IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received April 13, 2016, accepted May 1, 2016, date of publication May 26, 2016, date of current version July 7, 2016.

Digital Object Identifier 10.1109/ACCESS.2016.2573264

Spatio-Temporal Kronecker Compressive Sensing for Traffic Matrix Recovery

DINGDE JIANG[1], (Member, IEEE), LAISEN NIE[1], ZHIHAN LV[2], AND HOUBING SONG[3], (Senior Member, IEEE)
[1]School of Computer Science and Engineering, Northeastern University, Shenyang 110819, China
[2]Department of Computer Science, University College London, London WC1E 6BT, U.K.
[3]Department of Electrical and Computer Engineering, West Virginia University, Montgomery, WV 25136, USA

Corresponding author: D. Jiang (jiangdd@mail.neu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61571104 and Grant 61071124, in part by the General Project of Scientific Research of the Education Department of Liaoning Province under Grant L20150174, in part by the Program for New Century Excellent Talents in University under Grant NCET-11-0075, in part by the Fundamental Research Funds for the Central Universities under Grant N150402003, Grant N120804004, and Grant N130504003, in part by the Science and Technology Program of Shaanxi Province under Grant No. 2016KW-032, and in part by the State Scholarship Fund under Grant 201208210013.

ABSTRACT A traffic matrix is generally used by several network management tasks in a data center network, such as traffic engineering and anomaly detection. It gives a flow-level view of the network traffic volume. Despite the explicit importance of the traffic matrix, it is significantly difficult to implement a large-scale measurement to build an absolute traffic matrix. Generally, the traffic matrix obtained by the operators is imperfect, i.e., some traffic data may be lost. Hence, we focus on the problems of recovering these missing traffic data in this paper. To recover these missing traffic data, we propose the spatio-temporal Kronecker compressive sensing method, which draws on Kronecker compressive sensing. In our method, we account for the spatial and temporal properties of the traffic matrix to construct a sparsifying basis that can sparsely represent the traffic matrix. Simultaneously, we consider the low-rank property of the traffic matrix and propose a novel recovery model. We finally assess the estimation error of the proposed method by recovering real traffic.

INDEX TERMS Traffic matrix recovery, Kronecker compressive sensing, matrix completion, network measurement, network management.

I. INTRODUCTION

As a crucial technique in cloud computing, the data center network has gained much more attention. The number of services provided by data center networks is much larger than before. Therefore, various types of network management functions (e.g., traffic engineering and anomaly detection) are used in a data center network to guarantee its efficient implementation [1]–[3]. A traffic matrix (TM) is used as a crucial reference by various network management operations. Initially, despite the explicit importance of TMs, it is significantly difficult to exactly and directly measure a TM. The expensive consumption of monitoring techniques has triggered the development of TM estimation techniques.

The strategy of TM estimation can be defined as an inverse inference problem, where TM is estimated via link counts and routing information [2]–[10]. The dominant issue in TM estimation is that the inference problem is under-constrained. Motivated by this, numerous methodologies have emerged to address this under-constrained feature of TM estimation, e.g., the principal component analysis method in [6], the tomogravity method in [7] and the route change method in [8]. These previous methods usually refer to a prior of the TM or a statistical model as additional information to handle the under-constrained feature of the TM estimation problem. With the advent of novel applications, the traffic flow reveals much more complicated statistical features. Hence, some hybrid methods have been proposed, e.g., the works in [9] and [10]. Considering the spatial property of the TM, the authors in [11] propose a modified gravity model for TM estimation. The authors in [12] focus on the TM estimation problem in a Software Defined Network, and propose an evolutionary approach. In [13], the authors take into account the spatial and temporal properties of TM, and propose a regularized optimization model for TM estimation. Monitoring techniques can implement a direct measurement for obtaining the TM. Nevertheless, the traffic data from direct measurement may be imperfect. The works in [14] and [15] focus on the TM recovery problem via partial and imperfect traffic data (i.e., the real TM measurements may be lost).

The common reasons for imperfect traffic data are multiple [14], [15]. First, flow collection devices that implement direct measurement may fail. Second, considering the scalability requirements, flow-level direct measurement may not occur at the edge of a network, although the data from edge nodes is significantly crucial for building a TM. Further, it is difficult to ensure that all devices in our networks support flow-level measurement. Hence, the traffic data that operators obtain may be those with missing entries.

Matrix completion techniques and other interpolation algorithms are popular approaches to recovering the missing traffic data of a TM [14], [15]. However, the current interpolation algorithms cannot accurately recover the entries of a traffic matrix for a data center network due to the complicated statistical features of traffic flows. On the one hand, a traffic matrix exhibits a spatially structured low-rank property. On the other hand, for the time domain, each origin and destination (OD) flow pair has various statistical properties, e.g., multifractal properties and a heavy-tailed distribution [16]–[21]. Consequently, recovering the missing data of the traffic matrix still is a significantly great challenge under the existing technical conditions.

Motivated by these issues, we address the problem of recovering a traffic matrix with missing data in this paper. The method proposed in this paper draws from the idea of the Kronecker Compressive Sensing (KCS) method [22]–[24]. We refer to the proposed method as the Spatio-Temporal Kronecker Compressive Sensing method (STKCS). In our method, we first explore a sparse representation of the TM with respect to different dimensions. Then, according to the proposed sparse representation of the TM, an available recovery model is built based on the KCS method. Our contributions in this paper are as follows:

- We study the problem of network traffic recovery in a data center network. The data center network plays a crucial role in cloud computing. However, how to obtain the network traffic information of a data center network has not been given enough attention. Although several works have considered the TM problem in a data center network, they did not consider the noise caused by missing data.
- We propose the STKCS method to recover the missing data of a TM. In our method, we first consider the spatial and temporal properties of the TM simultaneously to find a sparse representation of it. With this representation, the problem can be solved via a convex optimization. From the evaluation of the STKCS method, it is deemed suitable for capturing the time-varying property of network traffic in a data center network.
- We model the characteristics of missing traffic data in a data center network. As far as we know, this paper is the first to research the characteristics of missing traffic data. Most of the existing works usually address the problem of modeling the characteristics of traffic flow in a data center network.

The rest of this paper is organized as follows. We briefly introduce the background of the traffic matrix recovery problem in section 2. We then present the KCS theory in section 3. In section 4, we propose our method. To evaluate the performance of our approach, we provide numerical results by using a real traffic data set in section 5. Finally, we conclude our work in section 6.

II. PROBLEM STATEMENT

For a network with Q nodes, there are $N = Q^2$ OD pairs. Without loss of generality, we denote is a TM by M, and its entries are $m(n, t)$, $n = 1, 2, \ldots, N$, $t = 1, 2, \ldots, L$. Each entry of the traffic matrix describes an average of network traffic over a time interval. For instance, if the TM is measured for 5 minute slot, each entry of TM is the average of the traffic during the 5-minute time slot. A state-of-the-art approach to estimating such a TM is the network tomography method, which uses link loads and the routing matrix to infer the TM [3]–[5]. From current traffic monitoring techniques, we can collect flow-level network traffic using direct measurement of each node in our networks. Then, the traffic data will be sent from each node to the network management station. Nevertheless, the challenge here is that the traffic data may be lost [14], [15]. Hence, we still cannot easily obtain an available TM. The network management station always receives a TM with missing entries. We denote this matrix received by the network management station as

$$M' = \mathcal{A}(M), \tag{1}$$

where $\mathcal{A}$ is the operator that denotes the process of missing data. This operator can be absolutely expressed by using the Hadamard product

$$M' = B \odot M. \tag{2}$$

The $N \times L$ matrix B, whose entries are 1 or 0, describes the condition of the missing data. That is, $b_{n,t} = 0$ means that the traffic data $m(n, t)$ is lost. Obviously, the matrix B is a stochastic matrix because the traffic data are lost irregularly. We here propose an available approach to estimating the traffic matrix M via its direct measurement M'.

III. KRONECKER COMPRESSIVE SENSING

Compressive sensing is a non-adaptive sampling technique that is widely used in image and signal processing [22], [23]. It utilizes a stochastic matrix as the measurement matrix to carry out sampling and compression at the same time. For the decoding procedure, it can exactly recover the original data from few measurements of itself. The reason that one can precisely recover the original data is that two rigorous conditions are held. One is that the original data must be sparse or compressible. The other is that the measurement matrix has to obey the restricted isometry property (RIP), as mentioned in [22]–[24].

The compressive sensing theory has been vastly extended, corresponding to the practical problems that have emerged in various research fields. The traditional compressive sensing theory is usually used in signal and image processing. Most of the literature focuses on the problems of 1-D signal and 2-D image data. To address multidimensional data, M. Duarte and R. Baraniuk proposed the Kronecker Compressive Sensing theory [24]. It takes advantage of the sparsity of the data in various dimensions to implement a non-adaptive sampling. Namely, it provides a scheme for the design of a sensing matrix for multidimensional data. Such a sensing matrix ensures that the multidimensional data can be sparsely represented for each dimension and obeys the RIP.

A stochastic measurement matrix is used to implement sampling and compression tasks simultaneously, which can be denoted as $y = \Phi x$. Φ is the stochastic measurement matrix [22], [23]. Each row of Φ can be considered a sensor [24]. The vector y is a sampling of x in terms of the measurement matrix Φ. As mentioned above, x must be sparse or compressible, i.e., it can be denoted by a sparse representation $x = \Psi\alpha$. α is the coefficient vector with respect to the sparsifying basis Ψ. The RIP has an explicit unitary nature. In other words, if the measurement matrix Φ has the RIP, then the sensing matrix $\Theta = \Phi\Psi$ will also have this nature. This sampling technique is more suitable for problems of 1-D or 2-D data. Different from the traditional compressive sensing theory, KCS in [24] explores a sampling scheme for multidimensional data. In the KCS theory, assume that for a J-D data $x \in \mathbb{R}^{N_1\times N_2\times\ldots\times N_J}$, we define the first dimension as a 1-section of x, namely,

$$x(:, n_2, \ldots, n_J) = [x(1, n_2, \ldots, n_J), \ldots, x(N_1, n_2, \ldots, n_J)], \quad (3)$$

where $n_i = 1, 2, \ldots, N_i$, $i = 2, 3, \ldots, J$. This definition can be extended to a j-section of x. For instance, for 3-D data $x \in \mathbb{R}^{N_1\times N_2\times N_3}$, $x(n_1, :, n_3) = [x(n_1, 1, n_3), \ldots, x(n_1, N_2, n_3)]$ is the 2-section of x. The KCS theory assumes that for the J-section of x, there are J sparse representations of x. Their sparsifying bases are $\Psi_1, \Psi_2, \ldots, \Psi_J$, respectively. In this case, the sparsifying basis of $x \in \mathbb{R}^{N_1\times N_2\times\ldots\times N_J}$ can be defined as $\Psi_{KCS} = \Psi_1 \otimes \Psi_2 \otimes \ldots \otimes \Psi_J$. Then, one can carry out a non-adaptive sampling for x, which can be denoted by

$$y_{KCS} = \Phi_{KCS}\Psi_{KCS}\alpha_{KCS}. \quad (4)$$

y_{KCS} is the sampling of x_{KCS}, and

$$\Phi_{KCS} = \begin{bmatrix} \Phi_1 & 0 & \cdots & 0 \\ 0 & \Phi_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \Phi_J \end{bmatrix}. \quad (5)$$

$\Psi_{KCS}\alpha_{KCS}$ is the vector-reshaped representation of x [24]. The decoding procedure for a receiving terminal is to compute α_{KCS} from its measurements y_{KCS}. To accurately recover the sparse data α_{KCS}, the measurement matrix Φ_{KCS} must obey the RIP. Because determining whether a sensing matrix obeys the RIP is difficult, mutual coherence is involved to provide a necessary condition for the exact recovery of α_{KCS} [24].

IV. SPATIO-TEMPORAL KRONECKER COMPRESSIVE SENSING

A. SPARSE REPRESENTATION OF TRAFFIC MATRICES

TMs have a remarkable low-rank feature. Specifically, considering the spatial properties, the volume of each traffic flow exhibits a power-law property [6], [11], [12], [15]. Partial large (or crucial) traffic flows provide the main power of the TM. In our method, we use the principal component analysis method to probe the sparse representation of the TM. The principal component analysis method uses singular value decomposition as a tool to research the low-rank features of a TM. The singular value decomposition of the transpose of the N-by-L matrix ($N < L$) is defined as

$$M^T = U\Sigma_N V^T. \quad (6)$$

The matrix V is an N-by-N orthogonal matrix. Σ_N is an N-by-N diagonal matrix whose diagonal entries are the singular values of the matrix M. U is an L-by-N matrix that describes the dynamic characteristics of the matrix M. In the principal component analysis method, one can find a low-rank approximation of the matrix M^T with K principal components, namely,

$$M \approx V\Sigma_K \mathrm{U}_{PC}^T, \quad (7)$$

where Σ_K is the diagonal matrix with the top K largest singular values of Σ_N. Subsequently, we can obtain an approximately sparse representation of the matrix M, i.e., $M \approx Vs$ ($s = \Sigma_K \mathrm{U}_{PC}^T$). In this case, V and s are the orthogonal basis and the corresponding coefficients, respectively. The missing data cannot change the principal components of the TM in practice, which means that the matrix M' can be represented by $M' \approx Vs'$. s' is the coefficient matrix in terms of M'. We now analyze the temporal features of the TM. In the above, we denote the missing data as M'. Considering a row of M', when the entries of the matrix M are lost, we state that the row of the TM measurements M' is sparse in the time domain. Hence, we can simply achieve a sparse representation of the traffic matrix measurements M' by using an L-by-L identity matrix I_L.

We have given two sparse representations of the TM measurements. As mentioned above, the two bases are V and I_L for the spatial and temporal properties, respectively. Therefore, we can gain a sparsifying basis by using the Kronecker product, denoted as

$$\Psi = I_L \otimes V = \begin{bmatrix} V & 0 & \cdots & 0 \\ 0 & V & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & V \end{bmatrix}. \quad (8)$$

In this equation, Ψ is LN-by-LN. We denote the columns of M' as m'_t $(t = 1, 2, \ldots, L)$. Now, we denote M' as a vector-reshaped representation, namely,

$$P = vec(M') = [{m'_1}^T, {m'_2}^T, \ldots, {m'_L}^T]^{\mathrm{T}}. \tag{9}$$

We project the LN-by-1 vector P onto LN-dimensional space with a basis Ψ. In this case, the following equation holds:

$$P = \Psi S, \tag{10}$$

where S is the coefficient vector with respect to the basis Ψ.

B. TRAFFIC MATRIX RECOVERY MODEL

In our recovery model, we denote the traffic matrix M as a vectorization, that is,

$$vec(M) = \Psi vec(\Sigma_K \mathrm{U}_{PC}^T). \tag{11}$$

If we also represent Eq. 2 as a vectorization, then we have

$$P = vec(B) \odot vec(M), \tag{12}$$

where $vec(B) = [b_1^T, b_2^T, \ldots, b_L^T]^{\mathrm{T}}$ and b_t is the t-th column of B. Eq. 12 is equivalent to

$$P = D \cdot vec(M), \tag{13}$$

where D is an LN-by-LN diagonal matrix whose diagonal entries are the entries of $vec(B)$ in turn. Combining Eq. 12 with Eq. 13, we have

$$P = D\Psi vec(\Sigma_K \mathrm{U}_{PC}^T). \tag{14}$$

For similarity, we denote $vec(\Sigma_K \mathrm{U}_{PC}^T)$ as γ. Then, we build our Spatio-Temporal Kronecker Compressive Sensing model, as shown in Eq. 14. In this model, the sensing matrix produced by us is $D\Psi$. The challenge here is how to obtain the sparsifying basis Ψ. The PCA method in [6] first assumes that the previous TM is known, and then it reduces the ill-posed property of the TM inverse inference equation via the singular value decomposition of both the previous and the measured TM. There is an implicit assumption in [6]. It assumes that the principal components of the unknown TM that needs to be estimated are the same as those of the previous TM. Hence, in this paper, this assumption will be inherited. In other words, the matrix V can be calculated by using the previous and measured TM. Consequently, Ψ in Eq. 14 will be built according to Eq. 8. Then, the missing data will be recovered by solving the following optimization problem:

$$\hat{\gamma} = \arg\min \|\gamma\|_1 \quad s.t.\ D\Psi\gamma = P, \tag{15}$$

where $\|\cdot\|_1$ is the ℓ_1-norm. The computation complexity for solving Eq. 15 in terms of existing methods (such as basis pursuit and interior point methods) is $\mathrm{O}\left(L^3N^3\right)$. The recovery model in Eq. 15 cannot ensure the non-negativity constraints of the estimated TM. Hence, we use the Iterative Proportional Fitting algorithm [6] to recalibrate the estimated TM so it obeys the network tomography model. We now briefly present the pseudo-code of STKCS as shown in Table 1. Some TM estimation methods assume that the TM has zero mean, typically, the principal component analysis method. Hence, in our method, we first center the TM. Then we implement the singular value decomposition for the measured TM $\tilde{M}$. After that, a sparse representation of the TM is obtained according to its spatial property. Then a sparse representation of the TM with respect to the spatio-temporal property is determined by the Kronecker product. From the sparse representation, the missing data will be recovered by a convex optimization model.

TABLE 1. Spatio-temporal Kronecker compressive sensing method.

Algorithm: Spatio-Temporal Kronecker Compressive Sensing
Input: $\tilde{M}$, M
Output: $\hat{M}$
Initialize: K
1. Preprocess $\tilde{M}$ such that is has zero mean
2. $V \leftarrow SVD(\tilde{M})$
3. $V \leftarrow V(:, 1:K)$
4. $\Psi \leftarrow I_L \otimes V$
5. Record position of the missing data and build the matrix B
6. $D \leftarrow reshape(B)$
7. $P \leftarrow D \cdot vec(M)$
8. $\Theta \leftarrow D\Psi$
9. Solve $P{=}\Theta\gamma$ according to Eq. 15
10. $\hat{M} \leftarrow V\gamma$

V. SIMULATION RESULTS AND ANALYSIS

In this section, we evaluate the performance of our method via simulations using real traffic data. Two different data sets are used in our simulations. The two data sets have 1500 and 1000 time slots, respectively. The traffic is measured every 5 and 10 minutes, respectively. The first 500 time slots are used as the prior of TM to build the sparsifying basis in our method. Further, we compare the STKCS method with an existing interpolation algorithm, e.g., the Sparsity Regularized Matrix Factorization (SRMF). SRMF uses a structured nature to build a novel compressive sensing framework to recover the missing traffic data. We set the data loss probability to 50% for STKCS and SRMF. The regularization parameter for SRMF is 0.1 that is as same as the configurations in [15].

A. STABILITY OF SPARSIFYING BASIS

In our method, a prior TM is used to build the sparsifying basis V, and we assume that the sparsifying basis is stable for the TM with missing entries. This assumption has a remarkable effect on the estimation error of our method. Hence, we first evaluate the stability of the sparsifying basis V. We use 1000 time slots from the first traffic data set in this simulation, and the first 500 time slots are adopted to build the basis V. For the other 500 time slots, we set the data loss probability to 0, 20%, 40%, 60%, and 80%, respectively. We implement the singular value decomposition under these cases, and replace their sparsifying bases by V. When the data loss probability is 0, that is the traffic data is completed, the basis is stable [6]. In this case, if we replace its basis by V, and the error is acceptable. The temporal relative error (TRE) is involved as a metric in this simulation, which is defined as

$$TRE\left(t\right)=\frac{\left\|\hat{m}\left(n,t\right)-m\left(n,t\right)\right\|_{2}}{\left\|m\left(n,t\right)\right\|_{2}},\qquad(16)$$

where $m(n, t)$ and $\hat{m}(n, t)$ are the traffic entries with original basis and the entries from the replaced basis.

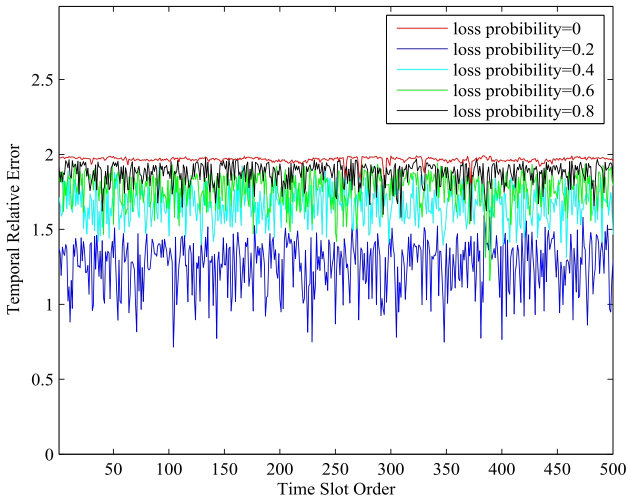


FIGURE 1. Stability of sparsifying basis.

Figure 1 plots the TREs of 5 cases. Obviously, when the loss probability is 20%, 40%, 60%, and 80%, their TREs are smaller than the case that the loss probability is 0. Thereby, the sparsfying basis is stable.

B. ESTIMATION ACCURACY

We assess the overall performance of our method by comparing each estimator with the real traffic data in Figs. 2 and 3. The y-axis denotes the real traffic entries, and the x-axis is indexed by the corresponding estimates.

For traffic data set 1 shown in Fig. 2, we see that STKCS underestimates the tremendous traffic entries. In contrast, considering the small traffic data, our method is outstanding with respect to unbiased estimation. Traffic data set 2 in Fig. 3 illustrates that STKCS is biased and unbiased for small and tremendous traffic entries, respectively.

C. TEMPORAL AND SPATIAL ERRORS

STKCS utilizes the spatio-temporal property of the TM to recover the missing traffic data. Therefore, in this subsection, we refer to the spatial relative error (SRE) and the TRE to validate the performance of STKCS. The SRE is defined as

$$SRE(n)=\frac{\left\|\hat{m}\left(n,t\right)-m\left(n,t\right)\right\|_{2}}{\left\|m\left(n,t\right)\right\|_{2}},\qquad(17)$$

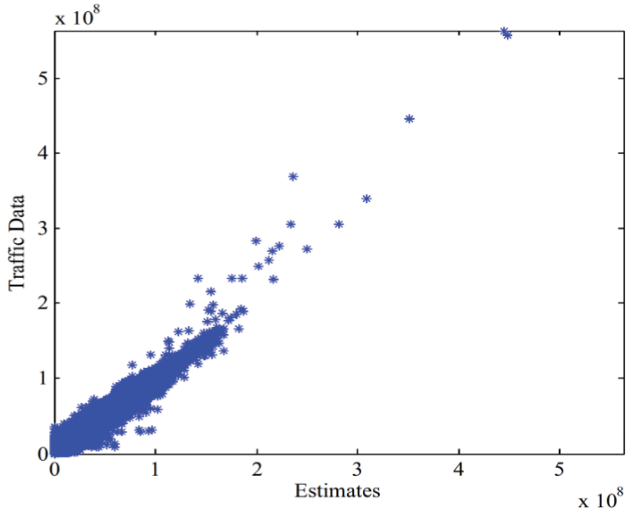


FIGURE 2. Real traffic data versus their estimates for traffic data set 1.

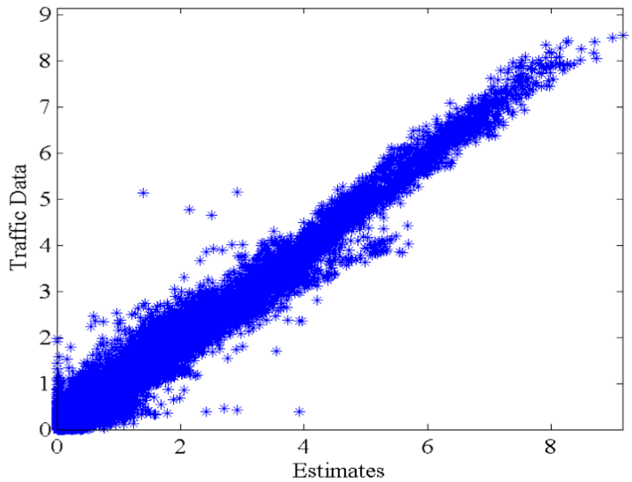


FIGURE 3. Real traffic data versus their estimates for traffic data set 2.

where $m(n, t)$ and $\hat{m}(n, t)$ are real traffic entries and their estimates, respectively. In this subsection, $m(n, t)$ and $\hat{m}(n, t)$ for the TRES shown by Eq. 16 have the same definitions in Eq. 17.

Figure 4 plots the SREs and TREs of STKCS and SRMF, respectively, for traffic data set 1. The x-axis of Fig. 4(a) expresses the identity (ID) of each OD pair. The y-axis is the SRE. All OD pairs are arranged in ascending order based on their averages. The red and blue lines denote STKCS and SRMF, respectively. For the small OD pairs, the SREs of SRMF show some large fluctuations. By contrast, the SREs of STKCS are significantly steady. Consequently, STKCS is much more stable, and able to estimate some small OD pairs. Figure 4(b) reveals the TREs of the two algorithms. The x-axis and y-axis are the time slot and TRE, respectively. Regarding the TRE shown in Fig. 4(b), we see that our method has a consistently low TRE compared with that of SRMF. Regarding traffic data set 2, as shown in Fig. 5(a),

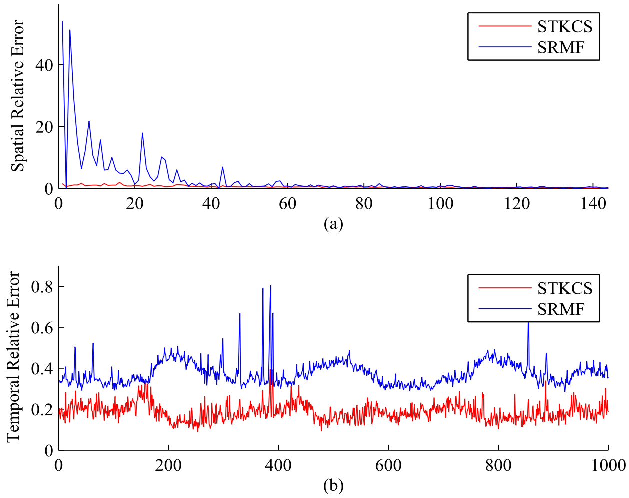


FIGURE 4. SRE and TRE of traffic data set 1. (a) Flow ID, from smallest to largest in mean. (b) Time slot order.

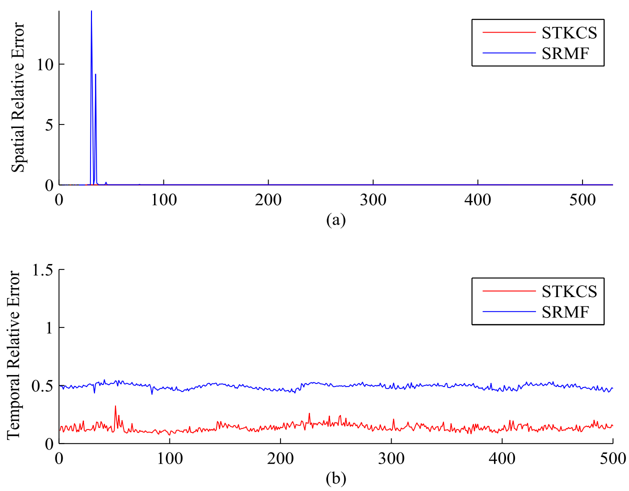


FIGURE 5. SRE and TRE of traffic data set 2. (a) Flow ID, from smallest to largest in mean. (b) Time slot order.

the TRE of SRMF has a huge jitter. Further, the TREs of STKCS are remarkably low. To analyze the SRE and TRE in depth, we include the cumulative distribution function in our simulations. Figure 6 indicates that our method outperforms the SRMF method with respect to SRE and TRE. Specifically, for approximately 80% of the OD pairs in traffic data set 1, the SREs of STKCS and SRMF are less than 0.77 and 2.67, respectively. Moreover, for 80% of the time slots, the TREs of the two algorithms are less than 0.22 and 0.42 in turn. Figure 7 shows that in traffic data set 2, the SREs of the two algorithms are less than 1.25 and 1.42 for approximately 70% of the OD pairs. The TREs are less than 0.16 and 0.51 for 80% of the time slots.

D. ESTIMATION BIAS

As shown in Figs. 2 and 3, our method appears to over- or underestimate more or less. Nevertheless, a biased

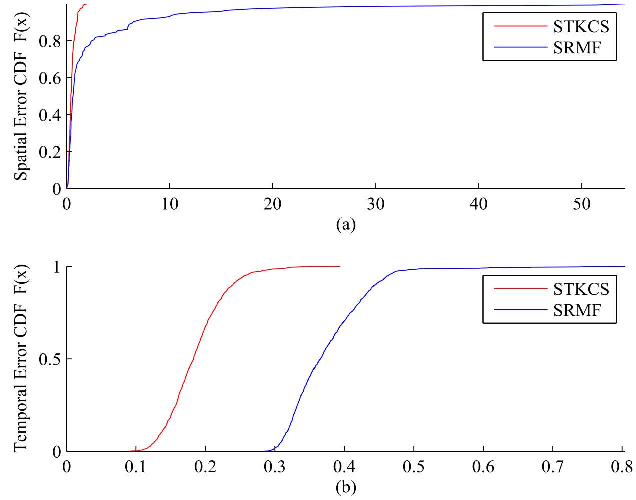


FIGURE 6. Cumulative distribution functions of SRE and TRE for traffic data set 1. (a) x = L2 norm, spatial error. (b) x = L2 norm, temporal error.

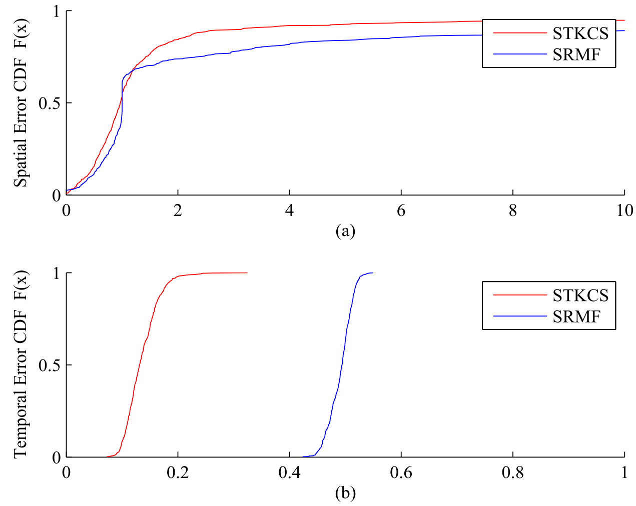


FIGURE 7. Cumulative distribution functions of SRE and TRE for traffic data set 2. (a) x = L2 norm, spatial error. (b) x = L2 norm, temporal error.

estimation is not necessarily bad [6]. Meanwhile, regarding an unbiased estimation, when it has a remarkably large variance, it may also be unavailable. Thus, we assess the bias and variance of the two algorithms in this subsection. Assessing the bias and variance of an estimation algorithm is useful for practical application. We here define the standard deviation as the metric of variance. The bias and standard deviation are denoted as follows:

$$\begin{cases} b(n) = \dfrac{\sum_{t=1}^{L} (\hat{m}(n,t) - m(n,t))}{L}, \\ s(n) = \sqrt{\dfrac{\sum_{t=1}^{L} (\hat{m}(n,t) - m(n,t) - b(n))^2}{L-1}}. \end{cases} \tag{18}$$

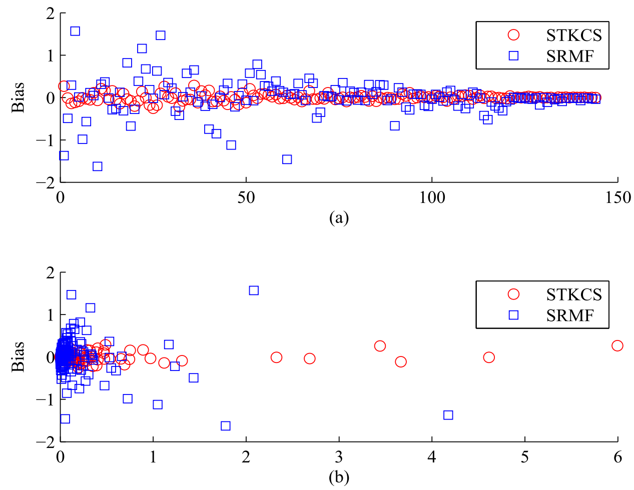


FIGURE 8. **Estimation bias and standard deviation for traffic data set 1. (a) Flow ID, from largest to smallest in mean. (b) Standard deviation in error.**

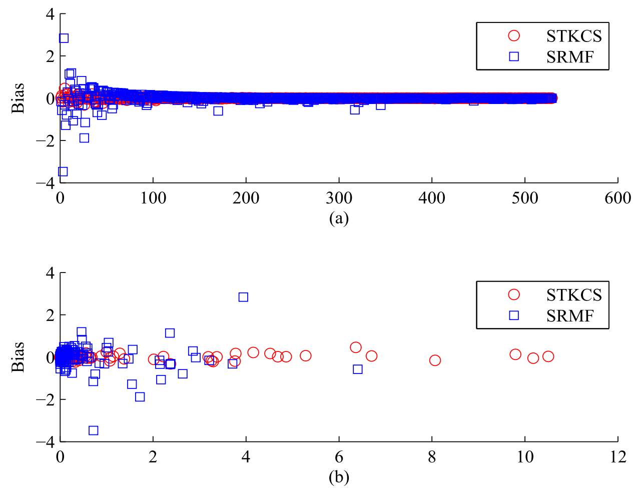


FIGURE 9. **Estimation bias and standard deviation for traffic data set 2. (a) Flow ID, from largest to smallest in mean. (b) Standard deviation in error.**

where L is the length of the collected traffic data. Figure 8(a) gives the bias of the two algorithms for traffic data set 1. The x-axis and y-axis denote the ID of each OD pair and the corresponding bias, respectively. Different from Figs. 4 and 5, all of the IDs in Fig. 8(a) are arranged in descending order based on their averages. Figure 8(a) states that the bias of SRMF is much larger than that of STKCS. Specifically, it is significantly explicit for the large OD pairs. In contrast, the bias of STKCS is relatively small for all OD pairs of traffic data set 1. We can obtain the same conclusion for traffic data set 2 via the simulation results shown in Fig. 9(a). Based on the bias of the two algorithms, we can reveal the bias of the two algorithms versus their standard deviation. Figures 8(b) and 9(b) indicate that our method has much larger standard deviation for few traffic entries. This means that STKCS tends to track the trend of an OD pair over long time intervals.

VI. CONCLUSION

In this paper, we propose an available interpolation algorithm to recover the missing entries of a traffic matrix. Our method, named Spatio-Temporal Kronecker Compressive Sensing, considers the spatial and temporal properties of traffic matrices simultaneously. Then, a sparse representation of traffic matrices is proposed based on the low-rank nature of traffic matrices. According to the sparse representation of a traffic matrix, we build an optimization model to recover the missing data of a traffic matrix. Finally, we assess the performance of STKCS by using two real traffic data sets. The simulation results declare that STKCS is outstanding for the estimation of error compared with previous methods.

ACKNOWLEDGMENT

The authors wish to thank the reviewers for their helpful comments.

REFERENCES

[1] Y. Tarutani, Y. Ohsita, S. Arakawa, and M. Murata, "Optical-layer traffic engineering with link load estimation for large-scale optical networks," *IEEE/OSA J. Opt. Commun. Netw.*, vol. 4, no. 1, pp. 38–52, Jan. 2012.

[2] Z. Hu, Y. Qiao, and J. Luo, "Coarse-grained traffic matrix estimation for data center networks," *Comput. Commun.*, vol. 65, no. 2, pp. 25–34, Feb. 2016.

[3] T. Toledo and T. Kolechkina, "Estimation of dynamic origin–destination matrices using linear assignment matrix approximations," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 2, pp. 618–626, Jun. 2013.

[4] M. Mardani and G. B. Giannakis, "Robust network traffic estimation via sparsity and low rank," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 4529–4533.

[5] K. Xu, M. Shen, Y. Cui, M. Ye, and Y. Zhong, "A model approach to the estimation of peer-to-peer traffic matrices," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 5, pp. 1101–1111, May 2014.

[6] A. Soule *et al.*, "Traffic matrices: Balancing measurements, inference and modeling," in *Proc. SIGMETRICS*, 2005, pp. 362–373.

[7] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Fast accurate computation of large-scale IP traffic matrices from link loads," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 31, no. 1, pp. 206–207, Jun. 2003.

[8] A. Soule, A. Nucci, R. L. Cruz, E. Leonardi, and N. Taft, "Estimating dynamic traffic matrices by using viable routing changes," *IEEE Trans. Netw.*, vol. 15, no. 3, pp. 485–498, Jun. 2007.

[9] T. O. Adelani and A. S. Alfa, "Hybrid techniques for large-scale IP traffic matrix estimation," in *Proc. IEEE Conf. Commun.*, May 2010, pp. 1–6.

[10] L. Tan and H. Zhou, "Tomofanout: A novel approach for large-scale IP traffic matrix estimation with excellent accuracy," *Ann. Telecommun.*, vol. 70, no. 3, pp. 149–158, Apr. 2015.

[11] H. Zhou, L. Tan, F. Ge, and S. Chan, "Traffic matrix estimation: Advanced-Tomogravity method based on a precise gravity model," *Int. J. Commun. Syst.*, vol. 28, no. 10, pp. 1709–1728, Jul. 2015.

[12] M. Polverini, A. Iacovazzi, A. Cianfrani, A. Baiocchi, and M. Listanti, "Traffic matrix estimation enhanced by SDNs nodes in real network topology," in *Proc. IEEE Conf. Comput. Commun. Workshops*, Apr./May 2015, pp. 300–305.

[13] Q. Zhang and T. Chu, "Structure regularized traffic monitoring model for traffic matrix estimation and anomaly detection," in *Proc. 34th Chin. Control Conf.*, Jul. 2015, pp. 4980–4985.

[14] Y. Zhang, M. Roughan, W. Willinger, and L. Qiu, "Spatio-temporal compressive sensing and Internet traffic matrices," in *Proc. ACM SIGCOMM Conf. Data Commun.*, 2009, pp. 267–278.

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

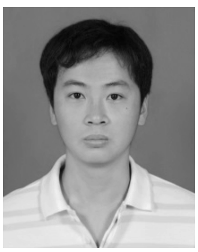
[16] K. Papagiannaki, N. Taft, and A. Lakhina, "A distributed approach to measure IP traffic matrices," in *Proc. 4th ACM SIGCOMM Conf. Internet Meas.*, 2004, pp. 161–174.

[17] P. Tune and M. Roughan, "Internet traffic matrices: A primer" *Recent Advances in Networking*, vol. 1, pp. 1–56, 2013.

[18] Z. Wang, K. Hu, K. Xu, B. Yin, and X. Dong, "Structural analysis of network traffic matrix via relaxed principal component pursuit," *Comput. Netw.*, vol. 56, no. 7, pp. 2049–2067, May 2012.
[19] E. Zhao and L. Tan, "A PCA based optimization approach for IP traffic matrix estimation," *J. Netw. Comput. Appl.*, vol. 57, pp. 12–20, Nov. 2015.
[20] H. Luo, Z. Chen, J. Cui, and H. Zhang, "An approach for efficient, accurate, and timely estimation of traffic matrices," in *Proc. IEEE Conf. Comput. Commun. Workshops*, Apr./May 2014, pp. 67–72.
[21] F. H. T. Vieira and L. L. Lee, "Adaptive wavelet-based multifractal model applied to the effective bandwidth estimation of network traffic flows," *IET Commun.*, vol. 3, no. 6, pp. 906–919, Jun. 2009.
[22] E. J. Candès, "Compressive sampling," in *Proc. Int. Congr. Math.*, vol. 3. Madrid, Spain, 2006, pp. 1433–1452.
[23] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
[24] M. F. Duarte and R. G. Baraniuk, "Kronecker compressive sensing," *IEEE Trans. Image Process.*, vol. 21, no. 2, pp. 494–504, Feb. 2012.

DINGDE JIANG (S'08–M'09) is a Professor with the College of Information Science and Engineering, Northeastern University, Shenyang, China.

He received the Ph.D. degree in communication and information systems from the University of Electronic Science and Technology of China, Chengdu, China, in 2009. From 2013 to 2014, he was a Visiting Scholar with the Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN, USA. His research focuses on network measurement, modeling and optimization, performance analysis, network management, network security in communication networks, particularly in software-defined networks, information-centric networking, energy-efficient networks, and cognitive networks. His research is supported by the National Science Foundation of China and the Program for New Century Excellent Talents in University of the Ministry of Education of China. He currently serves as an Editor for one international journal. He has served as a Technical Program Committee Member for several international conferences. He has received the best paper awards at international conferences.



LAISEN NIE is currently pursuing the Ph.D. degree in communication and information systems with the School of Computer Science and Engineering, Northeastern University, Shenyang, China. He is a Research Member with the Communication and Information Systems Institution, School of Computer Science and Engineering, Northeastern University. His research interests include network measurement and cognitive networks.

ZHIHAN LV received the Ph.D. degree in computer applied technology from the Ocean University of China, in 2012. He is an Engineer and Researcher of virtual/ augmented reality and multimedia major in mathematics and computer science, having plenty of work experience with respect to virtual reality and augmented reality projects, engaging in the application of computer visualization and computer vision. His research application fields widely range from everyday life to traditional research fields (i.e., geography, biology, and medicine). During the past years, he has finished several projects successfully with respect to PC, Website, Smartphone, and Smartglasses. Before that, he enjoyed 16 months of full-time research experience with the Centre National de la Recherche Scientifique, Paris (2010-2011). Afterwards, he fulfilled a two-year post-doctoral research experience with Umea University and was invited for a short teaching experience with the KTH Royal Institute of Technology, Sweden. Since 2012, he has held an Assistant Professor with the Chinese Academy of Science.

HOUBING SONG (M'12–SM'14) received the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, VA, in 2012.

In 2012, he joined the Department of Electrical and Computer Engineering, West Virginia University, Montgomery, WV, where he is currently an Assistant Professor and the Founding Director of both the Security and Optimization for Networked Globe Laboratory (SONG Laboratory), and the West Virginia Center of Excellence for Cyber- Physical Systems sponsored by the West Virginia Higher Education Policy Commission. In 2007, he was an Engineering Research Associate with the Texas A&M Transportation Institute. He is the Editor of four books, including *Cyber-Physical Systems: Foundations, Principles and Applications* (Waltham, MA, USA: Elsevier, 2016) and *Smart Cities: Foundations and Principles* (Hoboken, NJ, USA: Wiley, 2016). He has published more than 100 academic papers in peer-reviewed international journals and conferences. His research interests lie in the areas of cyber-physical systems, Internet of Things, cloud computing, big data analytics, connected vehicle, wireless communications and networking, and optical communications and networking.

Dr. Song is a member of ACM. He was a first recipient of the Golden Bear Scholar Award, the Highest Faculty Research Award at WVU. He is an Associate Editor for several international journals, and a Guest Editor of more than ten special issues. He was the General Chair of several international workshops, including the first IEEE International Workshop on Security and Privacy for Internet of Things and Cyber-Physical Systems (IOT/CPS-Security) within the IEEE ICC, London, U.K., 2015, and the first IEEE International Workshop on Big Data Analytics for Smart and Connected Health within the IEEE CHASE, in Washington, DC, 2016. He also served as the Technical Program Committee Chair of the fourth IEEE International Workshop on Cloud Computing Systems, Networks, and Applications (CCSNA), San Diego, USA, 2015. He is serving as the Publicity Chair of the 2017 IEEE Wireless Communications and Networking Conference (WCNC), San Francisco, CA, 2017, and the Technical Program Committee Chair of the fifth IEEE International Workshop on CCSNA, Washington, 2016. He has served on the Technical Program Committee for numerous international conferences, including ICC, GLOBECOM, and WCNC.