Inferring Smartphone Service Quality using Tensor Methods

Vaneet Aggarwal[†], Ajay Mahimkar[‡], Hongyao Ma[□], Zemin Zhang[♦], Shuchin Aeron[♦], Walter Willinger[△]

Purdue University [†] AT&T [‡] Harvard University [□] Tufts University [♦] Niksun [△]

***Abstract*—Cellular network providers collect and use a wide variety of data for assessing the service quality experienced by their smartphone users. The data is essential for tasks ranging from event detection, problem diagnosis, impact analysis, coverage and capacity planning, load balancing, and performance optimization. For example, service quality measurements and data from drive-by tests provide useful and detailed information about different aspects of quality of service such as dropped calls due to handovers or radio interference. However, a major challenge for effective service quality management in operational setup is the presence of missing or unavailable data. Furthermore, the cellular data is inherently multidimensional, i.e. is a function of several variables such as location, device type, and time. Motivated by recent advances in handling multidimensional data, we propose to use tensor algebraic models and methods for cellular data prediction. The main idea is to model the data as a low rank tensor and use a rank constrained interpolation for data prediction. We focus on two recently proposed algebraic models employing two different notions of tensor rank. We test and compare the performance of the two approaches on real-world data sets collected from an operational cellular network and indicate the regimes in which one method is superior to the other. Based on these observations the proposed algorithm chooses the best of the two approaches using cross-validation.**

I. Introduction

There has recently been an enormous increase in the usage of cellular voice and data services. Smartphone users rely heavily on such services for a variety of day-to-day activities and demand very high availability and reliability. To assess the user's service quality of experience, the cellular service providers collect and analyze a wide variety of data. The resulting data sets range from performance indicators and configuration files to workflow logs and alarm tickets and are invaluable for tasks such as incident or anomaly detection, troubleshooting of problems, impact assessment of ongoing network changes, planning of coverage and capacity, and large-scale performance optimizations [1], [2], [18].

One of the key challenges with effective service quality analysis is missing measurements. Missing measurements do occur in operational networks due to multiple reasons such as failure of measurement systems, overload scenarios, or degrading service conditions. In some cases, fine-grained measurements (e.g., drive tests based location-centric smartphone service performance) cannot be collected continuously in time across all the locations and for all the users. Thus, one needs to develop robust measures for filling-in or inferring missing measurements. The filled-in or completed data would be of immense practical value, not only for improving the accuracy of existing service performance analysis tools but also for developing new and better analysis methods. Traditionally, missing measurement inference in large networks relies on matrix (2-D) completion techniques ([7], [9], [19], [25], [28]). However, for our context of smartphone service quality inference, the information is available in more than two dimensions - (i) time, (ii) space or network location, (iii) smartphone type, and (iv) performance measurement. Therefore, in this paper, we explore the application of tensor models and methods for the challenging problem of "filling-in" or inferring the missing service quality measurements.

978-3-901882-85-2

Missing data has been widely studied in the areas such as network traffic analysis [7], [9], [19], [22], [26], [25], [28], [29], [30], [7], computer vision [4], localization in mobile networks [20], coverage estimation [23], and climate estimation [24]. A common theme in these works is that data sets of interest are intrinsically low-rank or can be well approximated by a low rank matrix plus some noise. To complete the data with missing entries, many matrix completion algorithms have been proposed [21], [5], [13], [6]. For multidimensional data, tensor completion has been studied using matrix unfoldings in [10], using Riemannian methods in [16], [8] using Hierarchical Tucker (HT) [12] tensor decomposition and using another tensor-SVD [14] like decomposition in [31]. Unlike the convex analytic approaches in [10], [31] for which, performance guarantees can be given, the approaches based on other tensor decompositions such as HT [16], [8] it is not feasible to provide global performance guarantees. Therefore, in this paper we focus on the methods considered in [10], [31]. On the other hand there is little work in terms of algorithm development when one is allowed to sample (non-adaptively) as well as take dense linear combinations, in particular collect average or aggregate statistics. In this paper, we show that this side-information significantly improves the accuracy and present an efficient algorithm to incorporate these constraints. We outline our main contributions below.

1. We consider two algebraic rank measures derived from using two different algebraic models to model tensor data. The first model is based on capturing the multilinear rank of a tensor obtained through the Singular Value Decomposition (SVD) of matrices constructed from the tensor using mode unfoldings or flattenings - a process where one extracts 1-D tensor fibers along the axes and stacks them as columns of a matrix. On the other hand the

second model is based on an approach that treats tensors as linear operators over commutative rings which in turn are constructed out of tensor fibers and employs a tensor-SVD (t-SVD) to derive tensor rank. The proposed algorithm uses the best of these two rank measures based on cross-validation for data completion.

2. We conduct an extensive evaluation of the proposed methods and use three real-world operational cellular network data sets (i.e., two, three and four dimensions) and show performance under different regimes of sampling rates. We show clear benefits over the naive slice by slice completion approach indicating that using tensor based approaches are superior for cellular data prediction, (see Section IV).
3. While both algorithms exploit the low tensor rank nature of the data, the algorithm based on the tensor-SVD exploits the periodicity in the data (see section II) by operating in the Fourier domain. We find that with more available data, exploiting the periodic structure (harmonics) in the data helps and thus the algorithm based on the tensor-SVD is better while when the available data is small, fitting a periodic structure to the data leads to over-fitting and thus the algorithm based on tensor-SVD do not perform as well. To the best of our knowledge this is the first time that it is reported that there is not one tensor method that dominates under all scenarios. Different operating regimes and data types can lead to different tradeoffs in prediction performance under various tensor algebraic models to model multidimensional data.
4. To the best of our knowledge, this is the first work that applies the concepts of tensor completion to network data. Furthermore, a novel element of this work is that it explores and compares completion with different algebraic rank measures when, in addition to element-wise samples, linear combinations of the elements are also known, e.g. we use the additional information of the average call quality along the temporal dimension.

II. Capturing Dependencies in Data

The smartphone users communicate via the cellular towers (known as NodeB in 3G and eNodeB in LTE). They can operate either in the Packet Switched (PS) mode for data services, Circuit Switched (CS) mode for voice services, or simultaneous PS and CS modes. The Radio Network Controller (RNC) manages the radio resources and connects the radio access network (RAN) to the core. We focus on UMTS in this paper, though our approach applies generically to LTE and beyond. As indicated before, we model the smartphone data as a multidimensional array of service performance measurements. In particular, we model the data using four dimensions: (i) *space* or network locations where data is measured or aggregated (e.g., cellular towers), (ii) *time* dimension captures the aggregate summaries of performance (e.g., every 15 minutes, hour, or day), (iii) *KPI*s - Key Performance Indicators capture the service performance experienced by the end-users, and (iv) *smartphone attributes* such as type, make, model and OS (operating system) version.

We now describe some of the KPI that we use in the paper. *Accessibility* captures successful calls established by the

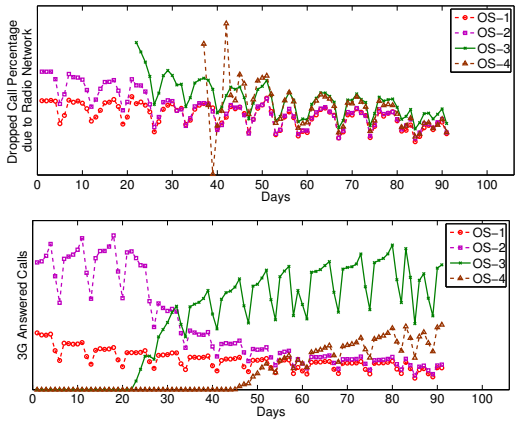

Fig. 1. Service performance dependency across operating system (OS) versions for a smartphone make.

smartphones on the cellular network. *Retainability* captures the retention of the call by the network. If the call was terminated abnormally by the network, then it would lead to a reduced retainability value. *Data throughput* captures the number of bits, bytes or packets delivered to the users over the cellular network. The end-to-end throughput is broken down into: (i) over the air interface or radio access network throughput and (ii) core throughput. Voice Erlangs captures the voice load carried by the cellular towers. For data sessions, total traffic volumes are measured in bytes and packets.

We use the following data sets for evaluation: two-dimensional (2D) *call detail records* (CDR) - space and time; three-dimensional (3D) *service performance indicators* - space, time and type of KPI; and four-dimensional (4D) *smartphone specific measurements* - space, time, KPI and smartphone attributes.

The challenge with analyzing multi-dimensional data is that the dependency structure is typically hidden in high dimensions. We can get glimpses of this structure when looking at two dimensions, but reconstructing the overall dependency structure from 2-dimensional slices is in general not feasible.

Figure 1 captures the dependency across operating system (OS) versions on a smartphone model using three months worth of data collected from an operational cellular network. The X-axis is the days and the Y-axis captures the dropped call percentage and the number of answered voice calls. Each point on the time-series is a daily aggregate across all the smartphones corresponding to the OS version. As can be seen from the figure, OS-4 is the most recent version of the operating system, OS-1 is the least popular based on the number of answered calls, OS-2 is decreasing in popularity, and OS-3 and OS-4 are growing in popularity. There is a dependency across operating system (OS) versions within a type of smartphone model. This information present in higher dimensions proves to be very valuable when interested in accurately completing any missing entries.

III. Method for Data Completion using the Algebraic Models

Although there are many types of tensor factorizations [12], we will focus on two types of approaches for characterizing the algebraic rank measure for tensors. One way

to capture algebraic dependency in tensors is to use mode unfoldings of the tensor [17]. The algebraic rank measure is the weighted sum of nuclear norms over each model unfolding, $\mathcal{C}(\boldsymbol{\mathcal{X}}) = \sum_m w_m \|(\boldsymbol{\mathcal{X}}^{(m)})\|_*$ [27], where w_m are the weights on each mode $\boldsymbol{\mathcal{X}}^{(m)}$. Algorithms for tensor completion that minimize this algebraic rank measure can be found in [11]. Second approach [15] preserves the tensor's relative orientation. Using an *orientation dependent* tensor-SVD decomposition, we consider the following algebraic rank measure, $\mathcal{C}(\boldsymbol{\mathcal{X}}) = \sum_{o=1}^{\binom{k}{2}} w_o \|\boldsymbol{\mathcal{X}}^{(o)}\|_{tnn}$, where w_o is the weight of orientation, k is the tensor order, and tensor nuclear norm is used as the convex relaxation of the tubal rank [31].

Assuming that the data has a low tensor rank, a natural approach to predict the missing entries from the given observations is to use the following complexity penalized tensor completion method.

$$\min_{\boldsymbol{\mathcal{X}}} \mathcal{C}(\boldsymbol{\mathcal{X}}) \quad s.t. \quad \mathcal{L}(\boldsymbol{\mathcal{X}}) = \mathbf{y}, \tag{1}$$

where $\mathcal{L}(\boldsymbol{\mathcal{X}}) = \mathbf{y}$ represents the known linear constraints. Here $\mathcal{L}$ incorporates both the fine-grained (element-wise samples) and the coarse-grained i.e. aggregated smartphone service quality measurements. Since the method incorporates the two algebraic rank measures considered in this paper through $\mathcal{C}(\boldsymbol{\mathcal{X}})$, the proposed algorithm chooses one of the two algebraic rank measures based on cross-validation. Using the cross-validation approach, we choose a certain percent of the available data as training data, and rest as test data. Tensor completion using the minimization of the two algebraic measures is independently performed to find the completion error on the test data. The algorithm then chooses the approach which gives better error performance. We use Alternating Direction Method of Multipliers (ADMM) [3] to solve the convex optimization problem expressed in Equation (1).

IV. Evaluation

We now evaluate the performance of the different completion methods, namely slice by slice matrix completion, SVD of mode unfoldings, and using the t-SVD with oriented tensor factorization. We measure the accuracy using the *Normalized Mean Square Error* and *Approximation Error* as follows:

$$\text{NMSE} = \frac{||\text{Actual Test Data - Predicted Test Data}||}{||\text{Actual Test Data - Mean Actual Training Data}||} \tag{2}$$

$$\text{Approximation Error} = 100 \times \text{NMSE}\% \tag{3}$$

A. Call Detail Records (2D)

We begin by considering 2-D data completion. We will show that using additional linear constraints, i.e. coarse grained information can lead to significant performance gains and therefore can be very useful in an operational setting. We form a matrix of size 1144×936 which is the data for all $39 \times 24 = 936$ hours and where the 1144 columns represent sectors on three RNCs (Radio Network Controllers) for which the data is measured. Each entry in the matrix is the number of successful calls. We artificially inject missing entries and then compare the results against the ground truth (i.e., the original data). To this end, we sample the matrix elements randomly (i.e., independently and equally likely) with probability p, and use these sampled points as our training data (or the input data that we use for the algorithm). Further, we consider the data when aggregated over H hours to be completely known. Lastly, to demonstrate that knowing linear combinations is more effective than knowing the percentage of matched total entries in the matrix, we perform our evaluation for different combinations of p and H.

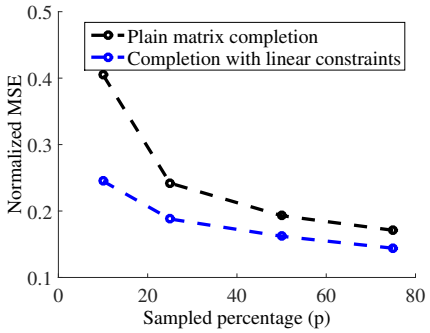

Fig. 2. Matrix completion for 2D call detail records.

Since the data is available on an hourly basis, having linear combinations aggregated over H hours is equivalent to working with $100/H\%$ of the data. For example, $H = 1$ means that the complete data is available; on the other hand, setting $H = 24$ (i.e., daily aggregates) is equivalent to working with $100/24 = 4.167\%$ of the data. When combined with sampling, consider for example the case $p = .25$ and $H = 24$. Here,we have 25% of the original data available as a result of sampling. In addition, we also have an equivalent of 4.167% of the original data available due to linear combinations. Thus, with this combination of parameters, we are in fact working with an equivalent to 29.167% of the original data. In Figure 2, we assume $H = 24$ (daily aggregates) and compare the accuracy of plain matrix completion (i.e., effectively working with $p+1/H$ fraction of the data) and matrix completion with linear constraints. We observe that having daily aggregates reduces the error as compared to having more more data (i.e., fewer missing values). For example, having 25% of the data available and using daily aggregates is more beneficial (i.e., higher accuracy) than having 50% of the original data and no daily aggregates. This result illustrates that knowing linear aggregates can often be much more beneficial than working with an equivalent amount of the original data.

B. Service performance indicators (3D)

In this case, we consider a three-dimensional tensor consisting of service performance data. This data set is of size $314 \times 360 \times 73$ for 314 RNCs (Radio Network Controllers) and for 360 hours and contains a total of 73 KPIs. The data contains several KPIs including voice and data accessibility, retainability, RRC, SRB and RAB success rates, paging success rates, uplink and downlink traffic, voice Erlangs or minutes of usage. We sample the tensor elements randomly (independent and equally likely) with probability p to obtain the incomplete tensor, and complete this tensor using our approach. Note that from Figure 3 we should expect that the

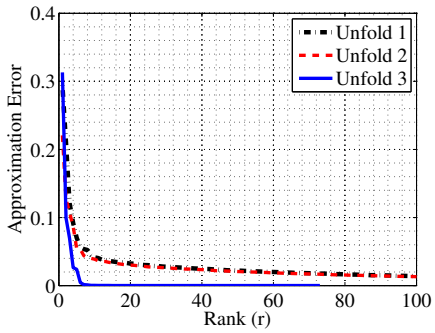

Fig. 3. Error with rank r approximation on the unfolds.

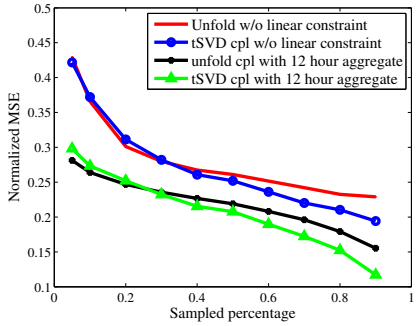

Fig. 4. Completing the 3D tensor with varying sampled data with/without aggregate linear constraints.

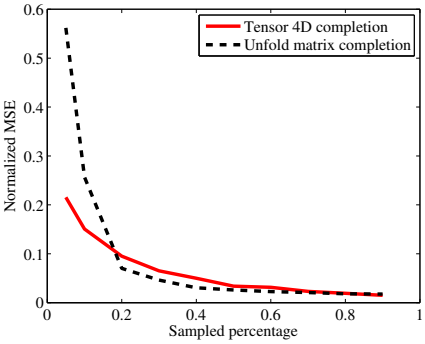

Fig. 5. Completing the 4D tensor with varying sampled data on the 68% available data.

method based on t-SVD to perform better compared to the method based on tensor unfolding.

Figure 4 depicts the completion based on the two rank-measures at very low to high sampling. The method based on tensor unfolding performs better at low to intermediate sampling rates since t-SVD based method might be over-fitting the data. At very low sampling rates, tensor unfolding based method performs very bad since the data may not be enough to complete the data just relying on low rank, while t-SVD performs better by exploiting both low rank and the periodic structure of the data. In the intermediate sampling, unfolding based method performs better since in this regime t-SVD may be over-fitting the noise. At high sampling, both the low rank and the periodic structure can be effectively exploited and hence results in improvement in completion.

C. Smartphone specific data (4D)

In this case, we consider a four-dimensional tensor extracted from smartphone specific measurements. The size of the tensor is $29 \times 253 \times 97 \times 5$. The first dimension is the number of days (29); 253 is the number of smartphone types; 97 is the number of network locations where the measurements are aggregated; and 5 is the number of KPIs from voice call detail records. This 4D tensor has inherently missing data because of the sparse population of users across certain types of smartphones. Thus, the missing data in this case is not random, and has a structure which is given by the available measurements.

Our tensor-based completion approach can be used to predict the places where there is no data. In this tensor, we have only 68% entries available. We use cross-validation to study the performance of our algorithm. We sample the available data with probability p (choosing each element among the available data randomly with probability p), and check the error on the remaining unsampled available data ($1-p$ fraction of 68% data). Figure 5 gives the error on the unsampled data as p increases. Based on the accuracy for different unfolds, we find that for this data, unfolding onto the first dimension yields the best results while using the t-SVD approach we see that fixing the tensor orientation as a $253 \times 97 \times 5 \times 29$ tensor (obtained by simply permuting the indices) yields the best results. Figure 5 shows the results for the two cases.

While both methods have good accuracy, the method based on tensor unfolding seems to perform better at low sampling rates while the method based on t-SVD seems to perform better at higher sampling rates. At extremely low sampling rates however, the unfold bases method does not perform as well, similar to our 3D results. We further note that at 90% sampling, both the tensor based methods have normalized MSE around 3% (better with tensor SVD based method), the approach based on completing two-dimensional slices independently gives an error of 27%. Thus, exploiting the multi-dimensional nature of the network data gives significant improvement in data completion.

V. Conclusions and Future Work

In this paper, we exploit multidimensional algebraic models and methods to infer the missing service quality measurements utilizing the multi-dimensional dependency structure in the data. The proposed approach uses the best of two different tensor factorizations (based on t-SVD and matricization) based on cross-validation. Using real-world data collected from operational cellular network, we demonstrated that our algorithm outperforms existing methods across different types of data sets. In the future, we will explore the suitability of our approach for real-time anomaly detection, statistical prediction of fine-grained service quality and root-cause classification.

Acknowledgement

We thank Supratim Deb, Zihui Ge, Sarat Puthenpura, Jennifer Yates, David Bastien, our shepherd Remi Badonnel and the CNSM anonymous reviewers for their insightful feedback on the paper. The paper co-authors Zemin Zhang and Shuchin Aeron are supported in part by NSF:CCF:1319653.

REFERENCES

[1] V. Aggarwal, E. Halepovic, J. Pang, S. Venkataraman, and H. Yan. Prometheus: Toward quality-of-experience estimation for mobile apps from passive network measurements. In *HotMobile*, 2014.

[2] A. Balachandran, V. Aggarwal, E. Halepovic, J. Pang, S. Seshan, S. Venkataraman, and H. Yan. Modeling web quality-of-experience on cellular networks. In *ACM MOBICOM*, 2014.

[3] S. Boyd. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[4] A. Buchanan and A. Fitzgibbon. Damped newton algorithms for matrix factorization with missing data. In *IEEE Computer Society Conference on CVPR*, volume 2, pages 316–322 vol. 2, 2005.

[5] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky. The convex geometry of linear inverse problems. *Foundations of Computational Mathematics*, 12(6):805–849, 2012.

[6] Y. Chen, S. Bhojanapalli, S. Sanghavi, and R. Ward. Completing Any Low-rank Matrix, Provably. *arXiv.org*, June 2013.

[7] Y.-C. Chen, L. Qiu, Y. Zhang, G. Xue, and Z. Hu. Robust network compressive sensing. In *ACM MOBICOM*, 2014.

[8] C. Da Silva and F. J. Herrmann. Optimization on the Hierarchical Tucker manifold - applications to tensor completion. *ArXiv e-prints*, May 2014.

[9] V. Erramill, M. Crovella, and N. Taft. An independent-connection model for traffic matrices. In *ACM IMC*, 2006.

[10] S. Gandy, B. Recht, and I. Yamada. Tensor completion and low-n-rank tensor recovery via convex optimization. *Inverse Problems*, 27(2):025010, 2011.

[11] D. Goldfarb and Z. Qin. Robust low-rank tensor recovery: Models and algorithms. *SIAM Journal on Matrix Analysis and Applications*, 35(1):225–253, 2014.

[12] L. Grasedyck, D. Kressner, and C. Tobler. A literature survey of low-rank tensor approximation techniques. *ArXiv e-prints*, Feb. 2013.

[13] P. Jain, P. Netrapalli, and S. Sanghavi. Low-rank matrix completion using alternating minimization. In *STOC*, 2013.

[14] M. E. Kilmer and C. D. Martin. Factorization strategies for third-order tensors. *Linear Algebra and its Applications*, Special Issue in Honor of G. W. Stewart's 70th birthday, Vol. 435(3):641–658, 2011.

[15] M. E. Kilmer and C. D. Martin. Factorization strategies for third-order tensors. *Linear Algebra and its Applications*, 435(3):641–658, 8 2011.

[16] D. Kressner, M. Steinlechner, and B. Vandereycken. Low-rank tensor completion by Riemannian optimization. *Bit Numerical Mathematics*, 54(2):447–468, June 2014.

[17] L. D. Lathauwer and B. D. Moor. From matrix to tensor: Multilinear algebra and signal processing. In J. McWhirter and e. I. Proudler, editors, *Mathematics in Signal Processing IV*, pages 1–15. Clarendon Press, Oxford, UK, 1998.

[18] A. Mahimkar, Z. Ge, J. Yates, C. Hristov, V. Cordaro, S. Smith, J. Xu, and M. Stockert. Robust assessment of changes in cellular networks. In *ACM CoNEXT*, 2013.

[19] A. Medina, N. Taft, K. Salamatian, S. Bhattacharyya, and C. Diot. Traffic matrix estimation: existing techniques and new directions. In *ACM SIGCOMM*, 2002.

[20] S. Rallapalli, L. Qiu, Y. Zhang, and Y.-C. Chen. Exploiting temporal stability and low-rank structure for localization in mobile networks. In *ACM MOBICOM*, 2010.

[21] B. Recht, W. Xu, and B. Hassibi. Necessary and sufficient conditions for success of the nuclear norm heuristic for rank minimization. In *47th IEEE Conference on Decision and Control*, pages 3065 –3070, Dec 2008.

[22] D. Rincón, M. Roughan, and W. Willinger. Towards a meaningful mra of traffic matrices. In *ACM IMC*, 2008.

[23] B. Sayrac, J. Riihijarvi, P. Mahonen, S. B. Jemaa, E. Moulines, and S. Grimoud. Improving coverage estimation for cellular networks with spatial bayesian prediction based on measurements. In *CellNet*, 2012.

[24] T. Schneider. Analysis of Incomplete Climate Data: Estimation of Mean Values and Covariance Matrices and Imputation of Missing Values. *Journal of Climate*, 14:853–871, Mar. 2001.

[25] A. Soule, A. Lakhina, N. Taft, K. Papagiannaki, K. Salamatian, A. Nucci, M. Crovella, and C. Diot. Traffic matrices: balancing measurements, inference and modeling. In *ACM SIGMETRICS*, 2005.

[26] A. Soule, A. Nucci, R. Cruz, E. Leonardi, and N. Taft. How to identify and estimate the largest traffic matrix elements in a dynamic environment. In *ACM SIGMETRICS*, 2004.

[27] R. Tomioka, T. Suzuki, K. Hayashi, and H. Kashima. Statistical performance of convex tensor decomposition. In *NIPS*, pages 972–980, 2011.

[28] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg. Fast accurate computation of large-scale ip traffic matrices from link loads. In *ACM SIGMETRICS*, 2003.

[29] Y. Zhang, M. Roughan, C. Lund, and D. Donoho. An information-theoretic approach to traffic matrix estimation. In *ACM SIGCOMM*, 2003.

[30] Y. Zhang, M. Roughan, W. Willinger, and L. Qiu. Spatio-temporal compressive sensing and internet traffic matrices. In *ACM SIGCOMM*, 2009.

[31] Z. Zhang, G. Ely, S. Aeron, N. Hao, and M. Kilmer. Novel methods for multilinear data completion and de-noising based on tensor-svd. In *IEEE CVPR*, June 2013.