \Omega(\mathbf{A}_0)\}$. For notational brevity, s1)-s3) will be henceforth denoted as $\{\Phi, \Omega, \Omega_R\}$. Noteworthy properties of these subspaces are: i) both Φ and $\Omega_R \subset \mathbb{R}^{L\times T}$, hence it is possible to directly compare elements from them; ii) $\mathbf{X}_0 \in \Phi$ and $\mathbf{RA}_0 \in \Omega_R$; and iii) if $\mathbf{Z} \in \Phi^{\perp}$ is added to $\mathbf{X}_0$, then $\text{rank}(\mathbf{Z} + \mathbf{X}_0) > r$.

For now, assume that the subspaces Ω_R and Φ are also known. This extra information helps identifiability of (2.1), because potentially troublesome solutions $\{\mathbf{X}_0 + \mathbf{RH}, \mathbf{A}_0 - \mathbf{H}\}$ are limited to a restricted class. If $\mathbf{X}_0 + \mathbf{RH} \notin \Phi$ or $\mathbf{A}_0 - \mathbf{H} \notin \Omega$, that candidate solution is not admissible since it is known a priori that $\mathbf{A}_0 \in \Omega$ and $\mathbf{X}_0 \in \Phi$. Under these assumptions, the following lemma puts forth the necessary and sufficient conditions guaranteeing the existence of a *unique* pair of matrices $\{\mathbf{X}_0 \in \Phi, \mathbf{A}_0 \in \Omega_R\}$, such that $\mathbf{Y}$ can be decomposed according to (2.1) – a notion known as *local identifiability* [25, 33].

Lemma 2.1 *Given subspaces* $\{\Phi, \Omega, \Omega_R\}$ *and matrices* $\{\mathbf{Y}, \mathbf{R}\}$*, there is a unique pair* $\{\mathbf{X}_0 \in \Phi, \mathbf{A}_0 \in \Omega_R\}$ *such that* $\mathbf{Y} = \mathbf{X}_0 + \mathbf{RA}_0$ *if and only if* $\Phi \cap \Omega_R = \{\mathbf{0}_{L\times T}\}$*, and*

$\mathbf{RH} \neq \mathbf{0}_{L\times T}, \forall \mathbf{H} \in \Omega \backslash \{\mathbf{0}_{F\times T}\}$.

Proof: Since by definition $\mathbf{X}_0 \in \Phi$ and $\mathbf{A}_0 \in \Omega$, one can represent every element in the *subspaces* Φ and Ω_R as $\mathbf{X}_0 + \mathbf{Z}_1$ and $\mathbf{RA}_0 + \mathbf{Z}_2$, respectively, where $\mathbf{Z}_1 \in \Phi$ and $\mathbf{Z}_2 \in \Omega_R$. Assume that $\Phi \cap \Omega_R = \{\mathbf{0}_{L\times T}\}$, and suppose by contradiction that there exist *nonzero* perturbations $\{\mathbf{Z}_1, \mathbf{Z}_2\}$ such that $\mathbf{Y} = \mathbf{X}_0 + \mathbf{Z}_1 + \mathbf{RA}_0 + \mathbf{Z}_2$. Then, $\mathbf{Z}_1 + \mathbf{Z}_2 = \mathbf{0}_{L\times T}$, meaning that $\mathbf{Z}_1$ and $\mathbf{Z}_2$ belong to the same subspace, which contradicts the assumption. Conversely, suppose there exists a non-zero $\mathbf{Z} \in \Omega_R \cap \Phi$. Clearly, $\{\mathbf{X}_0 + \mathbf{Z}, \mathbf{RA}_0 - \mathbf{Z}\}$ is a feasible solution where $\mathbf{X}_0 + \mathbf{Z} \in \Phi$ and $\mathbf{RA}_0 - \mathbf{Z} \in \Omega_R$. This contradicts the uniqueness assumption. In addition, the condition $\mathbf{RH} \neq \mathbf{0}, \mathbf{H} \in \Omega \backslash \{\mathbf{0}_{L\times T}\}$ ensures that $\mathbf{Z} = \mathbf{0}_{L\times T} \in \Phi \cap \Omega_R$ only when $\mathbf{Z} = \mathbf{RH} = \mathbf{0}_{L\times T}$ for $\mathbf{H} = \mathbf{0}_{F\times T}$.

In words, (2.1) is locally identifiable if and only if the subspaces Φ and Ω_R intersect transversally, and the sparse matrices in Ω are not annihilated by $\mathbf{R}$. This last condition is unique to the setting here, and is not present in [25] or [33].

Remark 2.1 (Orthogonal projection operators) Operator $\mathcal{P}_\Omega(\mathbf{X})$ ($\mathcal{P}_{\Omega^\perp}(\mathbf{X})$) denotes the orthogonal projection of $\mathbf{X}$ onto the subspace Ω (orthogonal complement $\Omega^\perp$). It simply sets those elements of $\mathbf{X}$ not in $\text{supp}(\mathbf{A}_0)$ to zero. Likewise, $\mathcal{P}_\Phi(\mathbf{X})$ ($\mathcal{P}_{\Phi^\perp}(\mathbf{X})$) denotes the orthogonal projection of $\mathbf{X}$ onto the subspace Φ (orthogonal complement $\Phi^\perp$). Let $\mathbf{P}_U := \mathbf{UU}'$ and $\mathbf{P}_V := \mathbf{VV}'$ denote, respectively, projection onto the column and row spaces of $\mathbf{X}_0$. It can be shown that $\mathcal{P}_\Phi(\mathbf{X}) = \mathbf{P}_U\mathbf{X} + \mathbf{XP}_V - \mathbf{P}_U\mathbf{XP}_V$, while the projection onto the complement subspace is $\mathcal{P}_{\Phi^\perp}(\mathbf{X}) = (\mathbf{I} - \mathbf{P}_U)\mathbf{X}(\mathbf{I} - \mathbf{P}_V)$. In addition, the following identities

$$\langle \mathcal{P}_\Phi(\mathbf{X}), \mathcal{P}_\Phi(\mathbf{Y}) \rangle = \langle \mathcal{P}_\Phi(\mathbf{X}), \mathbf{Y} \rangle = \langle \mathbf{X}, \mathcal{P}_\Phi(\mathbf{Y}) \rangle \tag{2.5}$$

of orthogonal projection operators such as $\mathcal{P}_\Phi(\cdot)$, will be invoked throughout the paper.

2.3.1 Incoherence measures

Building on Lemma 2.1, alternative sufficient conditions are derived here to ensure local identifiability. To quantify the overlap between Φ and Ω_R, consider the *incoherence* param-

eter

$$\mu(\Omega_R, \Phi) = \max_{\mathbf{Z} \in \Omega_R \backslash \{\mathbf{0}\}} \frac{\|\mathcal{P}_\Phi(\mathbf{Z})\|_F}{\|\mathbf{Z}\|_F}. \tag{2.6}$$

for which it holds that $\mu(\Omega_R, \Phi) \in [0, 1]$. The lower bound is achieved when Φ and Ω_R are orthogonal, while the upper bound is attained when $\Phi \cap \Omega_R$ contains a nonzero element. Assuming $\Phi \cap \Omega_R = \{\mathbf{0}_{L\times T}\}$, then $\mu(\Omega_R, \Phi) < 1$ represents the cosine of the angle between Φ and Ω_R [42]. From Lemma 2.1, it appears that $\mu(\Omega_R, \Phi) < 1$ guarantees $\Phi \cap \Omega_R = \{\mathbf{0}_{L\times T}\}$. As it will become clear later on, tighter conditions on $\mu(\Omega_R, \Phi)$ will prove instrumental to guarantee exact recovery of $\{\mathbf{X}_0, \mathbf{A}_0\}$ by solving (P1).

To measure the incoherence among subsets of columns of $\mathbf{R}$, which is tightly related to the second condition in Lemma 2.1, the restricted isometry constants (RICs) come handy [29]. The constant $\delta_k(\mathbf{R})$ measures the extent to which a k-subset of columns of $\mathbf{R}$ behaves like an isometry. It is defined as the smallest value satisfying

$$c(1 - \delta_k(\mathbf{R})) \leq \frac{\|\mathbf{R}\mathbf{u}\|^2}{\|\mathbf{u}\|^2} \leq c(1 + \delta_k(\mathbf{R})) \tag{2.7}$$

for every $\mathbf{u} \in \mathbb{R}^F$ with $\|\mathbf{u}\|_0 \leq k$ and for some positive normalization constant $c < 1$ [29]. For later use, introduce $\theta_{s_1,s_2}(\mathbf{R})$ which measures 'how orthogonal' are the subspaces generated by two disjoint column subsets of $\mathbf{R}$, with cardinality s_1 and s_2. Formally, $\theta_{s_1,s_2}(\mathbf{R})$ is the smallest value that satisfies

$$|\langle \mathbf{R}\mathbf{u}_1, \mathbf{R}\mathbf{u}_2 \rangle| \leq c\theta_{s_1,s_2}(\mathbf{R})\|\mathbf{u}_1\|\|\mathbf{u}_2\| \tag{2.8}$$

for every $\mathbf{u}_1, \mathbf{u}_2 \in \mathbb{R}^F$, where $\text{supp}(\mathbf{u}_1) \cap \text{supp}(\mathbf{u}_2) = \emptyset$ and $\|\mathbf{u}_1\|_0 \leq s_1, \|\mathbf{u}_2\|_0 \leq s_2$. The normalization constant c plays the same role as in $\delta_k(\mathbf{R})$. A wide family of matrices with small RICs have been introduced in e.g., [29].

All the elements are now in place to state this section's main result.

Proposition 2.1 *Assume that each column of* $\mathbf{A}_0$ *contains at most* k *nonzero elements. If* $\mu(\Omega_R, \Phi) < 1$ *and* $\delta_k(\mathbf{R}) < 1$*, then* $\Omega_R \cap \Phi = \{\mathbf{0}_{L\times T}\}$ *and* $\mathbf{R}\mathbf{H} \neq \mathbf{0}_{L\times T}, \mathbf{H} \in \Omega \backslash \{\mathbf{0}_{F\times T}\}$.

Proof: Suppose the intersection in nontrivial, meaning that there exists nonzero matrices $\mathbf{H} \in \Omega$ and $\mathbf{U}\mathbf{W}_1' + \mathbf{W}_2\mathbf{V}' \in \Phi$ satisfying $\mathbf{R}\mathbf{H} = \mathbf{U}\mathbf{W}_1' + \mathbf{W}_2\mathbf{V}'$. Vectorizing the last

equation and relying on the identity $\text{vec}(\mathbf{AXB}) = (\mathbf{B}' \otimes \mathbf{A})\text{vec}(\mathbf{X})$, one obtains a linear system of equations

$$[\mathbf{I}_T \otimes \mathbf{R} \ -\mathbf{I}_T \otimes \mathbf{U} \ -\mathbf{V} \otimes \mathbf{I}_L]\mathbf{w} = \mathbf{0}_{LT} \tag{2.9}$$

where $\mathbf{w} := [\text{vec}(\mathbf{H})' \ \text{vec}(\mathbf{W}_1') \ \text{vec}(\mathbf{W}_2')]'$. Define an $LT \times FT$ matrix $\mathbf{C}_1 := \mathbf{I}_T \otimes \mathbf{R}$ and the $LT \times (L+T)r$ matrix $\mathbf{C}_2 := [-\mathbf{I}_T \otimes \mathbf{U} \ -\mathbf{V} \otimes \mathbf{I}_L]$. The corresponding coefficients are $\mathbf{w}_1 := \text{vec}(\mathbf{H})$ and $\mathbf{w}_2 := [\text{vec}(\mathbf{W}_1')' \text{vec}(\mathbf{W}_2')]'$. Then, (2.9) implies there exists a $\mathbf{w}_1 \neq \mathbf{0}_{FT}$ such that $\mathbf{C}_1\mathbf{w}_1 + \mathbf{C}_2\mathbf{w}_2 = \mathbf{0}_{LT}$.

Consider two cases: i) $\mathbf{w}_2 = \mathbf{0}_{r(L+T)}$, and ii) $\mathbf{w}_2 \neq \mathbf{0}_{r(L+T)}$. Under i) $\mathbf{C}_1\mathbf{w}_1 = \mathbf{0}_{LT}$, and thus $\mathbf{R}\mathbf{w}_1^{(i)} = \mathbf{0}$ for some nonzero $\mathbf{w}_1^{(i)}$ with $i \in \{1, 2, ..., T\}$ where $\mathbf{w}_1 = [\mathbf{w}_1^{(1)} ... \mathbf{w}_1^{(T)}]$. Therefore, if $\|\mathbf{w}_1^{(i)}\|_0 \leq k$, $\delta_k(\mathbf{R}) < 1$ implies that $\mathbf{w}_1^{(i)} = \mathbf{0}_{LT}$, which is a contradiction. For ii) $\mu(\Omega_R, \Phi) < 1$ implies that there is no $\mathbf{w}_1$ with $\text{supp}(\mathbf{w}_1) \subseteq \text{supp}(\text{vec}(\mathbf{A}_0))$ and $\mathbf{w}_2 \in \mathbb{R}^{(L+T)r}$ such that $\mathbf{C}_1\mathbf{w}_1 + \mathbf{C}_2\mathbf{w}_2 = \mathbf{0}_{FT}$, since otherwise $|\langle \mathbf{C}_1\mathbf{w}_1, \mathbf{C}_2\mathbf{w}_2 \rangle| = \|\mathbf{C}_1\mathbf{w}_1\| \|\mathbf{C}_2\mathbf{w}_2\|$ which leads to $\mu(\Omega_R, \Phi) = 1$.

2.4 Exact Recovery via Convex Optimization

In addition to $\mu(\Omega_R, \Phi)$, there are other incoherence measures which play an important role in the conditions for exact recovery. Consider a feasible solution $\{\mathbf{X}_0 + a_{ij}\mathbf{R}\mathbf{e}_i\mathbf{e}_j', \mathbf{A}_0 - a_{ij}\mathbf{e}_i\mathbf{e}_j'\}$, where $(i,j) \notin \text{supp}(\mathbf{A}_0)$ and thus $a_{ij}\mathbf{e}_i\mathbf{e}_j' \notin \Omega$. It may then happen that $a_{ij}\mathbf{R}\mathbf{e}_i\mathbf{e}_j' \in \Phi$ and $\text{rank}(\mathbf{X}_0 + a_{ij}\mathbf{R}\mathbf{e}_i\mathbf{e}_j') = \text{rank}(\mathbf{X}_0) - 1$, while $\|\mathbf{A}_0 - a_{ij}\mathbf{e}_i\mathbf{e}_j'\|_0 = \|\mathbf{A}_0\|_0 + 1$, challenging identifiability when Φ and Ω_R are unknown. Similar complications will arise if $\mathbf{X}_0$ has a sparse row space that could be confused with the row space of $\mathbf{A}_0$. These issues motivate defining (recall $\mathbf{X}_0 = \mathbf{U}\mathbf{\Sigma}\mathbf{V}'$)

$$\gamma_R(\mathbf{U}) := \max_{i,j} \frac{\|\mathbf{P}_U\mathbf{R}\mathbf{e}_i\mathbf{e}_j'\|_F}{\|\mathbf{R}\mathbf{e}_i\mathbf{e}_j'\|_F}, \quad \gamma(\mathbf{V}) := \max_i \|\mathbf{P}_V\mathbf{e}_i\|_F$$

where $\gamma_R(\mathbf{U}), \gamma(\mathbf{V}) \leq 1$. The maximum of $\gamma_R(\mathbf{U})$ $[\gamma(\mathbf{V})]$ is attained when $\mathbf{R}\mathbf{e}_i\mathbf{e}_j'$ $[\mathbf{e}_i]$ is in the column [row] space of $\mathbf{X}_0$ for some (i,j). Small values of $\gamma_R(\mathbf{U})$ and $\gamma(\mathbf{V})$ imply that the column and row spaces of $\mathbf{X}_0$ do not contain the columns of $\mathbf{R}$ and sparse vectors, respectively.

Another identifiability issue arises when $\mathbf{X}_0 = \mathbf{RH}$ for some sparse matrix $\mathbf{H} \in \Omega$. In this case, each column of $\mathbf{X}_0$ is spanned by a few columns of $\mathbf{R}$. Consider the parameter

$$\xi_R(\mathbf{U}, \mathbf{V}) := \|\mathbf{R}'\mathbf{U}\mathbf{V}'\|_\infty = \max_{i,j} |\mathbf{e}_i{}'\mathbf{R}'\mathbf{U}\mathbf{V}\mathbf{e}_j|.$$

A small value of $\xi_R(\mathbf{U}, \mathbf{V})$ implies that each column of $\mathbf{X}_0$ is spanned by sufficiently many columns of $\mathbf{R}$. To understand this property, consider the SVD $\mathbf{X}_0 = \mathbf{U}\mathbf{\Sigma}\mathbf{V}' = \sum_{i=1}^r \sigma_i \mathbf{u}_i \mathbf{v}_i'$. The k-th column of $\mathbf{X}_0$ is then $\sum_{i=1}^r \sigma_i \mathbf{u}_i v_{i,k}$, and its projection onto the l-th column of $\mathbf{R}$ is

$$\Big|\langle \mathbf{R}\mathbf{e}_l, \sum_{i=1}^r \sigma_i \mathbf{u}_i v_{i,k} \rangle\Big| = \Big|\sum_{i=1}^r \langle \mathbf{R}\mathbf{e}_l, \mathbf{u}_i \rangle \sigma_i v_{i,k}\Big| \leq \sigma_{\max} \xi_R(\mathbf{U}, \mathbf{V})$$

where $\sigma_{\max}$ is the largest singular value of $\mathbf{X}_0$. Since the energy of $\sum_{i=1}^r \sigma_i \mathbf{u}_i v_{i,k}$ is somehow allocated along the directions $\mathbf{R}\mathbf{e}_l$, if all the aforementioned projections can be made arbitrarily small, then sufficiently many nonzero terms in the expansion are needed to account for all this energy.

2.4.1 Main result

Theorem 2.1 *Consider given matrices $\mathbf{Y} \in \mathbb{R}^{L\times T}$ and $\mathbf{R} \in \mathbb{R}^{L\times F}$ obeying $\mathbf{Y} = \mathbf{X}_0 + \mathbf{R}\mathbf{A}_0 = \mathbf{U}\mathbf{\Sigma}\mathbf{V}' + \mathbf{R}\mathbf{A}_0$, with $r := rank(\mathbf{X}_0)$ and $s := \|\mathbf{A}_0\|_0$. Assume that every row and column of $\mathbf{A}_0$ has at most k nonzero elements, and that $\mathbf{R}$ has orthonormal rows. If the following conditions*

I) $\chi := \omega[(1 - \mu(\Phi, \Omega_R))^2(1 - \delta_k(\mathbf{R}))]^{-1} < 1/2$*; and*

II) $\lambda_{max} := \sqrt{s^{-1}}\Big[\alpha^{-1} - \mu(\Phi, \Omega_R)\sqrt{rc(1 + \delta_k(\mathbf{R}))}\Big] > \lambda_{min} := \beta\xi_R(\mathbf{U}, \mathbf{V})$

hold, where

$$\begin{aligned}\omega &:= \theta_{1,1}(\mathbf{R})[\sqrt{2}k + s\gamma^2(\mathbf{V})] \\ &\quad + (1 + \delta_1(\mathbf{R}))\left[\sqrt{2}k\gamma_R^2(\mathbf{U}) + k\gamma^2(\mathbf{V}) + s\gamma_R^2(\mathbf{U})\gamma^2(\mathbf{V})\right] \\ \alpha &:= 1 + \left(c^{-1}\omega^{-1}\chi - 1\right)^{1/2}, \quad \beta := (1 - 2\chi)^{-1}\end{aligned}$$

then there exists $\lambda \in (\lambda_{\min}, \lambda_{\max})$ for which the convex program (P1) exactly recovers $\{\mathbf{X}_0, \mathbf{A}_0\}$.

Note that I) alone is already more stringent than the pair of conditions $\mu(\Omega_R, \Phi) < 1$ and $\delta_k(\mathbf{R}) < 1$ needed for local identifiability (cf. Proposition 2.1). Satisfaction of the conditions in Theorem 2.1 hinges upon the values of the incoherence parameters $\mu(\Omega_R, \Phi), \gamma_R(\mathbf{U}), \gamma(\mathbf{V})$, $\xi_R(\mathbf{U}, \mathbf{V})$, and the RICs $\delta_k(\mathbf{R})$ and $\theta_{1,1}(\mathbf{R})$. In particular, $\{\omega, \alpha, \beta, \chi\}$ are increasing functions of these parameters, and it is readily observed from I) and II) that the smaller $\{\omega, \alpha, \beta\}$ are, the more likely the conditions are met. Furthermore, the incoherence parameters are increasing functions of the rank r and sparsity level s. The RIC $\delta_k(\mathbf{R})$ is also an increasing function of k, the maximum number of nonzero elements per row/column of $\mathbf{A}_0$. Therefore, for sufficiently small values of $\{r, s, k\}$, the sufficient conditions of Theorem 2.1 can be indeed satisfied.

It is worth noting that not only s, but also the position of the nonzero entries in $\mathbf{A}_0$ plays an important role in satisfying I) and II). This is manifested through k, for which a small value indicates the entries of $\mathbf{A}_0$ are sufficiently spread out, i.e., most entries do not cluster along a few rows or columns of $\mathbf{A}_0$. Moreover, no restriction is placed on the magnitude of these entries, since as seen later on it is only the positions that affect optimal recovery via (P1).

Remark 2.2 (Row orthonormality of R) Assuming $\mathbf{R}\mathbf{R}' = \mathbf{I}_L$ is equivalent to supposing that $\mathbf{R}$ is full-rank. This is because for a full row-rank $\mathbf{R} = \mathbf{U}_R\mathbf{\Sigma}_R\mathbf{V}_R{}'$, one can pre-multiply both sides of (2.1) with $\mathbf{\Sigma}_R^{-1}\mathbf{U}_R{}'$ to obtain $\tilde{\mathbf{R}} := \mathbf{V}_R{}'$ with orthonormal rows.

2.4.2 Induced recovery results for principal components pursuit and compressed sensing

Before delving into the proof of the main result, it is instructive to examine how the sufficient conditions in Theorem 2.1 simplify for the subsumed PCP and CS problems. In PCP one has $\mathbf{R} = \mathbf{I}_L$, which implies $\Omega_R = \Omega$ and $\delta_k(\mathbf{R}) = \theta_{1,1}(\mathbf{R}) = 0$ so that one readily arrives at the following result.

Corollary 2.1 *Consider given* $\mathbf{Y} \in \mathbb{R}^{L\times T}$ *obeying* $\mathbf{Y} = \mathbf{X}_0 + \mathbf{A}_0 = \mathbf{U}\mathbf{\Sigma}\mathbf{V}' + \mathbf{A}_0$, *with* $r := rank(\mathbf{X}_0)$ *and* $s := \|\mathbf{A}_0\|_0$. *If the following conditions*

$\mathcal{I}$) $\mu(\Phi,\Omega)+2\sqrt{k(\gamma^2(\mathbf{U})+\gamma^2(\mathbf{V}))}<1$*; and*

$\mathcal{II}$) $\lambda_{max}:=\sqrt{s^{-1}}(\alpha^{-1}-\mu(\Phi,\Omega)\sqrt{r})>\lambda_{min}:=\beta\xi(\mathbf{U},\mathbf{V})$

hold, where

$$\alpha:=1+[(1-\mu(\Phi,\Omega))^{-2}-1]^{1/2},$$
$$\beta:=\left[1-4k(\gamma^2(\mathbf{U})+\gamma^2(\mathbf{V}))(1-\mu(\Phi,\Omega))^{-2}\right]^{-1}$$

then there exists $\lambda\in(\lambda_{\min},\lambda_{\max})$ *for which the convex program (P1) with* $\mathbf{R}=\mathbf{I}_L$ *exactly recovers* $\{\mathbf{X}_0,\mathbf{A}_0\}$*.*

In Section 2.6, random matrices $\{\mathbf{X}_0,\mathbf{A}_0,\mathbf{R}\}$ drawn from natural ensembles are shown to satisfy I) and II) with high probability. In this case, it is possible to arrive at simpler conditions (depending only on r, s, and the matrix dimensions) for exact recovery in the context of PCP; see Remark 2.6 which compares Corollary 2.1 with the existing results for PCP. Corollary 2.1, on the other hand, offers general conditions stemming from a purely deterministic approach. The best deterministic recovery results for PCP appear to be those reported in [34].

In the CS setting one has $\mathbf{X}_0=\mathbf{0}_{L\times T}$, which implies $\mu(\Phi,\Omega_R)=\xi_R(\mathbf{U},\mathbf{V})=\gamma_R(\mathbf{U})=\gamma(\mathbf{V})=0$. As a result, Theorem 2.1 simply boils down to a RIC-dependent sufficient condition for the exact recovery of $\mathbf{A}_0$ as stated next.

Corollary 2.2 *Consider given matrices* $\mathbf{Y}\in\mathbb{R}^{L\times T}$ *and* $\mathbf{R}\in\mathbb{R}^{L\times F}$ *obeying* $\mathbf{Y}=\mathbf{R}\mathbf{A}_0$*. Assume that the number of nonzero elements per column of* $\mathbf{A}_0$ *does not exceed* k*. If*

$$\delta_k(\mathbf{R})+k\theta_{1,1}(\mathbf{R})<1 \tag{2.10}$$

holds, then (P1) with $\mathbf{X}=\mathbf{0}_{L\times T}$ *exactly recovers* $\mathbf{A}_0$*.*

To place (2.10) in context, consider normalizing the rows of $\mathbf{R}$. For such a compression matrix it is known that $\delta_k(\mathbf{R})\leq(k-1)\theta_{1,1}(\mathbf{R})$, see e.g., [114]. Using this bound together with (2.10), one arrives at the stricter condition $k<\frac{1}{2}\left(1+\theta_{1,1}^{-1}(\mathbf{R})\right)$. This last condition is identical to the one reported in [44], which guarantees the success of ℓ_1-norm minimization in

recovering sparse solutions to under-determined systems of linear equations. The conditions have been improved in recent works; see e.g., [114] and references therein.

2.5 Proof of the Main Result

In what follows, conditions are first derived under which $\{\mathbf{X}_0, \mathbf{A}_0\}$ is the *unique* optimal solution of (P1). In essence, these conditions are expressed in terms of certain dual certificates. Then, Section 2.5.2 deals with the construction of a valid dual certificate.

2.5.1 Unique optimality conditions

Recall the *nonsmooth* optimization problem (P1), and its Lagrangian

$$\mathcal{L}(\mathbf{X}, \mathbf{A}, \mathbf{M}) = \|\mathbf{X}\|_* + \lambda\|\mathbf{A}\|_1 + \langle \mathbf{M}, \mathbf{Y} - \mathbf{X} - \mathbf{R}\mathbf{A}\rangle \tag{2.11}$$

where $\mathbf{M} \in \mathbb{R}^{L\times T}$ is the matrix of dual variables (multipliers) associated with the constraint in (P1). From the characterization of the subdifferential for nuclear- and ℓ_1-norm (see e.g., [22]), the subdifferential of the Lagrangian at $\{\mathbf{X}_0, \mathbf{A}_0\}$ is given by (recall that $\mathbf{X}_0 = \mathbf{U}\mathbf{\Sigma}\mathbf{V}'$)

$$\partial_{\mathbf{X}}\mathcal{L}(\mathbf{X}_0, \mathbf{A}_0, \mathbf{M}) = \left\{\mathbf{U}\mathbf{V}' + \mathbf{W} - \mathbf{M} : \|\mathbf{W}\| \le 1, \quad \mathcal{P}_\Phi(\mathbf{W}) = \mathbf{0}_{L\times T}\right\} \tag{2.12}$$

$$\partial_{\mathbf{A}}\mathcal{L}(\mathbf{X}_0, \mathbf{A}_0, \mathbf{M}) = \left\{\lambda\text{sign}(\mathbf{A}_0) + \lambda\mathbf{F} - \mathbf{R}'\mathbf{M} : \|\mathbf{F}\|_\infty \le 1, \quad \mathcal{P}_\Omega(\mathbf{F}) = \mathbf{0}_{F\times T}\right\}. \tag{2.13}$$

The optimality conditions for (P1) assert that $\{\mathbf{X}_0, \mathbf{A}_0\}$ is an optimal (not necessarily unique) solution if and only if

$$\mathbf{0}_{F\times T} \in \partial_{\mathbf{A}}\mathcal{L}(\mathbf{X}_0, \mathbf{A}_0, \mathbf{M}) \text{ and } \mathbf{0}_{L\times T} \in \partial_{\mathbf{X}}\mathcal{L}(\mathbf{X}_0, \mathbf{A}_0, \mathbf{M}).$$

This can be shown equivalent to finding the pair $\{\mathbf{W}, \mathbf{F}\}$ that satisfies: i) $\|\mathbf{W}\| \le 1$, $\mathcal{P}_\Phi(\mathbf{W}) = \mathbf{0}_{L\times T}$; ii) $\|\mathbf{F}\|_\infty \le 1$, $\mathcal{P}_\Omega(\mathbf{F}) = \mathbf{0}_{F\times T}$; and iii) $\lambda\text{sign}(\mathbf{A}_0) + \lambda\mathbf{F} = \mathbf{R}'(\mathbf{U}\mathbf{V}' + \mathbf{W})$. In general, i)-iii) may hold for multiple solution pairs. However, the next lemma asserts that a slight tightening of the optimality conditions i)-iii) leads to a *unique* optimal solution for (P1). See Appendix for a proof.

Lemma 2.2 *Assume that each column of* $\mathbf{A}_0$ *contains at most* k *nonzero elements, as well as* $\mu(\Omega_R, \Phi) < 1$ *and* $\delta_k(\mathbf{R}) < 1$*. If there exists a dual certificate* $\mathbf{\Gamma} \in \mathbb{R}^{L\times T}$ *satisfying*

C1) $\mathcal{P}_{\Phi}(\mathbf{\Gamma}) = \mathbf{U}\mathbf{V}'$

C2) $\mathcal{P}_{\Omega}(\mathbf{R}'\mathbf{\Gamma}) = \lambda \text{sgn}(\mathbf{A}_0)$

C3) $\|\mathcal{P}_{\Phi^{\perp}}(\mathbf{\Gamma})\| < 1$

C4) $\|\mathcal{P}_{\Omega^{\perp}}(\mathbf{R}'\mathbf{\Gamma})\|_{\infty} < \lambda$

then $\{\mathbf{X}_0, \mathbf{A}_0\}$ *is the unique optimal solution of (P1).*

The remainder of the proof deals with the construction of a dual certificate $\mathbf{\Gamma}$ that meets C1)-C4). To this end, tighter conditions [I) and II) in Theorem 2.1] for the existence of $\mathbf{\Gamma}$ are derived in terms of the incoherence parameters and the RICs. For the special case $\mathbf{R} = \mathbf{I}_L$, the conditions in Lemma 2.2 boil down to those in [33, Prop. 2] for PCP. However, the dual certificate construction techniques used in [33] do not carry over to the setting considered here, where a compression matrix $\mathbf{R}$ is present.

2.5.2 Dual certificate construction

Condition C1) in Lemma 2.2 implies that $\mathbf{\Gamma} = \mathbf{U}\mathbf{V}' + (\mathbf{I} - \mathbf{P}_U)\mathbf{X}(\mathbf{I} - \mathbf{P}_V)$, for arbitrary $\mathbf{X} \in \mathbb{R}^{L\times T}$ (cf. Remark 2.1). Upon defining $\mathbf{Z} := \mathbf{R}'(\mathbf{I} - \mathbf{P}_U)\mathbf{X}(\mathbf{I} - \mathbf{P}_U)$ and $\mathbf{B}_{\Omega} := \lambda \text{sign}(\mathbf{A}_0) - \mathcal{P}_{\Omega}(\mathbf{R}'\mathbf{U}\mathbf{V}')$, C1) and C2) are equivalent to $\mathcal{P}_{\Omega}(\mathbf{Z}) = \mathbf{B}_{\Omega}$.

To express $\mathcal{P}_{\Omega}(\mathbf{Z}) = \mathbf{B}_{\Omega}$ in terms of the unrestricted matrix $\mathbf{X}$, first vectorize $\mathbf{Z}$ to obtain $\text{vec}(\mathbf{Z}) = [(\mathbf{I} - \mathbf{P}_V) \otimes \mathbf{R}'(\mathbf{I} - \mathbf{P}_U)]\,\text{vec}(\mathbf{X})$. Define $\mathbf{A} := (\mathbf{I} - \mathbf{P}_V) \otimes \mathbf{R}'(\mathbf{I} - \mathbf{P}_U)$ and an $s \times LT$ matrix $\mathbf{A}_{\Omega}$ formed with those s rows of $\mathbf{A}$ associated with those elements in $\text{supp}(\mathbf{A}_0)$. Likewise, define $\mathbf{A}_{\Omega^{\perp}}$ which collects the remaining rows from $\mathbf{A}$ such that $\mathbf{A} = \mathbf{\Pi}[\mathbf{A}'_{\Omega}, \mathbf{A}'_{\Omega^{\perp}}]'$ for a suitable row permutation matrix $\mathbf{\Pi}$. Finally, let $\mathbf{b}_{\Omega}$ be the vector of length s containing those elements of $\mathbf{B}_{\Omega}$ with indices in $\text{supp}(\mathbf{A}_0)$. With these definitions, C1) and C2) can be expressed as $\mathbf{A}_{\Omega}\text{vec}(\mathbf{X}) = \mathbf{b}_{\Omega}$.

To upper-bound the left-hand side of C3) in terms of $\mathbf{X}$, use the assumption $\mathbf{R}\mathbf{R}' = \mathbf{I}_L$ to arrive at

$$\begin{aligned}\|\mathcal{P}_{\Phi^\perp}(\mathbf{\Gamma})\| &= \|\mathbf{R}'(\mathbf{I}-\mathbf{P}_U)\mathbf{X}(\mathbf{I}-\mathbf{P}_V)\| \\ &\leq \|\mathbf{R}'(\mathbf{I}-\mathbf{P}_U)\mathbf{X}(\mathbf{I}-\mathbf{P}_V)\|_F = \|\mathbf{A}\text{vec}(\mathbf{X})\|.\end{aligned}$$

Similarly, the left-hand side of C4) can be bounded as

$$\begin{aligned}\|\mathcal{P}_{\Omega^\perp}(\mathbf{R}'\mathbf{\Gamma})\|_\infty &= \|\mathcal{P}_{\Omega^\perp}(\mathbf{Z}) + \mathcal{P}_{\Omega^\perp}(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_\infty \\ &\leq \|\mathcal{P}_{\Omega^\perp}(\mathbf{Z})\|_\infty + \|\mathcal{P}_{\Omega^\perp}(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_\infty \\ &= \|\mathbf{A}_{\Omega^\perp}\text{vec}(\mathbf{X})\|_\infty + \|\mathcal{P}_{\Omega^\perp}(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_\infty.\end{aligned}$$

In a nutshell, if one can find $\mathbf{X} \in \mathbb{R}^{L\times T}$ such that

c1) $\mathbf{A}_\Omega \text{vec}(\mathbf{X}) = \mathbf{b}_\Omega$

c2) $\|\mathbf{A}\text{vec}(\mathbf{X})\| < 1$

c3) $\|\mathbf{A}_{\Omega^\perp}\text{vec}(\mathbf{X})\|_\infty + \|\mathcal{P}_{\Omega^\perp}(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_\infty < \lambda$

hold for some positive λ, then C1)-C4) would be satisfied as well.

The final steps of the proof entail: i) finding an appropriate candidate solution $\hat{\mathbf{X}}$ such that c1) holds; and ii) deriving conditions in terms of the incoherence parameters and RICs that guarantee $\hat{\mathbf{X}}$ meets the required bounds in c2) and c3) for a range of λ values. The following lemma is instrumental to accomplishing i), and its proof can be found in Appendix.

Lemma 2.3 *Assume that each column of* $\mathbf{A}_0$ *contains at most* k *nonzero elements, as well as* $\mu(\Omega_R, \Phi) < 1$ *and* $\delta_k(\mathbf{R}) < 1$. *Then matrix* $\mathbf{A}_\Omega$ *has full row rank, and its minimum singular value is bounded below as*

$$\sigma_{min}(\mathbf{A}'_\Omega) \geq c^{1/2}(1-\delta_k(\mathbf{R}))^{1/2}(1-\mu(\Phi,\Omega_R)).$$

According to Lemma 2.3, the least-norm (LN) solution $\hat{\mathbf{X}}_{\text{LN}} := \arg\min_{\mathbf{X}} \left\{\|\mathbf{X}\|_F^2 : \mathbf{A}_\Omega \text{vec}(\mathbf{X}) = \mathbf{b}_\Omega\right\}$ exists, and is given by

$$\text{vec}(\hat{\mathbf{X}}_{\text{LN}}) = \mathbf{A}'_\Omega \left(\mathbf{A}_\Omega \mathbf{A}'_\Omega\right)^{-1} \mathbf{b}_\Omega. \tag{2.14}$$

Remark 2.3 (Candidate dual certificate) From the arguments at the beginning of this section, the candidate dual certificate is $\hat{\mathbf{\Gamma}} := \mathbf{U}\mathbf{V}' + (\mathbf{I} - \mathbf{P}_U)\hat{\mathbf{X}}_{\mathrm{LN}}(\mathbf{I} - \mathbf{P}_V)$.

The LN solution is an attractive choice, since it facilitates satisfying c2) and c3) which require norms of $\mathrm{vec}(\mathbf{X})$ to be small. Substituting the LN solution (2.14) into the left hand side of c2) yields (define $\mathbf{Q} := \mathbf{A}_{\Omega^\perp}\mathbf{A}_\Omega'\left(\mathbf{A}_\Omega\mathbf{A}_\Omega'\right)^{-1}$ for notational brevity)

$$\begin{aligned}\|\mathbf{A}\mathrm{vec}(\hat{\mathbf{X}}_{\mathrm{LN}})\| &= \left\| \begin{pmatrix} \mathbf{A}_\Omega \\ \mathbf{A}_{\Omega^\perp} \end{pmatrix} \mathbf{A}_\Omega' \left(\mathbf{A}_\Omega\mathbf{A}_\Omega'\right)^{-1} \mathbf{b}_\Omega \right\| \\ &= \left\| \begin{pmatrix} \mathbf{I} \\ \mathbf{Q} \end{pmatrix} \mathbf{b}_\Omega \right\| \leq (1 + \|\mathbf{Q}\|)\,\|\mathbf{b}_\Omega\|. \end{aligned} \tag{2.15}$$

Moreover, substituting (2.14) in the left hand side of c3) results in

$$\|\mathbf{Q}\mathbf{b}_\Omega\|_\infty + \|\mathcal{P}_{\Omega^\perp}(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_\infty \leq \|\mathbf{Q}\|_{\infty,\infty}\|\mathbf{b}_\Omega\|_\infty + \|\mathcal{P}_{\Omega^\perp}(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_\infty. \tag{2.16}$$

Next, upper-bounds are obtained for $\|\mathbf{Q}\|$ and $\|\mathbf{Q}\|_{\infty,\infty}$; see Appendix for a proof.

Lemma 2.4 *Assume that each column and row of* $\mathbf{A}_0$ *contains at most* k *nonzero elements. If* $\mu(\Omega_R, \Phi) < 1$ *and* $\delta_k(\mathbf{R}) < 1$ *hold, then*

$$\|\mathbf{Q}\| \leq \nu_1 := \left[\frac{1}{c(1 - \delta_k(\mathbf{R}))(1 - \mu(\Omega_R, \Phi))^2} - 1\right]^{1/2}.$$

If the tighter condition I) holds instead, then

$$\|\mathbf{Q}\|_{\infty,\infty} \leq \nu_2 := \frac{\omega}{(1 - \mu(\Omega_R, \Phi))^2(1 - \delta_k(\mathbf{R})) - \omega}.$$

Going back to (2.15)-(2.16), note that $\|\mathbf{B}_\Omega\|_\infty = \|\mathbf{b}_\Omega\|_\infty$ and $\|\mathbf{B}_\Omega\|_F = \|\mathbf{b}_\Omega\|$, which can be respectively upper-bounded as

$$\begin{aligned}\|\mathbf{B}_\Omega\|_\infty &= \|\lambda\mathrm{sign}(\mathbf{A}_0) - \mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_\infty \\ &\leq \lambda + \|\mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_\infty \qquad (2.17) \\ \|\mathbf{B}_\Omega\|_F &= \|\lambda\mathrm{sign}(\mathbf{A}_0) - \mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_F \\ &\leq \lambda\sqrt{s} + \|\mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_F. \qquad (2.18)\end{aligned}$$

Finally, $\|\mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_F$ itself can be bounded above as

$$\begin{aligned}
\|\mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_F^2 &= |\langle \mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}'), \mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}')\rangle| \\
&\overset{(a)}{=} |\langle \mathbf{R}'\mathbf{U}\mathbf{V}', \mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}')\rangle| \\
&= |\langle \mathbf{U}\mathbf{V}', \mathbf{R}\mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}')\rangle| \\
&\overset{(b)}{=} |\langle \mathcal{P}_\Phi(\mathbf{U}\mathbf{V}'), \mathcal{P}_\Phi(\mathbf{R}\mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}'))\rangle| \\
&\overset{(c)}{\leq} \|\mathcal{P}_\Phi(\mathbf{U}\mathbf{V}')\|_F \|\mathcal{P}_\Phi(\mathbf{R}\mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}'))\|_F \\
&\overset{(d)}{\leq} \|\mathbf{U}\mathbf{V}'\|_F \mu(\Phi, \Omega_r) \|\mathbf{R}\mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_F \\
&\overset{(e)}{\leq} \sqrt{r}\mu(\Phi, \Omega_r) c^{1/2} (1 + \delta_k(\mathbf{R}))^{1/2} \\
&\quad \times \|\mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_F
\end{aligned} \tag{2.19}$$

where (a) is due to (2.5), (b) follows because $\mathbf{U}\mathbf{V}' \in \Phi$ (thus $\mathcal{P}_\Phi(\mathbf{U}\mathbf{V}') = \mathbf{U}\mathbf{V}'$) and from the property in (2.5). Moreover, (c) is a direct result of the Cauchy-Schwarz inequality, while (d) and (e) come from (2.6) and (2.7), respectively, and the assumption that number of nonzero elements per column of $\mathbf{A}_0$ does not exceed k. All in all, $\|\mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_F \leq \sqrt{r}\mu(\Phi, \Omega_R) c^{1/2} (1 + \delta_k(\mathbf{R}))^{1/2}$ and (2.18) becomes

$$\|\mathbf{B}_\Omega\|_F \leq \lambda\sqrt{s} + \sqrt{r}\mu(\Phi, \Omega_r) c^{1/2} (1 + \delta_k(\mathbf{R}))^{1/2}. \tag{2.20}$$

Upon substituting (2.17), (2.20) and the bounds in Lemma 2.4 into (2.15) and (2.16), one finds that c2) and c3) hold if there exists $\lambda > 0$ such that

$$(1 + \nu_1)\left[\lambda\sqrt{s} + \sqrt{r}\mu(\Omega_R, \Phi) c^{1/2} (1 + \delta_k(\mathbf{R}))^{1/2}\right] < 1 \tag{2.21a}$$

$$\nu_2\left(\lambda + \|\mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_\infty\right) + \|\mathcal{P}_{\Omega^\perp}(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_\infty < \lambda \tag{2.21b}$$

hold. Recognizing that $\xi_R(\mathbf{U}, \mathbf{V}) = \max\{\|\mathcal{P}_\Omega(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_\infty, \|\mathcal{P}_{\Omega^\perp}(\mathbf{R}'\mathbf{U}\mathbf{V}')\|_\infty\}$, the left-hand side of (2.21b) can be further bounded. After straightforward manipulations, one deduces that conditions (2.21a) and (2.21b) are satisfied for $\lambda \in (\lambda_{\min}, \lambda_{\max})$ if $\nu_2 < 1$, where

$$\begin{aligned}
\lambda_{\min} &:= \left(\frac{1 + \nu_2}{1 - \nu_2}\right) \xi_R(\mathbf{U}, \mathbf{V}) \\
\lambda_{\max} &:= \frac{1}{\sqrt{s}}\left[(1 + \nu_1)^{-1} - \sqrt{r}\mu(\Omega_R, \Phi) c^{1/2} (1 + \delta_k(\mathbf{R}))^{1/2}\right].
\end{aligned}$$

Clearly, it is still necessary to ensure $\lambda_{\max} > \lambda_{\min}$ so that the LN solution (2.14) meets the requirements c1)-c3) [equivalently, $\hat{\mathbf{\Gamma}}$ in Remark 2.3 satisfies C1)-C4) from Lemma 2.2]. Condition $\lambda_{\max} > \lambda_{\min}$ is equivalent to II) in Theorem 2.1, and the proof is now complete.

Remark 2.4 (Satisfiability) From a high-level vantage point, Theorem 2.1 asserts that (P1) recovers $\{\mathbf{X}_0, \mathbf{A}_0\}$ when the components $\mathbf{X}_0$ and $\mathbf{R}\mathbf{A}_0$ are sufficiently incoherent, and the compression matrix $\mathbf{R}$ has good restricted isometry properties. It should be noted though, that given a triplet $\{\mathbf{X}_0, \mathbf{A}_0, \mathbf{R}\}$ in general one cannot directly check whether the sufficient conditions I) and II) hold, since e.g., $\delta_k(\mathbf{R})$ is NP-hard to compute [29]. This motivates finding a class of (possibly random) matrices $\{\mathbf{X}_0, \mathbf{A}_0, \mathbf{R}\}$ satisfying I) and II), the subject dealt with next.

2.6 Matrices Satisfying the Conditions for Exact Recovery

This section investigates triplets $\{\mathbf{X}_0, \mathbf{A}_0, \mathbf{R}\}$ satisfying the conditions of Theorem 2.1, henceforth termed admissible matrices. Specifically, it will be shown that low-rank, sparse, and compression matrices drawn from certain random ensembles satisfy the sufficient conditions of Theorem 2.1 with high probability.

2.6.1 Uniform sparsity model

Matrix $\mathbf{A}_0$ is said to be generated according to the *uniform sparsity* model, when drawn uniformly at random from the collection of all matrices with support size s. There is no restriction on the amplitude of the nonzero entries. An attractive property of this model is that it guarantees (with high probability) that no single row or column will monopolize most nonzero entries of $\mathbf{A}_0$, for sufficiently large $\mathbf{A}_0$ and appropriate scaling of the sparsity level. This property is formalized in the following lemma (for simplicity in exposition it is henceforth assumed that that $\mathbf{A}_0$ is a square matrix, i.e., $F = T$).

Lemma 2.5 [33] *If $\mathbf{A}_0 \in \mathbb{R}^{F\times F}$ is generated according to the uniform sparsity model with $\|\mathbf{A}_0\|_0 = s$, then the maximum number k of nonzero elements per column or row of $\mathbf{A}_0$ is*

bounded as

$$k \leq \frac{s}{F} \log(F)$$

with probability higher than $1 - \mathcal{O}(F^{-\zeta})$*, for* $s = \mathcal{O}(\zeta F)$*.*

In practice, it is simpler to work with the Bernoulli model that specifies $\text{supp}(\mathbf{A}_0) = \{(i,j) : b_{i,j} = 1\}$, where $\{b_{i,j}\}$ are independent and identically distributed (i.i.d.) Bernoulli random variables taking value one with probability $\pi := s/F^2$, and zero with probability $1 - \pi$. There are three important observations regarding the Bernoulli model. First, $|\text{supp}(\mathbf{A}_0)|$ is a random variable, whose expected value is s and matches the uniform sparsity model. Second, arguing as in [25, Lemma 2.2] one can claim that if (P1) exactly recovers $\{\mathbf{X}_0, \mathbf{A}_0\}$ from data $\mathbf{Y} = \mathbf{X}_0 + \mathbf{R}\mathbf{A}_0$, it will also exactly recover $\{\mathbf{X}_0, \check{\mathbf{A}}_0\}$ from $\check{\mathbf{Y}} = \mathbf{X}_0 + \mathbf{R}\check{\mathbf{A}}_0$ when $\text{supp}(\check{\mathbf{A}}_0) \subseteq \text{supp}(\mathbf{A}_0)$ and the nonzero entries coincide. Third, following the logic of [28, Section II.C] one can prove that the failure rate[1] for the uniform sparsity model is bounded by twice the failure rate corresponding to the Bernoulli model. As a result, any recovery guarantee established for the Bernoulli model holds for the uniform sparsity model as well.

In addition to the bound for k in Lemma 2.5, the Bernoulli model can be used to bound $\mu(\Phi, \Omega_R)$ in terms of the incoherence parameters $\{\gamma_R(\mathbf{U}), \gamma(\mathbf{V})\}$ and the RIC $\delta_k(\mathbf{R})$. For a proof, see Appendix.

Lemma 2.6 *Let* $\Lambda := \sqrt{c(1 + \delta_1(\mathbf{R}))} \left[\gamma_R^2(\mathbf{U}) + \gamma^2(\mathbf{V})\right]^{1/2}$ *and* $n := \max\{L, F\}$*. Suppose* $\mathbf{A}_0 \in \mathbb{R}^{F \times F}$ *is generated according to the Bernoulli model with* $\Pr(b_{i,j} = 1) = \pi$*, and* $\mathbf{R}\mathbf{R}' = \mathbf{I}_L$*. Then, there exist positive constants* C *and* τ *such that*

$$\mu(\Phi, \Omega_R) \leq \sqrt{c^{-1}(1 - \delta_k(\mathbf{R}))^{-1}\pi} \left[C\Lambda\sqrt{\log(LF)/\pi} + \tau\Lambda \log(n) + 1\right]^{1/2} \tag{2.22}$$

holds with probability at least $1 - n^{-C\pi\Lambda\tau}$ *if* $\delta_k(\mathbf{R})$ *and the right-hand side of (2.22) do not exceed one.*[2]

---

[1] The failure rate is defined as $\Pr(\hat{\mathbf{A}} \neq \mathbf{A}_0)$, where $\hat{\mathbf{A}}$ is the solution of (P1).

[2] Even though one has $n = F$ and $\pi = s/F^2$ in the problem studied here, Lemma 2.6 is stated using n and π to retain generality.

Consider (2.22) when Λ is small enough so that the quantity inside the square brackets is close to one. One obtains $\mu(\Phi, \Omega_R) \leq \sqrt{c^{-1}(1-\delta_k(\mathbf{R}))^{-1}\pi}$, which reduces to the bound $\mu(\Phi, \Omega) \leq \sqrt{\pi}$ derived in [25, Section 2.5] for the special case $\mathbf{R} = \mathbf{I}_L$. Hence, the price paid in terms of coherence increase due to $\mathbf{R}$ is roughly $\sqrt{c^{-1}(1-\delta_k(\mathbf{R}))^{-1}} > 1$. As expected, (2.22) also shows that for $\mathbf{R}$ with small RICs the incoherence between subspaces Φ and Ω_R becomes smaller, and identifiability is more likely.

The result in Lemma 2.6 allows one to 'eliminate' $\mu(\Phi, \Omega_R)$ from the sufficient conditions in Theorem 2.1, which can thus be expressed only in terms of $\{\gamma_R(\mathbf{U}), \gamma(\mathbf{V}), \xi_R(\mathbf{U}, \mathbf{V})\}$ and the RICs of $\mathbf{R}$. In the following sections, random low-rank and compression matrices giving rise to small incoherence parameters and RICs are described.

2.6.2 Random orthogonal model

Among other implications, matrices $\mathbf{X}_0$ and $\mathbf{R}$ with small $\gamma_R(\mathbf{U})$ and $\xi_R(\mathbf{U}, \mathbf{V})$ are such that the columns of $\mathbf{R}$ (approximately) fall outside the column space of $\mathbf{X}_0$. From a design perspective, this suggests that the choice of an admissible $\mathbf{X}_0$ (or in general an ensemble of low-rank matrices) should take into account the structure of $\mathbf{R}$, and vice versa. However, in the interest of simplicity one could seek conditions dealing with $\mathbf{X}_0$ and $\mathbf{R}$ *separately*, that still ensure $\gamma_R(\mathbf{U})$ and $\xi_R(\mathbf{U}, \mathbf{V})$ are small. This way one can benefit from the existing theory on incoherent low-rank matrices developed in the context of matrix completion [27], and matrices with small RICs useful for CS [28, 114]. Admittedly, the price paid is in terms of stricter conditions that will reduce the set of admissible matrices.

In this direction, the next lemma bounds $\gamma_R(\mathbf{U})$ and $\xi_R(\mathbf{U}, \mathbf{V})$ in terms of $\gamma(\mathbf{U}) := \max_i \|\mathbf{P}_U \mathbf{e}_i\|$, $\gamma(\mathbf{V})$ and $\delta_k(\mathbf{R})$.

Lemma 2.7 *If* $\eta(\mathbf{R}) := \max_i \|\mathbf{R}\mathbf{e}_i\|_1 / \|\mathbf{R}\mathbf{e}_i\|$, *it then holds that*

$$\gamma_R(\mathbf{U}) \leq \eta(\mathbf{R})\gamma(\mathbf{U}) \tag{2.23}$$

$$\xi_R(\mathbf{U}, \mathbf{V}) \leq \sqrt{c(1+\delta_1(\mathbf{R}))}\eta(\mathbf{R})\gamma(\mathbf{U})\gamma(\mathbf{V}). \tag{2.24}$$

Proof: Starting from the definition

$$\gamma_R(\mathbf{U}) = \max_i \frac{\|\mathbf{P}_U \mathbf{R}\mathbf{e}_i\|}{\|\mathbf{R}\mathbf{e}_i\|} = \max_i \frac{\|\mathbf{P}_U \sum_\ell \mathbf{e}_\ell \mathbf{e}_\ell' \mathbf{R}\mathbf{e}_i\|}{\|\mathbf{R}\mathbf{e}_i\|}$$
$$\stackrel{(a)}{\leq} \max_i \frac{\sum_\ell \|\mathbf{P}_U \mathbf{e}_\ell\| |\mathbf{e}_\ell' \mathbf{R}\mathbf{e}_i|}{\|\mathbf{R}\mathbf{e}_i\|} \stackrel{(b)}{\leq} \gamma(\mathbf{U}) \max_i \frac{\|\mathbf{R}\mathbf{e}_i\|_1}{\|\mathbf{R}\mathbf{e}_i\|} \tag{2.25}$$

where (a) follows from the Cauchy-Schwarz inequality, and (b) from the definition of $\gamma(\mathbf{U})$.

Likewise, applying the definition of $\xi_R(\mathbf{U}, \mathbf{V})$ one obtains

$$\begin{aligned} \xi_R(\mathbf{U}, \mathbf{V}) &= \max_{i,j} |\mathbf{e}_i' \mathbf{R}' \mathbf{U} \mathbf{V}' \mathbf{e}_j| \\ &\stackrel{(c)}{\leq} \max_i \|\mathbf{U}' \mathbf{R}\mathbf{e}_i'\| \max_i \|\mathbf{V}' \mathbf{e}_j\| \\ &\leq \sqrt{c(1 + \delta_1(\mathbf{R}))} \gamma_R(\mathbf{U}) \gamma(\mathbf{V}) \\ &\stackrel{(d)}{\leq} \sqrt{c(1 + \delta_1(\mathbf{R}))} \eta(\mathbf{R}) \gamma(\mathbf{U}) \gamma(\mathbf{V}) \end{aligned} \tag{2.26}$$

where (c) follows from the Cauchy-Schwarz inequality, and (d) is due to (2.25).

The bounds (2.23) and (2.24) are proportional to $\gamma(\mathbf{U})$ and $\gamma(\mathbf{V})$. This prompts one to consider incoherent rank-r matrices $\mathbf{X}_0 = \mathbf{U}\mathbf{\Sigma}\mathbf{V}'$ generated from the *random orthogonal* model, which is specified as follows. The singular vectors forming the columns of $\mathbf{U}$ and $\mathbf{V}$ are drawn uniformly at random from the collection of rank-r partial isometries in $\mathbb{R}^{L \times r}$ and $\mathbb{R}^{F \times r}$, respectively. There is no need for $\mathbf{U}$ and $\mathbf{V}$ to be statistically independent, and no restriction in placed on the singular values in the diagonal of $\mathbf{\Sigma}$. The adequacy of the random orthogonal model in generating incoherent low-rank matrices is justified by the following lemma (recall $T = F \geq L$).

Lemma 2.8 [33] *If $\mathbf{X}_0 = \mathbf{U}\mathbf{\Sigma}\mathbf{V}' \in \mathbb{R}^{L \times F}$ is generated according to the random orthogonal model with rank$(\mathbf{X}_0) = r$, then*

$$\max\{\gamma(\mathbf{U}), \gamma(\mathbf{V})\} \leq \sqrt{\frac{\max\{r, \log(F)\}}{F}}$$

with probability exceeding $1 - \mathcal{O}(F^{-3} \log(F))$.

2.6.3 Random compressive matrices

With reference to Lemma 2.7 [cf. (2.23) and (2.24)], it is clear that an incoherent $\mathbf{X}_0$ alone may not suffice to yield small $\gamma_R(\mathbf{U})$ and $\xi_R(\mathbf{U},\mathbf{V})$. In addition, $\eta(\mathbf{R}) \in [1,\sqrt{L}]$ should be as close as possible to one. This can be achieved e.g., when $\mathbf{R}$ is sparse across each column. Note that the lower bound of unity is attained when $\mathbf{R}$ has at most a single nonzero element per column, as it is the case when $\mathbf{R} = \mathbf{I}_L$.

The aforementioned observations motivate considering block-diagonal compression matrices $\mathbf{R} \in \mathbb{R}^{L\times F}$, consisting of blocks $\{\mathbf{R}_i \in \mathbb{R}^{\ell\times f}\}$ where $\ell \leq f$. The number of blocks is $n_b := F/f$ assuming that f divides F. The i-th block is generated according to the *bounded orthonormal* model as follows; see e.g., [114]. For some positive constant K, (deterministically) choose a unitary matrix $\mathbf{\Psi} \in \mathbb{R}^{f\times f}$ with bounded entries

$$\max_{(t,k)\in\mathcal{F}\times\mathcal{F}} |\mathbf{\Psi}_{t,k}| \leq K \tag{2.27}$$

where $\mathcal{F} := \{1,...,f\}$. For each $i = 1,\ldots,n_b$ form $\mathbf{R}_i := \mathbf{\Theta}_{T^{(i)}}\mathbf{\Psi}$, where $\mathbf{\Theta}_{T^{(i)}} := [\mathbf{e}_{t_1^{(i)}},\ldots,\mathbf{e}_{t_\ell^{(i)}}]' \in \mathbb{R}^{\ell\times f}$ is a random row subsampling matrix that selects the rows of $\mathbf{\Psi}$ indexed by $\mathcal{T}^{(i)} := \{t_1^{(i)},...,t_\ell^{(i)}\} \subset \mathcal{F}$. In words, $\mathbf{\Theta}_{T^{(i)}}$ is formed by those ℓ rows of $\mathbf{I}_f$ indexed by $\mathcal{T}^{(i)}$. The row indices in $\mathcal{T}^{(i)}$ are selected independently at random, with uniform probability $1/f$ from $\mathcal{F}$. By construction, $\mathbf{R}_i\mathbf{R}_i' = \mathbf{I}_\ell, i = 1,\ldots,n_b$, which ensures $\mathbf{R}\mathbf{R}' = \mathbf{I}_L$ as required by Theorem 2.1. Most importantly, the next lemma states that such a construction of $\mathbf{R}_i$ leads to small RICs with high probability; see e.g., [114] for the proof.

Lemma 2.9 [114] *Let $\mathbf{R}_i \in \mathbb{R}^{\ell\times f}$ be generated according to the bounded orthonormal model. If for some $k_i \in [1,f]$, $\epsilon \in (0,1)$ and $\mu \in (0,1/2]$ the following condition*

$$\frac{\ell}{\log(10\ell)} \geq DK^2\mu^{-2}s\log^2(100k_i)\log(4f)\log(7\epsilon^{-1}) \tag{2.28}$$

holds where the constant $D \leq 243,150$, then $\delta_{k_i}(\mathbf{R}_i) \leq \mu$ with probability greater than $1-\epsilon$.

Lemma 2.9 asserts that for large enough ℓ, the RIC $\delta_{k_i}(\mathbf{R}_i) = \mathcal{O}(\log(100k_i)\log(10\ell)\log(4f)^{1/2}\sqrt{k_i/\ell})$ with overwhelming probability.

Let k_i denote the maximum number of nonzero elements per 'trimmed' column of $\mathbf{A}_0$, the trimming being defined by the block of rows of $\mathbf{A}_0$ that are multiplied by $\mathbf{R}_i$ when

carrying out the product $\mathbf{R}\mathbf{A}_0$. With these definitions, the RIC of $\mathbf{R}$ is bounded as $\delta_k(\mathbf{R}) \leq \max_i\{\delta_{k_i}(\mathbf{R}_i)\}$. For $\delta_k(\mathbf{R})$ to be small as required by Theorem 2.1, the k_i should be much smaller than ℓ. Since $\mathbf{A}_0$ is generated according to the uniform sparsity model outlined in Section 2.6.1, its nonzero elements are uniformly spread across rows and columns as per Lemma 2.5. Formally, it holds that $k_i \leq \kappa := (s/Fn_b)\log(Fn_b)$ with probability $1-\mathcal{O}([Fn_b]^{-\zeta})$, where $s = \|\mathbf{A}_0\|_0 = \zeta Fn_b$; see e.g., [20]. Accordingly, from Lemma 2.9 one can infer that $\delta_k(\mathbf{R}) = \mathcal{O}(\log(100\kappa)\log(10\ell)\log(4f)^{1/2}\sqrt{\kappa/\ell})$ with high probability. Note that the bound for $\delta_k(\mathbf{R})$ depends on k through the variable s in κ, and the relationship between s and k in Lemma 2.5. Regarding the RIC $\theta_{1,1}(\mathbf{R})$, it is bounded as $\theta_{1,1}(\mathbf{R}) \leq \delta_2(\mathbf{R})$ [29]. The normalization constant c in (2.7) and (2.8) also equals $L/F \ll 1$. Recalling $\eta(\mathbf{R})$ (cf. Lemma 2.7) which was subject of the initial discussion in this section, it turns out that for such a construction of $\mathbf{R}$ one obtains $\eta(\mathbf{R}) \leq \sqrt{\ell} \ll \sqrt{L}$.

Remark 2.5 (Row and column permutations) The class of admissible compression matrices can be extended to matrices which are block diagonal up to row and column permutations. Let $\mathbf{\Pi}_r$ ($\mathbf{\Pi}_c$) denote, respectively, the row (column) permutation matrices that render $\mathbf{R}$ block diagonal. Instead of (2.1) consider $\mathbf{\Pi}_r\mathbf{Y} = \mathbf{\Pi}_r\mathbf{X}_0 + \mathbf{\Pi}_r\mathbf{R}\mathbf{\Pi}_c\mathbf{\Pi}_c'\mathbf{A}_0$ and note that $\mathbf{\Pi}_r\mathbf{X}_0$ has the same coherence parameters as $\mathbf{X}_0$, while $\mathbf{\Pi}_r\mathbf{R}\mathbf{\Pi}_c$ has the same RICs as $\mathbf{R}$, and $\mathbf{\Pi}_c'\mathbf{A}_0$ is still uniformly sparse. Thus, one can feed the transformed data to (P1) and since $\mathbf{\Pi}_r$ and $\mathbf{\Pi}_c$ are invertible, $\{\mathbf{X}_0, \mathbf{A}_0\}$ can be readily obtained from the recovered $\{\mathbf{\Pi}_r\mathbf{X}_0, \mathbf{\Pi}_c'\mathbf{A}_0\}$.

2.6.4 Closing the loop

According to Lemmata 2.6 and 2.7, the incoherence parameters $\mu(\Phi, \Omega_R)$, $\gamma_R(\mathbf{U})$ and $\xi_R(\mathbf{U}, \mathbf{V})$ which play a critcal role toward exact decomposability in Theorem 2.1, can be upper-bounded in terms of $\gamma(\mathbf{U})$ and $\gamma(\mathbf{V})$. For random matrices $\{\mathbf{X}_0, \mathbf{A}_0, \mathbf{R}\}$ drawn from specific ensembles, Lemmata 2.5, 2.8 and 2.9 assert that the incoherence parameters $\gamma(\mathbf{U})$ and $\gamma(\mathbf{V})$ as well as the RICs $\delta_k(\mathbf{R})$ and $\theta_{1,1}(\mathbf{R})$, are bounded above in terms of $r = \text{rank}(\mathbf{X}_0)$, the degree of sparsity $s = \|\mathbf{A}_0\|_0$, and the underlying matrix dimensions L, F, ℓ, f. Alternative sufficient conditions for exact recovery, expressible only in terms of

the aforementioned basic parameters, can be obtained by combining the bounds of this section along with I) and II) in Theorem 2.1. Hence, in order to guarantee that (P1) recovers $\{\mathbf{X}_0, \mathbf{A}_0\}$ with high probability and for given matrix dimensions, it suffices to check feasibility of a set of inequalities in r and s.

To this end, focus on the asymptotic case where L and F are large enough, while $F = T$ for simplicity in exposition. Recall the conditions of Theorem 2.1 and suppose $\delta_k(\mathbf{R}) = o(1)$ and $\mu(\Phi, \Omega_R) = o(1)$. This results in $\alpha \approx \sqrt{F/L}$ and $\chi \approx \omega$ when $L \ll F$. Satisfaction of I) and II) then requires $\mathcal{O}(1)$ summands in both sides of II) when multiplied with $\alpha\sqrt{s}$, which gives rise to $\xi_R(\mathbf{U}, \mathbf{V}) = \mathcal{O}(\sqrt{L/Fs})$, $\mu(\Phi, \Omega_R) = \mathcal{O}(1/\sqrt{r})$, and $\omega = \mathcal{O}(1) < 1$. The latter which is indeed the bottleneck constraint can be satisfied if $\theta_{1,1}(\mathbf{R}) = \mathcal{O}(1/k)$, $\theta_{1,1}(\mathbf{R})\gamma^2(\mathbf{V}) = \mathcal{O}(1/s)$, $\gamma_R^2(\mathbf{U}) = \mathcal{O}(1/k)$, $\gamma^2(\mathbf{V}) = \mathcal{O}(1/k)$, and $\gamma_R^2(\mathbf{U})\gamma_R^2(\mathbf{V}) = \mathcal{O}(1/s)$. Utilizing the bounds in Lemmata 2.6–2.9 establishes the next corollary.

Corollary 2.3 *Consider given matrices $\mathbf{Y} \in \mathbb{R}^{L\times F}$ and $\mathbf{R} \in \mathbb{R}^{L\times F}$ obeying $\mathbf{Y} = \mathbf{X}_0 + \mathbf{R}\mathbf{A}_0$, where $r := rank(\mathbf{X}_0)$ and $s := \|\mathbf{A}_0\|_0$. Suppose that: (i) $\mathbf{X}_0$ is generated according to the random orthogonal model; (ii) $\mathbf{A}_0$ is generated according to the uniform sparsity model; and (ii) $\mathbf{R} = bdiag(\mathbf{R}_1, \ldots, \mathbf{R}_{n_b})$ with blocks $\mathbf{R}_i \in \mathbb{R}^{\ell\times f}$ generated according to the bounded orthogonal model. Define $\tilde{r} := \max\{r, \log(F)\}$. If r and s satisfy*

i) $\tilde{r} \precsim \frac{F}{\ell}$

ii) $s \precsim \min\left\{\frac{F^2}{\ell\log(F)\tilde{r}}, \frac{F^2}{\tilde{r}^2}, \frac{F\sqrt{\ell}}{\log(10\ell)\log^{1/2}(4f)\tilde{r}}\right\}$

iii) $\sqrt{s}\log\left(100\frac{sf}{F^2}\log\left(\frac{F^2}{f}\right)\right) \prec \left[\frac{F^2\ell}{f\log(F^2/f)\log^2(f)}\right]^{1/2}$

there is a positive λ for which (P1) recovers $\{\mathbf{X}_0, \mathbf{A}_0\}$ with high probability.

Remark 2.6 (Results for principal component pursuit) *For an ensemble of random matrices $\{\mathbf{X}_0, \mathbf{A}_0\}$, the induced recovery results for PCP in Corollary 2.1 are simplified and compared here with those obtained in [25], [33], and [34]. To be aligned with [25] and [34], the ρ-incoherent low-rank matrix model in [25] is adopted for $\mathbf{X}_0$, where $\gamma(\mathbf{U}) = \gamma(\mathbf{V}) = \sqrt{\rho r/L}$, and $\xi(\mathbf{U}, \mathbf{V}) = \sqrt{\rho r}/L$ for some constant $\rho > 0$. Matrix $\mathbf{A}_0$ is also drawn from*

the uniform sparsity model outlined in Section 2.6.1. From Corollary 2.1 and the results in Lemmata 2.5-2.6, it follows that $s \precsim \frac{L^2}{\log(L)} \min\{\frac{1}{r}, \frac{L}{r^3}\}$ *suffices for exact recovery with high probability. In particular, if* $r \leq \sqrt{L}$*, the pair* (r, s) *should only satisfy* $sr \precsim \frac{L^2}{\log(L)}$*. In contrast, results in [33] only offer recovery guarantees for rank and sparsity levels up to* $s\sqrt{r} \precsim \frac{L^{3/2}}{\log(L)}$*, which are weaker than those derived from Corollary 2.1 as* $r \leq L$*. The results in [33] have been improved in [34, Theorem 3], which allows rank and sparsity levels up to* $sr \precsim \frac{L^2}{\log(L)}$ *as obtained from Corollary 2.1. Note that Corollary 2.1, [33], and [34] offer* deterministic *reconstruction guarantees, where [34] yields the best results. Still in the aforementioned random setting, the condition induced from Corollary 2.1 is comparable with [34] thanks to the existing tight probabilistic bounds for* $\mu(\Phi, \Omega)$*. The results in [25] however, build on the uniform sparsity model for* $\mathbf{A}_0$*, and provide superior probabilistic guarantees up to* $s \precsim L^2$ *and* $r \precsim L$*.*

It is worth noting that in the presence of the compression matrix $\mathbf{R}$ more stringent conditions are imposed on the rank and sparsity level, as stated in Corollary 2.3. This is mainly because of the dominant summand $\theta_{1,1}(\mathbf{R})[\sqrt{2}k + s\gamma^2(V)]$ in ω (cf. Theorem 2.1), which limits the extent to which r and s can be increased. If the correlation between any two columns of $\mathbf{R}$ is small, then higher rank and less sparse matrices can be exactly recovered.

2.7 Algorithms

This section deals with iterative algorithms to solve the non-smooth convex optimization problem (P1).

2.7.1 Accelerated proximal gradient (APG) algorithm

The class of accelerated proximal gradient algorithms were originally studied in [110, 111], and they have been popularized for ℓ_1-norm regularized regression; mostly due to the success of the fast iterative shrinkage-thresholding algorithm (FISTA) [14]. Recently, APG algorithms have been applied to matrix-valued problems such as those arising with nuclear-norm regularized estimators for matrix completion [139], and for (stable) PCP [80, 163].

APG algorithms offer several attractive features, most notably a convergence rate guarantee of $\mathcal{O}(1/\sqrt{\epsilon})$ 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 developed here builds on the APG iterations in [80], proposed to solve the stable PCP problem. One can relax the equality constraint in (P1) and instead solve

$$\text{(P2)} \quad \min_{\mathbf{S}} \quad \left\{ \nu\|\mathbf{X}\|_* + \nu\lambda\|\mathbf{A}\|_1 + \frac{1}{2}\|\mathbf{Y}-\mathbf{X}-\mathbf{R}\mathbf{A}\|_F^2 \right\}$$

with $\mathbf{S} := [\mathbf{X}', \mathbf{A}']'$, where the least-square term penalizes violations of the equality constraint, and $\nu > 0$ is a penalty coefficient. When ν approaches zero, (P2) achieves the optimal solution of (P1) [17]. The gradient of $f(\mathbf{S}) := \frac{1}{2}\|\mathbf{Y}-\mathbf{X}-\mathbf{R}\mathbf{A}\|_F^2$ is Lipschitz continuous with a (minimum) Lipschitz constant $L_f = \lambda_{\max}([\mathbf{I}_L \;\; \mathbf{R}]'[\mathbf{I}_L \;\; \mathbf{R}])$, i.e., $\|\nabla f(\mathbf{S}_1) - \nabla f(\mathbf{S}_2)\| \leq L_f\|\mathbf{S}_1 - \mathbf{S}_2\|$, $\forall\, \mathbf{S}_1, \mathbf{S}_2$ in the domain of f.

Instead of directly optimizing the cost in (P2), APG algorithms minimize a sequence of overestimators, obtained at judiciously chosen points $\mathbf{T}$. Define $g(\mathbf{S}) := \nu\|\mathbf{X}\|_* + \nu\lambda\|\mathbf{A}\|_1$ and form the quadratic approximation

$$\begin{aligned} Q(\mathbf{S},\mathbf{T}) &:= f(\mathbf{T}) + \langle \nabla f(\mathbf{T}), \mathbf{S}-\mathbf{T}\rangle + \frac{L_f}{2}\|\mathbf{S}-\mathbf{T}\|_F^2 + g(\mathbf{S}) \\ &= \frac{L_f}{2}\|\mathbf{S}-\mathbf{G}\|_F^2 + g(\mathbf{S}) + f(\mathbf{T}) - \frac{1}{2L_f}\|\nabla f(\mathbf{T})\|_F^2 \end{aligned} \tag{2.29}$$

where $\mathbf{G} := \mathbf{T} - (1/L_f)\nabla f(\mathbf{T})$. With $k = 1, 2, \ldots$ denoting iterations, APG algorithms generate the sequence of iterates

$$\begin{aligned} \mathbf{S}[k] &:= \arg\min_{\mathbf{S}} Q(\mathbf{S}, \mathbf{T}[k-1]) \\ &= \arg\min_{\mathbf{S}} \left\{ \frac{L_f}{2}\|\mathbf{S}-\mathbf{G}[k]\|_F^2 + g(\mathbf{S}) \right\} \end{aligned} \tag{2.30}$$

where the second equality follows from the fact that the last two summands in (2.29) do not depend on $\mathbf{S}$. There are two key aspects to the success of APG algorithms. First, is the selection of the points $\mathbf{T}[k]$ where the sequence of approximations $Q(\mathbf{S}, \mathbf{T}[k])$ are formed, since these strongly determine the algorithm's convergence rate. The choice $\mathbf{T}[k] = \mathbf{S}[k] + \frac{t[k-1]-1}{t[k]}(\mathbf{S}[k] - \mathbf{S}[k-1])$, where $t[k] = \left[1 + \sqrt{4t^2[k-1]+1}\right]/2$, has been

shown to significantly accelerate the algorithm resulting in convergence rate no worse than $\mathcal{O}(1/k^2)$ [14]. The second key element stems from the possibility of efficiently solving the sequence of subproblems (2.30). For the particular case of (P2), note that (2.30) decomposes into

$$\mathbf{X}[k+1] := \arg\min_{\mathbf{X}} \left\{ \frac{L_f}{2}\|\mathbf{X} - \mathbf{G}_X[k]\|_F^2 + \nu\|\mathbf{X}\|_* \right\} \tag{2.31}$$

$$\mathbf{A}[k+1] := \arg\min_{\mathbf{A}} \left\{ \frac{L_f}{2}\|\mathbf{A} - \mathbf{G}_A[k]\|_F^2 + \nu\lambda\|\mathbf{A}\|_1 \right\} \tag{2.32}$$

where $\mathbf{G}[k] = [\mathbf{G}_X'[k] \ \mathbf{G}_A'[k]]'$. Letting $\mathcal{S}_\tau(\mathbf{M})$ with (i,j)-th entry given by $\text{sign}(m_{i,j})\max\{|m_{i,j}|-\tau, 0\}$ 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. [80])

$$\mathbf{X}[k+1] = \mathbf{U}\mathcal{S}_{\frac{\nu}{L_f}}[\mathbf{\Sigma}]\mathbf{V}', \quad \mathbf{A}[k+1] = \mathcal{S}_{\frac{\lambda\nu}{L_f}}[\mathbf{G}_A[k]]. \tag{2.33}$$

A continuation technique is employed to speed-up convergence of the APG algorithm. The penalty parameter ν is initialized with a large value ν_0, and is decreased geometrically until it reaches the target value of $\bar{\nu}$. The APG algorithm is tabulated as Algorithm 1. Similar to [80] and [139], the iterations terminate whenever the norm of $\mathbf{Z}[k+1]$ in

$$\mathbf{Z}[k+1] := \begin{bmatrix} L_f(\mathbf{T}_X[k] - \mathbf{X}[k+1]) + (\mathbf{X}[k+1] + \mathbf{R}\mathbf{A}[k+1] - \mathbf{T}_X[k] - \mathbf{R}\mathbf{T}_A[k]) \\ L_f(\mathbf{T}_A[k] - \mathbf{A}[k+1]) + \mathbf{R}'(\mathbf{X}[k+1] + \mathbf{R}\mathbf{A}[k+1] - \mathbf{T}_X[k] - \mathbf{R}\mathbf{T}_A[k]) \end{bmatrix}$$

drops below some prescribed tolerance, i.e., $\|\mathbf{Z}[k+1]\|_F \leq \text{tol} \times \max(1, L_f\|\mathbf{X}[k]\|_F)$. As detailed in [139], the quantity $\|\mathbf{Z}[k+1]\|_F$ upper bounds the distance between the origin and the set of subgradients of the cost in (P2), evaluated at $\mathbf{S}[k+1]$.

Before concluding this section, it is worth noting that Algorithm 1 has good convergence performance, and quantifiable iteration complexity as asserted in the following proposition adapted from [14, 80].

Proposition 2.2 [80] *Let $h(.)$ and $\{\bar{\mathbf{A}}, \bar{\mathbf{X}}\}$ denote, respectively, the cost and an optimal solution of (P2) when $\nu := \bar{\nu}$. For $k > k_0 := \frac{\log(\nu_0/\bar{\nu})}{\log(1/\upsilon)}$, the iterates $\{\mathbf{A}[k], \mathbf{X}[k]\}$ generated*

Algorithm 1 : APG solver for (P1)

input $\mathbf{Y}, \mathbf{R}, \lambda, \upsilon, \nu_0, \bar{\nu} L_f = \lambda_{\max}([\mathbf{I}_L \ \mathbf{R}]'[\mathbf{I}_L \ \mathbf{R}])$

initialize $\mathbf{X}[0] = \mathbf{X}[-1] = \mathbf{0}_{L\times T}$, $\mathbf{A}[0] = \mathbf{A}[-1] = \mathbf{0}_{F\times T}$, $t[0] = t[-1] = 1$, and set $k = 0$.

while not converged **do**

$\mathbf{T}_X[k] = \mathbf{X}[k] + \frac{t[k-1]-1}{t[k]} (\mathbf{X}[k] - \mathbf{X}[k-1])$.

$\mathbf{T}_A[k] = \mathbf{A}[k] + \frac{t[k-1]-1}{t[k]} (\mathbf{A}[k] - \mathbf{A}[k-1])$.

$\mathbf{G}_X[k] = \mathbf{T}_X[k] + \frac{1}{L_f} (\mathbf{Y} - \mathbf{T}_X[k] - \mathbf{R}\mathbf{T}_A[k])$.

$\mathbf{G}_A[k] = \mathbf{T}_A[k] + \frac{1}{L_f}\mathbf{R}' (\mathbf{Y} - \mathbf{T}_X[k] - \mathbf{R}\mathbf{T}_A[k])$.

$\mathbf{U}\mathbf{\Sigma}\mathbf{V}' = \text{svd}(\mathbf{G}_X[k]), \quad \mathbf{X}[k+1] = \mathbf{U}\mathcal{S}_{\nu[k]/L_f}(\mathbf{\Sigma})\mathbf{V}'$.

$\mathbf{A}[k+1] = \mathcal{S}_{\lambda\nu[k]/L_f}(\mathbf{G}_A[k])$.

$t[k+1] = \left[1 + \sqrt{4t^2[k]+1}\right]/2$

$\nu[k+1] = \max\{\upsilon\nu[k], \bar{\nu}\}$

$k \leftarrow k+1$

end while

return $\mathbf{X}[k]$, $\mathbf{A}[k]$

by Algorithm 1 satisfy

$$|h(\mathbf{A}[k], \mathbf{X}[k]) - h(\bar{\mathbf{A}}, \bar{\mathbf{X}})| \leq \frac{4\|\mathbf{A}[k_0] - \bar{\mathbf{A}}\|_F^2}{(k-k_0+1)^2} + \frac{4\|\mathbf{X}[k_0] - \bar{\mathbf{X}}\|_F^2}{(k-k_0+1)^2}.$$

2.7.2 Alternating-direction method-of-multipliers (ADMM) algorithm

The alternating-direction method of multipliers (AD-MoM) is an iterative augmented Lagrangian method especially well-suited for parallel processing [18], which has been proven successful to tackle the optimization tasks encountered e.g., in statistical learning problems [105], [21]. While the AD-MoM could be directly applied to (P1), $\mathbf{R}$ couples the entries of $\mathbf{A}$ and it turns out this yields more difficult ℓ_1-norm minimization subproblems per iteration. To overcome this challenge, a common technique is to introduce an auxiliary

(decoupling) variable $\mathbf{B}$, and formulate the following optimization problem

$$\text{(P3)} \quad \min_{\{\mathbf{X},\mathbf{A},\mathbf{B}\}} \|\mathbf{X}\|_* + \lambda\|\mathbf{A}\|_1$$
$$\text{s. to} \quad \mathbf{Y} = \mathbf{X} + \mathbf{R}\mathbf{B} \tag{2.34}$$
$$\mathbf{B} = \mathbf{A} \tag{2.35}$$

which is equivalent to (P1). To tackle (P3), associate Lagrange multipliers $\tilde{\mathbf{M}}$ and $\bar{\mathbf{M}}$ with the constraints (2.34) and (2.35), respectively. Next, introduce the quadratically *augmented* Lagrangian function

$$\begin{aligned}\mathcal{L}(\mathbf{X},\mathbf{A},\mathbf{B},\tilde{\mathbf{M}},\bar{\mathbf{M}}) = \|\mathbf{X}\|_* + \lambda\|\mathbf{A}\|_1 \\ + \langle\tilde{\mathbf{M}},\mathbf{B}-\mathbf{A}\rangle + \langle\bar{\mathbf{M}},\mathbf{Y}-\mathbf{X}-\mathbf{R}\mathbf{B}\rangle \\ + \frac{c}{2}\|\mathbf{Y}-\mathbf{X}-\mathbf{R}\mathbf{B}\|_F^2 + \frac{c}{2}\|\mathbf{A}-\mathbf{B}\|_F^2\end{aligned} \tag{2.36}$$

where c is a positive penalty coefficient. Splitting the primal variables into two groups $\{\mathbf{X},\mathbf{A}\}$ and $\{\mathbf{B}\}$, the AD-MoM solver entails an iterative procedure comprising three steps per iteration $k = 1, 2, \ldots$

[S1] Update dual variables:

$$\tilde{\mathbf{M}}[k] = \tilde{\mathbf{M}}[k-1] + c(\mathbf{B}[k] - \mathbf{A}[k]) \tag{2.37}$$
$$\bar{\mathbf{M}}[k] = \bar{\mathbf{M}}[k-1] + c(\mathbf{Y} - \mathbf{X}[k] - \mathbf{R}\mathbf{B}[k]) \tag{2.38}$$

[S2] Update first group of primal variables:

$$\mathbf{X}[k+1] = \arg\min_{\mathbf{X}} \left\{\frac{c}{2}\|\mathbf{Y}-\mathbf{X}-\mathbf{R}\mathbf{B}[k]\|_F^2 - \langle\bar{\mathbf{M}}[k],\mathbf{X}\rangle + \|\mathbf{X}\|_*\right\}. \tag{2.39}$$
$$\mathbf{A}[k+1] = \arg\min_{\mathbf{A}} \left\{\frac{c}{2}\|\mathbf{A}-\mathbf{B}[k]\|_F^2 - \langle\tilde{\mathbf{M}}[k],\mathbf{A}\rangle + \lambda\|\mathbf{A}\|_1\right\}. \tag{2.40}$$

[S3] Update second group of primal variables:

$$\mathbf{B}[k+1] = \arg\min_{\mathbf{B}} \left\{\frac{c}{2}\|\mathbf{Y}-\mathbf{X}[k+1]-\mathbf{R}\mathbf{B}\|_F^2 + \frac{c}{2}\|\mathbf{A}[k+1]-\mathbf{B}\|_F^2 - \langle\mathbf{R}'\bar{\mathbf{M}}[k] - \tilde{\mathbf{M}}[k],\mathbf{B}\rangle\right\} \tag{2.41}$$

This three-step procedure implements a block-coordinate descent on the augmented Lagrangian, with dual variable updates. The minimization (2.39) can be recast as (2.31), hence $\mathbf{X}[k+1]$ is iteratively updated through singular value thresholding. Likewise, (2.40) can be put in the form (2.32) and the entries of $\mathbf{A}[k+1]$ are updated via parallel soft-thresholding operations. Finally, (2.41) is a strictly convex unconstrained quadratic program, whose closed-form solution is obtained as the root of the linear equation corresponding to the first-order condition for optimality. The AD-MoM solver is tabulated under Algorithm 2. Suitable termination criteria are suggested in [21, p. 18].

Conceivably, F can be quite large, thus inverting the $F \times F$ matrix $\mathbf{R}'\mathbf{R}+\mathbf{I}_F$ to update $\mathbf{B}[k+1]$ could be complex computationally. Fortunately, the inversion needs to be carried out once, and can be performed and cached off-line. In addition, to reduce the inversion cost, the SVD of the compression matrix $\mathbf{R} = \mathbf{U}_R\mathbf{\Sigma}_R\mathbf{V}'_R$ can be obtained first, and the matrix inversion lemma can be subsequently employed to obtain $[\mathbf{R}'\mathbf{R}+\mathbf{I}_F]^{-1} = [\mathbf{I}_L - \mathbf{V}_R\mathbf{C}\mathbf{V}'_R]$, where $\mathbf{C} := \text{diag}\left(\frac{\sigma_1^2}{1+\sigma_1^2}, ..., \frac{\sigma_L^2}{1+\sigma_p^2}\right)$ and $p = \text{rank}(\mathbf{R}) \ll F$. Finally, note that the AD-MoM algorithm converges to the global optimum of the convex program (P1) as stated in the next proposition.

Proposition 2.3 [18] *For any value of the penalty coefficient $c > 0$, the iterates $\{\mathbf{X}[k], \mathbf{A}[k]\}$ converge to the optimal solution of (P1) as $k \to \infty$.*

Before moving on to performance evaluation, a couple of remarks are in order.

Remark 2.7 (Trade-off between stability and convergence rate) *The APG algorithm exhibits a convergence rate guarantee of $\mathcal{O}(1/k^2)$ [110], while AD-MoM only attains $\mathcal{O}(1/k)$ [57]. For the problem considered here, APG needs an appropriate continuation technique to achieve the predicted performance [80]. Extensive numerical tests with Algorithm 1 suggest that the convergence rate can vary considerably for different choices e.g., of the matrix $\mathbf{R}$. The AD-MoM algorithm on the other hand exhibits less variability in terms of performance, and only requires tuning c. It is also better suited for the constrained formulation (P1), since it does not need to resort to a relaxation.*

Algorithm 2 : AD-MoM solver for (P1)

input $\mathbf{Y}, \mathbf{R}, \lambda, c$

initialize $\mathbf{X}[0] = \bar{\mathbf{M}}[-1] = \mathbf{0}_{L\times T}$, $\mathbf{A}[0] = \mathbf{B}[0] = \tilde{\mathbf{M}}[-1] = \mathbf{0}_{F\times T}$, and set $k = 0$.

while not converged **do**

[S1] Update dual variables:

$\tilde{\mathbf{M}}[k] = \tilde{\mathbf{M}}[k-1] + c(\mathbf{B}[k] - \mathbf{A}[k])$

$\bar{\mathbf{M}}[k] = \bar{\mathbf{M}}[k-1] + c(\mathbf{Y} - \mathbf{X}[k] - \mathbf{R}\mathbf{B}[k])$

[S2] Update first group of primal variables:

$\mathbf{U}\mathbf{\Sigma}\mathbf{V}' = \mathrm{svd}(\mathbf{Y} - \mathbf{R}\mathbf{B}[k] + c^{-1}\bar{\mathbf{M}}[k]), \quad \mathbf{X}[k+1] = \mathbf{U}\mathcal{S}_{1/c}(\mathbf{\Sigma})\mathbf{V}'.$

$\mathbf{A}[k+1] = c^{-1}\mathcal{S}_{\lambda}(\tilde{\mathbf{M}}[k] + c\mathbf{B}[k]).$

[S3] Update second group of primal variables:

$\mathbf{B}[k+1] = \mathbf{A}[k+1] + (\mathbf{R}'\mathbf{R} + \mathbf{I}_F)^{-1}\left[\mathbf{R}'(\mathbf{Y} - \mathbf{X}[k+1] - \mathbf{R}\mathbf{A}[k+1]) - c^{-1}(\tilde{\mathbf{M}}[k] - \mathbf{R}'\bar{\mathbf{M}}[k])\right]$

$k \leftarrow k+1$

end while

return $\mathbf{A}[k], \mathbf{X}[k]$

Remark 2.8 (Distributed algorithms) *In the anomaly detection context outlined in Section 2.8.2, implementing Algorithms 1 and 2 presumes that network nodes communicate their local link traffic measurements to a central monitoring station, which uses their aggregation in* $\mathbf{Y}$ *to unveil anomalies. While for the most part this is the prevailing operational paradigm adopted in current networks, there are limitations associated with this architecture. For instance, fusing all this information may entail excessive communication overhead. Moreover, minimizing the exchanges of raw measurements may be desirable to reduce unavoidable communication errors that translate to noise and missing data. Performing the optimization in a centralized fashion also raises robustness concerns, since the central monitoring station represents an isolated point of failure. These reasons motivate devising* fully-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 in an algorithmic companion paper [95], which puts forth a general framework for in-network sparsity-regularized rank minimization.*

2.8 Performance Evaluation

The performance of (P1) is assessed in this section via computer simulations.

Selection of tuning parameters. Theorem 2.1 provides a range of parameters $\lambda \in (\lambda_{\min}, \lambda_{\max})$ such that (P1) exactly recovers $\{\mathbf{X}_0, \mathbf{A}_0\}$ in (2.1). However, it may be infeasible to compute $\{\lambda_{\min}, \lambda_{\max}\}$, since they depend on e.g., $\delta_k(\mathbf{R})$ which is NP-hard to evaluate [29]. Besides, in practice the observations (2.1) are typically contaminated with noise $\mathbf{E} \in \mathbb{R}^{L\times T}$ [cf. (2.3)]. To account for the noise, the optimization problem (P2) is considered, where for convenience the sparsity and rank-controlling parameters are redefined here as $\lambda_1 := \nu\lambda$ and $\lambda_* := \nu$, respectively. To tune $\{\lambda_1, \lambda_*\}$, a simple strategy is to optimize the relative error $\|\hat{\mathbf{A}} - \mathbf{A}_0\|_F/\|\mathbf{A}_0\|_F$, with $\mathbf{A}_0$ and $\hat{\mathbf{A}}$ denoting the true and estimated sparse matrices, respectively. In particular, one needs to perform a grid search over the bounded two-dimensional region $\mathcal{R} := \{(\lambda_1, \lambda_*) : \lambda_1 \in (0, \|\mathbf{R}'\mathbf{Y}\|_\infty],\ \lambda_* \in (0, \|\mathbf{Y}\|]\}$. The corresponding bounds are derived from the optimality conditions for (P2), which indicate that for $(\lambda_1, \lambda_*) \in \mathcal{R}^c$ the optimal solution is $\{\mathbf{0}_{L\times T}, \mathbf{0}_{F\times T}\}$.

Practical rules that do not require knowledge of $\mathbf{A}_0$ can be devised along the lines of [7] and [26]. Supposing that the true values are zero, choosing $\lambda_1 > \|\mathbf{R}'\mathbf{E}\|_\infty$ and $\lambda_* > \|\mathbf{E}\|$ the estimator (P2) outputs $\{\hat{\mathbf{X}} = \mathbf{0}_{L\times T}, \hat{\mathbf{A}} = \mathbf{0}_{F\times T}\}$. In general this choice mitigates noise, but it may overshrink the true values. To avoid overshrinking, these parameters can be chosen close to their corresponding lower bounds, e.g., pick $\lambda_* = \|\mathbf{E}\|$ and $\lambda_1 = \|\mathbf{R}'\mathbf{E}\|_\infty$. One can further simplify the candidate parameters by making the following reasonable assumptions: i) Gaussian noise $e_{l,t} \sim \mathcal{N}(0, \sigma^2)$, and ii) large dimensions $F, T \to \infty$. It is then known that $(\sqrt{F}+\sqrt{T})^{-1}\|\mathbf{E}\| \to \sigma$, almost surely, see e.g., [26], and thus one can pick $\lambda_* = (\sqrt{F}+\sqrt{T})\sigma$. Also, large deviation tail bounding implies that $\|\mathbf{R}'\mathbf{E}\|_\infty \leq 4\sigma \max_i \|\mathbf{R}\mathbf{e}_i\|_2 \log(FT)$ with high probability, which suggests selecting $\lambda_1 = \sigma \max_i \|\mathbf{R}\mathbf{e}_i\|_2 \log(FT)$. Notice that in the noiseless case ($\sigma = 0$) one can pick $\lambda = \lambda_1/\lambda_* = \max_i \|\mathbf{R}\mathbf{e}_i\| \log(FT)/(\sqrt{F} + \sqrt{T})$.

2.8.1 Exact recovery

Data matrices are generated according to $\mathbf{Y} = \mathbf{X}_0 + \mathbf{V}'_R\mathbf{A}_0$. The low-rank component $\mathbf{X}_0$ is generated from the bilinear factorization model $\mathbf{X}_0 = \mathbf{V}'_R\mathbf{W}\mathbf{Z}'$, where $\mathbf{W}$ and $\mathbf{Z}$ are $L \times r$

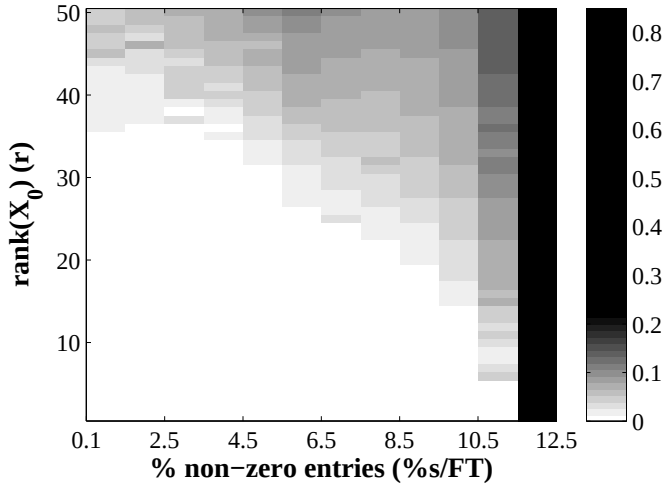


Figure 2.1: Relative error $e_r := \|\mathbf{A}_0 - \hat{\mathbf{A}}\|_F / \|\mathbf{A}_0\|_F$ for various values of r and s where $L = 105$, $F = 210$, and $T = 420$. White represents exact recovery ($e_r \approx 0$), while black represents $e_r \approx 1$.

and $T \times r$ matrices with i.i.d. entries drawn from Gaussian distributions $\mathcal{N}(0, 1/L)$ and $\mathcal{N}(0, 1/T)$, respectively. Every entry of $\mathbf{A}_0$ is randomly drawn from the set $\{-1, 0, 1\}$ with $\Pr(a_{i,j} = -1) = \Pr(a_{i,j} = 1) = \pi/2$. The columns of $\mathbf{V}_R \in \mathbb{R}^{F \times L}$ comprise the right singular vectors of the random matrix $\mathbf{R} = \mathbf{U}_R \mathbf{\Sigma}_R \mathbf{V}_R'$, with i.i.d. Bernoulli entries with parameter $1/2$ (cf. Remark 2.2). The dimensions are $L = 105$, $F = 210$, and $T = 420$. To demonstrate that (P1) is capable of recovering the exact values of $\{\mathbf{X}_0, \mathbf{A}_0\}$, the optimization problem is solved for a wide range of values of r and s using the APG algorithm (cf. Algorithm 1).

Let $\hat{\mathbf{A}}$ denote the solution of (P1) for a suitable value of λ. Fig. 2.1 depicts the relative error in recovering $\mathbf{A}_0$, namely $\|\hat{\mathbf{A}} - \mathbf{A}_0\|_F / \|\mathbf{A}_0\|_F$ for various values of r and s. It is apparent that (P1) succeeds in recovering $\mathbf{A}_0$ for sufficiently sparse $\mathbf{A}_0$ and low-rank $\mathbf{X}_0$ from the observed data $\mathbf{Y}$. Interestingly, in cases such as $s = 0.1 \times FT$ or $r = 0.3 \times \min(L, T)$ there is hope for recovery. In this example, one can exactly recover $\{\mathbf{X}_0, \mathbf{A}_0\}$ when $s = 0.0127 \times FT$ and $r = 0.2381 \times \min(L, T)$. A similar trend is observed for the recovery of $\mathbf{X}_0$, and the corresponding plot is omitted to avoid unnecessary repetition. For different sizes of the matrix $\mathbf{R}$, performance results averaged over ten realizations of the experiment are listed in Table 2.1. The smaller the compression ratio L/F becomes, less observations

Table 2.1: Recovery performance by varying the size of $\mathbf{R}$ when $r = 10$ and $\pi = 0.05$.

L	$\text{rank}(\mathbf{X}_0)$	$\|\mathbf{A}_0\|_0$	$\text{rank}(\hat{\mathbf{X}})$	$\|\hat{\mathbf{A}}\|_0$	$\|\hat{\mathbf{A}} - \mathbf{A}_0\|_F / \|\mathbf{A}_0\|_F$
F	10	4410	10	4419	2.0809×10^{-6}
$F/2$	10	4410	10	4407	6.4085×10^{-5}
$F/3$	10	4410	10	9365	7.76×10^{-2}
$F/5$	10	4410	14	14690	6.331×10^{-1}

Table 2.2: Performance comparison of LS-PCP and Algorithm 1 averaged over ten random realizations

.

Algorithm	$r = 5,\ \pi = 0.01$	$r = 5,\ \pi = 0.05$	$r = 10,\ \pi = 0.01$	$r = 10,\ \pi = 0.05$
LS-PCP	0.6901	0.6975	0.7001	0.7023
Algorithm 1	7.81×10^{-6}	3.037×10^{-5}	1.69×10^{-5}	6.4×10^{-5}

are available and performance degrades accordingly. In particular, the error performance degrades significantly for a challenging instance where $L/F = 0.2$ and $r = 0.4 \times \min(L, F)$ (cf. the last row of Table 2.1).

The results of [25] and [33] assert that exact recovery of $\{\mathbf{X}_0, \mathbf{A}_0\}$ from the observations $\mathbf{Y} = \mathbf{X}_0 + \mathbf{A}_0$ is possible under some technical conditions. Even though the algorithms therein are not directly applicable here due to the presence of $\mathbf{R}$, one may still consider applying PCP after suitable pre-processing of $\mathbf{Y}$. One possible approach is to find the LS estimate of the superposition $\mathbf{X}_0 + \mathbf{A}_0$ as $\hat{\mathbf{Y}} = \mathbf{R}^{\dagger}\mathbf{Y}$, and then feed a PCP algorithm with $\hat{\mathbf{Y}}$ to obtain $\{\mathbf{X}_0, \mathbf{A}_0\}$. Comparisons between (P1) and the aforesaid two-step procedure are summarized in Table 2.2. It is apparent that the heuristic performs very poorly, which is mainly due to the null space of matrix $\mathbf{R}$ (when $F = 2L$) that renders LS estimation inaccurate.

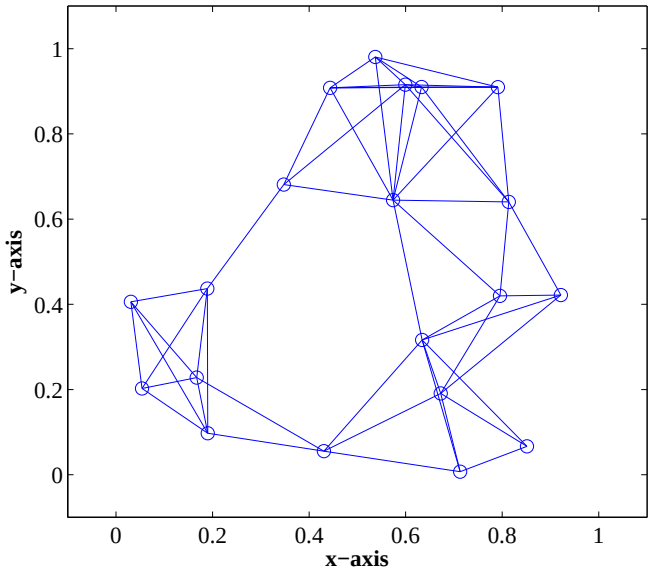


Figure 2.2: Network topology graph.

2.8.2 Unveiling network anomalies

Synthetic network data. A network of $N = 20$ agents is considered as a realization of the random geometric graph model, that is, agents are randomly placed on the unit square and two agents communicate with each other if their Euclidean distance is less than a prescribed communication range of 0.35; see Fig. 2.2. The network graph is bidirectional and comprises $L = 106$ links, and $F = N(N-1) = 380$ OD flows. For each candidate OD pair, minimum hop count routing is considered to form the routing matrix $\mathbf{R}$. With $r = 10$, matrices $\{\mathbf{X}_0, \mathbf{A}_0\}$ are generated as explained in Section 2.8.1. With reference to (2.2), the entries of $\mathbf{E}$ are i.i.d., zero-mean, Gaussian with variance σ^2, i.e., $e_{l,t} \sim \mathcal{N}(0, \sigma^2)$.

Real network data. Real data including OD flow traffic levels are collected from the operation of the Internet2 network (Internet backbone network across USA) [2]. OD flow traffic levels are recorded for a three-week operation of Internet2 during Dec. 8–28, 2003 [72]. Internet2 comprises $N = 11$ nodes, $L = 41$ links, and $F = 121$ flows. Given the OD flow traffic measurements, the link loads in $\mathbf{Y}$ are obtained through multiplication with the Internet2 routing matrix [2]. Even though $\mathbf{Y}$ is 'constructed' here from flow measurements, link loads can be typically acquired from simple network management protocol (SNMP)

traces [137]. The available OD flows are a superposition of 'clean' and anomalous traffic, i.e., the sum of unknown 'ground-truth' low-rank and sparse matrices $\mathbf{X}_0 + \mathbf{A}_0$ adhering to (2.2) when $\mathbf{R} = \mathbf{I}_L$. Therefore, PCP is applied first to obtain an estimate of the 'ground-truth' $\{\mathbf{X}_0, \mathbf{A}_0\}$. The estimated $\mathbf{X}_0$ exhibits three dominant singular values, confirming the low-rank property of $\mathbf{X}_0$.

Comparison with the PCA-based method. To highlight the merits of the proposed anomaly detection algorithm, its performance is compared with the workhorse PCA-based approach of [72]. 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 [72] is limited to a single anomalous flow per time slot. Different from [72], 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.

For the synthetic data case, ROC curves are depicted in Fig. 2.3 (top), 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 $P_F = 10^{-4}$ and $P_D = 0.97$, Fig. 2.3 (bottom) illustrates the magnitude of the true and estimated anomalies across flows and time. Similar results are depicted for the Internet2 data in Fig. 2.4, where it is also apparent that the proposed method markedly outperforms PCA in terms of detection performance. For an instance of $P_F = 0.04$ and $P_D = 0.93$, Fig. 2.4 (bottom) shows the effectiveness of the proposed algorithm in terms of unveiling the anomalous flows and time instants.

Remark 2.9 (Incoherence conditions) *For the matrices involved in the anomaly detection problem, some of the incoherence conditions required by Theorem 2.1 may not hold. For instance, with* $\mathbf{X}_0 = \mathbf{R}\mathbf{Z}_0$ *[cf. (2.2)] quantity* $\gamma_R(\mathbf{U})$ *may not be small enough. In addition, it is challenging to find binary* $\{0, 1\}$ *routing matrices with desirable RICs. Still, the conditions in Theorem 2.1 are only sufficient and the numerical tests in this section demonstrate*

that the proposed algorithm performs well in practice. This observation naturally motivates follow-up research aimed at closing this gap between theory and practice.

2.9 Closing Comments

This paper deals with recovery of low-rank plus *compressed* sparse matrices via convex optimization. The corresponding task arises with network traffic monitoring, dynamic MRI, and singing voice separation from music accompaniment, while it encompasses compressive sampling and principal components pursuit. To estimate the unknowns, a convex optimization program is formulated that mininimizes a trade-off between the nuclear and ℓ_1-norm of the low-rank and sparse components, respectively, subject to a data modeling constraint. A deterministic approach is adopted to characterize local identifiability and sufficient conditions for exact recovery via the aforementioned convex program. Intuitively, the obtained conditions require: i) incoherent, sufficiently low-rank and sparse components; and ii) a compression matrix that behaves like an isometry when operating on sparse vectors. Because these conditions are in general NP-hard to check, it is shown that matrices drawn from certain random ensembles can be recovered with high probability. First-order iterative algorithms are developed to solve the nonsmooth optimization problem, which converge to the globally optimal solution with quantifiable complexity. Numerical tests with synthetic and real network data corroborate the effectiveness of the novel approach in unveiling traffic anomalies across flows and time.

One can envision several extensions to this work, which provide new and challenging directions for future research. For instance, it seems that the requirement of an orthonormal compression matrix is only a restriction imposed by the method of proof utilized here. There should be room for tightening the bounds used in the process of constructing the dual certificate, and hence obtain milder conditions for exact recovery. Building on [34, 163], it would also be interesting to study stability of the proposed estimator in the presence of noise and missing data. In addition, one is naturally tempted to search for a broader class of matrices satisfying the exact recovery conditions, including e.g., non block-diagonal and binary routing (compression) matrices arising with the network anomaly detection task.

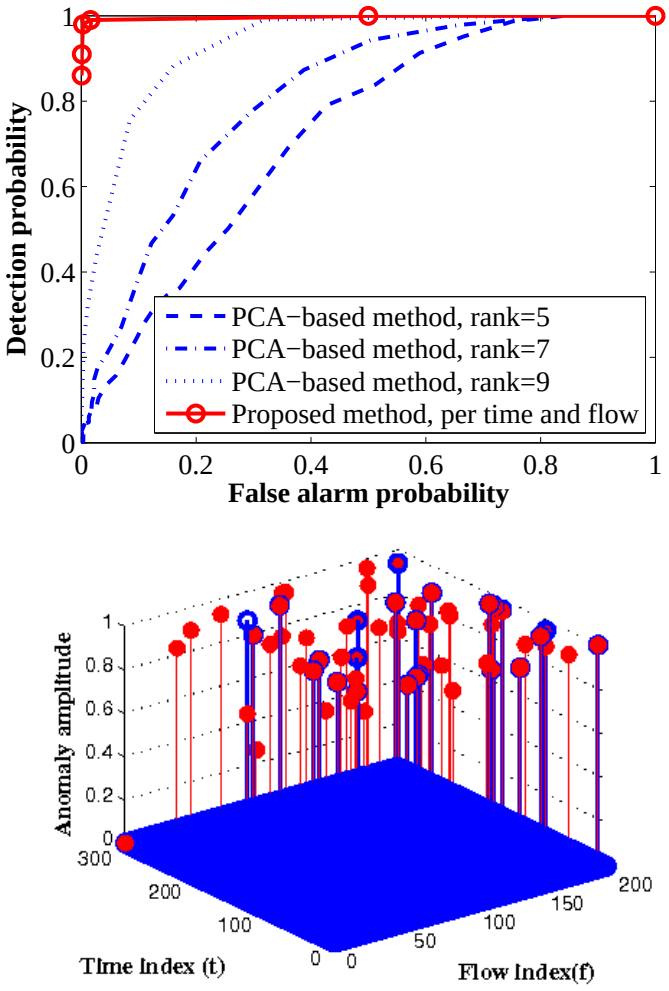


Figure 2.3: Performance for synthetic data. (Top) ROC curves of the proposed versus the PCA-based method with $\pi = 0.001$, $r = 10$ and $\sigma = 0.1$. (Bottom) Amplitude of the true and estimated anomalies for $P_F = 10^{-4}$ and $P_D = 0.97$. Lines with open and filled circle markers denote the true and estimated anomalies, respectively.

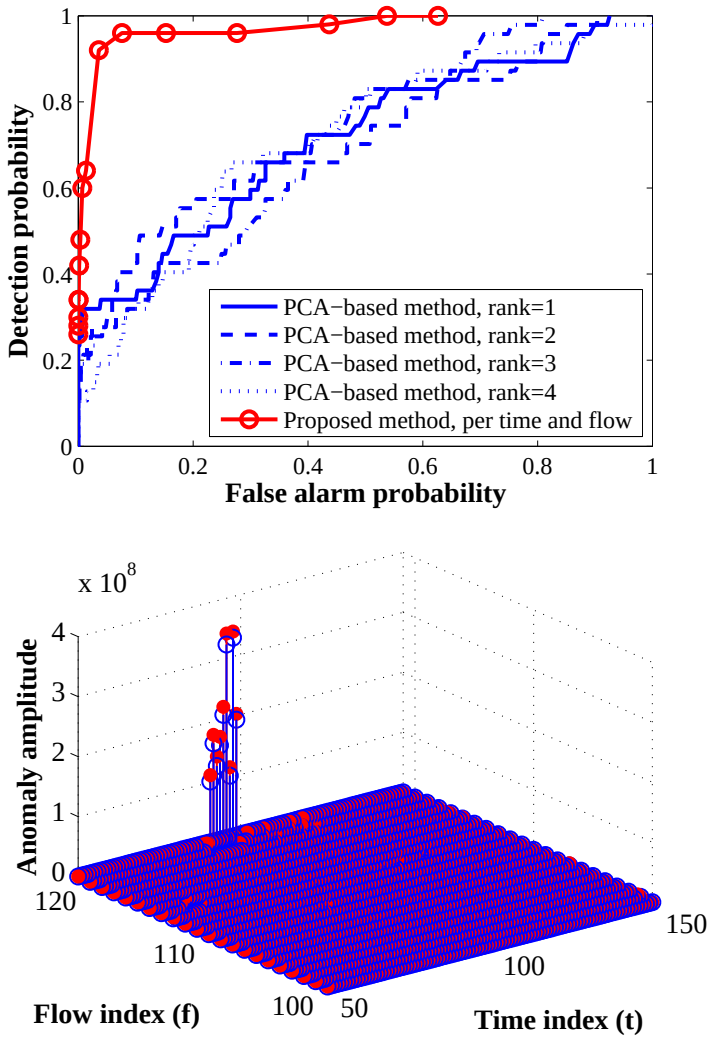


Figure 2.4: Performance for Internet2 network data. (Top) ROC curves of the proposed versus the PCA-based method. (Bottom) Amplitude of the true and estimated anomalies for $P_F = 0.04$ and $P_D = 0.93$. Lines with open and filled circle markers denote the true and estimated anomalies, respectively.

Chapter 3

Tomographic Low-Rank and Sparse Recovery: Applications to Network Traffic Monitoring

3.1 Introduction

Emergence of multimedia services and Internet-friendly portable devices is multiplying network traffic volume day by day [149]. Moreover, the advent of diverse networks of intelligent devices including those deployed to monitor the smart power grid, transportation networks, medical information networks, and cognitive radio networks, will transform the communication infrastructure to an even more complex and heterogeneous one. Thus, ensuring compliance to service-level agreements necessitates ground-breaking management and monitoring tools providing operators with informative depictions of the network state. One such atlas (set of maps) can offer a flow-time depiction of the network origin-destination (OD) flow traffic. Situational awareness provided by such maps will be the key enabler for effective routing and congestion control, network health management, risk analysis, security assurance, and proactive network failure prevention. Acquiring such diagnosis/prognosis maps for large networks however is an arduous task. This is mainly because the number

of OD pairs grows promptly as the network size grows, while probing exhaustively all OD pairs becomes impractical even for moderate-size networks [128]. In addition, OD flows potentially undergo anomalies arising due to e.g., cyberattacks and network failures [74], and the acquired measurements typically encounter misses, outliers, and errors.

Towards creating traffic maps, one typically has access to: (D1) link counts comprising the superposition of OD flows per link; these counts can be readily obtained using the single network management protocol (SNMP) [74]; and (D2) *partial* OD-flow counts recorded using e.g., the NetFlow protocol [74]. Extensive studies of backbone Internet Protocol (IP) networks reveals that the nominal OD-flow traffic is spatiotemporally correlated mainly due to common temporal patterns across OD flows, and exhibits periodic trends (e.g., daily or weekly) across time [74]. This renders the nominal traffic having a small intrinsic dimensionality. Moreover, traffic volume anomalies rarely occur across flows and time [11, 74, 113]. Given the observations (D1) and/or (D2), ample research has been carried out over the years to tackle the ill-posed traffic inference task relying on various techniques that leverage the traffic features as prior knowledge; see e.g., [32, 67, 97, 120, 146, 158, 162, 166] and references therein.

To date, the main body of work on traffic inference relies on least-squares (LS) and Gaussian [32, 162] or Poisson models [146], and entropy regularization [166]. None of these methods however takes spatiotemporal dependencies of the traffic into account. To enhance estimation accuracy by exploiting the spatiotemporal dependencies of traffic, attempts have been made in [120] and [97]. Using the prior spatial and temporal structures of traffic, [120] applies rank regularization along with matrix factorization to discover the global low-rank traffic matrix from the link and/or flow counts. The model in [120] is however devoid of anomalies, which can severely deteriorate traffic estimation quality. In the context of anomaly detection, our companion work [97] capitalizes on the low-rank of traffic and sparsity of anomalies to unveil the traffic volume anomalies from the link loads (D1). Without OD-flow counts however, the nominal flow-level traffic cannot be identified using the approach of [97].

The present work addresses these limitations by introducing a novel framework that

efficiently and scalably constructs network traffic maps. Leveraging recent advances in compressive sensing and rank minimization, first, a novel estimator is put forth, to effect sparsity and low rank attributes for the anomalous and nominal traffic components through ℓ_1- and nuclear-norm, respectively. The recovery performance of the sought estimator is then analyzed in the noise-free setting following a deterministic approach along the lines of [33]. Sufficient incoherence conditions are derived based on the angle between certain subspaces to ensure the retrieved traffic and anomaly matrices coincide with the true ones. The recovery conditions yield valuable insights about the network structures and data acquisition strategies giving rise to accurate traffic estimation. Intuitively, one can expect accurate traffic estimation if: (a) NetFlow measures sufficiently many randomly selected OD flows; (b) the OD paths are sufficiently "spread out" so as the routes form a column-incoherent routing matrix; (c) the nominal traffic is sufficiently low dimensional; and, (d) anomalies are sporadic enough.

Albeit insightful, the accurate-recovery conditions in practical networks may not hold. For instance, it may happen that a specific flow undergoes a bursty anomaly lasting for a long time [11], or certain OD flows may be inaccessible for the entire time horizon of interest with no NetFlow samples at hand. With the network practical challenges however come opportunities to exploit certain structures, and thus cope with the aforementioned challenges. This work bridges this "theory-practice" gap by incorporating the spatiotemporal patterns of the nominal and anomalous traffic, both of which can be learned from historical data. Adopting a Bayesian approach, a novel estimator is introduced for the traffic following a bilinear characterization of the nuclear- and ℓ_1-norms. The resultant nonconvex problem entails quadratic regularizers loaded with inverse correlation matrices to effect structured sparsity and low rank for anomalous and nominal traffic matrices, respectively. A systematic approach for learning traffic correlations from historical data is also devised taking advantage of the (cyclo)stationary nature of traffic. Alternating majorization-minimization algorithms are also developed to obtain iterative estimates, which are provably convergent.

Simulated tests with synthetic network and real Internet-data corroborate the effectiveness of the novel schemes, especially in reducing the number of acquired NetFlow samples

needed to attain a prescribed estimation accuracy. In addition, the proposed optimization-based approach opens the door for efficient in-network and online processing along the lines of our companion works in [95] and [96]. The novel ideas can also be applicable to various other inference tasks dealing with recovery of structured low-rank and sparse matrices.

The rest of this paper starts with preliminaries and problem statement in Section 3.2. The novel estimator to map out the nominal and anomalous traffic is discussed in Section 3.3, and pertinent reconstruction claims are established in Section 3.4. Sections 3.5 and 3.6 deal with incorporating the spatiotemporal patterns of traffic to improve estimation quality. Certain practical issues are addressed in Section 3.8. Simulated tests are reported in Section 3.9, and finally Section 3.10 draws the conclusions.

3.2 Preliminaries and Problem Statement

Consider a backbone IP network described 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. A set of end-to-end flows $\mathcal{F}$ with $|\mathcal{F}| = F$ traverse different OD pairs. In backbone networks, the number of OD flows far exceeds the number of physical links ($F \gg L$). Per OD-flow, multipath routing is considered where each flow traverses multiple possibly overlapping paths to reach its intended destination. Letting $x_{f,t}$ denote the unknown traffic level of flow $f \in \mathcal{F}$ at time t, link $\ell \in \mathcal{L}$ carries the fraction $r_{\ell,f} \in [0, 1]$ of this flow; clearly, $r_{\ell,f} = 0$ if flow f is not routed through link ℓ. The total traffic carried by link ℓ is then the weighted superposition of flows routed through link ℓ, that is, $\sum_{f\in\mathcal{F}} r_{\ell,f} x_{f,t}$. The weights $\{r_{\ell,f}\}$ form the routing matrix $\mathbf{R} \in [0, 1]^{L\times F}$, which is assumed fixed and given. These weights are not arbitrary but must respect the flow conservation law $\sum_{\ell\in\mathcal{L}_{\text{in}}(n)} r_{\ell,f} = \sum_{\ell\in\mathcal{L}_{\text{out}}(n)} r_{\ell,f},\ \forall f \in \mathcal{F}$, where $\mathcal{L}_{\text{in}}(n)$ and $\mathcal{L}_{\text{out}}(n)$ denote the sets of incoming and outgoing links to node $n \in \mathcal{N}$, respectively.

It is not uncommon for some of flow rates to experience sudden changes, which are termed *traffic volume anomalies* that are typically due to the network failures, or cyberattacks [74]. With $a_{f,t}$ denoting the unknown traffic volume anomaly of flow f at time t, the

traffic carried by link ℓ at time t is

$$y_{\ell,t} = \sum_{f \in \mathcal{F}} r_{\ell,f}(x_{f,t} + a_{f,t}) + v_{\ell,t}, \quad t \in \mathcal{T} \tag{3.1}$$

where the time horizon $\mathcal{T}$ comprises T slots, and $v_{\ell,t}$ accounts for the measurement errors. In IP networks, link loads can be readily measured via SNMP supported by most routers [74]. Introducing the matrices $\mathbf{Y} := [y_{\ell,t}], \mathbf{V} := [v_{\ell,t}] \in \mathbb{R}^{L \times T}$, $\mathbf{X} := [x_{f,t}]$, and $\mathbf{A} := [a_{f,t}] \in \mathbb{R}^{F \times T}$, link counts in (3.1) can be expressed in a compact matrix form as

$$\mathbf{Y} = \mathbf{R}\,(\mathbf{X} + \mathbf{A}) + \mathbf{V}. \tag{3.2}$$

Here, matrices $\mathbf{X}$ and $\mathbf{A}$ contain, respectively, the *nominal* and *anomalous* traffic flows over the time horizon $\mathcal{T}$. Inferring $(\mathbf{X}, \mathbf{A})$ from the compressed measurements $\mathbf{Y}$ is a severely underdetermined task (recall that $L \ll F$), necessitating additional data to ensure identifiability and improve estimation accuracy. A useful such source is the direct flow-level measurements

$$z_{f,t} = x_{f,t} + a_{f,t} + w_{f,t}, \ t \in \mathcal{T}, \ f \in \mathcal{F} \tag{3.3}$$

where $w_{f,t}$ accounts for measurement errors. The flow traffic in (3.3) is sampled via NetFlow [74] at each origin node. This however incurs high cost which means that one can have measurements (3.3) only for few (f,t) pairs [74]. To account for missing flow-level data, collect the available pairs (f,t) in the set $\Pi \in [F] \times [T]$; introduce also the matrices $\mathbf{Z}_\Pi := [z_{f,t}], \mathbf{W}_\Pi := [w_{f,t}] \in \mathbb{R}^{F \times T}$, where $z_{f,t} = w_{f,t} = 0$ for $(f,t) \notin \Pi$, and associate the sampling operator $\mathcal{P}_\Pi$ with the set Π, which assigns entries of its matrix argument not in Π equal to zero, and keeps the rest unchanged. As with $\mathbf{X}$, it holds that $\mathcal{P}_\Pi(\mathbf{X}) \in \mathbb{R}^{F \times T}$. The flow counts in (3.3) can then be compactly written as

$$\mathbf{Z}_\Pi = \mathcal{P}_\Pi\,(\mathbf{X} + \mathbf{A}) + \mathbf{W}_\Pi. \tag{3.4}$$

Besides periodicity, temporal patterns common to traffic flows render rows (correspondingly columns) of $\mathbf{X}$ correlated, and thus $\mathbf{X}$ exhibits a few dominant singular values which make it (approximately) low rank [74]. Anomalies on the other hand are expected to occur occasionally, as only a small fraction of flows are supposed to be anomalous at any given

time instant, which means $\mathbf{A}$ is sparse. Anomalies may exhibit certain patterns e.g., failure at a part of the network may simultaneously render a subset of flows anomalous; or certain flows may be subject to bursty malicious attacks over time.

Given the link counts $\mathbf{Y}$ obeying (3.2) along with the partial flow-counts $\mathbf{Z}_\Pi$ adhering to (3.4), and with $\{\mathbf{R}, \Pi\}$ known, this paper aims at accurately estimating the unknown *low-rank* nominal and *sparse* anomalous traffic pair $(\mathbf{X}, \mathbf{A})$.

3.3 Maps of Nominal and Anomalous Traffic

In order to estimate the unknowns of interest, a natural estimator accounting for the low rank of $\mathbf{X}$ and the sparsity of $\mathbf{A}$ will be sought to minimize the rank of $\mathbf{X}$, and the number of nonzero entries of $\mathbf{A}$ measured by its ℓ_0-(pseudo) norm. Unfortunately, both rank and ℓ_0-norm minimization problems are in general NP-hard [29, 109, 117]. The nuclear-norm $\|\mathbf{X}\|_* := \sum_k \sigma_k(\mathbf{X})$, where $\sigma_k(\mathbf{X})$ signifies the k-th singular value of $\mathbf{X}$, and the ℓ_1-norm $\|\mathbf{A}\|_1 := \sum_{f,t} |a_{f,t}|$ are typically adopted as *convex* surrogates [29, 117]. Accordingly, one solves

$$\text{(P1)} \quad (\hat{\mathbf{X}}, \hat{\mathbf{A}}) = \arg \min_{(\mathbf{X},\mathbf{A})} \frac{1}{2}\|\mathbf{Y} - \mathbf{R}(\mathbf{X} + \mathbf{A})\|_F^2 + \frac{1}{2}\|\mathcal{P}_\Pi(\mathbf{Z} - \mathbf{X} - \mathbf{A})\|_F^2 + \lambda_* \|\mathbf{X}\|_* + \lambda_1 \|\mathbf{A}\|_1$$

where $\lambda_1, \lambda_* \geq 0$ are the sparsity- and rank-controlling parameters. From a network operation perspective, the estimate $\hat{\mathbf{A}}$ maps out the network health-state across both time and flows. A large value $|\hat{a}_{f,t}|$ indicates that at time instant t flow f exhibits a sever anomaly, and therefore appropriate traffic engineering and security tasks need to be run to mitigate the consequences. The estimated map of nominal traffic $\hat{\mathbf{X}}$ is also a viable input for network planning tasks.

From the recovery standpoint, (P1) subsumes several important special cases, which deal with recovery of $\hat{\mathbf{X}}$ and/or $\hat{\mathbf{A}}$. In the absence of flow counts, i.e., $\Pi = \emptyset$, exact recovery of the *sparse* anomaly matrix $\hat{\mathbf{A}}$ from link loads is established in [97]. The key to this is the sparsity present, which enables recovery from compressed linear-measurements.

However, the (possibly huge) nullspace of $\mathbf{R}$ challenges identifiability of the nominal traffic matrix $\mathbf{X}$, as will be delineated later. Moreover, with only flow counts partially available, (P1) boils down to the so-termed robust principal component pursuit (PCP), for which exact reconstruction of the *low-rank* nominal traffic component is established in [33]. Instrumental role in this case is played by the dependencies among entries of the low-rank component, reflected in the observations. Indeed, the matrix of anomalies is not recoverable since observed entries do not convey any information about the unobserved anomalies. Furthermore, without the sparse matrix, i.e., $\mathbf{A} = \mathbf{0}$, and only with flow counts partially available, (P1) boils down to the celebrated matrix completion problem studied e.g., in [26], which can be applied to interpolate the traffic of unreachable OD flows from the observed ones at the edge routers.

The aforementioned considerations regarding recovery in these special cases make one hopeful to retrieve $\mathbf{X}$ and $\mathbf{A}$ via (P1). Before delving into the analysis of (P1), it is worth noting that [51] has recently studied recovery of compressed low-rank-plus-sparse matrices, also known as compressive PCP, where the compression is performed by an orthogonal projection onto a low-dimensional subspace, and the support of the sparse matrix is presumed uniformly random. The results require certain subspace incoherence conditions to hold, which in the considered traffic estimation task impose strong restrictions on the routing matrix $\mathbf{R}$ and the sampling operator $\mathcal{P}_{\Pi}(\cdot)$. Furthermore, it is unclear how to relate the subspace incoherence conditions to the well-established incoherence measures adopted in the context of matrix completion and compressive sampling, which are satisfied by various classes of random matrices; see e.g., [29, 31].

Before closing this section, it is important to recognize that albeit few the NetFlow measurement $\mathbf{Z}_{\Pi}$, they play an important role in estimating $\mathbf{X}$. In principle, if one merely knows the link counts $\mathbf{Y}$, it is impossible to accurately identify $\mathbf{X}$ when the only prior information about $\mathbf{X}$ and $\mathbf{A}$ is that they are sufficiently low-rank and sparse, respectively. This identifiability issue is formalized in the next lemma.

Lemma 3.1 *With $\mathcal{N}_R$ denoting the nullspace of $\mathbf{R}$, and $\mathbf{X}_0 = \mathbf{U}_0\mathbf{\Sigma}_0\mathbf{V}_0'$, if $\mathcal{N}_R \neq \emptyset$, and one only knows $\{\mathbf{Y}, \mathbf{R}\}$, then for any $\mathbf{W} \in \mathcal{N}_R$ the matrix pair $\{\mathbf{X}_1 := \mathbf{X}_0 + \mathbf{W}\mathbf{V}_0', \mathbf{A}_0\}$:*

(i) is feasible, and (ii) it satisfies $\text{rank}(\mathbf{X}_1) \leq \text{rank}(\mathbf{X}_0) =: r$.

Proof: Clearly (i) holds true since $\mathbf{RW} = \mathbf{0}$, and subsequently $\mathbf{R}(\mathbf{A}_0 + \mathbf{X}_1) = \mathbf{R}(\mathbf{A}_0 + \mathbf{X}_0) + \mathbf{RWV}_0' = \mathbf{Y}$. Also, (ii) readily follows from Sylvester's inequality [58] which implies that $\text{rank}(\mathbf{U}_0\mathbf{\Sigma}_0\mathbf{V}_0^{'} + \mathbf{WV}_0^{'}) \leq \min\{\text{rank}(\mathbf{X}_0 + \mathbf{WV}_0^{'}), \text{rank}(\mathbf{V}_0)\} \leq \text{rank}(\mathbf{V}_0) = r$.

3.4 Reconstruction Guarantees

This section studies the exact reconstruction performance of (P1) in the absence of noise, namely $\mathbf{V} = \mathbf{0}$ and $\mathbf{W}_\Pi = \mathbf{0}$. The corresponding formulation can be expressed as

$$\begin{aligned} \text{(P2)} \quad (\hat{\mathbf{X}}, \hat{\mathbf{A}}) = \arg \min_{(\mathbf{X},\mathbf{A})} \quad & \|\mathbf{X}\|_* + \lambda \|\mathbf{A}\|_1 \\ \text{s.to} \quad & \mathbf{Y} = \mathbf{R}\left(\mathbf{X} + \mathbf{A}\right), \quad \mathbf{Z}_\Pi = \mathcal{P}_\Pi\left(\mathbf{X} + \mathbf{A}\right). \end{aligned}$$

In the sequel, identifiability of $(\mathbf{X}, \mathbf{A})$ from the linear measurements $\{\mathbf{Y}, \mathbf{Z}_\Pi\}$ is pursued first, followed by technical conditions based on certain incoherence measures, to guarantee $(\hat{\mathbf{X}} = \mathbf{X}_0, \hat{\mathbf{A}} = \mathbf{A}_0)$, where $\mathbf{X}_0$ and $\mathbf{A}_0$ are the *true* low-rank and sparse matrices of interest.

3.4.1 Local identifiability

Let $r := \text{rank}(\mathbf{X}_0)$ and $s := \|\mathbf{A}_0\|_0$ denote the rank and sparsity level of the true matrices of interest. The first issue to address is identifiability, asserting that there is a *unique* pair $(\mathbf{X}_0, \mathbf{A}_0)$ fulfilling the data constraints: (d1) $\mathbf{Y} = \mathbf{R}(\mathbf{X}_0 + \mathbf{A}_0)$ and (d2) $\mathbf{Z}_\Pi = \mathcal{P}_\Pi(\mathbf{X}_0 + \mathbf{A}_0)$. Apparently, if multiple solutions exist, one cannot hope finding $(\mathbf{X}_0, \mathbf{A}_0)$. Before examining this issue, introduce the subspaces: (s1) $\mathcal{N}_R := \{\mathbf{H} : \mathbf{RH} = \mathbf{0}_{L\times T}\}$ as the nullspace of the linear operator $\mathbf{R}$, and (s2) $\mathcal{N}_\Pi := \{\mathbf{H} \in \mathbb{R}^{F\times T} : \text{supp}(\mathbf{H}) \subseteq \Pi^\perp\}$ as the nullspace of the linear operator $\mathcal{P}_\Pi(.)$ [$\Pi^\perp$ is the complement of Π]. If there exists a perturbation pair $(\mathbf{H}_1, \mathbf{H}_2)$ with $\mathbf{H}_1 + \mathbf{H}_2 \in \mathcal{N}_R \cap \mathcal{N}_\Pi$ so that $\mathbf{X}_0 + \mathbf{H}_1$ and $\mathbf{A}_0 + \mathbf{H}_2$ are still low-rank and sparse, one may pick the pair $(\mathbf{X}_0 + \mathbf{H}_1, \mathbf{A}_0 + \mathbf{H}_2)$ as another legitimate solution. This section aims at resolving such identifiability issues.

Let $\mathbf{U}_0\mathbf{\Sigma}_0\mathbf{V}_0'$ denote the singular value decomposition (SVD) of $\mathbf{X}_0$, and consider the subspaces: (s3) $\Phi_{X_0} := \{\mathbf{Z} \in \mathbb{R}^{F\times T} : \mathbf{Z} = \mathbf{U}_0\mathbf{W}_1' + \mathbf{W}_2\mathbf{V}_0',\ \mathbf{W}_1 \in \mathbb{R}^{T\times r},\ \mathbf{W}_2 \in \mathbb{R}^{F\times r}\}$ of

matrices in either the column or row space of $\mathbf{X}_0$; (s4) $\Omega_{A_0} := \{\mathbf{H} \in \mathbb{R}^{F\times T} : \text{supp}(\mathbf{H}) \subseteq \text{supp}(\mathbf{A}_0)\}$ of matrices whose support is contained in that of $\mathbf{A}_0$. Noteworthy properties of these subspaces are: (i) since Φ_{X_0} and $\Omega_{A_0} \subset \mathbb{R}^{F\times T}$, it is possible to directly compare elements from them; (ii) $\mathbf{X}_0 \in \Phi_{X_0}$ and $\mathbf{A}_0 \in \Omega_{A_0}$; and (iii) if $\mathbf{Z} \in \mathbf{\Phi}_X^{\perp}$ is added to $\mathbf{X}_0$, then $\text{rank}(\mathbf{Z} + \mathbf{X}_0) > r$, and likewise $\mathbf{Z} \in \mathcal{N}_\Omega$, for any $\mathbf{Z} \in \Omega_{A_0}^{\perp}$.

Suppose temporarily that the subspaces Φ_{X_0} and Ω_{A_0} are also known. This extra piece of information helps identifiability based on data (d1) and (d2) since the potentially troublesome solutions

$$\Upsilon_1 := \{(\mathbf{X}_0 + \mathbf{H}_1, \mathbf{A}_0 + \mathbf{H}_2) : \mathbf{H}_1 + \mathbf{H}_2 \in \mathcal{N}_R \cap \mathcal{N}_\Pi\} \tag{3.5}$$

are restricted to a smaller set. If $(\mathbf{X}_0 + \mathbf{H}_1, \mathbf{A}_0 + \mathbf{H}_2) \notin \Upsilon_2$, where

$$\Upsilon_2 := \{(\mathbf{X}_0 + \mathbf{H}_1, \mathbf{A}_0 + \mathbf{H}_2) : \mathbf{H}_1 \in \Phi_{X_0},\ \mathbf{H}_2 \in \Omega_{A_0}\} \tag{3.6}$$

that candidate solution is not admissible since it is known a priori that $\mathbf{X}_0 \in \Phi_{X_0}$ and $\mathbf{A}_0 \in \Omega_{A_0}$. This notion of exploiting additional knowledge to assure uniqueness is known as *local identifiability* [33]. Global identifiability from (d1) and (d2) is not guaranteed. However, local identifiability will become essential later on to establish the main result. With these preliminaries, the following lemma puts forth the necessary and sufficient conditions for local identifiability.

Lemma 3.2 *Matrices* $(\mathbf{X}_0, \mathbf{A}_0)$ *satisfy (d1) and (d2) uniquely if and only if (c1)* $\Phi_{X_0} \cap \Omega_{A_0} = \{\mathbf{0}\}$*; and, (c2)* $\Upsilon_1 \cap \Upsilon_2 = \{\mathbf{0}\}$.

Condition (c1) implies that for the solutions in Υ_2 to be admissible, $\mathbf{H}_1 + \mathbf{H}_2$ must belong to the subspace $\Phi_{X_0} \oplus \Omega_{A_0}$. Accordingly, (c2) holds true if

$$\mathcal{N}_R \cap \mathcal{N}_\Pi \cap (\Phi_{X_0} \oplus \Omega_{A_0}) = \{\mathbf{0}\}. \tag{3.7}$$

Notice that (c1) appears also in the context of low-rank-plus-sparse recovery results in [25, 33]. However, (c2) is unique to the setting here. It captures the impact of the overlap between the nullspace of $\mathbf{R}$ and the operator $\mathcal{P}_\Pi(\cdot)$. Finding simpler sufficient conditions to assure (c1) and (c2) is dealt with next.

3.4.2 Incoherence measures

The overlap between any pair of subspaces $\{\Phi_{X_0}, \Omega_{A_0}, \mathcal{N}_R, \mathcal{N}_\Pi\}$ plays a crucial role in identifiability and exact recovery as seen e.g., from Lemma 3.1. To quantify the overlap of the subspaces e.g., Φ_{X_0} and Ω_{A_0}, consider the *incoherence* parameter

$$\mu(\Phi_{X_0}, \Omega_{A_0}) := \max_{\substack{\mathbf{X} \in \Omega_{A_0} \\ \|\mathbf{X}\|_F = 1}} \|\mathcal{P}_{\Phi_{X_0}}(\mathbf{X})\|_F, \tag{3.8}$$

which clearly satisfies $\mu(\Phi_{X_0}, \Omega_{A_0}) \in [0, 1]$. The lower bound is achieved when Φ_{X_0} and Ω_{A_0} are orthogonal, whereas the upperbound is attained when $\Phi_{X_0} \cap \Omega_{A_0}$ contains a nonzero element. To gain further geometric intuition, $\mu(\Phi_{X_0}, \Omega_{A_0})$ represents the cosine of the angle between subspaces when they have trivial intersection, namely $\Phi_{X_0} \cap \Omega_{A_0} = \{\mathbf{0}\}$ [42]. Small values of $\mu(\Phi_{X_0}, \Omega_{A_0})$ indicate sufficient separation between Φ_{X_0} and Ω_{A_0}, and thus less chance of ambiguity when discerning $\mathbf{X}_0$ from $\mathbf{A}_0$.

It will be seen later that (c1) requires $\mu(\Phi_{X_0}, \Omega_{A_0}) < 1$. In addition, to ensure (c2) one needs the incoherence parameter $\mu(\mathcal{N}_R \cap \mathcal{N}_\Pi, \Phi_{X_0} \oplus \Omega_{A_0}) < 1$. In fact, $\mu(\mathcal{N}_R \cap \mathcal{N}_\Pi, \Phi_{X_0} \oplus \Omega_{A_0})$ captures the ambiguity inherent to the nullspace of the compression and sampling operators. It depends on all subspaces (s1)–(s4), and it is desirable to express it in terms of the incoherence of different subspace pairs, namely $\mu(\mathcal{N}_R, \Omega_{A_0})$, $\mu(\mathcal{N}_R, \Phi_{X_0})$, $\mu(\mathcal{N}_\Pi, \Omega_{A_0})$, and $\mu(\mathcal{N}_\Pi, \Phi_{X_0})$. This is formalized in the next claim.

Proposition 3.1 *Assume that* $\mu(\Omega_{A_0}, \Phi_{X_0}) < 1$. *If either* $\dim(\mathcal{N}_R \cap \mathcal{N}_\Pi) = 0$*; or,* $\dim(\mathcal{N}_R \cap \mathcal{N}_\Pi) \geq 1$ *and*

$$\chi := \Big[\frac{\mu(\mathcal{N}_\Pi, \Phi_{X_0}) + \mu(\mathcal{N}_R, \Omega_{A_0})\mu(\mathcal{N}_\Pi, \Omega_{A_0})}{1 - \mu(\Omega_{A_0}, \Phi_{X_0})}\Big]^{1/2} < 1$$

hold, then $\Phi_{X_0} \cap \Omega_{A_0} = \{\mathbf{0}\}$ *and* $\mathcal{N}_R \cap \mathcal{N}_\Pi \cap (\Phi_{X_0} \oplus \Omega_{A_0}) = \{\mathbf{0}\}$.

Proof: Since $\mu(\Omega_{A_0}, \Phi_{X_0}) < 1$ and $\dim(\Phi_{\mathrm{X_0}} \oplus \Omega_{\mathrm{A_0}} \oplus (\mathcal{N}_\mathrm{R} \cap \mathcal{N}_\Pi)) = \dim(\Phi_{\mathrm{X_0}}) + \dim(\Omega_{\mathrm{A_0}}) + \dim(\mathcal{N}_\mathrm{R} \cap \mathcal{N}_\Omega)$, [51, Lemma 11] implies that

$$\begin{aligned}\mu^2(\Phi_{X_0} \oplus \Omega_{A_0}, \mathcal{N}_R \cap \mathcal{N}_\Pi) \leq \big[1 - \mu(\Phi_{X_0}, \Omega_{A_0})\big]^{-1} \\ \times \big[\mu^2(\Phi_{X_0}, \mathcal{N}_R \cap \mathcal{N}_\Pi) + \mu^2(\Omega_{A_0}, \mathcal{N}_R \cap \mathcal{N}_\Pi)\big].\end{aligned} \tag{3.9}$$

The result then follows by bounding $\mu^2(\Phi_{X_0}, \mathcal{N}_R \cap \mathcal{N}_\Pi) \leq \mu(\Phi_{X_0}, \mathcal{N}_R)\mu(\Phi_{X_0}, \mathcal{N}_\Pi)$ using the fact that $\mathcal{N}_R \cap \mathcal{N}_\Pi \in \mathcal{N}_R, \mathcal{N}_\Pi$ [likewise for $\mu(\Omega_{A_0}, \mathcal{N}_R \cap \mathcal{N}_\Pi)$], and $\mathcal{N}_R \cap \mathcal{N}_{\Phi_{X_0}} \neq \{\mathbf{0}\}$.

Apparently, small values of $\mu(\mathcal{N}_R, \Omega_{A_0})$ and $\mu(\mathcal{N}_\Pi, \Phi_{X_0})$ gives rise to a small χ. In fact, $\mu(\mathcal{N}_R, \Omega_{A_0})$ measures whether $\mathcal{N}_R$ contains sparse elements, and it is tightly related to the incoherence among the sparse column-subsets of $\mathbf{R}$. For row-orthonormal compression matrices in particular, where $\mathbf{R}\mathbf{R}' = \mathbf{I}$, the incoherence reduces to the restricted isometry constant of $\mathbf{R}$, see e.g., [29]. Moreover, $\mu(\mathcal{N}_\Pi, \Phi_{X_0})$ measures whether the low-rank matrices fall into the nullspace of the subsampling operator $\mathcal{P}_\Pi(\cdot)$, that is tightly linked to the incoherence metrics introduced in the context of matrix completion; see e.g., [27]. It is worth mentioning that a wide class of matrices resulting in small incoherence $\mu(\mathcal{N}_R, \Omega_{A_0})$, $\mu(\mathcal{N}_\Pi, \Phi_{X_0})$ and $\mu(\Omega_{A_0}, \Phi_{X_0})$ are provided in [29], [27], [25], which give rise to a sufficiently small value of χ.

3.4.3 Exact recovery via convex optimization

Besides $\mu(\Omega_{A_0}, \Phi_{X_0})$ and χ, there are other incoherence measures which play an important role in the conditions for exact recovery. These measures are introduced to avoid ambiguity when the (feasible) perturbations $\mathbf{H}_1$ and $\mathbf{H}_2$ do not necessarily belong to the subspaces Φ_{X_0} and Ω_{A_0}, respectively. Before moving on, it is worth noting that these measures resemble the ones for matrix completion and decomposition problems; see e.g., [25, 27]. For instance, consider a feasible solution $\{\mathbf{X}_0 + a_{i,j}\mathbf{e}_i\mathbf{e}_j', \mathbf{A}_0 + a_{i,j}\mathbf{e}_i\mathbf{e}_j'\}$, where $(i,j) \notin \text{supp}(\mathbf{A}_0)$, and thus $a_{i,j}\mathbf{e}_i\mathbf{e}_j' \notin \Omega_{A_0}$. It may happen that $a_{i,j}\mathbf{e}_i\mathbf{e}_j' \in \Phi_{X_0}$ and $\text{rank}(\mathbf{X}_0 + a_{i,j}\mathbf{e}_i\mathbf{e}_j') = \text{rank}(\mathbf{X}_0) - 1$, while $\|\mathbf{A}_0 - a_{i,j}\mathbf{e}_i\mathbf{e}_j'\|_0 = \|\mathbf{A}_0\|_0 + 1$, thus challenging identifiability when Φ_{X_0} and Ω_{A_0} are unknown. Similar complications arise if $\mathbf{X}_0$ has a sparse row space that can be confused with the row space of $\mathbf{A}_0$. These issues motivate defining

$$\gamma(\mathbf{U}_0) := \max_i \|\mathbf{P}_U \mathbf{e}_i\|, \qquad \gamma(\mathbf{V}_0) := \max_i \|\mathbf{P}_V \mathbf{e}_i\| \tag{3.10}$$

where $\mathbf{P}_U := \mathbf{U}_0\mathbf{U}_0'$ (resp. $\mathbf{P}_V := \mathbf{V}_0\mathbf{V}_0'$) are the projectors onto the column (row) space of $\mathbf{X}_0$. Notice that $\gamma(\mathbf{U}_0), \gamma(\mathbf{V}_0) \in [0,1]$. The maximum of $\gamma(\mathbf{U}_0)$ (resp. $\gamma(\mathbf{V}_0)$) is attained when $\mathbf{e}_i$ is in the column (row) space of $\mathbf{X}_0$ for some i. Small values of $\gamma(\mathbf{U}_0)$ (resp. $\gamma(\mathbf{V}_0)$) imply that the column (row) spaces of $\mathbf{X}_0$ do not contain sparse vectors, respectively.

Another identifiability instance arises when $\mathbf{X}_0$ is sparse, in which case each column of $\mathbf{X}_0$ is spanned by a few canonical basis vectors. Consider the parameter

$$\gamma(\mathbf{U}_0, \mathbf{V}_0) := \|\mathbf{U}_0\mathbf{V}_0'\|_\infty = \max_{i,j} |\mathbf{e}_i{}'\mathbf{U}_0\mathbf{V}_0\mathbf{e}_j|. \tag{3.11}$$

A small value of $\gamma(\mathbf{U}_0, \mathbf{V}_0)$ indicates that each column of $\mathbf{X}_0$ is spanned by sufficiently many canonical basis vectors. It is worth noting that $\gamma(\mathbf{U}_0, \mathbf{V}_0)$ can be bounded in terms of $\gamma(\mathbf{U}_0)$ and $\gamma(\mathbf{V}_0)$, but it is kept here for the sake of generality.

From (c2) in Lemma 3.1 it is evident that the dimension of the nullspace $\mathcal{N}_R \cap \mathcal{N}_\Pi$ is critical for identifiability. In essence, the lower $\dim(\mathcal{N}_R \cap \mathcal{N}_\Pi)$ is, the higher is the chance for exact reconstruction. In order to quantify the size of the nullspace, define

$$\tau(\mathcal{N}_R, \mathcal{N}_\Pi) := \max_{\substack{\mathbf{X} \in \mathcal{N}_R \cap \mathcal{N}_\Pi \\ \|\mathbf{X}\|=1}} \|\mathbf{X}\|_\infty \tag{3.12}$$

which will appear later in the exact recovery conditions. All elements are now in place to state the main result.

3.4.4 Main result

Theorem 3.1 *Let $(\mathbf{X}_0, \mathbf{A}_0)$ denote the true low-rank and sparse matrix pair of interest, and define $\mathbf{X}_0 := \mathbf{U}_0\mathbf{\Sigma}_0\mathbf{V}_0'$, $r := rank(\mathbf{X}_0)$, and $s := \|\mathbf{A}_0\|_0$. Assume that $\mathbf{A}_0$ has at most k nonzero elements per column, and define the incoherence parameters $\alpha := \mu(\Omega_{A_0}, \Phi_{X_0})$, $\beta := \mu(\Omega_{A_0}, \mathcal{N}_R)$, $\xi := \mu(\mathcal{N}_\Pi, \Phi_{X_0})$, $\nu := \mu(\mathcal{N}_R, \Omega_{A_0} \cap \mathcal{N}_\Pi)$, $\eta := \gamma(\mathbf{U}_0) + \gamma(\mathbf{V}_0)$, $\tau := \tau(\mathcal{N}_R, \mathcal{N}_\Pi)$, $\gamma := \gamma(\mathbf{U}_0, \mathbf{V}_0)$. Given $\mathbf{Y}$ and $\mathbf{Z}_\Pi$ adhering to (d1) and (d2), respectively, with known $\mathbf{R}$ and Π, if $\chi < 1$, and*

$$\text{(I)} \quad \lambda_{\max} := (\frac{1}{k})\frac{1 - \alpha - \alpha^3(1-\alpha^2) - ge/f}{1 + \alpha^2(1-\alpha^2) + he/f}$$
$$> \lambda_{\min} := \frac{\gamma + qg/f}{1 - \eta\alpha k - kqh/f} \geq 0$$
$$\text{(II)} \quad f := 1 - \nu\beta - (\xi + \alpha\nu)(1-\alpha^2)(\xi + \alpha\beta) > 0$$

hold, where

$$g := \xi + \alpha(\xi + \alpha\nu)(1-\alpha^2)\alpha, \quad h := \nu + \alpha(1-\alpha^2)(\xi + \alpha\nu)$$
$$q := \tau + \eta\alpha + \eta\xi, \quad e := \alpha(1-\alpha^2)(\xi + \alpha\beta) + 1 + \nu$$

then for any $\lambda_{\min} \leq \lambda \leq \lambda_{\max}$ *the convex program (P1) yields* $(\hat{\mathbf{X}} = \mathbf{X}_0, \hat{\mathbf{A}} = \mathbf{A}_0)$.

The identifiability claim of Theorem 1 under certain conditions, holds deterministically. In addition, Theorem 1 guarantees reconstruction for several important special cases such as PCP and matrix completion as discussed in Section 3.3. Its proof given in the Appendix is inspired by [33], which deals with PCP (meaning recovery of low-rank plus compressed sparse matrices). The proof technique first derives conditions in terms of certain dual variables, thus ensuring that the true $(\mathbf{X}_0, \mathbf{A}_0)$ provide the unique optimal solution of (P2). Subsequently, under conditions (I) and (II) the proof constructs dual certificates. Relative to [33] however, the model here introduces new challenges due to the nullspace of compression and sampling operators that necessitate in part a proof strategy distinct from [33].

3.4.5 Satisfiability

Satisfaction of the conditions in Theorem 3.1 hinges upon the incoherence parameters $\{\alpha, \gamma, \eta, \xi, \tau\}$ whose sufficiently small values fulfill (I) and (II). In fact, these parameters are increasing functions of the rank r and the sparsity level s. In particular, $\{\alpha, \gamma, \eta\}$ capture the ambiguity of the additive components $\mathbf{X}_0$ and $\mathbf{A}_0$, and are known to be small enough for small values of $\{r, s, k\}$; see e.g., [27, 33]. Regarding χ, recall that it is an increasing function of β and ξ, where ξ takes a small value when NetFlow samples an adequately large subset of OD flows uniformly at random. Moreover, in large-scale networks with distant OD node pairs, and routing paths that are sufficiently "spread out", the sparse column-subsets of $\mathbf{R}$ tend to be incoherent, and thus β takes a small value. Likewise, for sufficiently many NetFlow samples and column-incoherent routing matrices, τ takes a small value.

Finding a proper class of matrices satisfying (I) and (II) and simplifying the conditions in terms of more interpretable quantities such as r, s, k is a daunting task, and goes beyond the scope of this paper. However, to gain insight about possible admissible matrices and the roadblocks involved, as it is customary in the context of low rank and sparse recovery (see e.g., [26,33,97]), focus on a class of random matrices generated as follows: "random orthogonal" low-rank matrices $\mathbf{X}_0 = \mathbf{U}\mathbf{\Sigma}\mathbf{V}'$, where the orthonormal singular vector matrices

$\mathbf{U} \in \mathbb{R}^{F\times r}$ and $\mathbf{V} \in \mathbb{R}^{T\times r}$ are uniformly drawn from the collection of rank-r partial isometries; sparse matrix $\mathbf{A}_0$ with a uniform support Ω_{A_0}; the uniformly sampled set Π; and the "bounded orthonormal" compression matrix $\mathbf{R}$ with $\mathbf{R}\mathbf{R}' = \mathbf{I}_L$. According to [26, 33, 97] it then follows that $\eta = \mathcal{O}(\sqrt{r/F})$ and $\gamma = \mathcal{O}(\log(F)\sqrt{r}/F)$. Moreover, it holds that $\alpha = \mathcal{O}(s/(FT))$, and $\xi = \mathcal{O}(|\Pi_\perp|/(FT))$. Parameter β also coincides with the restricted isometry constant of $\mathbf{R}$ that is $\mathcal{O}(\sqrt{k/L})$. What is left to specify is τ – a challenging task because it involves intersection of the null spaces $\mathcal{N}_R$ and $\mathcal{N}_\Pi$. In fact, simplifying τ is the major roadblock toward finding simple recovery conditions, and it is left for future research. Nonetheless, for sufficiently low-rank $\mathbf{X}_0$ and "sparse enough" $\mathbf{A}_0$, namely $r, s \ll$, and with large ambient dimensions $L, F, T \gg$, the aforementioned coherence quantities can be rendered arbitrarily small, and consequently the conditions of Theorem 3.1 can be met with high probability.

3.4.6 ADMM algorithm

This section introduces an iterative solver for the convex program (P2) using the alternating direction method of multipliers (ADMM) method. ADMM is an iterative augmented Lagrangian method especially well-suited for parallel processing [18], and has been proven successful to tackle the optimization tasks encountered e.g., in statistical learning; see e.g., [21]. While ADMM could be directly applied to (P2), $\mathbf{R}$ couples the entries of $\mathbf{A}$ and $\mathbf{X}$ leading to computationally demanding nuclear- and ℓ_1-norm minimization subtasks per iteration. To overcome this hurdle, a common trick is to introduce auxiliary (decoupling) variables $\{\mathbf{B}, \mathbf{O}\}$, and formulate the following optimization problem

$$\begin{aligned}(\text{P3}) \quad \min_{\{\mathbf{A},\mathbf{X},\mathbf{O},\mathbf{B}\}} & \|\mathbf{X}\|_* + \lambda\|\mathbf{A}\|_1 \\ \text{s. to } & \mathbf{Y} = \mathbf{R}(\mathbf{O}+\mathbf{B}), \quad \mathbf{Z}_\Pi = \mathcal{P}_\Pi(\mathbf{O}+\mathbf{B}) \\ & \mathbf{B} = \mathbf{A}, \quad \mathbf{O} = \mathbf{X},\end{aligned}$$

which is equivalent to (P2). To tackle (P3), associate the Lagrange multipliers $\{\mathbf{M}_y, \mathbf{M}_z, \mathbf{M}_a, \mathbf{M}_x\}$ with the constraints, and then introduce the quadratically *augmented*

Lagrangian function

$$\begin{aligned}\mathcal{L}(\mathbf{X},\mathbf{A},\mathbf{B},\mathbf{O};\mathbf{M}_y,\mathbf{M}_z,\mathbf{M}_a,\mathbf{M}_x) :=& \|\mathbf{X}\|_* + \lambda\|\mathbf{A}\|_1 + \langle \mathbf{M}_y, \mathbf{Y} - \mathbf{R}(\mathbf{O}+\mathbf{B})\rangle + \langle \mathbf{M}_a, \mathbf{B}-\mathbf{A}\rangle \\ &+ \langle \mathbf{M}_z, \mathbf{Z}_\Pi - \mathcal{P}_\Pi(\mathbf{O}+\mathbf{B})\rangle + \langle \mathbf{M}_x, \mathbf{O}-\mathbf{X}\rangle \\ &+ \frac{c}{2}\|\mathbf{Y}-\mathbf{R}(\mathbf{O}+\mathbf{B})\|_F^2 + \frac{c}{2}\|\mathbf{Z}_\Pi - \mathcal{P}_\Pi(\mathbf{O}+\mathbf{B})\|_F^2 \\ &+ \frac{c}{2}\|\mathbf{B}-\mathbf{A}\|_F^2 + \frac{c}{2}\|\mathbf{O}-\mathbf{X}\|_F^2 \end{aligned} \tag{3.13}$$

where $c > 0$ is a penalty coefficient. Splitting the primal variables into two groups $\{\mathbf{X}, \mathbf{B}\}$ and $\{\mathbf{A}, \mathbf{O}\}$, the ADMM solver entails an iterative procedure comprising three steps per iteration $k = 1, 2, \ldots$

[S1] Update dual variables:

$$\mathbf{M}_y[k] = \mathbf{M}_y[k-1] + c(\mathbf{Y} - \mathbf{R}(\mathbf{O}[k] + \mathbf{B}[k])) \tag{3.14}$$

$$\mathbf{M}_z[k] = \mathbf{M}_z[k-1] + c(\mathbf{Z}_\Pi - \mathcal{P}_\Pi(\mathbf{O}+\mathbf{B})) \tag{3.15}$$

$$\mathbf{M}_a[k] = \mathbf{M}_a[k-1] + c(\mathbf{B}[k] - \mathbf{A}[k]) \tag{3.16}$$

$$\mathbf{M}_x[k] = \mathbf{M}_x[k-1] + c(\mathbf{O}[k] - \mathbf{X}[k]) \tag{3.17}$$

[S2] Update first group of primal variables:

$$\begin{aligned}\mathbf{A}[k+1] &= \arg\min_{\mathbf{A}\in\mathbb{R}^{F\times T}} \left\{ \frac{c}{2}\|\mathbf{A} - \mathbf{B}[k]\|_F^2 - \langle \mathbf{M}_a[k], \mathbf{A}\rangle + \lambda\|\mathbf{A}\|_1 \right\}. \\ \mathbf{O}[k+1] &= \arg\min_{\mathbf{O}\in\mathbb{R}^{F\times T}} \left\{ \frac{c}{2}\|\mathbf{O} - \mathbf{X}[k]\|_F^2 + \frac{c}{2}\|\mathbf{Y} - \mathbf{R}(\mathbf{O}+\mathbf{B}[k])\|_F^2 \right. \\ &\quad + \frac{c}{2}\|\mathbf{Z}_\Pi - \mathcal{P}_\Pi(\mathbf{O}+\mathbf{B}[k])\|_F^2 \\ &\quad \left. + \langle \mathbf{M}_x[k] - \mathbf{R}'\mathbf{M}_y[k] - \mathcal{P}_\Pi(\mathbf{M}_z[k]), \mathbf{O}\rangle \right\}. \end{aligned}$$

[S3] Update second group of primal variables:

$$\begin{aligned}\mathbf{X}[k+1] &= \arg\min_{\mathbf{X}\in\mathbb{R}^{F\times T}} \left\{ \frac{c}{2}\|\mathbf{X} - \mathbf{O}[k]\|_F^2 - \langle \mathbf{M}_x[k], \mathbf{X}\rangle + \|\mathbf{X}\|_* \right\} \\ \mathbf{B}[k+1] &= \arg\min_{\mathbf{B}\in\mathbb{R}^{F\times T}} \{ \frac{c}{2}\|\mathbf{A}[k] - \mathbf{B}\|_F^2 + \frac{c}{2}\|\mathbf{Y} - \mathbf{R}(\mathbf{O}[k]+\mathbf{B})\|_F^2 \\ &\quad + \frac{c}{2}\|\mathbf{Z}_\Pi - \mathcal{P}_\Pi(\mathbf{O}[k]+\mathbf{B})\|_F^2 \\ &\quad + \langle \mathbf{M}_a[k] - \mathbf{R}'\mathbf{M}_y[k] - \mathcal{P}_\Pi(\mathbf{M}_z[k]), \mathbf{B}\rangle \} \end{aligned}$$

The resulting iterative solver is tabulated under Algorithm 3. Here, $[\mathcal{S}_\tau(\mathbf{X})]_{i,j} := \text{sgn}(x_{i,j})\max\{|x_{i,j}|-\tau, 0\}$ refers to the soft-thresholding operator; the vectors $\{\mathbf{y}_t, \mathbf{o}_t, \mathbf{a}_t, \mathbf{b}_t, \mathbf{z}_t\}$ and $\{\mathbf{x}_t, \mathbf{m}_t^z, \mathbf{m}_t^a, \mathbf{m}_t^x, \mathbf{m}_t^y\}$ denote the t-th column of their corresponding matrix arguments, and the diagonal matrix $\mathbf{\Pi}_t \in \{0,1\}^{P\times P}$ is unity at (i,i)-th entry if $(i,t) \in \Pi$, and zero otherwise. Algorithm 3 reveals that the update for the anomaly matrix entails a soft-thresholding operator to promote sparsity, while the nominal traffic is updated via singular value thresholding to effect low rank. The updates for $\mathbf{B}$ and $\mathbf{O}$ are also parallelized across the rows. Due to convexity of (P3), Algorithm 3 with two Gauss-Seidel block updates is convergent to the global optimum of (P2) as stated next.

Proposition 3.2 *[18] For any value of the penalty coefficient $c > 0$, the iterates $\{\mathbf{X}[k], \mathbf{A}[k]\}$ converge to the optimal solution of (P2) as $k \to \infty$.*

3.5 Incorporating Spatiotemporal Correlation Information

Being convex (P1) is appealing, and as Theorem 3.1 asserts for the noiseless case it reconstructs reliably the underlying traffic when: (c1) the anomalous traffic is sufficiently "sporadic" across time and flows; (c2) the nominal traffic matrix is sufficiently low-rank with non-spiky singular vectors; (c3) NetFlow *uniformly* samples OD flows; and, (c4) the routing paths are sufficiently "spread out." In practical networks however, these conditions may be violated, and as a consequence (P1) may perform poorly. For instance, if a bursty anomaly occurs, (c1) does not hold. A particular OD flow may also be inaccessible to sample via NetFlow, that violates (c3). Apparently, in the latter case, knowing the cross-correlation of a missing OD flow with other flows enables accurate interpolation of misses.

Inherent patterns of the nominal traffic matrix $\mathbf{X}$ and the anomalous traffic matrix $\mathbf{A}$ can be learned from historical/training data $\{\mathbf{x}_t, \mathbf{a}_t\}_{t\in\mathcal{H}}$, where $\mathbf{x}_t$ and $\mathbf{a}_t$ denote the network-wide nominal and anomalous traffic vectors at time t. Given the training data $\{\mathbf{x}_t, \mathbf{a}_t\}_{t\in\mathcal{H}}$, link counts $\mathbf{Y}$ obeying (3.2) as well as the partial flow-counts $\mathbf{Z}_\Pi$ adhering to (3.4), and with $\{\mathbf{R}, \Pi\}$ known, the rest of this paper deals with estimating the matrix pair $(\mathbf{X}, \mathbf{A})$.

Algorithm 3 : ADMM solver for (P2)

input $\mathbf{Y}, \mathbf{Z}_\Pi, \Pi, \mathbf{R}, \lambda, c, \{\mathbf{H}_t := (\mathbf{I}_F + \mathbf{\Pi}_t + \mathbf{R}'\mathbf{R})^{-1}\}_{t=1}^T$

initialize $\mathbf{M}_y[-1] = \mathbf{0}_{L\times T}$, $\mathbf{X}[0] = \mathbf{O}[0] = \mathbf{A}[0] = \mathbf{B}[0] = \mathbf{M}_z[-1] = \mathbf{M}_a[-1] = \mathbf{M}_x[-1] = \mathbf{0}_{F\times T}$, and set $k = 0$.

while not converged **do**

[S1] Update dual variables:

$\mathbf{M}_y[k] = \mathbf{M}_y[k-1] + c(\mathbf{Y} - \mathbf{R}(\mathbf{O}[k] + \mathbf{B}[k]))$

$\mathbf{M}_z[k] = \mathbf{M}_z[k-1] + c(\mathbf{Z}_\Pi - \mathcal{P}_\Pi(\mathbf{O}[k] + \mathbf{B}[k]))$

$\mathbf{M}_a[k] = \mathbf{M}_a[k-1] + c(\mathbf{B}[k] - \mathbf{A}[k])$

$\mathbf{M}_x[k] = \mathbf{M}_x[k-1] + c(\mathbf{O}[k] - \mathbf{X}[k])$

[S2] Update first group of primal variables:

$\mathbf{A}[k+1] = \mathcal{S}_{\frac{\lambda}{c}}(c^{-1}\mathbf{M}_a[k] + \mathbf{B}[k])$.

Update in parallel $(t = 1, \ldots, T)$

$\mathbf{o}_t[k+1] = \mathbf{H}_t\big(c\mathbf{x}_t[k] + c\mathbf{\Pi}_t\mathbf{z}_t + c\mathbf{R}'\mathbf{y}_t - c[\mathbf{\Pi}_t + \mathbf{R}'\mathbf{R}]\mathbf{b}_t[k] + \mathbf{R}'\mathbf{m}_t^y[k] + \mathbf{\Pi}_t\mathbf{m}_t^z[k] - \mathbf{m}_t^x[k]\big)$

[S3] Update second group of primal variables:

$\mathbf{U}\mathbf{\Sigma}\mathbf{V}' = \mathrm{svd}(\mathbf{O}[k+1] + c^{-1}\mathbf{M}_x[k]), \quad \mathbf{X}[k+1] = \mathbf{U}\mathcal{S}_{1/c}(\mathbf{\Sigma})\mathbf{V}'$

Update in parallel $(t = 1, \ldots, T)$

$\mathbf{b}_t[k+1] = \mathbf{H}_t\big(c\mathbf{a}_t[k+1] + c\mathbf{\Pi}_t\mathbf{z}_t + c\mathbf{R}'\mathbf{y}_t - c[\mathbf{\Pi}_t + \mathbf{R}'\mathbf{R}]\mathbf{o}_t[k+1] + \mathbf{R}'\mathbf{m}_t^y[k] + \mathbf{\Pi}_t\mathbf{m}_t^z[k] - \mathbf{m}_t^a[k]\big)$

$k \leftarrow k+1$

end while

return $(\mathbf{A}[k], \mathbf{X}[k])$

3.5.1 Bilinear factorization

The first step toward incorporating correlation information is to use the bilinear characterization of the nuclear norm. Using singular value decomposition [58], one can always factorize the low-rank component as $\mathbf{X} = \mathbf{L}\mathbf{Q}'$, where $\mathbf{L} \in \mathbb{R}^{F\times\rho}$, $\mathbf{Q} \in \mathbb{R}^{T\times\rho}$, for some $\rho \geq \mathrm{rank}(\mathbf{X})$. The nuclear-norm can then be redefined as (see e.g., [134])

$$\|\mathbf{X}\|_* := \min_{\mathbf{X}=\mathbf{L}\mathbf{Q}'} \frac{1}{2}\{\|\mathbf{L}\|_F^2 + \|\mathbf{Q}\|_F^2\}. \tag{3.18}$$

For the scalar case, (3.18) leads to the identity $|a| = \min_{a=bc} \frac{1}{2}(|b|^2+|c|^2)$. The latter implies that the ℓ_1-norm of $\mathbf{A}$ can be alternatively defined as

$$\|\mathbf{A}\|_1 := \min_{\mathbf{A}=\mathbf{B}\odot\mathbf{C}} \frac{1}{2}\{\|\mathbf{B}\|_F^2 + \|\mathbf{C}\|_F^2\} \tag{3.19}$$

where $\mathbf{B}, \mathbf{C} \in \mathbb{R}^{F\times T}$. For notational convenience, let $\mathbf{U} := [\mathbf{Y}',\ \mathbf{Z}_{\Pi}']$ and the corresponding linear operator $\mathcal{P}(\mathbf{X}) := [(\mathbf{R}\mathbf{X})',\ \mathcal{P}_{\Omega}(\mathbf{X})']$. Leveraging (3.18) and (3.19), one is prompted to recast (P1) as

$$\text{(P4)} \quad \min_{\{\mathbf{L},\mathbf{Q},\mathbf{B},\mathbf{C}\}} \ \frac{1}{2}\|\mathbf{U} - \mathcal{P}(\mathbf{L}\mathbf{Q}' + \mathbf{B}\odot\mathbf{C})\|_F^2 + \frac{\lambda_*}{2}\big\{\|\mathbf{L}\|_F^2 + \|\mathbf{Q}\|_F^2\big\} + \frac{\lambda_1}{2}\big\{\|\mathbf{B}\|_F^2 + \|\mathbf{C}\|_F^2\big\}.$$

This Frobenius-norm regularization doubles the number of optimization variables for the sparse component $\mathbf{A}$ ($2FT$), but reduces the variable count for the low-rank component $\mathbf{X}$ to $\rho(F+T)$. Regarding performance, the bilinear factorization incurs no loss of optimality as stated in the next lemma.

Lemma 3.3 *If $\hat{\mathbf{X}}$ denotes the optimal low-rank solution of (P1) and $\rho \geq \text{rank}(\hat{\mathbf{X}})$, then (P4) is equivalent to (P1).*

Proof: It readily follows from (3.18) and (3.19) along with the commutative property of minimization which allows taking minimization first with respect to (w.r.t.) $\{\mathbf{L}, \mathbf{Q}\}$ and then w.r.t. $\{\mathbf{B}, \mathbf{C}\}$.

3.6 Bayesian Traffic and Anomaly Estimates

This section recasts (P4) in a Bayesian framework by adopting the AWGN model

$$\mathbf{U} = \mathcal{P}(\mathbf{X} + \mathbf{A}) + \mathbf{E},$$

where $\mathbf{E}$ contains independent identically distributed (i.i.d.) entries drawn from $\mathcal{N}(0, \sigma^2)$. As in (3.18) $\mathbf{X}$ is also factorized as $\mathbf{L}\mathbf{Q}'$ with the independent factors $\mathbf{L} := [\mathbf{l}_1, \ldots, \mathbf{l}_\rho]$ and $\mathbf{Q} := [\mathbf{q}_1, \ldots, \mathbf{q}_\rho]$. Matrices $\mathbf{L}$ and $\mathbf{Q}$ are formed by i.i.d. columns obeying $\mathbf{l}_i \sim \mathcal{N}(0, \mathbf{R}_L)$

and $\mathbf{q}_i \sim \mathcal{N}(0, \mathbf{R}_Q)$, respectively, for positive-definite correlation matrices $\mathbf{R}_L \in \mathbb{R}^{F\times F}$ and $\mathbf{R}_Q \in \mathbb{R}^{T\times T}$. Without loss of generality (w.l.o.g.), in order to avoid the scalar ambiguity in $\mathbf{X} = \mathbf{L}\mathbf{Q}'$ set $\mathrm{tr}(\mathbf{R}_\mathrm{L}) = \mathrm{tr}(\mathbf{R}_\mathrm{Q})$. Likewise, the anomaly matrix is factored as $\mathbf{A} = \mathbf{B} \odot \mathbf{C}$ with the independent factors $\mathbf{b} := \mathrm{vec}(\mathbf{B}) \in \mathbb{R}^{FT}$ and $\mathbf{c} := \mathrm{vec}(\mathbf{C}) \in \mathbb{R}^{FT}$ drawn from $\mathbf{b} \sim \mathcal{N}(0, \mathbf{R}_B)$ and $\mathbf{c} \sim \mathcal{N}(0, \mathbf{R}_C)$, with positive-definite correlation matrices $\mathbf{R}_B, \mathbf{R}_C \in \mathbb{R}^{FT\times FT}$, respectively.

For the considered AWGN model with priors, the maximum a posteriori (MAP) estimator of $(\mathbf{X}, \mathbf{A})$ is given by the solution of

$$\begin{aligned}(\mathrm{P5}) \quad \min_{\{\mathbf{L},\mathbf{Q},\mathbf{B},\mathbf{C}\}} & \frac{1}{2}\|\mathbf{U} - \mathcal{P}(\mathbf{L}\mathbf{Q}' + \mathbf{B}\odot\mathbf{C})\|_F^2 \\ & + \frac{\lambda_1}{2}\left[\mathbf{b}'\mathbf{R}_B^{-1}\mathbf{b} + \mathbf{c}'\mathbf{R}_C^{-1}\mathbf{c}\right] + \frac{\lambda_*}{2}\left[\mathrm{tr}(\mathbf{L}'\mathbf{R}_L^{-1}\mathbf{L}) + \mathrm{tr}(\mathbf{Q}'\mathbf{R}_Q^{-1}\mathbf{Q})\right]\end{aligned}$$

for $\lambda_1 = \lambda_* = \sigma^2$, where different weights λ_1 and λ_* are considered here for generality. Observe that (P5) specializes to (P4) upon choosing $\mathbf{R}_L = \mathbf{I}_F$, $\mathbf{R}_Q = \mathbf{I}_T$, and $\mathbf{R}_B = \mathbf{R}_C = \mathbf{I}_{FT}$. Lemma 3.3 then implies that the convex program (P1) yields the MAP optimal estimator for the considered statistical model so long as the factors contain i.i.d. Gaussian entries. With respect to the statistical model for the low-rank and sparse components, as it will become clear later on, $\mathbf{R}_L$ ($\mathbf{R}_Q$) captures the correlation among columns (rows) of $\mathbf{X}$; likewise, $\mathbf{R}_B$ and $\mathbf{R}_C$ capture the correlation among entries of $\mathbf{A}$.

Albeit clear in this section statistical formulation, the adopted model $\mathbf{X} = \mathbf{L}\mathbf{Q}'$ promotes low rank as a result of $\mathrm{rank}(\mathbf{X}) \leq \rho$, but it is not obvious whether $\mathbf{A} = \mathbf{B} \odot \mathbf{C}$ effects sparsity. The latter will rely on the fact that the product of two independent Gaussian random variables is heavy tailed. To recognize this, consider the independent scalar random variables $b \sim \mathcal{N}(0,1)$ and $c \sim \mathcal{N}(0,1)$. The product random variable $a = bc$ can then be expressed as $bc = \frac{1}{4}(b+c)^2 - \frac{1}{4}(b-c)^2$, where $S_1 := \frac{1}{4}(b+c)^2$ and $S_2 := \frac{1}{4}(b-c)^2$ are central χ^2-distributed random variables. Since $\mathbb{E}[(a-b)(a+b)] = 0$, the random variables S_1 and S_2 are independent, and consequently the characteristic function of a admits the simple form $\Phi_a(\omega) = \Phi_{S_1}(\omega)\Phi_{S_2}(\omega) = 1/(\sqrt{1+4\omega^2})$. Applying the inverse Fourier transform to $\Phi_a(\omega)$, yields the probability density function $p_a(x) = (1/\sqrt{2\pi})k_0(x/2)$, where $k_0(x) := \int_0^\infty [\cos(\omega x)]/(\sqrt{1+4\omega^2})\, d\omega$ denotes the modified Bessel function of second-kind, which

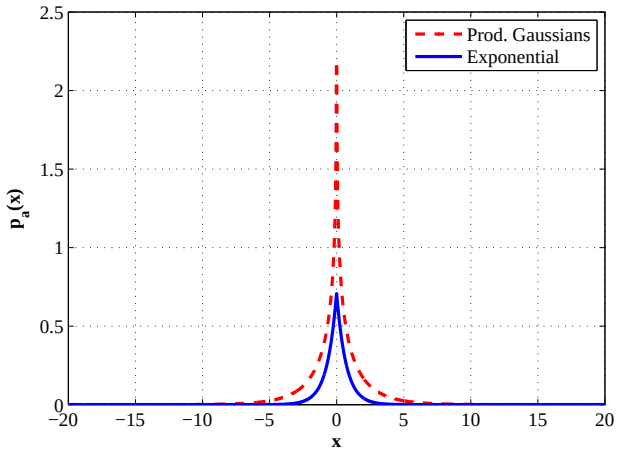


Figure 3.1: Sparsity promoting priors with zero mean and unity variance.

is tightly approximated with $\sqrt{\pi/(2x)}e^{-x}$ for $x > 1$ [123, p. 20]. One can then readily deduce that $p_a(x) = \sqrt{\pi/(2x)}e^{-|x|}$ behaves similar to the Laplacian distribution, which is well known to promote sparsity. In contrast with the Laplacian distribution however, the product of Gaussian random variables incurs a slightly lighter tail as depicted in Fig. 3.1. It is worth commenting that the correlated multivariate Laplacian distribution is an alternative prior distribution to postulate for the sparse component. However, its complicated form [46] renders the optimization for the MAP estimator intractable.

Remark 3.1 (nonzero mean) *In general, one can allow nonzero mean for the factors in the adopted statistical model, and subsequently replaces correlations with covariances. This can be useful e.g., to estimate the nominal traffic which is inherently positive valued. The mean values are assumed zero here for simplicity.*

3.6.1 Learning the correlation matrices

Implementing (P5) requires first obtaining the correlation matrices $\{\mathbf{R}_L, \mathbf{R}_Q, \mathbf{R}_B, \mathbf{R}_C\}$ from the second-order statistics of $(\mathbf{X}, \mathbf{A})$, or their estimates based on training data. Given second-order statistics of the unknown nominal-traffic matrix $\mathbf{X}$, matrices $\{\mathbf{R}_L, \mathbf{R}_Q\}$ can be readily found as explained in the next lemma. The proof is along the lines of [13], hence

it is omitted for brevity.

Lemma 3.4 *Under the Gaussian bilinear model for* $\mathbf{X}$*, and with* $\mathrm{tr}(\mathbf{R}_{\mathrm{L}}) = \mathrm{tr}(\mathbf{R}_{\mathrm{Q}})$*, it holds that*

$$\mathbf{R}_Q = \rho\mathbb{E}[\mathbf{X}'\mathbf{X}]/(\mathbb{E}[\|\mathbf{X}\|_F^2])^{1/2},$$
$$\mathbf{R}_L = \rho\mathbb{E}[\mathbf{X}\mathbf{X}']/(\mathbb{E}[\|\mathbf{X}\|_F^2])^{1/2}.$$

It is evident that $\mathbf{R}_L$ captures *temporal* correlation of the network traffic (columns of $\mathbf{X}$), while $\mathbf{R}_Q$ captures the *spatial* correlation across OD flows (rows of $\mathbf{X}$).

For real data where the distribution of unknowns is not available, $\{\mathbf{R}_L, \mathbf{R}_Q\}$ are typically estimated from the training data, which can be e.g., past estimates of nominal and anomalous traffic. For instance, consider $\{\mathbf{R}_L, \mathbf{R}_Q\}$ estimates as input to (P5) for estimating the traffic at day $K+1$ (corresponding to time horizon $\mathcal{T}$) with T time instants, from the training data $\{\mathbf{x}_t\}_{t=1}^{KT}$ collected during the past K days. Apparently, reliable correlation estimates cannot be formed for general nonstationary processes. Empirical analysis of Internet traffic suggests adopting the following assumptions [74]: (a1) Process $\{\mathbf{x}_t\}$ is cyclostationary with a day-long period due to large-scale periodic trends in the nominal traffic; and (a2) OD flows are uncorrelated as their origins are mutually unrelated. One can also take into account weekly or monthly periodicity of traffic usage to further improve the accuracy of the correlation estimates.

Let r_t denote the remainder of dividing t by T. For time slots $t_1, t_2 \in \mathcal{T}$, (a1) asserts that the vector subprocesses $\{\mathbf{x}_{kT+r_{t_1}}\}_{k=0}^{K-1}$ and $\{\mathbf{x}_{kT+r_{t_2}}\}_{k=0}^{K-1}$ are stationary, and thus one can consistently estimate $\mathbb{E}[\mathbf{x}'_{r_{t_1}}\mathbf{x}_{r_{t_2}}]$, to obtain $\mathbf{R}_Q$ via the sample correlation $\frac{1}{K}\sum_{k=0}^{K-1}\mathbf{x}_{kT+r_{t_1}}\mathbf{x}'_{kT+r_{t_2}}$ [54]. Likewise, the normalization term $\mathbb{E}[\|\mathbf{X}\|_F^2]$ is estimated relying on (a1) as $\frac{1}{K}\sum_{t=1}^{T}\sum_{k=0}^{K-1}\|\mathbf{x}_{kT+t}\|^2$. Estimating $\mathbf{R}_L$ on the other hand relies on (a2). Let $\boldsymbol{\xi}'_f \in \mathbb{R}^T$ denote the time-series of traffic associated with OD flow f, namely the f-th row of $\mathbf{X}$. It then follows from (a2) that $\mathbb{E}[\boldsymbol{\xi}_{f_1}\boldsymbol{\xi}_{f_2}] = (\mathbb{E}[\boldsymbol{\xi}_{f_1}])'(\mathbb{E}[\boldsymbol{\xi}_{f_2}])$ for $f_1 \neq f_2 \in \mathcal{F}$, where due to (a1), $\mathbb{E}[\xi_{f,t}]$ ($\xi_{f,t}$ signifies the t-th entry of $\boldsymbol{\xi}_f$) is estimated via the sample mean $\frac{1}{K}\sum_{k=0}^{K-1}x_{f,kT+r_t}$. Moreover, for $f_1 = f_2 = f$, the estimate for $\mathbb{E}[\boldsymbol{\xi}'_f\boldsymbol{\xi}_f]$ is $\frac{1}{K}\sum_{k=0}^{K-1}\sum_{t=1}^{T}\xi^2_{f,kT+r_t}$.

Given the second-order statistics of $\mathbf{A}$, the correlation matrices $\mathbf{R}_B$ and $\mathbf{R}_C$ are obtained next.

Lemma 3.5 *Under the Gaussian bilinear model for* $\mathbf{a} = \text{vec}(\mathbf{A}')$*, it holds that* $\mathbb{E}[\mathbf{a}\mathbf{a}'] = \mathbf{R}_B \odot \mathbf{R}_C$.

In order to avoid the scalar ambiguity present in $\mathbf{R}_B$ and $\mathbf{R}_C$, assume equal-magnitude entries $|[\mathbf{R}_B]_{i,j}| = |[\mathbf{R}_C]_{i,j}| = |\big[\mathbb{E}[\mathbf{a}\mathbf{a}']\big]_{i,j}|^{1/2}$, $\forall(i,j)$. Apparently, for a diagonal correlation matrix $\mathbb{E}[\mathbf{a}\mathbf{a}']$, the factors are uniquely determined as $[\mathbf{R}_B]_{i,i} = [\mathbf{R}_C]_{i,i} = \big[\mathbb{E}[\mathbf{a}\mathbf{a}']\big]_{i,i}^{1/2}$, $\forall i$. However, when nonzero off-diagonals are present, there may exist a sign ambiguity, and the signs should be assigned appropriately to guarantee that $\mathbf{R}_B$ and $\mathbf{R}_C$ are positive definite.

Correlation matrices $\{\mathbf{R}_B, \mathbf{R}_C\}$ required to run (P5) over the time horizon $\mathcal{T}$ ($|\mathcal{T}| = T$) are estimated from the training data $\{\mathbf{a}_t\}_{t=1}^{KT}$ collected e.g., over the past K days. Due to the diverse nature of anomalies, developing a universal methodology to learn $\mathbf{R}_B$ and $\mathbf{R}_C$ is an ambitious objective. Depending on the nature of anomalies, the learning process is possible under certain assumptions. One such reasonable assumption is that anomalies of different flows are uncorrelated, but for each OD flow, the anomalous traffic is stationary and possibly correlated over time. This model is appropriate e.g., when different flows are subject to bursty anomalies arising from unrelated external sources.

For the stationary anomaly process of flow f, namely $\{a_{f,t}\}_t$, let $R_a^{(f)}(\tau) := \mathbb{E}[a_{f,t-\tau}a_{f,t}]$ denote the time-invariant cross-correlation. Let also $\boldsymbol{\alpha}_f'$ denote the f-th row of $\mathbf{A}$, and introduce the correlation matrix $\mathbf{R}_a^{(f)} := \mathbb{E}[\boldsymbol{\alpha}_f\boldsymbol{\alpha}_f'] \in \mathbb{R}^T$, which is Toeplitz with entries $[\mathbf{R}_a^{(f)}]_{i,i+\tau} = R_a^{(f)}(\tau)$, $i \in [T], \tau = 0, \ldots, T-1$. Accordingly, $\mathbb{E}[\mathbf{a}\mathbf{a}']$ is a block-diagonal matrix with blocks $\mathbf{R}_a^{(f)}$, and subsequently Lemma 3.5 implies that $\mathbf{R}_B$ and $\mathbf{R}_C$ are block diagonal with Toeplitz blocks $\mathbf{R}_b^{(f)}$ and $\mathbf{R}_c^{(f)}$, respectively. Under the equal-magnitude assumption for the entries of $\mathbf{R}_B$ and $\mathbf{R}_c$, the entries of $\mathbf{R}_b^{(f)}$ and $\mathbf{R}_c^{(f)}$ are readily obtained as

$$\begin{aligned}\big[\mathbf{R}_b^{(f)}\big]_{i,i+\tau} &= \big|R_a^{(f)}(\tau)\big|^{1/2},\\ \big[\mathbf{R}_c^{(f)}\big]_{i,i+\tau} &= \big|R_a^{(f)}(\tau)\big|^{1/2}\text{sgn}\big(R_a^{(f)}(\tau)\big).\end{aligned} \tag{3.20}$$

Notice that if $|R_a^{(f)}(\tau)|$ decays sufficiently fast as τ grows, $\mathbf{R}_B$ and $\mathbf{R}_C$ become positive definite [131]. Finally, thanks to the stationarity of $\{a_{f,t}\}_t$, $R_a(\tau)$ can be consistently estimated using $\frac{1}{KT-\tau}\sum_{t=\tau+1}^{KT} a_{f,t-\tau}a_{f,t}$. It is worth noting that the considered model renders the sparsity regularizer in (P5) separable across rows of $\mathbf{A}$, which in turn induces row-wise sparsity.

3.7 Alternating Majorization-Minimization Algorithm

In order to efficiently solve (P5), an alternating minimization (AM) scheme is developed here by alternating among four matrix variables $\{\mathbf{L}, \mathbf{Q}, \mathbf{B}, \mathbf{C}\}$. The algorithm entails iterations updating one matrix variable at a time, while keeping the rest are kept fixed at their up-to-date values. In particular, iteration k comprises orderly updates of four matrices $\mathbf{L}[k] \to \mathbf{Q}[k] \to \mathbf{B}[k] \to \mathbf{C}[k]$. For instance, $\mathbf{L}[k]$ is updated given the latest updates $\{\mathbf{Q}[k-1], \mathbf{B}[k-1], \mathbf{C}[k-1]\}$ as $\mathbf{L}[k] = \arg\min_{\mathbf{L}} g_L^{(k)}(\mathbf{L})$, where

$$g_L^{(k)}(\mathbf{L}) := \frac{1}{2}\|\mathbf{U} - \mathcal{P}(\mathbf{L}\mathbf{Q}'[k-1] + \mathbf{B}[k-1] \odot \mathbf{C}[k-1])\|_F^2 + \frac{\lambda_*}{2}\mathrm{tr}\big(\mathbf{L}'\mathbf{R}_{\mathrm{L}}^{-1}\mathbf{L}\big) \tag{3.21}$$

Likewise, $\mathbf{Q}[k]$, $\mathbf{B}[k]$, and $\mathbf{C}[k]$ are updated by respectively minimizing $g_Q^{(k)}, g_B^{(k)}$, and $g_C^{(k)}$, which are given similar to $g_L^{(k)}$ based on latest updates of the corresponding variables.

Functions $\{g_L^{(k)}, g_Q^{(k)}, g_B^{(k)}, g_C^{(k)}\}$ are strongly convex quadratic programs due to regularization with positive definite correlations in the regularizer, and thus their solutions admits closed form after inverting certain possibly large-size matrices. For instance, updating $\mathbf{L}[k]$ requires inverting an $F\rho \times F\rho$ matrix. This however may not be affordable since in practice the number of flows F is typically $\mathcal{O}(N^2)$, which can be too large. To cope with this curse of dimensionality, instead of $\{g_L^{(k)}, g_Q^{(k)}, g_B^{(k)}, g_C^{(k)}\}$ judicious surrogates $\{\tilde{g}_L^{(k)}, \tilde{g}_Q^{(k)}, \tilde{g}_B^{(k)}, \tilde{g}_C^{(k)}\}$, chosen based on the second-order Taylor-expansion around the previous updates, are minimized. As will be clear later, adopting these surrogates avoids inversion, and parallelizes

the computations. The aforementioned surrogate for $g_L^{(k)}$ around $\mathbf{L}[k-1]$ is given as

$$\begin{aligned}\tilde{g}_L^{(k)}(\mathbf{L}) :=& g_L^{(k)}(\mathbf{L}[k-1]) + \mathrm{tr}\big((\mathbf{L}-\mathbf{L}[k-1])'\nabla g_L^{(k)}(\mathbf{L}[k-1])\big) \\ &+ \frac{\mu_L[k]}{2}\|\mathbf{L}-\mathbf{L}[k-1]\|_F^2 \end{aligned} \tag{3.22}$$

for some $\mu_L[k] \geq \sigma_{\max}\big[\nabla^2 g_L^{(k)}(\mathbf{L}[k-1])\big]$ (likewise for $\tilde{g}_Q^{(k)}, \tilde{g}_B^{(k)}$, and $\tilde{g}_C^{(k)}$). It is useful to recognize that each surrogate, say $\tilde{g}_L^{(k)}$, has the following properties: (i) it majorizes $g_L^{(k)}$, namely $g_L^{(k)}(\mathbf{L}) \leq \tilde{g}_L^{(k)}(\mathbf{L}),\ \forall \mathbf{L}$; and it is locally tight, which means that (ii) $g_L^{(k)}(\mathbf{L}[k-1]) = \tilde{g}_L^{(k)}(\mathbf{L}[k-1])$; and, (iii) $\nabla g_L^{(k)}(\mathbf{L}[k-1]) = \nabla \tilde{g}_L^{(k)}(\mathbf{L}[k-1])$.

The sought approximation leads to an iterative procedure, where iteration k entails orderly updating $\{\mathbf{L}[k], \mathbf{Q}[k], \mathbf{B}[k], \mathbf{C}[k]\}$ by minimizing $\tilde{g}_L^{(k)}, \tilde{g}_Q^{(k)}, \tilde{g}_B^{(k)}, \tilde{g}_C^{(k)}$, respectively; e.g., the update for $\mathbf{L}[k]$ is

$$\mathbf{L}[k] = \arg\min_{\mathbf{L}\in\mathbb{R}^{F\times\rho}} \tilde{g}_L^{(k)}(\mathbf{L}) = \mathbf{L}[k-1] - (\mu_L[k])^{-1}\nabla g_L^{(k)}(\mathbf{L}[k-1])$$

which is a nothing but a single step of gradient descent on $g_L^{(k)}$. Upon defining the residual matrices $\mathbf{\Phi}_y(\mathbf{L},\mathbf{Q},\mathbf{B},\mathbf{C}) := \mathbf{R}(\mathbf{L}\mathbf{Q}' + \mathbf{B}\odot\mathbf{C}) - \mathbf{Y}$ and $\mathbf{\Phi}_z(\mathbf{L},\mathbf{Q},\mathbf{B},\mathbf{C}) := \mathcal{P}_\Pi(\mathbf{L}\mathbf{Q}' + \mathbf{B}\odot\mathbf{C}) - \mathbf{Z}_\Pi$, the overall algorithm is listed in Table 4.

All in all, Algorithm 4 amounts to an iterative block-coordinate-descent scheme with four block updates per iteration, each minimizing a tight surrogate of (P5). Since each subproblem is smooth and strongly convex, the convergence follows from [116] as stated next.

Proposition 3.3 *[116] Upon choosing* $\{c_L' \geq \mu_L[k] \geq \sigma_{\max}\big[\nabla^2 g_L^{(k)}(\mathbf{L}[k-1])\big]\}_{k=1}^{\infty}$ *for some* $c_L' > 0$ *(likewise for* $\mu_Q[k], \mu_B[k], \mu_C[k]$*), the iterates* $\{\mathbf{L}[k], \mathbf{Q}[k], \mathbf{B}[k], \mathbf{C}[k]\}$ *generated by Algorithm 4 converge to a stationary point of (P5).*

Remark 3.2 (Fast algorithms) *In order to speed up the gradient descent iterations per block of Algorithm 4, Nesterov-type acceleration techniques along the lines of those introduced in e.g., [110] can be deployed, which can improve the* $\mathcal{O}(1/k)$ *convergence rate of the standard gradient descent to* $\mathcal{O}(1/k^2)$.

Algorithm 4 : Alternating majorization-minimization solver for (P5)

input $\mathbf{Y}, \mathbf{Z}_\Pi, \Pi, \mathbf{R}, \mathbf{R}_L, \mathbf{R}_Q, \mathbf{R}_B, \mathbf{R}_C, \lambda_*, \lambda_1,$
and $\{\mu_L[k], \mu_Q[k], \mu_B[k], \mu_C[k]\}_{k=1}^{\infty}$.

initialize $\mathbf{L}[0], \mathbf{Q}[0], \mathbf{B}[0], \mathbf{C}[0]$ at random, and set $k = 0$.

while not converged **do**

[S1] Update L

$\mathbf{F}[k] = \mathbf{R}'\mathbf{\Phi}_y(\mathbf{L}[k], \mathbf{Q}[k], \mathbf{B}[k], \mathbf{C}[k]) + \mathbf{\Phi}_z(\mathbf{L}[k], \mathbf{Q}[k], \mathbf{B}[k], \mathbf{C}[k])$

$\mathbf{L}[k+1] = \mathbf{L}[k] - \frac{1}{\mu_L[k]}\big(\mathbf{F}[k]\mathbf{Q}[k] + \lambda_* \mathbf{R}_L^{-1}\mathbf{L}[k]\big)$

[S2] Update Q

$\mathbf{G}[k] = \mathbf{\Phi}'_y(\mathbf{L}[k+1], \mathbf{Q}[k], \mathbf{B}[k], \mathbf{C}[k])\mathbf{R} + \mathbf{\Phi}'_z(\mathbf{L}[k+1], \mathbf{Q}[k], \mathbf{B}[k], \mathbf{C}[k])$

$\mathbf{Q}[k+1] = \mathbf{Q}[k] - \frac{1}{\mu_Q[k]}\Big[\mathbf{G}[k]\mathbf{L}[k+1] + \lambda_* \mathbf{R}_Q^{-1}\mathbf{Q}[k]\Big]$

[S3] Update B

$\mathbf{H}[k] = \mathbf{R}'\mathbf{\Phi}_y(\mathbf{L}[k+1], \mathbf{Q}[k+1], \mathbf{B}[k], \mathbf{C}[k]) + \mathbf{\Phi}_z(\mathbf{L}[k+1], \mathbf{Q}[k+1], \mathbf{B}[k], \mathbf{C}[k])$

$\mathbf{B}[k+1] = \mathbf{B}[k] - \frac{1}{\mu_B[k]}\Big[\mathbf{C}[k] \odot \mathbf{H}[k] + \lambda_1 \mathrm{unvec}\big(\mathbf{R}_{\mathrm{B}}^{-1}\mathrm{vec}(\mathbf{B}[\mathrm{k}])\big)\Big]$

[S4] Update C

$\mathbf{E}[k] = \mathbf{R}'\mathbf{\Phi}_y(\mathbf{L}[k+1], \mathbf{Q}[k+1], \mathbf{B}[k+1], \mathbf{C}[k]) + \mathbf{\Phi}_y(\mathbf{L}[k+1], \mathbf{Q}[k+1], \mathbf{B}[k+1], \mathbf{C}[k])$

$\mathbf{C}[k+1] = \mathbf{C}[k] - \frac{1}{\mu_C[k]}\Big[\mathbf{B}[k] \odot \mathbf{E}[k] + \lambda_1 \mathrm{unvec}\big(\mathbf{R}_{\mathrm{C}}^{-1}\mathrm{vec}(\mathbf{C}[\mathrm{k}])\big)\Big]$

$k \leftarrow k + 1$

end while

return $(\mathbf{A}[k] = \mathbf{B}[k] \odot \mathbf{C}[k], \mathbf{X}[k] = \mathbf{L}[k]\mathbf{Q}'[k])$

3.8 Practical Considerations

Before assessing their relevance to large-scale networks, the proposed algorithms must address additional practical issues. Those relate to the fact that network data are typically decentralized, streaming, subject to outliers as well as misses, and the routing matrix may be either unknown or dynamically changing over time. This section sheds light on solutions to cope with such practical challenges.

3.8.1 Inconsistent partial measurements

Certain network links may not be easily accessible to collect measurements, or, their measurements might be lost during the communication process due to e.g., packet drops. Let

Π_y collect the available link measurements during the time horizon $\mathcal{T}$. In addition, certain link or flow counts may not be consistent with the adopted model in (3.2) and (3.4). To account for possible presence of outliers introduce the matrices $\mathbf{O}_y \in \mathbb{R}^{L\times T}$ and $\mathbf{O}_z \in \mathbb{R}^{F\times T}$, which are nonzero at the positions associated with the outlying measurements, and zero elsewhere. The link-count model (3.2) should then be modified to

$$\mathbf{Y}_{\Pi_y} = \mathcal{P}_{\Pi_y}(\mathbf{R}(\mathbf{X}+\mathbf{A}) + \mathbf{O}_y + \mathbf{V}),$$

and the flow counts to

$$\mathbf{Z}_{\Pi} = \mathcal{P}_{\Pi}(\mathbf{X}+\mathbf{A}+\mathbf{O}_z+\mathbf{W})$$

Typically the outliers constitute a small fraction of measurements, thus rendering $\{\mathbf{O}_y, \mathbf{O}_z\}$ sparse. The optimization task (P1) can then be modified to take into account the misses and outliers as follows

$$\text{(P6)}\ (\hat{\mathbf{X}},\hat{\mathbf{A}}) = \arg\min_{\{\mathbf{X},\mathbf{A},\mathbf{O}_y,\mathbf{O}_z\}} \frac{1}{2}\|\mathcal{P}_{\Pi_y}(\mathbf{Y}-\mathbf{R}(\mathbf{X}+\mathbf{A})-\mathbf{O}_y)\|_F^2 + \frac{1}{2}\|\mathcal{P}_{\Pi}(\mathbf{Z}-\mathbf{X}-\mathbf{A}-\mathbf{O}_z)\|_F^2 + \lambda_*\|\mathbf{X}\|_* + \lambda_1\|\mathbf{A}\|_1 + \lambda_y\|\mathbf{O}_y\|_1 + \lambda_z\|\mathbf{O}_z\|_1$$

where λ_y and λ_z control the density of link- and flow-level outliers, respectively. Again, one can employ ADMM-type algorithms to solve (P6).

Routing information may not also be revealed in certain applications due to e.g., privacy reasons. In this case, each network link can potentially carry an unknown fraction of every OD flow. Let $\mathcal{L}_{\text{in}}(n)$ and $\mathcal{L}_{\text{out}}(n)$ denote the set of incoming and outgoing links to node $n \in \mathcal{N}$. The routing variables then must respect the flow conservation constraints, that is formally $\mathbf{R} \in \mathcal{R} := \{\mathbf{R} \in [0,1]^{L\times F} \ : \ \sum_{\ell\in\mathcal{L}_{\text{in}}(n)} r_{\ell,f} = \sum_{\ell\in\mathcal{L}_{\text{out}}(n)} r_{\ell,f},\ \forall f \in \mathcal{F}, n \in \mathcal{N}\}$. Taking the unknown routing variables into account, the optimization task to estimate the traffic is formulated as

$$\text{(P7)}\ \ (\hat{\mathbf{X}},\hat{\mathbf{A}}) = \arg\min_{\{\mathbf{X},\mathbf{A},\mathbf{R}\in\mathcal{R}\}} \frac{1}{2}\|\mathbf{Y}-\mathbf{R}(\mathbf{X}+\mathbf{A})\|_F^2 + \frac{1}{2}\|\mathcal{P}_{\Pi}(\mathbf{Z}-\mathbf{X}-\mathbf{A})\|_F^2 + \lambda_*\|\mathbf{X}\|_* + \lambda_1\|\mathbf{A}\|_1$$

which is nonconvex due to the presence of bilinear terms in the LS cost.

3.8.2 Real-time operation

Monitoring of large-scale IP networks necessitates collecting massive amounts 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 challenges estimating traffic and anomalies. In dynamic networks routing tables are constantly readjusted to effect traffic load balancing and avoid congestion caused by e.g., traffic congestion anomalies or network infrastructure failures. On top of the previous arguments, in practice the measurements are acquired sequentially across time, which motivates updating previously obtained estimates rather than recomputing new ones from scratch each time a new datum becomes available.

To account for routing changes, let $\mathbf{R}_t \in \mathbb{R}^{L\times F}$ denote the routing matrix at time t. The observed link counts at time instant t then adhere to $\mathbf{y}_t = \mathbf{R}_t(\mathbf{x}_t + \mathbf{a}_t) + \mathbf{v}_t,\ t = 1, 2, \ldots,$ where $\mathbf{y}_t \in \mathbb{R}^L$, and the partial flow counts at time t obey

$$\mathbf{z}_{\Pi_t} = \mathcal{P}_{\Pi_t}(\mathbf{x}_t + \mathbf{a}_t + \mathbf{w}_t),\ t = 1, 2, \ldots,$$

where $\mathbf{z}_{\Pi_t} \in \mathbb{R}^F$, and Π_t indexes the OD flows measured at time t. In order to estimate the nominal and anomalous traffic components $(\mathbf{x}_t, \mathbf{a}_t)$ at time instant t in real time, given only the past observations $\{\mathbf{y}_\tau, \mathbf{z}_{\Pi_\tau}\}_{\tau=1}^t$, the framework developed in our companion paper [96] can be adopted. Building on the fact that the traffic traces $\{\mathbf{x}_t\}_{t=1}^\infty$ lie in a low-dimensional linear subspace, say $\mathcal{L}$, one can postulate $\mathbf{x}_t = \mathbf{L}\mathbf{q}_t$ for $\mathbf{L} \in \mathbb{R}^{F\times\rho}$ with $\rho \ll F$, where $\mathbf{L}$ spans the subspace $\mathcal{L}$. Pursuing the ideas in [96], the nuclear-norm characterization in (3.18), which enjoys separability across time, can be applied to formulate exponentially-weighted LS estimators. The corresponding optimization task can then be solved via alternating minimization algorithms [96].

It is worth commenting that the companion work [96] aims primarily at identifying the anomalies $\mathbf{a}_t$ from link counts, which requires slow variations of the routing matrix to ensure $\{\mathbf{R}_t\mathbf{x}_t\}_{t=1}^\infty$ lie in a low-dimensional subspace. However, the tomography task considered in the present paper imposes no restriction on the routing matrix. Indeed, routing variability helps estimation of the nominal traffic $\mathbf{x}_t$. More precisely, suppose that $\{\mathbf{R}_t\}$ are sufficiently

distinct so as the intersection of the nullspaces $\bigcap_t \mathcal{N}_{R_t}$ has a small dimension. Consequently, it is less likely to find an alternative feasible solution $\mathbf{X}_1 := \mathbf{X}_0 + \mathbf{H}$ with $\mathbf{H} := [\mathbf{h}_1, \ldots, \mathbf{h}_F]$ and $\mathbf{h}_t \in \mathcal{N}_{R_t}$ such that $\mathbf{H} \in \Phi_{X_0}$ (cf. Section 3.4); see also Lemma 3.1. Further analysis of this intriguing phenomenon goes beyond the scope of the present paper, and will be pursued as future research.

3.8.3 Decentralized implementation

Algorithms 3 and 4 demand each network node (router) $n \in \mathcal{N}$ continuously communicate the local measurements of its incident links as well as the OD-flow counts originating at node n, to a central monitoring station. While this is typically the prevailing operational paradigm adopted in current network technologies, there are limitations associated with this architecture. Collecting all these data at the routers may lead to excessive protocol overhead, especially for large-scale networks with high acquisition rate. In addition, with the exchange of raw measurements missing data due to communication errors are inevitable. Performing the optimization in a centralized fashion raises robustness concerns as well, since the central monitoring station represents an isolated point of failure.

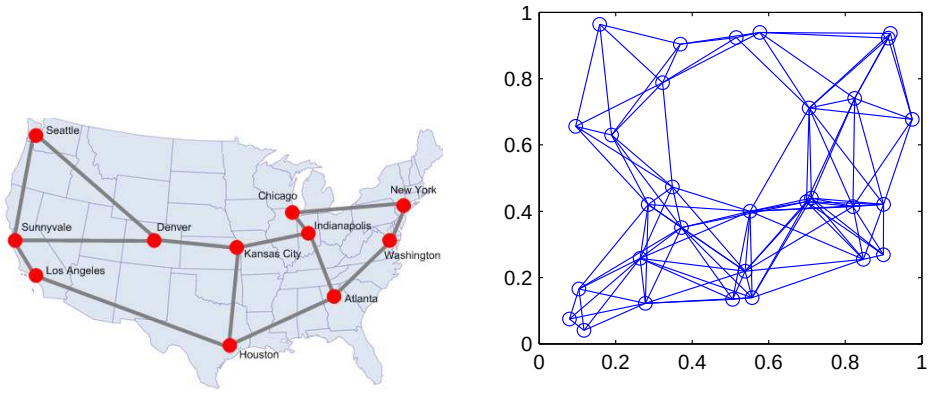
The aforementioned reasons motivate devising fully distributed iterative algorithms in large-scale networks, which allocate the network tomography functionality to the routers. In a nutshell, per iteration, nodes carry out simple computational tasks locally, relying on their own local measurements. Subsequently, local estimates are refined after exchanging messages only with directly connected neighbors, which facilitates percolation of information to the entire network. The ultimate goal is for the network nodes to consent on the global map of network-traffic-state $(\hat{\mathbf{X}}, \hat{\mathbf{A}})$, which remains close to the one obtained via the centralized counterpart with the entire network data available at once. Building on the separable characterization of the nuclear norm in (3.18), and adopting ADMM method as a basic tool to carry out distributed optimization, a generic framework for decentralized sparsity-regularized rank minimization was put forth in our companion paper [95]. In the context of network anomaly detection, the results there are encouraging and the proposed ideas can be applied to solve also (P1) in a distributed fashion.

3.9 Performance Evaluation

Performance of the novel schemes is assessed in this section via computer simulations with both synthetic and real network data as described below.

Synthetic network data. The network topology is generated according to a random geometric graph model, where the nodes are randomly placed in a unit square, and two nodes are connected with an edge if their distance is less than a prescribed threshold d_c. In general, to form the routing matrix each OD pair takes K nonoverlapping paths, each determined according to the minimum hop-count algorithm. After finding the routes, links carrying no traffic are discarded. Clearly, the number of links varies according to d_c. The underlying traffic matrix $\mathbf{X}_0$ follows the bilinear model $\mathbf{X}_0 = \mathbf{L}\mathbf{Q}'$, with the factors $\mathbf{L} \in \mathbb{R}^{F\times\rho}$ and $\mathbf{Q} \in \mathbb{R}^{T\times\rho}$ having i.i.d. Gaussian entries $\mathcal{N}(0, 1/F)$ and $\mathcal{N}(0, 1/T)$, respectively. Entries of the anomaly matrix $\mathbf{A}_0$ are also randomly drawn from the set $\{-1, 0, 1\}$ with probability (w.p.) $\Pr(a_{f,t} = -1) = \Pr(a_{f,t} = 1) = p/2$, and $\Pr(a_{f,t} = 0) = 1 - p$. The link loads are then formed as $\mathbf{Y} = \mathbf{R}(\mathbf{X}_0 + \mathbf{A}_0)$. A subset of OD flows is also sampled uniformly at random to form the partial OD flow-level measurements $\mathbf{Z}_\Pi = \mathbf{\Pi} \odot (\mathbf{X}_0 + \mathbf{A}_0)$, where each entry of $\mathbf{\Pi} \in \{0, 1\}^{F\times T}$ is i.i.d. Bernoulli distributed taking value one w.p. π, and zero w.p. $1 - \pi$.

Real network data. Real data including OD flow traffic levels are collected from the operation of the Internet-2 network (Internet backbone network across USA) [2], shown in Fig. 3.2 (a). Internet-2 comprises $N = 11$ nodes, $L = 41$ links, and $F = 121$ OD flows. Flow traffic levels are recorded every five-minute interval, for a three-week operational period during December 8-28, 2003 [2,73]. The collected flow levels are the aggregation of clean and anomalous traffic components, that is sum of unknown "ground-truth" low-rank and sparse matrices $\mathbf{X}_0 + \mathbf{A}_0$. The "ground truth" components are then discerned from their aggregate after applying robust PCP algorithms developed e.g., in [25]. The recovered $\mathbf{X}_0$ exhibits three dominant singular values, confirming the low-rank property of the nominal traffic matrix. Also, after retaining only the significant spikes with magnitude larger than the threshold $50\|\mathbf{Y}\|_F/LT$, the formed anomaly matrix $\mathbf{A}_0$ has 1.10% nonzero entries. The link loads in $\mathbf{Y}$ are obtained through multiplication of the aggregate traffic with the Internet-2



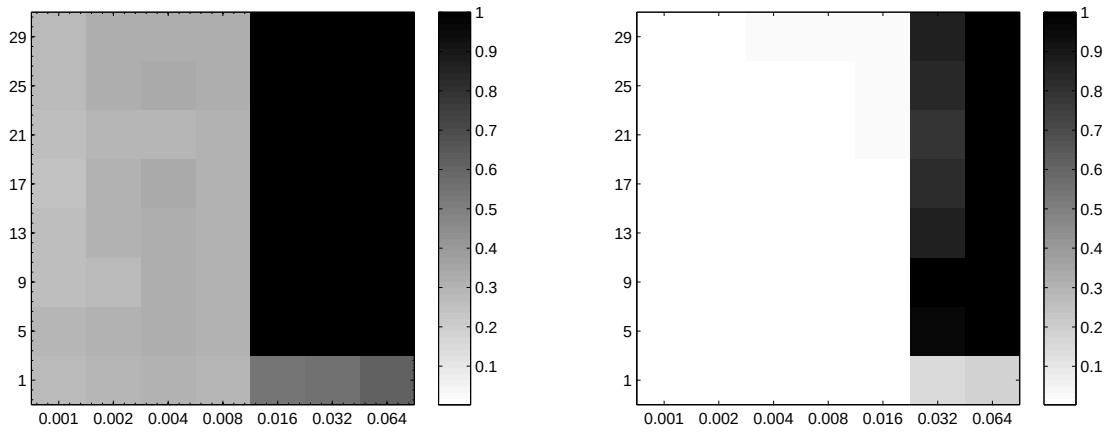
(a) (b)

Figure 3.2: Network topology graphs. (a) Internet-2. (b) Random synthetic network with $N = 30$ and $d_c = 0.35$.

routing matrix. Even though $\mathbf{Y}$ is "constructed" here from flow measurements, link loads are acquired from SNMP traces [137]. Moreover, the aggregate flow traffic matrix $\mathbf{X}_0 + \mathbf{A}_0$ is sampled uniformly at random with probability π to form $\mathbf{Z}_\Pi$. In practice, these samples are acquired via NetFlow protocol [162].

3.9.1 Exact recovery validation

To demonstrate the merits of (P2) in accurately recovering the true values $(\mathbf{X}_0, \mathbf{A}_0)$, it is solved for a wide range of rank r and (average) sparsity levels $s = pFT$ using the ADMM solver in Algorithm 3. Synthetic data is generated as described before for a random network with $N = 30$, $d_c = 0.35$, and $F = T = N(N-1)/3$; see Fig. 3.2(b). For F randomly selected OD pairs, K nonoverlapping paths are chosen to carry the traffic. Each path is created based on the minimum-hop count routing algorithm to form the routing matrix. A random fraction of the origin's traffic is also assigned to each path. The gray-scale plots in Fig. 3.3 show phase transition for the relative estimation error $e_{x+a} = e_x + e_a$, including both nominal $e_x := \|\hat{\mathbf{X}} - \mathbf{X}_0\|_F / \|\mathbf{X}_0\|_F$, and anomalous traffic estimation error $e_a := \|\hat{\mathbf{A}} - \mathbf{A}_0\|_F / \|\mathbf{A}_0\|_F$ under various percentage of misses. Top figure is associated with $K = 1$, while for the bottom figure $K = 3$. The parameter λ in (P2) is also tuned to optimize the performance.



(a) (b)

Figure 3.3: Relative estimation error e_{x+a} for various values of rank (r) and sparsity level ($s = pFT$) where $F = T = 290$ and $\pi = 0.25$. (a) Single-path routing versus (b) multipath routing ($K = 3$). White represents exact recovery ($e_{x+a} \approx 0$), while black represents $e_{x+a} \approx 1$.

When single-path routing is used, the network entails $L = 159$ physical links. In this case, the routing matrix $\mathbf{R} \in \{0, 1\}^{159 \times 290}$ has a huge nullspace with $\dim(\mathcal{N}_R) = 127$, and as a result Fig. 3.3 (top) indicates that accurate recovery is possible only for relatively small values of r and s. However, when multipath routing ($K = 3$) is used, there are more $L = 227$ physical links involved in carrying the traffic of OD flows. This shrinks the nullspace of $\mathbf{R} \in [0, 1]^{227 \times 290}$ to $\dim(\mathbf{R}) = 68$, and improves the isometry property of $\mathbf{R}$ for sparse vectors. As a result, under traffic of higher dimensionality and denser anomalies accurate traffic estimation is possible; see Fig. 3.3 (bottom).

3.9.2 Traffic and anomaly maps

Real Internet-2 data is considered to portray the traffic based on (P1) every 42-hour interval, which amounts to time horizon of $T = 504$ time bins.

Convergence of Algorithm 3. The iterative ADMM solver is run with various percentages of NetFlow data sampled uniformly at random when $c = 1$ and with a fixed value of λ. Evolution of the cost in (P3) is depicted in Fig. 3.4 across iterations and runtime. Note the

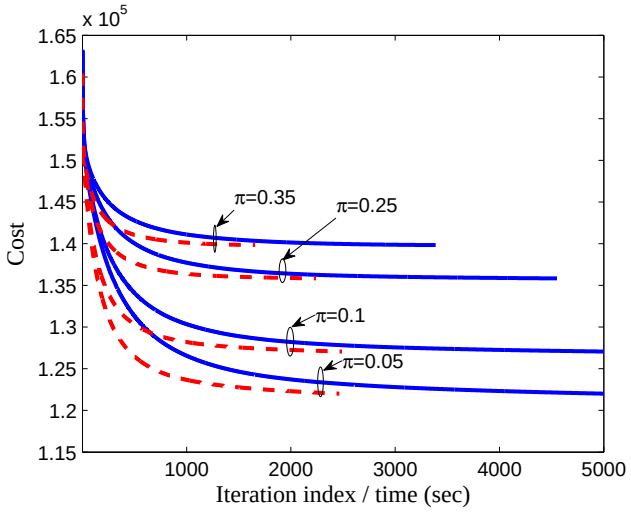


Figure 3.4: Cost of (P2) versus the iteration index (solid) and run-time (dashed) for various NetFlow sampling rates.

Matlab codes for this purpose are by no means optimized and the updates for [S2] and [S3] are not implemented in parallel; doing so can lead to a significant runtime advantage. It is apparent that Algorithm 3 as claimed by Proposition 3.1 converges after about a dozen hundreds of iterations. In addition, convergence becomes faster for larger values of π since it improves the conditioning of the augmented Lagrangian (3.13); that is, the quadratic subproblems w.r.t. $\mathbf{B}$ and $\mathbf{O}$ admit a larger minimum-eigenvalue for the Hessian matrix.

Impact of NetFlow data. The role of NetFlow measurements on the traffic estimation performance is depicted in Fig. 3.5 plotting the relative error e_{x+a} for various percentages of NetFlow samples (π). Normally, the estimation accuracy improves as π grows, where the improvement seems more pronounced for the nominal traffic. When only the link loads are available, adding 10% NetFlow samples enhances the nominal-traffic estimation accuracy by 45%, while the one for the anomalous traffic is improved by 18%. This observation corroborates the effectiveness of exploiting partial NetFlow samples toward mapping out the network traffic.

Traffic profiles. For $\pi = 0.1$, the true and estimated traffic time-series are illustrated in Fig. 3.6 for three representative OD flows originating from the CHIN autonomous system located at Chicago. The depicted time-series correspond to three different rows of $\hat{\mathbf{X}}$ and $\hat{\mathbf{A}}$

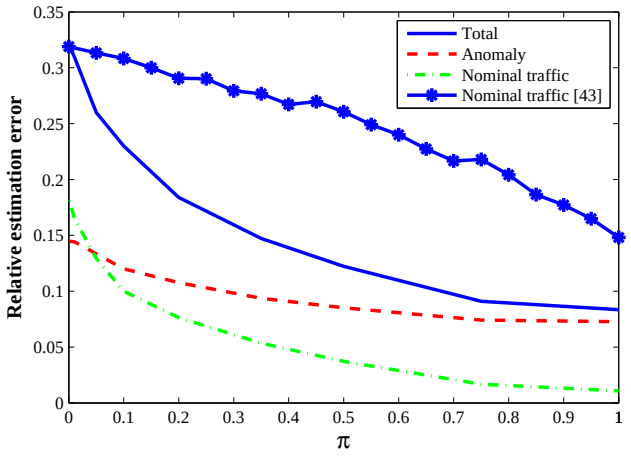
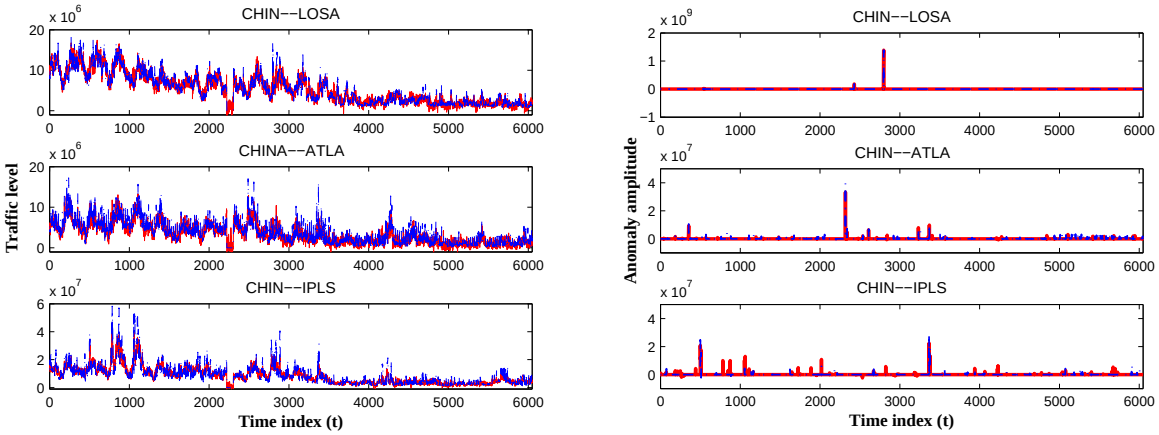


Figure 3.5: Relative Estimation error versus percentage of NetFlow samples.

returned by (P1). It is apparent that the traffic variations are closely tracked and significant spikes are correctly picked by (P1). It pinpoints confidently a significant anomaly occurring within 9:20 P.M.–9:25 P.M., December 11, 2003, in the flow CHIN–LOSA, which traverses several physical links. High false alarm declared for the CHIN–IPLS flow is also because it visits only a single link, and thus not revealing enough information.

Unveiling anomalies. Identifying anomalous patterns is pivotal towards proactive network security tasks. The resultant estimated map $\hat{\mathbf{A}}$ returned by (P1) offers a depiction of the network health-state across both time and flows. Our previous work in [97] and [96] deals with creating such a map with only the link loads $\mathbf{Y}$ at hand (i.e., $\Pi = \emptyset$), and the primary goal is to recover $\hat{\mathbf{A}}$. The purported results in [96, 97] are promising and could markedly outperform state-of-art workhorse PCA-based approaches in e.g., [72, 158]. Relative to [96, 97], the current work however allows additional partial flow-level measurements. This naturally raises the question how effective this additional information is toward identifying the anomalies. As seen in Fig. 3.5, taking more NetFlow samples is useful, but beyond a certain threshold it does not offer any extra appeal.

Schemes for comparison. Despite its importance, a fair and comprehensive comparison with the existing alternatives in e.g., [67, 120, 159, 162] is subtle mainly because they either adopt different assumptions, or, utilize different data sources. It is also important to stress



(a) (b)

Figure 3.6: Nominal (a) and anomalous (b) traffic portrays for three representative OD flows when $\pi = 0.1$. True traffic is dashed blue and the estimated one is solid red.

once more that past works are either devoid of anomalies and/or spatiotemporal traffic correlations, that are unique to the present paper. Nonetheless, to highlight the merits of the novel schemes we compare them with the tomogravity-based scheme of [162] that leverages the same SNMP and partial NetFlow data. The crux of [162] is to simply model SNMP link counts acquired per time instant t as $\mathbf{y}_t = \mathbf{R}\mathbf{x}_t + \mathbf{v}_t$, with $\mathbf{v}_t \sim \mathcal{N}(0, \sigma^2\mathbf{I})$; and likewise the NetFlow counts of the observed OD flows as $\mathbf{z}_{f,t} = x_{f,t} + w_{f,t},\ f \in \Pi_t$, with $w_{f,t} \sim \mathcal{N}(0, \sigma^2)$. For the unobservable OD flows, the gravity model [67, 159] is employed to postulate the prior $x_{f,t} = z^{(g)}_{f,t} + n_{f,t},\ f \in \Pi_t^{\perp}$, where $z^{(g)}_{f,t}$ is output of the simple gravity method fed with the link loads. Noise $n_{f,t}$ is Gaussian $\mathcal{N}(0, \sigma^2/\kappa^2)$ with the scalar κ capturing the reliability of gravity model relative to NetFlow measurements. Putting data together, the OD-flow traffic is estimated by solving the LS program

$$\min_{\mathbf{x}} \ \|\mathbf{y}_t - \mathbf{R}\mathbf{x}\|^2 + \|\mathbf{z}_{\Pi_t} - \mathbf{x}_{\Pi_t}\|^2 + \kappa\|\mathbf{z}^{(g)}_{\Pi_t^{\perp}} - \mathbf{x}_{\Pi_t^{\perp}}\|^2 \tag{3.23}$$

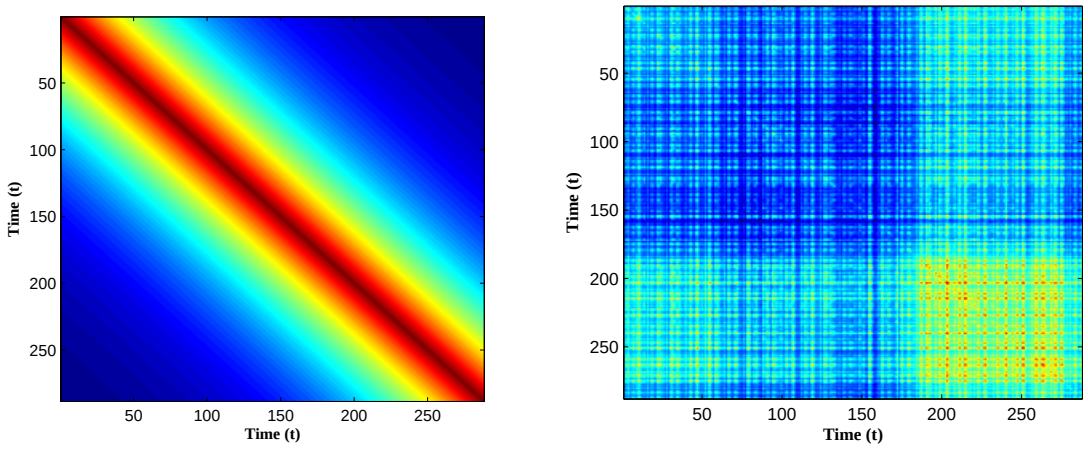
Fig. 3.5 compares the relative error of (3.23) against our novel scheme with various percentages of NetFlow data used to estimate the nominal traffic. Note the value of κ is optimized to achieve the best performance for (3.23). The large error gap is due to the over-simplified model in [162] that ignores the anomalies and the spatiotemporal traffic correlations.

3.9.3 Estimation with spatiotemporal correlation information

This section evaluates the effectiveness of (P5) and demonstrates the usefulness of traffic correlation information. Training data from the week December 8-15, 2003 are used to estimate the Internet-2 traffic on the next day, December 16, 2003. The nominal "ground truth" traffic matrix $\mathbf{X}_0$ described earlier is considered, and for validation purposes bursty anomalies are synthetically injected to form the aggregate traffic $\mathbf{X}_0 + \mathbf{A}_0$, which is then used to generate $\mathbf{Y}$ and $\mathbf{Z}_\Pi$. To simulate the NetFlow samples, suppose 10% of randomly selected OD flows are inaccessible for the entire time horizon, and the rest are sampled only 10% of time, resulting in 9% flow-level measurements available.

Bursty anomalies. To generate anomalies $\mathbf{X}_0$, envision a scenario where a subset of OD flows undergo bursty anomalies while the rest are clean. Per flow f bursty anomalies are generated according to the random multiplicative process $\{a_{f,t} = \gamma_f b_{f,t} c_{f,t}\}_t$, with mutually independent stationary processes $\{c_{f,t}\}$ and $\{b_{f,t}\}$. The former is a correlated Gaussian process, and the latter is a correlated $\{0,1\}$-Bernoulli process to model the bursts. The Gaussian process obeys the first-order auto-regressive model $c_{f,t} = \theta c_{f,t-1} + \sigma_n n_{f,t}$, with $c_{f,0} = 0$ and $n_{f,t} \sim \mathcal{N}(0,1)$ for some $\theta < 1$. The Bernoulli process also adheres to $b_{f,t} = d_{f,t} b_{f,t-1} + (1 - d_{f,t}) e_{f,t}$, where the independent random variables $d_{f,t}$ and $e_{f,t}$ obey $d_{f,t} \sim \text{Ber}(\alpha)$ and $e_{f,t} \sim \text{Ber}(\nu)$, respectively. Initial variable $b_{f,0}$ is also generated as $\text{Ber}(\nu)$.

Learning correlations. Owing to the stationarity of processes $\{b_{f,t}\}$ and $\{c_{f,t}\}$, process $\{a_{f,t}\}$ is stationary, and as a result $R_a^{(f)}(\tau) = \gamma_f^2 R_b^{(f)}(\tau) R_c^{(f)}(\tau)$, with the corresponding correlations given as $R_c^{(f)}(\tau) = \theta^\tau \sigma_n^2/(1-\theta^2)$ and $R_b^{(f)}(\tau) = \nu(1-\nu)\alpha^\tau + \nu$. Set $\gamma_f = 50$, $\theta = 0.999$, $\sigma_n = 0.005$, $\alpha = 0.98$, and $\nu = 0.03$. The correlation matrices $\{\mathbf{R}_B, \mathbf{R}_C\}$ with Toeplitz blocks are then obtained from (3.20). Moreover, to account for the cyclostationarity of traffic with a day-long periodicity, the correlation matrices $\{\mathbf{R}_L, \mathbf{R}_Q\}$ are learned as elaborated in Section 3.6.1. The resulting temporal correlation matrices $\mathbf{R}_B$ and $\mathbf{R}_Q$, learned based on the traffic data December 8-15, 2003, are displayed in Fig. 3.7, where 288 data points in each axis correspond to 24 hours. The sharp transition noticed in Fig. 3.7 (b) happens at $3:45$ p.m. that signifies a sudden increase in the traffic usage for the rest



(a) (b)

Figure 3.7: Sample correlations $\mathbf{R}_B$ (a) and $\mathbf{R}_Q$ (b) learned based on historical traffic data during December 8-15, 2003.

of the day.

Traffic maps. Fig. 3.9 depicts the time series of estimated and true nominal traffic for the IPLS–CHIN OD flow (see Fig. 3.2). For this flow, no direct NetFlow sample is collected. It is apparent that (P5) which uses the knowledge of traffic spatiotemporal correlation tracks fairly well the underlying traffic, whereas (P1) cannot even track the large-scale variations of traffic. This demonstrates the nonidentifiability of (P1) when only a small fraction 9% of OD flows are nonuniformly sampled, and notably around 10% of rows of $\mathbf{X}_0$ are not directly observable. (P5) however interpolates the traffic associated with unobserved OD flows with the observed ones through the correlation matrices $\{\mathbf{R}_L, \mathbf{R}_Q\}$. The resulting relative estimation error for (P5) is $e_x = 0.19$, which is well below $e_x = 0.62$ for (P1). The correlation knowledge also helps discovering the anomalous traffic patterns as seen from Fig. 3.8, where in particular (P5) attains $e_a = 0.27$, while (P1) does $e_a = 0.73$. Interestingly, the anomaly map revealed by (P1) tends to spot the anomalies intermittently since the ℓ_1-norm regularizer weighs all flows and time-instants equally.

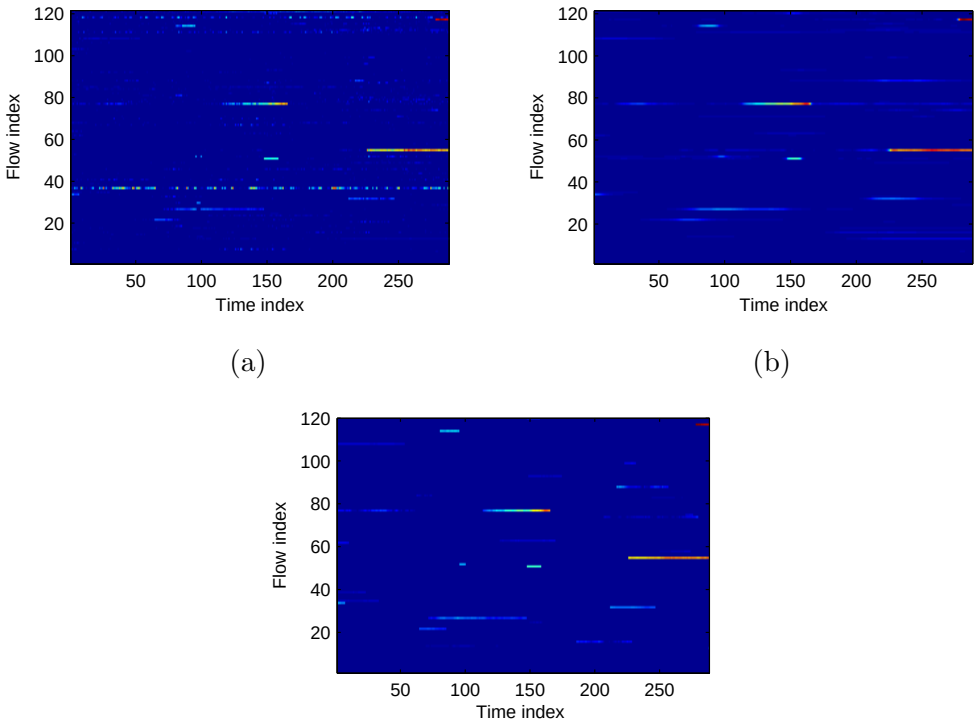


Figure 3.8: Estimated and "ground truth" (c) anomaly maps across time and flows without using correlation (a), and after using correlation information (b).

3.10 Conclusions and Future Work

This paper taps on recent advances in low-rank and sparse recovery to create maps of nonminal and anomalous traffic as a valuable input for network management and proactive security tasks. A novel tomographic framework is put forth which subsumes critical network monitoring tasks including traffic estimation, anomaly identification, and traffic interpolation. Leveraging low intrinsic-dimensionality of nominal traffic as well as the sparsity of anomalies, a convex program is formulated with ℓ_1- and nuclear-norm regularizers, with the link loads and a small subsets of flow counts as the available data. Under certain circumstances on the true traffic and anomalies in addition to the routing and OD-flow sampling strategies, sufficient conditions are derived, which guarantee accurate estimation

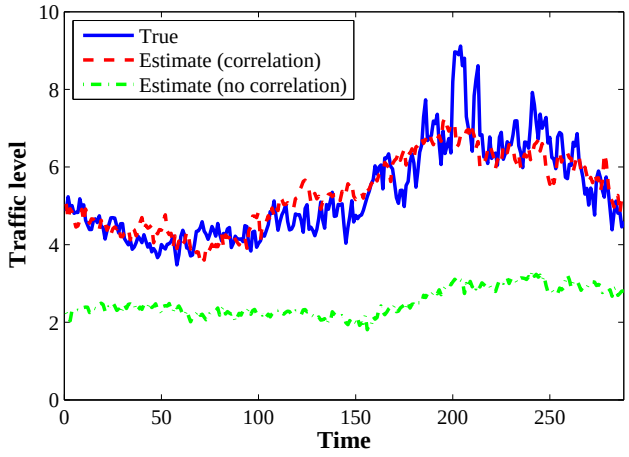


Figure 3.9: True and estimated traffic of IPLS-CHIN flow.

of the traffic.

For practical networks where the said conditions are possibly violated, additional knowledge about inherent traffic patterns are incorporated through correlations by adopting a Bayesian approach and taking advantage of the bilinear characterization of the ℓ_1- and nuclear-norm. A systematic approach is also devised to learn the correlations using (cyclo)stationary historical traffic data. Simulated tests with synthetic and real Internet data confirm the efficacy of the novel estimators. There are yet intriguing unanswered questions that go beyond the scope of the current paper, but worth pursuing as future research. One such question pertains to comparing performance of novel schemes against existing alternatives over different datasets and under fair conditions. It is also important to quantify a minimal count of sampled OD flows for a realistic network scenario with a given routing matrix, which assures accurate traffic estimation. Another avenue to explore involves adoption of tensor models along the lines of [13,66,98] to further exploit the network topological information toward improving the traffic estimation accuracy.

Chapter 4

Decentralized Rank Minimization and Sparsity Regularization

4.1 Introduction

Let $\mathbf{X} := [x_{l,t}] \in \mathbb{R}^{L\times T}$ be a *low-rank* matrix [$\text{rank}(\mathbf{X}) \ll \min(L,T)$], and $\mathbf{A} := [a_{f,t}] \in \mathbb{R}^{F\times T}$ be a *sparse* matrix with support size considerably smaller than FT. Consider also a matrix $\mathbf{R} := [r_{l,f}] \in \mathbb{R}^{L\times F}$ and a set $\Omega \subseteq \{1,\ldots,L\} \times \{1,\ldots,T\}$ of index pairs (l,t) that define a sampling of the entries of $\mathbf{X}$. Given $\mathbf{R}$ and a number of (possibly) noise corrupted measurements[1]

$$y_{l,t} = x_{l,t} + \sum_{f=1}^{F} r_{l,f} a_{f,t} + v_{l,t}, \quad (l,t) \in \Omega \tag{4.1}$$

the goal is to estimate low-rank $\mathbf{X}$ and sparse $\mathbf{A}$, by denoising the observed entries and imputing the missing ones. Introducing the sampling operator $\mathcal{P}_\Omega(\cdot)$ which sets the entries of its matrix argument not in Ω to zero and leaves the rest unchanged, the data model can be compactly written in matrix form as

$$\mathcal{P}_\Omega(\mathbf{Y}) = \mathcal{P}_\Omega(\mathbf{X} + \mathbf{R}\mathbf{A} + \mathbf{V}). \tag{4.2}$$

[1]The notation adopted here is motivated by the anomaly detection problem outlined in in the previous chapters, where $\mathbf{R}$ denotes the routing matrix, F stands for flows, L for links and T for time steps, while $\mathbf{A}$ is a matrix of anomalies.

A natural estimator accounting for the low rank of $\mathbf{X}$ and the sparsity of $\mathbf{A}$ will be sought to fit the data $\mathcal{P}_\Omega(\mathbf{Y})$ in the least-squares (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; see e.g. [26], [97], [25], [33] for related problems subsumed by the one described here. Unfortunately, both rank and ℓ_0-norm minimization are in general NP-hard problems [37, 109]. The nuclear norm $\|\mathbf{X}\|_* := \sum_k \sigma_k(\mathbf{X})$, where $\sigma_k(\mathbf{X})$ denotes the k-th singular value of $\mathbf{X}$, and the ℓ_1-norm $\|\mathbf{A}\|_1 := \sum_{f,t} |a_{f,t}|$, are typically adopted as surrogates to rank($\mathbf{X}$) and $\|\mathbf{A}\|_0$, respectively [29, 48]. Accordingly, one solves

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

where $\lambda_*, \lambda_1 \geq 0$ are rank- and sparsity-controlling parameters. Being convex (P1) is appealing, and some of its special instances are known to attain good performance in theory and practice. For instance, when no data are missing (P1) can be used to unveil traffic anomalies in networks [97]. Identifiability results in [97] establish that $\mathbf{X}$ and $\mathbf{A}$ can be exactly recovered in the absence of noise, even when $\mathbf{R}$ is a fat (compression) operator. When $\mathbf{R}$ equals the identity matrix, (P1) reduces to the so-termed robust principal component analysis (PCA), for which exact recovery results are available in [25] and [33]. Moreover, for the special case $\mathbf{R} \equiv \mathbf{0}_{L\times F}$, (P1) offers a low-rank matrix completion alternative with well-documented merits; see e.g., [27] and [26]. Stable recovery results in the presence of noise are also available for matrix completion and robust PCA [26, 163]. Earlier efforts dealing with the recovery of sparse vectors in noise led to similar performance guarantees; see e.g., [19].

In all these works, the samples $\mathcal{P}_\Omega(\mathbf{Y})$ and matrix $\mathbf{R}$ are assumed centrally available, so that they can be jointly processed to estimate $\mathbf{X}$ and $\mathbf{A}$ by e.g., solving (P1). Collecting all this information can be challenging in various applications of interest, or it may be even impossible in e.g., wireless sensor networks (WSNs) operating under stringent power budget constraints. In other cases such as the Internet or collaborative marketing studies, agents providing private data for e.g., fitting a low-rank preference model, may not be willing to share their training data but only the learning results. Performing the optimization in a centralized fashion raises robustness concerns as well, since the central processor represents

an isolated point of failure. Several customized iterative algorithms have been proposed to solve instances of (P1), and have been shown effective in tackling low- to medium-size problems; see e.g., [97], [27], [117]. However, most algorithms require computation of singular values per iteration and become prohibitively expensive when dealing with high-dimensional data [118]. All in all, the aforementioned reasons motivate the reduced-complexity *decentralized* algorithm for nuclear and ℓ_1-norm minimization developed in this paper.

In a similar vein, stochastic gradient algorithms were recently developed for large-scale problems entailing regularization with the nuclear norm [96, 118]. Even though iterations in [118] are highly paralellizable, they are not applicable to networks of arbitrary topology. There are also several studies on decentralized estimation of sparse signals via ℓ_1-norm regularized regression; see e.g., [38, 63, 105]. Different from the treatment here, the data model of [105] is devoid of a low-rank component and all the observations $\mathbf{Y}$ are assumed available (but decentralized across several interconnected agents). Formally, the model therein is a special case of (4.2) with $T = 1$, $\mathbf{X} = \mathbf{0}_{L\times T}$, and $\Omega = \{1, \ldots, L\} \times \{1, \ldots, T\}$, in which case (P1) boils down to finding the least-absolute shrinkage and selection operator (Lasso) [19].

Building on the general model (4.2) and the centralized estimator (P1), this paper develops decentralized algorithms to estimate low-rank and sparse matrices, based on in-network processing of a small subset of noise-corrupted and spatially-decentralized measurements (Section 4.3). This is a challenging task however, since the non-separable nuclear-norm present in (P1) is not amenable to decentralized minimization. To overcome this limitation, results from [23] and [134] on alternative characterizations of the nuclear norm are leveraged in Section 4.3.1, to obtain for the first time a separable yet non-convex cost that can be minimized in a decentralized fashion via the alternating-direction method of multipliers (ADMM) [18]. The resultant iterations entail reduced-complexity optimization subtasks per agent, and affordable message passing only between single-hop neighbors (Section 4.3.3). Interestingly, the decentralized (non-convex) estimator provably attains *the global* optimum of its centralized counterpart (P1), provided it converges and a qualification condition is satisfied; see also [4, 23, 117] for related results in the context of centralized smooth opti-

mization.

In a nutshell, this work connects the exact and stable recovery in e.g., [25, 26, 33, 97] to in-network minimization, so that one can recover (in a stable manner) the unknown low-rank and sparse matrices only through local computations and message exchanges. To demonstrate the generality of the proposed estimator and its algorithmic framework, three networking-related application domains are outlined in Section 4.4, namely: i) unveiling traffic volume anomalies for large-scale networks [72, 97]; ii) robust PCA [25, 33], and iii) low-rank matrix completion for network-wide path latency prediction [79]. Numerical tests with synthetic and real network data drawn from these application domains corroborate the effectiveness and convergence of the novel decentralized algorithms, as well as their centralized performance benchmarks (Section 4.5).

Section 4.6 concludes the paper, while several technical details are deferred to the Appendix.

4.2 Preliminaries and Problem Statement

Consider N networked agents capable of performing some local computations, as well as exchanging messages among directly connected neighbors. An agent should be understood as an abstract entity, e.g., a sensor in a WSN, or a router monitoring Internet traffic. The network is modeled as an undirected graph $G(\mathcal{N}, \mathcal{L})$, where the set of nodes $\mathcal{N} := \{1, \ldots, N\}$ corresponds to the network agents, and the edges (links) in $\mathcal{L} := \{1, \ldots, L\}$ represent pairs of agents that can communicate. Agent $n \in \mathcal{N}$ communicates with its single-hop neighboring peers in $\mathcal{J}_n$, and the size of the neighborhood will be henceforth denoted by $|\mathcal{J}_n|$. To ensure that the data from an arbitrary agent can eventually percolate through the entire network, it is assumed that:

(a1) *Graph G is connected; i.e., there exists a (possibly) multi-hop path connecting any two agents.*

With reference to the low-rank and sparse matrix recovery problem outlined in Section 4.1, in the network setting envisioned here each agent $n \in \mathcal{N}$ acquires a few in-

complete and noise-corrupted rows of matrix $\mathbf{Y} \in \mathbb{R}^{L\times T}$. Specifically, the local data available to agent n is matrix $\mathcal{P}_{\Omega_n}(\mathbf{Y}_n)$, where $\mathbf{Y}_n \in \mathbb{R}^{L_n\times T}$, $\sum_{n=1}^{N} L_n = L$, and $\mathbf{Y} := [\mathbf{Y}_1', \ldots, \mathbf{Y}_N']' = \mathbf{X}+\mathbf{R}\mathbf{A}+\mathbf{V}$. The index pairs in Ω_n are those in Ω for which the row index matches the rows of $\mathbf{Y}$ observed by agent n. Additionally, suppose that agent n has available the local matrix $\mathbf{R}_n \in \mathbb{R}^{L_n\times F}$, containing a row subset of $\mathbf{R}$ associated with the observed rows in $\mathbf{Y}_n$, i.e, $\mathbf{R} := [\mathbf{R}_1', \ldots, \mathbf{R}_N']'$. With regards to the decision variables, partition also $\mathbf{X} := [\mathbf{X}_1', \ldots, \mathbf{X}_N']' \in \mathbb{R}^{L\times T}$ similar to $\mathbf{R}$ and $\mathbf{Y}$, where $\mathbf{X}_n \in \mathbb{R}^{L_n\times T}$, $n = 1, \ldots, N$. Agents collaborate to form the wanted estimator (P1) in a decentralized fashion, which can be equivalently rewritten as (define $g_n(\mathbf{X}_n, \mathbf{A}) := \frac{1}{2}\|\mathcal{P}_{\Omega_n}(\mathbf{Y}_n - \mathbf{X}_n - \mathbf{R}_n\mathbf{A})\|_F^2$)

$$\min_{\{\mathbf{X},\mathbf{A}\}} \sum_{n=1}^{N} \left[g_n(\mathbf{X}_n, \mathbf{A}) + \frac{\lambda_*}{N}\|\mathbf{X}\|_* + \frac{\lambda_1}{N}\|\mathbf{A}\|_1 \right].$$

The objective of this paper is to develop a decentralized algorithm for sparsity-regularized rank minimization via (P1), based on in-network processing of the locally available data. The described setup naturally suggests three characteristics that the algorithm should exhibit: c1) agent $n \in \mathcal{N}$ should obtain an estimate of $\mathbf{X}_n$ and $\mathbf{A}$, which coincides with the corresponding solution of the centralized estimator (P1) that uses the entire data $\mathcal{P}_\Omega(\mathbf{Y})$; c2) processing per agent should be kept as simple as possible; and c3) the overhead for inter-agent communications should be affordable and confined to single-hop neighborhoods.

4.3 Distributed Algorithm for In-Network Operation

To facilitate reducing the computational complexity and memory storage requirements of the decentralized algorithm sought, it is henceforth assumed that:

(a2) *The decision variable* $\mathbf{X}$ *in (P1) has rank at most* ρ.

Analysis with real Internet traffic data reveals that origin-to-destination flow traffic matrices have rank$[\mathbf{X}] \in [5, 8]$; hence, one can safely choose $\rho = 10$ [72]. In addition, recall that the rank of the solution $\hat{\mathbf{X}}$ in (P1) is controlled by the choice of λ_*, and can be made small enough for sufficiently large λ_*. As argued next, the smaller the value of ρ, the more efficient the algorithm becomes.

Because $\text{rank}(\hat{\mathbf{X}}) \leq \rho$, (P1)'s search space is effectively reduced and one can factorize the decision variable as $\mathbf{X} = \mathbf{L}\mathbf{Q}'$, where $\mathbf{L}$ and $\mathbf{Q}$ are $L \times \rho$ and $T \times \rho$ matrices, respectively. Adopting this reparametrization of $\mathbf{X}$ in (P1), and defining $r_n(\mathbf{L}_n, \mathbf{Q}, \mathbf{A}) := \frac{1}{2}\|\mathcal{P}_{\Omega_n}(\mathbf{Y}_n - \mathbf{L}_n\mathbf{Q}' - \mathbf{R}_n\mathbf{A})\|_F^2$, one obtains the following equivalent optimization problem

$$\text{(P2)} \quad \min_{\{\mathbf{L},\mathbf{Q},\mathbf{A}\}} \sum_{n=1}^{N} \left[r_n(\mathbf{L}_n, \mathbf{Q}, \mathbf{A}) + \frac{\lambda_*}{N}\|\mathbf{L}\mathbf{Q}'\|_* + \frac{\lambda_1}{N}\|\mathbf{A}\|_1 \right]$$

which is non-convex due to the bilinear terms $\mathbf{L}_n\mathbf{Q}'$, and $\mathbf{L} := [\mathbf{L}_1', \ldots, \mathbf{L}_N']'$. The number of variables is reduced from $LT + FT$ in (P1), to $\rho(L + T) + FT$ in (P2). The savings can be significant when ρ is in the order of a few dozens, and both L and T are large. The dominant FT-term in the variable count of (P2) is due to $\mathbf{A}$, which is sparse and can be efficiently handled even when both F and T are large. Problem (P2) is still not amenable to decentralized 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-agent summands.

4.3.1 A separable nuclear norm regularization

To address (i), consider the following neat characterization of the nuclear norm [117, 134]

$$\|\mathbf{X}\|_* := \min_{\{\mathbf{L},\mathbf{Q}\}} \quad \frac{1}{2}\left\{\|\mathbf{L}\|_F^2 + \|\mathbf{Q}\|_F^2\right\}, \quad \text{s. to } \mathbf{X} = \mathbf{L}\mathbf{Q}'. \tag{4.3}$$

For an arbitrary matrix $\mathbf{X}$ with SVD $\mathbf{X} = \mathbf{U}_X\mathbf{\Sigma}_X\mathbf{V}_X'$, the minimum in (4.3) is attained for $\mathbf{L} = \mathbf{U}_X\mathbf{\Sigma}_X^{1/2}$ and $\mathbf{Q} = \mathbf{V}_X\mathbf{\Sigma}_X^{1/2}$. The optimization (4.3) is over all possible bilinear factorizations of $\mathbf{X}$, so that the number of columns of $\mathbf{L}$ and $\mathbf{Q}$ is also a variable. Leveraging (4.3), the following reformulation of (P2) provides an important first step towards obtaining a decentralized estimator:

$$\text{(P3)} \quad \min_{\{\mathbf{L},\mathbf{Q},\mathbf{A}\}} \sum_{n=1}^{N} \left[r_n(\mathbf{L}_n, \mathbf{Q}, \mathbf{A}) + \frac{\lambda_*}{2N}\left\{N\|\mathbf{L}_n\|_F^2 + \|\mathbf{Q}\|_F^2\right\} + \frac{\lambda_1}{N}\|\mathbf{A}\|_1 \right].$$

Under (a2) and building on (4.3), it readily follows that the separable Frobenius-norm regularization in (P3) comes with no loss of optimality, meaning that (P1) and (P3) admit identical solutions. This equivalence ensures that by finding the global minimum of (P3) [which can have significantly fewer variables than (P1)], one can recover the optimal solution

of (P1). However, since (P3) is non-convex, it may have stationary points which need not be globally optimal. Interestingly, the next proposition offers a global optimality certificate for the stationary points of (P3). For a proof, see the Appendix.

Proposition 4.1 *Let* $\{\bar{\mathbf{L}}, \bar{\mathbf{Q}}, \bar{\mathbf{A}}\}$ *be a stationary point of (P3). If* $\|\mathcal{P}_\Omega(\mathbf{Y} - \bar{\mathbf{L}}\bar{\mathbf{Q}}' - \mathbf{R}\bar{\mathbf{A}})\| \leq \lambda_*$ *(no subscript in* $\|.\|$ *signifies spectral norm), then* $\{\hat{\mathbf{X}} = \bar{\mathbf{L}}\bar{\mathbf{Q}}', \hat{\mathbf{A}} = \bar{\mathbf{A}}\}$ *is the globally optimal solution of (P1).*

Note that the noise variance certainly affects the value of $\|\mathcal{P}_\Omega(\mathbf{Y} - \bar{\mathbf{L}}\bar{\mathbf{Q}}' - \mathbf{R}\bar{\mathbf{A}})\|$, and thus satisfaction of the qualification inequality in Proposition 1.

Remark 4.1 (Proposition 1 in context) *The ideas leading to Proposition 1 were sparked by the results of [23], which introduced the bilinear factorization* $\mathbf{X} = \mathbf{L}\mathbf{Q}'$ *as a viable alternative for rank relaxation in semidefinite programming. Noteworthy extensions include learning operators with spectral regularization [4], and rank minimization with the nuclear-norm [117]. However, relative to [4,23,117] Proposition 1 has differences and makes distinct contributions. Unlike [4] and [117] which deal with smooth cost functions, the* ℓ_1*-norm regularization promoting sparsity in* $\mathbf{A}$ *renders the cost of (P3) non-smooth. Different from [23], Proposition 1 links the* stationary points *of the non-convex (P3) with the global optima of (P1). (Instead, [23] relates* local minima *of a related non-convex problem with global optima of its convex counterpart.) This difference bears practical importance since most iterative solvers of nonconvex problems such as (P3), can at most guarantee solutions that are stationary points.*

4.3.2 Local variables and consensus constraints

To decompose the cost function in (P3), 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 agent n. These local estimates are

utilized to form the separable *constrained* minimization problem

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

$$\text{s. t.} \quad \mathbf{B}_n = \mathbf{A}_n, \quad n \in \mathcal{N}$$

$$\mathbf{Q}_n = \mathbf{Q}_m,\ \mathbf{A}_n = \mathbf{A}_m, \quad m \in \mathcal{J}_n,\ n \in \mathcal{N}.$$

For reasons that will become clear later on, additional variables $\{\mathbf{B}_n\}_{n=1}^N$ were introduced to split the ℓ_2-norm fitting-error part of the cost of (P4), from the ℓ_1-norm regularization on the $\{\mathbf{A}_n\}_{n=1}^N$ (cf. Remark 4.4). These extra variables are not needed if $\mathbf{R}'\mathbf{R} = \mathbf{I}_F$. The set of additional constraints $\mathbf{B}_n = \mathbf{A}_n$ ensures that, in this sense, nothing changes in going from (P3) to (P4). Most importantly, (P3) and (P4) are equivalent optimization problems under (a1). The equivalence should be understood in the sense that $\hat{\mathbf{Q}}_1 = \hat{\mathbf{Q}}_2 = \ldots = \hat{\mathbf{Q}}_N = \hat{\mathbf{Q}}$ and likewise for $\mathbf{A}$, where $\{\hat{\mathbf{Q}}_n, \hat{\mathbf{A}}_n\}_{n\in\mathcal{N}}$ and $\{\hat{\mathbf{Q}}, \hat{\mathbf{A}}\}$ are the optimal solutions of (P4) and (P3), respectively. Of course, the corresponding estimates of $\mathbf{L}$ will coincide as well. 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 (P3). To arrive at the desired decentralized algorithm, it is convenient to reparametrize the consensus constraints in (P4) as

$$\mathbf{Q}_n = \bar{\mathbf{F}}_n^m,\ \mathbf{Q}_m = \tilde{\mathbf{F}}_n^m,\ \text{and}\ \bar{\mathbf{F}}_n^m = \tilde{\mathbf{F}}_n^m,\ m \in \mathcal{J}_n,\ n \in \mathcal{N} \tag{4.4}$$

$$\mathbf{A}_n = \bar{\mathbf{G}}_n^m,\ \mathbf{A}_m = \tilde{\mathbf{G}}_n^m,\ \text{and}\ \bar{\mathbf{G}}_n^m = \tilde{\mathbf{G}}_n^m,\ m \in \mathcal{J}_n,\ n \in \mathcal{N} \tag{4.5}$$

where $\{\bar{\mathbf{F}}_n^m, \tilde{\mathbf{F}}_n^m, \bar{\mathbf{G}}_n^m, \tilde{\mathbf{G}}_n^m\}_{n\in\mathcal{N}}^{m\in\mathcal{J}_n}$ are auxiliary optimization variables that will be eventually eliminated.

4.3.3 The alternating-direction method of multipliers

To tackle the constrained minimization problem (P4), associate Lagrange multipliers $\mathbf{M}_n$ with the splitting constraints $\mathbf{B}_n = \mathbf{A}_n$, $n \in \mathcal{N}$. Likewise, associate additional dual variables $\bar{\mathbf{C}}_n^m$ and $\tilde{\mathbf{C}}_n^m$ ($\bar{\mathbf{D}}_n^m$ and $\tilde{\mathbf{D}}_n^m$) with the first pair of consensus constraints in (4.4) [respectively

(4.5)]. Next introduce the quadratically *augmented* Lagrangian function

$$\mathcal{L}_c\left(\mathcal{V}_1,\mathcal{V}_2,\mathcal{V}_3,\mathcal{M}\right) = \sum_{n=1}^{N}\left[r_n(\mathbf{L}_n,\mathbf{Q}_n,\mathbf{B}_n)+\frac{\lambda_*}{2N}\{N\|\mathbf{L}_n\|_F^2+\|\mathbf{Q}_n\|_F^2\}+\frac{\lambda_1}{N}\|\mathbf{A}_n\|_1\right]$$
$$+\sum_{n=1}^{N}\langle\mathbf{M}_n,\mathbf{B}_n-\mathbf{A}_n\rangle+\frac{c}{2}\sum_{n=1}^{N}\|\mathbf{B}_n-\mathbf{A}_n\|_F^2$$
$$+\sum_{n=1}^{N}\sum_{m\in\mathcal{J}_n}\left\{\langle\bar{\mathbf{C}}_n^m,\mathbf{Q}_n-\bar{\mathbf{F}}_n^m\rangle+\langle\tilde{\mathbf{C}}_n^m,\mathbf{Q}_m-\tilde{\mathbf{F}}_n^m\rangle+\langle\bar{\mathbf{D}}_n^m,\mathbf{A}_n-\bar{\mathbf{G}}_n^m\rangle+\langle\tilde{\mathbf{D}}_n^m,\mathbf{A}_m-\tilde{\mathbf{G}}_n^m\rangle\right\}$$
$$+\frac{c}{2}\sum_{n=1}^{N}\sum_{m\in\mathcal{J}_n}\left\{\|\mathbf{Q}_n-\bar{\mathbf{F}}_n^m\|_F^2+\|\mathbf{Q}_m-\tilde{\mathbf{F}}_n^m\|_F^2+\|\mathbf{A}_n-\bar{\mathbf{G}}_n^m\|_F^2+\|\mathbf{A}_m-\tilde{\mathbf{G}}_n^m\|_F^2\right\} \tag{4.6}$$

where c is a positive penalty coefficient, and the primal variables are split into three groups $\mathcal{V}_1 := \{\mathbf{Q}_n,\mathbf{A}_n\}_{n=1}^N$, $\mathcal{V}_2 := \{\mathbf{L}_n\}_{n=1}^N$, and $\mathcal{V}_3 := \{\mathbf{B}_n,\bar{\mathbf{F}}_n^m,\tilde{\mathbf{F}}_n^m,\bar{\mathbf{G}}_n^m,\tilde{\mathbf{G}}_n^m\}_{n\in\mathcal{N}}^{m\in\mathcal{J}_n}$. For notational convenience, collect all multipliers in $\mathcal{M} := \{\mathbf{M}_n,\bar{\mathbf{C}}_n^m,\tilde{\mathbf{C}}_n^m,\bar{\mathbf{D}}_n^m,\tilde{\mathbf{D}}_n^m\}_{n\in\mathcal{N}}^{m\in\mathcal{J}_n}$. Note that the remaining constraints in (4.4) and (4.5), namely $C_V := \{\bar{\mathbf{F}}_n^m=\tilde{\mathbf{F}}_n^m,\ \bar{\mathbf{G}}_n^m=\tilde{\mathbf{G}}_n^m,\ m\in\mathcal{J}_n,\ n\in\mathcal{N}\}$, have not been dualized.

To minimize (P4) in a decentralized fashion, a variation of the alternating-direction method of multipliers (ADMM) will be adopted here. The ADMM is an iterative augmented Lagrangian method especially well-suited for parallel processing [18], which has been proven successful to tackle the optimization tasks encountered e.g., with decentralized estimation problems [105, 125]. The proposed solver entails an iterative procedure comprising four steps per iteration $k = 1, 2, \ldots$

[S1] Update dual variables for all $n\in\mathcal{N}$, $m\in\mathcal{J}_n$:

$$\mathbf{M}_n[k] = \mathbf{M}_n[k-1]+\mu(\mathbf{B}_n[k]-\mathbf{A}_n[k]) \tag{4.7}$$
$$\bar{\mathbf{C}}_n^m[k] = \bar{\mathbf{C}}_n^m[k-1]+\mu(\mathbf{Q}_n[k]-\bar{\mathbf{F}}_n^m[k]) \tag{4.8}$$
$$\tilde{\mathbf{C}}_n^m[k] = \tilde{\mathbf{C}}_n^m[k-1]+\mu(\mathbf{Q}_m[k]-\tilde{\mathbf{F}}_n^m[k]) \tag{4.9}$$
$$\bar{\mathbf{D}}_n^m[k] = \bar{\mathbf{D}}_n^m[k-1]+\mu(\mathbf{A}_n[k]-\bar{\mathbf{G}}_n^m[k]) \tag{4.10}$$
$$\tilde{\mathbf{D}}_n^m[k] = \tilde{\mathbf{D}}_n^m[k-1]+\mu(\mathbf{A}_m[k]-\tilde{\mathbf{G}}_n^m[k]). \tag{4.11}$$

[S2] Update first group of primal variables:

$$\mathcal{V}_1[k+1] = \arg\min_{\mathcal{V}_1}\mathcal{L}_c\left(\mathcal{V}_1,\mathcal{V}_2[k],\mathcal{V}_3[k],\mathcal{M}[k]\right). \tag{4.12}$$

[S3] Update second group of primal variables:

$$\mathcal{V}_2[k+1] = \arg\min_{\mathcal{V}_2} \mathcal{L}_c\left(\mathcal{V}_1[k+1], \mathcal{V}_2, \mathcal{V}_3[k], \mathcal{M}[k]\right). \tag{4.13}$$

[S4] Update auxiliary primal variables:

$$\mathcal{V}_3[k+1] = \arg\min_{\mathcal{V}_3 \in C_V} \mathcal{L}_c\left(\mathcal{V}_1[k+1], \mathcal{V}_2[k+1], \mathcal{V}_3, \mathcal{M}[k]\right). \tag{4.14}$$

This four-step procedure implements a block-coordinate descent method with dual variable updates. At each step of minimizing the augmented Lagrangian, the variables not being updated are treated as fixed and are substituted with their most up-to-date values. Different from ADMM, the alternating-minimization step here generally cycles over three groups of primal variables $\mathcal{V}_1$-$\mathcal{V}_3$ (cf. two groups in ADMM [17]). In some special instances detailed in Section 4.4.3, cycling over two groups of variables only is sufficient. In [S1], $\mu > 0$ is the step size of the subgradient ascent iterations (4.7)-(4.11). While it is common in ADMM implementations to select $\mu = c$, a distinction between the step size and the penalty parameter is made explicit here in the interest of generality.

Reformulating the estimator (P1) to its equivalent form (P4) renders the augmented Lagrangian in (4.6) highly decomposable. 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 agents $n \in \mathcal{N}$. This in turn leads to highly parallelized, simplified recursions corresponding to the aforementioned four steps. Specifically, it is shown in Appendix that if the multipliers are initialized to zero, [S1]-[S4] constitute the decentralized algorithm tabulated under Algorithm 5. In addition, define the soft-thresholding matrix $\mathcal{S}_\tau(\mathbf{M})$ with (i,j)-th entry given by $\text{sign}(m_{i,j})\max\{|m_{i,j}| - \tau, 0\}$, where $m_{i,j}$ denotes the (i,j)-th entry of $\mathbf{M}$.

Remark 4.2 (Simplification of redundant variables) *Careful inspection of Algorithm 5 reveals that the inherently redundant auxiliary variables and multipliers* $\{\bar{\mathbf{F}}_n^m, \tilde{\mathbf{F}}_n^m, \bar{\mathbf{G}}_n^m, \tilde{\mathbf{G}}_n^m\}$ *and* $\{\tilde{\mathbf{C}}_n^m, \tilde{\mathbf{D}}_n^m\}$ *have been eliminated. Agent* n *does not need to separately keep track of all its non-redundant multipliers* $\{\bar{\mathbf{C}}_n^m, \bar{\mathbf{D}}_n^m\}_{m\in\mathcal{J}_n}$*, but only to update their respective (scaled) sums* $\mathbf{O}_n[k] := 2\sum_{m\in\mathcal{J}_n} \bar{\mathbf{C}}_n^m[k]$ *and* $\mathbf{P}_n[k] := 2\sum_{m\in\mathcal{J}_n} \bar{\mathbf{D}}_n^m[k]$.

Algorithm 5 : ADMM solver per agent $n \in \mathcal{N}$

input $\mathbf{Y}_n, \Omega_n, \mathbf{R}_n, \lambda_*, \lambda_1, c, \mu$

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 $m \in \mathcal{J}_n$

[S1] Update local dual variables:

$\mathbf{M}_n[k] = \mathbf{M}_n[k-1] + \mu(\mathbf{B}_n[k] - \mathbf{A}_n[k])$

$\mathbf{O}_n[k] = \mathbf{O}_n[k-1] + \mu \sum_{m\in\mathcal{J}_n}(\mathbf{Q}_n[k] - \mathbf{Q}_m[k])$

$\mathbf{P}_n[k] = \mathbf{P}_n[k-1] + \mu \sum_{m\in\mathcal{J}_n}(\mathbf{A}_n[k] - \mathbf{A}_m[k])$

[S2] Update first group of local primal variables:

$\mathbf{Q}_n[k+1] = \arg\min_{\mathbf{Q}} \Big\{ r_n(\mathbf{L}_n[k], \mathbf{Q}, \mathbf{B}_n[k]) + \frac{\lambda_*}{2N}\|\mathbf{Q}\|_F^2 + \langle \mathbf{O}_n[k], \mathbf{Q}\rangle + c\sum_{m\in\mathcal{J}_n} \Big\| \mathbf{Q} - \frac{\mathbf{Q}_n[k]+\mathbf{Q}_m[k]}{2} \Big\|_F^2 \Big\}$

$\mathbf{H}_n[k+1] := \mathbf{M}_n[k] + c\mathbf{B}_n[k] - \mathbf{P}_n[k] + c\sum_{m\in\mathcal{J}_m}(\mathbf{A}_n[k] + \mathbf{A}_m[k])$

$\mathbf{A}_n[k+1] = [c(1+2|\mathcal{J}_n|)]^{-1}\mathcal{S}_{\lambda_1/N}(\mathbf{H}_n[k+1])$

[S3] Update second group of local primal variables:

$\mathbf{L}_n[k+1] = \arg\min_{\mathbf{L}} \big\{ r_n(\mathbf{L}, \mathbf{Q}_n[k+1], \mathbf{B}_n[k]) + \frac{\lambda_*}{2}\|\mathbf{L}\|_F^2 \big\}$

[S4] Update auxiliary local primal variables:

$\mathbf{B}_n[k+1] = \arg\min_{\mathbf{B}} \Big\{ r_n(\mathbf{L}_n[k+1], \mathbf{Q}_n[k+1], \mathbf{B}) + \langle \mathbf{M}_n[k], \mathbf{B}\rangle + \frac{c}{2}\|\mathbf{B} - \mathbf{A}_n[k+1]\|_F^2 \Big\}$

Broadcast $\{\mathbf{Q}_n[k+1], \mathbf{A}_n[k+1]\}$ to neighbors $m \in \mathcal{J}_n$

end for

return $\mathbf{A}_n, \mathbf{Q}_n, \mathbf{L}_n$

Remark 4.3 (Computational and communication cost) *The main computational burden of the algorithm stems from solving unconstrained quadratic programs locally to update* $\{\mathbf{Q}_n, \mathbf{L}_n, \mathbf{B}_n\}$*, and to carry out simple soft-thresholding operations to update* $\mathbf{A}_n$*. On a per-iteration basis, network agents communicate their updated local estimates* $\{\mathbf{Q}_n[k], \mathbf{A}_n[k]\}$ *with their neighbors, to carry out the 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 when* ρ *is small. In addition, the* $F \times T$ *matrix* $\mathbf{A}_n[k]$ *can be communicated efficiently after few iterations required to consent on the common support (especially when the local estimates are initialized to zero). Observe that the dual variables need not be exchanged.*

Remark 4.4 (General sparsity-promoting regularization) *Even though $\lambda_1\|\mathbf{A}\|_1$ was adopted in (P1) to encourage sparsity in the entries of $\mathbf{A}$, the algorithmic framework here can accommodate more general structured sparsity-promoting penalties $\psi(\mathbf{A})$. To maintain the per-agent computational complexity at affordable levels, the minimum requirement on the admissible penalties is that the proximal operator*

$$prox_{\psi}(\tilde{\mathbf{Y}}) := \arg\min_{\mathbf{A}} \left[\frac{1}{2}\|\tilde{\mathbf{Y}} - \mathbf{A}\|_F^2 + \psi(\mathbf{A})\right] \tag{4.15}$$

is given in terms of vector or (and) scalar soft-thresholding operators. In addition to the ℓ_1-norm (Lasso penalty), this holds for the sum of row-wise ℓ_2-norms (group Lasso penalty [156]), or, a linear combination of the aforementioned two – the so-termed hierarchical Lasso penalty that encourages sparsity across and within the rows of $\mathbf{A}$ [133]. All this is possible since by introducing the cost-splitting variables $\mathbf{B}_n$, the local sparse matrix updates are $\mathbf{A}_n[k+1] = prox_{\psi}(\tilde{\mathbf{Y}}_n[k])$ for suitable $\tilde{\mathbf{Y}}_n[k]$ (see Appendix). Relying on similar ideas, proximal-splitting algorithms have been successfully adopted for various signal processing tasks [40], and for parallel optimization [39].

When employed to solve non-convex problems such as (P4), ADMM (or its variant used here) offers no convergence guarantees. However, there is ample experimental evidence in the literature that supports empirical convergence of ADMM, especially when the non-convex problem at hand exhibits "favorable" structure. For instance, (P4) is bi-convex and gives rise to the strictly convex optimization subproblems (4.12)-(4.14), which admit unique closed-form solutions per iteration. This observation and the linearity of the constraints endow Algorithm 5 with good convergence properties – extensive numerical tests including those presented in Section 4.5 demonstrate that this is indeed the case. While a formal convergence proof goes beyond the scope of this paper, the following proposition proved in Appendix asserts that upon convergence, Algorithm 5 attains consensus and global optimality.

Proposition 4.2 *If the sequence of iterates $\{\mathbf{Q}_n[k], \mathbf{L}_n[k], \mathbf{A}_n[k]\}_{n\in\mathcal{N}}$ generated by Algorithm 5 converge to $\{\bar{\mathbf{Q}}_n, \bar{\mathbf{L}}_n, \bar{\mathbf{A}}_n\}_{n\in\mathcal{N}}$, and (a1) holds, then: i) $\bar{\mathbf{Q}}_n = \bar{\mathbf{Q}}_m$, $\bar{\mathbf{A}}_n =$*

$\bar{\mathbf{A}}_m,\ n,m \in \mathcal{N}$*; and ii) if* $\|\mathcal{P}_\Omega(\mathbf{Y}-\bar{\mathbf{L}}\bar{\mathbf{Q}}_1'-\mathbf{R}\bar{\mathbf{A}}_1)\| \leq \lambda_*$*, then* $\hat{\mathbf{X}}=\bar{\mathbf{L}}\bar{\mathbf{Q}}_1'$ *and* $\hat{\mathbf{A}}=\bar{\mathbf{A}}_1$*, where* $\{\hat{\mathbf{A}},\hat{\mathbf{X}}\}$ *is the global optimum of (P1).*

4.4 Applications

This section outlines a few applications that could benefit from the decentralized sparsity-regularized rank minimization framework described so far. In each case, the problem statement calls for estimating low-rank $\mathbf{X}$ and (or) sparse $\mathbf{A}$, given decentralized data adhering to an application-dependent model subsumed by (4.2). Customized algorithms are thus obtained as special cases of the general iterations in Algorithm 5.

4.4.1 Unveiling traffic anomalies in backbone networks

In the backbone of large-scale networks, origin-to-destination (OD) traffic flows experience abrupt 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 [72]. Unveiling such anomalies is a crucial task in engineering network traffic. This is a challenging task however, since the available data are usually high-dimensional noisy link-load measurements, which comprise the superposition of *unobservable* OD flows as explained next.

The network is modeled as in Section 4.2 (as also discussed in the previous chapters), and transports a set of end-to-end flows $\mathcal{F}$ (with $|\mathcal{F}| := F$) associated with specific OD pairs. For backbone networks, the number of network layer flows is typically much larger than the number of physical links ($F \gg L$). Single-path routing is considered here to send the traffic flow from an origin to its intended destination. Accordingly, for a particular flow multiple links connecting the corresponding OD pair are chosen to carry the traffic. Sparing details that can be found in [97], the traffic $\mathbf{Y} := [y_{l,t}] \in \mathbb{R}^{L\times T}$ carried over links $l \in \mathcal{L}$ and measured at time instants $t \in [1,T]$ can be compactly expressed as

$$\mathbf{Y} = \mathbf{R}\,(\mathbf{Z}+\mathbf{A}) + \mathbf{V} \tag{4.16}$$

where the fat routing matrix $\mathbf{R} := [r_{\ell,f}] \in \{0,1\}^{L\times F}$ is fixed and given, $\mathbf{Z} := [z_{f,t}]$ denotes the unknown "clean" traffic flows over the time horizon of interest, and $\mathbf{A} := [a_{f,t}]$ collects the traffic volume anomalies. These data are decentralized. Agent n acquires a few rows of $\mathbf{Y}$ corresponding to the link-load traffic measurements $\mathbf{Y}_n \in \mathbb{R}^{L_n\times T}$ from its outgoing links, and has available its local routing table $\mathbf{R}_n$ which indicates the OD flows routed through n. Assuming a suitable ordering of links, the per-agent quantities relate to their global counterparts in (4.16) through $\mathbf{Y} := [\mathbf{Y}_1', \ldots, \mathbf{Y}_N']'$ and $\mathbf{R} := [\mathbf{R}_1', \ldots, \mathbf{R}_N']'$.

Common temporal patterns among the traffic flows in addition to their periodic behavior, render most rows (respectively columns) of $\mathbf{Z}$ linearly dependent, and thus $\mathbf{Z}$ typically has low rank [72]. Anomalies are expected to occur sporadically over time, and only last for short periods relative to the (possibly long) measurement interval $[1, T]$. In addition, only a small fraction of the flows are supposed to be anomalous at any given time instant. This renders the anomaly matrix $\mathbf{A}$ sparse across rows and columns. Given local measurements $\{\mathbf{Y}_n\}_{n\in\mathcal{N}}$ and the routing tables $\{\mathbf{R}_n\}_{n\in\mathcal{N}}$, the goal is to estimate $\mathbf{A}$ in a decentralized fashion, by capitalizing on the sparsity of $\mathbf{A}$ and the low-rank property of $\mathbf{Z}$. Since the primary goal is to recover $\mathbf{A}$, define $\mathbf{X} := \mathbf{RZ}$ which inherits the low-rank property from $\mathbf{Z}$, and consider [cf. (4.16)]

$$\mathbf{Y} = \mathbf{X} + \mathbf{RA} + \mathbf{V}. \tag{4.17}$$

Model (4.17) is a special case of (4.2), when all the entries of $\mathbf{Y}$ are observed, i.e., $\Omega = \{1, \ldots, L\} \times \{1, \ldots, T\}$. Note that $\mathbf{RA}$ is not sparse even though $\mathbf{A}$ is itself sparse, hence principal components pursuit is not applicable here [163]. Instead, the following estimator is adopted to unveil network anomalies [97]

$$\{\hat{\mathbf{X}}, \hat{\mathbf{A}}\} = \arg\min_{\{\mathbf{X},\mathbf{A}\}} \sum_{n=1}^{N} \left[\frac{1}{2}\|\mathbf{Y}_n - \mathbf{X}_n - \mathbf{R}_n\mathbf{A}\|_F^2 + \frac{\lambda_*}{N}\|\mathbf{X}\|_* + \frac{\lambda_1}{N}\|\mathbf{A}\|_1\right]$$

which is subsumed by (P1). Accordingly, a decentralized algorithm can be readily obtained by simplifying the general iterations under Algorithm 5, the subject dealt with next.

Distributed Algorithm for Unveiling Network Anomalies (DUNA). For the specific case here in which $\Omega = \{1, \ldots, L\} \times \{1, \ldots, T\}$, the residuals in Algorithm 5 reduce to

Algorithm 6 : DUNA per agent $n \in \mathcal{N}$

input $\mathbf{Y}_n, \mathbf{R}_n, \lambda_*, \lambda_1, c, \mu$

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 $m \in \mathcal{J}_n$

[S1] Update local dual variables:

$\mathbf{M}_n[k] = \mathbf{M}_n[k-1] + \mu(\mathbf{B}_n[k] - \mathbf{A}_n[k])$

$\mathbf{O}_n[k] = \mathbf{O}_n[k-1] + \mu \sum_{m\in\mathcal{J}_n}(\mathbf{Q}_n[k] - \mathbf{Q}_m[k])$

$\mathbf{P}_n[k] = \mathbf{P}_n[k-1] + \mu \sum_{m\in\mathcal{J}_n}(\mathbf{A}_n[k] - \mathbf{A}_m[k])$

[S2] Update first group of local primal variables:

$\mathbf{G}_n[k+1] := (\mathbf{Y}_n - \mathbf{R}_n\mathbf{B}_n[k])'\mathbf{L}_n[k] - \mathbf{O}_n[k] + c\sum_{m\in\mathcal{J}_n}(\mathbf{Q}_n[k] + \mathbf{Q}_m[k])$

$\mathbf{Q}_n[k+1] = \mathbf{G}_n[k+1]\left[\mathbf{L}_n'[k]\mathbf{L}_n[k] + (\lambda_*/N + 2c|\mathcal{J}_n|)\underline{\mathrm{I}}_\rho\right]^{-1}$

$\mathbf{H}_n[k+1] := \mathbf{M}_n[k] + c\mathbf{B}_n[k] - \mathbf{P}_n[k] + c\sum_{m\in\mathcal{J}_m}(\mathbf{A}_n[k] + \mathbf{A}_m[k])$

$\mathbf{A}_n[k+1] = [c(1+2|\mathcal{J}_n|)]^{-1}\mathcal{S}_{\lambda_1/N}(\mathbf{H}_n[k+1])$

[S3] Update second group of local primal variables:

$\mathbf{L}_n[k+1] = (\mathbf{Y}_n - \mathbf{R}_n\mathbf{B}_n[k])\,\mathbf{Q}_n[k+1] \times [\mathbf{Q}_n'[k+1]\mathbf{Q}_n[k+1] + \lambda_*\mathbf{I}_\rho]^{-1}$

[S4] Update auxiliary local primal variables:

$\mathbf{S}_n[k+1] := \mathbf{R}_n'(\mathbf{Y}_n - \mathbf{L}_n[k+1]\mathbf{Q}_n'[k+1]) - \mathbf{M}_n[k] + c\mathbf{A}_n[k+1]$

$\mathbf{B}_n[k+1] = [\mathbf{R}_n'\mathbf{R}_n + c\mathbf{I}_F]^{-1}\mathbf{S}_n[k+1]$

Broadcast $\{\mathbf{Q}_n[k+1], \mathbf{A}_n[k+1]\}$ to neighbors $m \in \mathcal{J}_n$

end for

return $\mathbf{A}_n, \mathbf{Q}_n, \mathbf{L}_n$

$r_n(\mathbf{L}_n, \mathbf{Q}_n, \mathbf{B}_n) := \frac{1}{2}\|\mathbf{Y}_n - \mathbf{L}_n\mathbf{Q}_n' - \mathbf{R}_n\mathbf{B}_n\|_F^2$. Accordingly, to update the primal variables $\mathbf{Q}_n[k+1]$, $\mathbf{L}_n[k+1]$ and $\mathbf{B}_n[k+1]$ as per Algorithm 5, one needs to solve respective unconstrained strictly convex quadratic optimization problems. These admit closed-form solutions detailed under Algorithm 6. The DUNA updates of the local anomaly matrices $\mathbf{A}_n[k+1]$ are given in terms of soft-thresholding operations, as in Algorithm 5.

Conceivably, the number of flows F can be quite large, thus inverting the $F \times F$ matrix $\mathbf{R}_n'\mathbf{R}_n + c\mathbf{I}_F$ to update $\mathbf{B}_n[k+1]$ could be complex computationally. Fortunately, the inversion needs to be carried out once, and can be performed and cached off-line. In addition, to reduce the inversion cost, the SVD of the local routing matrices $\mathbf{R}_n = \mathbf{U}_{R_n}\mathbf{\Sigma}_{R_n}\mathbf{V}_{R_n}'$

can be obtained first, and the matrix inversion lemma can be subsequently employed to obtain $[\mathbf{R}_n'\mathbf{R}_n + c\mathbf{I}_F]^{-1} = (1/c)\left[\mathbf{I}_p - \mathbf{V}_{R_n}\mathbf{C}\mathbf{V}_{R_n}'\right]$, where $\mathbf{C} := \text{diag}\left(\frac{\sigma_1^2}{c+\sigma_1^2}, ..., \frac{\sigma_p^2}{c+\sigma_p^2}\right)$ and $p = \text{rank}(\mathbf{R}_n) \ll F$. This computational shortcut is commonly adopted in statistical learning algorithms when ridge regression estimates are sought, and the number of variables is much larger than the number of elements in the training set [56, Ch. 18]. During the operational phase of the algorithm, the main computational burden of DUNA comes from repeated inversions of (small) $\rho \times \rho$ matrices, and parallel soft-thresholding operations. The communication overhead is identical to the one incurred by Algorithm 5 (cf. Remark 4.3).

Remark 4.5 (Incomplete link traffic measurements) *In general, one can allow for missing traffic data and the DUNA updates are still expressible in closed form.*

4.4.2 In-network robust principal component analysis

Principal component analysis (PCA) is the workhorse of high-dimensional data analysis and dimensionality reduction, with numerous applications in statistics, networking, engineering, and the biobehavioral sciences; see, e.g., [64]. Nowadays ubiquitous e-commerce sites, complex networks such as the Web, and urban traffic surveillance systems generate massive volumes of data. As a result, extracting the most informative, yet low-dimensional structure from high-dimensional datasets is of paramount importance [56].

Data obeying postulated low-rank models include also outliers, which are samples not adhering to those nominal models. Unfortunately, similar to LS estimates PCA is very sensitive to the outliers [64]. While robust approaches to PCA are available, recently polynomial-time algorithms with remarkable performance guarantees have emerged for low-rank matrix recovery in the presence of sparse – but otherwise arbitrarily large – errors [25,33,163]. Robust PCA is of great interest in networking-related applications. One can think of decentralized estimation using reduced-dimensionality sensor observations [125], and unveiling anomalous flows in backbone networks from Netflow data [3]; see also Section 4.5.2.

In the network setting of Section 4.2, each agent $n \in \mathcal{N}$ acquires F_n outlier-plus-noise corrupted rows of matrix $\mathbf{Y} := [\mathbf{Y}_1', \ldots, \mathbf{Y}_N']'$, where $\sum_{n=1}^N F_n = F$. Local data can thus be modeled as $\mathbf{Y}_n = \mathbf{X}_n + \mathbf{A}_n + \mathbf{V}_n$, where $\mathbf{X} := [\mathbf{X}_1', \ldots, \mathbf{X}_N']'$ has low rank. Agents

want to estimate $\mathbf{X}_n$ (and the outliers $\mathbf{A}_n$) in a decentralized fashion by forming the global estimator [163]

$$\{\hat{\mathbf{X}}, \hat{\mathbf{A}}\} = \arg \min_{\{\mathbf{X},\mathbf{A}\}} \sum_{n=1}^{N} \left[\frac{1}{2}\|\mathbf{Y}_n - \mathbf{X}_n - \mathbf{A}_n\|_F^2 + \frac{\lambda_*}{N}\|\mathbf{X}\|_* + \lambda_1\|\mathbf{A}_n\|_1 \right] \tag{4.18}$$

which is once more a special case of (P1) when $\mathbf{R} = \mathbf{I}_F$.

Distributed Robust Principal Component Analysis (DRPCA) Algorithm. Regarding the general decentralized formulation in (P4), the first constraint is no longer needed since $\mathbf{R} = \mathbf{I}_F$ [cf. the discussion after (P4)]. As agent n is interested in estimating $\mathbf{A}_n$ and $\|\mathbf{A}\|_1$ is separable over the rows of $\mathbf{A}$, the only required constraints are $\mathbf{Q}_n = \mathbf{Q}_m,\ m \in \mathcal{J}_n,\ n \in \mathcal{N}$. These are associated with the dual variables $\mathbf{O}_n$ per agent, and are updated according to Algorithm 7. All in all, each agent stores and recursively updates the primal variables $\{\mathbf{Q}_n, \mathbf{L}_n\}$, along with the $F_n \times T$ matrix $\mathbf{A}_n$.

Mimicking the procedure that led to Algorithm 5, one finds that primal variable updates in DRPCA are expressible in closed form. In particular, the local outlier matrix $\mathbf{A}_n[k+1]$ minimizes the Lasso cost

$$\mathbf{A}_n[k+1] = \arg \min_{\{\mathbf{A}_n\}} \left\{ \frac{1}{2}\|\mathbf{Y}_n - \mathbf{L}_n[k+1]\mathbf{Q}_n'[k+1] - \mathbf{A}_n\|_F^2 + \lambda_1\|\mathbf{A}_n\|_1 \right\}$$

and is given in terms of soft-thresholding operations as seen in Algorithm 7 [observe that $\mathbf{A}_n[k+1] = \text{prox}_{\|\cdot\|_1}(\mathbf{Y}_n - \mathbf{L}_n[k+1]\mathbf{Q}_n'[k+1])$, where $\text{prox}_\psi(\cdot)$ is defined in (4.15)].

DRPCA iterations are simple with small $\rho \times \rho$ matrices inverted per iteration to update $\mathbf{L}_n$ and $\mathbf{Q}_n$ (see Algorithm 7). Regarding communication cost, each agent only broadcasts a $T \times \rho$ matrix $\mathbf{Q}_n$ to its neighbors.

4.4.3 Distributed low-rank matrix completion

The ability to recover a low-rank matrix from a subset of its entries is the leitmotif of recent advances for localization of wireless sensors [108], Internet traffic analysis [79], [160], and preference modeling for recommender systems [6]. In the *low-rank matrix completion* problem, given a limited number of (possibly) noise corrupted entries of a low-rank matrix $\mathbf{X}$, the goal is to recover the entire matrix while denoising the observed entries, and accurately imputing the missing ones.

Algorithm 7 : DRPCA algorithm per agent $n \in \mathcal{N}$

input $\mathbf{Y}_n, \lambda_*, \lambda_1, c, \mu$

initialize $\mathbf{A}_n[1] = \mathbf{0}_{F_n \times T}$, $\mathbf{O}[0] = \mathbf{0}_{T \times \rho}$, and $\mathbf{L}_n[1]$, $\mathbf{Q}_n[1]$ at random.

for $k = 1, 2, \ldots$ **do**

Receive $\{\mathbf{Q}_m[k]\}$ from neighbors $m \in \mathcal{J}_n$

[S1] Update local dual variables:

$\mathbf{O}_n[k] = \mathbf{O}_n[k-1] + \mu \sum_{m \in \mathcal{J}_n} (\mathbf{Q}_n[k] - \mathbf{Q}_m[k])$

[S2] Update first group of local primal variables:

$\mathbf{G}_n[k+1] := (\mathbf{Y}_n - \mathbf{A}_n[k])' \mathbf{L}_n[k] - \mathbf{O}_n[k] + c \sum_{m \in \mathcal{J}_n} (\mathbf{Q}_n[k] + \mathbf{Q}_m[k])$

$\mathbf{Q}_n[k+1] = \mathbf{G}_n[k+1] \left[\mathbf{L}_n'[k] \mathbf{L}_n[k] + (\lambda_*/N + 2c|\mathcal{J}_n|)\mathbf{I}_\rho\right]^{-1}$

[S2] Update second group of local primal variables:

$\mathbf{L}_n[k+1] = (\mathbf{Y}_n - \mathbf{A}_n[k])\, \mathbf{Q}_n[k+1] \times \left[\mathbf{Q}_n'[k+1] \mathbf{Q}_n[k+1] + \lambda_* \mathbf{I}_\rho\right]^{-1}$

[S3] Update third group of local primal variables:

$\mathbf{A}_n[k+1] = \mathcal{S}_{\lambda_1} \left(\mathbf{Y}_n - \mathbf{L}_n[k+1] \mathbf{Q}_n'[k+1]\right)$

Broadcast $\{\mathbf{Q}_n[k+1]\}$ to neighbors $m \in \mathcal{J}_n$

end for

return $\mathbf{A}_n, \mathbf{Q}_n, \mathbf{L}_n$

In the network setting envisioned here, agent $n \in \mathcal{N}$ has available L_n incomplete and noise-corrupted rows of $\mathbf{Y} := [\mathbf{Y}_1', \ldots, \mathbf{Y}_N']'$. Local data can thus be modeled as $\mathcal{P}_{\Omega_n}(\mathbf{Y}_n) = \mathcal{P}_{\Omega_n}(\mathbf{X}_n + \mathbf{V}_n)$. Relying on in-network processing, agents aim at completing their own rows by forming the global estimator

$$\hat{\mathbf{X}} = \arg\min_{\mathbf{X}} \sum_{n=1}^{N} \left[\frac{1}{2} \|\mathcal{P}_{\Omega_n}(\mathbf{Y}_n - \mathbf{X}_n)\|_F^2 + \frac{\lambda_*}{N} \|\mathbf{X}\|_* \right] \tag{4.19}$$

which exploits the low-rank property of $\mathbf{X}$ through nuclear-norm regularization. Estimator (4.19) was proposed in [26], and solved centrally whereby all data $\mathcal{P}_{\Omega_n}(\mathbf{Y}_n)$ is available to feed e.g., an off-the-shelf semidefinite programming (SDP) solver. The general estimator in (P1) reduces to (4.19) upon setting $\mathbf{R} = \mathbf{0}_{L \times F}$ and $\lambda_1 = 0$. Hence, it is possible to derive a *decentralized* algorithm for low-rank matrix completion by specializing Algorithm 5 to the setting here.

Before discussing the algorithmic details, a brief parenthesis is in order to touch upon properties of local sampling operators. Operator $\mathcal{P}_{\Omega_n}$ is a linear orthogonal projector, since

it projects its matrix argument onto the *subspace* $\Psi_n := \{\mathbf{Z} \in \mathbb{R}^{L_n \times T} : \text{supp}(\mathbf{Z}) \in \Omega_n\}$ of matrices with support contained in Ω_n. Linearity of $\mathcal{P}_{\Omega_n}$ implies that $\text{vec}(\mathcal{P}_{\Omega_n}(\mathbf{Z})) = \mathbf{A}_{\Omega_n}\text{vec}(\mathbf{Z})$, where $\mathbf{A}_{\Omega_n} \in \mathbb{R}^{L_n \times T}$ is a symmetric and idempotent projection matrix that will prove handy later on. To characterize $\mathbf{A}_{\Omega_n}$, introduce an $L_n \times T$ masking matrix $\mathbf{\Omega}_n$ whose (l,t)-th entry equals one when $(l,t) \in \Omega_n$, and zero otherwise. Since $\mathcal{P}_{\Omega_n}(\mathbf{Z}) = \mathbf{\Omega}_n \odot \mathbf{Z}$, from standard properties of the vec$(\cdot)$ operator it follows that $\mathbf{A}_{\Omega_n} = \text{diag}(\text{vec}(\mathbf{\Omega}_n))$.

Distributed Matrix Completion (DMC) Algorithm. Going back to the general decentralized formulation in (P4), since there is no sparse component $\mathbf{A}$ in the matrix completion problem (4.19), the only constraints that remain are $\mathbf{Q}_n = \mathbf{Q}_m$, $m \in \mathcal{J}_n$, $n \in \mathcal{N}$. These correspond to the dual variables $\mathbf{O}_n[k]$ per agent, and are updated as shown in Algorithm 8.

In the absence of $\{\mathbf{A}_n\}_{n\in\mathcal{N}}$ and the auxiliary variables $\{\mathbf{B}_n\}_{n\in\mathcal{N}}$, it suffices to cycle over two groups of primal variables to arrive at the DMC iterations. The primal variable updates can be readily obtained by capitalizing on the properties of the vec$(\cdot)$ operator. In particular, Algorithm 5 indicates that the recursions for $\mathbf{Q}_n$ are given by [let $\mathbf{q} := \text{vec}(\mathbf{Q}')$]

$$\mathbf{q}_n[k+1] = \arg\min_{\mathbf{q}} \Big\{ \frac{1}{2}\|\mathbf{A}_{\Omega_n}(\text{vec}(\mathbf{Y}_n) - (\mathbf{I} \otimes \mathbf{L}_n[k])\mathbf{q}\|^2) + \frac{\lambda_*}{2N}\|\mathbf{q}\|^2 + \langle \text{vec}(\mathbf{O}_n{}'[k]), \mathbf{q}\rangle + c \sum_{m\in\mathcal{J}_n} \Big\| \mathbf{q} - \frac{\text{vec}(\mathbf{Q}_n{}'[k] + \mathbf{Q}_m{}'[k])}{2} \Big\|^2 \Big\}. \tag{4.20}$$

Likewise, $\mathbf{L}_n$ is updated by solving the following subproblem per iteration (let $\mathbf{l} := \text{vec}(\mathbf{L})$)

$$\mathbf{l}_n[k+1] = \arg\min_{\mathbf{l}} \Big\{ \frac{1}{2}\|\mathbf{A}_{\Omega_n}(\text{vec}(\mathbf{Y}_n) - (\mathbf{Q}_n[k+1] \otimes \mathbf{I}_{L_n})\mathbf{l})\|^2 + \frac{\lambda_*}{2}\|\mathbf{l}\|^2 \Big\}. \tag{4.21}$$

Both (4.20) and (4.21) are unconstrained convex quadratic problems, which admit the closed-form solutions tabulated under Algorithm 8.

The per-agent computational complexity of the DMC algorithm is dominated by repeated inversions of $\rho \times \rho$ and $\rho L_n \times \rho L_n$ matrices to obtain $\mathbf{E}_n[k+1]$ and $\mathbf{D}_n[k+1]$, respectively, and matrix multiplications to update $\mathbf{Q}_n[k+1]$ and $\mathbf{L}_n[k+1]$ (cf. Algorithm 8). Notice that $\mathbf{E}_n[k+1] \in \mathbb{R}^{\rho T \times \rho T}$ has block-diagonal structure with blocks of size $\rho \times \rho$. Overall, the per-iteration complexity across the network is upper bounded by $\mathcal{O}(\rho^3 NT)$, which grows linearly with the network size. This is affordable since in practice ρ is typically small for a number of applications of interest (cf. the low-rank assumption). In addition,

Algorithm 8 : DMC algorithm per agent $n \in \mathcal{N}$

Input $\mathbf{Y}_n, \Omega_n, \mathbf{A}_{\Omega_n}, \lambda_*, c, \mu$

Initialize $\mathbf{O}[0] = \mathbf{0}_{T\times\rho}$, and $\mathbf{L}_n[1]$, $\mathbf{Q}_n[1]$ at random

for $k = 1, 2, \ldots$ **do**

Receive $\{\mathbf{Q}_m[k]\}$ from neighbors $m \in \mathcal{J}_n$

[S1] Update local dual variables:

$\mathbf{O}_n[k] = \mathbf{O}_n[k-1] + \mu \sum_{m\in\mathcal{J}_n}(\mathbf{Q}_n[k] - \mathbf{Q}_m[k])$

[S2] Update first group of local primal variables:

$\mathbf{E}_n[k+1] = \{(\mathbf{I}_T \otimes \mathbf{L}'_n[k])\mathbf{A}_{\Omega_n}(\mathbf{I}_T \otimes \mathbf{L}_n[k]) + (\lambda_*/N + 2c|\mathcal{J}_n|)\mathbf{I}_{\rho T}\}^{-1}$

$\mathbf{G}_n[k+1] := (\mathbf{I}_T \otimes \mathbf{L}'_n[k])\mathbf{A}_{\Omega_n}\text{vec}(\mathbf{Y}_n) - \text{vec}(\mathbf{O}'_n[k]) + c\text{vec}(\sum_{m\in\mathcal{J}_n}(\mathbf{Q}'_n[k] + \mathbf{Q}'_m[k]))$

$\mathbf{Q}'_n[k+1] = \text{unvec}\big(\mathbf{E}_n[k+1]\mathbf{G}_n[k+1]\big)$

[S3] Update second group of local primal variables:

$\mathbf{D}_n[k+1] := \{(\mathbf{Q}'_n[k+1] \otimes \mathbf{I}_{L_n})\mathbf{A}_{\Omega_n}(\mathbf{Q}_n[k+1] \otimes \mathbf{I}_{L_n}) + \lambda_*\mathbf{I}_{\rho L_n}\}^{-1}$

$\mathbf{L}_n[k+1] = \text{unvec}\,(\mathbf{D}_n[k+1]\ (\mathbf{Q}'_n[k+1] \otimes \mathbf{I}_{L_n})\mathbf{A}_{\Omega_n}\text{vec}(\mathbf{Y}_n))$

Broadcast $\{\mathbf{Q}_n[k+1]\}$ to neighbors $m \in \mathcal{J}_n$

end for

Return $\mathbf{Q}_n, \mathbf{L}_n$

L_n, the number of row vectors acquired per agent, and T, the number of time instants for data collection, can be controlled by the designer to accommodate a prescribed maximum computational complexity. One can also benefit from the decomposability of (4.21) and (4.20) across rows of $\mathbf{L}$ and $\mathbf{Q}$, respectively, and parallelize the row updates. This way, one only needs to invert $\rho \times \rho$ matrices. On a per-iteration basis, network agents communicate their updated local estimates $\mathbf{Q}_n[k]$ only with their neighbors, in order to carry out the updates of primal and dual variables during the next iteration. In terms of 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. Observe that the dual variables $\mathbf{O}_n[k]$ need not be exchanged, and the overall communication cost does not depend on the network size N.

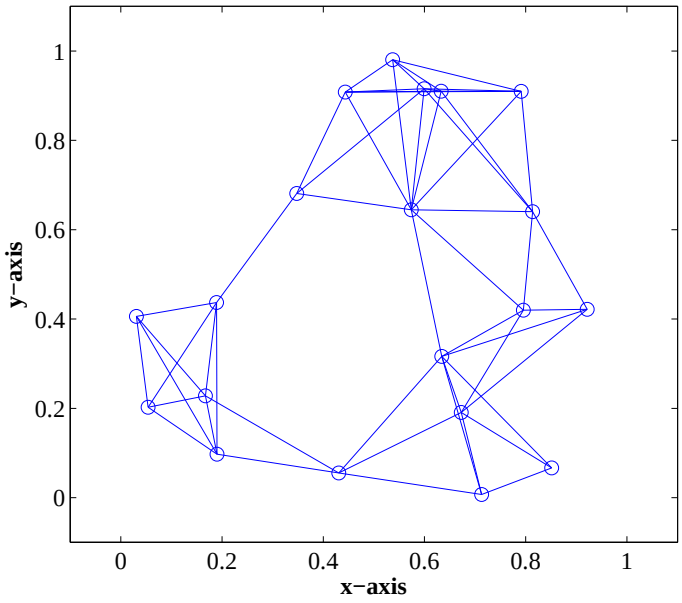


Figure 4.1: A network of $N = 20$ agents.

4.5 Numerical Tests

This section corroborates convergence and gauges performance of the proposed algorithms, when tested on the applications of Section 4.4 using synthetic and real network data.

Synthetic network data. A network of $N = 20$ agents is considered as a realization of the random geometric graph model, that is, agents are randomly placed on the unit square and two agents communicate with each other if their Euclidean distance is less than a prescribed communication range of $d_c = 0.35$; see Fig. 4.1. The network graph is bidirectional and comprises $L = 106$ links, and $F = N(N-1) = 380$ OD flows. The entries of $\mathbf{V}$ are independent and identically distributed (i.i.d.), zero-mean, Gaussian with variance σ^2; i.e., $v_{l,t} \sim N(0, \sigma^2)$. Low-rank matrices with rank r are generated from the bilinear factorization model $\mathbf{X}_0 = \mathbf{W}\mathbf{Z}'$, where $\mathbf{W}$ and $\mathbf{Z}$ are $L \times r$ and $T \times r$ matrices with i.i.d. entries drawn from Gaussian distributions $N(0, 100/F)$ and $N(0, 100/T)$, respectively. Every entry of $\mathbf{A}_0$ is randomly drawn from the set $\{-1, 0, 1\}$ with $\Pr(a_{i,j} = -1) = \Pr(a_{i,j} = 1) = \pi/2$. Unless otherwise stated, $r = 3$, $\rho = 3$ and $T = F = 380$ are used throughout. Different values of

σ, and π are examined.

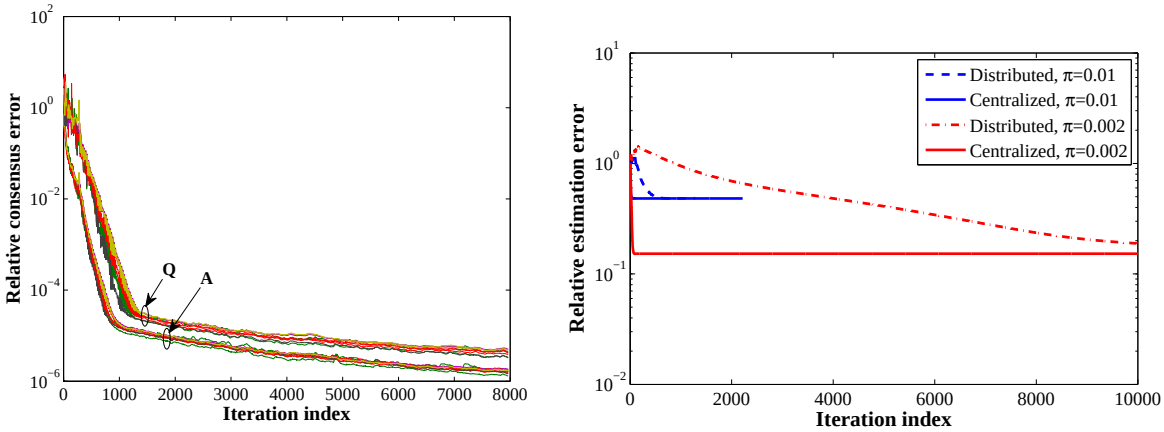
Internet2 network data. Real data including OD flow traffic levels and end-to-end latencies are collected from the operation of the Internet2 network (Internet backbone network across USA) [2]. Both versions of the Internet2 network, referred as v1 and v2, are considered. OD flow traffic levels are recorded for a three-week operation of Internet2-v1 during Dec. 8–28, 2008 [72], and are used to assess performance of DUNA and DRPCA (see Sections 4.5.1 and 4.5.2 next). Internet2-v1 contains $N = 11$ agents, $L = 41$ links, and $F = 121$ flows. To test the DMC algorithm, end-to-end flow latencies are collected from the operation of Internet2-v2 during Aug. 18–22, 2011 [2]. The Internet2-v2 network comprises $N = 9$ agents, $L = 26$ links, and $F = 81$ flows.

Selection of tuning parameters. The sparsity- and rank-controlling parameters λ_1 and λ_* are tuned to optimize performance. The optimality conditions for (P1) indicate that for $\lambda_1 > \|\mathbf{R}'\mathbf{Y}\|_\infty$ and $\lambda_* > \|\mathbf{Y}\|$, $\{\mathbf{X}_0 = \mathbf{0}_{L\times T}, \mathbf{A}_0 = \mathbf{0}_{F\times T}\}$ is the unique optimal solution. This in turn confines the search space for λ_1 and λ_* to the intervals $(0, \|\mathbf{R}'\mathbf{Y}\|_\infty]$ and $(0, \|\mathbf{Y}\|]$, respectively. In addition, for the case of matrix completion and robust PCA one can use the heuristic rules proposed in e.g., [26] and [25].

4.5.1 Unveiling network anomalies

Data is generated from $\mathbf{Y} = \mathbf{R}(\mathbf{X}_0 + \mathbf{A}_0) + \mathbf{V}$, where the routing matrix $\mathbf{R}$ is obtained after determining shortest-path routes of the OD flows. For $\mu = c = 0.1$, DUNA is run until convergence is attained. These values were experimentally chosen to obtain the fastest convergence rate. The time evolution of consensus among agents is depicted in Fig. 4.2 (left), for representative agents in the network. The metric of interest here is the relative error $\|\mathbf{Q}_n[k] - \bar{\mathbf{Q}}[k]\|_F / \|\bar{\mathbf{Q}}[k]\|_F$ per agent n, which compares the corresponding local estimate with the network-wide average $\bar{\mathbf{Q}}[k] := \frac{1}{N}\sum_{n=1}^{N} \mathbf{Q}_n[k]$; and likewise for the $\mathbf{A}_n[k]$. Fig. 4.2 (left) shows that DUNA converges and agents consent on the global matrices $\{\mathbf{Q}, \mathbf{A}\}$ as $k \to \infty$.

To corroborate that DUNA attains the centralized performance, the accelerated proximal gradient algorithm of [97] is employed to solve (P1) after collecting all the per-agent data



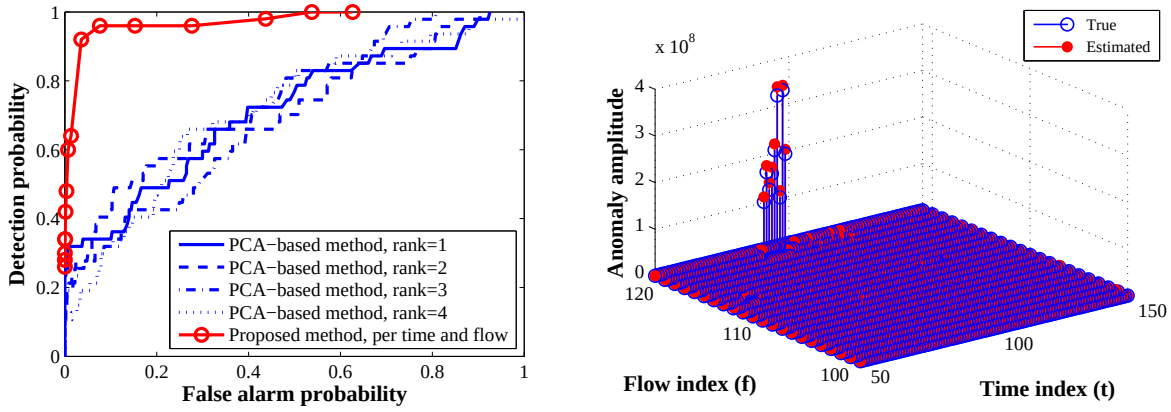
(a) (b)

Figure 4.2: Performance of DUNA. (left) Relative consensus error for representative network agents with $\sigma = 0.01$ and $\pi = 0.01$. (right) Relative estimation error for decentralized and centralized algorithms under various sparsity levels.

in a central processing unit. For both the decentralized and centralized schemes, Fig. 4.2 (right) depicts the evolution of the relative error $\|\hat{\mathbf{A}}[k] - \mathbf{A}_0\|_F / \|\mathbf{A}_0\|_F$ for various sparsity levels, where $\hat{\mathbf{A}}[k] := \bar{\mathbf{A}}[k]$ for DUNA. It is apparent that the decentralized estimator approaches the performance of its centralized counterpart, thus corroborating convergence and global optimality as per Proposition 4.2.

Unveiling Internet2-v1 network anomalies from SNMP measurements. Given the OD flow traffic measurements discussed at the beginning of Section 4.5, the link loads in $\mathbf{Y}$ are obtained through multiplication with the Internet2-v1 routing matrix [2]. Even though $\mathbf{Y}$ is "constructed" here from flow measurements, link loads can be typically acquired from simple network management protocol (SNMP) traces [74]. The available OD flows are a superposition of "clean" and anomalous traffic, i.e., the sum of unknown "ground-truth" low-rank and a sparse matrices $\mathbf{X}_0 + \mathbf{A}_0$ adhering to (4.16) when $\mathbf{R} = \mathbf{I}_F$. Therefore, the proposed algorithms are applied first to obtain a reasonably precise estimate of the "ground-truth" $\{\mathbf{X}_0, \mathbf{A}_0\}$. The estimated $\mathbf{X}_0$ exhibits three dominant singular values, confirming the low-rank property of $\mathbf{X}_0$.

The receiver operation characteristic (ROC) curves in Fig. 4.3 (left) highlight the merits



(a) (b)

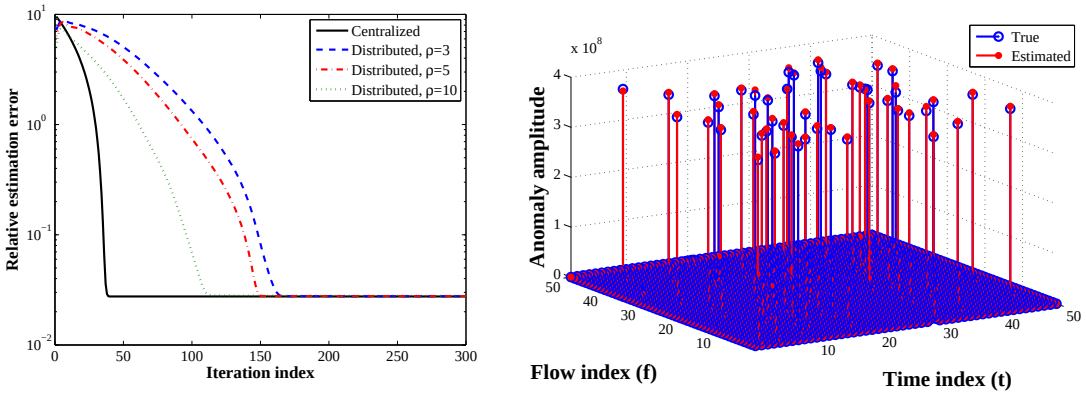
Figure 4.3: Unveiling anomalies from Internet2-v1 SNMP data. (left) ROC curves of the proposed versus the PCA-based method. (right) Amplitude of the true and estimated anomalies for $\rho = 5$, $P_{\text{FA}} = 0.04$ and $P_{\text{D}} = 0.93$.

of (P1) when used to identify Internet2-v1 network anomalies. Even at low false alarm rates of e.g., $P_{\text{FA}} = 0.04$, the anomalies are accurately detected ($P_{\text{D}} = 0.93$). In addition, DUNA consistently outperforms the landmark PCA-based method of [72], and can identify multiple anomalous flows. Fig. 4.3 (right) illustrates the magnitude of the true and estimated anomalies across flows and time.

4.5.2 Robust PCA

Next, the convergence and effectiveness of the proposed DRPCA algorithm is corroborated with the aid of computer simulations. An $F \times T$ data matrix is generated as $\mathbf{Y} = \mathbf{X}_0 + \mathbf{A}_0 + \mathbf{V}$, and the centralized estimator (4.18) is obtained using the ADMM method proposed in [25]. In the network setting, each agent has available $L_n = 19$ rows of $\mathbf{Y}$. Fig. 4.2 (right) is replicated as Fig. 4.4 (left) for the robust PCA problem dealt with here, and for different values of ρ [the assumed upper bound on $\text{rank}(\mathbf{X}_0)$]. It is again apparent that DRPCA converges and approaches the performance of (4.18) as $k \to \infty$.

Unveiling Internet2-v1 network anomalies from Netflow measurements. Suppose a router $n \in \mathcal{N}$ monitors the traffic volume of OD flows to unveil anomalies using e.g., the



(a) (b)

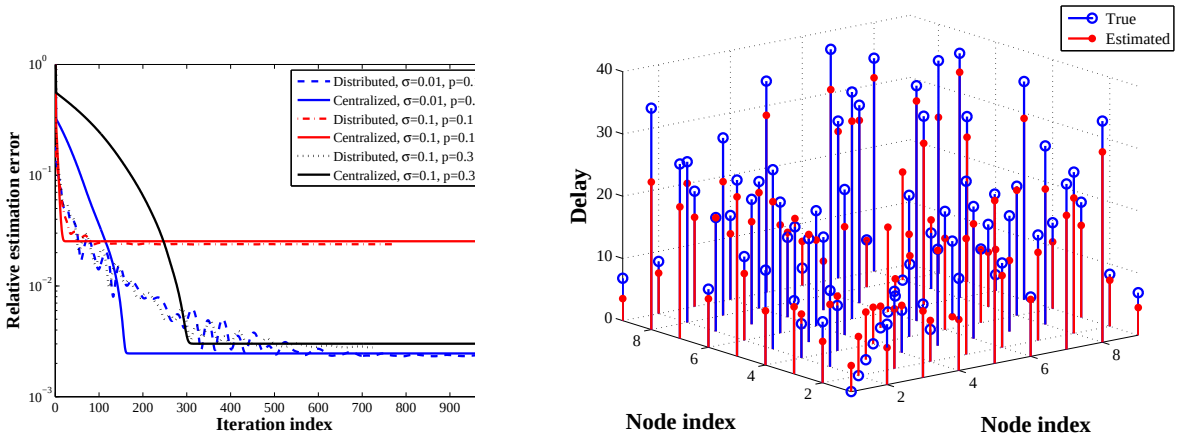
Figure 4.4: Performance of DRPCA. (left) Relative estimation error for decentralized and centralized algorithms under different ρ. (right) Amplitude of true and estimated anomalies using Internet2-v1 network data when $\rho = 5$, $P_{\text{FA}} = 10^{-3}$ and $P_{\text{D}} = 0.98$.

Netflow protocol [3]. Collect the time-series of all OD flows as the rows of the $F \times T$ matrix $\mathbf{Y} = \mathbf{X}_0 + \mathbf{A}_0 + \mathbf{V}$, where $\mathbf{A}_0$ and $\mathbf{V}$ account for anomalies and noise, respectively. As elaborated in Section 4.4.1, the common temporal patterns across flows renders the traffic matrix $\mathbf{X}_0$ low-rank. Owing to the difficulties of measuring the large number of OD flows, in practice only a few entries of $\mathbf{Y}$ are typically available [74], or, link traffic measurements are utilized as in Section 4.5.1 (recall that $L \ll F$). In this example, because the Internet2-v1 network data comprises only $F = 121$ flows, it is assumed that $\Omega = \{1, ..., F\} \times \{1, ..., T\}$.

To better assess performance, large spikes are injected into 1% randomly selected entries of the ground truth-traffic matrix $\mathbf{X}_0$ estimated in Section 4.5.1. The DRPCA algorithm is run on this Internet2-v1 Netflow data to identify the anomalies. The results are depicted in Fig. 4.4 (right). DRPCA accurately identifies the anomalies, achieving $P_{\text{D}} = 0.98$ when $P_{\text{FA}} = 10^{-3}$.

4.5.3 Low-rank matrix completion

In addition to the synthetic data specifications outlined at the beginning of this section, the sampling set Ω is picked uniformly at random, where each entry of the matrix $\mathbf{\Omega}$ is a



(a) (b)

Figure 4.5: Performance of DMC. (left) Relative estimation error for decentralized and centralized algorithms under various noise strengths and percentage of available entries. (right) Predicted and true end-to-end delays of Internet2-v2 network for $p = 0.2$.

Bernoulli random variable taking the value one with probability $1 - p$. Data for the matrix completion problem is thus generated as $\mathcal{P}_\Omega(\mathbf{Y}) = \mathcal{P}_\Omega(\mathbf{X}_0 + \mathbf{V}) = \mathbf{\Omega} \odot (\mathbf{X}_0 + \mathbf{V})$, where $\mathbf{Y}$ is an $L \times T$ matrix with $L = T = 106$. The data available to agent n is $\mathcal{P}_{\Omega_n}(\mathbf{Y}_n)$, which corresponds to a row subset of $\mathcal{P}_\Omega(\mathbf{Y})$.

As with the previous test cases, it is shown first that the DMC algorithm converges to the (centralized) solution of (4.19). To this end, the centralized singular value thresholding algorithm is used to solve (4.19) [27], when all data $\mathcal{P}_\Omega(\mathbf{Y})$ is available for processing. For both the decentralized and centralized schemes, Fig. 4.5 (left) depicts the evolution of the relative error $\|\hat{\mathbf{X}}[k] - \mathbf{X}_0\|_F / \|\mathbf{X}_0\|_F$ for different values of σ (noise strength), and percentage of missing entries (controlled by p). Regardless of the values of σ and p, the error trends clearly show the convergent behavior of the DMC algorithm and corroborate Proposition 4.2. Interestingly, for small noise levels where the estimation error approaches zero, the decentralized estimator recovers $\mathbf{X}_0$ almost *exactly*.

Internet2-v2 network latency prediction. End-to-end network latency information is critical towards enforcing quality-of-service constraints in many Internet applications. How-

ever, probing all pairwise delays becomes infeasible in large-scale networks. If one collects the end-to-end latencies of source-sink pairs (i, j) in a delay matrix $\mathbf{X} := [x_{i,j}] \in \mathbb{R}^{N \times N}$, strong dependencies among path delays render $\mathbf{X}$ low-rank [79]. This is mainly because the paths with nearby end nodes often overlap and share common bottleneck links. This property of $\mathbf{X}$ along with the decentralized-processing requirements of large-scale networks, motivates well the adoption of the DMC algorithm for networkwide path latency prediction. Given the n-th row of $\mathbf{X}$ is partially available to agent n, the goal is to impute the missing delays through agent collaboration.

The DMC algorithm is tested here using the real path latency data collected from the operation of Internet2-v2. Spectral analysis of $\mathbf{X}_0$ reveals that the first four singular values are markedly dominant, demonstrating that $\mathbf{X}_0$ is low rank. A fraction of the entries in $\mathbf{X}_0$ are purposely dropped to yield an incomplete delay matrix $\mathcal{P}_\Omega(\mathbf{X}_0)$. After running the DMC algorithm, the true and predicted latencies are depicted in Fig. 4.5 (right) (for 20% missing data). The relative prediction error is around 10%.

4.5.4 Comparison with centralized processing

As a means of offering additional design insights, this section presents performance tradeoffs that become relevant as the network size increases. Specifically, comparisons in terms of running time are carried out with respect to the centralized processing benchmark (P1). Throughout, a network modeled as a square grid (uniform lattice) with K agents per row/column (i.e., $N = K^2$ total agents) is adopted. The lattice exhibits a more uniform degree distribution than the random geometric graph, since the only possible degree values are $\{2, 3, 4\}$, regardless of N. The DRPCA algorithm is tested with data generated as outlined in Section 4.5.2. Relevant parameter choices are $r = 3$, $\rho = 5$, $\pi = 0.01$.

To gauge running times as the network grows, consider a fixed size data matrix $\mathbf{Y} \in \mathbb{R}^{L \times T}$ with $L = T = 2,500$. The rows of $\mathbf{Y}$ are split among agents so that each agent has available L/K^2 rows. This way comparisons can be carried out on equal footing because even when network sizes differ, the same network-wide problem is solved.

The evolution of the relative estimation error for the DRPCA algorithm under various

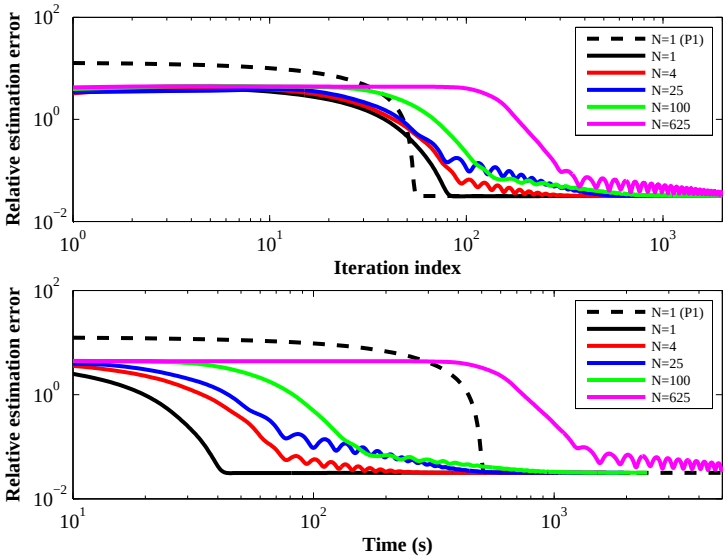


Figure 4.6: Relative DRPCA estimation error versus iteration index and CPU time, under different network sizes when $\rho = 5$, $\sigma = 0.01$, and $\pi = 0.01$.

network sizes ($N = K^2$) is depicted in Fig. 4.6. The error is plotted both against iteration index k and CPU time. The centralized benchmark offered by the ADMM-based algorithm in [25], is adopted to solve (P1) for the robust PCA special case. Convergence time of the decentralized algorithm is competitive with its centralized counterpart for small-size networks ($N \leq 100$ agents). It is apparent that as the network size increases, convergence becomes slower as local data need to percolate the entire (larger) network, under the constraint of single-hop message exchanges. It is worth noting that the results in Fig. 4.6 were obtained using simple (by no means performance-optimized) Matlab scripts for Algorithm 7. Naturally, there is considerable room for improvement in terms of software implementation.

Remark 4.6 (In-network versus centralized) *Albeit a fusion center (FC)-based solver may incur less run time, there are well-documented advantages favoring decentralized algorithms when it comes to signal and information processing over large-scale networks; see e.g., [16]. Three design considerations advocating decentralized in-network over FC-based implementations are: i) robustness against single-agent (FC) failure; ii) reduction of noise affecting inter-agent exchanges is more effective when communicating local estimates rather*

than raw data with the FC [106]; and iii) higher communication overhead is incurred by FC-based schemes when agents implement time-adaptive (online) signal processing algorithms. Of course, all these factors are application dependent and it is up to the network operator to adopt the algorithm that best suits the given specifications and resource constraints.

4.6 Concluding Summary

A framework for decentralized sparsity-regularized rank minimization is developed in this paper, that is suitable for (un)supervised inference tasks carried over networks. By resorting to the ADMM and an alternative characterization of the nuclear norm (originally proposed to relax matrix rank constraints in semidefinite programs), the novel decentralized algorithm, if convergent, provably attains the performance of the centralized benchmark. Fundamental problems such as in-network compressed sensing, matrix completion, and principal component pursuit, are all captured under the same umbrella.

With regards to applications, focus is placed on key network health monitoring tasks geared to obtaining full yet succinct representation of network metrics, such as end-to-end path delays, as well as prompt and accurate identification of network anomalies from possibly partial and corrupted measurements. Numerical tests with synthetic and real network data drawn from these application domains corroborate the effectiveness and convergence of the novel decentralized algorithm, and its centralized performance guarantees. Regarding network anomaly identification, the formulation here 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 markedly outperforms the sparsity-agnostic workhorse PCA-based method of [72].

An interesting future direction is to devise and analyze the performance of decentralized *online* algorithms for sparsity-regularized rank minimization, capable of processing network data in real time. In this context, exciting possibilities emerge by bringing together recent advances in online rank-minimization [96, 118], and decentralized adaptive algorithms developed for estimation and tracking over networks [106, 126]. In addition, it is of interest to rigorously establish convergence of Algorithm 5. Such results could markedly broaden the

applicability of ADMM for large-scale optimization over networks, even in the presence of non-convex but highly structured and separable cost functions.

Chapter 5

Online Sparsity-Regularized Rank Minimization: Applications to Tracking Network Anomalies

5.1 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 QoS provisioning of the end users. These so-termed *traffic volume anomalies* could be due to unexpected failures in networking equipment, cyberattacks (e.g., denial of service (DoS) attacks), or, intruders which hijack the network services [137]. Unveiling such anomalies in a promptly manner is a crucial monitoring task towards engineering network traffic. This is a challenging task however, since the available data are usually high-dimensional, noisy and possibly incomplete link-load measurements, which are the superposition of *unobservable* OD flows. Several studies have experimentally demonstrated the low intrinsic dimensionality of the nominal traffic subspace, that is, the intuitive *low-rank* property of the traffic matrix in the absence of anomalies, which is mainly due to common temporal patterns across OD flows, and periodic behavior across time [74, 160]. Exploiting the low-rank structure of

the anomaly-free traffic matrix, a landmark principal component analysis (PCA)-based method was put forth in [72] to identify network anomalies; see also [8] for a distributed implementation. A limitation of the algorithm in [72] is that it cannot identify multiple anomalous flows. Most importantly, [72] has 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.

Capitalizing on the low-rank property of the traffic matrix and the sparsity of the anomalies, the fresh look advocated here permeates benefits from rank minimization [25–27], and compressive sampling [29, 30], to perform *dynamic anomalography*. The aim is to construct a map of network *anomalies* in real time, that offers a succinct depiction of the network 'health state' across both the flow and time dimensions (Section 5.2). Different from the *batch* centralized and distributed anomalography algorithms in [97] and [95], the focus here is on devising *online* (adaptive) algorithms that are capable of efficiently processing link measurements and track network anomalies 'on the fly'; see also [135] for a 'model-free' approach that relies on the kernel recursive LS (RLS) algorithm. Online monitoring algorithms are attractive for operation in dynamic network environments, since they can cope with traffic nonstationarities arising due to routing changes and missing data. Accordingly, the novel online estimator entails an exponentially-weighted least-squares (LS) cost regularized with the sparsity-promoting ℓ_1-norm of the anomalies, and the nuclear norm of the nominal traffic matrix. After recasting the non-separable nuclear norm into a form amenable to online optimization (Section 5.3.1), a real-time algorithm for dynamic anomalography is developed in Section 5.4 based on alternating minimization. Each time a new datum is acquired, anomaly estimates are formed via the least-absolute shrinkage and selection operator (Lasso), e.g, [56, p. 68], and the low-rank nominal traffic subspace is refined using RLS [131]. Convergence analysis is provided under simplifying technical assumptions in Section 5.4.2. For situations where reducing computational complexity is critical, an online stochastic gradient algorithm based on Nesterov's accelaration technique [14, 110] is developed as well (Section 5.5.1). The possibility of implementing the anomaly trackers in a distributed fashion is further outlined in Section 5.5.2, where several directions for future

research are also delineated.

Extensive numerical tests involving both synthetic and real network data corroborate the effectiveness of the proposed algorithms in unveiling network anomalies, as well as their tracking capabilities when traffic routes are slowly time-varying, and the network monitoring station acquires incomplete link traffic measurements (Section 5.6). Different from [158] which employs a two-step batch procedure to learn the nominal traffic subspace first, and then unveil anomalies via ℓ_1-norm minimization, the approach here estimates both quantities jointly and attains better performance as illustrated in Section 5.6.2. Concluding remarks are given in Section 5.7, while most technical details relevant to the convergence proof in Section 5.4.3 are deferred to the Appendix.

5.2 Modeling Preliminaries and Problem Statement

Consider a backbone Internet protocol (IP) network naturally modeled as a directed graph $G(\mathcal{N}, \mathcal{L})$, 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 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 adopted here, that is, a given flow's traffic is carried through multiple links connecting the corresponding source-destination pair along a single path. Let $r_{l,f}$, $l \in \mathcal{L}$, $f \in \mathcal{F}$, denote the flow to link assignments (routing), which take the value one whenever flow f is carried over link l, and zero otherwise. Unless otherwise stated, the routing matrix $\mathbf{R} := [r_{l,f}] \in \{0,1\}^{L \times F}$ is assumed fixed and given. Likewise, let $z_{f,t}$ denote the unknown traffic rate of flow f at time t, measured in e.g., Mbps. At any given time instant t, the traffic carried over link l is then the superposition of the flow rates routed through link l, i.e., $\sum_{f \in \mathcal{F}} r_{l,f} z_{f,t}$.

It is not uncommon for some of the flow rates to experience unusual abrupt changes. These so-termed *traffic volume anomalies* are typically due to unexpected network failures, or cyberattacks (e.g., DoS attacks) which aim at compromising the services offered by the network [137]. Let $a_{f,t}$ denote the unknown traffic volume anomaly of flow f at time t. In

the presence of anomalous flows, the measured traffic carried by link l over a time horizon $t \in [1, T]$ is then given by

$$y_{l,t} = \sum_{f \in \mathcal{F}} r_{l,f}(z_{f,t} + a_{f,t}) + v_{l,t},\ t = 1, ..., T \tag{5.1}$$

where the noise variables $v_{l,t}$ account for measurement errors and unmodeled dynamics.

In IP networks, traffic volume can be readily measured on a per-link basis using off-the-shelf tools such as the simple network management protocol (SNMP) supported by most routers. Missing entries in the link-level measurements $y_{l,t}$ may however 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 [119]. To model missing link measurements, collect the tuples (l, t) associated with the available observations $y_{l,t}$ in the set $\Omega \in [1, 2, ..., L] \times [1, 2, ..., T]$. Introducing the matrices $\mathbf{Y} := [y_{l,t}], \mathbf{V} := [v_{l,t}] \in \mathbb{R}^{L \times T}$, and $\mathbf{Z} := [z_{f,t}], \mathbf{A} := [a_{f,t}] \in \mathbb{R}^{F \times T}$, the (possibly incomplete) set of measurements in (5.1) can be expressed in compact matrix form as

$$\mathcal{P}_\Omega(\mathbf{Y}) = \mathcal{P}_\Omega(\mathbf{R}\,(\mathbf{Z} + \mathbf{A}) + \mathbf{V}) \tag{5.2}$$

where the sampling operator $\mathcal{P}_\Omega(.)$ sets the entries of its matrix argument not in Ω to zero, and keeps the rest unchanged. Matrix $\mathbf{Z}$ contains the nominal traffic flows over the time horizon of interest. Common temporal patterns among the traffic flows in addition to their periodic behavior, render most rows (respectively columns) of $\mathbf{Z}$ linearly dependent, and thus $\mathbf{Z}$ typically has low rank. This intuitive property has been extensively validated with real network data; see e.g, [74]. Anomalies in $\mathbf{A}$ are expected to occur sporadically over time, and last shortly 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).

Given measurements $\mathcal{P}_\Omega(\mathbf{Y})$ adhering to (5.2) and the binary-valued routing matrix $\mathbf{R}$, the main goal of this paper is to accurately estimate the anomaly matrix $\mathbf{A}$, by capitalizing on the sparsity of $\mathbf{A}$ and the low-rank property of $\mathbf{Z}$. Special focus will be placed on

devising online (adaptive) algorithms that are capable of efficiently processing link measurements and tracking network anomalies in real time. This critical monitoring task is termed *dynamic anomalography*, and 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 f-th flow at time t is deemed anomalous, otherwise it is healthy. 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.

5.3 Unveiling Anomalies via Sparsity and Low Rank

Consider the nominal link-count traffic matrix $\mathbf{X} := \mathbf{RZ}$, which inherits the low-rank property from $\mathbf{Z}$. Since the primary goal is to recover $\mathbf{A}$, the following observation model

$$\mathcal{P}_\Omega(\mathbf{Y}) = \mathcal{P}_\Omega(\mathbf{X} + \mathbf{RA} + \mathbf{V}) \tag{5.3}$$

can be adopted instead of (5.2). A natural estimator leveraging the low rank property of $\mathbf{X}$ and the sparsity of $\mathbf{A}$ will be sought next. The idea is to fit the incomplete data $\mathcal{P}_\Omega(\mathbf{Y})$ to the model $\mathbf{X} + \mathbf{RA}$ in the least-squares (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 [37, 109]. Typically, the nuclear norm $\|\mathbf{X}\|_*$ and the ℓ_1-norm $\|\mathbf{A}\|_1$ are adopted as surrogates, since they are the closest *convex* approximants to rank($\mathbf{X}$) and $\|\mathbf{A}\|_0$, respectively [29, 117, 141]. Accordingly, one solves

$$\text{(P1)} \qquad \min_{\{\mathbf{X},\mathbf{A}\}} \frac{1}{2}\|\mathcal{P}_\Omega(\mathbf{Y} - \mathbf{X} - \mathbf{RA})\|_F^2 + \lambda_*\|\mathbf{X}\|_* + \lambda_1\|\mathbf{A}\|_1 \tag{5.4}$$

where $\lambda_*, \lambda_1 \geq 0$ are rank- and sparsity-controlling parameters. When an estimate $\hat{\sigma}_v^2$ of the noise variance is available, guidelines for selecting λ_* and λ_1 have been proposed in [163].

Being convex (P1) is appealing, and it yields reliable performance when full data are available, i.e., $\Omega = \emptyset$ [97]. In the presence of missing data, one has to ensure that the sampled subset of links provides sufficient information to identify anomalous flows. Intuitively, for

high estimation accuracy each flow must traverse sufficiently many links, whereas network links should not be overloaded by too many flows. These properties typically hold for large-scale networks with distant OD node pairs, and routing paths that are sufficiently 'spread-out.' Developing identifiability conditions when link measurements are incomplete is an open problem, and constitutes an interesting future research direction.

Model (5.3) and its estimator (P1) are quite general, as discussed in the ensuing remark.

Remark 5.1 (Subsumed paradigms) *When there is no missing data and* $\mathbf{X} = \mathbf{0}_{L\times T}$*, one is left with an under-determined sparse signal recovery problem typically encountered with compressive sampling (CS); see e.g., [29] and the tutorial account [30]. The decomposition* $\mathbf{Y} = \mathbf{X} + \mathbf{A}$ *corresponds to principal component pursuit (PCP), also referred to as robust principal component analysis (PCA) [25, 33]. PCP was adopted for network anomaly detection using flow (not link traffic) measurements in [3]. For the idealized noise-free setting (*$\mathbf{V} = \mathbf{0}_{L\times T}$*), sufficient conditions for exact recovery are available for both of the aforementioned special cases [25, 29, 33]. However, the superposition of a low-rank plus a compressed sparse matrix in (5.3) further challenges identifiability of* $\{\mathbf{X}, \mathbf{A}\}$*; see [97] 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 [36]. 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 adopted for low-rank matrix completion, to impute the missing entries of a low-rank matrix observed in noise, i.e.,* $\mathcal{P}_\Omega(\mathbf{Y}) = \mathcal{P}_\Omega(\mathbf{X} + \mathbf{V})$ *[26].*

Albeit convex, (P1) is a non-smooth optimization problem (both the nuclear and ℓ_1-norms are not differentiable at the origin). In addition, scalable algorithms to unveil anomalies in large-scale networks should effectively overcome the following challenges: (c1) the problem size can easily become quite large, since the number of optimization variables is $(L+F)T$; (c2) existing iterative solvers for (P1) typically rely on costly SVD computations per iteration; see e.g., [97]; and (c3) different from the Frobenius and ℓ_1-norms, (column-wise) nonseparability of the nuclear-norm challenges online processing when new columns of

$\mathcal{P}_\Omega(\mathbf{Y})$ arrive sequentially in time. In the remainder of this section, the 'big data' challenges (c1) and (c2) are dealt with to arrive at an efficient batch algorithm for anomalography. Tracking network anomalies is the main subject of Section 5.4.

To address (c1) and reduce the computational complexity and memory storage requirements of the algorithms sought, it is henceforth assumed that an upper bound $\rho \geq \text{rank}(\hat{\mathbf{X}})$ is a priori available [$\hat{\mathbf{X}}$ is the estimate obtained via (P1)]. As argued next, the smaller the value of ρ, the more efficient the algorithm becomes. Small values of ρ are well motivated due to the low intrinsic dimensionality of network flows. For instance, experiments with Internet-2 network data [2] show that $\rho = 5$ suffices [72]; see also [74]. Because $\text{rank}(\hat{\mathbf{X}}) \leq \rho$, (P1)'s search space is effectively reduced and one can factorize the decision variable as $\mathbf{X} = \mathbf{L}\mathbf{Q}'$, where $\mathbf{L}$ and $\mathbf{Q}$ are $L \times \rho$ and $T \times \rho$ matrices, respectively. It is possible to interpret the columns of $\mathbf{X}$ (viewed as points in $\mathbb{R}^L$) as belonging to a low-rank 'nominal traffic subspace', spanned by the columns of $\mathbf{L}$. The rows of $\mathbf{Q}$ are thus the projections of the columns of $\mathbf{X}$ onto the traffic subspace.

Adopting this reparametrization of $\mathbf{X}$ in (P1), and defining $r(\mathbf{L}, \mathbf{Q}, \mathbf{A}) := \frac{1}{2}\|\mathcal{P}_\Omega(\mathbf{Y} - \mathbf{L}\mathbf{Q} - \mathbf{R}\mathbf{A})\|_F^2$, one arrives at an equivalent optimization problem

$$\text{(P2)} \qquad \min_{\{\mathbf{L},\mathbf{Q},\mathbf{A}\}} r(\mathbf{L}, \mathbf{Q}, \mathbf{A}) + \lambda_* \|\mathbf{L}\mathbf{Q}'\|_* + \lambda_1 \|\mathbf{A}\|_1$$

which is non-convex due to the bilinear terms $\mathbf{L}\mathbf{Q}'$. The number of variables is reduced from $(L+F)T$ in (P1), to $\rho(L+T)+FT$ in (P2). The savings can be significant when ρ is small, and both L and T are large. Note that the dominant FT-term in the variable count of (P2) is due to $\mathbf{A}$, which is sparse and can be efficiently handled even when both F and T are large.

5.3.1 A separable low-rank regularization

To address (c2) [along with (c3) as it will become clear in Section 5.4], consider the following alternative characterization of the nuclear norm [117, 118]

$$\|\mathbf{X}\|_* := \min_{\{\mathbf{L},\mathbf{Q}\}} \quad \frac{1}{2}\left\{\|\mathbf{L}\|_F^2 + \|\mathbf{Q}\|_F^2\right\}, \quad \text{s. t.} \quad \mathbf{X} = \mathbf{L}\mathbf{Q}'. \tag{5.5}$$

The optimization (5.5) is over all possible bilinear factorizations of $\mathbf{X}$, so that the number of columns ρ of $\mathbf{L}$ and $\mathbf{Q}$ is also a variable. Leveraging (5.5), the following reformulation of (P2) provides an important first step towards obtaining an online algorithm:

$$\text{(P3)} \qquad \min_{\{\mathbf{L},\mathbf{Q},\mathbf{A}\}} r(\mathbf{L},\mathbf{Q},\mathbf{A}) + \frac{\lambda_*}{2}\left\{\|\mathbf{L}\|_F^2 + \|\mathbf{Q}\|_F^2\right\} + \lambda_1\|\mathbf{A}\|_1.$$

As asserted in [95, Lemma 1], adopting the separable Frobenius-norm regularization in (P3) comes with no loss of optimality relative to (P1), provided $\rho \geq \text{rank}(\hat{\mathbf{X}})$. By finding the global minimum of (P3) [which could have considerably less variables than (P1)], one can recover the optimal solution of (P1). However, since (P3) is non-convex, it may have stationary points which need not be globally optimum. Interestingly, the next proposition shows that under relatively mild assumptions on $\text{rank}(\hat{\mathbf{X}})$ and the noise variance, every stationary point of (P3) is globally optimum for (P1). For a proof, see [95, Appendix].

Proposition 5.1 *Let* $\{\bar{\mathbf{L}},\bar{\mathbf{Q}},\bar{\mathbf{A}}\}$ *be a stationary point of (P3). If* $\|\mathcal{P}_\Omega(\mathbf{Y}-\bar{\mathbf{L}}\bar{\mathbf{Q}}'-\mathbf{R}\bar{\mathbf{A}})\| \leq \lambda_*$*, then* $\{\hat{\mathbf{X}} := \bar{\mathbf{L}}\bar{\mathbf{Q}}', \hat{\mathbf{A}} = \bar{\mathbf{A}}\}$ *is the globally optimal solution of (P1).*

The qualification condition $\|\mathcal{P}_\Omega(\mathbf{Y}-\bar{\mathbf{L}}\bar{\mathbf{Q}}'-\mathbf{R}\bar{\mathbf{A}})\| \leq \lambda_*$ captures tacitly the role of ρ. In particular, for sufficiently small ρ the residual $\|\mathcal{P}_\Omega(\mathbf{Y}-\bar{\mathbf{L}}\bar{\mathbf{Q}}'-\mathbf{R}\bar{\mathbf{A}})\|$ becomes large and consequently the condition is violated [unless λ_* is large enough, in which case a sufficiently low-rank solution to (P1) is expected]. The condition on the residual also implicitly enforces $\text{rank}(\hat{\mathbf{X}}) \leq \rho$, which is necessary for the equivalence between (P1) and (P3). Note also that selecting a large value of ρ does not ensure satisfaction of the condition in Proposition 5.1. In fact, other factors such as the noise variance and routing matrix structure are critical as well.

5.3.2 Batch block coordinate-descent algorithm

The block coordinate-descent (BCD) algorithm is adopted here to solve the batch non-convex optimization problem (P3). BCD is an iterative method which has been shown efficient in tackling large-scale optimization problems encountered with various statistical inference tasks, see e.g., [17]. The proposed solver entails an iterative procedure comprising three steps per iteration $k = 1, 2, \ldots$

[S1] Update the anomaly map:

$$\mathbf{A}[k+1] = \arg\min_{\mathbf{A}} \left[r(\mathbf{L}[k], \mathbf{Q}[k], \mathbf{A}) + \lambda_1 \|\mathbf{A}\|_1 \right].$$

[S2] Update the nominal traffic subspace:

$$\mathbf{L}[k+1] = \arg\min_{\mathbf{L}} \left[r(\mathbf{L}, \mathbf{Q}[k], \mathbf{A}[k+1]) + \frac{\lambda_*}{2} \|\mathbf{L}\|_F^2 \right].$$

[S3] Update the projection coefficients:

$$\mathbf{Q}[k+1] = \arg\min_{\mathbf{Q}} \left[r(\mathbf{L}[k+1], \mathbf{Q}, \mathbf{A}[k+1]) + \frac{\lambda_*}{2} \|\mathbf{Q}\|_F^2 \right].$$

To update each of the variable groups, the cost of (P3) is minimized while fixing the rest of the variables to their most up-to-date values. The minimization in [S1] decomposes over the columns of $\mathbf{A} := [\mathbf{a}_1, \ldots, \mathbf{a}_T]$. At iteration k, these columns are updated in parallel via Lasso

$$\mathbf{a}_t[k+1] = \arg\min_{\mathbf{a}} \left[\frac{1}{2} \|\mathbf{\Omega}_t(\mathbf{y}_t - \mathbf{L}[k]\mathbf{q}_t[k] - \mathbf{R}\mathbf{a})\|_2^2 + \lambda_1 \|\mathbf{a}\|_1 \right], \quad t = 1, \ldots, T \tag{5.6}$$

where $\mathbf{y}_t$ and $\mathbf{q}_t[k]$ respectively denote the t-th column of $\mathbf{Y}$ and $\mathbf{Q}'[k]$, while the diagonal matrix $\mathbf{\Omega}_t \in \mathbb{R}^{L \times L}$ contains a one on its l-th diagonal entry if $y_{l,t}$ is observed, and a zero otherwise. To keep computational complexity at a minimum, in practice each iteration of the proposed algorithm minimizes (5.6) inexactly. This is achieved for each $t = 1, \ldots, T$, by performing a single pass of the cyclic coordinate-descent algorithm in [56, p. 92] over each one of the F scalar entries in $\mathbf{a}_t[k+1]$; see Algorithm 9 for the resulting iterations, and Appendix for further details. As shown at the end of this section, this inexact minimization suffices to claim convergence to a stationary point of (P3).

Similarly, in [S2] and [S3] the minimizations that give rise to $\mathbf{L}[k+1]$ and $\mathbf{Q}[k+1]$ are separable over their respective rows. For instance, the l-th row $\mathbf{l}_l'$ of the traffic subspace matrix $\mathbf{L} := [\mathbf{l}_1, \ldots, \mathbf{l}_L]'$ is updated as the solution of the following ridge-regression problem

$$\mathbf{l}_l[k+1] = \arg\min_{\mathbf{l}} \left[\frac{1}{2} \|((\mathbf{y}_l^r)' - \mathbf{l}'\mathbf{Q}'[k] - (\mathbf{r}_l^r)'\mathbf{A}[k+1])\mathbf{\Omega}_l^r\|_2^2 + \frac{\lambda_*}{2} \|\mathbf{l}\|_2^2 \right] \tag{5.7}$$

Algorithm 9 : Batch BCD algorithm for unveiling network anomalies

input $\mathcal{P}_\Omega(\mathbf{Y}), \Omega, \mathbf{R}, \lambda_*$, and λ_1.

initialize $\mathbf{L}[1]$ and $\mathbf{Q}[1]$ at random.

for $k = 1, 2, \ldots$ **do**

[S1] Update the anomaly map:

for $f = 1, \ldots, F$ **do**

$$\tilde{\mathbf{y}}_t^{(-f)}[k+1] = \mathbf{\Omega}_t(\mathbf{y}_t - \mathbf{L}[k]\mathbf{q}_t[k] - \textstyle\sum_{f'=1}^{f-1} \mathbf{r}_{f'} a_{f',t}[k+1] - \sum_{f'=f+1}^{F} \mathbf{r}_{f'} a_{f',t}[k]), \quad t = 1, \ldots, T.$$

$$a_{f,t}[k+1] = \text{sign}(\mathbf{r}_f' \tilde{\mathbf{y}}_t^{(-f)}[k+1]) \left[|\mathbf{r}_f' \tilde{\mathbf{y}}_t^{(-f)}[k+1]| - \lambda_1 \right]_+ / \|\mathbf{\Omega}_t \mathbf{r}_f\|_2, \quad t = 1, \ldots, T.$$

end for

$$\mathbf{A}[k+1] = \left[[a_{1,1}[k+1], \ldots, a_{F,1}[k+1]]', \ldots, [a_{1,T}[k+1], \ldots, a_{F,T}[k+1]]' \right].$$

[S2] Update the nominal traffic subspace:

$$\mathbf{l}_l[k+1] = \left(\lambda_* \mathbf{I}_\rho + \mathbf{Q}'[k] \mathbf{\Omega}_l^r \mathbf{Q}[k]\right)^{-1} \mathbf{Q}'[k] \mathbf{\Omega}_l^r (\mathbf{y}_l^r - \mathbf{A}'[k+1] \mathbf{r}_l^r), \quad l = 1, \ldots, L.$$

$$\mathbf{L}[k+1] = [\mathbf{l}_1[k+1], \ldots, \mathbf{l}_L[k+1]]'.$$

[S3] Update the projection coefficients:

$$\mathbf{q}_t[k+1] = \left(\lambda_* \mathbf{I}_\rho + \mathbf{L}'[k+1] \mathbf{\Omega}_t \mathbf{L}[k+1]\right)^{-1} \mathbf{L}'[k+1] \mathbf{\Omega}_t (\mathbf{y}_t - \mathbf{R}\mathbf{a}_t[k+1]), \quad t = 1, \ldots, T.$$

$$\mathbf{Q}[k+1] = [\mathbf{q}_1[k+1], \ldots, \mathbf{q}_T[k+1]]'.$$

end for

return $\hat{\mathbf{A}} := \mathbf{A}[\infty]$ and $\hat{\mathbf{X}} := \mathbf{L}[\infty]\mathbf{Q}'[\infty]$.

where $(\mathbf{y}_l^r)'$ and $(\mathbf{r}_l^r)'$ represent the l-th row of $\mathbf{Y}$ and $\mathbf{R}$, respectively. The t-th diagonal entry of the diagonal matrix $\mathbf{\Omega}_l^r \in \mathbb{R}^{T \times T}$ is an indicator variable testing whether measurement $y_{l,t}$ is available. Because (5.7) is an unconstrained convex quadratic program, the first-order optimality condition yields the closed-form solution tabulated under Algorithm 9. A similar regularized LS problem yields $\mathbf{q}_t[k+1]$, $t = 1, \ldots, T$; see Algorithm 9 for the details and a description of the overall BCD solver. The novel batch scheme for unveiling network anomalies is less complex computationally than the accelerated proximal gradient algorithm in [97], since Algorithm 9's iterations are devoid of SVD computations. Different from [97], Algorithm 9 can also accommodate missing link measurements.

Despite being non-convex and non-differentiable, (P3) has favorable structure which facilitates convergence of the iterates generated by Algorithm 9. Specifically, the resulting cost is convex in each block variable when the rest are fixed. The non-smooth ℓ_1-norm

is also separable over the entries of its matrix argument. Accordingly, [144, Theorem 5.1] guarantees convergence of the BCD algorithm to a stationary point of (P3). This result together with Proposition 5.1 establishes the next claim.

Proposition 5.2 *If a subsequence* $\{\mathbf{X}[k] := \mathbf{L}[k]\mathbf{Q}'[k], \mathbf{A}[k]\}$ *of iterates generated by Algorithm 9 satisfies* $\|\mathcal{P}_\Omega(\mathbf{Y} - \mathbf{X}[k] - \mathbf{R}\mathbf{A}[k])\| \leq \lambda_*$, *then it converges to the optimal solution set of (P1) as* $k \to \infty$.

In practice, it is desirable to monitor anomalies in real time and accomodate time-varying traffic routes. These reasons motivate devising algorithms for *dynamic* anomalography, the subject dealt with next.

5.4 Dynamic Anomalography

Monitoring of large-scale IP networks necessitates collecting massive amounts 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 or network infrastructure failures. To account for slowly time-varing routing tables, let $\mathbf{R}_t \in \mathbb{R}^{L\times F}$ denote the routing matrix at time t[1]. In this dynamic setting, the partially observed link counts at time t adhere to [cf. (5.3)]

$$\mathcal{P}_{\Omega_t}(\mathbf{y}_t) = \mathcal{P}_{\Omega_t}(\mathbf{x}_t + \mathbf{R}_t\mathbf{a}_t + \mathbf{v}_t),\ t = 1, 2, \ldots \tag{5.8}$$

where the link-level traffic $\mathbf{x}_t := \mathbf{R}_t\mathbf{z}_t$, for $\mathbf{z}_t$ from the (low-dimensional) traffic subspace. 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 network-level traffic vectors $\{\mathbf{z}_t\}$ live

[1]Fixed size routing matrices $\mathbf{R}_t$ are considered here for convenience, where L and F correspond to upper bounds on the number of physical links and flows transported by the network, respectively. If at time t some links are not used at all, or, less than F flows are present, the corresponding rows and columns of $\mathbf{R}_t$ will be identically zero.

in a low-dimensional subspace, the same may not be true for the link-level traffic $\{\mathbf{x}_t\}$ 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 instance, data collected from the operation of Internet-2 network reveals that only a few rows of $\mathbf{R}_t$ change per week [2]. It is thus safe to assume that $\{\mathbf{x}_t\}$ still lies in a low-dimensional subspace, and exploit the temporal correlations of the observations to identify the anomalies.

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 $\{\mathcal{P}_{\Omega_\tau}(\mathbf{y}_\tau), \Omega_\tau\}_{\tau=1}^t$, naturally placing more importance to recent measurements. To this end, one possible adaptive counterpart to (P3) is the exponentially-weighted LS estimator found by minimizing the empirical cost

$$\min_{\{\mathbf{L},\mathbf{Q},\mathbf{A}\}} \sum_{\tau=1}^{t} \beta^{t-\tau} \left[\frac{1}{2} \|\mathcal{P}_{\Omega_\tau}(\mathbf{y}_\tau - \mathbf{L}\mathbf{q}_\tau - \mathbf{R}_\tau \mathbf{a}_\tau)\|_2^2 + \frac{\lambda_*}{2\sum_{u=1}^{t} \beta^{t-u}} \|\mathbf{L}\|_F^2 + \frac{\lambda_*}{2} \|\mathbf{q}_\tau\|_2^2 + \lambda_1 \|\mathbf{a}_\tau\|_1 \right] \tag{5.9}$$

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 non-stationary environments. In the case of static routing ($\mathbf{R}_t = \mathbf{R}, t = 1, 2, \ldots$) and infinite memory ($\beta = 1$), the formulation (5.9) coincides with the batch estimator (P3). This is the reason for the time-varying factor weighting $\|\mathbf{L}\|_F^2$.

5.4.1 Tracking network anomalies

Towards deriving a real-time, computationally efficient, and recursive solver of (5.9), an alternating minimization method is adopted in which iteration k coincides with the time scale t of data acquisition. A justification in terms of minimizing a suitable approximate cost function is discussed in detail in Section 5.4.2. Per time instant t, a new datum

$\{\mathcal{P}_{\Omega_t}(\mathbf{y}_t), \Omega_t\}$ is drawn and $\{\mathbf{q}_t, \mathbf{a}_t\}$ are jointly estimated via

$$\{\mathbf{q}[t], \mathbf{a}[t]\} = \arg\min_{\{\mathbf{q},\mathbf{a}\}} \left[\frac{1}{2}\|\mathcal{P}_{\Omega_t}(\mathbf{y}_t - \mathbf{L}[t-1]\mathbf{q} - \mathbf{R}_t\mathbf{a})\|_2^2 + \frac{\lambda_*}{2}\|\mathbf{q}\|_2^2 + \lambda_1\|\mathbf{a}\|_1\right]. \quad (5.10)$$

It turns out that (5.10) can be efficiently solved. Fixing $\mathbf{a}$ to carry out the minimization with respect to $\mathbf{q}$ first, one is left with an ℓ_2-norm regularized LS (ridge-regression) problem

$$\begin{aligned}\mathbf{q}[t] &= \arg\min_{\mathbf{q}} \left[\frac{1}{2}\|\mathcal{P}_{\Omega_t}(\mathbf{y}_t - \mathbf{L}[t-1]\mathbf{q} - \mathbf{R}_t\mathbf{a})\|_2^2 + \frac{\lambda_*}{2}\|\mathbf{q}\|^2\right] \\ &= \left(\lambda_*\mathbf{I}_\rho + \mathbf{L}'[t-1]\mathbf{\Omega}_t\mathbf{L}[t-1]\right)^{-1}\mathbf{L}'[t-1]\mathcal{P}_{\Omega_t}(\mathbf{y}_t - \mathbf{R}_t\mathbf{a}). \quad (5.11)\end{aligned}$$

Note that $\mathbf{q}[t]$ is an affine function of $\mathbf{a}$, and the update rule for $\mathbf{q}[t]$ is not well defined until $\mathbf{a}$ is replaced with $\mathbf{a}[t]$. Towards obtaining an expression for $\mathbf{a}[t]$, define $\mathbf{D}[t] := (\lambda_*\mathbf{I}_\rho + \mathbf{L}[t-1]\mathbf{\Omega_t}\mathbf{L}'[t-1])^{-1}\mathbf{L}'[t-1]$ for notational convenience, and substitute (5.11) back into (5.10) to arrive at the Lasso estimator

$$\mathbf{a}[t] = \arg\min_{\mathbf{a}} \left[\frac{1}{2}\|\mathbf{F}[t](\mathbf{y}_t - \mathbf{R}_t\mathbf{a})\|_2^2 + \lambda_1\|\mathbf{a}\|_1\right] \quad (5.12)$$

where $\mathbf{F}[t] := \left[\mathbf{\Omega}_t - \mathbf{\Omega}_t\mathbf{L}[t-1]\mathbf{D}[t]\mathbf{\Omega}_t, \sqrt{\lambda_*}\mathbf{\Omega}_t\mathbf{D}'[t]\right]'$. The diagonal matrix $\mathbf{\Omega}_t$ was defined in Section 5.3.2, see the discussion after (5.6).

In the second step of the alternating-minimization scheme, the updated subspace matrix $\mathbf{L}[t]$ is obtained by minimizing (5.9) with respect to $\mathbf{L}$, while the optimization variables $\{\mathbf{q}_\tau, \mathbf{a}_\tau\}_{\tau=1}^t$ are fixed and take the values $\{\mathbf{q}[\tau], \mathbf{a}[\tau]\}_{\tau=1}^t$. This yields

$$\mathbf{L}[t] = \arg\min_{\mathbf{L}} \left[\frac{\lambda_*}{2}\|\mathbf{L}\|_F^2 + \sum_{\tau=1}^t \beta^{t-\tau}\frac{1}{2}\|\mathcal{P}_{\Omega_\tau}(\mathbf{y}_\tau - \mathbf{L}\mathbf{q}[\tau] - \mathbf{R}_\tau\mathbf{a}[\tau])\|_2^2\right]. \quad (5.13)$$

Similar to the batch case, (5.13) decouples over the rows of $\mathbf{L}$ which are obtained in parallel via

$$\mathbf{l}_l[t] = \arg\min_{\mathbf{l}} \left[\frac{\lambda_*}{2}\|\mathbf{l}\|^2 + \sum_{\tau=1}^t \beta^{t-\tau}\omega_{l,\tau}(y_{l,\tau} - \mathbf{l}'\mathbf{q}[\tau] - (\mathbf{r}_{l,\tau}^r)'\mathbf{a}[\tau])^2\right], \quad l = 1, \ldots, L \quad (5.14)$$

where $\omega_{l,\tau}$ denotes the l-th diagonal entry of $\mathbf{\Omega}_\tau$. For $\beta = 1$, subproblems (5.14) can be efficiently solved using the RLS algorithm [131]. Upon defining $\mathbf{s}_l[t] := \sum_{\tau=1}^t \beta^{t-\tau}\omega_{l,\tau}(y_{l,\tau} - \mathbf{r}_{l,\tau}'\mathbf{a}[\tau])\mathbf{q}[\tau]$, $\mathbf{H}_l[t] := \sum_{\tau=1}^t \beta^{t-\tau}\omega_{l,\tau}\mathbf{q}[\tau]\mathbf{q}'[\tau] + \lambda_*\mathbf{I}_\rho$, and $\mathbf{M}_l[t] := \mathbf{H}_l^{-1}[t]$, with $\beta = 1$ one

Algorithm 10 : Online algorithm for tracking network anomalies

input $\{\mathcal{P}_{\Omega_t}(\mathbf{y}_t), \mathbf{\Omega}_t, \mathbf{R}_t\}_{t=1}^{\infty}$, β, λ_*, and λ_1.

initialize $\mathbf{G}_l[0] = \mathbf{0}_{\rho\times\rho}$, $\mathbf{s}_l[0] = \mathbf{0}_\rho$, $l = 1, ..., L$, and $\mathbf{L}[0]$ at random.

for $t = 1, 2, \ldots$ **do**

$\mathbf{D}[t] = (\lambda_*\mathbf{I}_\rho + \mathbf{L}'[t-1]\mathbf{\Omega}_t\mathbf{L}[t-1])^{-1}\mathbf{L}'[t-1]$.

$\mathbf{F}[t] = \left[\mathbf{\Omega}_t - \mathbf{\Omega}_t\mathbf{L}[t-1]\mathbf{D}[t]\mathbf{\Omega}_t, \sqrt{\lambda_*}\mathbf{\Omega}_t\mathbf{D}'[t]\right]'$.

$\mathbf{a}[t] = \arg\min_{\mathbf{a}} \left[\frac{1}{2}\|\mathbf{F}[t](\mathbf{y}_t - \mathbf{R}_t\mathbf{a})\|^2 + \lambda_1\|\mathbf{a}\|_1\right]$.

$\mathbf{q}[t] = \mathbf{D}[t]\mathbf{\Omega}_t(\mathbf{y}_t - \mathbf{R}_t\mathbf{a}[t])$.

$\mathbf{G}_l[t] = \beta\mathbf{G}_l[t-1] + \omega_{l,t}\mathbf{q}[t]\mathbf{q}[t]', \quad l = 1, \ldots, L$.

$\mathbf{s}_l[t] = \beta\mathbf{s}_l[t-1] + \omega_{l,t}(y_{l,t} - \mathbf{r}'_{l,t}\mathbf{a}[t])\mathbf{q}[t], \quad l = 1, \ldots, L$.

$\mathbf{l}_l[t] = (\mathbf{G}_l[t] + \lambda_*\mathbf{I}_\rho)^{-1}\mathbf{s}_l[t], \quad l = 1, ..., L$.

return $\hat{\mathbf{a}}_t := \mathbf{a}[t]$ and $\hat{\mathbf{x}}_t := \mathbf{L}[t]\mathbf{q}[t]$.

end for

simply updates

$$\mathbf{s}_l[t] = \mathbf{s}_l[t-1] + \omega_{l,t}(y_{l,t} - \mathbf{r}'_{l,t}\mathbf{a}[t])\mathbf{q}[t]$$

$$\mathbf{M}_l[t] = \mathbf{M}_l[t-1] - \omega_{l,t}\frac{\mathbf{M}_l[t-1]\mathbf{q}[t]\mathbf{q}'[t]\mathbf{M}_l[t-1]}{1 + \mathbf{q}'[t]\mathbf{M}_l[t-1]\mathbf{q}[t]}$$

and forms $\mathbf{l}_l[t] = \mathbf{M}_l[t]\mathbf{s}_l[t]$, for $l = 1, \ldots, L$.

However, for $0 < \beta < 1$ the regularization term $(\lambda_*/2)\|\mathbf{l}\|^2$ in (5.14) makes it impossible to express $\mathbf{H}_l[t]$ in terms of $\mathbf{H}_l[t-1]$ plus a rank-one correction. Hence, one cannot resort to the matrix inversion lemma and update $\mathbf{M}_l[t]$ with quadratic complexity only. Based on direct inversion of $\mathbf{H}_l[t]$, $l = 1, \ldots, L$, the overall recursive algorithm for tracking network anomalies is tabulated under Algorithm 10. The per iteration cost of the L inversions (each $\mathcal{O}(\rho^3)$, which could be further reduced if one leverages also the symmetry of $\mathbf{H}_l[t]$) is affordable for moderate number of links, because ρ is small when estimating low-rank traffic matrices. Still, for those settings where computational complexity reductions are at a premium, an online stochastic gradient descent algorithm is described in Section 5.5.1.

Remark 5.2 (Robust subspace trackers) *Algorithm 10 is closely related to timely robust subspace trackers, which aim at estimating a low-rank subspace* $\mathbf{L}$ *from grossly corrupted and possibly incomplete data, namely* $\mathcal{P}_{\Omega_t}(\mathbf{y}_t) = \mathcal{P}_{\Omega_t}(\mathbf{L}\mathbf{q}_t + \mathbf{a}_t + \mathbf{v}_t)$, $t = 1, 2, \ldots$. *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 [70]. The second-order RLS-type algorithm in [152] extends the seminal projection approximation subspace tracking algorithm [154] to handle missing data. When outliers are present, robust counterparts can be found in [36, 65, 107]. Relative to all aforementioned works, the estimation problem here is more challenging due to the presence of the fat (compression) matrix $\mathbf{R}_t$; see [97] for fundamental identifiability issues related to the model (5.3).

5.4.2 Convergence Analysis

This section studies the convergence of the iterates generated by Algorithm 10, for the infinite memory special case i.e., when $\beta = 1$. Upon defining the function

$$g_t(\mathbf{L}, \mathbf{q}, \mathbf{a}) := \frac{1}{2}\|\mathcal{P}_{\Omega_t}(\mathbf{y}_t - \mathbf{L}\mathbf{q} - \mathbf{R}_t\mathbf{a})\|_2^2 + \frac{\lambda_*}{2}\|\mathbf{q}\|_2^2 + \lambda_1\|\mathbf{a}\|_1 \tag{5.15}$$

in addition to $\ell_t(\mathbf{L}) := \min_{\{\mathbf{q},\mathbf{a}\}} g_t(\mathbf{L}, \mathbf{q}, \mathbf{a})$, when $\beta = 1$ Algorithm 10 aims at minimizing the following *average* cost function at time t

$$C_t(\mathbf{L}) := \frac{1}{t}\sum_{\tau=1}^{t} \ell_\tau(\mathbf{L}) + \frac{\lambda_*}{2t}\|\mathbf{L}\|_F^2. \tag{5.16}$$

Normalization (by t) ensures that the cost function does not grow unbounded as time evolves. For fixed routing $\{\mathbf{R}_\tau = \mathbf{R}\}_{\tau=1}^{t}$, (5.16) it is essentially identical to the batch estimator in (P3) up to a scaling, which does not affect the value of the minimizers. Note that as time evolves, minimization of C_t becomes increasingly complex computationally. Even evaluating C_t is challenging for large t, since it entails solving t Lasso problems to minimize all g_τ and defining the functions ℓ_τ, $\tau = 1, \ldots, T$. Hence, at time t the subspace estimate $\mathbf{L}[t]$ is obtained by minimizing the *approximate* cost function [cf. (5.13) when $\beta = 1$]

$$\hat{C}_t(\mathbf{L}) = \frac{1}{t}\sum_{\tau=1}^{t} g_\tau(\mathbf{L}, \mathbf{q}[\tau], \mathbf{a}[\tau]) + \frac{\lambda_*}{2t}\|\mathbf{L}\|_F^2 \tag{5.17}$$

in which $\{\mathbf{q}[t], \mathbf{a}[t]\}$ are obtained based on the prior subspace estimate $\mathbf{L}[t-1]$ after solving [cf. (5.10)]

$$\{\mathbf{q}[t], \mathbf{a}[t]\} = \arg\min_{\{\mathbf{q},\mathbf{a}\}} g_t(\mathbf{L}[t-1], \mathbf{q}, \mathbf{a}). \tag{5.18}$$

Obtaining $\mathbf{q}[t]$ this way resembles the projection approximation adopted in [154], and can only be evaluated after $\mathbf{a}[t]$ is obtained [cf. (5.11)]. Since $\hat{C}_t(\mathbf{L})$ is a smooth convex function, the minimizer $\mathbf{L}[t] = \arg\min_{\mathbf{L}} \hat{C}_t(\mathbf{L})$ is the solution of the quadratic equation $\nabla \hat{C}_t(\mathbf{L}[t]) = \mathbf{0}_{L\times\rho}$.

So far, it is apparent that the approximate cost function $\hat{C}_t(\mathbf{L}[t])$ overestimates the target cost $C_t(\mathbf{L}[t])$, for $t = 1, 2, \ldots$. However, it is not clear whether the dictionary iterates $\{\mathbf{L}[t]\}_{t=1}^{\infty}$ converge, and most importantly, how well can they optimize the target cost function C_t. The good news is that $\hat{C}_t(\mathbf{L}[t])$ asymptotically approaches $C_t(\mathbf{L}[t])$, and the subspace iterates null $\nabla C_t(\mathbf{L}[t])$ as well, both as $t \to \infty$. The latter result is summarized in the next proposition, which is proved in the next section.

Proposition 5.3 *Assume that: a1)* $\{\Omega_t\}_{t=1}^{\infty}$ *and* $\{\mathbf{y}_t\}_{t=1}^{\infty}$ *are independent and identically distributed (i.i.d.) random processes; a2)* $\|\mathcal{P}_{\Omega_t}(\mathbf{y}_t)\|_\infty$ *is uniformly bounded; a3) iterates* $\{\mathbf{L}[t]\}_{t=1}^{\infty}$ *are in a compact set* $\mathcal{L} \subset \mathbb{R}^{L\times\rho}$*; a4)* $\hat{C}_t(\mathbf{L})$ *is positive definite, namely* $\lambda_{\min}\left[\nabla^2 \hat{C}_t(\mathbf{L})\right] \geq c$ *for some* $c > 0$*; and a5)* $\sigma_{\min}(\mathbf{S}[t]) \geq c_0$*, where the matrix* $\mathbf{S}[t] \in \mathbb{R}^{(L+\rho)\times|\mathrm{supp}(\mathbf{a}[t])|}$ *contains the columns of* $\mathbf{F}[t]\mathbf{R}_t$ *associated with the elements in supp*$(\mathbf{a}[t])$*, and* c_0 *is a positive constant. Then* $\lim_{t\to\infty} \nabla C_t(\mathbf{L}[t]) = \mathbf{0}_{L\times\rho}$ *almost surely (a.s.), which implies that the subspace iterates* $\{\mathbf{L}[t]\}_{t=1}^{\infty}$ *asymptotically coincide with the stationary points of (P3) when the routing remains invariant, i.e., when* $\mathbf{R}_t = \mathbf{R}$*,* $t = 1, 2, \ldots$.

To clearly delineate the scope of the analysis, it is worth commenting on the assumptions a1)-a5) and the factors that influence their satisfaction. Regarding a1), the acquired data is assumed statistically independent across time as it is customary when studying the stability and performance of online (adaptive) algorithms [124, 131]. While independence is required for tractability, a1) may be grossly violated because OD flows are correlated across time (cf. the low-rank property of $\mathbf{Z}$ and $\mathbf{X}$). Still, in accordance with the adaptive filtering folklore e.g., [124, p. 321], as $\beta \to 1$ the upshot of the analysis based on

i.i.d. data extends accurately to the pragmatic setting whereby the link-counts and missing data patterns exhibit spatiotemporal correlations. Uniform boundedness of $\mathcal{P}_{\Omega_t}(\mathbf{y}_t)$ [cf. a2)] is satisfied in practice, since the traffic is always limited by the (finite) capacity of the physical links. The bounded subspace requirement in a3) is a technical assumption that simplifies the arguments of the ensuing proof, and has been corroborated via extensive computer simulations including those in Section 5.6. It is apparent that the sampling set Ω_t plays a key role towards ensuring that a4) and a5) are satisfied. Intuitively, if the missing entries tend to be only few and somehow uniformly distributed across links and time, they will not markedly increase coherence of the regression matrices $\mathbf{F}[t]\mathbf{R}_t$, and thus compromise the uniqueness of the Lasso solutions. This also increases the likelihood that $\nabla^2\hat{C}_t(\mathbf{L}) = \frac{\lambda_*}{t}\mathbf{I}_{L\rho} + \frac{1}{t}\sum_{\tau=1}^{t}(\mathbf{q}[\tau]\mathbf{q}'[\tau]) \otimes \mathbf{\Omega}_\tau \succeq c\mathbf{I}_{L\rho}$ holds. As argued in [60], if needed one could incorporate additional regularization terms in the cost function to enforce a4) and a5). Before moving on to the proof, a remark is in order.

Remark 5.3 (Performance guarantees) *In line with Proposition 5.2, one may be prompted to ponder whether the online estimator offers the performance guarantees of the nuclear-norm regularized estimator (P1), for which stable/exact recovery have been well documented e.g., in [25, 97, 163]. Specifically, given the learned traffic subspace* $\bar{\mathbf{L}}$ *and the corresponding* $\bar{\mathbf{Q}}$ *and* $\bar{\mathbf{A}}$ *[obtained via (5.10)] over a time window of size* T*, is* $\{\hat{\mathbf{X}} := \bar{\mathbf{L}}\bar{\mathbf{Q}}', \hat{\mathbf{A}} := \bar{\mathbf{A}}\}$ *an optimal solution of (P1) when* $T \to \infty$*? This in turn requires asymptotic analysis of the optimality conditions for (P1), and is left for future research. Nevertheless, empirically the online estimator attains the performance of (P1), as evidenced by the numerical tests in Section 5.6.*

5.4.3 Proof of Proposition 5.3

The main steps of the proof are inspired by [60], which studies convergence of an online dictionary learning algorithm using the theory of martingale sequences; see e.g., [71]. However, relative to [60] the problem here introduces several distinct elements including: i) missing data with a time-varying pattern Ω_t; ii) a non-convex bilinear term where the tall subspace matrix $\mathbf{L}$ plays a role similar to the fat dictionary in [60], but the multiplica-

tive projection coefficients here are not sparse; and iii) the additional bilinear terms $\mathbf{R}_t\mathbf{a}_t$ which entail sparse coding of $\mathbf{a}_t$ as in [60], but with a known regression (routing) matrix. Hence, convergence analysis becomes more challenging and demands, in part, for a new treatment. Accordingly, in the sequel emphasis will be placed on the novel aspects specific to the problem at hand.

The basic structure of the proof consists of three preliminary lemmata, which are subsequently used to establish that $\lim_{t\to\infty}\nabla C_t(\mathbf{L}[t])=\mathbf{0}_{L\times\rho}$ a.s. through a simple argument. The first lemma deals with regularity properties of functions $\hat{C}_t$ and C_t, which will come handy later on; see Appendix for a proof.

Lemma 5.1 *If a2) and a5) hold, then the functions: i)* $\{\mathbf{a}_t(\mathbf{L}),\mathbf{q}_t(\mathbf{L})\}=\arg\min_{\{\mathbf{q},\mathbf{a}\}} g_t(\mathbf{L},\mathbf{q},\mathbf{a})$*, ii)* $g_t(\mathbf{L},\mathbf{q}[t],\mathbf{a}[t])$*, iii)* $\ell_t(\mathbf{L})$*, and iv)* $\nabla\ell_t(\mathbf{L})$ *are Lipschitz continuous for* $\mathbf{L}\in\mathcal{L}$ *(*$\mathcal{L}$ *is a compact set), with constants independent of* t*.*

The next lemma (proved in Appendix) asserts that the distance between two subsequent traffic subspace estimates vanishes as $t\to\infty$, a property that will be instrumental later on when establishing that $\hat{C}_t(\mathbf{L}[t])-C_t(\mathbf{L}[t])\to 0$ a.s.

Lemma 5.2 *If a2)-a5) hold, then* $\|\mathbf{L}[t+1]-\mathbf{L}[t]\|_F=\mathcal{O}(1/t)$*.*

The previous lemma by no means implies that the subspace iterates converge, which is a much more ambitious objective that may not even hold under the current assumptions. The final lemma however, asserts that the cost sequence indeed converges with probability one; see Appendix for a proof.

Lemma 5.3 *If a1)-a5) hold, then* $\hat{C}_t(\mathbf{L}[t])$ *converges a.s. Moreover,* $\hat{C}_t(\mathbf{L}[t])-C_t(\mathbf{L}[t])\to 0$ *a.s.*

Putting the pieces together, in the sequel it is shown that the sequence $\{\nabla\hat{C}_t(\mathbf{L}[t])-\nabla C_t(\mathbf{L}[t])\}_{t=1}^{\infty}$ converges a.s. to zero, and since $\nabla\hat{C}_t(\mathbf{L}[t])=\mathbf{0}_{L\times\rho}$ by algorithmic construction, the subspace iterates $\{\mathbf{L}[t]\}_{t=1}^{\infty}$ coincide with the stationary points of the target cost function C_t. To this end, it suffices to prove that every convergent *subsequence* nulls the

gradient ∇C_t asymptotically, which in turn implies that the entire sequence converges to the set of stationary points of the batch problem (P3).

Since $\mathcal{L}$ is compact by virtue of a3), one can always pick a convergent subsequence $\{\mathbf{L}[t]\}_{t=1}^{\infty}$ whose limit point is $\mathbf{L}^*$, say[2]. Consider the positive-valued decreasing sequence $\{\alpha_t\}_{t=1}^{\infty}$ that converges to zero slower than $\hat{C}_t(\mathbf{L}[t])-C_t(\mathbf{L}[t])$ does, and recall that $\hat{C}_t(\mathbf{L}[t]+\alpha_t\mathbf{U}) \geq C_t(\mathbf{L}[t]+\alpha_t\mathbf{U})$ for any $\mathbf{U}\in\mathbb{R}^{L\times\rho}$. From the mean-value theorem and for arbitrary $\mathbf{U}$, expanding both sides of the inequality around the point $\mathbf{L}[t]$ one arrives at

$$\begin{aligned}\hat{C}_t(\mathbf{L}[t]) + \alpha_t \mathrm{tr}\{\mathbf{U}'\nabla\hat{C}_t(\mathbf{L}[t])\} + \alpha_t \mathrm{tr}\{\mathbf{U}'(\nabla\hat{C}_t(\mathbf{\Theta}_1[t]) - \nabla\hat{C}_t(\mathbf{P}[t]))\} \geq \\ C_t(\mathbf{L}[t]) + \alpha_t \mathrm{tr}\{\mathbf{U}'\nabla C_t(\mathbf{L}[t])\} + \alpha_t \mathrm{tr}\{\mathbf{U}'(\nabla C_t(\mathbf{\Theta}_2[t]) - \nabla C_t(\mathbf{P}[t]))\}\end{aligned}$$

for some $\mathbf{\Theta}_1[t], \mathbf{\Theta}_2[t] \in \mathbb{R}^{L\times\rho}$ on the line segment connecting $\mathbf{L}[t]$ and $\mathbf{L}[t]+\alpha_t\mathbf{U}$. Taking limit as $t\to\infty$ and applying Lemma 5.3 it follows that

$$\begin{aligned}\lim_{t\to\infty} \mathrm{tr}\{\mathbf{U}'(\nabla\hat{C}_t(\mathbf{L}[t]) - \nabla C_t(\mathbf{L}[t]))\} + \lim_{t\to\infty} \mathrm{tr}\{\mathbf{U}'(\nabla\hat{C}_t(\mathbf{\Theta}_1[t]) - \nabla\hat{C}_t(\mathbf{P}[t]))\} \\ + \lim_{t\to\infty} \mathrm{tr}\{\mathbf{U}'(\nabla C_t(\mathbf{P}[t]) - \nabla C_t(\mathbf{\Theta}_2[t]))\} \geq 0, \quad \text{a.s.} \qquad (5.19)\end{aligned}$$

For the quadratic function $\hat{C}_t$, uniform boundedness of the Hessian $\nabla^2\hat{C}_t(\mathbf{L}) = \frac{\lambda_*}{t}\mathbf{I}_{L\rho} + \frac{1}{t}\sum_{\tau=1}^{t}(\mathbf{q}[\tau]\mathbf{q}'[\tau])\otimes\mathbf{\Omega}_\tau$ implies that $\nabla\hat{C}_t$ is Lipschitz. Furthermore, since $\nabla\ell_\tau$ is Lipschitz as per Lemma 5.1, ∇C_t is Lipschitz as well. Consequently, according to the Cauchy-Schwarz inequality

$$\begin{aligned}\left|\mathrm{tr}\{\mathbf{U}'(\nabla C_t(\mathbf{P}[t]) - \nabla C_t(\mathbf{\Theta}_2[t]))\}\right| &\leq \|\mathbf{U}\|_F \|\nabla C_t(\mathbf{P}[t]) - \nabla C_t(\mathbf{\Theta}_2[t])\|_F \\ &\leq c\ \|\mathbf{U}\|_F \|\mathbf{P}[t] - \mathbf{\Theta}_2[t]\|_F \overset{(a)}{\leq} c\ \alpha_t \|\mathbf{U}\|_F^2 \qquad (5.20)\end{aligned}$$

for some constant $c>0$, where (a) holds since $\mathbf{\Theta}_2[t]$ is a convex combination of $\mathbf{L}[t]$ and $\mathbf{L}[t]+\alpha_t\mathbf{U}$. Likewise, one can bound the second term on the left-hand-side of (5.19). Accordingly, it holds that

$$\lim_{t\to\infty} \mathrm{tr}\{\mathbf{U}'(\nabla C_t(\mathbf{P}[t]) - \nabla C_t(\mathbf{\Theta}_2[t]))\} = \lim_{t\to\infty} \mathrm{tr}\{\mathbf{U}'(\nabla\hat{C}_t(\mathbf{P}[t]) - \nabla\hat{C}_t(\mathbf{\Theta}_1[t]))\} = 0.$$

[2]Formally, the subsequence should be denoted as $\{\mathbf{L}[t(i)]\}_{i=1}^{\infty}$, but a slight abuse of notation is allowed for simplicity.

All in all, the second and third terms in (5.19) vanish and one is left with

$$\lim_{t\to\infty} \mathrm{tr}\{\mathbf{U}'(\nabla \hat{C}_t(\mathbf{L}_t) - \nabla C_t(\mathbf{L}_t))\} \geq 0. \tag{5.21}$$

Because $\mathbf{U} \in \mathbb{R}^{L\times\rho}$ is arbitrary, (5.21) can only hold if $\lim_{t\to\infty}(\nabla \hat{C}_t(\mathbf{L}[t]) - \nabla C_t(\mathbf{L}[t])) = \mathbf{0}_{L\times\rho}$ a.s., which completes the proof. ■

5.5 Further Algorithmic Issues

For completeness, this section outlines a couple of additional algorithmic aspects relevant to anomaly detection in *large-scale* networks. Firstly, a lightweight first-order algorithm is developed as an alternative to Algorithm 10, which relies on fast Nesterov-type gradient updates for the traffic subspace. Secondly, the possibility of developing distributed algorithms for dynamic anomalography is discussed.

5.5.1 Fast stochastic-gradient algorithm

Reduction of the computational complexity in updating the traffic subspace $\mathbf{L}$ is the subject of this section. The basic alternating minimization framework in Section 5.4.1 will be retained, and the updates for $\{\mathbf{q}[t], \mathbf{a}[t]\}$ will be identical to those tabulated under Algorithm 10. However, instead of solving an unconstrained quadratic program per iteration to obtain $\mathbf{L}[t]$ [cf. (5.13)], the refinements to the subspace estimate will be given by a (stochastic) gradient algorithm.

As discussed in Section 5.4.2, in Algorithm 10 the subspace estimate $\mathbf{L}[t]$ is obtained by minimizing the empirical cost function $\hat{C}_t(\mathbf{L}) = (1/t)\sum_{\tau=1}^{t} f_\tau(\mathbf{L})$, where

$$f_t(\mathbf{L}) := \frac{1}{2}\|\mathbf{\Omega}_t(\mathbf{y}_t - \mathbf{L}\mathbf{q}[t] - \mathbf{R}_t\mathbf{a}[t])\|_2^2 + \frac{\lambda_*}{2t}\|\mathbf{L}\|_F^2 + \frac{\lambda_*}{2}\|\mathbf{q}[t]\|_2^2 + \lambda_1\|\mathbf{a}[t]\|_1, \quad t = 1, 2, \ldots \tag{5.22}$$

By the law of large numbers, if data $\{\mathcal{P}_{\Omega_t}(\mathbf{y}_t)\}_{t=1}^{\infty}$ are stationary, solving $\min_{\mathbf{L}} \lim_{t\to\infty} \hat{C}_t(\mathbf{L})$ yields the desired minimizer of the *expected* cost $\mathbb{E}[C_t(\mathbf{L})]$, where the expectation is taken with respect to the unknown probability distribution of the data. A standard approach to achieve this same goal – typically with reduced computational complexity – is to drop the

expectation (or the sample averaging operator for that matter), and update the nominal traffic subspace via a stochastic gradient iteration [131]

$$\begin{aligned}\mathbf{L}[t] &= \arg\min_{\mathbf{L}} Q_{(1/\tilde{\mu}[t]),t}(\mathbf{L}, \mathbf{L}[t-1]) \\ &= \mathbf{L}[t-1] - \tilde{\mu}[t]\nabla f_t(\mathbf{L}[t-1]) \end{aligned} \tag{5.23}$$

where $\tilde{\mu}[t]$ is a stepsize, $Q_{\mu,t}(\mathbf{L}_1, \mathbf{L}_2) := f_t(\mathbf{L}_2) + \langle \mathbf{L}_1 - \mathbf{L}_2, \nabla f_t(\mathbf{L}_2)\rangle + \frac{\mu}{2}\|\mathbf{L}_1 - \mathbf{L}_2\|_f^2$, and $\nabla f_t(\mathbf{L}) = -\mathbf{\Omega}_t(\mathbf{y}_t - \mathbf{L}\mathbf{q}[t] - \mathbf{R}_t\mathbf{a}[t])\mathbf{q}'[t] + (\lambda_*/t)\mathbf{L}$. In the context of adaptive filtering, stochastic gradient algorithms such as (5.22) are known to converge typically slower than RLS. This is expected since RLS can be shown to be an instance of Newton's (second-order) optimization method [131].

Building on the increasingly popular *accelerated* gradient methods for (batch) smooth optimization [14, 110], the idea here is to speed-up the learning rate of the estimated traffic subspace (5.23), without paying a penalty in terms of computational complexity per iteration. The critical difference between standard gradient algorithms and the so-termed Nesterov's variant, is that the accelerated updates take the form $\mathbf{L}[t] = \tilde{\mathbf{L}}[t] - \tilde{\mu}[t]\nabla f_t(\tilde{\mathbf{L}}[t])$, which relies on a judicious linear combination $\tilde{\mathbf{L}}[t-1]$ of the previous pair of iterates $\{\mathbf{L}[t-1], \mathbf{L}[t-2]\}$. Specifically, the choice $\tilde{\mathbf{L}}[t] = \mathbf{L}[t-1] + \frac{k[t-1]-1}{k[t]}(\mathbf{L}[t-1] - \mathbf{L}[t-2])$, where $k[t] = \left[1 + \sqrt{4k^2[t-1]+1}\right]/2$, has been shown to significantly accelerate batch gradient algorithms resulting in convergence rate no worse than $\mathcal{O}(1/k^2)$; see e.g., [14] and references therein. Using this acceleration technique in conjunction with a backtracking stepsize rule [17], a fast online stochastic gradient algorithm for unveiling network anomalies is tabulated under Algorithm 11. Different from Algorithm 10, no matrix inversions are involved in the update of the traffic subspace $\mathbf{L}[t]$. Clearly, a standard (non accelerated) stochastic gradient descent algorithm with backtracking stepsize rule is subsumed as a special case, when $k[t] = 1$, $t = 0, 1, 2, \ldots$

Convergence analysis of Algorithm 11 is beyond the scope of this paper, and will only be corroborated using computer simulations in Section 5.6. It is worth pointing out that since a non-diminishing stepsize is adopted, asymptotically the iterates generated by Algorithm 11 will hover inside a ball centered at the minimizer of the expected cost, with radius

Algorithm 11 : Online stochastic gradient algorithm for unveiling network anomalies

input $\{\mathbf{y}_t, \mathbf{R}_t, \mathbf{\Omega}_t\}_{t=1}^{\infty}$, $\rho, \lambda_*, \lambda_1, \eta > 1$.

initialize $\mathbf{L}[0]$ at random, $\mu[0] > 0$, $\tilde{\mathbf{L}}[1] := \mathbf{L}[0]$, and $k[1] := 1$.

for $t = 1, 2, \ldots$ **do**

$\mathbf{D}[t] = (\lambda_* \mathbf{I}_\rho + \mathbf{L}'[t-1]\mathbf{\Omega}_t \mathbf{L}[t-1])^{-1} \mathbf{L}'[t-1]$

$\mathbf{F}'[t] := \left[\mathbf{\Omega}_t - \mathbf{\Omega}_t \mathbf{L}[t-1]\mathbf{D}[t]\mathbf{\Omega}_t, \sqrt{\lambda_*}\mathbf{\Omega}_t \mathbf{D}'[t]\right]$

$\mathbf{a}[t] = \arg\min_{\mathbf{a}} \left[\frac{1}{2}\|\mathbf{F}[t](\mathbf{y}_t - \mathbf{R}_t \mathbf{a})\|^2 + \lambda_1 \|\mathbf{a}\|_1\right]$

$\mathbf{q}[t] = \mathbf{D}[t]\mathbf{\Omega}_t(\mathbf{y}_t - \mathbf{R}_t \mathbf{a}_t)$

Find the smallest nonnegative integer $i[t]$ such that with $\bar{\mu} := \eta^{i[t]}\mu[t-1]$

$$f_t(\tilde{\mathbf{L}}[t] - (1/\bar{\mu})\nabla f_t(\tilde{\mathbf{L}}[t])) \leq Q_{\bar{\mu},t}(\tilde{\mathbf{L}}[t] - (1/\bar{\mu})\nabla f_t(\tilde{\mathbf{L}}[t]), \tilde{\mathbf{L}}[t])$$

holds, and set $\mu[t] = \eta^{i[t]}\mu[t-1]$.

$\mathbf{L}[t] = \tilde{\mathbf{L}}[t] - (1/\mu[t])\nabla f_t(\tilde{\mathbf{L}}[t])$.

$k[t+1] = \frac{1+\sqrt{1+4k^2[t]}}{2}$.

$\tilde{\mathbf{L}}[t+1] = \mathbf{L}[t] + \left(\frac{k[t]-1}{k[t+1]}\right)(\mathbf{L}[t] - \mathbf{L}[t-1])$.

end for

return $\hat{\mathbf{x}}[t] := \mathbf{L}[t]\mathbf{q}[t], \hat{\mathbf{a}}[t] := \mathbf{a}[t]$.

proportional to the noise variance.

5.5.2 In-network anomaly trackers

Implementing Algorithms 9-11 presumes that network nodes continuously communicate their local link traffic measurements to a central monitoring station, which uses their aggregation in $\{\mathcal{P}_{\Omega_t}(\mathbf{y}_t)\}_{t=1}^{\infty}$ to unveil network anomalies. While for the most part this is the prevailing operational paradigm adopted in current network technologies, it is fair to say there are limitations associated with this architecture. For instance, collecting all this information centrally may lead to excessive protocol overhead, especially when the rate of data acquisition is high at the routers. Moreover, minimizing the exchanges of raw measurements may be desirable to reduce unavoidable communication errors that translate to missing data. Performing the optimization in a centralized fashion raises robustness concerns as well, since the central monitoring station represents an isolated point of failure.

These reasons motivate devising *fully-distributed* iterative algorithms for dynamic anomalography in large-scale networks, embedding the network anomaly detection functionality to the routers. In a nutshell, per iteration nodes carry out simple computational tasks locally, relying on their own link count measurements (a few entries of the network-wide vector $\mathbf{y}_t$ corresponding to the router 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, and attain (or at least come close to) the estimation performance of the centralized counterpart which has all data $\{\mathcal{P}_{\Omega_t}(\mathbf{y}_t)\}_{t=1}^{\infty}$ available.

Relying on the alternating-directions method of multipliers (AD-MoM) as the basic tool to carry out distributed optimization, a general framework for in-network sparsity-regularized rank minimization was put forth in a companion paper [95]. In the context of network anomaly detection, results therein are encouraging yet there is ample room for improvement and immediate venues for future research open up. For instance, the distributed algorithms of [95] can only tackle the batch formulation (P3), so extensions to a dynamic network setting, e.g., building on the ideas here to devise distributed anomaly trackers seems natural. To obtain desirable tradeoffs in terms of computational complexity and speed of convergence, developing and studying algorithms for distributed optimization based on Nesterov's acceleration techniques emerges as an exciting and rather pristine research direction; see [62] for early work dealing with separable batch optimization.

5.6 Performance Tests

Performance of the proposed batch and online estimators is assessed in this section via computer simulations using both synthetic and real network data.

Selection of tuning parameters. In the batch case, λ_1 and λ_* are tuned to optimize the relative error $\|\hat{\mathbf{A}}-\mathbf{A}_0\|_F/\|\mathbf{A}_0\|_F$, with $\mathbf{A}_0$ and $\hat{\mathbf{A}}$ denoting the true and estimated anomaly matrices, respectively. In particular, one needs to perform a grid search over the bounded two-dimensional region $\mathcal{R} := \{(\lambda_1,\lambda_*) : \lambda_1 \in (0, \|\mathbf{R}'\mathcal{P}_\Omega(\mathbf{Y})\|_\infty],\ \lambda_* \in (0, \|\mathcal{P}_\Omega(\mathbf{Y})\|]\}$. The corresponding bounds are derived from the optimality conditions for (P1), which indi-

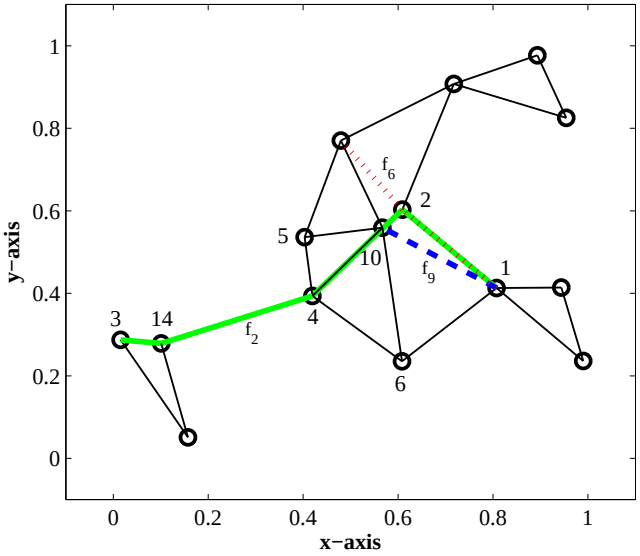
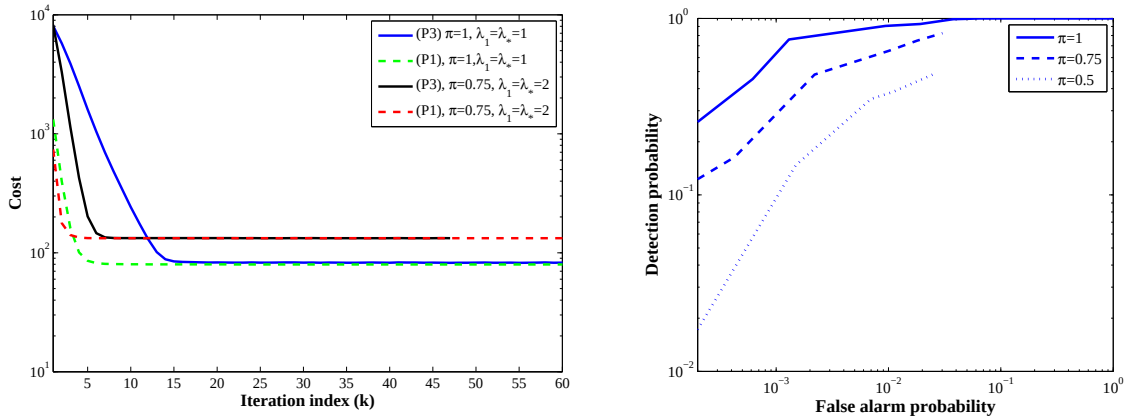


Figure 5.1: Synthetic network topology graph, and the paths used for routing three flows.

cate that for $(\lambda_1, \lambda_*) \in \mathcal{R}^c$ the optimal solution is $\{\mathbf{0}_{L\times T}, \mathbf{0}_{F\times T}\}$. Practical rules that do not require knowledge of $\mathbf{A}_0$ can be devised along the lines of [7] and [26]. Supposing that the true values are zero, choosing $\lambda_1 > \|\mathbf{R}'\mathcal{P}_\Omega(\mathbf{V})\|_\infty$ and $\lambda_* > \|\mathcal{P}_\Omega(\mathbf{V})\|$ the estimator (P1) outputs $\{\hat{\mathbf{X}} = \mathbf{0}_{L\times T}, \hat{\mathbf{A}} = \mathbf{0}_{F\times T}\}$. This mitigates noise, but it may overshrink the true values. To avoid overshrinking, these parameters can be chosen close to their corresponding lower bounds, e.g., pick $\lambda_* = \|\mathcal{P}_\Omega(\mathbf{V})\|$ and $\lambda_1 = \|\mathbf{R}'\mathcal{P}_\Omega(\mathbf{V})\|_\infty$. One can further simplify the candidate parameters by making the following reasonable assumptions: i) Gaussian noise $v_{l,t} \sim \mathcal{N}(0, \sigma^2)$, ii) uniform sampling with each entry of Ω chosen independently with probability π, and iii) large dimensions $F, T \to \infty$. It is then known that $(\sqrt{F} + \sqrt{T})^{-1}\|\mathcal{P}_\Omega(\mathbf{V})\| \to \sqrt{\pi}\sigma$, almost surely, see e.g., [26], and thus one can pick $\lambda_* = (\sqrt{F} + \sqrt{T})\sqrt{\pi}\sigma$. Also, large-deviation tail bounding implies that $\|\mathbf{R}'\mathcal{P}_\Omega(\mathbf{V})\|_\infty \leq 4\sigma \max_i \|\mathbf{R}\mathbf{e}_i\|_2 \log(FT)$ with high probability, which suggests selecting $\lambda_1 = \sigma \max_i \|\mathbf{R}\mathbf{e}_i\|_2 \log(FT)$. The said regularization parameters can also be used for online processing (upon setting $T = t$) , where they naturally increase as time evolves.

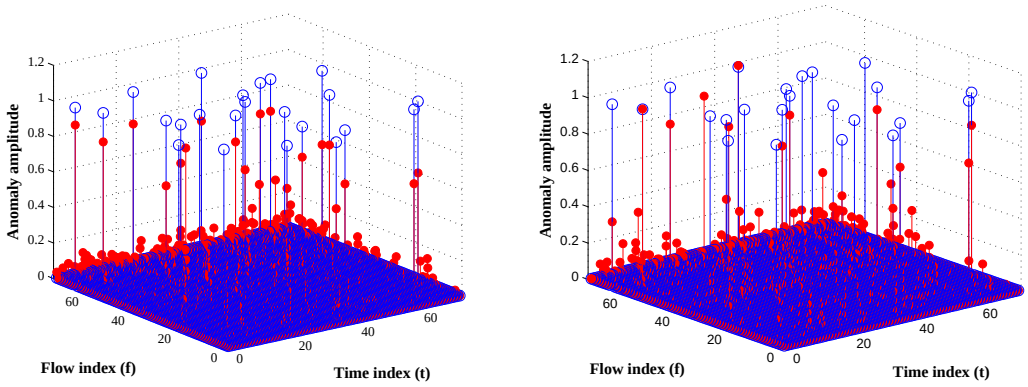


(a) (b)

Figure 5.2: Performance of the batch estimator (P3) for $p = 0.005$ and different amounts of missing data. (a) Cost of the estimators (P1) and (P3) versus iteration index when $\sigma = 10^{-2}$. (b) ROC curves when $\sigma = 10^{-1}$.

5.6.1 Synthetic-network data tests

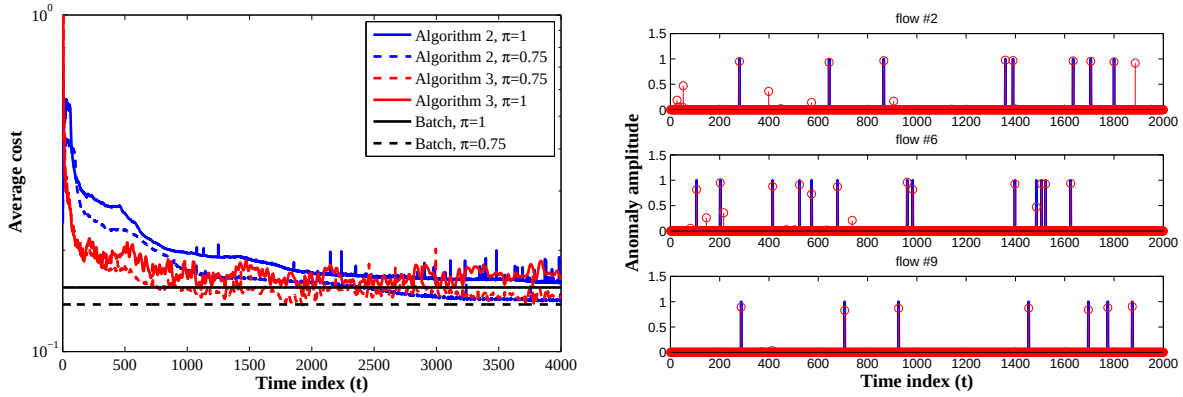
Synthetic network example. A network of $N = 15$ nodes is considered as a realization of the random geometric graph model with agents randomly placed on the unit square, and two agents link if their Euclidean distance is less than a prescribed communication range of $d_c = 0.35$; see Fig. 5.1. The network graph is bidirectional and comprises $L = 52$ links, and $F = N(N-1) = 210$ OD flows. For each candidate OD pair, minimum hop count routing is considered to form the routing matrix $\mathbf{R}$. Entries of $\mathbf{v}_t$ are i.i.d., zero-mean, Gaussian with variance σ^2; i.e., $v_{l,t} \sim \mathcal{N}(0, \sigma^2)$. Flow-traffic vectors $\mathbf{z}_t$ are generated from the low-dimensional subspace $\mathbf{U} \in \mathbb{R}^{F \times r}$ with i.i.d. entries $u_{f,i} \sim \mathcal{N}(0, 1/F)$, and projection coefficients $w_{i,t} \sim N(0,1)$ such that $\mathbf{z}_t = \mathbf{U}\mathbf{w}_t$. Every entry of $\mathbf{a}_t$ is randomly drawn from the set $\{-1, 0, 1\}$, with $\Pr(a_{f,t} = -1) = \Pr(a_{f,t} = 1) = p/2$. Entries of $\mathbf{Y}$ are sampled uniformly at random with probability π to form the diagonal sampling matrix $\mathbf{\Omega}_t$. The observations at time instant t are generated according to $\mathcal{P}_{\Omega_t}(\mathbf{y}_t) = \mathbf{\Omega}_t(\mathbf{R}\mathbf{z}_t + \mathbf{R}\mathbf{a}_t + \mathbf{v}_t)$. Unless otherwise stated, $r = 2$, $\rho = 5$, and $\beta = 0.99$ are used throughout. Different values of σ, p and π are tested.



(a) (b)

Figure 5.3: Amplitude of the true (blue) and estimated (red) anomalies for $\sigma = 10^{-1}$. (a) $\pi = 1$ (no missing data), $P_{\text{FA}} = 0.021$ and $P_{\text{D}} = 0.96$. (b) $\pi = 0.75$, $P_{\text{FA}} = 0.016$ and $P_{\text{D}} = 0.69$.

Performance of the batch estimator. To demonstrate the merits of the batch BCD algorithm for unveiling network anomalies (Algorithm 9), simulated data are generated for a time interval of size $T = 100$. For validation purposes, the benchmark estimator (P1) is iteratively solved by alternating minimization over $\mathbf{A}$ (which corresponds to Lasso) and $\mathbf{X}$. The minimizations with respect to $\mathbf{X}$ can be carried out using the iterative singular-value thresholding (SVT) algorithm [24]. Note that with full data, SVT requires only a single SVD computation. In the presence of missing data however, the SVT algorithm may require several SVD computations until convergence, rendering the said algorithm prohibitively complex for large-scale problems. In contrast, Algorithm 9 only requires simple $\rho \times \rho$ inversions. Fig. 5.2 (a) depicts the convergence of the respective algorithms used to solve (P1) and (P3), for different amounts of missing data (controlled by π). It is apparent that both estimators attain identical performance after a few tens of iterations, as asserted by Proposition 5.1. To corroborate the effectiveness of Algorithm 9 in unveiling network anomalies across flows and time, the ROC curves are plotted for various percentages of missing link observations in Fig. 5.2 (b) when $\sigma = 10^{-1}$. To discard spurious estimates, the hypothesis test $\hat{a}_{f,t} \underset{<\mathcal{H}_0}{\overset{\geq\mathcal{H}_1}{}} 0.1$ is considered, with anomalous and anomaly-free hypotheses $\mathcal{H}_1$ and $\mathcal{H}_0$, respectively. Apparently, an inferior detection performance is expected as the



(a) (b)

Figure 5.4: Performance of the online estimator for $\sigma = 10^{-2}$, $p = 0.005$, $\lambda_1 = 0.11$, and $\lambda_* = 0.36$. (a) Evolution of the average cost $C_t(\mathbf{L}[t])$ of the online algorithms versus the batch counterpart (P3). (b) Amplitude of true (solid) and estimated (circle markers) anomalies via the online Algorithm 10, for three representative flows when $\pi = 1$ (no missing data).

percentage of missing data increases. Note that when link observations are missing ($\pi < 1$), some flows may not be identifiable because they may traverse none of the observed links. For such flows, the anomalous traffic is assumed zero. Hence, as it is seen in Fig. 5.2 (b), the maximum achievable detection probability equals the fraction of (partially) observed flows. For the instances of $(P_{\text{FA}} = 0.021, P_{\text{D}} = 0.96)$ and $(P_{\text{FA}} = 0.016, P_{\text{D}} = 0.69)$ corresponding to $\pi = 1$ and $\pi = 0.75$, respectively, Fig. 5.3 depicts the magnitude of the true and estimated anomalies.

Performance of the online algorithms. To confirm the convergence and effectiveness of the online Algorithms 10 and 11, simulation tests are carried out for infinite memory $\beta = 1$ and invariant routing matrix $\mathbf{R}$. Fig. 5.4 (a) depicts the evolutions of the average cost $C_t(\mathbf{L}_t)$ in (5.16) for different amounts of missing data $\pi = 0.75, 1$ when the noise level is $\sigma = 10^{-2}$. It is evident that for both online algorithms the average cost converges (possibly within a ball) to its batch counterpart in (P3) normalized by the window size $T = t$. Impressively, this observation together with the one in Fig. 5.2 (a) corroborate that the online estimators can attain the performance of the benchmark estimator, whose

stable/exact recovery performance is well documented e.g., in [27, 97, 163]. It is further observed that the more data are missing, the more time it takes to learn the low-rank nominal traffic subspace, which in turn slows down convergence.

To examine the tracking capability of the online estimators, Fig. 5.4 (b) depicts the estimated versus true anomalies over time as Algorithm 10 evolves for three representative flows indicated on Fig. 5.1, namely f_2, f_6, f_9 corresponding to the $f = 2, 6, 9$-th rows of $\mathbf{A}_0$. Setting the detection threshold to the value 0.1 as before, for the flows f_2, f_6, f_9 Algorithm 10 attains detection rate $P_{\rm D} = 0.83, 1, 1$ at false alarm rate $P_{\rm FA} = 0.0171, 0.0040, 0.0081$, respectively. The quantification error per flow is also around $P_{\rm Q} = 0.7606, 0.5863, 0.4028$, respectively. As expected, more false alarms are declared at early iterations as the low-rank subspace has not been learnt accurately. Upon learning the subspace performance improves and almost all anomalies are identified. Careful inspection of Fig. 5.4 (b) reveals that the anomalies for f_9 are better identified visually than those for f_2. As shown in Fig. 5.1, f_2 is carried over links $(1,2), (2,4), (4,14), (14,3)$ each one carrying $33, 31, 35, 22$ additional flows, respectively, whereas f_9 is aggregated over link $(1,3)$ with only 2 additional flows. Hence, identifying f_2's anomalies from the highly-superimposed load of links $(1,2), (2,4), (4,14), (14,3)$ is a more challenging task relative to link $(1,3)$. This simple example manifests the fact that the detection performance strongly depends on the network topology and the routing policy implemented, which determine the routing matrix. In accordance with [97], the coherence of sparse column subsets of the routing matrix plays an important role in identifying the anomalies. In essence, the more incoherent the column subsets of $\mathbf{R}$ are, the better recovery performance one can attain. An intriguing question left here to address in future research pertains to desirable network topologies giving rise to incoherent routing matrices.

Tracking routing changes. The measurement model in (5.8) has two time-varying attributes which challenge the identification of anomalies. The first one is missing measurement data arising from e.g., packet losses during the data collection process, and the second one pertains to routing changes due to e.g., network congestion or link failures. It is thus important to test whether the proposed online algorithm succeeds in tracking these changes.

As discussed earlier, missing data are sampled uniformly at random. To assess the impact of routing changes on the recovery performance, a simple probabilistic model is adopted where each time instant a single link fails, or, returns to the operational state. Let $\mathbf{\Phi}$ denote the adjacency matrix of the network graph G, where $[\mathbf{\Phi}]_{i,j} = 1$ if there exists a physical link joining nodes i and j, and zero otherwise. Similarly, the active links involved in routing the data at time t are represented by the effective adjacency matrix $\mathbf{\Phi}_t^{\text{eff}}$. At time instant $t+1$, a biased coin is tossed with small success probability α, and one of the links, say $(i,j) \in \mathbf{\Phi}_t^{\text{eff}}$, is chosen uniformly at random and removed from G while ensuring that the network remains connected. Likewise, an edge $(\ell,k) \in \mathbf{\Phi} \backslash \mathbf{\Phi}_t^{\text{eff}}$ is added with the same probability α. The resulting adjacency matrix is then $\mathbf{\Phi}_{t+1}^{\text{eff}} = \mathbf{\Phi}_t^{\text{eff}} + \mathbb{1}_{\{b_{1,t}\}}\mathbf{e}_\ell\mathbf{e}_k' - \mathbb{1}_{\{b_{1,t}\}}\mathbf{e}_i\mathbf{e}_j'$, where the indicator function $\mathbb{1}_{\{x\in\mathcal{X}\}}$ equals one when $x \in \mathcal{X}$, and zero otherwise; and $b_{1,t}, b_{2,t} \sim \text{Ber}(\alpha)$ are i.i.d. Bernoulli random variables. The minimum hop-count algorithm is then applied to $\mathbf{\Phi}_{t+1}^{\text{eff}}$, to update the routing matrix $\mathbf{R}_{t+1}$. Note that $\mathbf{R}_{t+1} = \mathbf{R}_t$ with probability $(1-\alpha)^2$.

The performance is tested here for fast and slowly varying routing corresponding to $\alpha = 0.1$ and $\alpha = 0.01$, respectively, when $\beta = 0.9$. A metric of interest is the average square error in estimating the anomalies, namely $e_t^a := \frac{1}{t}\sum_{i=1}^t \|\hat{\mathbf{a}}_i - \mathbf{a}_i\|_2^2$, and the link traffic, namely $e_t^x := \frac{1}{t}\sum_{i=1}^t \|\hat{\mathbf{x}}_i - \mathbf{x}_i\|_2^2$. Fig. 5.5 (a) plots the average estimation error for various noise variances and amounts of missing data. The estimation error decreases quickly and after learning the subspace it becomes almost invariant. To evaluate the support recovery performance of the online estimator, define the average detection and false alarm rate

$$P_{\text{D}} := \frac{\sum_{\tau=1}^t \sum_{f=1}^F \mathbb{1}_{\{\hat{a}_{f,\tau}\geq 0.1, a_{f,\tau}\geq 0.1\}}}{\sum_{\tau=1}^t \sum_{f=1}^F \mathbb{1}_{\{a_{f,\tau}\geq 0.1\}}}, \qquad P_{\text{FA}} := \frac{\sum_{\tau=1}^t \sum_{f=1}^F \mathbb{1}_{\{\hat{a}_{f,\tau}\geq 0.1, a_{f,\tau}\leq 0.1\}}}{\sum_{\tau=1}^t \sum_{f=1}^F \mathbb{1}_{\{a_{f,\tau}\leq 0.1\}}}. \tag{5.24}$$

Inspecting Fig. 5.5 (b) one observes that for $\alpha = 0.01$ and $\pi = 0.8$, increasing the noise variance from 10^{-5} to 10^{-2} lowers the detection probability by 10%. Moreover, when $\sigma = 10^{-5}$ and $\alpha = 0.01$, dropping 20% of the observations renders the estimator misdetect 11% more anomalies. The routing changes from $\alpha = 0.01$ to $\alpha = 0.1$ when $\sigma = 10^{-5}$ and $\pi = 0.8$ comes with an adverse effect of about 6% detection-rate decrease. For a few representative network links and flows Fig. 5.5 (c) and (d) illustrate how Algorithm 10

tracks the anomalies and link-level traffic. Note that in Fig. 5.5 (c) link 12 is dropped for the time period $t \in [220, 420]$, and thus the traffic level becomes zero. The flows being carried over link 31 are also varying due to routing changes, which occur at time instants $t = 220, 940$ when the traffic is not tracked accurately.

5.6.2 Real-network data tests

Internet-2 network example. Real data including OD flow traffic levels are collected from the operation of the Internet-2 network (Internet backbone network across USA) [2], shown in Fig. 5.6. Flow traffic levels are recorded every 5-minute intervals, for a three-week operational period of Internet-2 during Dec. 8–28, 2008 [2]. Internet-2 comprises $N = 11$ nodes, $L = 41$ links, and $F = 121$ flows. Given the OD flow traffic measurements, the link loads in $\mathbf{Y}$ are obtained through multiplication with the Internet-2 routing matrix, which in this case remains invariant during the three weeks of data acquisition [2]. Even though $\mathbf{Y}$ is "constructed" here from flow measurements, link loads can be typically acquired from SNMP traces [137].

The available OD flows are incomplete due to problems in the data collection process. In addition, flows can be modeled as the superposition of "clean" plus anomalous traffic, i.e., the sum of some unknown "ground-truth" low-rank and sparse matrices $\mathcal{P}_{\Omega}(\mathbf{X}_0 + \mathbf{A}_0)$. Therefore, setting $\mathbf{R} = \mathbf{I}_F$ in (P1) one can first run the batch Algorithm 9 to estimate the "ground-truth" components $\{\mathbf{X}_0, \mathbf{A}_0\}$. The estimated $\mathbf{X}_0$ exhibits three dominant singular values, confirming the low-rank property of the nominal traffic matrix. To be on the conservative side, only important spikes with magnitude greater than the threshold level $50\|\mathbf{Y}\|_F/LT$ are retained as benchmark anomalies (nonzero entries in $\mathbf{A}_0$).

Comparison with PCA-based batch estimators [72], [158]. To highlight the merits of the batch estimator (P3), its performance is compared with the spatial PCA-based schemes reported in [72] and [158]. These methods capitalize on the fact that the anomaly-free traffic matrix has low-rank, while the presence of anomalies considerably increases the rank of $\mathbf{Y}$. Both algorithms rely on a two-step estimation procedure: (s1) perform PCA on the data $\mathbf{Y}$ to extract the (low-rank) anomaly-free link traffic matrix $\tilde{\mathbf{X}}$; and (s2) declare

anomalies based on the residual traffic $\tilde{\mathbf{Y}} := \mathbf{Y} - \tilde{\mathbf{X}}$. The algorithms in [158] and [72] differ in the way (s2) is performed. On its operational phase, the algorithm in [72] declares the presence of an anomaly at time t, when the projection of $\mathbf{y}_t$ onto the anomalous subspace exceeds a prescribed threshold. It is clear that the aforementioned method is unable to identify anomalous flows. On the other hand, the network anomography approach of [158] capitalizes on the sparsity of anomalies, and recovers the anomaly matrix by minimizing $\|\tilde{\mathbf{A}}\|_1$, subject to the linear constraints $\tilde{\mathbf{Y}} = \mathbf{R}\tilde{\mathbf{A}}$.

The aforementioned methods require a priori knowledge on the rank of the anomaly-free traffic matrix, and assume there is no missing data. To carry out performance comparisons, the detection rate will be adopted as figure of merit, which measures the algorithm's success in identifying anomalies across both flows and time instants. ROC curves are depicted in Fig. 5.7 (a), for different values of the rank required to run the PCA-based methods. It is apparent that the estimator (P3) obtained via Algorithm 9 markedly outperforms both PCA-based methods in terms of detection performance. This is somehow expected, since (P3) advocates joint estimation of the anomalies and the nominal traffic matrix. For an instance of $P_{\text{FA}} = 0.04$ and $P_{\text{D}} = 0.93$, Fig. 5.7 (b) illustrates the effectiveness of the proposed algorithm in terms of unveiling the anomalous flows and time instants.

Online operation. Algorithm 10 is tested here with the Internet-2 network data under two scenarios: with and without missing data. For the incomplete data case, a randomly chosen subset of link counts with cardinality $0.15 \times LT$ is discarded. The penalty parameters are tuned as $\lambda_1 = 0.7$ and $\lambda_* = 1.4$. The evolution of the average anomaly and traffic estimation errors, and average detection and false alarm rates are depicted in Fig. 5.8 (a), (b), respectively. Note how in the case of full-data, after about a week the traffic subspace is accurately learned and the detection (false alarm) rates approach the values 0.72 (0.011). It is further observed that even with 15% missing data, the detection performance degrades gracefully. Finally, Fig. 5.8(c)[(d)] depicts how three representative link traffic levels [OD flow anomalies] are accurately tracked over time.

5.7 Concluding remarks

An online algorithm is developed in this paper to perform a critical network monitoring task termed *dynamic anomalography*, meaning to unveil traffic volume anomalies in backbone networks adaptively. Given link-level traffic measurements (noisy superpositions of OD flows) acquired sequentially in time, the goal is to construct a *map* of anomalies in *real time*, that summarizes the network 'health state' along both the flow and time dimensions. Online algorithms enable tracking of anomalies in nonstationary environments, typically arising due to to e.g., routing changes and missing data. The resultant online schemes offer an attractive alternative to batch algorithms, since they scale gracefully as the number of flows in the network grows, or, the time window of data acquisition increases. Comprehensive numerical tests with both synthetic and real network data corroborate the effectiveness of the proposed algorithms and their tracking capabilities, and show that they outperform existing workhorse approaches for network anomaly detection.

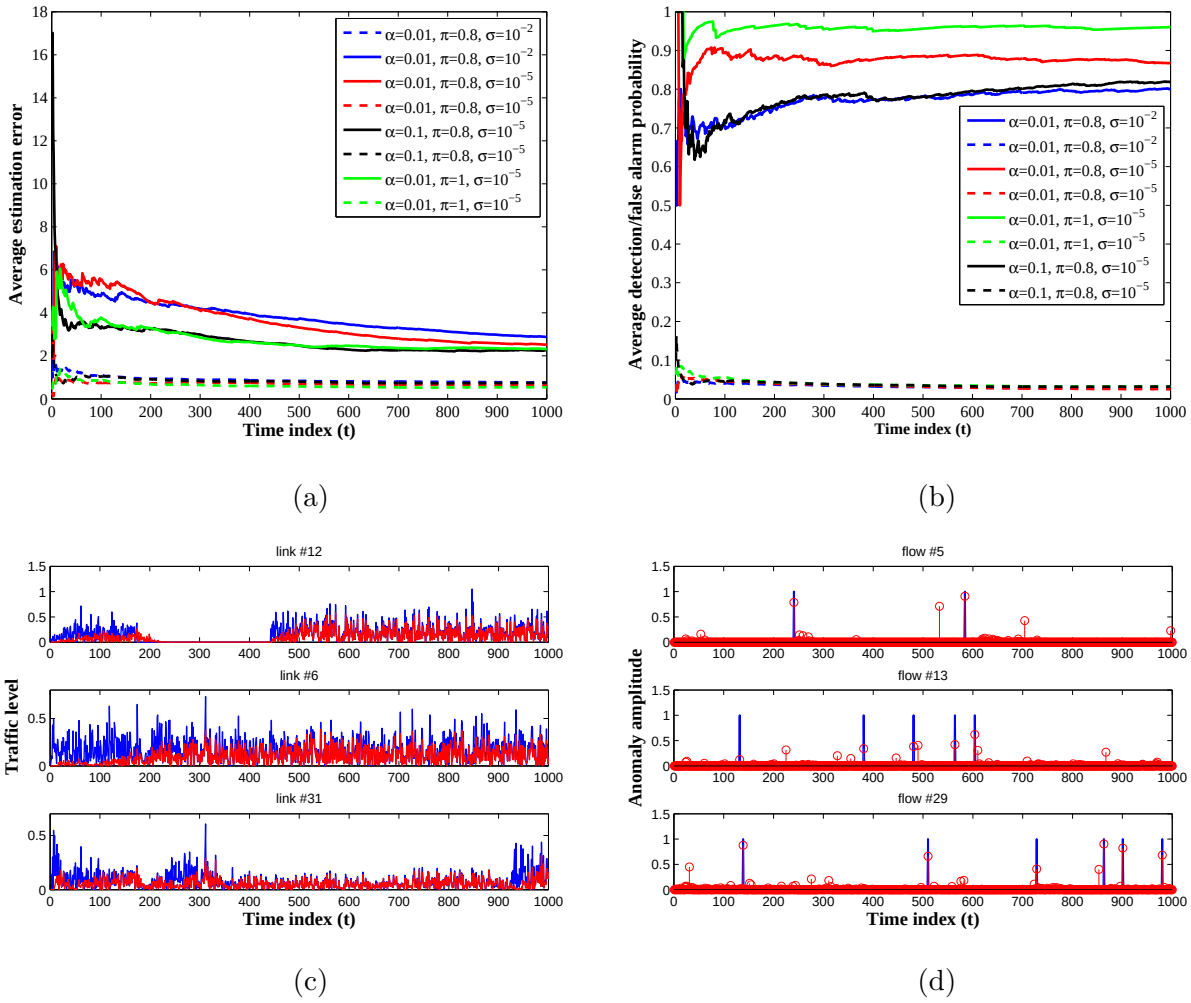
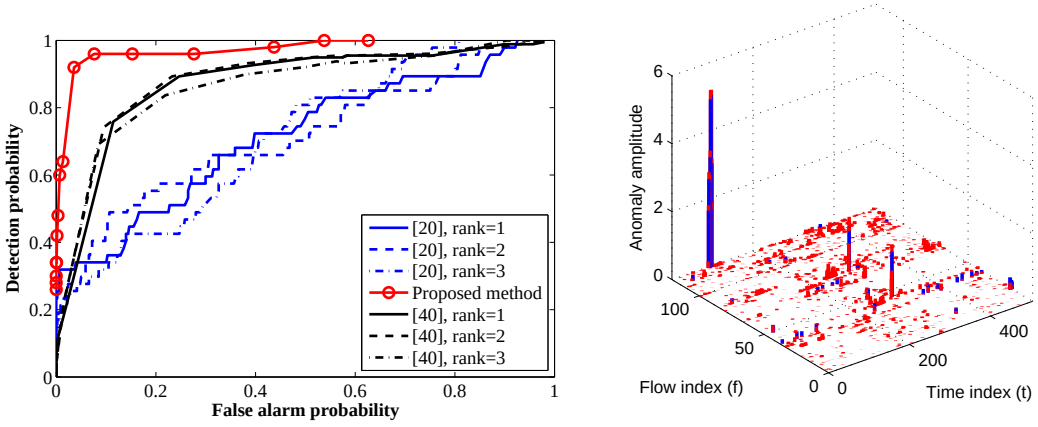


Figure 5.5: Tracking routing changes for $p = 0.005$. (a) Evolution of average anomaly (dotted) and traffic (solid) estimation errors. (b) Evolution of average detection (solid) and false alarm (dotted) rates. (c) Estimated (red) versus true (blue) link traffic for three representative links. (d) Estimated (circle markers) versus true (solid) anomalies for three representative flows when $\pi = 0.8$, $\sigma = 10^{-5}$, and $\alpha = 0.01$.

Figure 5.6: Internet-2 network topology graph.



(a) (b)

Figure 5.7: Performance of the batch estimator for Internet-2 network data. (a) ROC curves of the proposed versus the PCA-based methods. (b) Amplitude of the true (blue) and estimated (red) anomalies for $P_{\text{FA}} = 0.04$ and $P_{\text{D}} = 0.93$.

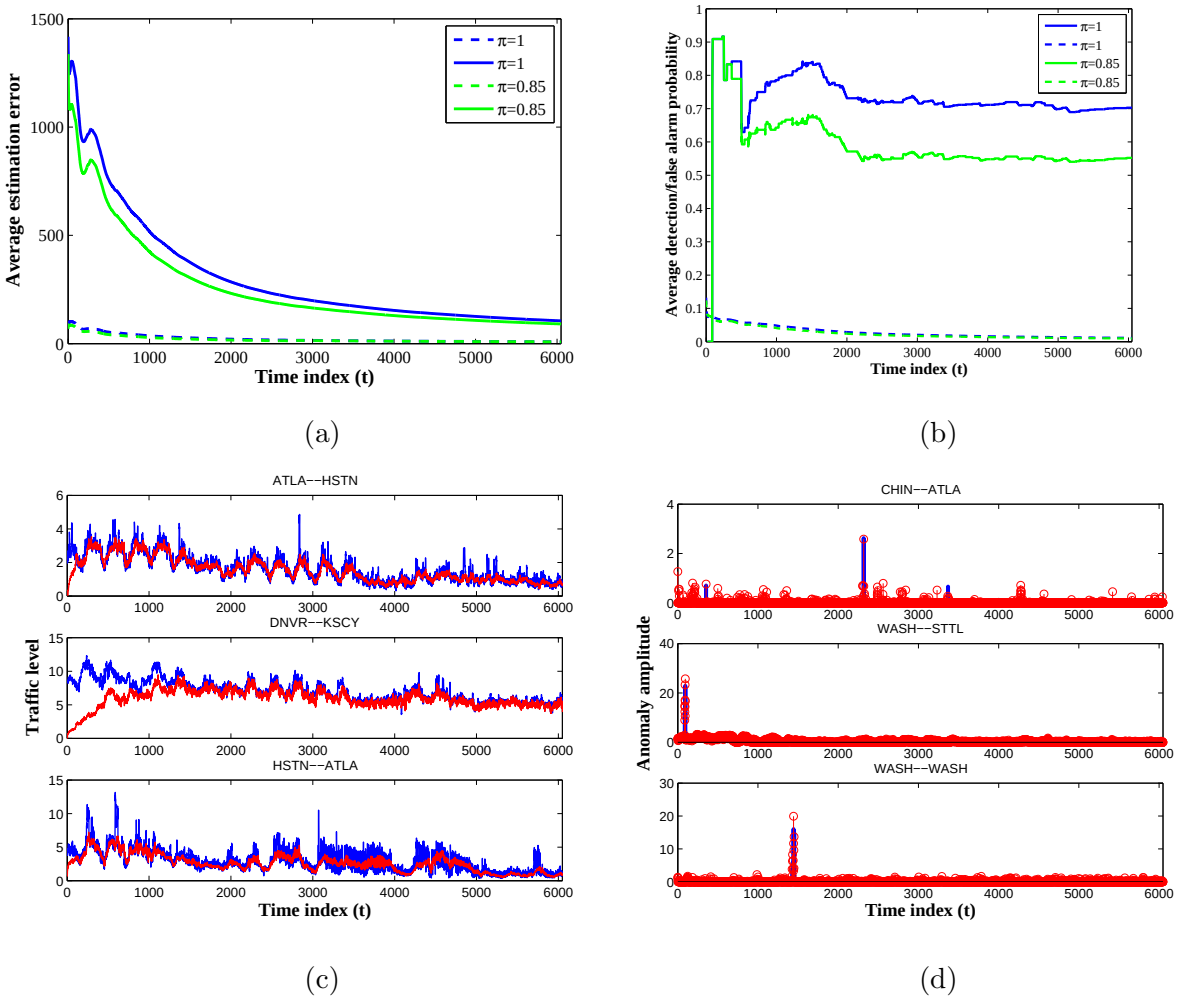


Figure 5.8: Performance of the online estimator for Internet-2 network data. (a) Evolution of average anomaly (dotted) and traffic (solid) estimation errors. (b) Evolution of average detection (solid) and false alarm (dotted) rates. (c) Estimated (red) versus true (blue) link traffic for thee representative links. (d) Estimated (circle markers) versus true (solid) anomalies for three representative flows when $\pi = 0.85$.

Chapter 6

Big Data Tensor Subspace Learning: Applications to Dynamic MRI

6.1 Introduction

With sensors continuously collecting massive amounts of information, this is undoubtedly an era of 'data deluge.' Learning from large volumes of data is expected to bring groundbreaking advances in science and engineering along with consequent improvements in quality of life. Magnetic resonance imaging (MRI) is among the principal technologies in this scientific revolution. Since its inception in the 70s [75, 90], MRI has emerged as a premier tool for biomedical imaging, allowing visualization of not only the structural and functional, but also of the physiological information of living subjects at both macroscopic and microscopic levels unreachable by human vision [78]. In recent years, MRI has also shown tremendous potential for dynamic processing. Through different protocols, dynamic MRI is able to provide images of tissues, perfusion, diffusion, spectroscopy, and susceptibility, both qualitatively as well as quantitatively in 3D high resolution and in real time for every patient. The abundance of such disparate big MRI data across time offers unprecedented opportuni-

ties to understand, diagnose, and treat diseases. Nevertheless, with such big blessings come big challenges. Dynamic MRI data are acquired in the spatial-frequency domain across time (called k-t space). Acquisition is often slow, allowing only a limited amount of data to be collected per time slot to ensure that motion is frozen. Data contain also corrupted measurements due to noise and system imperfections. Consequently, only low-resolution dynamic images can be acquired by state-of-the-art MRI scanners as dictated by Nyquist's sampling theory.

A great deal of research has been carried out over the last decades to accelerate the MRI scanning process. In essence, any piece of work to reconstruct the ground-truth images from undersampled data in one way or another exploits the spatial and/or temporal correlations of MR images. The advent of compressive sampling (CS) moved attention towards leveraging the parsimonious nature of MR images by means of sparsity. The celebrated work of [84] leverages the fact that MR images are well represented with only a small fraction of Fourier or Wavelet coefficients, and hence it is sparse across certain Fourier or Wavelet dictionaries. The image can then be simply reconstructed from undersampled k-space data via ℓ_1-norm minimization, where in certain cases attains order-three accelerations relative to the Nyquist sampling; see also [115] which further accelerates [84] by learning a sparsifying dictionary adaptive to data, that is able to take the local image features into account; and also [165] for a Bayesian approach developed to cope with dynamics by using approximate message passing. Low-rank models have also recently gained popularity; a few noteworthy representatives more relevant to the present study include [55, 77, 81, 112, 140, 142, 157]. For a series of dynamic MR images, treated as columns of a matrix, [112] and [140] models the background as a low-rank matrix, and the moving part as a sparse weight vector over a proper dictionary. Nuclear and ℓ_1-norm regularization are deployed along the lines of [97, 163] to procure the reconstruction.

To the best of our knowledge past work on dynamic MRI treats every image as a vector, whereas here images are modeled as multidimensional arrays with a matrix or tensor structure. In general, one can benefit from the tensor representation of an image sequence because it preserves spatial structure, and allows other dimensions such as those corresponding to

patients and coils in parallel MRI apart from the x-y coordinates and time. The present study envisions the dynamic MR images as a tensor factorized into multiple matrices based on the CP/PARAFAC decomposition; see e.g., [68]. The advocated model further builds on the fact that MR images are typically a superposition of a slowly-varying or stationary background and a foreground or innovation that pertains to nonstationarities arising due to motion or contrast variations. The underlying tensor is then well approximated with a low-rank plus a sparse tensor, with the low rank accounting for background and the sparsity for the foreground. Bearing this in mind, the present study brings forth novel reconstruction schemes to infer the low rank and sparse tensor components from highly undersampled k-space data. The sought schemes are systematically formulated as optimization programs leveraging the sparsity and low rank effected by the tensor rank and ℓ_1-norm regularization.

Stochastic alternating minimization is then adopted to develop online solvers of the sought programs, where the updates occur per acquisition of a new datum. In essence, in each update, the solver recursively learns the latent tensor subspace pertaining to the low-rank component, and subsequently projects the new acquired image onto the subspace to infer the sparse foreground as the residual. The corresponding iterates are provably convergent. The resultant algorithms are also highly parallelizable, and thus attractive for MRI scans with high temporal and spatial resolution. For the parallel dynamic MRI task, reconstruction schemes are introduced that either interpolate the misses in the k-space or in the image domain pursuing a tomographic approach. The proposed schemes offer real-time reconstruction of MR images 'on the fly.' Furthermore, the intermediate subspace estimates, returned by the online updates, can be utilized to devise adaptive sketching strategies that can further accelerate the acquisition process by ranking the k-space data according to their importance level. Preliminary numerical tests on cardiac cine MRI of a human dataset show ten-fold acceleration relative to Nyquist sampling, with only a few passes over the data. Last but not least, the scope of the proposed framework goes beyond the dynamic MRI task, and can indeed cater to other 'Big Data' inference tasks.

The rest of this paper is organized as follows. Section II introduces preliminaries on tensor PARAFAC and rank regularization and advocates a model to arrive at a generic

optimization formalism for reconstruction. Section III then develops recursive solvers to learn the tensor subspace. Adaptive sketching strategies to further accelerate the MR acquisition process are proposed in Section IV. Subsequently, Section V focuses on the dynamic parallel MRI application, where two reconstruction schemes are proposed to either interpolate misses in the k-space or in the image domain in a tomographic manner. Finally, real-data tests are reported in Section VI, while conclusions are drawn in Section VII.

6.2 Preliminaries and Problem Statement

As modern and massive datasets become increasingly complex and heterogeneous, in many application setups one encounters data structures indexed by three or more variables giving rise to a tensor, instead of just two variables as in the matrix settings. A few examples of time-indexed, tensor data include [5]: (i) images acquired in parallel MRI across various coils, as well as snapshots across time and patients, collected in a five-dimensional array with (phase encoding, frequency encoding, coil, time, patient); and (ii) Electroencephalograms (EEGs), where the signal of each electrode is a time-frequency matrix; thus, data from multiple channels is three-dimensional (temporal, spectral, and spatial) and may be incomplete if electrodes become loose or disconnected for a period of time.

6.2.1 PARAFAC decomposition and low-rank tensors

For multiple, say $M \geq 2$, vectors $\mathbf{a}_i \in \mathbb{R}^{N_i \times 1}$, the outer product $\mathbf{a}_1 \circ \ldots \circ \mathbf{a}_M$ is a $N_1 \times \ldots \times N_M$ rank-one M-way array with $(n_1, \ldots, n_M)$-th entry given by $\Pi_{i=1}^{M} \mathbf{a}_i(n_i)$. Note that this comprises a generalization of the matrix case with $M = 2$, where $\mathbf{a}_1 \circ \mathbf{a}_2 = \mathbf{a}_1 \mathbf{a}_2^{'}$ is a rank-one matrix. The rank of a tensor $\underline{\mathbf{X}}$ is defined as the minimum number of outer products required to synthesize $\underline{\mathbf{X}}$. The PARAFAC model is arguably the most basic tensor model because of its direct relationship to tensor rank. Specifically, it is natural to form a *low-rank approximation* of tensor $\underline{\mathbf{X}} \in \mathbb{R}^{N_1 \times \ldots \times N_M}$ as

$$\underline{\mathbf{X}} \approx \sum_{r=1}^{R} \mathbf{a}_r^{(1)} \circ \ldots \circ \mathbf{a}_r^{(M)}. \tag{6.1}$$

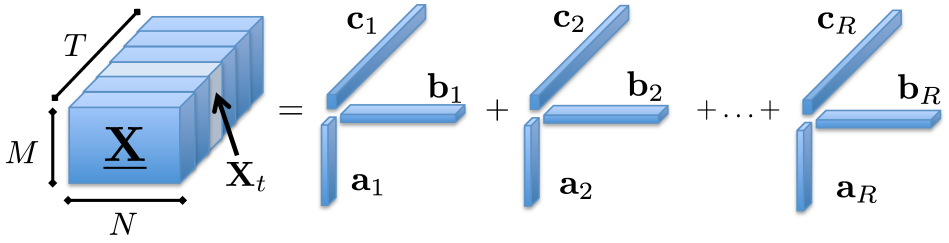


Figure 6.1: A rank-R PARAFAC decomposition of the three-way tensor $\underline{\mathbf{X}}$.

When the approximation is exact, (6.1) is the PARAFAC decomposition of $\underline{\mathbf{X}}$. Accordingly, the minimum value R for which the exact decomposition is possible is (by definition) the rank of $\underline{\mathbf{X}}$. Different from the matrix case, there is no straightforward algorithm to determine the rank of a given tensor, a problem that has been shown to be NP-hard [68]. For a survey of algorithmic approaches to obtain approximate PARAFAC decompositions, the reader is referred to [68].

With reference to (6.1), introduce the factor matrices $\mathbf{A}_i := [\mathbf{a}_1^{(i)}, \ldots, \mathbf{a}_R^{(i)}] \in \mathbb{R}^{N_i \times R}$, $i \in [M]$. Let $\underline{\mathbf{X}}_\ell^{(k)}$, $\ell \in [N_k]$ denote the ℓ-th 'slab' of $\underline{\mathbf{X}}$ along its k-th mode, such that $\underline{\mathbf{X}}_\ell^{(k)}(n_1, \ldots, n_{k-1}, n_{k+1}, \ldots, n_M) = \underline{\mathbf{X}}(n_1, \ldots, \ell, \ldots, n_M)$. The ℓ-th slice can then be expressed as

$$\underline{\mathbf{X}}_\ell^{(k)} = \sum_{r=1}^{R} \boldsymbol{\gamma}_{\ell,r}^{(k)} \mathbf{a}_r^{(1)} \circ \ldots \circ \mathbf{a}_r^{(k-1)} \circ \mathbf{a}_r^{(k+1)} \circ \ldots \circ \mathbf{a}_r^{(M)}, \quad \ell = 1, \ldots, N_k \tag{6.2}$$

where $\boldsymbol{\gamma}_\ell \in \mathbb{R}^R$ denotes the ℓ-th row of $\mathbf{A}_k$ (recall that $\mathbf{a}_r^{(k)}$ instead represents the r-th column of $\mathbf{A}_k$). To gain intuition, imagine a data cube with $M = 3$, where the slices form matrices, and for instance the ℓ-th slice across the tube dimension is expressed as $\mathbf{X}_\ell = \sum_{r=1}^{R} \gamma_{\ell,r}^{(3)} \mathbf{a}_r^{(1)} \mathbf{a}_r^{(2)'}$. It is apparent that a slice $\mathbf{X}_\ell$ can be represented as a linear combination of R rank-one matrices $\{\mathbf{a}_r^{(1)} \mathbf{a}_r^{(2)'}\}_{r=1}^R$, which constitute the bases for the tensor fiber subspace. In the same manner, one can argue that for a general M-order tensor the rank-one tensors $\{\mathbf{a}_r^{(1)} \circ \ldots \circ \mathbf{a}_r^{(k-1)} \circ \mathbf{a}_r^{(k+1)} \circ \ldots \circ \mathbf{a}_r^{(M)}\}_{r=1}^R$ form the bases for the tensor subspace along the k-th mode.

This study primarily aims at discovering this latent subspace that can be equivalently expressed in compact form by the matrices $\{\mathbf{A}_i\}_{i=1, i\neq k}^M$. This will be handy later. Given $\underline{\mathbf{X}}$, under some mild technical conditions, matrices $\{\mathbf{A}_i\}_{i=1}^M$ are unique up to a common column

permutation and scaling (meaning PARAFAC is identifiable for $M \geq 3$); see e.g. [15,69]. It is worth commenting that a factor matrix $\mathbf{A}_i,\ i \in [M]$, is not necessarily orthogonal, and it may even consist of linearly dependent columns. With these attractive features, PARAFAC has become the model of choice when one is primarily interested in revealing latent structure in multiway data arrays. Considering the analysis of a dynamic social network for instance, each of the rank-one factors could correspond to communities that e.g., persist or form and dissolve dynamically across time.

PARAFAC due to its connection to tensor rank can be used to postulate low-rank tensor models. However, as mentioned earlier even finding the tensor rank is an NP-hard problem. Parallel to the matrix nuclear-norm, tractable surrogates can be adopted for the tensor CP-rank that approximates the rank through the norm of factors. One such surrogate for $M = 3$ is proposed in our companion work [13]. One can readily generalize the results of [13] to M-way arrays to arrive at

$$\mathcal{Q}(\underline{\mathbf{X}}) := \min_{\{\mathbf{A}_i \in \mathbb{R}^{N_i \times R}\}_{i=1}^{M}} \ \frac{1}{M}\sum_{i=1}^{M} \|\mathbf{A}_i\|_F^2 \ \ rms.to \ \ \mathbf{X} = \sum_{r=1}^{R} \mathbf{a}_r^{(1)} \circ \ldots \circ \mathbf{a}_r^{(M)} \tag{6.3}$$

Moreover, broadening the scope of [13] it can be shown that using $\mathcal{Q}(\underline{\mathbf{X}})$ as regularizer induces low rank as formalized next.

Lemma 6.1 *If $\sigma_r = \Pi_{m=1}^{M} \|\mathbf{a}_r^{(m)}\|,\ r \in [R]$, denotes r-th tensor singular value of $\underline{\mathbf{X}}$, it holds that*

$$\mathcal{Q}(\underline{\mathbf{X}}) = \left(\sum_{r=1}^{R} |\sigma_r|^{2/M} \right)^{2/M}.$$

It is apparent that the surrogate $\mathcal{Q}(\underline{\mathbf{X}})$ resembles the nonconvex $\ell_{2/M}$-norm that effects sparsity across singular values with even higher rate than the traditional ℓ_1-norm. Moreover, as expected for the matrix case with $M = 2$, the rank regularizer $\mathcal{Q}$ coincides with the convex nuclear norm of the matrix.

6.2.2 Low-rank plus sparse tensor

In various application domains, the physical process of interest represented by a tensor can be viewed as the superposition of background and foreground components. The background

accounts for slowly varying or stationary behaviors, and the foreground captures the innovations or nonstationary variations of the process. Supposing that innovations occur rarely, one can model the process at time t as $\underline{\mathbf{X}}_t = \underline{\mathbf{L}}_t + \underline{\mathbf{S}}_t$, where the stationary process $\{\underline{\mathbf{L}}_t\}$ lives in a low-dimensional subspace of tensors, call it $\mathcal{L}$, and the foreground component $\underline{\mathbf{S}}_t$ contains only a few nonzero elements. With this model in mind, finding the latent components $\{\underline{\mathbf{L}}_t, \underline{\mathbf{S}}_t\}$ is an arduous task mainly because (i) only a limited number of observations are available relative to the ambient signal dimension, and (ii) the observations are typically streaming over time. Let $\{\mathbf{y}_t \in \mathbb{R}^{L_t}\}$ denotes the vector data stream that relates to the process of interest through the data per entry linear model

$$y_t^{(\ell)} = \langle \underline{\mathbf{L}}_t + \underline{\mathbf{S}}_t, \underline{\mathbf{W}}_t^{(\ell)} \rangle + v_t^{(\ell)}, \quad \ell = 1, \ldots, L_t \tag{6.4}$$

where, the projection tensor $\underline{\mathbf{W}}_t^{(\ell)} \in \mathbb{R}^{N_1 \times \ldots \times N_{M-1}}$ sketches the signal features, and the noise term $v_t^{(\ell)}$ represents the errors and unmodeled dynamics. This is the model of choice in several modern big data applications such as dynamic MRI, where the ground-truth sequence of images forms a three-way data cube, and per time instant t a small subset of k-space data, denoted by $[\mathcal{F}(\mathbf{X}_t)]_{i_\ell, j_\ell}$, $(i_\ell, j_\ell) \in \Omega_t$ are acquired ($\mathcal{F}$ here is the Fourier transform operator that is special choice of the projection tensor $\underline{\mathbf{W}}_t^{(\ell)}$ in (6.4)); see also Section 6.4 for further details.

Assume temporarily that one has only access to a batch of observations (6.4) over the time horizon $t \in [1, T]$ with T time slots. Collect the $(M-1)$-way signals $\{\underline{\mathbf{L}}_t\}_{t=1}^T$ into a larger M-way tensor $\underline{\mathbf{L}}$ with the M-th mode representing time; and form $\underline{\mathbf{S}}$ likewise. Low dimensionality of $\mathcal{L}$ implies $\underline{\mathbf{L}}$ is of low CP-rank. The innovation tensor $\underline{\mathbf{S}}$ is also sparse. All in all, we wish to identify $\{\underline{\mathbf{L}}_t, \underline{\mathbf{S}}_t\}_{t=1}^T$ given the observations $\{\mathbf{y}_t\}_{t=1}^T$ and the corresponding sketching weights $\{\underline{\mathbf{W}}_t^{(\ell)}\}_{t=1}^T$, assuming $\underline{\mathbf{L}}$ and $\underline{\mathbf{S}}$ are low-rank and sparse tensors, respectively. Upon defining $\boldsymbol{\gamma}_t \in \mathbb{R}^R$ as the t-th row of $\mathbf{A}_M[t]$, a natural estimator

for $(\hat{\underline{\mathbf{L}}}, \hat{\underline{\mathbf{S}}})$ is

$$\text{(P1)} \quad (\hat{\underline{\mathbf{L}}}, \hat{\underline{\mathbf{S}}}) = \arg \min_{\{\{\mathbf{A}_m\}_{m=1}^M, \{\boldsymbol{\gamma}_t\}_{t=1}^T, \underline{\mathbf{S}}\}} \frac{1}{2} \sum_{t=1}^{T} \sum_{\ell=1}^{L_t} \left(y_t^{(\ell)} - \langle \underline{\mathbf{L}}_t + \underline{\mathbf{S}}_t, \underline{\mathbf{W}}_t^{(\ell)} \rangle\right)^2 + \frac{\lambda_*}{2} \sum_{m=1}^{M} \|\mathbf{A}_m\|_F^2 + \lambda_1 \|\underline{\mathbf{S}}\|_1$$

$$\text{s. to} \qquad \underline{\mathbf{L}}_t = \sum_{r=1}^{R} \gamma_{t,r} \mathbf{a}_r^{(1)} \circ \ldots \circ \mathbf{a}_r^{(M-1)}, \quad t = 1, \ldots, T$$

which fits the data to the postulated model (6.4) in the least-squares (LS) sense, and promotes low rank through the regularizer $\sum_{m=1}^{M} \|\mathbf{A}_m\|_F^2$, and sparsity via the ℓ_1-norm regularizer $\|\underline{\mathbf{S}}\|_1 = \sum_{n_1,\ldots,n_M} |s_{n_1,\ldots,n_2}|$. Here, λ_1 and λ_* tune the desired rank and sparsity levels. Note all the data and optimization variables in (P1) are assumed real-valued.

In the setup of interest to big data applications, the ambient dimensions $\{N_m\}_{m=1}^{M-1}$ are quite large, and the tensor slices are streaming over time, i.e., $N_M \to \infty$. Before delving into the analysis to devise online solvers of (P1) for streaming observations, a couple of noteworthy properties of (P1) will be useful. First, the rank regularization avoids the scaling ambiguity associated with the multilinear terms as formalized next.

Lemma 6.2 *Every stationary point of (P1) gives rise to a tensor subspace having basis vectors with common norm; that is,* $\|\mathbf{a}_r^{(m)}\| = \|\mathbf{a}_r^{(m')}\|, \forall m, m' \in [M-1]$, *and* $\forall r \in [R]$. *Equal norm basis vectors fix the scaling ambiguity inherent to the PARAFAC model.*

Proof: It readily follows after equating the gradient of (P1)'s objective w.r.t. $\mathbf{a}_r^{(m)}$ (see also (6.8)) to zero, and taking the inner product of both sides with $\mathbf{a}_r^{(m)}$. ■

Lemma 6.2 implies that the locally minimum factor matrices $\{\mathbf{A}_m\}_{m=1}^{M-1}$ associated with different tensor modes have identical Frobenius norms.

For large-scale datasets with a huge number of features $\prod_{m=1}^{M-1} N_m$ solving (P1) incurs prohibitive complexity and storage to run batch solvers. In addition, certain applications demand real-time processing upon acquisition of a new datum based on the past and current data, namely $\{y_\tau^{(\ell)}, \ell \in [L_t]\}_{\tau=1}^{t}$. In essence, (P1) involves $R(N_1 + \ldots + N_{M-1} + t)$ variables associated with the low-rank components plus $t \prod_{i=1}^{M-1} N_i$ coming from the sparse terms.

Apparently, growth of t can easily burden the storage and computational units. These obstacles press the need for recursive online solvers that can acquire the data sequentially and perform simple update tasks. The ensuing section introduces machinery to cope with the aforementioned hurdles to arrive at efficient online solvers.

6.3 Tensor Subspace Learning

As elaborated in Section 6.2.2 the low-rank component $\underline{\mathbf{L}}_t$ lies in a low-dimensional subspace $\mathcal{L} \subset \mathbb{R}^{N_1 \times \ldots \times N_{m-1}}$. With reference to CP rank, $\mathcal{L}$ is characterized by a small number of R rank-one $(M-1)$-way arrays $\{\mathbf{a}_r^{(1)} \circ \ldots \circ \mathbf{a}_r^{(M-1)}\}_{r=1}^R$, expressed in the compact matrix form $\{\mathbf{A}_m\}_{m=1}^{M-1}$. Learning these time-invariant factor matrices is the first step towards reconstructing the low-rank and sparse tensors of interest. With the streaming observations in mind, at t-th acquisition time one is given $T = t$ data snapshots, and accordingly with a slight abuse of notation one is motivated to recast (P1) in the separable form

$$\text{(P2)} \quad \min_{\{\mathbf{A}_m\}_{m=1}^{M-1}} \frac{1}{2t} \sum_{\tau=1}^{t} \min_{\boldsymbol{\gamma}_\tau, \underline{\mathbf{S}}_\tau} \left\{ \sum_{\ell=1}^{L_t} \Big(y_\tau^{(\ell)} - \sum_{r=1}^{R} \gamma_{\tau,r} \langle \mathbf{a}_r^{(1)} \circ \ldots \circ \mathbf{a}_r^{(M-1)}, \underline{\mathbf{W}}_t^{(\ell)} \rangle - \langle \underline{\mathbf{S}}_\tau, \underline{\mathbf{W}}_t^{(\ell)} \rangle \Big)^2 + \frac{\lambda_*}{2} \|\boldsymbol{\gamma}_\tau\|^2 + \lambda_1 \|\underline{\mathbf{S}}_\tau\|_1 \right\} + \frac{\lambda_*}{2t} \sum_{m=1}^{M-1} \|\mathbf{A}_m\|_F^2.$$

Apparently, finding the optimal solutions of the nonconvex program (P2) becomes computationally challenging especially for large values of M. Hence, one needs to devise valid approximations that can afford simple iterative updates while approaching the optimal solution. One such approximation for online rank minimization leveraging the separable nuclear-norm regularization (6.3) was introduced in [96] for the matrix case ($M = 2$), in the context of unveiling network anomalies, and in [98] for imputation of three-way tensors. Along these lines, online solvers are developed next for general $M \geq 3$ case.

6.3.1 Stochastic alternating minimization

Towards deriving a real-time, computationally efficient, and recursive solver of (P2), an alternating-minimization (AM) method is adopted in which iterations coincide with the

time-scale t of data acquisition. In accordance with (P2), consider the instantaneous regularized LS loss

$$f_t(\{\mathbf{A}_i\}_{i=1}^{M-1};\underline{\mathbf{S}}_t,\boldsymbol{\gamma}_t) := \frac{1}{2}\sum_{\ell=1}^{L_t}\Big(y_t^{(\ell)} - \sum_{r=1}^{R}\gamma_{t,r}\langle \mathbf{a}_r^{(1)}\circ\ldots\circ\mathbf{a}_r^{(M-1)},\underline{\mathbf{W}}_t^{(\ell)}\rangle - \langle\underline{\mathbf{S}}_t,\underline{\mathbf{W}}_t^{(\ell)}\rangle\Big)^2 + (\lambda_*/2t)\sum_{m=1}^{M-1}\|\mathbf{A}_m\|_F^2. \tag{6.5}$$

The iterative procedure adopted here consists of two major steps. The first step (S1) relies on recently updated subspace, namely $\{\hat{\mathbf{A}}_m[t-1]\}_{m=1}^{M-1}$ to solve the inner optimization which yields $\underline{\mathbf{S}}_t$ and $\boldsymbol{\gamma}_t$ through $(\hat{\underline{\mathbf{S}}}_t,\hat{\boldsymbol{\gamma}}_t) = \arg\min \quad f(\{\mathbf{A}_i\}_{i=1}^{M-1};\underline{\mathbf{S}}_t,\boldsymbol{\gamma}_t)$. In the second step (S2) the tensor subspace $\hat{\mathcal{L}}_t$ is updated by moving $\{\hat{\mathbf{A}}_m\}_{m=1}^{M-1}$ along the opposite direction of the gradient, namely $-\nabla f_t(\{\mathbf{A}_i\}_{i=1}^{M-1};\hat{\underline{\mathbf{S}}}_t,\hat{\boldsymbol{\gamma}}_t)$.

Towards (S1), introduce $z_t^{(\ell)} := y_t^{(\ell)} - \langle\underline{\mathbf{S}}_t,\underline{\mathbf{W}}_t^{(\ell)}\rangle$, and correspondingly the vector $\mathbf{z}_t\in\mathbb{R}^{L_t}$; define also the matrix $\boldsymbol{\Phi}_t := [\boldsymbol{\phi}_t^{(1)},\ldots,\boldsymbol{\phi}_t^{(L_t)}]'\in\mathbb{R}^{L_t\times R}$, where $[\boldsymbol{\phi}_t^{(\ell)}]_r := \langle\mathbf{a}_r^{(1)}\circ\ldots\circ\mathbf{a}_r^{(M-1)},\underline{\mathbf{W}}_t^{(\ell)}\rangle$. The projection of $\mathbf{z}_t$ onto the low-dimensional subspace $\hat{\mathcal{L}}_t$ is then obtained via solving the LS ridge-regression problem

$$\hat{\boldsymbol{\gamma}}_t = \arg\min_{\boldsymbol{\gamma}\in\mathbb{R}^R}\frac{1}{2}\|\mathbf{z}_t-\boldsymbol{\Phi}_t\boldsymbol{\gamma}\|^2 + \frac{\lambda_*}{2}\|\boldsymbol{\gamma}\|^2$$

which admits the closed-form $\hat{\boldsymbol{\gamma}}_t = (\boldsymbol{\Phi}_t'\boldsymbol{\Phi}_t+\lambda_*\mathbf{I}_R)^{-1}\boldsymbol{\Phi}_t'\mathbf{z}_t$, that linearly depends on the sparse component. To avoid $R\times R$ inversion, consider the SVD $\boldsymbol{\Phi}_t = \mathbf{U}_t\boldsymbol{\Sigma}_t\mathbf{V}_t$ to end up with $\hat{\boldsymbol{\gamma}}_t = \mathbf{V}_t\boldsymbol{\Sigma}_t^{-1}\mathbf{D}_t\mathbf{U}_t'\mathbf{z}_t$, where $\mathbf{D}_t\in\mathbb{R}^{R\times R}$ is a diagonal matrix with $[\mathbf{D}_t]_{i,i} = \sigma_i^2/(\sigma_i^2+\lambda_*)$. To obtain the sparse component one needs to first find $\mathbf{G}_t$ from the factorization $\mathbf{G}_t'\mathbf{G}_t := \mathbf{I}_{L_t} - \mathbf{U}_t\mathbf{D}_t\mathbf{U}_t'$. Upon defining the matrix $\boldsymbol{\Omega}_t := [\mathrm{vec}(\underline{\mathbf{W}}_t^{(1)}),\ldots,\mathrm{vec}(\underline{\mathbf{W}}_t^{(L_t)})]'\in\mathbb{R}^{L_t\times\prod_{m=1}^{M-1}N_m}$, the sparse tensor $\underline{\mathbf{S}}_t$ is the solution of the LASSO problem (see e.g., [138])

$$\hat{\underline{\mathbf{S}}}_t = \arg\min_{\mathbf{s}\in\mathbb{R}^{\prod_{m=1}^{M-1}N_m}}\quad \frac{1}{2}\|\mathbf{G}_t\mathbf{y}_t - \mathbf{G}_t\boldsymbol{\Omega}_t\mathbf{s}\|^2 + \lambda_1\|\mathbf{s}\|_1.$$

The second step (S2) deals with updating the factor matrices given the available projection coefficients $\{(\underline{\mathbf{S}}_\tau,\boldsymbol{\gamma}_\tau)\}_{\tau=1}^t$ via solving

$$\{\mathbf{A}_m[t]\}_{m=1}^{M-1} = \arg\min_{\{\mathbf{A}_m\}_{m=1}^{M-1}} C_t(\{\mathbf{A}_m\}_{m=1}^{M-1}) := \frac{1}{t}\sum_{\tau=1}^{t} f_\tau(\{\mathbf{A}_m\}_{m=1}^{M-1};\hat{\underline{\mathbf{S}}}_\tau,\hat{\boldsymbol{\gamma}}_\tau). \tag{6.6}$$

Apparently, this gives rise to a nonconvex program for $M \geq 3$ due to the multilinear terms in the LS cost, and it is challenging to solve optimally. To mitigate this computational challenge, a proper quadratic approximant of f_t is

$$\tilde{f}_t(\{\mathbf{A}_m\}_{m=1}^{M-1}; \hat{\mathbf{S}}_t, \hat{\boldsymbol{\gamma}}_t) = f_t(\{\mathbf{A}_m[t-1]\}_{m=1}^{M-1}; \hat{\mathbf{S}}_t, \hat{\boldsymbol{\gamma}}_t) + \sum_{m=1}^{M-1} \langle \nabla_{\mathbf{A}_m} f_t(\{\mathbf{A}_m[t-1]\}_{m=1}^{M-1}; \underline{\hat{\mathbf{S}}}_t, \hat{\boldsymbol{\gamma}}_t), \mathbf{A}_m - \mathbf{A}_m[t-1] \rangle + \frac{\alpha_t}{2} \sum_{m=1}^{M-1} \|\mathbf{A}_m - \mathbf{A}_m[t-1]\|_F^2$$

where $\alpha_t \geq \max_m \left\{ \sigma_{\max}\left[\nabla^2_{\mathbf{A}_m} f_t(\{\mathbf{A}_m[t-1]\}_{m=1}^{M-1}; \hat{\mathbf{S}}_t, \hat{\boldsymbol{\gamma}}_t)\right] \right\}$. With regards to the surrogate $\tilde{f}_t$, it is useful to recognize that it is locally tight, meaning that (i) $f_t(\{\mathbf{A}_m[t-1]\}_{m=1}^{M-1}; \hat{\mathbf{S}}_t, \hat{\boldsymbol{\gamma}}_t) = \tilde{f}_t(\{\mathbf{A}_m[t-1]\}_{m=1}^{M-1}; \hat{\mathbf{S}}_t, \hat{\boldsymbol{\gamma}}_t)$, and similarly $\nabla f_t(\{\mathbf{A}_m[t-1]\}_{m=1}^{M-1}; \hat{\mathbf{S}}_t, \hat{\boldsymbol{\gamma}}_t) = \nabla \tilde{f}_t(\{\mathbf{A}_m[t-1]\}_{m=1}^{M-1}; \hat{\mathbf{S}}_t, \hat{\boldsymbol{\gamma}}_t)$; and (ii) it upper bounds f_t, that is $f_t(\{\mathbf{A}_m\}_{m=1}^{M-1}; \hat{\mathbf{S}}_t, \hat{\boldsymbol{\gamma}}_t) \leq \tilde{f}_t(\{\mathbf{A}_m\}_{m=1}^{M-1}; \hat{\mathbf{S}}_t, \hat{\boldsymbol{\gamma}}_t)$, for all $\mathbf{A}_m \in \mathbb{R}^{N_m \times R}$, and $m \in [M-1]$.

Apart from tightness, separability across factors is another nice feature of $\tilde{f}_t$ because it enables parallel implementation. Plugging in the approximation $\tilde{f}_t$ into the cost C_t, yields $\tilde{C}_t := (1/t)\sum_{\tau=1}^{t} \tilde{f}_\tau$, the minimizer of which is obtained (after equating the gradient to zero) as

$$\mathbf{A}_m[t] = \frac{1}{\bar{\alpha}_t} \sum_{\tau=1}^{t} \alpha_\tau \left\{ \mathbf{A}_m[\tau-1] - \alpha_\tau \nabla_{\mathbf{A}_m} f_\tau(\{\mathbf{A}_m[\tau-1]\}_{m=1}^{M-1}; \hat{\mathbf{S}}_\tau, \hat{\boldsymbol{\gamma}}_\tau) \right\}$$

where $\bar{\alpha}_t := \sum_{\tau=1}^{t} \alpha_\tau$. After rearranging one arrives at the recursion

$$\begin{aligned}
\mathbf{A}_m[t] &= \big(\frac{1}{\bar{\alpha}_t}\big) \underbrace{\sum_{\tau=1}^{t-1} \alpha_\tau \left\{ \mathbf{A}_m[\tau-1] - \alpha_\tau^{-1} \nabla_{\mathbf{A}_m} f_\tau(\{\mathbf{A}_m[\tau-1]\}_{m=1}^{M-1}; \hat{\mathbf{S}}_\tau, \hat{\boldsymbol{\gamma}}_\tau) \right\}}_{:=\bar{\alpha}_{t-1}\mathbf{A}_m[t-1]} \\
&\quad + \big(\frac{\alpha_t}{\bar{\alpha}_t}\big) \left\{ \mathbf{A}_m[t-1] - \alpha_t \nabla_{\mathbf{A}_m} f_t(\{\mathbf{A}_m[t-1]\}_{m=1}^{M-1}; \hat{\mathbf{S}}_t, \hat{\boldsymbol{\gamma}}_t) \right\} \\
&= \mathbf{A}_m[t-1] - (\bar{\alpha}_t)^{-1} \nabla_{\mathbf{A}_m} f_t(\{\mathbf{A}_m[t-1]\}_{m=1}^{M-1}; \hat{\mathbf{S}}_t, \hat{\boldsymbol{\gamma}}_t) \quad m \in [M-1]. \qquad (6.7)
\end{aligned}$$

Interestingly, (6.7) is nothing but a single stochastic gradient-descent step.

The gradient is separable across columns of $\mathbf{A}_m$. Considering it w.r.t. each basis vector

Algorithm 12 Online sparsity-regularized tensor subspace learning

input $\{y_t^{(\ell)}, \underline{\mathbf{W}}_t^{(\ell)},\ \ell \in [L_t]\}_{t=1}^{\infty}, \{\mu_t\}_{t=1}^{\infty}, \lambda_*, \lambda_1, R, M.$

initialize $\{\mathbf{A}_m[1]\}_{m=1}^{M}$ at random.

for $t = 1, 2, \ldots$ **do**

Projection coefficients update

$[\mathbf{\Phi}_t]_{\ell,r} = \langle \mathbf{a}_r^{(1)} \circ \ldots \circ \mathbf{a}_r^{(M-1)}, \underline{\mathbf{W}}_t^{(\ell)} \rangle,\ \ \mathbf{\Phi}_t = \mathbf{U}_t \mathbf{\Sigma}_t \mathbf{V}_t^{'}$

$\mathbf{D}_t = \mathrm{diag}\big[\sigma_1(\sigma_1^2 + \lambda_*)^{-1}, \ldots, \sigma_R(\sigma_R^2 + \lambda_*)^{-1}\big], \quad \mathbf{\Omega}_t := [\mathrm{vec}(\underline{\mathbf{W}}_t^{(1)}), \ldots, \mathrm{vec}(\underline{\mathbf{W}}_t^{(L_t)})]^{'}$

Cholesky factorization $\mathbf{I}_{L_t} - \mathbf{U}_t \mathbf{D}_t \mathbf{U}_t^{'} = \mathbf{G}_t^{'} \mathbf{G}_t$

$\hat{\underline{\mathbf{S}}}_t = \arg\min_{\mathbf{s} \in \mathbb{R}^{\Pi_{m=1}^{M-1} N_m}} \ \ \frac{1}{2}\|\mathbf{G}_t \mathbf{y}_t - \mathbf{G}_t \mathbf{\Omega}_t \mathbf{s}\|^2 + \lambda_1 \|\mathbf{s}\|_1$

$\mathbf{z}_t := \mathbf{y}_t - \mathbf{\Omega}_t \mathbf{s}_t$

$\boldsymbol{\gamma}_t = \mathbf{V}_t \mathbf{\Sigma}_t^{-1} \mathbf{D}_t \mathbf{U}_t^{'} \mathbf{z}_t$

$e_t^{(\ell)} := z_t^{(\ell)} - \langle \boldsymbol{\phi}_t^{(\ell)}, \boldsymbol{\gamma}_t \rangle$

Parallel subspace update $\big[(m, r) \in [M] \times [R]\big]$

$\mathbf{a}_r^{(m)}[t] = (1 - \mu_t \lambda_*/t)\mathbf{a}_r^{(m)}[t-1] + \mu_t \hat{\gamma}_{t,r} \Big(\sum_{\ell=1}^{L_t} e_t^{(\ell)} \underline{\mathbf{W}}_t^{(\ell)} \Big) \times_{i=1, i \neq m}^{M-1} \mathbf{a}_r^{(i)}$

return $\Big(\{\mathbf{A}_m[t]\}_{m=1}^{M-1}, \hat{\mathbf{S}}_t, \hat{\boldsymbol{\gamma}}_t \Big)$

end for

$\mathbf{a}_r^{(m)}$ leads to the closed-form expression

$$\nabla_{\mathbf{a}_r^{(m)}} f_t(\mathbf{A}_1, \ldots, \mathbf{A}_{M-1}) = (\lambda_*/t)\mathbf{a}_r^{(m)} - \sum_{\ell=1}^{L} \hat{\gamma}_{t,r} \Big(z_t^{(\ell)} - \sum_{r=1}^{R} \hat{\gamma}_{t,r} \langle \mathbf{a}_r^{(1)} \circ \ldots \circ \mathbf{a}_r^{(M-1)}, \underline{\mathbf{W}}_t^{(\ell)} \rangle \Big) \underline{\mathbf{W}}_t^{(\ell)} \times_{i=1, i \neq m}^{M-1} \mathbf{a}_r^{(i)}. \quad (6.8)$$

All in all, the gradient iterations for learning the tensor subspace proceed in parallel as follows

$$\mathbf{a}_r^{(m)}[t] = \mathbf{a}_r^{(m)}[t-1] - \mu_t \nabla_{\mathbf{a}_r^{(m)}} f_t(\{\mathbf{A}_m[t-1]\}_{m=1}^{M-1}; \underline{\mathbf{S}}_t, \boldsymbol{\gamma}_t), \quad r \in [R],\ m \in [M-1] \quad (6.9)$$

where $\mu_t = \bar{\alpha_t}^{-1}$ is the shorthand notation for the step size. The resulting algorithm is listed under Table 12. Following the proof techniques in [96], Algorithm 12 can be convergent. The formal statement will be reported in the next version of the paper.

6.3.2 Implementation issues

Implementing Algorithm 12 per time instant t involves updating the projection coefficients as well as the tensor subspace. The latter is nicely parallelizable across the basis index r and the factor index m, and hence MR updates can be carried out simultaneously. The former however mainly entails SVD computation of $\mathbf{\Phi}_t \in \mathbb{R}^{L_t \times R}$, Cholesky factorization of an $L_t \times L_t$ matrix and running LASSO, that turns out to be the most cumbersome task. Fast off-the-shelf solvers such as LARS [45] can efficiently serve this purpose. One can also leverage the recent advances in online optimization [130] to linearize the LASSO cost and augment a proximal term that leads to a simple soft thresholding operation. The exact operation count of the algorithm will be specified for the applications outlined in the ensuing sections.

The ensuing section focuses on an important application of tensor subspace learning that tailor the sought model and reconstruction schemes to MRI.

6.4 Dynamic and Parallel MRI

MRI nowadays serves as a major imaging modality for noninvasive diagnosis of diseases in clinical practice [52]. However, the slow acquisition speeds introduce motions causing image artifacts, that hinder imaging of moving objects such as the heart, and the contrast-changing objects such as the flowing blood in difussion MRI for angiography. Dynamic MRI aspires to cope with these challenges by acquiring a low-spatial yet high-temporal resolution sequence of images [52]. This renders a possibly sizable portion of k-space data per snapshot inaccurate or missing, but the high temporal correlation of images can be leveraged to interpolate misses.

While (dynamic) MRI has been widely used in clinical practice, relative to other medical imaging techniques such as computerized tomography (CT) it suffers from long acquisition times to collect the data needed for creating artifact-free images. Certain types of scans may take several minutes to acquire the necessary data. Parallel imaging has recently emerged as a robust means to accelerate the MRI acquisition process. Parallel MRI uses a phased

array of coils, each one sensitive to signals returned from a limited spatial region of the imaged object. The receiver coils are arranged in a way that their sensitivity profiles cover the desired field of view. Parallel imaging techniques are adapted to properly combine the images acquired across various coils.

The aforementioned parallel imaging techniques however can accommodate only rectilinear sampling patterns. On the other hand, allowing arbitrary and possibly adaptive sampling trajectories can lead to significant acceleration. This section advocates an approach that leverages the spatiotemporal correlation of an MRI sequence captured through a low tensor rank, which enables tracking of the dynamics from the subsampled data. The collected data per time instant t form a fourth-order tensor with (k_x, k_y, c, t)th entry referring respectively to phase encoding, frequency encoding, coil index, and time. For simplicity in exposition, we will start with the plain MRI sequence acquired with a single coil, postponing the general multicoil case to a later section.

Recall that MRI reconstructs the ground-truth sequence of images, denote it by $\{\mathbf{L}_t\}$, from the (possibly undersampled) k-space data given by

$$y_t^{(\ell)} = [\mathcal{F}(\mathbf{L}_t)]_{i_\ell, j_\ell} + v_t^{(\ell)}, \quad (i_\ell, j_\ell) \in \Omega_t \tag{6.10}$$

where $\mathcal{F}(\cdot)$ denotes the two-dimensional discrete Fourier transform (DFT) operator, and the set $\Omega_t \subset [N_1] \times [N_2]$ indexes the acquired k-space data. The observations are complex-valued, meaning $y_t^{(\ell)} = \mathcal{R}\{y_t^{(\ell)}\} + j\mathcal{I}\{y_t^{(\ell)}\}$ consists of two real-valued observations. The acquisition time is clearly proportional to the sample count $\sum_{\tau=1}^{t} |\Omega_\tau|$, and it is desired to be as small as possible.

In what follows, first a tomographic approach is put forth that relies on image-domain correlations to reconstruct images from their undersampled projections onto Fourier bases. An alternative interpolation-based scheme will be introduced later that exploits correlations in the k-space to impute the missing k-space data, and subsequently reconstruct the image. To set up notation, the two-dimensional DFT operator $\mathcal{F}(\cdot)$ can be written in compact matrix form as $\mathcal{F}(\mathbf{X}_t) = \mathbf{F}_\ell \mathbf{X}_t \mathbf{F}_r$, with the left and right matrices $\mathbf{F}_\ell \in \mathbb{R}^{N_1 \times N_1}$ and $\mathbf{F}_r \in \mathbb{R}^{N_2 \times N_2}$, respectively, denoting orthonormal symmetric DFT matrices. The sketching matrix in (6.4) is then $\mathbf{W}_t^{(\ell)} = \mathbf{F}_\ell \mathbf{e}_{i_\ell} \mathbf{e}_{j_\ell}^{'} \mathbf{F}_r$, and subsequently the projection $\langle \mathbf{X}_t, \mathbf{W}_t^{(\ell)} \rangle =$

$[\mathcal{F}(\mathbf{X}_t)]_{i_\ell,j_\ell}$ corresponding to the (i_ℓ, j_ℓ)-th DFT coefficient.

6.4.1 Tomographic MRI

With reference to (6.10), our goal is to benefit from the low-CP rank of the underlying tensor sequence $\{\mathbf{L}_t\}$ to draw inference from undersampled k-space data. Medical images besides being spatially correlated, typically pertain to the same organ perhaps under different poses or states. Hence, MRI snapshots are considerably correlated. Upon decomposing the Fourier operator as $\mathcal{F} = \mathcal{F}_R + j\mathcal{F}_I$, the real and imaginary parts are given by

$$\mathcal{R}\{y_t^{(\ell)}\} = [\mathcal{F}_R(\mathbf{L}_t)]_{i_\ell,j_\ell} + \mathcal{R}\{v_t^{(\ell)}\}, \quad \mathcal{I}\{y_t^{(\ell)}\} = [\mathcal{F}_I(\mathbf{L}_t)]_{i_\ell,j_\ell} + \mathcal{I}\{v_t^{(\ell)}\}$$

where the real operator $\mathcal{F}_R$ can be written as $\mathcal{F}_R(\mathbf{X}) = \mathbf{F}_\ell^{(R)}\mathbf{X}\mathbf{F}_r^{(R)} - \mathbf{F}_\ell^{(I)}\mathbf{X}\mathbf{F}_r^{(I)}$, and subsequently the corresponding projection yields $\mathbf{W}_t^{(\ell)} = \mathbf{F}_\ell^{(R)}\mathbf{e}_{i_\ell}\mathbf{e}_{j_\ell}'\mathbf{F}_r^{(R)} - \mathbf{F}_\ell^{(I)}\mathbf{e}_{i_\ell}\mathbf{e}_{j_\ell}'\mathbf{F}_r^{(I)}$. Similarly, the operator $\mathcal{F}_I$ admits the matrix form $\mathcal{F}_I(\mathbf{X}) = \mathbf{F}_\ell^{(I)}\mathbf{X}\mathbf{F}_r^{(R)} + \mathbf{F}_\ell^{(R)}\mathbf{X}\mathbf{F}_r^{(I)}$, with the corresponding projection given by $\mathbf{W}_t^{(\ell)} = \mathbf{F}_\ell^{(I)}\mathbf{e}_{i_\ell}\mathbf{e}_{j_\ell}'\mathbf{F}_r^{(R)} + \mathbf{F}_\ell^{(R)}\mathbf{e}_{i_\ell}\mathbf{e}_{j_\ell}'\mathbf{F}_r^{(I)}$.

Per snapshot t collect the complex k-space data in $\tilde{\mathbf{y}}_t$, and group the real and imaginary parts, respectively, in $\mathcal{R}\{\tilde{\mathbf{y}}_t\}$ and $\mathcal{I}\{\tilde{\mathbf{y}}_t\}$ to form the real measurement vector $\mathbf{y}_t' := [\mathcal{R}\{\tilde{\mathbf{y}}_t\}', \mathcal{I}\{\tilde{\mathbf{y}}_t\}']$. Vector $\mathbf{y}_t$ is now the input to our subspace learning Algorithm 12. For ease of exposition, suppose next that the innovations per image are inconsequential, and correspondingly ignore the sparse components ($\mathbf{z}_t = \mathbf{y}_t$). Consider now the complex matrix $[\tilde{\mathbf{\Phi}}_t]_{\ell,r} := [\mathcal{F}(\mathbf{a}_r^{(1)}[t] \circ \mathbf{a}_r^{(2)}[t])]_{i_\ell,j_\ell}$, where (i_ℓ, j_ℓ) corresponds to the ℓ-th k-space datum, and use it to form the real matrix $\mathbf{\Phi}_t' := [\mathcal{R}\{\tilde{\mathbf{\Phi}}_t\}' \ \mathcal{I}\{\tilde{\mathbf{\Phi}}_t\}'] \in \mathbb{R}^{2|\Omega_t|\times R}$. For the estimates $(\{\mathbf{A}_i[t-1]\}_{i=1}^2, \hat{\boldsymbol{\gamma}}_t)$, the fitting residual becomes $e_t^{(\ell)} := y_t^{(\ell)} - \langle \mathbf{\Psi}_t^{(\ell)}, \hat{\boldsymbol{\gamma}}_t \rangle$. Consequently, the

gradient in (6.8) admits the simple form

$$\nabla_{\mathbf{a}_r^{(1)}} f_t(\mathbf{A}_1,\mathbf{A}_2) = (\lambda_*/t)\mathbf{a}_r^{(1)} - \gamma_{t,r}\sum_{(i_\ell,j_\ell)\in\Omega_t}\mathcal{R}\{e_t^{(\ell)}\}\Big([\mathcal{F}_R(\mathbf{a}_r^{(2)})]_{j_\ell}\mathcal{F}_R(\mathbf{e}_{i_\ell}) - [\mathcal{F}_I(\mathbf{a}_r^{(2)})]_{j_\ell}\mathcal{F}_I(\mathbf{e}_{i_\ell})\Big)$$
$$- \gamma_{t,r}\sum_{(i_\ell,j_\ell)\in\Omega_t}\mathcal{I}\{e_t^{(\ell)}\}\Big([\mathcal{F}_R(\mathbf{a}_r^{(2)})]_{j_\ell}\mathcal{F}_I(\mathbf{e}_{i_\ell}) + [\mathcal{F}_I(\mathbf{a}_r^{(2)})]_{j_\ell}\mathcal{F}_R(\mathbf{e}_{i_\ell})\Big)$$
$$\nabla_{\mathbf{a}_r^{(2)}} f_t(\mathbf{A}_1,\mathbf{A}_2) = (\lambda_*/t)\mathbf{a}_r^{(2)} - \gamma_{t,r}\sum_{(i_\ell,j_\ell)\in\Omega_t}\mathcal{R}\{e_t^{(\ell)}\}\Big([\mathcal{F}_R(\mathbf{a}_r^{(1)})]_{i_\ell}\mathcal{F}_R(\mathbf{e}_{j_\ell}) - [\mathcal{F}_I(\mathbf{a}_r^{(1)})]_{i_\ell}\mathcal{F}_I(\mathbf{e}_{j_\ell})\Big)$$
$$- \gamma_{t,r}\sum_{(i_\ell,j_\ell)\in\Omega_t}\mathcal{I}\{e_t^{(\ell)}\}\Big([\mathcal{F}_R(\mathbf{a}_r^{(1)})]_{i_\ell}\mathcal{F}_I(\mathbf{e}_{j_\ell}) + [\mathcal{F}_I(\mathbf{a}_r^{(1)})]_{i_\ell}\mathcal{F}_R(\mathbf{e}_{j_\ell})\Big) \quad (6.11)$$

for $r \in [R]$; and likewise for $\nabla_{\mathbf{a}_r^{(2)}} f_t(\mathbf{A}_1,\mathbf{A}_2)$. Note the gradient is real-valued and mainly involves DFT operations. The resultant iterates are listed under Algorithm 13 after defining $\mathbf{p}^{(m,n)}(\mathbf{x}) := [\mathcal{F}_R(\mathbf{x})]_n\mathcal{F}_R(\mathbf{e}_m) - [\mathcal{F}_I(\mathbf{x})]_n\mathcal{F}_I(\mathbf{e}_m)$, and $\mathbf{q}^{(m,n)}(\mathbf{x}) := [\mathcal{F}_R(\mathbf{x})]_n\mathcal{F}_I(\mathbf{e}_m) + [\mathcal{F}_I(\mathbf{x})]_n\mathcal{F}_R(\mathbf{e}_m)$.

Before moving onto parallel MRI, a few remarks are in order. First, the estimator (P2) takes advantage of spatiotemporal image correlations through low rank that is a global metric. One can further leverage the local features of the image by either patching or using cross-correlation of image pixels. Algorithm 13 can be easily modified to do so as will be elaborated in Chapter 7. Further, it is important to recognize that Algorithm 13 imposes no restriction on Ω_t. Thus, one can sample along arbitrary trajectories by appropriately designing an excitation sequence for the gradient coils [41]. In essence, one can adaptively choose Ω_t per snapshot to reduce the acquisition time. Further discussion of adaptive k-space sampling is deferred to the next section.

6.4.2 Tomographic parallel MRI

In order to accelerate the acquisition process, parallel MRI utilizes multiple coils each one sensitive to a specific region of the image. Consider the $N_1 \times N_2$ ground-truth image $\mathbf{L}_t$ acquired by C coils. The c-th coil's sensitivity to the image pixels is described by matrix $\mathbf{T}_t^{(t)} \in {\mathbb{R}_+}^{N_1\times N_2}$. A large value $[\mathbf{T}_t^{(c)}]_{i,j}$ indicates a "good view" of (i,j)-th pixel. Ideally, the sensitivity matrices $\{\mathbf{T}_t^{(t)}\}_{c=1}^C$ are expected non-overlapping and cover the entire image, that is $\mathbf{T}_t^{(t)} \odot \mathbf{T}_{c'}^{(t)} = \mathbf{0}$, and $\sum_{c=1}^C \mathbf{T}_t^{(t)} = \mathbf{1}\mathbf{1}'$. Let $\mathbf{L}_t^{(c)} := \mathbf{T}_t^{(t)} \odot \mathbf{L}_t$ denote the true

Algorithm 13 Online tomographic dynamic MRI

input $\{\Omega_t; y_t^{(\ell)},\ (i_\ell, j_\ell) \in \Omega_t\}_{t=1}^{\infty}, \{\mu_t\}_{t=1}^{\infty}, \lambda_*, R.$

initialize $\{\mathbf{a}_r^{(1)}[1], \mathbf{a}_r^{(2)}[1]\}_{r=1}^{R}$ at random.

for $t = 1, 2, \ldots$ **do**

$$[\tilde{\mathbf{\Phi}}_t]_{\ell,r} = [\mathcal{F}(\mathbf{a}_r^{(1)}[t] \circ \mathbf{a}_r^{(2)}[t])]_{i_\ell, j_\ell},\ \ \mathbf{\Phi}_t = \left[\begin{array}{c} \mathcal{R}\{\tilde{\mathbf{\Phi}}_t\} \\ \mathcal{I}\{\tilde{\mathbf{\Phi}}_t\} \end{array}\right] = \mathbf{U}_t \mathbf{\Sigma}_t \mathbf{V}_t'$$

$$\mathbf{B}_t = \mathrm{diag}\big[\sigma_1(\sigma_1^2 + \lambda_*)^{-1}, \ldots, \sigma_R(\sigma_R^2 + \lambda_*)^{-1}\big]$$

$$\boldsymbol{\gamma}_t = \mathbf{V}_t \mathbf{B}_t \mathbf{U}_t' \mathbf{y}_t$$

$$e_t^{(\ell)} := y_{i_\ell, j_\ell}^{(t)} - \langle \tilde{\boldsymbol{\phi}}_t^{(\ell)}, \boldsymbol{\gamma}_t \rangle$$

Parallel subspace updates $(r \in [R])$

$$\begin{aligned}
\mathbf{a}_r^{(1)}[t+1] = (1 - \lambda_* \mu_t / t)\mathbf{a}_r^{(1)}[t] \\
+\gamma_{t,r} \textstyle\sum_{(i_\ell, j_\ell) \in \Omega_t} \mathcal{R}\{e_t^{(\ell)}\} \Big([\mathcal{F}_R(\mathbf{a}_r^{(2)}[t])]_{j_\ell} \mathcal{F}_R(\mathbf{e}_{i_\ell}) - [\mathcal{F}_I(\mathbf{a}_r^{(2)}[t])]_{j_\ell} \mathcal{F}_I(\mathbf{e}_{i_\ell}) \Big) \\
+\gamma_{t,r} \textstyle\sum_{(i_\ell, j_\ell) \in \Omega_t} \mathcal{I}\{e_t^{(\ell)}\} \Big([\mathcal{F}_R(\mathbf{a}_r^{(2)}[t])]_{j_\ell} \mathcal{F}_I(\mathbf{e}_{i_\ell}) + [\mathcal{F}_I(\mathbf{a}_r^{(2)}[t])]_{j_\ell} \mathcal{F}_R(\mathbf{e}_{i_\ell}) \Big) \\
\mathbf{a}_r^{(2)}[t+1] = (1 - \lambda_* \mu_t / t)\mathbf{a}_r^{(2)}[t] \\
+\gamma_{t,r} \textstyle\sum_{(i_\ell, j_\ell) \in \Omega_t} \mathcal{R}\{e_t^{(\ell)}\} \Big([\mathcal{F}_R(\mathbf{a}_r^{(1)}[t])]_{i_\ell} \mathcal{F}_R(\mathbf{e}_{j_\ell}) - [\mathcal{F}_I(\mathbf{a}_r^{(1)}[t])]_{i_\ell} \mathcal{F}_I(\mathbf{e}_{j_\ell}) \Big) \\
+\gamma_{t,r} \textstyle\sum_{(i_\ell, j_\ell) \in \Omega_t} \mathcal{I}\{e_t^{(\ell)}\} \Big([\mathcal{F}_R(\mathbf{a}_r^{(1)}[t])]_{i_\ell} \mathcal{F}_I(\mathbf{e}_{j_\ell}) + [\mathcal{F}_I(\mathbf{a}_r^{(1)}[t])]_{i_\ell} \mathcal{F}_R(\mathbf{e}_{j_\ell}) \Big)
\end{aligned}$$

return $\{\mathbf{a}_r^{(1)}[t+1], \mathbf{a}_r^{(2)}[t+1]\}_{r=1}^{R}$

end for

image from the viewpoint of the c-th coil. Various techniques in parallel imaging have been introduced to appropriately combine the acquired images across coils to reconstruct the true one. Among others, SENSE and GRAPPA are commonly used in practice [41].

Each coil in the SENSE method, reconstructs an aliased image based on the subsampled k-space data (usually a fraction of phase encoding rows is selected). Then, the aliased images (or pixels) at various coils, each a linear combination of different pixels, are used to jointly reconstruct the ground-truth image. Clearly, SENSE leverages the spatial diversity across coils. However, it requires knowledge of the coil sensitivity maps, namely $\{\mathbf{T}_t^{(t)}\}_{c=1}^{C}$. On the other hand, the GRAPPA technique works with the raw undersampled k-space data, and interpolates the missing ones to reconstruct the image. The crux of this method is that the acquired k-space data per coil pertains to the ground-truth object *weighted* by the coil sensitivities, and thus the k-space of the object is convolved with the k-space of coil sensitivities smearing the k-space information. Hence, knowledge of missing k-space

samples is present in the acquired data.

Our fresh idea here is to build a four-way tensor by collecting the k-space measurements across coils and time, and then leverage the tensor low rank to identify $\mathbf{L}_t$ from limited measurements. Let $y_t^{(\ell_c)}$ represent an acquired k-space data by the c-th coil at time instant t, that adheres to

$$y_t^{(\ell_c)} = [\mathcal{F}(\mathbf{L}_t^{(c)})]_{i_c,j_c} + v_t^{(\ell_c)}, \quad (i_c, j_c) \in \Omega_t^{(c)},\ t \in [T] \tag{6.12}$$

where ℓ_c corresponds to the (i_c, j_c)-th k-space datum. Here, $\Omega_t^{(c)}$ indexes the available k-space data at coil c. To reduce acquisition time, the sample count $|\Omega_t^{(c)}|$ is desired to be as small as possible. The observations $y_t^{(\ell_c)}$ then form a four-way incomplete tensor $\underline{\mathbf{Y}} \in \mathbb{R}^{N_1 \times N_2 \times C \times T}$.

Collect the matrix slices $\mathbf{L}_t^{(c)}$ across various coils in the three-way tensor $\underline{\mathbf{L}}_t \in \mathbb{R}^{N_1 \times N_2 \times C}$. Assuming the underlying tensor sequence $\{\underline{\mathbf{L}}_t\}$ belongs to a low-dimensional subspace, our tensor subspace tracking schemes can be employed to learn the factor matrices $\{\hat{\mathbf{A}}_i\}_{i=1}^3$, and as a byproduct return

$$\hat{\mathbf{L}}_t^{(c)} \approx \sum_{r=1}^{R} \hat{\gamma}_{t,r} \big(\mathbf{a}_r^{(3)}[t]\big)_c \mathbf{a}_r^{(1)}[t] \circ \mathbf{a}_r^{(2)}[t] \tag{6.13}$$

'on the fly.' In the ideal case where $\hat{\mathbf{L}}_t^{(c)} = \mathbf{L}_t^{(c)}$ and the sensitivities are non-overlapping, namely $\mathbf{T}_t^{(c)} \odot \mathbf{T}_{c'}^{(t)} = \mathbf{0}$, the image is simply obtained via $\mathbf{L}_t = \sum_{c=1}^{C} \mathbf{L}_t^{(c)}$. Notwithstanding, no knowledge of coil sensitivities $\{\mathbf{T}_t^{(c)}\}_{c=1}^{C}$ is needed here. This simple approach yields a reasonable estimate of the underlying image by using $\hat{\mathbf{L}}_t = \sum_{c=1}^{C} \hat{\mathbf{L}}_t^{(c)}$.

To detail our reconstruction scheme as before for parallel MRI, let $\tilde{\mathbf{y}}_t$ collect the complex-valued observations, and form as before the measurement vector $\mathbf{y}_t' := [\mathcal{R}\{\mathbf{y}_t\}', \mathcal{I}\{\mathbf{y}_t\}'] \in \mathbb{R}^{2C\sum_{c=1}^{C}|\Omega_t^{(t)}|}$. Likewise, define the complex-valued matrix $[\tilde{\mathbf{\Phi}}_t]_{\ell,r} := [\mathbf{a}_r^{(3)}]_c[\mathcal{F}(\mathbf{a}_r^{(1)} \circ \mathbf{a}_r^{(2)})]_{i_c,j_c}$, and correspondingly $\mathbf{\Phi}_t' := [\mathcal{R}\{\tilde{\mathbf{\Phi}}_t\}' \mathcal{I}\{\tilde{\mathbf{\Phi}}_t\}']$. Recall that $\mathcal{F} = \mathcal{F}_R + j\mathcal{F}_I$.

Recalling that the gradient is readily obtained as (cf. (6.8))

$$\nabla_{\mathbf{a}_r^{(1)}} f_t(\mathbf{A}_1,\mathbf{A}_2,\mathbf{A}_3) = (\lambda_*/t)\mathbf{a}_r^{(1)} - \gamma_{t,r}\sum_{c=1}^{C}\sum_{(i_c,j_c)\in\Omega_t^{(c)}}[\mathbf{a}_r^{(3)}]_c\Big\{\mathcal{R}\{e_t^{(\ell_c)}\}\Big([\mathcal{F}_R(\mathbf{a}_r^{(2)})]_{j_c}\mathcal{F}_R(\mathbf{e}_{i_c}) - [\mathcal{F}_I(\mathbf{a}_r^{(2)})]_{j_c}\mathcal{F}_I(\mathbf{e}_{i_c})\Big) + \mathcal{I}\{e_t^{(\ell_c)}\}\Big([\mathcal{F}_R(\mathbf{a}_r^{(2)})]_{j_c}\mathcal{F}_I(\mathbf{e}_{i_c}) + [\mathcal{F}_I(\mathbf{a}_r^{(2)})]_{j_c}\mathcal{F}_R(\mathbf{e}_{i_c})\Big)\Big\}$$

and likewise w.r.t. $\mathbf{a}_r^{(2)}$. For the coil dimension basis vector $[\mathbf{a}_r^{(3)}]$ one can easily arrive at

$$\nabla_{\mathbf{a}_r^{(3)}} f_t(\mathbf{A}_1,\mathbf{A}_2,\mathbf{A}_3) = (\lambda_*/t)\mathbf{a}_r^{(3)} - \gamma_{t,r}\sum_{c=1}^{C}\sum_{(i_c,j_c)\in\Omega_t^{(c)}}\Big\{\mathcal{R}\{e_t^{(\ell_c)}\}\Big([\mathcal{F}_R(\mathbf{a}_r^{(1)})]_{i_c}[\mathcal{F}_R(\mathbf{a}_r^{(2)})]_{j_c} - [\mathcal{F}_I(\mathbf{a}_r^{(1)})]_{i_c}[\mathcal{F}_I(\mathbf{a}_r^{(2)})]_{j_c}\Big) + \mathcal{I}\{e_t^{(\ell_c)}\}\Big([\mathcal{F}_I(\mathbf{a}_r^{(1)})]_{i_c}[\mathcal{F}_R(\mathbf{a}_r^{(2)})]_{j_c} + [\mathcal{F}_R(\mathbf{a}_r^{(1)})]_{i_c}[\mathcal{F}_I(\mathbf{a}_r^{(2)})]_{j_c}\Big)\Big\}\mathbf{e}_c.$$

The resulting iterates are tabulated under Algorithm 14, where the following scalar quantities are introduced for notational brevity:

$$\pi^{(i,j)} := [\mathcal{F}_R(\mathbf{a}_r^{(1)}[t])]_i[\mathcal{F}_R(\mathbf{a}_r^{(2)}[t])]_j - [\mathcal{F}_I(\mathbf{a}_r^{(1)}[t])]_i[\mathcal{F}_I(\mathbf{a}_r^{(2)}[t])]_j$$
$$\kappa^{(i,j)} := [\mathcal{F}_R(\mathbf{a}_r^{(1)}[t])]_i[\mathcal{F}_I(\mathbf{a}_r^{(2)}[t])]_j + [\mathcal{F}_I(\mathbf{a}_r^{(1)}[t])]_i[\mathcal{F}_R(\mathbf{a}_r^{(2)}[t])]_j$$

Remark 1 [Coil sensitivity information]: Once again, Algorithm 14 does not require knowledge of coil sensitivities that can be challenging infeasible to estimate accurately. However, if one additionally knows the coil sensitivities $\{\mathbf{T}_t^{(t)}\}_{c=1}^C$, more accurate reconstruction can be accomplished by incorporating them into the tensor model. One idea is to stack the acquired images across different coils and time instants as the slices of a three-way tensor, and learn a subspace with two factor matrices as in the standard MRI but with a linear transformation tensor $\underline{\mathbf{W}}_t^{(\ell)}$ that depends on the coil sensitivities.

Remark 2 [Implementation]: Algorithm 14 admits simple updates mainly involving two-dimensional DFT computations that are amenable to efficient implementation via modern fast Fourier algorithms (FFT).

Algorithm 14 Online tomographic, dynamic, and parallel MRI

input $\{\Omega_t; y_t^{(\ell_c)},\ (i_c,j_c)\in\Omega_t^{(c)}\}_{t=1}^{\infty}, \{\mu_t\}_{t=1}^{\infty}, \lambda_*, R, C.$

initialize $\{\mathbf{a}_r^{(1)}[1], \mathbf{a}_r^{(2)}[1], \mathbf{a}_r^{(3)}[1]\}_{r=1}^{R}$ at random.

for $t = 1, 2, \ldots$ **do**

$[\tilde{\boldsymbol{\phi}}_t^{(c,r)}]_{\ell_c} = [\mathbf{a}_r^{(3)}]_c[\mathcal{F}(\mathbf{a}_r^{(1)}[t] \circ \mathbf{a}_r^{(2)}[t])]_{i_c,j_c},\ \tilde{\boldsymbol{\phi}}_t^{(r)} := [\tilde{\boldsymbol{\phi}}_t^{(1,r)}, \ldots, \tilde{\boldsymbol{\phi}}_t^{(C,r)}]$

$\boldsymbol{\Phi}_t = \begin{bmatrix} \mathcal{R}\{\tilde{\boldsymbol{\phi}}_t^{(r)}\} \\ \mathcal{I}\{\tilde{\boldsymbol{\phi}}_t^{(r)}\} \end{bmatrix} = \mathbf{U}_t\boldsymbol{\Sigma}_t\mathbf{V}_t'$

$\mathbf{B}_t = \mathrm{diag}\big[\sigma_1(\sigma_1^2+\lambda_*)^{-1}, \ldots, \sigma_R(\sigma_R^2+\lambda_*)^{-1}\big]$

$\boldsymbol{\gamma}_t = \mathbf{V}_t\mathbf{B}_t\mathbf{U}_t'\mathbf{y}_t$

$e_t^{(\ell_c)} := y_t^{(\ell_c)} - \langle\tilde{\boldsymbol{\phi}}_t^{(\ell_c)}, \boldsymbol{\gamma}_t\rangle$

Parallel subspace update ($\forall r \in [R]$)

$$\mathbf{a}_r^{(1)}[t+1] = (1-\lambda_*\mu_t/t)\mathbf{a}_r^{(1)}[t] + \gamma_{t,r}\sum_{c=1}^{C}\sum_{(i_c,j_c)\in\Omega_t^{(c)}}(\mathbf{a}_r^{(3)}[t])_c\Big\{\mathcal{R}\{e_t^{(\ell_c)}\}\mathbf{p}_{i_\ell,j_\ell}(\mathbf{a}_r^{(2)}[t]) + \mathcal{I}\{e_t^{(\ell_c)}\}\mathbf{q}_{i_\ell,j_\ell}(\mathbf{a}_r^{(2)}[t])\Big\}$$

$$\mathbf{a}_r^{(2)}[t+1] = (1-\lambda_*\mu_t/t)\mathbf{a}_r^{(2)}[t] + \gamma_{t,r}\sum_{c=1}^{C}\sum_{(i_c,j_c)\in\Omega_t^{(c)}}(\mathbf{a}_r^{(3)}[t])_c\Big\{\mathcal{R}\{e_t^{(\ell_c)}\}\mathbf{p}_{j_\ell,i_\ell}(\mathbf{a}_r^{(1)}[t]) + \mathcal{I}\{e_t^{(\ell_c)}\}\mathbf{q}_{j_\ell,i_\ell}(\mathbf{a}_r^{(1)}[t])\Big\}$$

$$\mathbf{a}_r^{(3)}[t+1] = (1-\lambda_*\mu_t/t)\mathbf{a}_r^{(3)}[t] + \gamma_{t,r}\sum_{c=1}^{C}\sum_{(i_c,j_c)\in\Omega_t^{(c)}}\Big\{\mathcal{R}\{e_t^{(\ell_c)}\}\pi_{i_c,j_c}^{(t)} + \mathcal{I}\{e_t^{(\ell_c)}\}\kappa_{i_c,j_c}^{(t)}\Big\}\mathbf{e}_c$$

return $\{\mathbf{a}_r^{(1)}[t+1], \mathbf{a}_r^{(2)}[t+1], \mathbf{a}_r^{(3)}[t+1]\}_{r=1}^{R}$

end for

6.4.3 Interpolation-based MRI

While the tomographic approach of the previous subsection seeks the spatial domain image directly from the incomplete k-space data, one can alternatively first interpolate the missing k-space data, and subsequently reconstruct the spatial domain image. Dealing with this two-step approach, the present section focuses on the standard MRI with $M = 3$, but extension to higher-order arrays, and in particular parallel MRI, is readily possible. Recall that $\{\mathbf{L}_t\}$ is the underlying image sequence of interest that lies in a low-dimensional subspace $\mathcal{L}$. Being *linear*, the Fourier operator preserves dimensionality, meaning that $\{\mathbf{X}_t = \mathcal{F}(\mathbf{L}_t)\}$ lies in a linear subspace, say $\mathcal{L}_F \subset \mathbb{C}^{N_1\times N_2}$ with $\dim(\mathcal{L}_F) \leq \dim(\mathcal{L})$. Specifically, for the complex matrix $\mathbf{X}_t = \mathbf{X}_t^{(R)} + j\mathbf{X}_t^{(I)}$ of k-space data, the real and imaginary parts each lie in low-dimensional subspaces of smaller or equal dimension than $\mathcal{L}$.

Accordingly, one can build on the low rank of tensors $\underline{\mathbf{X}}_t^{(R)}$ and $\underline{\mathbf{X}}_t^{(I)}$ to interpolate the misses from the known k-space entries. In this direction, given the partial k-space measurements

$$y_t^{(\ell)} = x_t^{(i_\ell, j_\ell)} + v_t^{(\ell)}, \quad (i_\ell, j_\ell) \in \Omega_t$$

one can postulate a trilinear model $\mathcal{R}\{x_t^{(i_\ell,j_\ell)}\} = \langle \boldsymbol{\alpha}_{i_\ell}, \boldsymbol{\beta}_{j_\ell}, \boldsymbol{\gamma}_t \rangle$ (likewise for the imaginary part $\mathcal{I}\{x_t(i_\ell, j_\ell)\}$) with $\boldsymbol{\alpha}_{i_\ell}, \boldsymbol{\beta}_{j_\ell}, \boldsymbol{\gamma}_t$ being rows of $\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3$, respectively. Choosing the sketching operator $\mathbf{W}_t^{(\ell)} = \mathbf{e}_{i_\ell}\mathbf{e}_{j_\ell}^{'}$, one can solve (P2) to form an interpolation $\hat{\mathbf{X}}_t^{(R)} := \hat{\mathbf{A}}_t \text{diag}(\hat{\boldsymbol{\gamma}}_t)\hat{\mathbf{B}}_t^{'}$ (and likewise for $\hat{\mathbf{X}}_t^{(I)}$). Forming $\hat{\mathbf{X}}_t = \hat{\mathbf{X}}_t^{(R)} + j\hat{\mathbf{X}}_t^{(I)}$, the ground-truth image can then be reconstructed based on the magnitude of $\mathcal{F}^{-1}(\hat{\mathbf{X}}_t)$, namely $[\hat{\mathbf{L}}_t]_{i,j} = \big|[\mathcal{F}^{-1}(\hat{\mathbf{X}}_t)]_{i,j}\big|$.

This approach amounts to imputation of missing tensor entries and was the focus of our work in [98] dealing with imputation of low-rank matrices and three-way tensors. The corresponding algorithm specialized to the MRI task is listed under Table 15 for the sake of completeness. It is indeed a special case of the general Algorithm 12 after dropping the sparse component, and fixing $[\boldsymbol{\Phi}_t]_{\ell,r} = [\mathbf{a}_r^{(1)}[t]]_{i_\ell}[\mathbf{b}_r^{(1)}[t]]_{j_\ell}$. The iterations admit a simple and interpretable form where the i_ℓ and j_ℓ rows of $\mathbf{A}_1[t]$ and $\mathbf{A}_2[t]$, respectively, are updated once the k-space datum $(i_\ell, j_\ell) \in \Omega_t$ arrives.

The sought interpolation-based approach proves successful when one can split each k-space $N_1 \times N_2$ image into $K_1 \times K_2$ (non)overlapping patches of size $n_1 \times n_2$, with $K_1 = N_1/n_1$ and $K_2 = N_2/n_2$. Patch sizes must be sufficient to preserve the spatiotemporal correlations, and form a tensor with low rank ρ. This idea reduces the variable count associated with the subspace from $(N_1 + N_2)R$ to $(n_1 + n_2)\rho$ which can lead to significant computational savings. In addition, the large number of frames facilitates learning the tensor subspace, especially for MRI scans with low temporal resolution. Moreover, adaptive sampling strategies, discussed in the next section, are immediately applicable to reduce the acquisition time. With reference to the matrix completion context, this approach needs the k-space data are picked uniformly, and as a result may not be as robust as the tomographic approach to various sampling patterns.

Consider the (m, n)-th patch of the t-th tensor slice that is linearly related to the under-

Algorithm 15 Online interpolation-based MRI

input $\{\Omega_t, y_t^{(\ell)},\ (i_\ell, j_\ell) \in \Omega_t\}_{t=1}^{\infty}, \{\mu_t\}_{t=1}^{\infty}, \lambda, R.$

initialize $(\mathbf{A}_1[1], \mathbf{A}_2[1])$.

for $t = 1, 2, \ldots$ **do**

Projection coefficients update

$[\mathbf{\Phi}_t]_{\ell,r} = [\mathbf{a}_r^{(1)}[t]]_{i_\ell,r}[\mathbf{a}_r^{(2)}[t]]_{j_\ell,r}\ ,\ \ \mathbf{\Phi}_t = \mathbf{U}_t\mathbf{\Sigma}_t\mathbf{V}_t^{'}$

$\mathbf{D}_t = \mathrm{diag}\big[\sigma_1(\sigma_1^2+\lambda)^{-1}, \ldots, \sigma_R(\sigma_R^2+\lambda)^{-1}\big],$

$\hat{\boldsymbol{\gamma}}_t = \mathbf{V}_t\mathbf{\Sigma}_t^{-1}\mathbf{D}_t\mathbf{U}_t^{'}\mathbf{y}_t$

$e_t^{(\ell)} := y_t^{(\ell)} - \langle \boldsymbol{\phi}_t^{(\ell)}, \boldsymbol{\gamma}_t \rangle$

Parallel subspace update $[r \in [R]]$

$\mathbf{a}_r^{(1)}[t] = (1-\mu_t\lambda/t)\mathbf{a}_r^{(1)}[t-1] + \mu_t\hat{\gamma}_{t,r}\sum_{(i_\ell,j_\ell)\in\Omega_t} e_t^{(\ell)}[\mathbf{a}_r^{(2)}]_{j_\ell}\mathbf{e}_{i_\ell}$

$\mathbf{a}_r^{(2)}[t] = (1-\mu_t\lambda/t)\mathbf{a}_r^{(2)}[t-1] + \mu_t\hat{\gamma}_{t,r}\sum_{(i_\ell,j_\ell)\in\Omega_t} e_t^{(\ell)}[\mathbf{a}_r^{(1)}]_{i_\ell}\mathbf{e}_{j_\ell}$

return $(\mathbf{A}_1[t], \mathbf{A}_2[t])$

end for

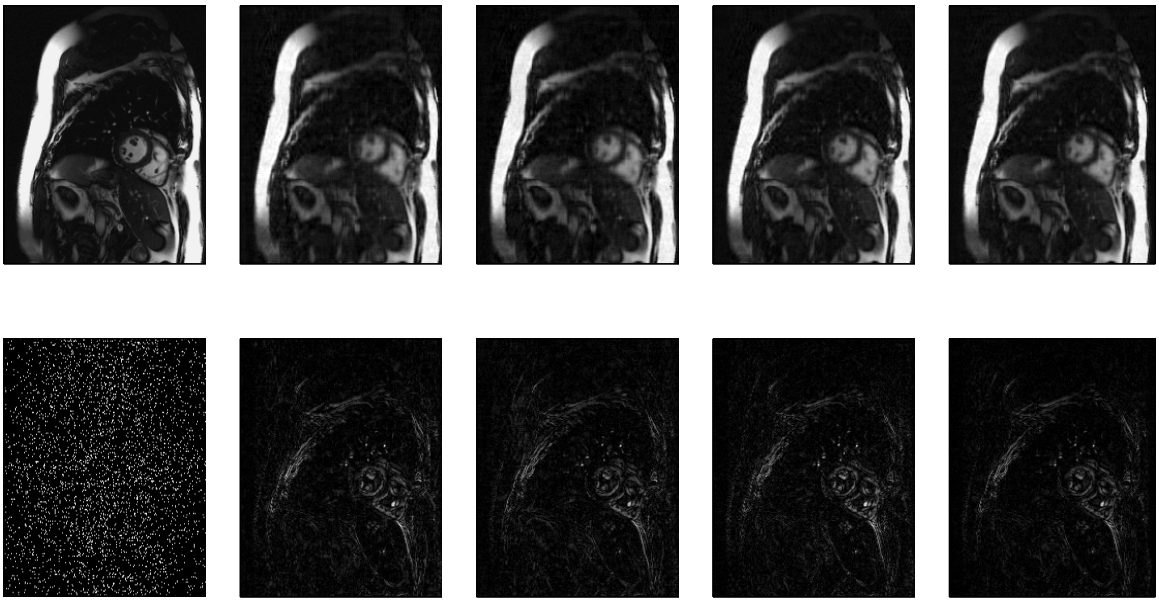
lying image $\mathbf{X}_t \approx \mathcal{F}_{m,n}(\mathbf{L}_t)$, that can be written in matrix form as $\mathcal{F}_{m,n}(\mathbf{L}_t) = \mathbf{F}_\ell^{(m)}\mathbf{L}_t\mathbf{F}_r^{(n)}$. Every two nonoverlapping patches $\mathbf{X}_t$ and $\mathbf{X}_{t'}$ indexed by (m,n) and (m',n') are orthogonal due to orthogonality of their corresponding operators $\mathcal{F}_{m,n}$ and $\mathcal{F}_{m',n'}$. Hence, either the paches should be overlapping or they should be chosen large enough so that sufficiently correlated samples can be used for imputation.

6.5 Numerical Tests

The performance of the proposed tensor model and online reconstruction schemes is assessed in this section via tests on real cardiac MRI data.

6.5.1 Cardiac MRI

To systematically evaluate the performance of the proposed method, free breathing human cardiac MR data was first simulated with quasi-periodic heartbeats. Dynamic cardiac cine images of size 200×256 across 256 frames were thus formed. Per k-space frame with 200 phase encodes and 256 frequency readouts, data were randomly undersampled with different



(a) (b) (c) (d) (e)

Figure 6.2: Results of applying tomoghraphic MRI to *in vivo* MRI dataset with uniform random sampling. (a) (top) Ground truth frame 200, (bottom) acquired k-space data undersampled randomly by a factor of 10; (b) (top) reconstructed image frame for $\hat{R} = 100$ and $\pi = 0.1$, (bottom) error magnitude; (c) (top) reconstructed image frame for $\hat{R} = 150$ and $\pi = 0.1$, and (bottom) error magnitude; (d) (top) reconstructed image frame for $\hat{R} = 100$ and $\pi = 0.25$, and (bottom) error magnitude; (e) (top) reconstructed image frame for $\hat{R} = 150$ and $\pi = 0.25$, and (bottom) error magnitude.

patterns across frames to form Ω_t. A number of undersampling trajectories were considered. Apparently, with only 256 frames available, random initialization with a single pass over data cannot yield satisfactory accuracy. Instead, several passes over the data were tested, where the outcome of the first pass with random initialization forms the initial subspace for the second pass, and so on. It was empirically observed that this procedure after a few passes over data converges, and leads to a relatively accurate reconstruction.

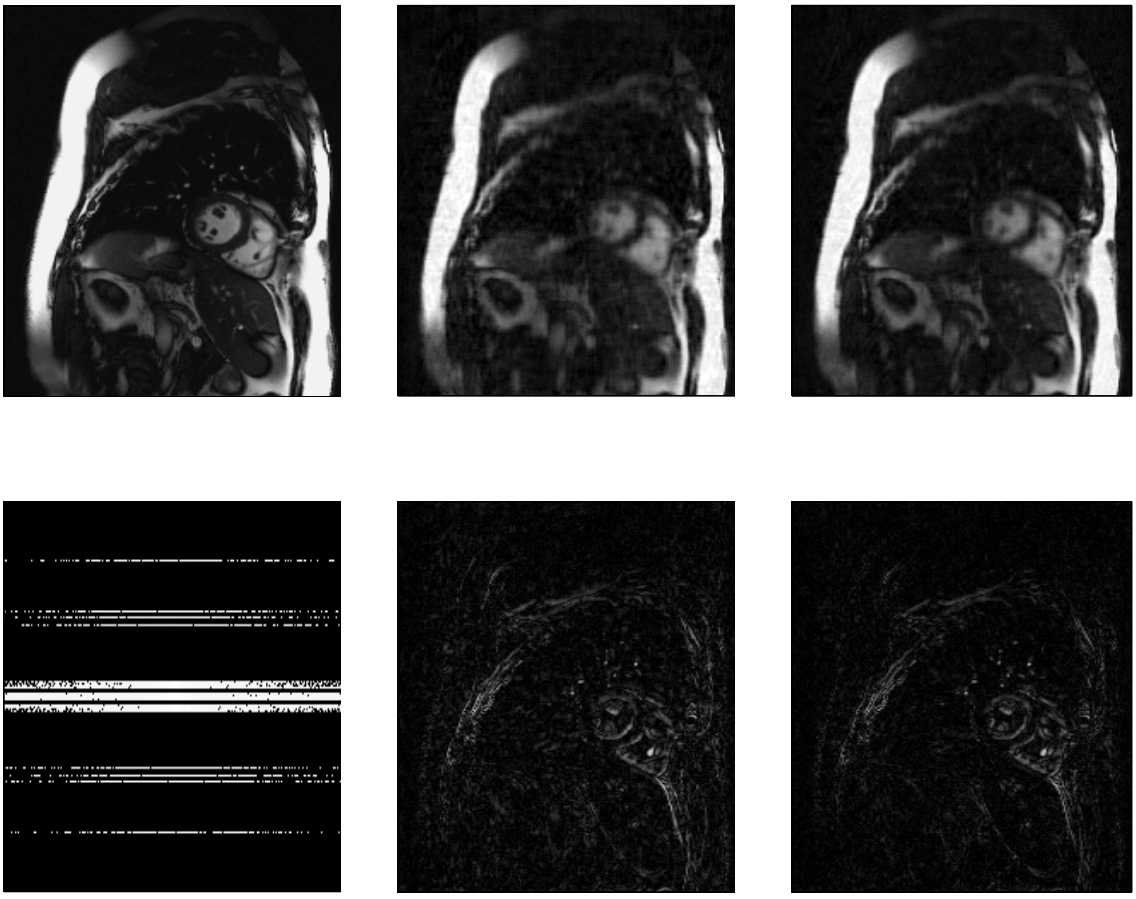
Subspace learning performance: Performance of the novel tomographic MRI scheme in Algorithm 13 is tested on k-space data sampled uniformly at random. Every element $(i, j) \in$

Ω_t is collected from the set $[200] \times [256]$ according to a Bernoulli distribution that selects an entry w.p. π, and skips it w.p. $1-\pi$. Hence, on average a fraction π of k-space data are acquired. With $\lambda = 10^{-1}$ and step-size $\mu_t = 0.1,\ \forall t$, Fig. 6.2 depicts a reconstructed frame against the trademark (full acquistion) under various adopted rank levels $\hat{R} \in \{100, 150\}$, and undersampling rates $\pi \in \{0.1, 0.25\}$. Setting $\hat{R} = 150$ and $\pi = 0.1$, with 90% of k-space data missing, the reconstructed image in Fig. 6.2(c) suffers from blurring artifacts but most of the image details are revealed. Increasing the sampling rate to $\pi = 0.25$ improves the image contrast, and identifies almost all image details. To quantify the reconstruction error, as a figure of merit, we evaluate $-20\log_{10}(\bar{e}_t)$ in decibels (dB), where $\bar{e}_t := \frac{1}{t}\sum_{\tau=1}^{t} e_\tau$ is the running-average error for the frame error $e_t := \|\mathbf{X}_t - \hat{\mathbf{X}}_t\|_F / \|\mathbf{X}_t\|_F$. With a few passes over the data, the tomographic interpolation scheme attains 27.95(dB) quality with ten fold acceleration ($\pi = 0.1$) when $\hat{R} = 150$. With four fold acceleration ($\pi = 0.25$) however it yields 30.45(dB) accuracy for $\hat{R} = 150$.

Nonuniform Cartesian sampling: Uniform undersampling used in the previous test may not be a prudent choice as it gives equal importance to all data. For real images typically most of the energy concentrates around the origin of the k-space [84]. In addition, due to physical and physiological constraints smoother sampling strategies are more appealing in practice. Grid Cartesian sampling that acquires a fraction of phase encoding lines constitutes the most popular sampling strategy in clinical practice. In order to put more importance on the lower frequencies carrying higher energy, the rows of the k-space data matrix are randomly picked following a nonuniform variable-density distribution [143]. To do so, for the 200 phase encoding rows associate the sets $\Omega_+ := \{101, \ldots, 200\}$ and $\Omega_- := \{100, \ldots, 1\}$ with positive and negative frequencies, respectively. Keep the low frequency rows $\{100, 101\}$, and draw $P/2 - 2$ rows $101 + i,\ i = 1, \ldots, 99$ according to the polynomial probability distribution $i^\alpha / \sum_{i=1}^{99} i^\alpha$ for some $\alpha \in [-1, -5]$ to form Ω_+. The set Ω_- then mirrors Ω_+. Draw sufficient trials to collect P distinct rows from $\Omega_- \cup \Omega_+$. The results are depicted in Fig. 6.3 for variable fractions of misses and rank levels. In particular, upon choosing $\hat{R} = 100$, for 10% sampling rate we obtain 32.64dB error, while increasing the sampling rate to 25% improves the error to 36.11dB.

Real-time reconstruction: The interpolation-based scheme in Algorithm 15 is tested for the setting described in the previous test. Patching is however used to facilitate real-time learning. Each 2-D k-space image is partitioned into 20 rectangular patches each of size 40×64, where altogether form a large complex-valued tensor $\underline{\mathbf{X}} = \underline{\mathbf{X}}_R + j\underline{\mathbf{X}}_I \in \mathbb{C}^{40\times 64\times 5,120}$. The singular values for the real and imaginary unfolded tensors (columns correspond to $\text{vec}(\mathbf{X}_t) \in \mathbb{R}^{2560}$) are plotted in Fig. 6.4. Clearly, the unfolded k-space data exhibits only five dominant singular values indicating high spatiotemporal correlation that can result in a low tensor rank for $\underline{\mathbf{X}}_R$ and $\underline{\mathbf{X}}_I$.

Partial k-space data are acquired in real-time according to a random uniform pattern. Algorithm 15 with random initialization is then separately run on tensors $\underline{\mathbf{X}}_R, \underline{\mathbf{X}}_I$, and image $\mathbf{L}_t$ is reconstructed as elaborated in Section 6.4.3. The retrieved frame 146 in real-time is compared against the benchmark in Fig. 6.6 for different ranks and undersampling factors. It is apparent that upon choosing $\pi = 0.25$ and $R = 100$ the reconstruction complies well with the ground-truth image at 18.2dB accuracy. Notice that this test considers only a single pass over the data, with the 146th frame reconstructed according to the subspace learned from the accumulated knowledge of past frames. To further quantify real-time performance, the evolution of the running average error $\bar{e}_t$ is plotted over iteration index t that here coincides with the acquisition time. After acquiring about 100 frames, the subspace is learned, and all the subsequent images are accurately recovered.



(a) (b) (c)

Figure 6.3: Results of applying tomoghraphic MRI to *in vivo* MRI dataset with variable-density Cartesian sampling ($\alpha = -1$). (a) (top) Ground truth frame 200, (bottom) acquired k-space data undersampled randomly by a factor of 10; (b) (top) reconstructed image frame for $\hat{R} = 100$ and $\pi = 0.1$, (bottom) error magnitude; (c) (top) reconstructed image frame for $\hat{R} = 150$ and $\pi = 0.25$, and (bottom) error magnitude.

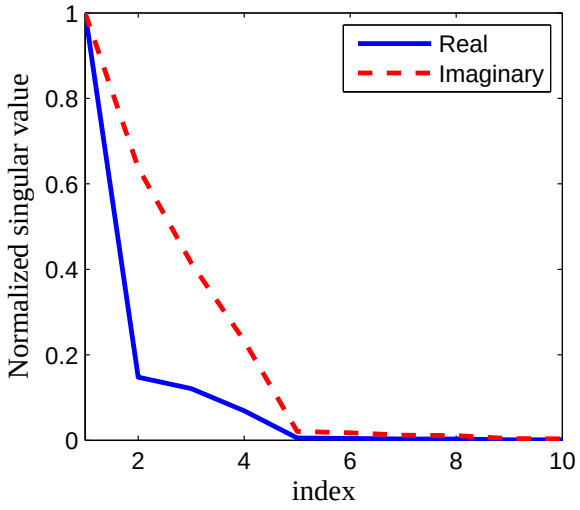


Figure 6.4: Singular values of the real and imaginary unfolded tensors.

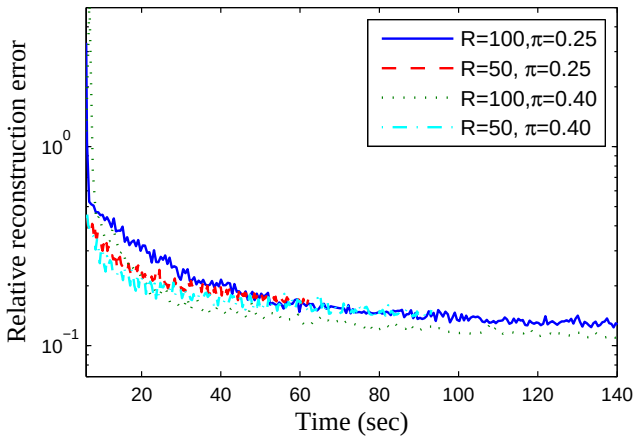


Figure 6.5: Frame reconstruction error, averaged over 10 random realizations, versus run-time for variable percentages of misses and rank levels.

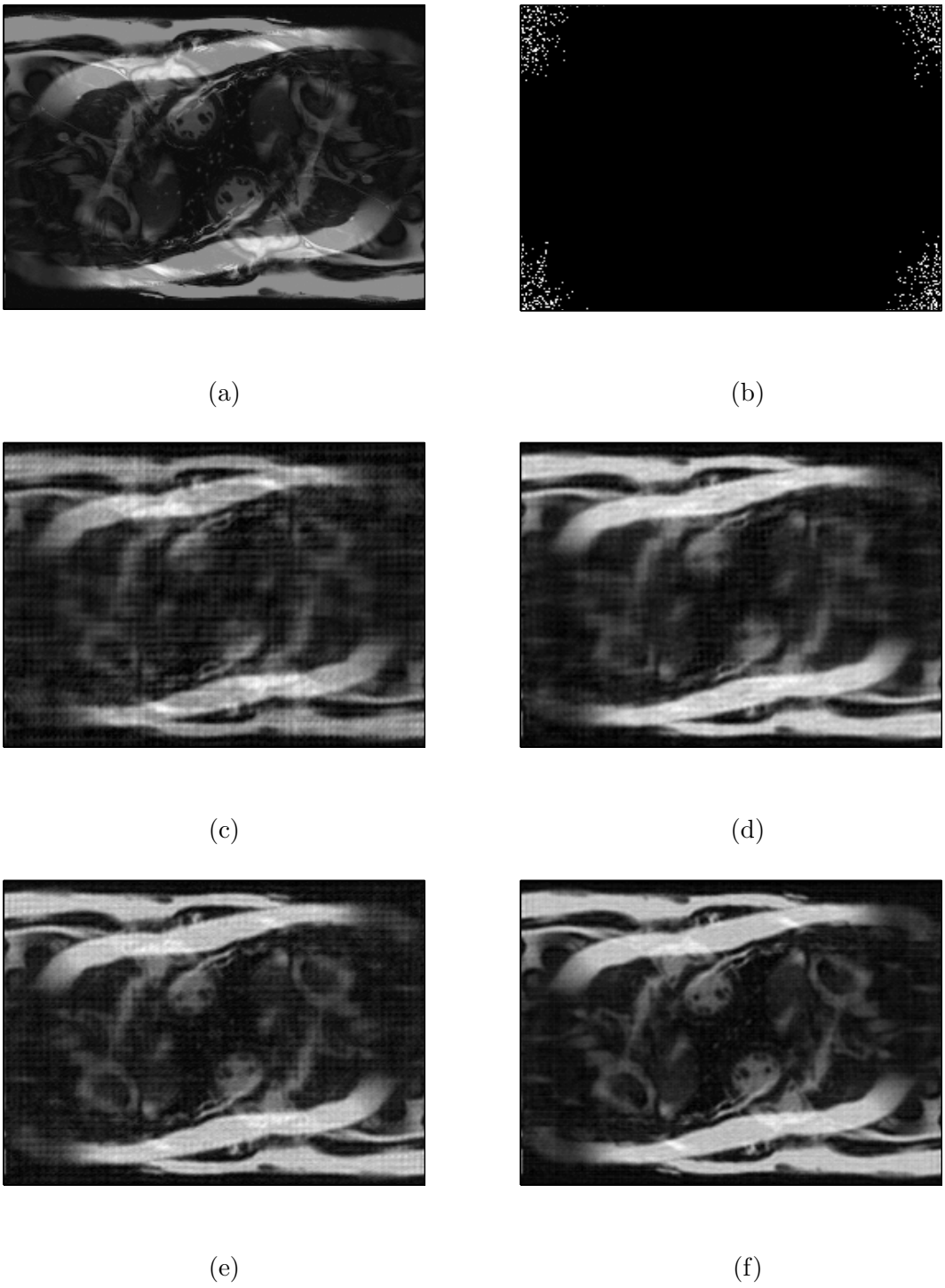


Figure 6.6: Real-time reconstruction of *in vivo* MRI dataset based on Algorithm 15. (a) Ground-truth frame 146; (b) acquired k-space data undersampled randomly by a factor of 4, reconstructed image frame for $\hat{R} = 50$ with (c) 25% and (d) 40% available data; reconstructed image frame for $\hat{R} = 100$ with (e) 25% and (f) 40% available data.

Chapter 7

Future Work

This dissertation dealt with learning from large-scale data that are typically streaming, subject to misses and anomalies, geographically spread, and has a multidimensional nature. To cope with these challenges, the intrinsic data low-dimensionality is exploited by means of sparsity and low-rank. Leveraging the ℓ_1- and nuclear-norm, this dissertation brings forth various low-complexity and effective data analytics that can be implemented in the online and decentralized fashion, and can handle multidimensional data structures. The efficacy of novel data analytics is corroborated on the important tasks of network traffic monitoring, and dynamic magnetic resonance imaging. There are still intriguing directions worth pursuing as future research. A few possible directions are pointed out in this final chapter.

7.1 Further Acceleration in Dynamic MRI

The adopted tensor model in Chapter 6 leverages the spatiotemporal correlations of the images by effecting low tensor rank approximations. Low rank however is a global property not necessarily capturing local spatial and temporal structure. In order to explicitly account for local structure, image patching and incorporation of correlation information learned from historical data are explored in this section. To ease exposition, these ideas are presented for the standard three-way MRI ($M = 3$) model, but generalizations to arbitrary order tensors

is straightforward.

7.1.1 Patching

Patching in k-space was discussed in Section 6.4.3 as a means of facilitating the interpolation process. The patched k-space data however, do not necessarily inherit the local structures present in the image domain. Alternatively, patching can be directly applied in the image domain to exploit local features. Patching is more popular for denoising purposes [115], that is generally different from the tomography task studied in this paper. For static MRI, [115] adopts image domain patching to train a sparsifying dictionary for overlapping patches. The premise is that overlapping patches create an averaging effect that can remove undersampling artifacts, thus accelerating the acquisition process.

The same idea can be adopted here to divide the underlying 2-D images into multiple possibly overlapping patches and learn a subspace for the resulting patch-based tensor. Specifically for our tomographic interpolation model (6.10), split each $N_1 \times N_2$ image $\mathbf{L}_t$ into PQ patches, each of size $n_1 \times n_2$ indexed by $(p,q) \in [P] \times [Q]$. Let $\mathcal{P}_{p,q}(\cdot) : \mathbb{R}^{N_1 \times N_2} \to \mathbb{R}^{n_1 \times n_2}$ denote the operator that extracts the (p,q)-th patch, namely $\mathbf{L}_t^{(p,q)} = \mathcal{P}_{p,q}(\mathbf{L}_t)$. Collecting patches $\{\mathbf{L}_t^{(p,q)}\}_{p,q,t}$ for T time instants into a tensor, the resulting $n_1 \times n_2 \times PQT$ tensor exhibits a low tensor rank thanks to the possibly high temporal correlation of overlapping patches. With this in mind, one can postulate a low-rank patch-based model $\mathbf{L}_t^{(p,q)} \approx \mathbf{A}_1 \mathrm{diag}(\boldsymbol{\gamma}_t^{(p,q)}) \mathbf{A}_2'$, and learn the subspace matrices $\mathbf{A}_1 \in \mathbb{R}^{n_1 \times R}$ and $\mathbf{A}_2 \in \mathbb{R}^{n_2 \times R}$.

All in all,our idea here amounts to fitting the patches to the low-rank model $\mathbf{L}_t^{(p,q)} \approx \mathbf{A}_1 \mathrm{diag}(\boldsymbol{\gamma}_t^{(p,q)}) \mathbf{A}_2'$, $(p,q) \in [P] \times [Q]$, $\forall t$, and denoising the available k-space data $y_t^{(\ell)} \approx [\mathcal{F}(\mathbf{L}_t)]_{i_\ell, j_\ell}$. This objective can be accomplished by solving the following program [cf. (P1)]

$$\text{(P3)} \quad \min_{\{\mathbf{A}_1, \mathbf{A}_2\}} \frac{1}{2t} \sum_{\tau=1}^{t} \min_{\mathbf{L}_\tau, \{\boldsymbol{\gamma}_\tau^{(p,q)}\}} \left\{ \sum_{p,q=1}^{P,Q} \left(\|\mathcal{P}_{p,q}(\mathbf{L}_\tau) - \mathbf{A}_1 \mathrm{diag}(\boldsymbol{\gamma}_\tau^{(p,q)}) \mathbf{A}_2'\|_F^2 + \frac{\lambda_*}{2} \|\boldsymbol{\gamma}_\tau^{(p,q)}\|^2 \right) \right.$$
$$\left. + \sum_{\ell=1}^{L_t} \left(y_\tau^{(\ell)} - [\mathcal{F}(\mathbf{L}_\tau)]_{i_\ell, j_\ell} \right)^2 \right\} + \frac{\lambda_*}{2t} \left(\|\mathbf{A}_1\|_F^2 + \|\mathbf{A}_2\|_F^2 \right).$$

Pursuing iterations similar to these stochastic alternating minimization ones employed by Algorithm 13 one can develop online solvers for (P3) 'on the fly.' The algorithmic details

are omitted due to space limitation. Note that the subspace now involves $R(n_1 + n_2)$ variables associated with $(\mathbf{A}_1, \mathbf{A}_2)$, thus lowering the subspace update cost relative to the non-patching used by Algorithm 13. This comes at the expense of solving for the projection coefficients $\{\boldsymbol{\gamma}_t^{(p,q)}\}_{p,q=1}^{P,Q}$ plus auxiliary variables in $\mathbf{L}_t$, that together sum up to $PQR+N_1N_2$ variables.

7.1.2 Incorporating correlation information

As an alternative to patching, the inherent image spatial patterns can be taken into account by using the cross-correlation among pixels. For medical imaging applications such knowledge can be learned from a complementary prescan or historical data. In order to leverage such a prior, a Bayesian approach along the lines of [13] can be pursued based on the multilinear nature of PARAFAC decomposition as well as the separable form of the tensor rank regularizer (6.3). Towards this end, consider the AWGN model $y_t^{(\ell)} = [\mathcal{F}(\mathbf{L}_t)]_{i_\ell,j_\ell} + v_t^{(\ell)}$, where the noise term obeys $v_t^{(\ell)} \sim \mathcal{N}(0, \sigma^2)$, and it is i.i.d. across time and space. Likewise, $\mathbf{L}_t$ is factorized as $\mathbf{L}_t = \sum_{r=1}^{R} \gamma_{t,r}\mathbf{a}_r^{(1)} \circ \mathbf{a}_r^{(2)}$, where each basis vector $\mathbf{a}_r^{(i)},\ i = 1, 2$, is Gaussian distributed with zero mean and covariance matrix $\mathbf{C}_i \in \mathbb{R}^{N_i \times N_i}$. Assume also that the bases associated with the same i are i.i.d., and mutually independent across different i's. Similarly, assume $\gamma_{t,r}$'s are i.i.d. according to $\mathcal{N}(0, 1)$. With regards to the covariances $\mathbf{C}_1$ and $\mathbf{C}_2$, they capture cross-correlation among the rows and columns of the ground-truth images, respectively; see [13] for further details.

For the considered AWGN model with priors, the maximum a posteriori (MAP) estimator is given as

$$\text{(P1)} \quad \min_{\{\mathbf{A}_1,\mathbf{A}_2\}} \ \frac{1}{2t}\sum_{\tau=1}^{t}\min_{\boldsymbol{\gamma}_\tau}\left\{\sum_{\ell=1}^{L_t}\Big(y_\tau^{(\ell)} - [\mathcal{F}(\mathbf{A}_1\text{diag}(\boldsymbol{\gamma}_\tau)\mathbf{A}_2^{'})]_{i_\ell,j_\ell}\Big)^2 + \frac{\lambda_*}{2}\|\boldsymbol{\gamma}_\tau\|^2\right\} + \frac{\lambda_*}{2t}\Big(\text{tr}\{\mathbf{A}_1^{'}\mathbf{C}_1\mathbf{A}_1\} + \text{tr}\{\mathbf{A}_2^{'}\mathbf{C}_2\mathbf{A}_2\}\Big)$$

for $\lambda_* = \sigma^2$. Again, (P1) can be solved in an online fashion along the lines of Algorithm 13 with identity covariances. The update step for the projection coefficients remain the same

as before, and the covariance matrices only modify the subspace update as follows

$$\mathbf{a}_r^{(1)}[t+1] = (\mathbf{I}_{N_1} - \frac{\lambda_* \mu_t}{t}\mathbf{C}_1)\mathbf{a}_r^{(1)}[t] + \gamma_{t,r} \sum_{(i_\ell, j_\ell)\in\Omega_t} \Big\{ \mathcal{R}\{e_t^{(\ell)}\}\mathbf{p}_{i_\ell,j_\ell}(\mathbf{a}_r^{(2)}[t]) + \mathcal{I}\{e_t^{(\ell)}\}\mathbf{q}_{i_\ell,j_\ell}(\mathbf{a}_r^{(2)}[t]) \Big\}$$
$$\mathbf{a}_r^{(2)}[t+1] = (\mathbf{I}_{N_1} - \frac{\lambda_* \mu_t}{t}\mathbf{C}_2)\mathbf{a}_r^{(2)}[t] + \gamma_{t,r} \sum_{(i_\ell, j_\ell)\in\Omega_t} \Big\{ \mathcal{R}\{e_t^{(\ell)}\}\mathbf{p}_{i_\ell,j_\ell}(\mathbf{a}_r^{(1)}[t]) + \mathcal{I}\{e_t^{(\ell)}\}\mathbf{q}_{i_\ell,j_\ell}(\mathbf{a}_r^{(1)}[t]) \Big\}. \tag{7.1}$$

Implementing (7.1) requires first estimating the correlation matrices $\mathbf{C}_1$ and $\mathbf{C}_2$ from training MRI datasets $\{\mathbf{L}_t\}_{t\in\mathcal{H}}$, obtained for instance from the past MRI examinations. The connection is made in the companion paper [13] that nicely characterizes $\mathbf{C}_1$ and $\mathbf{C}_2$ in terms of the image column- and row-wise correlations. Last but not least, (P4) trades off the correlation knowledge for a reduced number of samples L_t per time; thus it can further accelerate the MRI acquisition process.

7.2 Adaptive Sketching for Big Data Subspace Learning

While the missing data paradigm pertains to lack of information-bearing measurements, one can purposely skip data to either facilitate the acquisition process, or, to lower the computational burden of data processing algorithms. The former is well motivated by recent efforts towards accelerating long MRI scans creating artifacts especially for imaging moving objects. Suppose for instance the MR scanner is a priori aware of the best minimal subset of k-space data to acquire per cardiac snapshot. This would allow sampling the important data on time before the heart moves to another state.

Finding optimal sampling trajectory however poses formidable challenges especially during online acquisition. Typical subsampling strategies for instance in the context of randomized linear algebra assume the full data available offline to score their "importance" and accordingly select a subset of "most informative" features; see e.g., [88]. However, in real-time applications, take dynamic MRI as an instance, the acquisition is indeed the main challenge. With reference to the observation model (6.10), and considering the streaming nature of MR scanning, the goal is to adaptively design at time instant t the sketching operator $\{\underline{\mathbf{W}}_\ell^t\}_{\ell=1}^{L_t}$ giving rise to a minimal sample count L_t, while attaining a prescribed

reconstruction quality.

Although the streaming nature of data adds challenges to the sketching operation, online learning offers intermediate estimates of the latent low-dimensional subspace, namely $\hat{\mathcal{L}}_t$, that can be leveraged to devise *adaptive* sampling strategies. Focus on the selection operator $\underline{\mathbf{W}}^{(t)} = \mathbf{e}_{n_1} \circ \ldots \circ \mathbf{e}_{n_{M-1}}$, that picks the feature $(n_1, \ldots, n_{M-1}) \in \Omega_t$, with $\Omega_t \in [N_1] \times \ldots \times [N_{M-1}]$ and $|\Omega_t| = L_t$. In order to select Ω_t, the basic idea here is to score the features according to their level of importance measured by a certain statistical leverage score along the lines of [35, 88]. Note that [88], and [35] deal with *batch* processing of *vector* observations ($M = 1$). To maintain simplicity in exposing our tensor generalizations, consider three-way arrays ($M = 3$), where at time instant t one is given the subspace estimate $(\mathbf{A}[t-1], \mathbf{B}[t-1])$ and wishes to acquire a small subset of the k-space data $\mathbf{Y}_t$. Supposing slow temporal variations of the latent tensor subspace, namely $\mathbf{A}[t-1] \approx \mathbf{B}[t]$ and $\mathbf{B}[t-1] \approx \mathbf{B}[t]$, the (m,n)-th feature can be well approximated with the m-th row of $\mathbf{A}[t-1]$ and the n-th row of $\mathbf{B}[t-1]$, respectively, as $[\mathbf{Y}_t]_{m,n} \approx \sum_{r=1}^{R} \gamma_r^{(t)} [\mathbf{A}[t-1]]_{m,r} [\mathbf{B}[t-1]]_{n,r}$.

In essence, the columns of $\mathbf{A}$ and $\mathbf{B}$ span the column and row space of the tensor slice $\mathbf{Y}_t$. Matrix $\mathbf{A}$ and $\mathbf{B}$ are not necessarily orthonormal, but play the same role as the orthonormal matrices $\mathbf{U}$ and $\mathbf{V}$ forming the singular value decomposition $\mathbf{Y}_t = \mathbf{U}\mathbf{\Sigma}\mathbf{V}'$. In accordance with the matrix completion context [26, 35], for row and column spaces, introduce the incoherence measures $\mu_i := \|\mathbf{U}'\mathbf{e}_i\|^2$ and $\nu_j := \|\mathbf{V}'\mathbf{e}_j\|^2$, respectively, where μ_i is nothing but the projection of canonical basis $\mathbf{e}_i$ onto the column-space; and likewise for ν_j and the row space. It is well understood that the (i,j)-th entry of a low-rank matrix with high degree of coherence (μ_i, ν_j) is susceptible to "misidentification" if skipped because it is more aligned with the column and row space of the true matrix, and as a result it can be erroneously estimated without increasing the rank [26, 35].

Along this line of thought, $(\mathbf{A}, \mathbf{B})$ can play the same role as the orthonormal matrices $(\mathbf{U}, \mathbf{V})$. To this end, normalize the columns of $\mathbf{A}$ and $\mathbf{B}$ to end up with unity column-norm matrices $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{B}}$, respectively. At time instant t with the estimates $(\tilde{\mathbf{A}}[t-1], \tilde{\mathbf{B}}[t-1])$ available, one can associate the score $\rho_t(m,n) := \frac{1}{2R}(\|\mathbf{e}_m' \tilde{\mathbf{A}}[t-1]\|^2 + \|\mathbf{e}_n' \tilde{\mathbf{B}}[t-1]\|^2)$ to the (m,n)th sampling point. The scores $\{\rho_t(m,n)\}_{(m,n)\in[M]\times[N]}$ are nonnegative-valued

and sum up to one; thus, they can be interpreted as a probability distribution. One can then draw K random trials without replacement to collect important features from Ω_t. The number of sampled features $L_t = |\Omega_t|$ is a random variable $L_t \in \{1, \ldots, K\}$ when $K \leq MN$.

Following the same logic, one can utilize the factor matrices offered by the PARAFAC decomposition to form a probability distribution and perform random sampling for a general K-order tensor. Upon defining the column-normalized factor matrices $\{\tilde{\mathbf{A}}_k[t-1]\}_{k=1}^{K-1}$, the leverage score for $(n_1, \ldots, n_{K-1})$-th entry is defined as

$$\rho_t(n_1, \ldots, n_{K-1}) := \frac{1}{(K-1)R} \sum_{i=1}^{K-1} \|\mathbf{e}_{n_i}^{'} \tilde{\mathbf{A}}_i[t-1]\|^2. \tag{7.2}$$

The resulting random sampling policy can be used in conjunction with the iterates of Algorithm 12 to arrive at randomized tensor subspace learning schemes. Of course, the effectiveness of such iterations strongly depends on the choice of initialization.

7.3 Dynamic Tensor Spectral Clustering

In large-scale networks, nodes typically form clusters (communities) with higher interactions among the cluster members than between members and the rest of the network. These communities can be envisioned as organizational network units e.g., the scientific disciplines in citation and collaboration networks [10]. Identifying and tracking communities is an important yet challenging task across science and engineering. The challenges emanate from: c1) large number of nodes, c2) dynamic evolution of networks, and c3) lack of knowledge about (possible) interactions of each node. Imagine for instance the Facebook with more than one billion users developing new friendships every minute, where users may not reveal their friends. To be more precise, consider a network of N nodes, with similarity (e.g., adjacency) only between each node and a few of its neighbors, namely $s_{n,m}^{(t)}$, $(n,m) \in \Omega_t$. Given dynamic data $\{\{s_{m,n}^{(\tau)}\}_{(m,n)\in\Omega_\tau}\}_{\tau=1}^{t}$ with misses and noise, the goal is to track communities over time. The state-of-art community trackers rely on non-negative matrix factorization [89, 155, 161], that face scalability issues (c1), and need the entire possible pair-wise similarities (c3). In addition, heterogeneous environments with multiple network

views (e.g., interactions from both Facebook and Twitter) press the need for high-order arrays [83].

In the presence of c1-c3, our idea builds upon the first-order tensor subspace tracking schemes elaborated in Chapter 6. Upon defining $[\boldsymbol{\Omega}_t \odot \mathbf{Y}_t]_{m,n} := s^{(t)}_{m,n}$, the matrix $\boldsymbol{\Omega}_t \odot \mathbf{Y}_t \in \mathbb{R}^{N\times N}$ forms a tensor slice, which can be used to learn the factor matrices $(\hat{\mathbf{A}}_t, \hat{\mathbf{B}}_t)$, and subsequently impute the misses 'on the fly.' Fixing the tensor rank to the number of clusters K, using $(\hat{\mathbf{A}}_t, \hat{\mathbf{B}}_t)$, the similarity vector associated with node n can be estimated as $\hat{\mathbf{y}}_n = \sum_{k=1}^{K} \gamma_k^{(t)} \hat{b}_{n,k} \|\hat{\mathbf{a}}_k^{(t)}\| \Big(\hat{\mathbf{a}}_k^{(t)} / \|\hat{\mathbf{a}}_k^{(t)}\| \Big)$, where $\hat{\mathbf{a}}_k^{(t)}$ is the kth column of $\hat{\mathbf{A}}_t$, and $b_{n,k}^{(t)}$ is the (n,k)th entry of $\hat{\mathbf{B}}_t$. The coefficient $\gamma_k^{(t)} \hat{b}_{n,k} \|\hat{\mathbf{a}}_k^{(t)}\|$ can be interpreted as the membership strength of node n to kth cluster, characterized by the vector $\hat{\mathbf{a}}_k^{(t)} / \|\hat{\mathbf{a}}_k^{(t)}\|$. Toward this end, there are several important avenues to explore. One pertains to extension from three-way to higher-way tensors that can capture additional dimensions e.g, multiple network views. One can also leverage inherent network structures to enhance the tracking performance by incorporating state space models for the subspace factors. Moreover, unveiling anomalous events based on evolution of communities is another intriguing path to pursue.

Bibliography

[1] http://www.techamericafoundation.org

[2] http://internet2.edu/observatory/archive/data-collections.html

[3] A. Abdelkefi1, Y. Jiang, W. Wang, A. Aslebo, and O. Kvittem, "Robust traffic anomaly detection with principal component pursuit," in *Proc. of the ACM CoNEXT Student Workshop*, Philadelphia, USA, Nov. 2010.

[4] J. Abernethy, F. Bach, T. Evgeniou, and J.-P. Vert, "A new approach to collaborative filtering: Operator estimation with spectral regularization," *J. of Machine Learning Research*, vol. 10, pp. 803–826, 2009.

[5] E. Acar, D. M. Dunlavy, T. G. Kolda, and M. Mrup, "Scalable tensor factorizations for incomplete data," *Chemometrics and Intelligent Laboratory Systems*, vol. 106, pp. 41–56, Mar. 2011.

[6] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowledge and Data Engineering*, vol. 17, pp. 734–749, June 2005.

[7] A. Agarwal, S. Negahban, and M. J. Wainright, "Noisy matrix decomposition via convex relaxation: optimal rtaes in high dimensions," *Ann. Statist.*, vol. 40, pp. 1171–1197, Sept. 2012.

[8] T. Ahmed, M. Coates, and A. Lakhina, "Distributed principal component analysis on networks via directed graphical models," in *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Kyoto, Japan, Mar. 2012.

[9] M. M. B. Baingana and G. B. Giannakis, "Scalable kernel PCA with rank regularization for streaming data," in *Proc. ACM Conf. on Knowledge Discovery and Data Minning (KDD)*, Aug. 2015.

[10] L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan, "Group formation in large social networks: membership, growth, and evolution," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 44–54, ACM, 2006.

[11] P. Barford and D. Plonka, "Characteristics of network traffic flow anomalies," in *Proc. 1st ACM SIGCOMM Workshop on Internet Measurements*, San Francisco, CA, november 2001.

[12] R. Basri and D. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, pp. 218–233, 2003.

[13] J. A. Bazerque, G. Mateos, and G. B. Giannakis, "Rank regularization and Bayesian inference for tensor completion and extrapolation," *IEEE Trans. Signal Process.*, vol. 61, pp. 5689–5703, Nov. 2013.

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

[15] J. M. F. T. Berge and N. D. Sidiropoulos, "On uniqueness in CANDECOMP/PARAFAC," *Psychometrika*, vol. 67, pp. 399–409, Sept. 2002.

[16] D. P. Bertsekas, "Distributed dynamic programming," *IEEE Trans. Automatic Contr.*, vol. 27, no. 3, pp. 610–616, 1982.

[17] D. P. Bertsekas, *Nonlinear Programming*. Athena-Scientific, second ed., 1999.

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

[19] P. J. Bickel, Y. Ritov, and A. Tsybakov, "Simultaneous analysis of Lasso and Dantzig selector," *Ann. Statist.*, vol. 37, pp. 1705–1732, Apr. 2009.

[20] B. Bollobas, *Random Graphs*. Cambridge University Press, 2001.

[21] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learning*, vol. 3, pp. 1–122, 2011.

[22] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[23] S. Burer and R. D. Monteiro, "Local minima and convergence in low-rank semidefinite programming," *Mathematical Programming*, vol. 103, no. 3, pp. 427–444, 2005.

[24] J. F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. Optim.*, vol. 20, no. 4, pp. 1956–1982, 2008.

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

[26] E. J. Candès and Y. Plan, "Matrix completion with noise," *Proceedings of the IEEE*, vol. 98, pp. 925–936, 2009.

[27] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Found. Comput. Math.*, vol. 9, no. 6, pp. 717–772, 2009.

[28] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Info. Theory*, vol. 52, pp. 489–509, Feb. 2006.

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

[30] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, pp. 14–20, Mar. 2008.

[31] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational Mathematics*, vol. 9, no. 6, pp. 717–772, 2009.

[32] E. Cascetta, "Estimation of trip matrices from traffic counts and survey data: A generalized least-squares estimator," *Transportation Research, Part B: Methodological*, vol. 18, pp. 289–299, 1984.

[33] 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.

[34] Y. Chen, A. Jalali, S. Sanghavi, and C. Caramanis, "Low-rank matrix recovery from errors and erasures," *IEEE Trans. Info. Theory*, vol. 58, 2012. see also arXiv:1104.0354v2 [cs.IT].

[35] Y. Chen, S. Bhojanapalli, S. Sanghavi, and R. Ward, "Coherent matrix completion," in *Proc. of The 31st International Conference on Machine Learning*, Beijing, China, pp. 674–682, 2014.

[36] Q. Chenlu and N. Vaswani, "Recursive sparse recovery in large but correlated noise," in *Proc. of 49th Allerton Conf. on Communication, Control, and Computing*, pp. 752 –759, Sept. 2011.

[37] A. Chistov and D. Grigorev, "Complexity of quantifier elimination in the theory of algebraically closed fields," in *Math. Found. of Computer Science*, vol. 176 of *Lecture Notes in Computer Science*, pp. 17–31, Springer Berlin / Heidelberg, 1984.

[38] S. Chouvardas, K. Slavakis, Y. Kopsinis, and S. Theodoridis, "A sparsity-promoting adaptive algorithm for distributed learning," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5412–5425, 2012.

[39] P. L. Combettes and J.-C. Pesquet, "A proximal decomposition method for solving convex variational inverse problems," *Inverse Problems*, vol. 24, no. 6, pp. 1–27, 2008.

[40] P. L. Combettes and J.-C. Pesquet, "Proximal Splitting Methods in Signal Processing," in *Fixed-Point Algorithms for Inverse Problems in Science and Engineering* (H. H. Bauschke, R. S. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz, eds.), Springer Optimization and its Applications, pp. 185–212, Springer New York, 2011.

[41] A. Deshmane, V. Gulani, M. A. Griswold, and N. Seiberlich, "Parallel MR Imaging," *Journal of Magnetic Resonance Imaging*, vol. 36, pp. 55–72, July 2012.

[42] F. Deutsch, *Best Approximation in Inner Product Spaces*. Springer-Verlag, second ed., 2001.

[43] X. Ding, L. He, and L. Carin, "Bayesian Robust Principal Component Analysis," *IEEE Trans. Image Process.*, vol. 20, pp. 3419–3430, Dec. 2011.

[44] D. L. Donoho and M. Elad, "Optimally sparse representation in general (non-orthogonal) dictionaries via ℓ_1 minimization," *Proc. Natl. Acad. Sci.*, vol. 100, pp. 2197–2202, Mar. 2003.

[45] I. Drori and D. L. Donoho, "Solution of ℓ_1 Minimization Problems by LARS/Homotopy Methods," in *Proc. of Intl. Conf. on Acoustics, Speech and Signal Process.*, Toulouse, France, may 2006.

[46] T. Eltoft, T. Kim, and T.-W. Lee, "On the multivariate Laplace distribution," *IEEE Signal Process. Letters*, vol. 13, pp. 300–303, May 2006.

[47] F. Sheikholeslami, M. Mardani, and G. B. Giannakis, "Streaming support vector classification of big data with misses," in *Proc. of Asilomar Conf. on Control, Signal and Systems*, pp. 516–520, Nov. 2014.

[48] M. Fazel, H. Hindi, and S. P. Boyd, "A rank minimization heuristic with application to minimum order system approximation," in *Proc. of American Control Conference*, vol. 6, pp. 4734–4739, 2001.

[49] J. Finn, K. Nael, V. Deshpande, O. Ratib, and G. Laub, "Cardiac MR Imaging: State of the Technology," *Radiology*, vol. 241, no. 2, pp. 338–354, 2006.

[50] S. Gandy, B. Recht, and I. Yamada, "Tensor completion and low-n-rank tensor recovery via convex optimization," *Inverse Problems*, vol. 27, pp. 1–19, Feb. 2011.

[51] A. Ganesh, K. Min, J. Wright, and Y. Ma, "Principal component pursuit with reduced linear measurements," *arXiv:1202.6445v1 [cs.IT]*, 2012.

[52] H. Gao, "Prior rank, intensity and sparsity model (PRISM): a divide-and-conquer matrix decomposition model with low-rank coherence and sparse variation," in *Proc. of SPIE Optical Engineering Applications*, 2012.

[53] H. Gao, J. Cai, Z. Shen, and H. Zhao, "Robust principal component analysis-based four-dimensional computed tomography," *Physics in Medicine and Biology*, vol. 56, no. 11, pp. 3181–3198, 2011.

[54] G. B. Giannakis, *Cyclostationary Signal Analysis.* Chapter in Digital Signal Processing Handbook: V. K. Madisetti and D. Williams, Eds. Boca Raton, FL: CRC, 1998.

[55] J. P. Haldar and Z.-P. Liang, "Spatiotemporal imaging with partially separable functions: A matrix recovery approach," in *Proc. Intl. Symp. on Biomedical Imaging: From Nano to Macro*, Rotterdam, pp. 716–719, IEEE, 2010.

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

[57] B. He and X. Yuan, "On the O(1/t) convergence rate of alternating direction method," Technical Report, Nanjing University, 2011.

[58] R. A. Horn and C. R. Johnson, *Matrix Analysis.* Cambridge University Press, 1985.

[59] P. Huang, S. D. Chen, P. Smaragdis, and M. Hasegawa-Johnson, "Singing-voice separation from monaural recordings using robust principal component analysis," in *Proc. of Intl. Conf. on Acoustics, Speech and Signal Processing*, Kyoto, Japan, pp. 57–60, Mar. 2012.

[60] J. P. J. Mairal, J. Bach and G. Sapiro, "Online learning for matrix factorization and sparse coding," *J. of Machine Learning Research*, vol. 11, pp. 19–60, jan 2010.

[61] M. G. Jafari and M. D. Plumbley, "Fast dictionary learning for sparse representations of speech signals," *IEEE J. Sel. Topics Signal Process.*, vol. 5, pp. 1025–1031, Sept. 2011.

[62] D. Jakovetic, J. Xavier, and J. M. F. Moura, "Fast distributed gradient methods," arXiv:1112.2972v1 [cs.IT].

[63] D. Jakovetic, J. Xavier, and J. Moura, "Cooperative Convex Optimization in Networked Systems: Augmented Lagrangian Algorithms With Directed Gossip Communication," *IEEE Trans. Signal Process.*, vol. 59, pp. 3889 –3902, Aug. 2011.

[64] I. T. Jolliffe, *Principal Component Analysis.* New York: Springer, 2002.

[65] L. B. Jun He and A. Szlam, "Incremental Gradient on the Grassmannian for Online Foreground and Background Separation in Subsampled Video," in *Proc. of IEEE*

Conference on Computer Vision and Pattern Recognition, Providence, Rhode Island, June 2012.

[66] H. Kim, S. Lee, X. Ma, and C. Wang, "Higher-Order PCA for Anomaly Detection in Large-Scale Networks," in *Proc. of 3rd Workshop on Comp. Advances in Multi-Sensor Adaptive Proc.*, Aruba, Dutch Antilles, Dec. 2009.

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

[68] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[69] J. Kruskal, "Three-way arrays: Rank and uniqueness of trilinear decompositions with application to arithmetic complexity and statistics," *Lin. Alg. Applications*, vol. 18, no. 2, pp. 95–138, 1977.

[70] R. N. L. Balzano and B. Recht, "Online identification and tracking of subspaces from highly incomplete information," in *Proc. of Allerton Conference on Communication, Control, and Computing*, Monticello, USA, June 2010.

[71] T. S. L. Ljung, *Theory and Practice of Recursive Identification*. MIT Press, second edition, 1983.

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

[73] A. Lakhina, M. Crovella, and C. Diot, "Mining anomalies using traffic feature distributions," vol. 35, pp. 217–228, august 2005.

[74] A. Lakhinaa, K. Papagiannaki, M. Crovella, C. Diot, E. D. Kolaczyk, and N. Taft, "Structural analysis of network traffic flows," in *Proc. of ACM SIGMETRICS*, New York, NY, July.

[75] Lauterbur, "Image formation by induced local interactions: Examples employing nuclear magnetic resonance," *Nature*, vol. 242, pp. 190–191, march 1973.

[76] Y. Li and D. Wang, "Separation of singing voice from music accompaniment for monaural recordings," *IEEE Trans. Audio, Speech, and Language Process.*, vol. 15, no. 4, pp. 1475–1487, 2007.

[77] Z.-P. Liang, "Spatiotemporal imaging with partially separable functions," in *Proc. Intl. Symp. on Biomedical Imaging: From Nano to Macro, 2007. ISBI 2007*, Arlington, VA, pp. 988–991, IEEE, april 2007.

[78] Z.-P. . Liang and P. C. Lauterbur, *Principles of Magnetic Resonance Imaging: A Signal Processing Perspective*. The Institute of Electrical and Electronics Engineers Press, 2000.

[79] Y. Liao, W. Du, P. Geurts, and G. Leduc, "DMFSGD: A decentralized matrix factorization algorithm for network distance prediction," *IEEE/ACM Trans. Network.*, 2011. see also arXiv:1201.1174v1 [cs.NI].

[80] 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.

[81] S. G. Lingala, Y. Hu, E. DiBella, and M. Jacob, "Accelerated dynamic MRI exploiting sparsity and low-rank structure: k-t SLR," *IEEE Trans. on Medical Imaging*, vol. 30, pp. 1042–1054, May 2011.

[82] J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor Completion for Estimating Missing Values in Visual Data," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 35, pp. 208–220, Jan. 2013.

[83] X. Liu, S. Ji, W. Glanzel, and B. De Moor, "Multiview partitioning via tensor methods," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 25, no. 5, pp. 1056–1069, 2013.

[84] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magnetic Resonance in Medicine*, vol. 58, pp. 1182–1195, dec 2007.

[85] G. B. G. M. Mardani and L. Ying, "Accelerating dynamic MRI via tensor subspace tracking," in *Proc. of ISMRM 23rd Annual Meeting and Exhibition*, June 2015.

[86] G. S. K. S. M. Mardani, L. Ying and G. B. Giannakis, "Dynamic MRI using subspace tensor tracking," in *Proc. of 36th Engineering in Medicine and Biology Conference (EMBC)*, Aug. 2014.

[87] B. Madore, G. Glover, N. Pelc, *et al.*, "Unaliasing by Fourier-encoding the overlaps using the temporal dimension (UNFOLD), applied to cardiac imaging and fMRI," *Magnetic Resonance in Medicine*, vol. 42, no. 5, pp. 813–828, 1999.

[88] M. W. Mahoney, "Randomized algorithms for matrices and data," *Foundations and Trends® in Machine Learning*, vol. 3, no. 2, pp. 123–224, 2011.

[89] S. Mankad and G. Michailidis, "Structural and functional discovery in dynamic networks with non-negative matrix factorization," *Physical Review E*, vol. 88, no. 4, p. 042812, 2013.

[90] P. Mansfield, "Multi-planar image formation using NMR spin echoes," *Journal of Physics C: Solid State Physics*, vol. 10, no. 3, p. L55, 1977.

[91] M. Mardani and G. B. Giannakis, "Adaptive sketching for big data subspace learning," in *Proc. of European Signal and Information Processing Conference (EUSIPCO)*, Aug. 2015.

[92] M. Mardani and G. B. Giannakis, "Low rank plus sparse tensor learning: Quest for accelerated dynamic MRI," in *Proc. of International Symposium on Biomedical Imaging (ISBI)*, Apr. 2015.

[93] M. Mardani and G. B. Giannakis, "Estimating traffic and anomaly maps via network tomography," *IEEE/ACM Transactions on Networking*, August 2015 (to appear).

[94] M. Mardani, G. Mateos, and G. B. Giannakis, "Unveiling anomalies in large-scale networks via sparsity and low rank," in *Proc. of 45th Asilomar Conf. on Signal, Systems and Computers*, Pacific Grove, CA, pp. 403–407, Nov. 2011.

[95] M. Mardani, G. Mateos, and G. B. Giannakis, "Decentralized sparsity regularized rank minimization: Applications and algorithms," *IEEE Trans. Signal Process.*, vol. 61, pp. 5374–5388, Nov. 2013.

[96] M. Mardani, G. Mateos, and G. B. Giannakis, "Dynamic anomalography: Tracking network anomalies via sparsity and low rank," *IEEE J. Sel. Topics Signal Process.*, vol. 7, pp. 50–66, Feb. 2013.

[97] M. Mardani, G. Mateos, and G. B. Giannakis, "Recovery of low-rank plus compressed sparse matrices with application to unveiling traffic anomalie," *IEEE Trans. Info. Theory.*, vol. 59, pp. 5186–5205, Aug 2013.

[98] M. Mardani, G. Mateos, and G. B. Giannakis, "Subspace learning and imputation for streaming big data matrices and tensors," *IEEE Transactions on Signal Processing*, July 2015 (to appear).

[99] M. Mardani and G. B. Giannakis, "Dept. of ECE, University of Minnesota, Minneapolis, 55455, USA," in *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*, pp. 811–814, IEEE, 2013.

[100] M. Mardani and G. B. Giannakis, "Robust network traffic estimation via sparsity and low rank," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pp. 4529–4533, IEEE, 2013.

[101] M. Mardani, G. Mateos, and G. B. Giannakis, "Distributed nuclear norm minimization for matrix completion," in *13th IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 354–358, 2012.

[102] M. Mardani, G. Mateos, and G. B. Giannakis, "Exact recovery of low-rank plus compressed sparse matrices," in *IEEE Workshop on Statistical Signal Processing (SSP)*, pp. 49–52, 2012.

[103] M. Mardani, G. Mateos, and G. B. Giannakis, "Rank minimization for subspace tracking from incomplete data," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5681–5685, IEEE, 2013.

[104] M. Mardani, G. Mateos, and G. B. Giannakis, "Imputation of streaming low-rank tensor data," in *8th IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pp. 433–436, IEEE, 2014.

[105] G. Mateos, J. A. Bazerque, and G. B. Giannakis, "Distributed Sparse Linear Regression," *IEEE Trans. Signal Process.*, vol. 58, pp. 5262–5276, Oct. 2010.

[106] G. Mateos and G. B. Giannakis, "Distributed recursive least-squares: Stability and performance analysis," *IEEE Trans. Signal Process.*, vol. 60, pp. 3740–3754, July 2012.

[107] G. Mateos and G. B. Giannakis, "Robust PCA as bilinear decomposition with outlier-sparsity regularization," *IEEE Trans. Signal Process.*, Sept. 2012. see also arXiv:1111.1788v1 [stat.ML].

[108] A. Montanari and S. Oh, "On Positioning via Distributed Matrix Completion," in *Proc. IEEE Int. Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, Jerusalem, pp. 197 – 200, Dec. 2010.

[109] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comput.*, vol. 24, pp. 227–234, 1995.

[110] 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.

[111] Y. Nesterov, "Smooth minimization of nonsmooth functions," *Math. Prog.*, vol. 103, pp. 127–152, 2005.

[112] R. Otazo, E. Candès, and D. K. Sodickson, "Low-rank plus sparse matrix decomposition for accelerated dynamic MRI with separation of background and dynamic components," *Magnetic Resonance in Medicine*, April 2014.

[113] K. Papagiannaki, R. Cruz, and C. Diot, "Network performance monitoring at small time scales," in *Proc. 1st ACM SIGCOMM Workshop on Internet Measurements*, Miami Beach, Florida, october 2003.

[114] H. Rauhut, "Compressive sensing and structured random matrices," *in Theoretical Foundations and Numerical Methods for Sparse Recovery*, vol. 9, pp. 1–92, 2010.

[115] S. Ravishankar and Y. Bresler, "MR image reconstruction from highly undersampled k-space data by dictionary learning," *IEEE Trans. on Medical Imaging*, vol. 30, pp. 1028–1041, May 2011.

[116] M. Razaviyayn, M. Hong, and Z.-Q. Luo, "A unified convergence analysis of block successive minimization methods for nonsmooth optimization," *SIAM Journal on Optimization*, vol. 23, pp. 1126–1153, 2013.

[117] 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.

[118] B. Recht and C. Re, "Parallel stochastic gradient algorithms for large-scale matrix completion," 2011. (submitted).

[119] M. Roughan, "A case study of the accuracy of SNMP measurements," *Journal of Electrical and Computer Engineering*, Dec. 2010. Article ID 812979.

[120] M. Roughan, Y. Zhang, W. Willinger, and L. Qiu, "Spatio-Temporal Compressive Sensing and Internet Traffic Matrices," *IEEE/ACM Trans. Networking*, 2012.

[121] M. Rudelson and R. Vershynin, "Sampling from large matrices: An approach through geometric functional analysis," *Journal of ACM*, vol. 54, pp. 1–20, Dec. 2006.

[122] W. Rudin, *Principles of Mathematical Analysis*. McGraw-Hill, third ed., 1976.

[123] S. G. Samko, A. A. Kilbas, and O. I. Marichev, *Fractional Integrals and Derivatives*. Yverdon, Switzerland: Gordon and Breach: Springer, 1993.

[124] A. H. Sayed, *Fundamentals of Adaptive Filtering*. John Wiley & Sons, 2003.

[125] I. D. Schizas, G. B. Giannakis, and Z. Q. Luo, "Distributed estimation using reduced-dimensionality sensor observations," *IEEE Trans. Signal Process.*, vol. 55, pp. 4284–4299, Aug. 2007.

[126] I. D. Schizas, G. Mateos, and G. B. Giannakis, "Distributed LMS for consesus-based in-network adaptive processing," *IEEE Trans. Signal Process.*, vol. 57, pp. 2365–2381, June 2009.

[127] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, "Consensus in ad hoc WSNs with noisy links - Part I: Distributed estimation of deterministic signals," *IEEE Trans. Signal Process.*, vol. 56, pp. 350–364, Jan. 2008.

[128] Y. Shavitt, X. Sun, A. Wool, and B. Yener, "Computing the unmeasured: An algebraic approach to Internet mapping," in *Proc. IEEE Intl. Conf. on Computer Commun.*, Alaska, USA, April 2001.

[129] M. Signoretto, R. V. Plas, B. D. Moor, and J. A. K. Suykens, "Tensor Versus Matrix Completion: A Comparison With Application to Spectral Data," *IEEE Signal Process. Letters*, vol. 18, pp. 403–406, July 2011.

[130] K. Slavakis and G. B. Giannakis, "Online learning from block-convex functions by accelerated stochastic approximation," *Trans. Signal Process.*, 2014 (submitted).

[131] V. Solo and X. Kong, *Adaptive Signal Processing Algorithms: Stability and Performance*, Prentice Hall, 1995.

[132] P. Sprechmann, A. M. Bronstein, and G. Sapiro, "Real-time online singing voice separation from monaural recordings using robust low-rank modeling," in *Proc. of Annual Conf. of the Intl. Society for Music Info. Retrieval*, Porto, Portugal, Oct. 2012.

[133] P. Sprechmann, I. Ramirez, G. Sapiro, and Y. Eldar, "C-HiLasso: A Collaborative Hierarchical Sparse Modeling Framework," *IEEE Trans. Signal Process.*, vol. 59, pp. 4183–4198, Sept. 2011.

[134] N. Srebro and A. Shraibman, "Rank, trace-norm and max-norm," in *Proc. of Learning Theory*, pp. 545–560, 2005.

[135] t. Ahmed, M. Coates, and A. Lakhina, "Multivariate Online Anomaly Detection Using Kernel Recursive Least Squares," in *Proc. of IEEE/ACM International Conference on Computer Communications*, Anchorage, Alaska, May 2007.

[136] M. Talagrand, "New concentration inequalities in product spaces," *Invent. Math.*, vol. 126, pp. 505–563, Dec. 1996.

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

[138] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Royal. Statist. Soc B*, vol. 58, pp. 267–288, 1996.

[139] K.-C. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized least-squares problems," *Pacific J. Opt.*, vol. 6, pp. 615–640, 2010.

[140] B. Trémoulhéac, N. Dikaios, D. Atkinson, and S. Arridge, "Dynamic MR image reconstruction–separation from under-sampled (k,t)-space via low-rank plus sparse prior," *IEEE Trans. on Medical Imaging*, pp. 689–701, August 2014.

[141] J. Tropp, "Just relax: Convex programming methods for identifying sparse signals," *IEEE Trans. Info. Theory*, vol. 51, pp. 1030–1051, Mar. 2006.

[142] J. Trzasko, A. Manduca, and E. Borisch, "Local versus global low-rank promotion in dynamic MRI series reconstruction," in *Proc. Intl. Symp. Magnetic Resononance in Medecine*, Montral, Canada, p. 4371, 2011.

[143] C.-M. Tsai and D. G. Nishimura, "Reduced aliasing artifacts using variable-density k-space sampling trajectories," *Magnetic Resonance in Medicine*, vol. 43, no. 3, pp. 452–458, 2000.

[144] P. Tseng, "Convergence of a Block Coordinate Descent Method for Nondifferentiable Minimization," *Journal of optimization theory and applications*, vol. 109, pp. 475–494, 2001.

[145] A. W. Van Der Vaart, *Asymptotic Statistics*. Cambridge University Press, 2000.

[146] Y. Vardi, "Network tomography: Estimating source-destination traffic intensities from link data," *Journal of American Statistical Association*, vol. 91, pp. 365 – 377, 1996.

[147] A. E. Waters, A. C. Sankaranarayanan, and R. G. Baraniuk, "SpaRCS: Recovering Low-Rank and Sparse Matrices from Compressive Measurements," in *Proc. of Neural Information Processing Systems*, Granada, Spain, Dec. 2011.

[148] J. Wright, A. Ganesh, and K. M. Y. Ma, "Compressive Principal Component Pursuit," in *Proc. of Intl. Symp. on Information Theory*, Cambridge, MA, pp. 1276–1280, July 2012.

[149] X. Wu, K. Yu, , and X. Wang, "On the growth of Internet application flows: A complex network perspective," in *Proc. IEEE Intl. Conf. on Computer Commun.*, Shangai, China, 2011.

[150] L. Xing, B. Thorndyke, E. Schreibmann, Y. Yang, T. Li, G. Kim, G. Luxton, A. Koong, *et al.*, "Overview of image-guided radiation therapy," *Medical Dosimetry*, vol. 31, no. 2, pp. 91–112, 2006.

[151] H. Xu, C. Caramanis, and S. Sanghavi, "Robust PCA via outlier pursuit," 2010 (see also arXiv:1010.4237v2 [cs.LG]).

[152] R. C. Y. Chi, Y. C. Eldar, "Petrels: Subspace Estimation And Tracking From Partial Observations," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, Kyoto, Japan, Mar. 2012.

[153] M. M. Y. Shen and G. B. Giannakis, "Online sketching of big categorical data with absent features," in *Proc. of IEEE International Conference on Information Sciences and Systems (CISS)*, pp. 516–520, Mar. 2015.

[154] B. Yang, "Projection approximation subspace tracking," *IEEE Trans. Signal. Process.*, vol. 43, pp. 95–107, Jan. 1995.

[155] J. Yang and J. Leskovec, "Overlapping community detection at scale: a nonnegative matrix factorization approach," in *Proceedings of the sixth ACM international conference on Web search and data mining*, pp. 587–596, ACM, 2013.

[156] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *J. Royal. Statist. Soc B*, vol. 68, pp. 49–67, 2006.

[157] T. Zhang, J. M. Pauly, and I. R. Levesque, "Accelerating parameter mapping with a locally low rank constraint," *Magnetic Resonance in Medicine*, feb 2014.

[158] Y. Zhang, Z. Ge, A. Greenberg, and M. Roughan, "Network Anomography," in *Proc. of Interent Measurement Conference*, CA, USA, pp. 317–330, Oct. 2005.

[159] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Fast accurate computation of large-scale IP traffic matrices from link loads," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 31, pp. 206–217, ACM, 2003.

[160] Y. Zhang, M. Roughan, W. Willinger, and L. Qiu, "Spatio-temporal compressive sensing and internet traffic matrices," in *Proc. of ACM SIGCOM Conf. on Data Commun.*, New York, USA, Oct. 2009.

[161] Z.-Y. Zhang, Y. Wang, and Y.-Y. Ahn, "Overlapping community detection in complex networks using symmetric binary matrix factorization," *Physical Review E*, vol. 87, no. 6, p. 062803, 2013.

[162] Q. Zhao, Z. Ge, J. Wang, and J. Xu, "Robust traffic matrix estimation with imperfect information: Making use of multiple data sources," vol. 34, pp. 133–144, 2006.

[163] Z. Zhou, X. Li, J. Wright, E. Candès, and Y. Ma, "Stable Principal Component Pursuit," in *Proc. of Intl. Symp. on Information Theory*, Austin, TX, pp. 1518–1522, June 2010.

[164] H. Zhu, A. Cano, and G. B. Giannakis, "Distributed Consensus-Based Demodulation: Algorithms and Error Analysis," *IEEE Trans. on Wireless Communications*, vol. 9, 2010. (to appear).

[165] J. Ziniel and P. Schniter, "Dynamic compressive sensing of time-varying signals via approximate message passing," *IEEE Trans. Signal Process.*, vol. 61, pp. 5270 – 5284, November 2013.

[166] H. V. Zuylen and L. Willumsen, "The most likely trip matrix estimated from traffic counts," *Transportation Research, Part B: Methodological*, vol. 14, pp. 281–293, 1980.

Appendix

Proof of Lemma 2.2

Suppose $\{\mathbf{X}_0, \mathbf{A}_0\}$ is an optimal solution of (P1). For the nuclear norm and the ℓ_1-norm at point $\{\mathbf{X}_0, \mathbf{A}_0\}$ pick the subgradients $\mathbf{U}\mathbf{V}' + \mathbf{W}_0$ and $\text{sign}(\mathbf{A}_0) + \mathbf{F}_0$, respectively, satisfying the optimality condition

$$\lambda \text{sign}(\mathbf{A}_0) + \lambda \mathbf{F} = \mathbf{R}'(\mathbf{U}\mathbf{V}' + \mathbf{W}). \tag{7.3}$$

Consider a feasible solution $\{\mathbf{X}_0 + \mathbf{R}\mathbf{H}, \mathbf{A}_0 - \mathbf{H}\}$ for arbitrary nonzero $\mathbf{H}$. The subgradient inequality yields

$$\begin{aligned}\|\mathbf{X}_0 + \mathbf{R}\mathbf{H}\|_* + \lambda\|\mathbf{A}_0 - \mathbf{H}\| \geq \|\mathbf{X}_0\|_* + \lambda\|\mathbf{A}_0\|_1 \\ + \underbrace{\langle \mathbf{U}\mathbf{V}' + \mathbf{W}_0, \mathbf{R}\mathbf{H}\rangle - \lambda\langle \text{sgn}(\mathbf{A}_0) + \mathbf{F}_0, \mathbf{H}\rangle}_{:=\varphi(\mathbf{H})}.\end{aligned}$$

To guarantee uniqueness, $\varphi(\mathbf{H})$ must be positive. Rearranging terms one obtains

$$\varphi(\mathbf{H}) = \langle \mathbf{W}_0, \mathbf{R}\mathbf{H}\rangle - \lambda\langle \mathbf{F}_0, \mathbf{H}\rangle + \langle \mathbf{R}'\mathbf{U}\mathbf{V}' - \lambda \text{sign}(\mathbf{A}_0), \mathbf{H}\rangle. \tag{7.4}$$

The value of $\mathbf{W}_0$ can be chosen such that $\langle \mathbf{W}_0, \mathbf{R}\mathbf{H}\rangle = \|\mathcal{P}_{\Phi^\perp}(\mathbf{R}\mathbf{H})\|_*$. This is because, $\|\mathcal{P}_{\Phi^\perp}(\mathbf{R}\mathbf{H})\|_* = \sup_{\|\bar{\mathbf{W}}\|\leq 1} |\langle \bar{\mathbf{W}}, \mathcal{P}_{\Phi^\perp}(\mathbf{R}\mathbf{H})\rangle|$, thus there exists a $\bar{\mathbf{W}}$ such that $\langle \mathcal{P}_{\Phi^\perp}(\bar{\mathbf{W}}), \mathbf{R}\mathbf{H}\rangle = \|\mathcal{P}_{\Phi^\perp}(\mathbf{R}\mathbf{H})\|_*$. One can then choose $\mathbf{W}_0 := \mathcal{P}_{\Phi^\perp}(\bar{\mathbf{W}})$ since $\|\mathcal{P}_{\Phi^\perp}(\bar{\mathbf{W}})\| \leq \|\bar{\mathbf{W}}\| \leq 1$ and $\mathcal{P}_\Phi(\mathbf{W}_0) = \mathbf{0}_{L\times T}$. Similarly, if one selects $\mathbf{F}_0 := -\mathcal{P}_{\Omega^\perp}(\text{sign}(\mathbf{H}))$, which satisfies $\mathcal{P}_\Omega(\mathbf{F}_0) = \mathbf{0}_{F\times T}$ and $\|\mathbf{F}_0\|_\infty = 1$, then $\langle \mathbf{F}_0, \mathbf{H}\rangle = -\|\mathcal{P}_{\Omega^\perp}(\mathbf{H})\|_1$. Now, using (7.3), equation (7.4) is expressed as

$$\varphi(\mathbf{H}) = \|\mathcal{P}_{\Phi^\perp}(\mathbf{R}\mathbf{H})\| + \lambda\|\mathcal{P}_{\Omega^\perp}(\mathbf{H})\| + \langle \lambda\mathbf{F} - \mathbf{R}'\mathbf{W}, \mathbf{H}\rangle.$$

From the triangle inequality $|\langle \lambda \mathbf{F} - \mathbf{R}'\mathbf{W}, \mathbf{H}\rangle| \leq \lambda |\langle \mathbf{F}, \mathbf{H}\rangle| + |\langle \mathbf{R}'\mathbf{W}, \mathbf{H}\rangle|$, it thus follows that

$$\varphi(\mathbf{H}) \geq \left(\|\mathcal{P}_{\Phi^\perp}(\mathbf{RH})\|_* - |\langle \mathbf{R}'\mathbf{W}, \mathbf{H}\rangle| \right) + \lambda \left(\|\mathcal{P}_{\Omega^\perp}(\mathbf{H})\|_1 - |\langle \mathbf{F}, \mathbf{H}\rangle| \right). \tag{7.5}$$

Since $\mathcal{P}_{\Phi^\perp}(\mathbf{W}) = \mathbf{W}$, it is deduced that $|\langle \mathbf{W}, \mathbf{RH}\rangle| = |\langle \mathbf{W}, \mathcal{P}_{\Phi^\perp}(\mathbf{RH})\rangle| \leq \|\mathbf{W}\| \|\mathcal{P}_{\Phi^\perp}(\mathbf{RH})\|_*$. Likewise, $\mathcal{P}_{\Omega^\perp}(\mathbf{F}) = \mathbf{F}$ yields $|\langle \mathbf{F}, \mathbf{H}\rangle| = |\langle \mathbf{F}, \mathcal{P}_{\Omega^\perp}(\mathbf{H})\rangle| \leq \|\mathbf{F}\|_\infty \|\mathcal{P}_{\Omega^\perp}(\mathbf{H})\|_1$. As a result

$$\begin{aligned} \varphi(\mathbf{H}) &\geq (1 - \|\mathbf{W}\|)\|\mathcal{P}_{\Phi}(\mathbf{RH})\|_* + \lambda(1 - \|\mathbf{F}\|_\infty)\|\mathcal{P}_{\Omega^\perp}(\mathbf{H})\|_1 \\ &\geq (1 - \max\{\|\mathbf{W}\|, \|\mathbf{F}\|_\infty\})\{\|\mathcal{P}_{\Phi^\perp}(\mathbf{RH})\|_* + \lambda \|\mathcal{P}_{\Omega^\perp}(\mathbf{H})\|_1\}. \end{aligned} \tag{7.6}$$

Now, if $\|\mathbf{W}\| < 1$ and $\|\mathbf{F}\|_\infty < 1$, since $\Phi \cap \Omega_R = \{\mathbf{0}_{L\times T}\}$ and $\mathbf{RH} \neq \mathbf{0}_{L\times T}$, $\forall \mathbf{H} \in \Omega \backslash \{\mathbf{0}_{F\times T}\}$, there is no $\mathbf{H} \in \Omega$ for which $\mathbf{RH} \in \Phi$, and therefore, $\varphi(\mathbf{H}) > 0$.

Since $\mathbf{W}$ and $\mathbf{F}$ are related through (7.3), upon defining $\mathbf{\Gamma} := \mathbf{R}'(\mathbf{UV}' + \mathbf{W})$, which is indeed the dual variable for (P1), one can arrive at conditions C1)-C4). ■

Proof of Lemma 2.3

To establish that the rows of $\mathbf{A}_\Omega$ are linearly independent, it suffices to show that $\|\mathbf{A}'\text{vec}(\mathbf{H})\| > 0$, for all nonzero $\mathbf{H} \in \Omega$. It is then possible to

$$\begin{aligned} \|\mathbf{A}'\text{vec}(\mathbf{H})\| &= \|(\mathbf{I} - \mathbf{P}_V) \otimes (\mathbf{I} - \mathbf{P}_U)\mathbf{R}\text{vec}(\mathbf{H})\| \\ &= \|(\mathbf{I} - \mathbf{P}_U)\mathbf{RH}(\mathbf{I} - \mathbf{P}_V)\|_F \\ &= \|\mathcal{P}_{\Phi^\perp}(\mathbf{RH})\|_F = \|\mathbf{RH} - \mathcal{P}_{\Phi}(\mathbf{RH})\|_F \\ &\overset{(a)}{\geq} \|\mathbf{RH}\|_F - \|\mathcal{P}_{\Phi}(\mathbf{RH})\|_F \\ &\overset{(b)}{\geq} \|\mathbf{RH}\|_F (1 - \mu(\Omega_R, \Phi)) \end{aligned} \tag{7.7}$$

where (a) follows from the triangle inequality, and (b) from (2.6). The assumption $\delta_k(\mathbf{R}) < 1$ along with the fact that no column of $\mathbf{H}$ has more than k nonzero elements, imply that $\mathbf{RH} \neq \mathbf{0}_{L\times T}$. Since $\mu(\Omega_r, \Phi) < 1$ by assumption, the claim follows from (7.7).

To arrive at the desired bound on $\sigma_{\min}(\mathbf{A}'_\Omega)$, recall the definition of the minimum singular value [58]

$$\begin{aligned}
\sigma_{\min}(\mathbf{A}'_\Omega) &= \min_{\mathbf{H}\in\Omega\setminus\{\mathbf{0}_{F\times T}\}} \frac{\|\mathbf{A}'\mathrm{vec}(\mathbf{H})\|}{\|\mathrm{vec}(\mathbf{H})\|} \\
&= \min_{\mathbf{H}\in\Omega\setminus\{\mathbf{0}_{F\times T}\}} \frac{\|(\mathbf{I}-\mathbf{P}_U)\mathbf{R}\mathbf{H}(\mathbf{I}-\mathbf{P}_V)\|_F}{\|\mathbf{H}\|_F} \\
&\overset{(c)}{=} \min_{\mathbf{H}\in\Omega\setminus\{\mathbf{0}_{F\times T}\}} \frac{\|\mathbf{R}\mathbf{H}\|_F}{\|\mathbf{H}\|_F} \times \frac{\|\mathcal{P}_{\Phi^\perp}(\mathbf{R}\mathbf{H})\|_F}{\|\mathbf{R}\mathbf{H}\|_F} \\
&\overset{(d)}{\geq} c^{1/2}(1-\delta_k(\mathbf{R}))^{1/2} \min_{\mathbf{Z}\in\Omega_R\setminus\{\mathbf{0}_{L\times T}\}} \frac{\|\mathcal{P}_{\Phi^\perp}(\mathbf{Z})\|_F}{\|\mathbf{Z}\|_F} \\
&= c^{1/2}(1-\delta_k(\mathbf{R}))^{1/2} \min_{\mathbf{Z}\in\Omega_R\setminus\{\mathbf{0}\}} \frac{\|\mathbf{Z}-\mathcal{P}_{\Phi}(\mathbf{Z})\|_F}{\|\mathbf{Z}\|_F} \\
&\overset{(e)}{\geq} c^{1/2}(1-\delta_k(\mathbf{R}))^{1/2} \left(1 - \max_{\mathbf{Z}\in\Omega_R\setminus\{\mathbf{0}\}} \frac{\|\mathcal{P}_{\Phi}(\mathbf{Z})\|_F}{\|\mathbf{Z}\|_F}\right) \\
&\overset{(f)}{=} c^{1/2}(1-\delta_k(\mathbf{R}))^{1/2}(1-\mu(\Phi,\Omega_R)).
\end{aligned}$$

In obtaining (c), the assumption $\delta_k(\mathbf{R}) < 1$ along with the fact that no column of $\mathbf{H}$ has more than k nonzero elements was used to ensure that $\mathbf{R}\mathbf{H} \neq \mathbf{0}_{L\times T}$. In addition, (d) and (f) follow from the definitions (2.7) and (2.6), respectively, while (e) follows from the triangle inequality. ■

Proof of Lemma 2.4

Towards establishing the first bound, from the submultiplicative property of the spectral norm one obtains

$$\|\mathbf{Q}\| = \|\mathbf{A}_{\Omega^\perp}\mathbf{A}'_\Omega\left(\mathbf{A}_\Omega\mathbf{A}'_\Omega\right)^{-1}\| \leq \|\mathbf{A}_{\Omega^\perp}\|\|\mathbf{A}'_\Omega\left(\mathbf{A}_\Omega\mathbf{A}'_\Omega\right)^{-1}\|. \tag{7.8}$$

5 Next, upper bounds are derived for both factors on the right-hand side of (7.8). First, using the fact that $\mathbf{A}'\mathbf{A} = \mathbf{A}'_{\Omega}\mathbf{A}_{\Omega} + \mathbf{A}'_{\Omega^\perp}\mathbf{A}_{\Omega^\perp}$ one arrives at

$$\begin{aligned}
\|\mathbf{A}_{\Omega^\perp}\|^2 &= \max_{\mathbf{x}\neq\mathbf{0}} \frac{\mathbf{x}'\mathbf{A}'_{\Omega^\perp}\mathbf{A}_{\Omega^\perp}\mathbf{x}}{\|\mathbf{x}\|^2} \\
&= \max_{\mathbf{x}\neq\mathbf{0}} \frac{\mathbf{x}'(\mathbf{A}'\mathbf{A} - \mathbf{A}'_{\Omega}\mathbf{A}_{\Omega})\mathbf{x}}{\|\mathbf{x}\|^2} \\
&\leq \max_{\mathbf{x}\neq\mathbf{0}} \frac{\mathbf{x}'\mathbf{A}'\mathbf{A}\mathbf{x}}{\|\mathbf{x}\|^2} - \min_{\mathbf{x}\neq\mathbf{0}} \frac{\mathbf{x}'\mathbf{A}'_{\Omega}\mathbf{A}_{\Omega}\mathbf{x}}{\|\mathbf{x}\|^2} \\
&= \|\mathbf{A}\|^2 - \sigma^2_{\min}(\mathbf{A}'_{\Omega}). \qquad (7.9)
\end{aligned}$$

Note that $\mathbf{A}'_{\Omega}\left(\mathbf{A}_{\Omega}\mathbf{A}'_{\Omega}\right)^{-1}$ is the pseudo-inverse of the full row rank matrix $\mathbf{A}_{\Omega}$ (cf. Lemma 2.3), and thus $\|\mathbf{A}'_{\Omega}\left(\mathbf{A}_{\Omega}\mathbf{A}'_{\Omega}\right)^{-1}\| = \sigma^{-1}_{\min}(\mathbf{A}'_{\Omega})$ [58]. Substituting these two bounds into (7.8) yields

$$\|\mathbf{A}_{\Omega^\perp}\mathbf{A}'_{\Omega}\left(\mathbf{A}_{\Omega}\mathbf{A}'_{\Omega}\right)^{-1}\| \leq \left\{\left(\frac{\|\mathbf{A}\|}{\sigma_{\min}(\mathbf{A}'_{\Omega})}\right)^2 - 1\right\}^{1/2}. \qquad (7.10)$$

In addition, it holds that

$$\begin{aligned}
\|\mathbf{A}\|^2 &= \lambda_{\max}\left\{(\mathbf{I}-\mathbf{P}_V)\otimes\mathbf{R}'(\mathbf{I}-\mathbf{P}_U)\mathbf{R}\right\} \\
&= \lambda_{\max}\{(\mathbf{I}-\mathbf{P}_V)\} \times \lambda_{\max}\left\{\mathbf{R}'(\mathbf{I}-\mathbf{P}_U)\mathbf{R}\right\} \\
&\overset{(a)}{=} \|\mathbf{R}'(\mathbf{I}-\mathbf{P}_U)\|^2 \overset{(b)}{=} 1. \qquad (7.11)
\end{aligned}$$

where in (a) and (b) it was used that the rows of $\mathbf{R}$ are orthonormal, and the maximum singular value of a projection matrix is one. Substituting (7.11) and the bound of Lemma 2.3 into (7.10), leads to (2.4).

In order to prove the second bound, first suppose that $\|\mathbf{I}-\mathbf{A}_{\Omega}\mathbf{A}'_{\Omega}\|_{\infty,\infty} < 1$. Then, one can write

$$\begin{aligned}
\|\mathbf{A}_{\Omega^\perp}\mathbf{A}'_{\Omega}\left(\mathbf{A}_{\Omega}\mathbf{A}'_{\Omega}\right)^{-1}\|_{\infty,\infty} &\leq \|\mathbf{A}_{\Omega^\perp}\mathbf{A}'_{\Omega}\|_{\infty,\infty}\|\left(\mathbf{I}-(\mathbf{I}-\mathbf{A}_{\Omega}\mathbf{A}'_{\Omega})\right)^{-1}\|_{\infty,\infty} \\
&\leq \frac{\|\mathbf{A}_{\Omega^\perp}\mathbf{A}'_{\Omega}\|_{\infty,\infty}}{1-\|\mathbf{I}-\mathbf{A}_{\Omega}\mathbf{A}'_{\Omega}\|_{\infty,\infty}}. \qquad (7.12)
\end{aligned}$$

In what follows, separate upper bounds are derived for $\|\mathbf{A}_{\Omega^\perp}\mathbf{A}'_{\Omega}\|_{\infty,\infty}$ and $\|\mathbf{I}-\mathbf{A}_{\Omega}\mathbf{A}'_{\Omega}\|_{\infty,\infty}$. For notational convenience introduce $\mathcal{S} := \mathrm{supp}(\mathbf{A}_0)$ (resp. $\bar{\mathcal{S}}$ denotes the set complement).

Starting with the numerator in the right-hand side of (7.12)

$$\begin{aligned}\|\mathbf{A}_{\Omega^\perp}\mathbf{A}_\Omega'\|_{\infty,\infty} &= \max_i \|\mathbf{e}_i'\mathbf{A}_{\Omega^\perp}\mathbf{A}_\Omega'\|_1 \\ &= \max_i \sum_k |\langle \mathbf{e}_i'\mathbf{A}_{\Omega^\perp}, \mathbf{e}_k'\mathbf{A}_\Omega\rangle| \\ &= \max_j \sum_\ell |\langle \mathbf{e}_j'\mathbf{A}, \mathbf{e}_\ell'\mathbf{A}\rangle| \\ &= \max_{(j_1,j_2)\in\bar{\mathcal{S}}} \sum_{(\ell_1,\ell_2)\in\mathcal{S}} g(j_1,j_2,\ell_1,\ell_2) \end{aligned} \tag{7.13}$$

where $g(j_1,j_2,\ell_1,\ell_2) := |\langle \mathbf{R}\mathbf{e}_{j_1}\mathbf{e}_{j_2}'(\mathbf{I}-\mathbf{P}_V), (\mathbf{I}-\mathbf{P}_U)\mathbf{R}\mathbf{e}_{\ell_1}\mathbf{e}_{\ell_2}'\rangle|$. Following some manipulations, the summands in (7.13) can be expressed as

$$\begin{aligned} g(j_1,j_2,\ell_1,\ell_2) &= |\langle \mathbf{R}\mathbf{e}_{j_1}\mathbf{e}_{j_2}', (\mathbf{I}-\mathbf{P}_U)\mathbf{R}\mathbf{e}_{\ell_1}\mathbf{e}_{\ell_2}'\rangle \\ &\quad - \langle \mathbf{R}\mathbf{e}_{j_1}\mathbf{e}_{j_2}'\mathbf{P}_V, (\mathbf{I}-\mathbf{P}_U)\mathbf{R}\mathbf{e}_{\ell_1}\mathbf{e}_{\ell_2}'\rangle| \\ &= |\langle \mathbf{e}_{j_2}'\mathbf{e}_{\ell_2}, \mathbf{e}_{j_1}'\mathbf{R}'(\mathbf{I}-\mathbf{P}_U)\mathbf{R}\mathbf{e}_{\ell_1}\rangle \\ &\quad - \langle \mathbf{e}_{j_2}'\mathbf{P}_V\mathbf{e}_{\ell_2}, \mathbf{e}_{j_1}'\mathbf{R}'(\mathbf{I}-\mathbf{P}_U)\mathbf{R}\mathbf{e}_{\ell_1}\rangle| \\ &= |\mathbf{e}_{j_1}'\mathbf{R}'(\mathbf{I}-\mathbf{P}_U)\mathbf{R}\mathbf{e}_{\ell_1}\mathbb{1}_{\{j_2=\ell_2\}} \\ &\quad - (\mathbf{e}_{j_2}'\mathbf{P}_V\mathbf{e}_{\ell_2})(\mathbf{e}_{j_1}'\mathbf{R}'(\mathbf{I}-\mathbf{P}_U)\mathbf{R}\mathbf{e}_{\ell_1})|. \end{aligned} \tag{7.14}$$

Upon defining $x_{j_1,\ell_1} := \mathbf{e}_{j_1}'\mathbf{R}'(\mathbf{I}-\mathbf{P}_U)\mathbf{R}\mathbf{e}_{\ell_1}$ and $y_{j_2,\ell_2} := (\mathbf{e}_{j_2}'\mathbf{P}_V\mathbf{e}_{\ell_2})$, squaring g gives rise to

$$g^2(j_1,j_2,\ell_1,\ell_2) = x_{j_1,\ell_1}^2\mathbb{1}_{\{j_2=\ell_2\}} + y_{j_2,\ell_2}^2 x_{j_1,\ell_1}^2 - 2y_{j_2,\ell_2}x_{j_1,\ell_1}^2\mathbb{1}_{\{j_2=\ell_2\}}. \tag{7.15}$$

Since $y_{j_2,\ell_2}\mathbb{1}_{\{j_2=\ell_2\}} = \|\mathbf{P}_V\mathbf{e}_{j_2}\|^2\mathbb{1}_{\{j_2=\ell_2\}} \geq 0$, one can ignore the third summand in (7.15) to arrive at

$$g(j_1,j_2,\ell_1,\ell_2) \leq x_{j_1,\ell_1}[\mathbb{1}_{\{j_2=\ell_2\}} + y_{j_2,\ell_2}^2]^{1/2}. \tag{7.16}$$

Towards bounding the scalars x_{j_1,ℓ_1} and y_{j_2,ℓ_2}, rewrite $x_{j_1,\ell_1} := \mathbf{e}_{j_1}'\mathbf{R}'\mathbf{R}\mathbf{e}_{\ell_1} - \mathbf{e}_{j_1}'\mathbf{R}'\mathbf{P}_U\mathbf{R}\mathbf{e}_{\ell_1}$. If $j_1 = \ell_1$, it holds that $x_{j_1,\ell_1} \leq \|\mathbf{R}\mathbf{e}_{\ell_1}\|^2 \leq c(1+\delta_1(\mathbf{R}))$; otherwise,

$$\begin{aligned} x_{j_1,\ell_1} &\leq |\mathbf{e}_{j_1}'\mathbf{R}'\mathbf{R}\mathbf{e}_{\ell_1}| + |\mathbf{e}_{j_1}'\mathbf{R}'\mathbf{P}_U\mathbf{R}\mathbf{e}_{\ell_1}| \\ &\leq c\theta_{1,1}(\mathbf{R}) + c(1+\delta_1(\mathbf{R}))\gamma_R^2(\mathbf{U}). \end{aligned}$$

Moreover, $y_{j_2,\ell_2} \leq \|\mathbf{P}_V \mathbf{e}_{j_2}\| \|\mathbf{P}_V \mathbf{e}_{\ell_2}\| \leq \gamma^2(\mathbf{V})$. Plugging the bounds into (7.16) yields

$$g(j_1, j_2, \ell_1, \ell_2) \leq \big[c(1+\delta_1(\mathbf{R}))\mathbb{1}_{\{j_1=\ell_1\}} + c(\theta_{1,1}(\mathbf{R}) \\ + c(1+\delta_1(\mathbf{R}))\gamma_R^2(\mathbf{U}))\mathbb{1}_{\{j_1\neq\ell_1\}}\big] \left[\mathbb{1}_{\{j_2=\ell_2\}} + \gamma^4(\mathbf{V})\right]^{1/2}. \tag{7.17}$$

Plugging (7.17) into (7.13) one arrives at

$$\begin{aligned} \|\mathbf{A}_{\Omega^\perp} \mathbf{A}_\Omega'\|_{\infty,\infty} &\leq c[\sqrt{2}k + s\gamma^2(\mathbf{V})]\theta_{1,1}(\mathbf{R}) \\ &\quad + c(1+\delta_1(\mathbf{R}))\left[k\gamma^2(\mathbf{V}) + \sqrt{2}k\gamma_R^2(\mathbf{U})s\gamma_R^2(\mathbf{U})\gamma^2(\mathbf{V})\right] \\ &:= c\omega \end{aligned} \tag{7.18}$$

after using: i) $\mathcal{S} \cap \bar{\mathcal{S}} = \emptyset$ and consequently $j_2 \neq \ell_2$ when $j_1 = \ell_1$; and ii) $\gamma(\mathbf{V}) \leq 1$.

Moving on, consider bounding $\|\mathbf{I} - \mathbf{A}_\Omega \mathbf{A}_\Omega'\|_{\infty,\infty}$ that can be rewritten as

$$\begin{aligned} \|\mathbf{I} - \mathbf{A}_\Omega \mathbf{A}_\Omega{}'\|_{\infty,\infty} &= \max_i \|\mathbf{e}_i{}'(\mathbf{I} - \mathbf{A}_\Omega \mathbf{A}_\Omega{}')\|_1 \\ &= \max_i \left\{|1 - \|\mathbf{e}_i{}'\mathbf{A}_\Omega\|^2| + \sum_{k\neq i} |\langle \mathbf{e}_i{}'\mathbf{A}_\Omega, \mathbf{e}_k{}'\mathbf{A}_\Omega\rangle|\right\} \\ &= \max_{\substack{j=j_1+j_2 \\ (j_1,j_2)\in\mathcal{S}}} \left\{|1 - \|\mathbf{A}'\mathbf{e}_j\|^2| + \sum_{\ell\neq j} |\langle \mathbf{A}'\mathbf{e}_j, \mathbf{A}'\mathbf{e}_\ell\rangle|\right\}. \end{aligned} \tag{7.19}$$

In the sequel, an upper bound is derived for (7.19). Let (j_1, j_2) denote the element of $\mathcal{S}$ associated with j in (7.19). For the first summand inside the curly brackets in (7.19), consider lower bounding the norm of the j-th row of $\mathbf{A}$ as

$$\begin{aligned} \|\mathbf{A}'\mathbf{e}_j\| &= \|(\mathbf{I} - \mathbf{P}_U)\mathbf{R}\mathbf{e}_{j_1}\mathbf{e}_{j_2}'(\mathbf{I} - \mathbf{P}_V)\|_F \\ &= \|\mathcal{P}_{\Phi^\perp}(\mathbf{R}\mathbf{e}_{j_1}\mathbf{e}_{j_2}')\|_F \\ &= \|\mathbf{R}\mathbf{e}_{j_1}\mathbf{e}_{j_2}' - \mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_{j_1}\mathbf{e}_{j_2}')\|_F \\ &\geq \|\mathbf{R}\mathbf{e}_{j_1}\mathbf{e}_{j_2}'\| - \|\mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_{j_1}\mathbf{e}_{j_2}')\|_F \\ &\geq \|\mathbf{R}\mathbf{e}_{j_1}\mathbf{e}_{j_2}'\|(1 - \mu(\Phi, \Omega_R)) \\ &\geq c^{1/2}(1-\delta_1(\mathbf{R}))^{1/2}(1 - \mu(\Phi, \Omega_R)). \end{aligned}$$

Since $\delta_1(\mathbf{R}) < 1$ and $\mu(\Phi, \Omega_R) < 1$, one obtains $|1 - \|\mathbf{A}'\mathbf{e}_j\|^2| \leq 1 - c(1-\delta_1(\mathbf{R}))(1 - \mu(\Phi, \Omega_R))^2$.

For the second summand inside the curly brackets in (7.19), a procedure similar to the one used for bounding $\|\mathbf{A}_{\Omega^\perp}\mathbf{A}_\Omega{}'\|_{\infty,\infty}$ is pursued. First, observe that

$$\begin{aligned}\sum_{\ell\neq j}|\langle \mathbf{A}\mathbf{A}'\mathbf{e}_j,\mathbf{e}_\ell\rangle| &= \sum_{\ell\neq j}|\langle (\mathbf{I}-\mathbf{P}_V)\otimes\mathbf{R}'(\mathbf{I}-\mathbf{P}_U)\mathbf{R}\mathbf{e}_j,\mathbf{e}_\ell\rangle| \\ &= \sum_{(\ell_1,\ell_2)\in\mathcal{S}\setminus\{(j_1,j_2)\}} g(j_1,j_2,\ell_1,\ell_2) \qquad (7.20)\end{aligned}$$

to deduce that, up to a summand corresponding to the index pair (j_1, j_2), (7.20) is identical to the summation in (7.13). Following similar arguments to those leading to (7.17), one arrives at

$$\max_{\substack{j=j_1+j_2\\(j_1,j_2)\in\mathcal{S}}}\sum_{\ell\neq j}|\langle \mathbf{A}'\mathbf{e}_j,\mathbf{A}'\mathbf{e}_\ell\rangle| \leq c\omega.$$

Putting all the pieces together, (7.19) is bounded as

$$\|\mathbf{I}-\mathbf{A}_\Omega\mathbf{A}_\Omega'\|_{\infty,\infty} \leq 1-c(1-\delta_1(\mathbf{R}))(1-\mu(\Phi,\Omega_R))^2+c\omega. \qquad (7.21)$$

Note that because of the assumption $\omega < (1-\delta_1(\mathbf{R}))(1-\mu(\Phi,\Omega_R))^2$, $\|\mathbf{I}-\mathbf{A}_\Omega\mathbf{A}_\Omega'\|_{\infty,\infty}<1$ as supposed at the beginning of the proof. Substituting (7.18) and (7.21) into (7.12) yields the desired bound. ■

Proof of Lemma 2.6

The proof bears some resemblance with those available for the matrix completion problem [27], and PCP [25]. However, presence of the compression matrix $\mathbf{R}$ gives rise to unique challenges in some stages of the proof, which necessitate special treatment. In what follows, emphasis is placed on the distinct arguments required by the setting here.

The main idea is to obtain first an upper bound on the norm of the linear operator $\pi^{-1}\mathcal{P}_\Phi\mathbf{R}\mathcal{P}_\Omega\mathbf{R}'\mathcal{P}_\Phi-\mathcal{P}_\Phi$, which is then utilized to upper bound $\mu(\Phi,\Omega_R)=\|\mathcal{P}_\Phi\mathbf{R}\mathcal{P}_\Omega\|$. The former is established in the next lemma; see Appendix E for a proof.

Lemma 7.1 *Suppose $\mathcal{S} := supp(\mathbf{A}_0)$ is drawn according to the Bernoulli model with parameter π. Let $\Lambda := \sqrt{c(1+\delta_1(\mathbf{R}))[\gamma_R^2(\mathbf{U})+\gamma^2(\mathbf{V})]}$, and $n := \max\{L,F\}$. Then, there*

are positive numerical constants C and τ such that

$$\pi^{-1}\|\mathcal{P}_\Phi \mathbf{R}\mathcal{P}_\Omega \mathbf{R}'\mathcal{P}_\Phi - \pi\mathcal{P}_\Phi\| \leq C\sqrt{\frac{\log(LF)}{\pi}} + \tau\Lambda\log(n) \tag{7.22}$$

holds with probability higher than $1 - \mathcal{O}\left(n^{-C\pi\Lambda\tau}\right)$, provided that the right-hand side is less than one.

Building on (7.22), it follows that

$$\begin{aligned}\|\mathcal{P}_\Phi \mathbf{R}\mathcal{P}_\Omega \mathbf{R}'\mathcal{P}_\Phi\| - \pi &\overset{(a)}{\leq} \|\mathcal{P}_\Phi \mathbf{R}\mathcal{P}_\Omega \mathbf{R}'\mathcal{P}_\Phi\| - \pi\|\mathcal{P}_\Phi\| \\ &\overset{(b)}{\leq} \|\mathcal{P}_\Phi \mathbf{R}\mathcal{P}_\Omega \mathbf{R}'\mathcal{P}_\Phi - \pi\mathcal{P}_\Phi\| \\ &\leq C\sqrt{\pi\log(LF)} + \tau\pi\Lambda\log(n)\end{aligned} \tag{7.23}$$

where (a) and (b) come from $\|\mathcal{P}_\Phi\| \leq 1$ and the triangle inequality, respectively. In addition,

$$\begin{aligned}\|\mathcal{P}_\Omega(\mathbf{R}'\mathcal{P}_\Phi(\mathbf{X}))\|_F^2 &= |\langle \mathcal{P}_\Omega(\mathbf{R}'\mathcal{P}_\Phi(\mathbf{X})), \mathcal{P}_\Omega(\mathbf{R}'\mathcal{P}_\Phi(\mathbf{X}))\rangle| \\ &= |\langle \mathcal{P}_\Phi(\mathbf{R}(\mathcal{P}_\Omega(\mathbf{R}'\mathcal{P}_\Phi(\mathbf{X})))), \mathbf{X}\rangle| \\ &\leq \|\mathcal{P}_\Phi(\mathbf{R}(\mathcal{P}_\Omega(\mathbf{R}'\mathcal{P}_\Phi(\mathbf{X}))))\|_F \|\mathbf{X}\|_F\end{aligned} \tag{7.24}$$

for all $\mathbf{X} \in \mathbb{R}^{L\times F}$. Recalling the definition of the operator norm, it follows from (7.24) that $\mu(\Phi, \Omega_R) \leq \sqrt{c^{-1}(1-\delta_k(\mathbf{R}))^{-1}}\|\mathcal{P}_\Phi \mathbf{R}\mathcal{P}_\Omega \mathbf{R}'\mathcal{P}_\Phi\|^{1/2}$. Plugging the bound (7.23), the result follows readily. ■

Proof of Lemma 7.1

Start by noting that

$$\begin{aligned}\mathbf{R}'\mathcal{P}_\Phi(\mathbf{X}) &= \sum_{i,j} \langle \mathbf{R}'\mathcal{P}_\Phi(\mathbf{X}), \mathbf{e}_i\mathbf{e}_j'\rangle \mathbf{e}_i\mathbf{e}_j' \\ &= \sum_{i,j} \langle \mathbf{X}, \mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_i\mathbf{e}_j')\rangle \mathbf{e}_i\mathbf{e}_j'\end{aligned}$$

and apply the sampling operator to obtain

$$\mathcal{P}_\Omega(\mathbf{R}'\mathcal{P}_\Phi(\mathbf{X})) = \sum_{i,j} b_{i,j}\langle \mathbf{X}, \mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_i\mathbf{e}_j')\rangle \mathbf{e}_i\mathbf{e}_j'$$

where $\{b_{i,j}\}$ are Bernoulli-distributed i.i.d. random variables with $\Pr(b_{i,j}=1)=\pi$. Then,

$$\mathcal{P}_\Omega(\mathbf{R}\mathcal{P}_\Omega(\mathbf{R}'\mathcal{P}_\Phi(\mathbf{X}))) = \sum_{i,j} b_{i,j}\langle \mathbf{X}, \mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_i\mathbf{e}_j')\rangle \mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_i\mathbf{e}_j'). \tag{7.25}$$

Moreover, since $\mathbf{R}\mathbf{R}'=\mathbf{I}_L$ one finally arrives at

$$\begin{aligned}\mathcal{P}_\Phi(\mathbf{X}) &= \mathcal{P}_\Phi(\mathbf{R}\mathbf{R}'\mathcal{P}_\Phi(\mathbf{X})) \\ &= \sum_{i,j} b_{i,j}\langle \mathbf{X}, \mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_i\mathbf{e}_j')\rangle \mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_i\mathbf{e}_j').\end{aligned} \tag{7.26}$$

The next bound will also be useful later on

$$\begin{aligned}\|\mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_i\mathbf{e}_j{}')\|_F^2 &= \langle \mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_i\mathbf{e}_j{}'), \mathbf{R}\mathbf{e}_i\mathbf{e}_j{}'\rangle \\ &= \langle \mathbf{P}_U\mathbf{R}\mathbf{e}_i\mathbf{e}_j{}' + \mathbf{R}\mathbf{e}_i\mathbf{e}_j{}'\mathbf{P}_V - \mathbf{P}_U\mathbf{R}\mathbf{e}_i\mathbf{e}_j{}'\mathbf{P}_V, \mathbf{R}\mathbf{e}_i\mathbf{e}_j{}'\rangle \\ &\overset{(a)}{=} \|\mathbf{P}_U\mathbf{R}\mathbf{e}_i\mathbf{e}_j{}'\|_F^2 + \|\mathbf{R}\mathbf{e}_i\mathbf{e}_j{}'\mathbf{P}_V\|_F^2 - \|\mathbf{P}_U\mathbf{R}\mathbf{e}_i\mathbf{e}_j{}'\|_F^2\|\mathbf{P}_V\mathbf{e}_j\|_F^2 \\ &\leq c(1+\delta_1(\mathbf{R}))\gamma_R^2(\mathbf{U}) + c(1+\delta_1(\mathbf{R}))\gamma^2(\mathbf{V}) = \Lambda^2\end{aligned} \tag{7.27}$$

where (a) holds because $\langle \mathbf{P}_U\mathbf{R}\mathbf{e}_i\mathbf{e}_j{}'\mathbf{P}_V, \mathbf{R}\mathbf{e}_i\mathbf{e}_j{}'\rangle = \langle \mathbf{e}_i'\mathbf{R}\mathbf{P}_U\mathbf{R}\mathbf{e}_i, \mathbf{e}_j'\mathbf{P}_V\mathbf{e}_j\rangle$ and $\mathbf{P}_U = \mathbf{P}_U^2$ (likewise $\mathbf{P}_V$).

Defining the random variable $\Xi := \pi^{-1}\|\mathcal{P}_\Phi\mathbf{R}\mathcal{P}_\Omega\mathbf{R}'\mathcal{P}_\Phi - \pi\mathcal{P}_\Phi\|$ and using (7.26), one can write

$$\begin{aligned}\Xi &= \pi^{-1}\sup_{\|\mathbf{X}\|_F=1}\Big\|\sum_{i,j}(b_{i,j}-\pi)\langle \mathbf{X}, \mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_i\mathbf{e}_j')\rangle \mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_i\mathbf{e}_j')\Big\|_F \\ &= \pi^{-1}\sup_{\|\mathrm{vec}(\mathbf{X})\|=1}\Big\|\sum_{i,j}(b_{i,j}-\pi)\mathrm{vec}(\mathbf{X})'\mathrm{vec}[\mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_i\mathbf{e}_j')]\otimes \mathrm{vec}[\mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_i\mathbf{e}_j')]\Big\| \\ &= \pi^{-1}\Big\|\sum_{i,j}(b_{i,j}-\pi)\mathrm{vec}[\mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_i\mathbf{e}_j')]\otimes \mathrm{vec}[\mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_i\mathbf{e}_j')]\Big\|.\end{aligned} \tag{7.28}$$

Random variables $\{b_{i,j}-\pi\}$ are i.i.d. with zero mean, and thus one can utilize the spectral concentration inequality in [121, Lemma 3.5] to find

$$\mathbb{E}[\Xi] \leq C\sqrt{\frac{\log(LF)}{\pi}}\max_{i,j}\|\mathcal{P}_\Phi(\mathbf{R}\mathbf{e}_i\mathbf{e}_j')\|_F \overset{(b)}{\leq} C\sqrt{\frac{\log(LF)}{\pi}}\Lambda \tag{7.29}$$

for some constant $C>0$, where (b) is due to (7.27). Now, applying Talagrand's concentration tail bound [136] to the random variable Ξ yields

$$\Pr(|\Xi - \mathbb{E}[\Xi]| \geq t) \leq 3\exp\left(-\frac{t\log(2)}{K}\pi\min\{1,t\}\right) \tag{7.30}$$

for some constant $K > 0$, where $t := \tau\Lambda \log(n)$ and $n := \max\{L, F\}$. The arguments leading to (7.29) and (7.30) are similar those used in [27, Theorem 4.2] for the matrix completion problem, and details are omitted here. Putting (7.29) and (7.30) together it is possible to infer

$$\Xi \leq \mathbb{E}[\Xi] + t \leq C\sqrt{\frac{\log(LF)}{\pi}} + \tau\Lambda \log(n) \tag{7.31}$$

with probability higher than $1 - \mathcal{O}(n^{-C\pi\Lambda\tau})$, which completes the proof of the lemma. ■

Proof of the Main Result

In what follows, conditions are first derived under which the pair $(\mathbf{X}_0, \mathbf{A}_0)$ is the *unique* optimal solution of (P2). The sought conditions pertain to existence of certain dual certificates, which are then constructed in the later section.

Unique optimality conditions

Recall the *nonsmooth* optimization problem (P2), and its Lagrangian formed as

$$\begin{aligned}\mathcal{L}(\mathbf{X}, \mathbf{A}; \mathbf{M}_y, \mathbf{M}_z) =& \|\mathbf{X}\|_* + \lambda\|\mathbf{A}\|_1 + \langle \mathbf{M}_y, \mathbf{Y} - \mathbf{R}(\mathbf{X} + \mathbf{A})\rangle \\ &+ \langle \mathbf{M}_z, \mathbf{Z}_\Pi - \mathcal{P}_\Pi(\mathbf{X} + \mathbf{A})\rangle \end{aligned} \tag{7.32}$$

where $\mathbf{M}_y \in \mathbb{R}^{L\times T}$ and $\mathbf{M}_z \in \mathbb{R}^{F\times T}$ are the matrices of dual variables (multipliers) associated with the link and flow level constraints in (P2), respectively. From the characterization of the subdifferential for the nuclear- and the ℓ_1-norm (see e.g., [22]), the subdifferential of the Lagrangian at $(\mathbf{X}_0, \mathbf{A}_0)$ is given by (recall that $\mathbf{X}_0 = \mathbf{U}_0\mathbf{\Sigma}_0\mathbf{V}_0'$)

$$\begin{aligned}\partial_{\mathbf{X}}\mathcal{L}(\mathbf{X}_0, \mathbf{A}_0; \mathbf{M}_y, \mathbf{M}_z) = \Big\{ &\mathbf{U}_0\mathbf{V}_0^{'} + \mathbf{W} - \mathbf{R}'\mathbf{M}_y \\ &-\mathcal{P}_\Pi(\mathbf{M}_z) : \|\mathbf{W}\| \leq 1,\ \mathcal{P}_{\Phi_{X_0}}(\mathbf{W}) = \mathbf{0}_{F\times T}\Big\} \end{aligned} \tag{7.33}$$

$$\begin{aligned}\partial_{\mathbf{A}}\mathcal{L}(\mathbf{X}_0, \mathbf{A}_0; \mathbf{M}_y, \mathbf{M}_z) = \big\{ &\lambda \text{sign}(\mathbf{A}_0) + \lambda\mathbf{F} - \mathbf{R}'\mathbf{M}_y \\ &-\mathcal{P}_\Pi(\mathbf{M}_z) : \|\mathbf{F}\|_\infty \leq 1,\ \mathcal{P}_{\Omega_{A_0}}(\mathbf{F}) = \mathbf{0}_{F\times T}\Big\}. \end{aligned} \tag{7.34}$$

The optimality conditions for (P2) assert that $(\mathbf{X}_0, \mathbf{A}_0)$ is an optimal (not necessarily unique) solution if and only if

$$\mathbf{0}_{F\times T} \in \partial_{\mathbf{A}} \mathcal{L}(\mathbf{X}_0, \mathbf{A}_0; \mathbf{M}_y, \mathbf{M}_z)$$
$$\mathbf{0}_{F\times T} \in \partial_{\mathbf{X}} \mathcal{L}(\mathbf{X}_0, \mathbf{A}_0; \mathbf{M}_y, \mathbf{M}_z).$$

This is tantamount to existence of the dual variables $\{\mathbf{W}, \mathbf{F}, \mathbf{M}_y, \mathbf{M}_z\}$ satisfying: (i) $\|\mathbf{W}\| \leq 1$, $\mathcal{P}_{\Phi_{X_0}}(\mathbf{W}) = \mathbf{0}_{F\times T}$, (ii) $\|\mathbf{F}\|_\infty \leq 1$, $\mathcal{P}_{\Omega_{A_0}}(\mathbf{F}) = \mathbf{0}_{F\times T}$, and (iii) $\lambda \text{sign}(\mathbf{A}_0) + \lambda\mathbf{F} = \mathbf{U}\mathbf{V}' + \mathbf{W} = \mathbf{R}'\mathbf{M}_y - \mathcal{P}_{\Pi}(\mathbf{M}_z)$.

In essence, to eliminate $\mathbf{M}_y, \mathbf{M}_z$, one can alternatively interpret iii) as finding the dual variable $\mathbf{\Gamma} \in \mathcal{N}_R^\perp + \mathcal{N}_\Pi^\perp = (\mathcal{N}_R \cap \mathcal{N}_\Pi)^\perp$ such that $\mathbf{\Gamma} = \lambda \text{sign}(\mathbf{A}_0) + \lambda\mathbf{F} = \mathbf{U}\mathbf{V}' + \mathbf{W}$. Since $\mathbf{W} = \mathcal{P}_{\mathbf{\Phi}_X^\perp}(\mathbf{\Gamma})$ and $\mathbf{F} = \mathcal{P}_{\Omega_{A_0}^\perp}(\mathbf{\Gamma})$, conditions (i) and (ii) can also be simply recast in terms of $\mathbf{\Gamma}$. In general, (i)–(iii) may hold for multiple solution pairs. However, the next lemma asserts that a slight tightening of the optimality conditions (i)–(iii) leads to a *unique* optimal solution for (P2). The proof goes along the lines of [97, Lemma 2], and it is omitted here for conciseness.

Proposition 7.1 *If $(\mathbf{X}_0, \mathbf{A}_0)$ is locally identifiable from (c1) and (c2), and there exists a dual certificate $\mathbf{\Gamma} \in \mathbb{R}^{F\times T}$ satisfying*

C1) $\mathcal{P}_{\Phi_{X_0}}(\mathbf{\Gamma}) = \mathbf{U}_0\mathbf{V}_0'$

C2) $\mathcal{P}_{\Omega_{A_0}}(\mathbf{\Gamma}) = \lambda \text{sgn}(\mathbf{A}_0)$

C3) $\mathcal{P}_{\mathcal{N}_R \cap \mathcal{N}_\Pi}(\mathbf{\Gamma}) = \mathbf{0}$

C4) $\|\mathcal{P}_{\mathbf{\Phi}_X^\perp}(\mathbf{\Gamma})\| < 1$

C5) $\|\mathcal{P}_{\mathbf{\Omega}_A^\perp}(\mathbf{\Gamma})\|_\infty < \lambda$

then $(\mathbf{X}_0, \mathbf{A}_0)$ is the unique optimal solution to (P2).

The rest of the proof deals with construction of a valid dual certificate $\mathbf{\Gamma}$ that simultaneously meets C1–C5.

One should note that condition (iii) is a distinct feature of the recovery task pursued in this paper. In a similar context, in the robust PCP problem studied in [33], $\mathcal{N}_R = \emptyset, \mathcal{N}_\Pi = \emptyset$,

and thus C3 does not appear anymore. Likewise, the low-rank plus compressed sparse recovery task studied in [97] does not involve the intersection of subspaces as appearing in C3.

Dual certificate construction

The main steps of the construction are inspired by [33] which studies decomposition of low-rank plus sparse matrices, that is either $\Pi = \emptyset$ or $\mathbf{R} = \mathbf{I}_F$. However, relative to [33] the problem here brings up several new distinct elements including the null space of compression and sampling operators in C3, which further challenge construction of dual certificates, and demands, in part, a new treatment. In addition, different incoherence measures are introduced here which facilitate satisfiability for random ensembles. The construction involves two steps. In the first step, a candidate dual certificate is selected to fulfil C1–C3, whereas the second step assures the candidate dual certificate satisfies C4–C5 as well under certain technical conditions in terms of the incoherence parameters in Section 3.4.2.

Toward the first step, condition (II) in Theorem 3.1 implies local identifiability of the observation model, namely $\Omega_{A_0} \cap \Phi_{X_0} = \{\mathbf{0}\}$ and $(\Omega_{A_0} \oplus \Phi_{X_0}) \cap (\mathcal{N}_R \cap \mathcal{N}_\Pi) = \{\mathbf{0}\}$, and thus based on a property of direct-sum [58] there *exists* a *unique* certificate $\mathbf{\Gamma} \in \Omega_{A_0} \oplus \Phi_{X_0} \oplus (\mathcal{N}_R \cap \mathcal{N}_\Omega)$ with projections $\mathcal{P}_{\Omega_{A_0}}(\mathbf{\Gamma}) = \lambda \text{sign}(\mathbf{A}_0)$, $\mathcal{P}_{\Phi_{X_0}}(\mathbf{\Gamma}) = \mathbf{U}_0\mathbf{V}_0'$, and $\mathcal{P}_{\mathcal{N}_R \cap \mathcal{N}_\Pi}(\mathbf{\Gamma}) = \mathbf{0}$. This dual certificate can be expressed as

$$\mathbf{\Gamma} = \mathbf{\Gamma}_{\Omega_{A_0}} + \mathbf{\Gamma}_{\Phi_{X_0}} + \mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Pi}$$

with the components $\mathbf{\Gamma}_{\Omega_{A_0}} \in \Omega_{A_0}$, $\mathbf{\Gamma}_{\Phi_{X_0}} \in \Phi_{X_0}$, and $\mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Pi} \in \mathcal{N}_R \cap \mathcal{N}_\Pi$. As will be seen later, it is more convenient to represent $\mathbf{\Gamma}_{\Omega_{A_0}} = \epsilon_{\Omega_{A_0}} + \lambda \text{sign}(\mathbf{A}_0)$ and $\mathbf{\Gamma}_{\Phi_{X_0}} = \epsilon_{\Phi_{X_0}} + \mathbf{U}_0\mathbf{V}_0'$. From C1–C3, for the projection components $\{\epsilon_{\Omega_{A_0}}, \epsilon_{\Phi_{X_0}}, \mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Pi}\}$ it then holds that

$$\epsilon_{\Phi_{X_0}} = -\mathcal{P}_{\Phi_{X_0}}(\epsilon_{\Omega_{A_0}} + \lambda \text{sign}(\mathbf{A}_0) + \mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Pi}) \tag{7.35}$$

$$\epsilon_{\Omega_{A_0}} = -\mathcal{P}_{\Omega_{A_0}}(\epsilon_{\Phi_{X_0}} + \mathbf{U}_0\mathbf{V}_0' + \mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Pi}) \tag{7.36}$$

$$\mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Pi} = -\mathcal{P}_{\mathcal{N}_R \cap \mathcal{N}_\Pi}(\epsilon_{\Phi_{X_0}} + \mathbf{U}_0\mathbf{V}_0' + \epsilon_{\Omega_{A_0}} + \lambda \text{sign}(\mathbf{A}_0)). \tag{7.37}$$

The second step of the proof manages the candidate dual certificate $\mathbf{\Gamma}$ to satisfy C4 and C5 as well. The main idea is to tighten the conditions for local identifiability, and impose

additional conditions on the incoherence measures (c.f. Section 3.4.2) to ensure that C4 and C5 hold true. In this direction, one can begin by bounding

$$\begin{aligned}\|\mathcal{P}_{\mathbf{\Phi}_X^{\perp}}(\mathbf{\Gamma})\| &\leq \|\mathbf{\Gamma}_{\Omega_{A_0}} + \mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Omega}\| \\ &= \|\epsilon_{\Omega_{A_0}} + \lambda \text{sgn}(\mathbf{A}_0) + \mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Omega}\| \\ &\overset{(a)}{\leq} \|\mathcal{P}_{\Omega_{A_0}}(\epsilon_{\Phi_{X_0}} + \mathbf{U}_0\mathbf{V}_0' + \mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Omega})\| \\ &\quad + \lambda\|\text{sgn}(\mathbf{A}_0)\| + \|\mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Omega}\| \end{aligned} \tag{7.38}$$

and

$$\begin{aligned}\|\mathcal{P}_{\mathbf{\Omega}_A^{\perp}}(\mathbf{\Gamma})\|_\infty &\leq \|\mathbf{\Gamma}_{\Phi_{X_0}} + \mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Pi}\|_\infty \\ &= \|\epsilon_{\Phi_{X_0}} + \mathbf{U}_0\mathbf{V}_0^{'} + \mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Pi}\|_\infty \\ &\overset{(b)}{\leq} \|\mathcal{P}_{\Phi_{X_0}}(\epsilon_{\Omega_{A_0}} + \lambda\text{sgn}(\mathbf{A}_0) + \mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Pi})\|_\infty \\ &\quad + \|\mathbf{U}_0\mathbf{V}_0^{'}\|_\infty + \|\mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Pi}\|_\infty \end{aligned} \tag{7.39}$$

where (a) and (b) come from (7.35) and (7.36) after applying the triangle inequality. In order to bound the r.h.s. of (7.38) and (7.39), it is instructive first to recognize that $\|\mathbf{U}_0\mathbf{V}_0^{'}\|_\infty \leq \gamma(\mathbf{U}_0, \mathbf{V}_0)$, and

$$\|\text{sgn}(\mathbf{A}_0)\| \leq (\|\text{sgn}(\mathbf{A}_0)\|_{\infty,\infty}\|\text{sgn}(\mathbf{A}_0)\|_{1,1})^{1/2} = k \tag{7.40}$$

see e.g., [58]. In addition, building on (3.8) and (3.12), the first term in the r.h.s. of (7.38) is bounded as

$$\begin{aligned}&\|\mathcal{P}_{\Omega_{A_0}}(\epsilon_{\Phi_{X_0}} + \mathbf{U}_0\mathbf{V}_0^{'} + \mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Pi})\| \\ &\leq \|\mathcal{P}_{\Omega_{A_0}}\mathcal{P}_{\Phi_{X_0}}(\epsilon_{\Phi_{X_0}} + \mathbf{U}_0\mathbf{V}_0^{'})\| + \|\mathcal{P}_{\Omega_{A_0}}\mathcal{P}_{\mathcal{N}_\Pi}\mathcal{P}_{\mathcal{N}_R}(\mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Pi})\| \\ &\overset{(a)}{\leq} \mu(\Phi_{X_0}, \Omega_{A_0})\left(\|\epsilon_{\Phi_{X_0}}\| + 1\right) + \mu(\mathcal{N}_R, \Omega_{A_0} \cap \mathcal{N}_\Pi)\|\mathbf{\Gamma}_{\mathcal{N}_R \cap \mathcal{N}_\Omega}\| \end{aligned} \tag{7.41}$$

where (a) is due to the fact that $\mathcal{P}_{\Omega_{A_0} \cap \mathcal{N}_\Pi} = \mathcal{P}_{\Omega_{A_0}}\mathcal{P}_{\mathcal{N}_\Pi}$.

Proceeding in a similar manner as for (7.41), upon using (3.11) it follows that

$$\begin{aligned}
&\|\mathcal{P}_{\Phi_{X_0}}(\epsilon_{\Omega_{A_0}} + \lambda\text{sgn}(\mathbf{A}_0) + \mathbf{\Gamma}_{\mathcal{N}_R\cap\mathcal{N}_\Pi})\|_\infty \\
&\leq \gamma(\mathbf{U}_0, \mathbf{V}_0)\ \|\mathcal{P}_{\Phi_{X_0}}(\epsilon_{\Omega_{A_0}} + \lambda\text{sgn}(\mathbf{A}_0) + \mathbf{\Gamma}_{\mathcal{N}_R\cap\mathcal{N}_\Pi})\| \\
&\leq \gamma(\mathbf{U}_0, \mathbf{V}_0)\ \big[\|\mathcal{P}_{\Phi_{X_0}}(\epsilon_{\Omega_{A_0}} + \lambda\text{sgn}(\mathbf{A}_0)\| + \|\mathcal{P}_{\Phi_{X_0}}\mathcal{P}_{\mathcal{N}_\Pi}(\mathbf{\Gamma}_{\mathcal{N}_R\cap\mathcal{N}_\Pi})\|\big] \\
&\leq \gamma(\mathbf{U}_0, \mathbf{V}_0)\big[\mu(\Omega_{A_0}, \Phi_{X_0})\left(\|\epsilon_{\Omega_{A_0}}\| + \lambda k\right) + \mu(\Phi_{X_0}, \mathcal{N}_\Pi)\|\mathbf{\Gamma}_{\mathcal{N}_R\cap\mathcal{N}_\Pi}\|\big]. \qquad (7.42)
\end{aligned}$$

Focusing on (7.42) and (7.41), it is only left to bound $\|\epsilon_{\Omega_{A_0}}\|$, $\|\epsilon_{\Phi_{X_0}}\|$, and $\|\mathbf{\Gamma}_{\mathcal{N}_R\cap\mathcal{N}_\Omega}\|$. To this end, (7.36)-(7.37) are utilized to arrive at

$$\begin{aligned}
\|\epsilon_{\Phi_{X_0}}\| &= \|\mathcal{P}_{\Phi_{X_0}}(\epsilon_{\Omega_{A_0}} + \lambda\text{sign}(\mathbf{A}_0) + \mathbf{\Gamma}_{\mathcal{N}_R\cap\mathcal{N}_\Pi})\| \\
&\leq \mu(\Phi_{X_0}, \Omega_{A_0})\left(\|\epsilon_{\Omega_{A_0}}\| + \lambda k\right) + \mu(\Phi_{X_0}, \mathcal{N}_\Pi)\|\mathbf{\Gamma}_{\mathcal{N}_R\cap\mathcal{N}_\Pi}\| \qquad (7.43)
\end{aligned}$$

$$\begin{aligned}
\|\epsilon_{\Omega_{A_0}}\| &= \|\mathcal{P}_{\Omega_{A_0}}(\epsilon_{\Phi_{X_0}} + \mathbf{U}_0\mathbf{V}_0' + \mathbf{\Gamma}_{\mathcal{N}_R\cap\mathcal{N}_\Pi})\| \\
&\leq \mu(\Phi_{X_0}, \Omega_{A_0})\left(\|\epsilon_{\Phi_{X_0}}\| + 1\right) + \mu(\mathcal{N}_R, \Omega_{A_0})\|\mathbf{\Gamma}_{\mathcal{N}_R\cap\mathcal{N}_\Omega}\| \qquad (7.44)
\end{aligned}$$

and

$$\begin{aligned}
&\|\mathbf{\Gamma}_{\mathcal{N}_R\cap\mathcal{N}_\Omega}\| \\
&= \|\mathcal{P}_{\mathcal{N}_R\cap\mathcal{N}_\Omega}(\epsilon_{\Phi_{X_0}} + \mathbf{U}_0\mathbf{V}_0' + \epsilon_{\Omega_{A_0}} + \lambda\text{sign}(\mathbf{A}_0))\| \\
&\leq \mu(\Phi_{X_0}, \mathcal{N}_\Omega)\big(\|\epsilon_{\Phi_{X_0}}\| + 1\big) + \mu(\mathcal{N}_R, \Omega_{A_0}\cap\mathcal{N}_\Omega)\big(\|\epsilon_{\Omega_{A_0}}\| + \lambda k\big). \qquad (7.45)
\end{aligned}$$

For convenience introduce the notations $\alpha := \mu(\Phi_{X_0}, \Omega_{A_0})$, $\beta := \mu(\mathcal{N}_R, \Omega_{A_0})$, $\xi := \mu(\Phi_{X_0}, \mathcal{N}_\Pi)$, and $\nu := \mu(\mathcal{N}_R, \Omega_{A_0}\cap\mathcal{N}_\Pi)$. Then, after mixing (7.43)–(7.45) and doing some algebra it follows that

$$\|\Gamma_{\mathcal{N}_R\cap\mathcal{N}_\Omega}\| \leq \theta := \frac{\xi + \lambda k\nu + \alpha(\xi + \alpha\nu)(1-\alpha^2)(\alpha + \lambda k)}{1 - \nu\beta - (\xi + \alpha\nu)(1-\alpha^2)(\xi + \alpha\beta)} \qquad (7.46)$$

and

$$\begin{aligned}
\|\epsilon_{\Omega_{A_0}}\| \leq &\alpha + (1-\alpha^2)\alpha^2(\alpha + \lambda k) \\
&+ \big[\beta + \alpha^2(1-\alpha^2)\beta + \alpha\xi(1-\alpha^2)^{-1}\big]\theta \qquad (7.47)
\end{aligned}$$

$$\|\epsilon_{\Phi_{X_0}}\| \leq (1-\alpha^2)\big[\alpha(\alpha+\lambda k)+(\alpha\beta+\xi)\theta\big]. \tag{7.48}$$

At this point, it is important to recognize from (3.12) that $\|\mathbf{\Gamma}_{\mathcal{N}_R\cap\mathcal{N}_\Pi}\|_\infty \leq \tau\|\mathbf{\Gamma}_{\mathcal{N}_R\cap\mathcal{N}_\Pi}\|$.

Now building on (7.46)-(7.48), one can bound the terms in the r.h.s. of (7.38) and (7.39) from above in terms of $\{\alpha,\beta,\xi,\nu,k\}$. Finally, to fulfill C4 and C5, it suffices to confine their corresponding upper bounds to the values 1 and λ, respectively. This imposes the conditions

(a) $\lambda k+\alpha+\alpha(1-\alpha^2)\big[\alpha(\alpha+\lambda k)+(\alpha\beta+\xi)\theta\big]+(1+\nu)\theta<1$

(b) $\gamma+\eta\alpha\lambda k+(\tau+\eta\alpha+\eta\xi)\theta<\lambda$.

The conditions (a) and (b) imply that C1–C5 hold for the dual certificate $\mathbf{\Gamma}$ if there exists a valid $\lambda\in[\lambda_{\min},\lambda_{\max}]$, with $\lambda_{\max}\geq\lambda_{\min}\geq 0$. The resulting condition is then summarized in the assumptions (I) and (II) of Theorem 3.1, and the proof is now complete.

Proof of Proposition 4.1

Recall the cost function of (P3) defined as

$$f(\mathbf{L},\mathbf{Q},\mathbf{A}) := \frac{1}{2}\|\mathcal{P}_\Omega(\mathbf{Y}-\mathbf{L}\mathbf{Q}'-\mathbf{R}\mathbf{A})\|_F^2+\frac{\lambda_*}{2}\left(\|\mathbf{L}\|_F^2+\|\mathbf{Q}\|_F^2\right)+\lambda_1\|\mathbf{A}\|_1. \tag{7.49}$$

The stationary points $\{\bar{\mathbf{L}},\bar{\mathbf{Q}},\bar{\mathbf{A}}\}$ of (P3) are obtained by setting to zero the (sub)gradients, and solving [22]

$$\partial_{\mathbf{A}} f(\bar{\mathbf{L}},\bar{\mathbf{Q}},\bar{\mathbf{A}})=\mathbf{R}'\mathcal{P}_\Omega(\mathbf{Y}-\bar{\mathbf{L}}\bar{\mathbf{Q}}'-\mathbf{R}\bar{\mathbf{A}})-\lambda_1\text{sign}(\bar{\mathbf{A}})=\mathbf{0}_{F\times T} \tag{7.50}$$

$$\nabla_{\mathbf{L}} f(\bar{\mathbf{L}},\bar{\mathbf{Q}},\bar{\mathbf{A}})=\mathcal{P}_\Omega(\mathbf{Y}-\bar{\mathbf{L}}\bar{\mathbf{Q}}'-\mathbf{R}\bar{\mathbf{A}})\bar{\mathbf{Q}}-\lambda_*\bar{\mathbf{L}}=\mathbf{0}_{L\times\rho} \tag{7.51}$$

$$\nabla_{\mathbf{Q}'} f(\bar{\mathbf{L}},\bar{\mathbf{Q}},\bar{\mathbf{A}})=\bar{\mathbf{L}}'\mathcal{P}_\Omega(\mathbf{Y}-\bar{\mathbf{L}}\bar{\mathbf{Q}}'-\mathbf{R}\bar{\mathbf{A}})-\lambda_*\bar{\mathbf{Q}}'=\mathbf{0}_{\rho\times T}. \tag{7.52}$$

Clearly, every stationary point satisfies $\nabla_{\mathbf{L}} f(\bar{\mathbf{L}},\bar{\mathbf{Q}},\bar{\mathbf{A}})\bar{\mathbf{L}}'=\mathbf{0}_{L\times L}$ and $\bar{\mathbf{Q}}\nabla_{\mathbf{Q}'} f(\bar{\mathbf{L}},\bar{\mathbf{Q}},\bar{\mathbf{A}})=\mathbf{0}_{T\times T}$. It follows from the optimality conditions (7.50)-(7.52) that

$$\mathbf{R}'\mathcal{P}_\Omega(\mathbf{Y}-\bar{\mathbf{L}}\bar{\mathbf{Q}}'-\mathbf{R}\bar{\mathbf{A}})=\lambda_1\text{sign}(\bar{\mathbf{A}}) \tag{7.53}$$

$$\text{tr}(\mathcal{P}_\Omega(\mathbf{Y}-\bar{\mathbf{L}}\bar{\mathbf{Q}}'-\mathbf{R}\bar{\mathbf{A}})\bar{\mathbf{Q}}\bar{\mathbf{L}}')=\lambda_*\text{tr}\{\bar{\mathbf{Q}}\bar{\mathbf{Q}}'\}=\lambda_*\text{tr}\{\bar{\mathbf{L}}\bar{\mathbf{L}}'\}. \tag{7.54}$$

Define $\kappa(\mathbf{W}_1, \mathbf{W}_2) := \frac{1}{2}\left\{\mathrm{tr}\{\mathbf{W}_1\} + \mathrm{tr}\{\mathbf{W}_2\}\right\}$, and consider now the following *convex* optimization problem

$$\text{(P5)} \quad \min_{\{\mathbf{X},\mathbf{A},\mathbf{W}_1,\mathbf{W}_2\}} \left[\frac{1}{2}\|\mathcal{P}_\Omega(\mathbf{Y}-\mathbf{X}-\mathbf{R}\mathbf{A})\|_F^2 + \lambda_*\kappa(\mathbf{W}_1,\mathbf{W}_2) + \lambda_1\|\mathbf{A}\|_1\right]$$
$$\text{s. t.} \quad \mathbf{W} := \begin{pmatrix} \mathbf{W}_1 & \mathbf{X} \\ \mathbf{X}' & \mathbf{W}_2 \end{pmatrix} \succeq \mathbf{0} \tag{7.55}$$

which is *equivalent* to (P1). The equivalence can be readily inferred by minimizing (P5) with respect to $\{\mathbf{W}_1, \mathbf{W}_2\}$ first, and taking advantage of the following alternative characterization of the nuclear norm (see e.g., [117])

$$\|\mathbf{X}\|_* = \min_{\{\mathbf{W}_1,\mathbf{W}_2\}} \kappa(\mathbf{W}_1,\mathbf{W}_2), \quad \text{s. t. } \mathbf{W} \succeq \mathbf{0}.$$

In what follows, the optimality conditions for the conic program (P5) are explored. To this end, the Lagrangian is first formed as

$$\mathcal{L}(\mathbf{X},\mathbf{W}_1,\mathbf{W}_2,\mathbf{A},\mathbf{M}) = \frac{1}{2}\|\mathcal{P}_\Omega(\mathbf{Y}-\mathbf{X}-\mathbf{R}\mathbf{A})\|_F^2 + \lambda_*\kappa(\mathbf{W}_1,\mathbf{W}_2) - \langle\mathbf{M},\mathbf{W}\rangle + \lambda_1\|\mathbf{A}\|_1$$

where $\mathbf{M}$ denotes the dual variables associated with the conic constraint (7.55). For notational convenience, partition $\mathbf{M}$ in four blocks $\mathbf{M}_1 := [\mathbf{M}]_{11}$, $\mathbf{M}_2 := [\mathbf{M}]_{12}$, $\mathbf{M}_3 := [\mathbf{M}]_{22}$, and $\mathbf{M}_4 := [\mathbf{M}]_{21}$, in accordance with the block structure of $\mathbf{W}$ in (7.55), where $\mathbf{M}_1$ and $\mathbf{M}_3$ are $L\times L$ and $T\times T$ matrices. The optimal solution to (P5) must: (i) null the (sub)gradients

$$\nabla_{\mathbf{X}}\mathcal{L}(\mathbf{X},\mathbf{W}_1,\mathbf{W}_2,\mathbf{A},\mathbf{M}) = -\mathcal{P}_\Omega(\mathbf{Y}-\mathbf{X}-\mathbf{R}\mathbf{A}) - \mathbf{M}_2 - \mathbf{M}_4{}' \tag{7.56}$$

$$\partial_{\mathbf{A}}\mathcal{L}(\mathbf{X},\mathbf{W}_1,\mathbf{W}_2,\mathbf{A},\mathbf{M}) = -\mathbf{R}'\mathcal{P}_\Omega(\mathbf{Y}-\mathbf{X}-\mathbf{R}\mathbf{A}) - \lambda_1\mathrm{sign}(\mathbf{A}) \tag{7.57}$$

$$\nabla_{\mathbf{W}_1}\mathcal{L}(\mathbf{X},\mathbf{W}_1,\mathbf{W}_2,\mathbf{A},\mathbf{M}) = \frac{\lambda_*}{2}\mathbf{I}_L - \mathbf{M}_1 \tag{7.58}$$

$$\nabla_{\mathbf{W}_2}\mathcal{L}(\mathbf{X},\mathbf{W}_1,\mathbf{W}_2,\mathbf{A},\mathbf{M}) = \frac{\lambda_*}{2}\mathbf{I}_T - \mathbf{M}_3 \tag{7.59}$$

and satisfy (ii) the complementary slackness condition $\langle\mathbf{M},\mathbf{W}\rangle = 0$; (iii) primal feasibility $\mathbf{W} \succeq \mathbf{0}$; and (iv) dual feasibility $\mathbf{M} \succeq \mathbf{0}$.

Recall the stationary point of (P3), and introduce candidate primal variables $\tilde{\mathbf{X}} := \bar{\mathbf{L}}\bar{\mathbf{Q}}'$, $\tilde{\mathbf{A}} := \bar{\mathbf{A}}$, $\tilde{\mathbf{W}}_1 := \bar{\mathbf{L}}\bar{\mathbf{L}}'$ and $\tilde{\mathbf{W}}_2 := \bar{\mathbf{Q}}\bar{\mathbf{Q}}'$; and the dual variables $\tilde{\mathbf{M}}_1 := \frac{\lambda_*}{2}\mathbf{I}_L$, $\tilde{\mathbf{M}}_3 := \frac{\lambda_*}{2}\mathbf{I}_T$, $\tilde{\mathbf{M}}_2 := -(1/2)\mathcal{P}_\Omega(\mathbf{Y}-\bar{\mathbf{L}}\bar{\mathbf{Q}}'-\mathbf{R}\bar{\mathbf{A}})$, and $\tilde{\mathbf{M}}_4 := \tilde{\mathbf{M}}_2'$. Then, (i) holds since after

plugging the candidate primal and dual variables in (7.56)-(7.59), the subgradients vanish. Moreover, (ii) holds since

$$\begin{aligned}\langle\tilde{\mathbf{M}},\tilde{\mathbf{W}}\rangle &= \langle\tilde{\mathbf{M}}_1,\tilde{\mathbf{W}}_1\rangle+\langle\tilde{\mathbf{M}}_2,\tilde{\mathbf{X}}\rangle+\langle\tilde{\mathbf{M}}_2',\tilde{\mathbf{X}}'\rangle+\langle\tilde{\mathbf{M}}_3,\tilde{\mathbf{W}}_2\rangle\\ =&\frac{\lambda_*}{2}\langle\mathbf{I}_L,\bar{\mathbf{L}}\bar{\mathbf{L}}'\rangle+\frac{\lambda_*}{2}\langle\mathbf{L}_T,\bar{\mathbf{Q}}\bar{\mathbf{Q}}'\rangle-\langle\mathcal{P}_\Omega(\mathbf{Y}-\bar{\mathbf{L}}\bar{\mathbf{Q}}'-\mathbf{R}\bar{\mathbf{A}}),\bar{\mathbf{L}}\bar{\mathbf{Q}}'\rangle\\ =&\frac{\lambda_*}{2}\|\bar{\mathbf{L}}\|_F^2+\frac{\lambda_*}{2}\|\bar{\mathbf{Q}}\|_F^2-\lambda_*\|\bar{\mathbf{L}}\|_F^2=0\end{aligned}$$

where the last two equalities follow from (7.54). Condition (iii) is also met since

$$\begin{pmatrix}\bar{\mathbf{L}}\bar{\mathbf{L}}' & \bar{\mathbf{L}}\bar{\mathbf{Q}}'\\ \bar{\mathbf{Q}}\bar{\mathbf{L}}' & \bar{\mathbf{Q}}\bar{\mathbf{Q}}'\end{pmatrix}=\begin{pmatrix}\bar{\mathbf{L}}\\ \bar{\mathbf{Q}}\end{pmatrix}\begin{pmatrix}\bar{\mathbf{L}}\\ \bar{\mathbf{Q}}\end{pmatrix}'\succeq\mathbf{0}. \tag{7.60}$$

To satisfy (iv), based on a Schur complement argument [58] it suffices to enforce $\sigma_{\max}(\tilde{\mathbf{M}}_2)\leq\lambda_*/2$. ■

Derivation of Algorithm 5

It is shown here that [S1]-[S4] in Section 4.3.3 give rise to the set of recursions tabulated under Algorithm 5. To this end, recall the augmented Lagrangian function in (4.6) and focus first on [S4]. From the decomposable structure of $\mathcal{L}_c$, (4.14) decouples into simpler strictly convex sub-problems for $n\in\mathcal{N}$ and $m\in\mathcal{J}_n$, namely

$$\mathbf{B}_n[k+1]=\arg\min_{\mathbf{B}_n}\left\{r_n(\mathbf{L}_n[k+1],\mathbf{Q}_n[k+1],\mathbf{B}_n)+\langle\mathbf{M}_n[k],\mathbf{B}_n\rangle+\frac{c}{2}\|\mathbf{B}_n-\mathbf{A}_n[k+1]\|_F^2\right\} \tag{7.61}$$

$$\bar{\mathbf{F}}_n^m[k+1]=\arg\min_{\bar{\mathbf{F}}_n^m}\left\{\frac{c}{2}\big(\|\mathbf{Q}_n[k+1]-\bar{\mathbf{F}}_n^m\|_F^2+\|\mathbf{Q}_m[k+1]-\bar{\mathbf{F}}_n^m\|_F^2\big)-\langle\bar{\mathbf{C}}_n^m[k]+\tilde{\mathbf{C}}_n^m[k],\bar{\mathbf{F}}_n^m\rangle\right\} \tag{7.62}$$

$$\bar{\mathbf{G}}_n^m[k+1]=\arg\min_{\bar{\mathbf{G}}_n^m}\left\{\frac{c}{2}\big(\|\mathbf{A}_n[k+1]-\bar{\mathbf{G}}_n^m\|_F^2+\|\mathbf{A}_m[k+1]-\bar{\mathbf{G}}_n^m\|_F^2\big)-\langle\bar{\mathbf{D}}_n^m[k]+\tilde{\mathbf{D}}_n^m[k],\bar{\mathbf{G}}_n^m\rangle\right\}. \tag{7.63}$$

Note that in formulating (7.62) and (7.63), the auxiliary variables $\tilde{\mathbf{F}}_n^m$ and $\tilde{\mathbf{G}}_n^m$ were eliminated using the constraints $\bar{\mathbf{F}}_n^m=\tilde{\mathbf{F}}_n^m$ and $\bar{\mathbf{G}}_n^m=\tilde{\mathbf{G}}_n^m$, respectively. The unconstrained

quadratic problems (7.62) and (7.63) admit the closed-form solutions

$$\bar{\mathbf{F}}_n^m[k+1] = \tilde{\mathbf{F}}_n^m[k+1] = \frac{1}{2c}(\bar{\mathbf{C}}_n^m[k] + \tilde{\mathbf{C}}_n^m[k]) + \frac{1}{2}\left(\mathbf{Q}_n[k+1] + \mathbf{Q}_m[k+1]\right) \tag{7.64}$$

$$\bar{\mathbf{G}}_n^m[k+1] = \tilde{\mathbf{G}}_n^m[k+1] = \frac{1}{2c}(\bar{\mathbf{D}}_n^m[k] + \tilde{\mathbf{D}}_n^m[k]) + \frac{1}{2}\left(\mathbf{A}_n[k+1] + \mathbf{A}_m[k+1]\right). \tag{7.65}$$

Using (7.64) to eliminate $\bar{\mathbf{F}}_n^m[k]$ and $\tilde{\mathbf{F}}_n^m[k]$ from (4.8) and (4.9) respectively, a simple induction argument establishes that if the initial Lagrange multipliers obey $\bar{\mathbf{C}}_n^m[0] = -\tilde{\mathbf{C}}_n^m[0] = \mathbf{0}_{T\times\rho}$, then $\bar{\mathbf{C}}_n^m[k] = -\tilde{\mathbf{C}}_n^m[k]$ for all $k \geq 0$, where $n \in \mathcal{N}$ and $m \in \mathcal{J}_n$. Likewise, the same holds true for $\bar{\mathbf{D}}_n^m[k]$ and $\tilde{\mathbf{D}}_n^m[k]$. The collection of multipliers $\{\tilde{\mathbf{C}}_n^m[k], \tilde{\mathbf{D}}_n^m[k]\}_{n\in\mathcal{N}}^{m\in\mathcal{J}_n}$ is thus redundant, and (7.64)-(7.65) simplify to

$$\bar{\mathbf{F}}_n^m[k+1] = \tilde{\mathbf{F}}_n^m[k+1] = \frac{1}{2}\left(\mathbf{Q}_n[k+1] + \mathbf{Q}_m[k+1]\right), \;\; n \in \mathcal{N}, \;\; m \in \mathcal{J}_n \tag{7.66}$$

$$\bar{\mathbf{G}}_n^m[k+1] = \tilde{\mathbf{G}}_n^m[k+1] = \frac{1}{2}\left(\mathbf{A}_n[k+1] + \mathbf{A}_m[k+1]\right), \;\; n \in \mathcal{N}, \;\; m \in \mathcal{J}_n. \tag{7.67}$$

Observe that $\bar{\mathbf{F}}_n^m[k] = \bar{\mathbf{F}}_m^n[k]$ and $\bar{\mathbf{G}}_n^m[k] = \bar{\mathbf{G}}_m^n[k]$ for all $k \geq 0$, identities that will be used later on. By plugging (7.66) and (7.67) into (4.8) and (4.10) respectively, the non-redundant multiplier updates become

$$\bar{\mathbf{C}}_n^m[k] = \bar{\mathbf{C}}_n^m[k-1] + \frac{\mu}{2}\left(\mathbf{Q}_n[k] - \mathbf{Q}_m[k]\right), \;\; n \in \mathcal{N}, \; m \in \mathcal{J}_n \tag{7.68}$$

$$\bar{\mathbf{D}}_n^m[k] = \bar{\mathbf{D}}_n^m[k-1] + \frac{\mu}{2}\left(\mathbf{A}_n[k] - \mathbf{A}_m[k]\right), \;\; n \in \mathcal{N}, \; m \in \mathcal{J}_n. \tag{7.69}$$

If $\bar{\mathbf{C}}_n^m[0] = -\bar{\mathbf{C}}_m^n[0] = \mathbf{0}_{T\times\rho}$, then the structure of (7.68) reveals that $\bar{\mathbf{C}}_n^m[k] = -\bar{\mathbf{C}}_m^n[k]$ for all $k \geq 0$, where $n \in \mathcal{N}$ and $m \in \mathcal{J}_n$. Clearly, the same holds true for $\bar{\mathbf{D}}_n^m[k]$, and these identities will become handy in the sequel.

Moving on to [S3], (4.13) decouples into $|\mathcal{N}|$ unconstrained quadratic sub-problems

$$\mathbf{L}_n[k+1] = \arg\min_{\mathbf{L}_n}\left\{r_n(\mathbf{L}_n, \mathbf{Q}_n[k+1], \mathbf{B}_n[k]) + \frac{\lambda_*}{2}\|\mathbf{L}_n\|_F^2\right\}.$$

The minimization (4.12) in [S2] also decomposes into simpler sub-problems, both across agents and across the variables $\{\mathbf{Q}_n\}_{n\in\mathcal{N}}$ and $\{\mathbf{A}_n\}_{n\in\mathcal{N}}$, which are decoupled in the augmented Lagrangian when all other variables are fixed. Specifically, the per agent updates

of $\mathbf{Q}_n$ are given by

$$\mathbf{Q}_n[k+1] = \arg\min_{\mathbf{Q}_n} \Big\{ r_n(\mathbf{L}_n[k], \mathbf{Q}_n, \mathbf{B}_n[k]) + \frac{\lambda_*}{2N}\|\mathbf{Q}_n\|_F^2 + \sum_{m\in\mathcal{J}_n} \langle \bar{\mathbf{C}}_n^m[k] + \tilde{\mathbf{C}}_m^n[k], \mathbf{Q}_n\rangle + \frac{c}{2}\sum_{m\in\mathcal{J}_n}\Big(\|\mathbf{Q}_n - \bar{\mathbf{F}}_n^m[k]\|_F^2 + \|\mathbf{Q}_n - \tilde{\mathbf{F}}_m^n[k]\|_F^2\Big)\Big\} \tag{7.70}$$

where the corresponding update in the Algorithm 5 was obtained after using: i) $\bar{\mathbf{C}}_n^m[k] = \tilde{\mathbf{C}}_m^n[k]$ which follows from the identities $\bar{\mathbf{C}}_n^m[k] = -\tilde{\mathbf{C}}_n^m[k]$ and $\bar{\mathbf{C}}_n^m[k] = -\bar{\mathbf{C}}_m^n[k]$ established earlier; ii) the definition $\mathbf{O}_n(k) := 2\sum_{m\in\mathcal{J}_n}\bar{\mathbf{C}}_n^m[k]$; and iii) the identity $\bar{\mathbf{F}}_n^m[k] = \tilde{\mathbf{F}}_m^n[k]$, which allows to merge the identical quadratic penalty terms and eliminate both $\bar{\mathbf{F}}_n^m[k]$ and $\tilde{\mathbf{F}}_m^n[k]$ using (7.66).

Upon defining $\mathbf{P}_n(k) := 2\sum_{m\in\mathcal{J}_n}\bar{\mathbf{D}}_n^m[k]$ and following similar steps as the ones that led to (7.70), one arrives at

$$\mathbf{A}_n[k+1] = \arg\min_{\mathbf{A}_n} \Big\{ \frac{\lambda_1}{N}\|\mathbf{A}_n\|_1 - \langle \mathbf{M}_n[k], \mathbf{A}_n\rangle + \langle \mathbf{P}_n[k], \mathbf{A}_n\rangle + \frac{c}{2}\|\mathbf{B}_n[k] - \mathbf{A}_n\|_F^2 + c\sum_{m\in\mathcal{J}_n}\Big\|\mathbf{A}_n - \frac{\mathbf{A}_n[k] + \mathbf{A}_m[k]}{2}\Big\|_F^2\Big\}$$

This problem now is a separable instance of the Lasso (also related to the proximal operator of the ℓ_1-norm); hence, its solution is expressible in terms of the soft-thresholding operator as in Algorithm 5.

Proof of Proposition

4.2 Let $\bar{\mathbf{Q}}_n := \lim_{k\to\infty}\mathbf{Q}_n[k]$, and likewise for all other convergent sequences in Algorithm 5. Examination of the recursion for $\mathbf{O}_n[k]$ in the limit as $k\to\infty$, reveals that $\sum_{m\in\mathcal{J}_n}[\bar{\mathbf{Q}}_n - \bar{\mathbf{Q}}_m] = \mathbf{0}_{T\times\rho},\ \forall\, n\in\mathcal{N}$. Upon vectorizing the matrix quantities involved, this system of equations implies that the supervector $\bar{\mathbf{q}} := [\mathrm{vec}[\bar{\mathbf{Q}}_1]', \ldots, \mathrm{vec}[\bar{\mathbf{Q}}_N]']'$ belongs to the nullspace of $\mathbf{L}\otimes\mathbf{I}_{T\rho}$, where $\mathbf{L}$ is the Laplacian of the network graph $G(\mathcal{N},\mathcal{L})$. Under (a1), this guarantees that $\bar{\mathbf{Q}}_1 = \bar{\mathbf{Q}}_2 = \ldots = \bar{\mathbf{Q}}_N$. From the analysis of the limiting behavior of $\mathbf{P}_n[k]$, the same argument leads to $\bar{\mathbf{A}}_1 = \bar{\mathbf{A}}_2 = \ldots = \bar{\mathbf{A}}_N$, which establishes the consensus results in the statement of Proposition 4.2. Hence, one can go ahead and define $\bar{\mathbf{Q}} := \bar{\mathbf{Q}}_n$ and

$\bar{\mathbf{A}} := \bar{\mathbf{A}}_n$. Before moving on, note that convergence of $\mathbf{M}_n[k]$ implies that $\bar{\mathbf{B}}_n = \bar{\mathbf{A}}_n = \bar{\mathbf{A}}$, $n \in \mathcal{N}$. These observations guarantee that the limiting solution is feasible for (P4).

To prove the optimality claim it suffices to show that upon convergence, the fixed point $\{\bar{\mathbf{L}}, \bar{\mathbf{Q}}, \bar{\mathbf{A}}, \bar{\mathbf{B}}\}$ of the iterations comprising Algorithm 5 satisfies the Karush-Kuhn-Tucker (KKT) optimality conditions for (P4). Proposition 4.1 asserts that if $\|\mathcal{P}_\Omega(\mathbf{Y} - \mathbf{L}\mathbf{Q}' - \mathbf{R}\mathbf{A})\| \leq \lambda_*$, $\{\bar{\mathbf{L}}, \bar{\mathbf{Q}}, \bar{\mathbf{A}}\}$ is indeed an optimal solution to (P1). To this end, consider the updates of the primal variables in Algorithm 5, which satisfy

$$\nabla_{\mathbf{Q}_n} r_n(\mathbf{Q}_n[k+1], \mathbf{L}_n[k], \mathbf{B}_n[k]) + \frac{\lambda_*}{N}\mathbf{Q}_n[k+1] + \mathbf{O}_n[k+1] + 2c \sum_{m \in \mathcal{J}_n} \left(\mathbf{Q}_n[k+1] - \frac{\mathbf{Q}_n[k] + \mathbf{Q}_m[k]}{2} \right) = \mathbf{0}_{T \times \rho} \tag{7.71}$$

$$\nabla_{\mathbf{L}_n} r_n(\mathbf{Q}_n[k+1], \mathbf{L}_n[k+1], \mathbf{B}_n[k]) + \lambda_* \mathbf{L}_n[k+1] = \mathbf{0}_{L \times \rho} \tag{7.72}$$

$$\nabla_{\mathbf{B}_n} r_n(\mathbf{Q}_n[k+1], \mathbf{L}_n[k+1], \mathbf{B}_n[k+1]) + \mathbf{M}_n[k] + c(\mathbf{B}_n[k+1] - \mathbf{A}_n[k+1]) = \mathbf{0}_{F \times T}. \tag{7.73}$$

Taking the limit from both sides of (7.71)–(7.73), and summing up over all $n \in \mathcal{N}$ yields

$$\nabla_{\mathbf{Q}} r(\bar{\mathbf{Q}}, \bar{\mathbf{L}}, \bar{\mathbf{A}}) + \lambda_* \bar{\mathbf{Q}} = \mathbf{0}_{T \times \rho} \tag{7.74}$$

$$\nabla_{\mathbf{L}} r(\bar{\mathbf{Q}}, \bar{\mathbf{L}}, \bar{\mathbf{A}}) + \lambda_* \bar{\mathbf{L}} = \mathbf{0}_{L \times \rho} \tag{7.75}$$

$$\nabla_{\mathbf{B}} r(\bar{\mathbf{Q}}, \bar{\mathbf{L}}, \bar{\mathbf{A}}) + \sum_{n \in \mathcal{N}} \bar{\mathbf{M}}_n = \mathbf{0}_{F \times T} \tag{7.76}$$

where $r(\mathbf{L}, \mathbf{Q}, \mathbf{B}) := \frac{1}{2}\|\mathcal{P}_\Omega(\mathbf{Y} - \mathbf{L}\mathbf{Q}' - \mathbf{R}\mathbf{B})\|_F^2$. To arrive at (7.74), the assumption that $\bar{\mathbf{C}}_n^m[1] = \mathbf{0}$, $\forall m \in \mathcal{J}_n, n \in \mathcal{N}$ is used, and thus $\bar{\mathbf{C}}_n^m[k] = -\bar{\mathbf{C}}_m^n[k]$ which leads to $\sum_{n \in \mathcal{N}} \mathbf{O}_n[k] = \sum_{n \in \mathcal{N}} \sum_{m \in \mathcal{J}_n} \bar{\mathbf{C}}_n^m[k] = \mathbf{0}$.

Next, consider the auxiliary matrices $\mathbf{\Theta}_n := \bar{\mathbf{M}}_n - \bar{\mathbf{P}}_n + c(1 + 2|\mathcal{J}_n|)\bar{\mathbf{A}}$, $n \in \mathcal{N}$. In the limit as $k \to \infty$, the update recursion for $\mathbf{A}_n[k+1]$ in Algorithm 5 can be written as $c(1+2|\mathcal{J}_n|)\bar{\mathbf{A}} = \mathcal{S}\left(\mathbf{\Theta}_n, \lambda_1/N\right)$. Proceed by defining $\mathbf{\Psi}_n := \mathbf{\Theta}_n - c(1+2|\mathcal{J}_n|)\bar{\mathbf{A}}$, and observe that the input-output relationship of the soft-thresholding operator $\mathcal{S}$ yields

$$[\mathbf{\Psi}_n]_{f,t} = \begin{cases} \lambda_1/N, & [\bar{\mathbf{A}}]_{f,t} > 0, \\ -\lambda_1/N, & [\bar{\mathbf{A}}]_{f,t} < 0, \\ \xi_{f,t}^{(n)} : |\xi_{f,t}^{(n)}| \leq \lambda_1/N, & [\bar{\mathbf{A}}]_{f,t} = 0. \end{cases} \tag{7.77}$$

Given (7.77), define $\mathbf{\Gamma}_1 := \frac{1}{2}\left(\lambda_1 \mathbf{1}_F \mathbf{1}_T' + \sum_{n=1}^N \mathbf{\Psi}_n\right)$ and $\mathbf{\Gamma}_2 := \frac{1}{2}\left(\lambda_1 \mathbf{1}_F \mathbf{1}_T' - \sum_{n=1}^N \mathbf{\Psi}_n\right)$, and show that they satisfy the following properties: (i) $\mathbf{\Gamma}_1, \mathbf{\Gamma}_2 \geq \mathbf{0}$ (entrywise); (ii) $[\mathbf{\Gamma}_1]_{f,t} = 0$, if $[\bar{\mathbf{A}}]_{f,t} < 0$; (iii) $[\mathbf{\Gamma}_2]_{f,t} = 0$, if $[\bar{\mathbf{A}}]_{f,t} > 0$; (iv) $\mathbf{\Gamma}_1 + \mathbf{\Gamma}_2 = \lambda_1 \mathbf{1}_F \mathbf{1}_T'$; and (v) $\mathbf{\Gamma}_1 - \mathbf{\Gamma}_2 = \sum_{n\in\mathcal{N}} \bar{\mathbf{M}}_n$. Properties (i)-(iii) follow after adding up the result in (7.77) for $n = 1, 2, \ldots, N$. Property (iv) is readily checked from the definitions of $\mathbf{\Gamma}_1$ and $\mathbf{\Gamma}_2$. Finally, (v) follows since

$$\mathbf{\Gamma}_1 - \mathbf{\Gamma}_2 = \sum_{n=1}^N \mathbf{\Psi}_n = \sum_{n=1}^N \left(_n - c(1 + 2|\mathcal{J}_n|)\bar{\mathbf{A}}\right) = \sum_{n=1}^N \bar{\mathbf{M}}_n - \sum_{n=1}^N \bar{\mathbf{P}}_n = \sum_{n=1}^N \bar{\mathbf{M}}_n \tag{7.78}$$

where $\sum_{n=1}^N \bar{\mathbf{P}}_n = \mathbf{0}$ (from the identity $\sum_{n=1}^N \mathbf{P}_n[k] = \mathbf{0}$) is used to obtain the last equality.

The proof is concluded by noticing that properties (i)-(v) along with (7.74)-(7.76) comprise the KKT conditions for the following optimization problem

$$\begin{aligned} \min_{\{\mathbf{L},\mathbf{Q},\mathbf{A},\mathbf{T}\}} \; & r(\mathbf{L},\mathbf{Q},\mathbf{A}) + \frac{\lambda_*}{2}\left\{\|\mathbf{L}\|_F^2 + \|\mathbf{Q}\|_F^2\right\} + \lambda_1 \mathbf{1}_F' \mathbf{T} \mathbf{1}_T \\ \text{s. to } & -\mathbf{T} \leq \mathbf{A} \leq \mathbf{T} \quad \text{(entrywise)} \end{aligned}$$

where $\{\bar{\mathbf{L}}, \bar{\mathbf{Q}}, \bar{\mathbf{A}}\}$ and $\{\mathbf{\Gamma}_1, \mathbf{\Gamma}_2\}$ play the role of the optimal primal and dual variables, respectively. This last problem is clearly equivalent to (P4). ■

Update of the anomaly map in Algorithm 9

As argued in Section 5.3.2, the matrix Lasso problem under [S1] decomposes over the columns of $\mathbf{A} := [\mathbf{a}_1, \ldots, \mathbf{a}_T]$. Hence, it suffices to focus on the update of a single column, say $\mathbf{a}_t := [a_{1,t}, \ldots, a_{F,t}]'$, which boils down to solving [cf. (5.6)]

$$\mathbf{a}_t[k+1] = \arg\min_{\mathbf{a}} \left[\frac{1}{2}\left\|\mathbf{\Omega}_t\left(\mathbf{y}_t - \mathbf{L}[k]\mathbf{q}_t[k] - \sum_{f=1}^F \mathbf{r}_f a_{f,t}\right)\right\|_2^2 + \lambda_1 \sum_{f=1}^F |a_{f,t}|\right] \tag{7.79}$$

where $\mathbf{r}_f$ denotes the f-th column of $\mathbf{R}$.

Let $n = 0, 1, \ldots,$ denote the (inner) iteration index for the cyclic coordinate descent algorithm adopted to solve (7.79) [56, p. 92]. For the minimization at step k of the (outer) BCD iterations in Algorithm 9, the sequence of iterates $\mathbf{a}_t[k; n]$ are initialized as $\mathbf{a}_t[k; 0] := \mathbf{a}_t[k]$. At each step n, the scalar coordinates $a_{f,t}$ of vector $\mathbf{a}_t$ are updated

cyclically, by solving sequentially for $f = 1, 2, \ldots, F$

$$a_{f,t}[k; n+1] = \arg\min_{a} \left[\frac{1}{2} \left\| \tilde{\mathbf{y}}_t^{(-f)}[k; n+1] - \mathbf{\Omega}_t \mathbf{r}_f a \right\|_2^2 + \lambda_1 |a| \right] \tag{7.80}$$

$$\tilde{\mathbf{y}}_t^{(-f)}[k; n+1] := \mathbf{\Omega}_t \left(\mathbf{y}_t - \mathbf{L}[k]\mathbf{q}_t[k] - \sum_{f'=1}^{f-1} \mathbf{r}_{f'} a_{f',t}[k; n+1] - \sum_{f'=f+1}^{F} \mathbf{r}_{f'} a_{f',t}[k; n] \right). \tag{7.81}$$

Vector $\tilde{\mathbf{y}}_t^{(-f)}$ corresponds to the partial residual error without considering the contribution of the predictor $\mathbf{\Omega}_t \mathbf{r}_f$. The usefulness of a coordinate descent approach stems from the fact that the coordinate updates (7.80) amount to scalar Lasso-type optimizations. Skipping details that can be found in, e.g., [56, p. 93], the solutions are thus expressible in the closed form

$$a_{f,t}[k; n+1] = \text{sign}(\mathbf{r}_f' \tilde{\mathbf{y}}_t^{(-f)}[k; n+1]) \left[|\mathbf{r}_f' \tilde{\mathbf{y}}_t^{(-f)}[k; n+1]| - \lambda_1 \right]_+ / \|\mathbf{\Omega}_t \mathbf{r}_f\|_2 \tag{7.82}$$

which is oftentimes referred to as soft-thresholding of the partial residual $\tilde{\mathbf{y}}_t^{(-f)}$. Separability of the nondifferentiable ℓ_1-norm term in (7.79) is sufficient to guarantee the convergence of (7.82) to a minimizer of (7.79), as $n \to \infty$ [144]. Hence, the update $\mathbf{a}_t[k+1] := \lim_{n\to\infty} [a_{1,t}[k; n], \ldots, a_{F,t}[k; n]]'$ is well defined, and identical to the one in (7.79).

The rationale behind the actual anomaly map updates in Algorithm 9 hinges upon the fact that the solution of (7.79) does not need to be super accurate, since it is just an intermediate step in the outer loop defined by the BCD solver. In the relaxation pursued here, the inner iteration is halted after a single step (i.e., when $n = 1$) to yield an inexact minimizer of (7.79). In this case, the index n can be dropped and (7.81)-(7.82) simplify to the sequential updates for $f = 1, 2, \ldots, F$

$$\tilde{\mathbf{y}}_t^{(-f)}[k+1] := \mathbf{\Omega}_t \left(\mathbf{y}_t - \mathbf{L}[k]\mathbf{q}_t[k] - \sum_{f'=1}^{f-1} \mathbf{r}_{f'} a_{f',t}[k+1] - \sum_{f'=f+1}^{F} \mathbf{r}_{f'} a_{f',t}[k] \right)$$

$$a_{f,t}[k+1] = \text{sign}(\mathbf{r}_f' \tilde{\mathbf{y}}_t^{(-f)}[k+1]) \left[|\mathbf{r}_f' \tilde{\mathbf{y}}_t^{(-f)}[k+1]| - \lambda_1 \right]_+ / \|\mathbf{\Omega}_t \mathbf{r}_f\|_2$$

as tabulated under Algorithm 9.

Proof of Lemma 5.1

With $\mathbf{L}_1, \mathbf{L}_2 \in \mathcal{L}$ consider the function

$$u_t(\mathbf{a}, \mathbf{L}_1, \mathbf{L}_2) := \frac{1}{2}\|\mathbf{F}_t(\mathbf{L}_1)(\mathbf{y}_t - \mathbf{R}_t\mathbf{a})\|_2^2 - \frac{1}{2}\|\mathbf{F}_t(\mathbf{L}_2)(\mathbf{y}_t - \mathbf{R}_t\mathbf{a})\|_2^2 \tag{7.83}$$

where $\mathbf{F}_t(\mathbf{L}) := \left[\mathbf{\Omega}_t\left[\mathbf{I}_L - \mathbf{L}\mathbf{D}_t(\mathbf{L})\right]\mathbf{\Omega}_t,\ \sqrt{\lambda_*}\mathbf{\Omega}_t\mathbf{D}_t'(\mathbf{L})\right]'$, and $\mathbf{D}_t(\mathbf{L}) := (\lambda_*\mathbf{I}_\rho + \mathbf{L}'\mathbf{\Omega}_t\mathbf{L})^{-1}\mathbf{L}'$. From the convexity of the Lasso problem in (5.12) together with the mean-value theorem and a5), it can be readily inferred that

$$u_t(\mathbf{a}_t(\mathbf{L}_2), \mathbf{L}_1, \mathbf{L}_2) - u_t(\mathbf{a}_t(\mathbf{L}_1), \mathbf{L}_1, \mathbf{L}_2) \geq c_0\|\mathbf{a}_t(\mathbf{L}_2) - \mathbf{a}_t(\mathbf{L}_1)\|_2^2 \tag{7.84}$$

for some positive constant c_0. The rest of the proof deals with Lipschitz continuity of $u_t(., \mathbf{L}_1, \mathbf{L}_2)$. For $\mathbf{a}_1$ and $\mathbf{a}_2$ from a compact set $\mathcal{A}$, consider

$$\begin{aligned} 2|u_t(\mathbf{a}_1, \mathbf{L}_1, \mathbf{L}_2) - u_t(\mathbf{a}_2, \mathbf{L}_1, \mathbf{L}_2)| &= 2\langle \mathbf{R}_t'\left[\mathbf{F}_t'(\mathbf{L}_2)\mathbf{F}_t(\mathbf{L}_2) - \mathbf{F}_t'(\mathbf{L}_1)\mathbf{F}_t(\mathbf{L}_1)\right], (\mathbf{a}_2 - \mathbf{a}_1)\mathbf{y}_t'\rangle \\ &+ \left(\|\mathbf{F}_t(\mathbf{L}_1)\mathbf{R}_t\mathbf{a}_1\|_2^2 - \|\mathbf{F}_t(\mathbf{L}_1)\mathbf{R}_t\mathbf{a}_2\|_2^2\right) - \left(\|\mathbf{F}_t(\mathbf{L}_2)\mathbf{R}_t\mathbf{a}_1\|_2^2 - \|\mathbf{F}_t(\mathbf{L}_2)\mathbf{R}_t\mathbf{a}_2\|_2^2\right). \end{aligned} \tag{7.85}$$

Introducing the auxiliary variable $\mathbf{\Delta}_a := \mathbf{a}_2 - \mathbf{a}_1$, the last two summands in (7.85) can be bounded as

$$\begin{aligned} &\|\mathbf{F}_t(\mathbf{L}_1)\mathbf{R}_t\mathbf{a}_1\|_2^2 - \|\mathbf{F}_t(\mathbf{L}_1)\mathbf{R}_t\mathbf{a}_2\|_2^2 - \|\mathbf{F}_t(\mathbf{L}_2)\mathbf{R}_t\mathbf{a}_1\|_2^2 + \|\mathbf{F}_t(\mathbf{L}_2)\mathbf{R}_t\mathbf{a}_2\|_2^2 \\ &\quad = \left(\|\mathbf{F}_t(\mathbf{L}_1)\mathbf{R}_t\mathbf{\Delta}_a\|_2^2 - \|\mathbf{F}_t(\mathbf{L}_2)\mathbf{R}_t\mathbf{\Delta}_a\|_2^2\right) + 2\langle \mathbf{R}_t'\left[\mathbf{F}_t'(\mathbf{L}_2)\mathbf{F}_t(\mathbf{L}_2) - \mathbf{F}_t'(\mathbf{L}_1)\mathbf{F}_t(\mathbf{L}_1)\right], \mathbf{a}_2\mathbf{\Delta}_a'\rangle \\ &\quad \leq c_1\|\mathbf{F}_t(\mathbf{L}_2) - \mathbf{F}_t(\mathbf{L}_1)\|\|\mathbf{\Delta}_a\|_2^2 + c_2\|\mathbf{F}_t'(\mathbf{L}_2)\mathbf{F}_t(\mathbf{L}_2) - \mathbf{F}_t'(\mathbf{L}_1)\mathbf{F}_t(\mathbf{L}_1)\|\|\mathbf{\Delta}_a\|_2 \\ &\quad \leq c_3\|\mathbf{F}_t(\mathbf{L}_2) - \mathbf{F}_t(\mathbf{L}_1)\|\|\mathbf{\Delta}_a\|_2 \end{aligned} \tag{7.86}$$

for some constants $c_1, c_2, c_3 > 0$, since $\|\mathbf{F}_t(\mathbf{L})\|$ for $\mathbf{L} \in \mathcal{L}$, $\|\mathbf{\Delta}_a\|_2$, $\|\mathbf{a}_2\|_2$ for $\mathbf{a}_1, \mathbf{a}_2 \in \mathcal{A}$, and $\|\mathbf{R}_t\|$ are all uniformly bounded. The first summand on the right-hand side of (7.85) is similarly bounded (details omitted here). Next, to establish that $\mathbf{F}_t(\mathbf{L})$ is Lipschitz one can derive the following bound ($\mathbf{\Delta}_P := \mathbf{L}_2 - \mathbf{L}_1$)

$$\begin{aligned} \|\mathbf{F}_t(\mathbf{L}_2) - \mathbf{F}_t(\mathbf{L}_1)\| &\leq \|\mathbf{\Omega}_t\left[\mathbf{L}_2\mathbf{D}_t(\mathbf{L}_2) - \mathbf{L}_1\mathbf{D}_t(\mathbf{L}_1)\right]\mathbf{\Omega}_t\| + \sqrt{\lambda_*}\|\mathbf{\Omega}_t(\mathbf{D}_t'(\mathbf{L}_2) - \mathbf{D}_t'(\mathbf{L}_1))\| \\ &\leq \|\mathbf{L}_1\|(\|\mathbf{L}_1\| + \sqrt{\lambda_*})\|(\lambda_*\mathbf{I}_\rho + \mathbf{L}_2'\mathbf{\Omega}_t\mathbf{L}_2)^{-1} - (\lambda_*\mathbf{I}_\rho + \mathbf{L}_1'\mathbf{\Omega}_t\mathbf{L}_1)^{-1}\| \\ &\quad + \|\mathbf{\Delta}_P\|(\|\mathbf{L}_1\| + \|\mathbf{L}_2\| + \sqrt{\lambda_*})\|(\lambda_*\mathbf{I}_\rho + \mathbf{L}_2'\mathbf{\Omega}_t\mathbf{L}_2)^{-1}\|. \end{aligned} \tag{7.87}$$

Define $\mathbf{G}_t := \boldsymbol{\Delta}_P'\boldsymbol{\Omega}_t\mathbf{L}_1 + \boldsymbol{\Delta}_P'\boldsymbol{\Omega}_t\boldsymbol{\Delta}_P + \mathbf{L}_1'\boldsymbol{\Omega}_t\boldsymbol{\Delta}_P$ and $\mathbf{H}_{t,i} := \lambda_*\mathbf{I}_\rho + \mathbf{L}_i\boldsymbol{\Omega}_t\mathbf{L}_i'$, $i = 1, 2$, and consider the following identity

$$\mathbf{H}_{t,1}^{-1} = \big(\mathbf{H}_{t,1} + \mathbf{G}_t\big)^{-1} + \mathbf{H}_{t,1}^{-1}\mathbf{G}_t\big(\mathbf{H}_{t,1} + \mathbf{G}_t\big)^{-1}$$

The first term in the right-hand of (7.87) is then bounded as follows

$$\begin{aligned}\|(\lambda_*\mathbf{I}_\rho + \mathbf{L}_2'\boldsymbol{\Omega}_t\mathbf{L}_2)^{-1} - (\lambda_*\mathbf{I}_\rho + \mathbf{L}_1'\boldsymbol{\Omega}_t\mathbf{L}_1)^{-1}\| &= \|\big(\mathbf{H}_{t,1} + \mathbf{G}_t\big)^{-1} - \mathbf{H}_{t,1}^{-1}\| \\ &\leq \|\mathbf{H}_{t,1}^{-1}\|\|\mathbf{G}_t\|\|\big(\mathbf{H}_{t,1} + \mathbf{G}_t\big)^{-1}\| \leq \left(\frac{1}{\lambda_*}\right)^2\|\mathbf{G}_t\| \leq c_4\|\boldsymbol{\Delta}_P\|. \qquad (7.88)\end{aligned}$$

Putting the pieces together $\mathbf{F}_t(.)$ is found to be Lipschitz and subsequently (7.85) is bounded by a constant factor of $\|\boldsymbol{\Delta}_L\|\|\boldsymbol{\Delta}_a\|_2$. Substituting $\mathbf{a}_1 = \mathbf{a}_t(\mathbf{L}_1)$ and $\mathbf{a}_2 = \mathbf{a}_t(\mathbf{L}_2)$ along with the bound in (7.84) yields the desired result $\|\mathbf{a}_t(\mathbf{L}_2) - \mathbf{a}_t(\mathbf{L}_1)\|_2 \leq c_5\|\mathbf{L}_2 - \mathbf{L}_1\|$. Furthermore, from the relationship $\mathbf{q}_t = \mathbf{D}_t(\mathbf{L})\boldsymbol{\Omega}_t(\mathbf{y}_t - \mathbf{R}_t\mathbf{a}_t)$, Lipschitz continuity of $\mathbf{q}_t(\mathbf{L})$ readily follows.

Moreover, $g_t(\mathbf{L}, \mathbf{q}[t], \mathbf{a}[t])$ is a quadratic function on a compact set, and thus clearly Lipschitz continuous. To prove Lipschitz continuity of $\ell_t(\mathbf{L})$, recall the definition $\{\mathbf{q}_t(\mathbf{L}), \mathbf{a}_t(\mathbf{L})\} = \arg\min_{\{\mathbf{q},\mathbf{a}\}} g_t(\mathbf{L}, \mathbf{q}, \mathbf{a})$ to obtain after some algebra

$$\begin{aligned}\ell_t(\mathbf{L}_2) - \ell_t(\mathbf{L}_1) = \frac{1}{2}\|\mathcal{P}_{\Omega_t}(\mathbf{L}_2\mathbf{q}_t(\mathbf{L}_2) + \mathbf{R}_t\mathbf{a}_t(\mathbf{L}_2))\|_2^2 - \|\mathcal{P}_{\Omega_t}(\mathbf{L}_1\mathbf{q}_t(\mathbf{L}_1) + \mathbf{R}_t\mathbf{a}_t(\mathbf{L}_1))\|_2^2 \\ - \langle\mathcal{P}_{\Omega_t}(\mathbf{y}_t), \mathbf{L}_2\mathbf{q}_t(\mathbf{L}_2) + \mathbf{R}_t\mathbf{a}_t(\mathbf{L}_2) - \mathbf{L}_1\mathbf{q}_t(\mathbf{L}_1) - \mathbf{R}_t\mathbf{a}_t(\mathbf{L}_1)\rangle \\ + \frac{\lambda_*}{2}\big(\|\mathbf{q}_t(\mathbf{L}_2)\|_2^2 - \|\mathbf{q}_t(\mathbf{L}_1)\|_2^2\big) + \lambda_1\big(\|\mathbf{a}_t(\mathbf{L}_2)\|_1 - \|\mathbf{a}_t(\mathbf{L}_1)\|_1\big). \qquad (7.89)\end{aligned}$$

The first term in the right-hand side of (7.89) is bounded as

$$\begin{aligned}&\|\mathcal{P}_{\Omega_t}(\mathbf{L}_2\mathbf{q}_t(\mathbf{L}_2) + \mathbf{R}_t\mathbf{a}_t(\mathbf{L}_2))\|_2^2 - \|\mathcal{P}_{\Omega_t}(\mathbf{L}_1\mathbf{q}_t(\mathbf{L}_1) + \mathbf{R}_t\mathbf{a}_t(\mathbf{L}_1))\|_2^2 \leq \\ &\big(\|\mathcal{P}_{\Omega_t}(\mathbf{L}_2\mathbf{q}_t(\mathbf{L}_2) - \mathbf{L}_1\mathbf{q}_t(\mathbf{L}_1))\|_2 + \|\mathcal{P}_{\Omega_t}(\mathbf{R}_t\mathbf{a}_t(\mathbf{L}_2) - \mathbf{R}_t\mathbf{a}_t(\mathbf{L}_1))\|_2\big) \\ &\qquad \times \big(\|\mathcal{P}_{\Omega_t}(\mathbf{L}_2\mathbf{q}_t(\mathbf{L}_2) + \mathbf{R}_t\mathbf{a}_t(\mathbf{L}_2))\|_2 + \|\mathcal{P}_{\Omega_t}(\mathbf{L}_1\mathbf{q}_t(\mathbf{L}_1) + \mathbf{R}_t\mathbf{a}_t(\mathbf{L}_1))\|_2\big) \\ &\leq c_6\big(\|\mathbf{L}_2 - \mathbf{L}_1\|\|\mathbf{q}_t(\mathbf{L}_2)\|_2 + \|\mathbf{L}_1\|\|\mathbf{q}_t(\mathbf{L}_2) - \mathbf{q}_t(\mathbf{L}_1)\|_2 + \|\mathbf{R}_t\|\|\mathbf{a}_t(\mathbf{L}_2) - \mathbf{a}_t(\mathbf{L}_1)\|_2\big) \quad (7.90)\end{aligned}$$

for some constant $c_6 > 0$. The second one is bounded as

$$\begin{aligned}
&\langle \mathcal{P}_{\Omega_t}(\mathbf{y}_t), \mathbf{L}_2\mathbf{q}_t(\mathbf{L}_2) + \mathbf{R}_t\mathbf{a}_t(\mathbf{L}_2) - \mathbf{L}_1\mathbf{q}_t(\mathbf{L}_1) - \mathbf{R}_t\mathbf{a}_t(\mathbf{L}_1)\rangle \\
&\quad \le \|\mathcal{P}_{\Omega_t}(\mathbf{y}_t)\|_2\big(\|\mathcal{P}_{\Omega_t}(\mathbf{L}_2\mathbf{q}_t(\mathbf{L}_2) - \mathbf{L}_1\mathbf{q}_t(\mathbf{L}_1))\|_2 + \|\mathcal{P}_{\Omega_t}(\mathbf{R}_t\mathbf{a}_t(\mathbf{L}_2) - \mathbf{R}_t\mathbf{a}_t(\mathbf{L}_1))\|_2\big) \\
&\quad \le \|\mathcal{P}_{\Omega_t}(\mathbf{y}_t)\|_2\big(\|\mathbf{L}_2 - \mathbf{L}_1\|\|\mathbf{q}_t(\mathbf{L}_2)\|_2 + \|\mathbf{L}_1\|\|\mathbf{q}_t(\mathbf{L}_2) - \mathbf{q}_t(\mathbf{L}_1)\|_2 + \|\mathbf{R}_t\|\|\mathbf{a}_t(\mathbf{L}_2) - \mathbf{a}_t(\mathbf{L}_1)\|_2\big).
\end{aligned} \tag{7.91}$$

Finally, one can bound the third term in (7.89) as

$$\begin{aligned}
&\frac{\lambda_*}{2}\big(\|\mathbf{q}_t(\mathbf{L}_2)\|_2^2 - \|\mathbf{q}_t(\mathbf{L}_1)\|_2^2\big) + \lambda_1\big(\|\mathbf{a}_t(\mathbf{L}_2)\|_1 - \|\mathbf{a}_t(\mathbf{L}_1)\|_1\big) \le \\
&\qquad \frac{\lambda_*}{2}\|\mathbf{q}_t(\mathbf{L}_2) - \mathbf{q}_t(\mathbf{L}_1)\|_2\big(\|\mathbf{q}_t(\mathbf{L}_2)\|_2 + \|\mathbf{q}_t(\mathbf{L}_1)\|_2\big) + \lambda_1\sqrt{F}\|\mathbf{a}_t(\mathbf{L}_2) - \mathbf{a}_t(\mathbf{L}_1)\|_2.
\end{aligned} \tag{7.92}$$

Since $\mathbf{q}_t(\mathbf{L})$ and $\mathbf{a}_t(\mathbf{L})$ are Lipschitz as proved earlier, and $\mathbf{L}_1, \mathbf{L}_2 \in \mathcal{L}$ are uniformly bounded, the expressions in the right-hand side of (7.90)-(7.92) are upper bounded by a constant factor of $\|\mathbf{L}_2 - \mathbf{L}_1\|$, and so is $|\ell_t(\mathbf{L}_2) - \ell_t(\mathbf{L}_1)|$ after applying the triangle inequality to (7.89).

Regarding $\nabla\ell_t(\mathbf{L})$, notice first that since $\{\mathbf{q}_t(\mathbf{L}), \mathbf{a}_t(\mathbf{L})\}$ is the unique minimizer of $g_t(\mathbf{L}, \mathbf{q}, \mathbf{a})$ [cf. a5)], Danskin's theorem [17, Prop. B.25(a)] implies that $\nabla\ell_t(\mathbf{L}) = \mathcal{P}_{\Omega_t}(\mathbf{y}_t - \mathbf{L}\mathbf{q}_t(\mathbf{L}) - \mathbf{R}_t\mathbf{a}_t(\mathbf{L}))\mathbf{q}_t'(\mathbf{L})$. In the sequel, the triangle inequality will be used to split the norm in the right-hand side of

$$\begin{aligned}
&\big\|\nabla\ell_t(\mathbf{L}_2) - \nabla\ell_t(\mathbf{L}_1)\big\|_F = \big\|\mathcal{P}_{\Omega_t}(\mathbf{y}_t)\left[\mathbf{q}_t(\mathbf{L}_2) - \mathbf{q}_t(\mathbf{L}_1)\right]' - [\mathcal{P}_{\Omega_t}(\mathbf{L}_2\mathbf{q}_t(\mathbf{L}_2))\mathbf{q}_t'(\mathbf{L}_2) - \\
&\mathcal{P}_{\Omega_t}(\mathbf{L}_1\mathbf{q}_t(\mathbf{L}_1))\mathbf{q}_t'(\mathbf{L}_1)] - \left[\mathcal{P}_{\Omega_t}(\mathbf{R}_t\mathbf{a}_t(\mathbf{L}_2))\mathbf{q}_t'(\mathbf{L}_2) - \mathcal{P}_{\Omega_t}(\mathbf{R}_t\mathbf{a}_t(\mathbf{L}_1))\mathbf{q}_t'(\mathbf{L}_1)\right]\big\|_F.
\end{aligned} \tag{7.93}$$

The first term inside the norm is bounded as

$$\|\mathcal{P}_{\Omega_t}(\mathbf{y}_t)\left[\mathbf{q}_t(\mathbf{L}_2) - \mathbf{q}_t(\mathbf{L}_1)\right]'\|_F \le \|\mathcal{P}_{\Omega_t}(\mathbf{y}_t)\|_2\|\mathbf{q}_t(\mathbf{L}_2) - \mathbf{q}_t(\mathbf{L}_1)\|_2. \tag{7.94}$$

After some algebraic manipulations, the second term is also bounded as

$$\begin{aligned}
\|\mathcal{P}_{\Omega_t}(\mathbf{L}_2\mathbf{q}_t(\mathbf{L}_2))\mathbf{q}_t'(\mathbf{L}_2) - \mathcal{P}_{\Omega_t}(\mathbf{L}_1\mathbf{q}_t(\mathbf{L}_1))\mathbf{q}_t'(\mathbf{L}_1)\|_F &\le \|\mathbf{L}_2 - \mathbf{L}_1\|_F\|\mathbf{q}_t(\mathbf{L}_2)\|_2^2 \\
&\quad + \|\mathbf{q}_t(\mathbf{L}_2) - \mathbf{q}_t(\mathbf{L}_1)\|_2\big(\|\mathbf{q}_t(\mathbf{L}_2)\|_2 + \|\mathbf{q}_t(\mathbf{L}_1)\|_2\big)
\end{aligned} \tag{7.95}$$

and finally one can simply bound the third term as

$$\|\mathcal{P}_{\Omega_t}(\mathbf{R}_t\mathbf{a}_t(\mathbf{L}_2))\mathbf{q}_t'(\mathbf{L}_2) - \mathcal{P}_{\Omega_t}(\mathbf{R}_t\mathbf{a}_t(\mathbf{L}_1))\mathbf{q}_t'(\mathbf{L}_1)\|_F \leq \|\mathbf{R}_t\|\big(\|\mathbf{a}_t(\mathbf{L}_2) - \mathbf{a}_t(\mathbf{L}_1)\|_2\|\mathbf{q}_t(\mathbf{L}_1)\|_2 + \|\mathbf{q}_t(\mathbf{L}_2) - \mathbf{q}_t(\mathbf{L}_1)\|_2\|\mathbf{a}_t(\mathbf{L}_1)\|_2\big). \tag{7.96}$$

Since $\mathbf{a}_t(\mathbf{L})$ and $\mathbf{q}_t(\mathbf{L})$ are Lipschitz and uniformly bounded, from (7.94)-(7.96) one can easily deduce that $\nabla\ell_t(.)$ is indeed Lipschitz continuous. ■

Proof of Lemma 5.2

Exploiting that $\nabla\hat{C}_t(\mathbf{L}[t]) = \nabla\hat{C}_{t+1}(\mathbf{L}[t+1]) = \mathbf{0}_{L\times\rho}$ by algorithmic construction and the strong convexity assumption on $\hat{C}_t$ [cf. a4)], application of the mean-value theorem readily yields

$$\hat{C}_t(\mathbf{L}[t+1]) \geq \hat{C}_t(\mathbf{L}[t]) + \frac{c}{2}\|\mathbf{L}[t+1] - \mathbf{L}[t]\|_F^2$$
$$\hat{C}_{t+1}(\mathbf{L}[t]) \geq \hat{C}_{t+1}(\mathbf{L}[t+1]) + \frac{c}{2}\|\mathbf{L}[t+1] - \mathbf{L}[t]\|_F^2.$$

Upon defining the function $h_t(\mathbf{L}) := \hat{C}_t(\mathbf{L}) - \hat{C}_{t+1}(\mathbf{L})$ one arrives at

$$c\|\mathbf{L}[t+1] - \mathbf{L}[t]\|_F^2 \leq h_t(\mathbf{L}[t+1]) - h_t(\mathbf{L}[t]). \tag{7.97}$$

To complete the proof, it suffices to show that h_t is Lipschitz with constant $\mathcal{O}(1/t)$, and upper bound the right-hand side of (7.97) accordingly. Since [cf. (5.17)]

$$h_t(\mathbf{L}) = \frac{1}{t(t+1)}\sum_{\tau=1}^{t} g_\tau(\mathbf{L}, \mathbf{q}[\tau], \mathbf{a}[\tau]) - \frac{1}{t+1}g_{t+1}(\mathbf{L}, \mathbf{q}[t+1], \mathbf{a}[t+1]) + \frac{\lambda_*}{2t(t+1)}\|\mathbf{L}\|_F^2 \tag{7.98}$$

and $g_i(\mathbf{L})$ is Lipschitz according to Lemma 5.1, it follows that h_t is Lipschitz with constant $\mathcal{O}(1/t)$. ■

Proof of Lemma 5.3

The first step of the proof is to show that $\{\hat{C}_t(\mathbf{L}[t])\}_{t=1}^{\infty}$ is a quasi-matringale sequence, and hence convergent a.s. [71]. Building on the variations of $\hat{C}_t(\mathbf{L}[t])$, one can write

$$\begin{aligned}\hat{C}_{t+1}(\mathbf{L}[t+1]) - \hat{C}_t(\mathbf{L}[t]) &= \hat{C}_{t+1}(\mathbf{L}[t+1]) - \hat{C}_{t+1}(\mathbf{L}[t]) + \hat{C}_{t+1}(\mathbf{L}[t]) - \hat{C}_t(\mathbf{L}[t]) \\ &\overset{(a)}{\leq} \hat{C}_{t+1}(\mathbf{L}[t]) - \hat{C}_t(\mathbf{L}[t]) \\ &= \frac{1}{t+1}\left[g_{t+1}(\mathbf{L}[t], \mathbf{q}[t+1], \mathbf{a}[t+1]) - \frac{1}{t}\sum_{\tau=1}^{t} g_\tau(\mathbf{L}[\tau], \mathbf{q}[\tau], \mathbf{a}[\tau])\right] \\ &\overset{(b)}{\leq} \frac{1}{t+1}\left[g_{t+1}(\mathbf{L}[t], \mathbf{q}[t+1], \mathbf{a}[t+1]) - \frac{1}{t}\sum_{\tau=1}^{t} \ell_\tau(\mathbf{L}[t])\right] \qquad (7.99)\end{aligned}$$

where (a) uses that $\hat{C}_{t+1}(\mathbf{L}[t+1]) \leq \hat{C}_{t+1}(\mathbf{L}[t])$, and (b) follows from $C_t(\mathbf{L}[t]) \leq \hat{C}_t(\mathbf{L}[t])$.

Collect all past data in $\mathcal{F}_t = \{(\Omega_\tau, \mathbf{y}_\tau) : \tau \leq t\}$, and recall that under a1) the random processes $\{\Omega_t, \mathbf{y}_t\}$ are i.i.d. over time. Then, the expected variations of the approximate cost function are bounded as

$$\begin{aligned}\mathbb{E}\left[\hat{C}_{t+1}(\mathbf{L}[t+1]) - \hat{C}_t(\mathbf{L}[t])|\mathcal{F}_t\right] &\leq \frac{1}{t+1}\left(\mathbb{E}[g_{t+1}(\mathbf{L}[t], \mathbf{q}[t+1], \mathbf{a}[t+1])|\mathcal{F}_t] - \frac{1}{t}\sum_{\tau=1}^{t}\ell_\tau(\mathbf{L}[t])\right) \\ &\overset{(a)}{=} \frac{1}{t+1}\left(\mathbb{E}[\ell_1(\mathbf{L}[t])] - \frac{1}{t}\sum_{\tau=1}^{t}\ell_\tau(\mathbf{L}[t])\right) \\ &\leq \frac{1}{t+1}\sup_{\mathbf{L}[t]\in\mathcal{L}}\left(\mathbb{E}[\ell_1(\mathbf{L}[t])] - \frac{1}{t}\sum_{\tau=1}^{t}\ell_\tau(\mathbf{L}[t])\right) \qquad (7.100)\end{aligned}$$

where (a) follows from a1). Using the fact that $\ell_i(\mathbf{L}_t)$ is Lipschitz from Lemma 5.1, and uniformly bounded due to a2), Donsker's Theorem [145, Ch. 19.2] yields

$$\mathbb{E}\left[\sup_{\mathbf{L}[t]}\left|\mathbb{E}[\ell_1(\mathbf{L}[t])] - \frac{1}{t}\sum_{\tau=1}^{t}\ell_\tau(\mathbf{L}[t])\right|\right] = \mathcal{O}(1/\sqrt{t}). \qquad (7.101)$$

From (7.100) and (7.101) the expected non-negative variations can be readily bounded as

$$\mathbb{E}\left[\mathbb{E}\left[\hat{C}_{t+1}(\mathbf{L}[t+1]) - \hat{C}_t(\mathbf{L}[t])|\mathcal{F}_t\right]_+\right] = \mathcal{O}(1/t^{3/2}) \qquad (7.102)$$

and consequently

$$\sum_{t=1}^{\infty}\mathbb{E}\left[\mathbb{E}\left[\hat{C}_{t+1}(\mathbf{L}[t+1]) - \hat{C}_t(\mathbf{L}[t])|\mathcal{F}_t\right]_+\right] < \infty \qquad (7.103)$$

which indeed proves that $\{\hat{C}_t(\mathbf{L}[t])\}_{t=1}^{\infty}$ is a quasi-martingale sequence.

To prove the second part, define first $U_t(\mathbf{L}[t]) := C_t(\mathbf{L}[t]) - \frac{\lambda_*}{2t}\|\mathbf{L}[t]\|_F^2$ and $\hat{U}_t(\mathbf{L}[t]) := \hat{C}_t(\mathbf{L}[t]) - \frac{\lambda_*}{2t}\|\mathbf{L}[t]\|_F^2$ for which $U_t(\mathbf{L}[t]) - \hat{U}_t(\mathbf{L}[t]) = C_t(\mathbf{L}[t]) - \hat{C}_t(\mathbf{L}[t])$ holds. Following similar arguments as with $\hat{C}_t(\mathbf{L}[t])$, one can show that (7.103) holds for $\hat{U}_t(\mathbf{L}[t])$ as well. It is also useful to expand the variations

$$\begin{aligned}\hat{U}_{t+1}(\mathbf{L}[t+1]) - \hat{U}_t(\mathbf{L}[t]) &= \hat{U}_{t+1}(\mathbf{L}[t+1]) - \hat{U}_{t+1}(\mathbf{L}[t]) \\ &+ \frac{\ell_{t+1}(\mathbf{L}[t]) - U_t(\mathbf{L}[t])}{t+1} + \frac{U_t(\mathbf{L}[t]) - \hat{U}_t(\mathbf{L}[t])}{t+1}\end{aligned}$$

and bound their expectation conditioned on $\mathcal{F}_t$, to arrive at

$$\begin{aligned}\frac{U_t(\mathbf{L}[t]) - \hat{U}_t(\mathbf{L}[t])}{t+1} \leq &\left|\mathbb{E}\left[\hat{U}_{t+1}(\mathbf{L}[t+1]) - \hat{U}_{t+1}(\mathbf{L}[t])|\mathcal{F}_t\right]\right| + \left|\mathbb{E}\left[\hat{U}_{t+1}(\mathbf{L}[t+1]) - \hat{U}_t(\mathbf{L}[t])|\mathcal{F}_t\right]\right| \\ &+ \frac{1}{t+1}\left|\mathbb{E}[\ell_1(\mathbf{L}[t])] - \frac{1}{t}\sum_{\tau=1}^{t}\ell_\tau(\mathbf{L}[t])\right|. \qquad (7.104)\end{aligned}$$

Focusing on the right-hand side of (7.104), the second and third terms are both $\mathcal{O}(1/t^{3/2})$ since counterparts of (7.101) and (7.102) also hold for $\hat{U}_t(\mathbf{L}[t])$. With regards to the first term, using the fact that $\hat{C}_{t+1}(\mathbf{L}[t+1]) < \hat{C}_{t+1}(\mathbf{L}[t])$, from Lemma 5.1 and a4), it follows that $\hat{U}_{t+1}(\mathbf{L}[t+1]) - \hat{U}_{t+1}(\mathbf{L}[t]) = o(1/t)$. All in all,

$$\sum_{t=1}^{\infty}\frac{\hat{U}_t(\mathbf{L}[t]) - U_t(\mathbf{L}[t])}{t+1} < \infty \quad \text{a.s.} \qquad (7.105)$$

Defining $d_t(\mathbf{L}[t]) := \hat{U}_t(\mathbf{L}[t]) - U_t(\mathbf{L}[t])$, due to Lipschitz continuity of ℓ_t and g_t (cf. Lemma 5.1), and uniform boundedness of $\{\mathbf{L}_t\}_{t=1}^{\infty}$ [cf a3)], invoking Lemma 5.2 one can establish that $d_{t+1}(\mathbf{L}[t+1]) - d_t(\mathbf{L}[t]) = \mathcal{O}(1/t)$. Hence, Dirichlet's theorem [122] applied to the sum (7.105) asserts that $\lim_{t\to\infty} d_t(\mathbf{L}[t]) = 0$ a.s., and consequently $\lim_{t\to\infty}(\hat{C}_t(\mathbf{L}[t]) - C_t(\mathbf{L}[t])) = 0$ a.s. ■