INTERNATIONAL JOURNAL OF COMMUNICATION SYSTEMS
Int. J. Commun. Syst. 2015; **28**:889–900
Published online 3 January 2014 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/dac.2713

A compressive sensing-based network tomography approach to estimating origin–destination flow traffic in large-scale backbone networks

Laisen Nie and Dingde Jiang*,†

College of Information Science and Engineering, Northeastern University, Shenyang, China

SUMMARY

A traffic matrix can exhibit the volume of network traffic from origin nodes to destination nodes. It is a critical input parameter to network management and traffic engineering, and thus it is necessary to obtain accurate traffic matrix estimates. Network tomography method is widely used to reconstruct end-to-end network traffic from link loads and routing matrix in a large-scale Internet protocol backbone networks. However, it is a significant challenge because solving network tomography model is an ill-posed and under-constrained inverse problem. Compressive sensing reconstruction algorithms have been well known as efficient and precise approaches to deal with the under-constrained inference problem. Hence, in this paper, we propose a compressive sensing-based network traffic reconstruction algorithm. Taking into account the constraints in compressive sensing theory, we propose an approach for constructing a novel network tomography model that obeys the constraints of compressive sensing. In the proposed network tomography model, a framework of measurement matrix according to routing matrix is proposed. To obtain optimal traffic matrix estimates, we propose an iteration algorithm to solve the proposed model. Numerical results demonstrate that our method is able to pursuit the trace of each origin–destination flow faithfully.

Received 1 September 2013; Revised 31 October 2013; Accepted 10 November 2013

KEY WORDS: end-to-end network traffic; origin–destination flows; compressive sensing; singular value decomposition; traffic reconstruction

1. INTRODUCTION

A traffic matrix can describe end-to-end network traffic that reflects the network-level performance of communication networks from origin–destination (OD) nodes. Network operators are able to know the status of traffic flows through their network from the traffic matrix [1, 2]. With the development of the current networks, it is too hard to obtain accurate traffic matrices by direct measurement [1]. Thereby, the researchers exploit traffic matrix estimation approaches by using statistical inference techniques.

Network tomography method was first brought in by Vardi [2–10] to estimate OD flows from link loads and routing matrix. In this method, network traffic is modeled as Poisson distribution, and the method traces the shape of each flow by maximum likelihood. The network tomography problem can be denoted briefly as

$$Y = A \times Z, \tag{1}$$

*Correspondence to: Dingde Jiang, College of Information Science and Engineering, Northeastern University, Shenyang 110819, China.
†E-mail: jiangdingde@ise.neu.edu.cn

where matrices Y, A, and Z denote link loads, routing matrix, and traffic matrix, respectively. Each OD flow is a time series described by a row of traffic matrix Z. Each entry of traffic matrix Z denotes traffic volume of each OD flow. For OD nodes, the node type can affect the granularity of the traffic matrix. The node type consists of link-to-link, router-to-router and PoP-to-PoP (Point of Presence), and so on [7]. In this paper, we address the PoP-to-PoP traffic matrix of the large-scale Internet protocol (IP) backbone network. The routing matrix A, which is a deterministic matrix, is made up of entries 1 and 0. Equation (1) states a linear relationship between link loads and traffic matrix. Figure 1 depicts the framework of the network tomography method, in which one can achieve link loads Y easily from simple network management protocol (SNMP) [7]. SNMP is widely employed in IP networks for network management. In the SNMP system, a cyclic counter records the number of bytes passed on each interface. The recorded link load data is collected in the SNMP management information-base data. Then one can obtain the link load data by an SNMP poller that can circularly request the appropriate SNMP management information-based data. Moreover, routing matrix can be obtained from status information and configuration files of the network [1, 7]. Meanwhile, one assumes that the routing matrix A is static unless the network topology changes [7]. Unfortunately, it is extremely difficult to solve this inference problem, because network tomography is a highly ill-posed and under-constrained inverse problem. In other words, the number of links is already much smaller than that of OD pairs in the large-scale IP backbone networks. Without loss of generality, if the number of nodes is q in a network, then the number of OD flows is $N = q^2$. We assume that the number of links is M, and then $M < N$ for a general network [7]. Motivated by that issue, it has attracted lots of interests, and many research communities pursued more efficient approaches to overcome this ill-posed nature [1–9].

In this context, much work based on network tomography method has emerged because it was proposed, such as Tomogravity method [7], principal component analysis (PCA) method [8], fan out method and Kalman filtering method [1], and so on. These methods successfully recover the flow size by modeling the OD flows or measuring the partial OD flows. Nevertheless, statistical model is sensitive to prior information of traffic matrix. In addition, current communication networks are larger than before. Besides, kinds of communication networks (e.g., 3G and public switched telephone network) and different multi-media servers (e.g., voice over IP and peer-to-peer) are involved in our networks. As a result, our networks have become much more complex and heterogeneous [1, 2, 6]. In other words, we can't use the previous statistical models such as Gaussian model and Poisson model [3–6] as prior information. Hence, we have to exploit another approach to deal with this fundamental under-constrained inference problem.

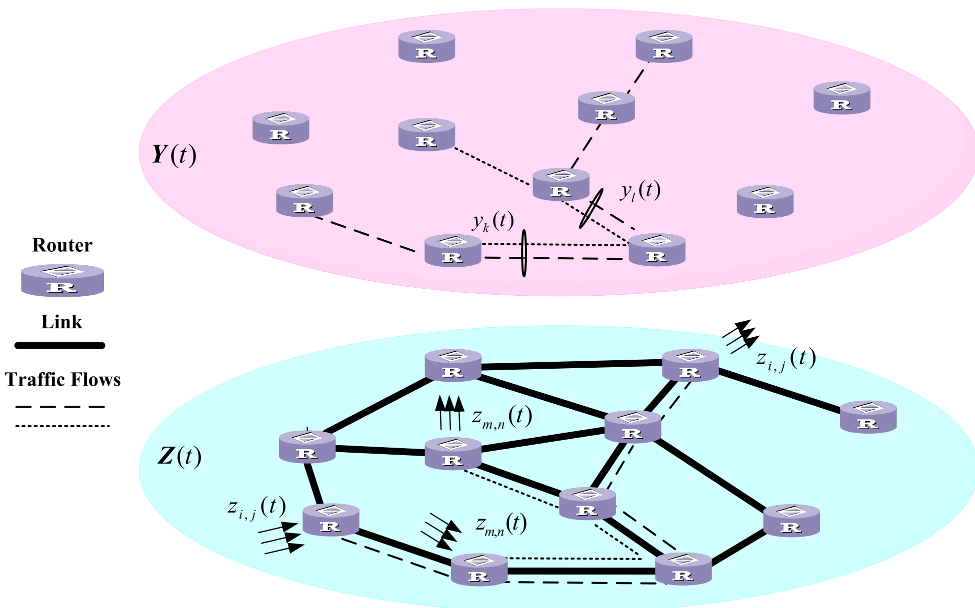

Figure 1. Network tomography framework.

DOI: 10.1002/dac

In this paper, we exploit a novel method that draws on ideas from 'compressive sensing (CS)' [11–22] and 'singular value decomposition (SVD)' [8] to cope with the ill-posed and under-constrained problem. As a new and critical theory, CS, which was proposed firstly in 2006 by E. Candes, J. Romberg, D. Donoho, and T. Tao, has been extensively used to find the optimal solution of an under-constrained linear system [11–15]. So far, it has not paid enough attention to solve network tomography model shown in Equation (1) by way of CS reconstruction algorithms except Zhang's work in [19], though CS has been involved in computer networks [20, 21]. The existing CS construction algorithms often assume the following: (i) the sensing matrix needs to follow the restricted isometry property (RIP) [13] and (ii) the transform coefficients must be sparse [11–14]. Unfortunately, network tomography model does not obey either of the two conditions. To overcome these issues, Zhang *et al.* proposed a novel CS construction named sparsity regularized matrix factorization [19] that can track the shape of each flow faithfully.

By modifying the traditional network tomography model, we put forward an approach to build a new network tomography model so that it obeys the constraints of CS. At first, we generate a sparsity basis by means of SVD to make the traffic matrix sparse. After that, according to routing matrix, we propose a framework of measurement matrix following the constraints of CS. In order to gain small estimation error, we also provide an iteration algorithm to tackle the new network tomography model.

The rest of this paper is organized as follows. In section 2, we will introduce the CS theory, and then we bring forward the algorithm to build our network tomography model. We will give numerical results to validate the performance of the new model in Section 3. Finally, we conclude our work in Section 4.

2. PROBLEM STATEMENT AND OUR METHODOLOGY

2.1. *Compressive sensing and under-constrained inverse problem*

As a new sampling theory, CS depicts that a signal can be reconstructed *via* incomplete measurements [11, 12]. Assume that for an N-dimensional signal x, the recovery of this signal from M measurements can be represented as

$$y = \Psi x. \tag{2}$$

The vector y is measurements and Ψ is a measurement matrix. An efficient approach to solve Equation (2) is ℓ_1-minimization problem, that is

$$\hat{x} = \arg\min \ \|x\|_1 \quad s.t. \quad y = \Psi x, \tag{3}$$

which is a convex optimization problem.

To recover x accurately, there are two crucial constraints [23]:

(1) The signal x must be sparse, which means that most of the entries in x are zero. In particular, if it just has $k << N$ nonzero entries, and then it is usually called k-sparse.
(2) In order to reconstruct a k-sparse signal, the measurement matrix Ψ needs to meet the following inequalities for any $3k$-sparse vector e:

$$(1-\delta_{3k})\|e\|_2 \le \|\Psi e\|_2 \le (1+\delta_{3k})\|e\|_2, \tag{4}$$

where $\delta_k \in (0,\ 1/3)$ is restricted isometry constant and $\|\cdot\|_2$ means the ℓ_2-norm. If previous inequalities hold, measurement matrix Ψ can obey the RIP.

In the previous text, constraints are useful for accurate recovery in CS theory. However, the signals are not completely sparse in fact, and it is significantly difficult to show whether a measurement matrix obeys the RIP or not. A simple approach to judge the RIP is so-called concentration inequality [15]. The concentration inequality has a unitary invariant nature. If we assume that $x = U\alpha$, where U is an orthogonal basis and α is the transform efficient vector under this basis,

DOI: 10.1002/dac

and then $\Psi x = \Psi U\alpha$. According to this underlying unitary invariant nature, matrix Ψ meets RIP that is equivalent to saying that ΨU also has the RIP. Hence, though a signal is non-sparse in time domain, one may map it onto another transform domain in order to attain a sparse representation of this signal.

Previous work provided numerical approaches to build measurement matrix. The Gaussian and Bernoulli matrices provide optimal conditions for the minimal number of required samples for sparse reconstruction [15]. For a Gaussian measurement matrix of which entries are independent and identically distributed (i.i.d), it needs $M \geq Ck\log(N/k)$ measurements at least to accurate recovery where C is a small constant.

2.2. *Our methodology*

Compressive sensing reconstruction algorithms are regarded as efficient and precise means to find out the single solution from an ill-posed and under-constrained inference problem. On the other hand, because traffic matrices are not sparse and routing matrices are deterministic, CS reconstruction algorithms are not able to tackle the inference problem described by Equation (1). To overcome the previous difficulties, we try to construct a modified network tomography model that can be solved by CS reconstruction algorithms.

Here, we provide the system model of our algorithm to plot our method in Figure 2. As shown in Figure 2, we extend the classical network tomography model by means of adding a random matrix and carrying out a certain matrix decomposition. The destination of the matrix decomposition is to describe our traffic matrices sparsely, that is, the transform coefficient vector is sparse. Finally, we recover end-to-end network traffic by solving the transform coefficients from the modified network tomography model. The specific approaches which are also the key contributions of this paper are as follows:

- Based on routing matrix, to satisfy the RIP, we put forward a framework of measurement matrix denoted as $\Psi = GC(\gamma)A$, which will be depicted in Section 2.2.2.
- We bring forward an iteration algorithm to solve our reconstruction model.

2.2.1. Sparsity basis. First of all, traffic matrix is not completely sparse, but it still exhibits an intensively low-rank property, that is, the certain K principal components can represent the traffic matrix approximately. In other words, partial OD flows occupy the most of the energy of the traffic matrix. The size of energy can be represented by the singular values of traffic matrix. These properties allow us to construct an orthogonal basis and make traffic matrix sparse by the SVD and low-rank approximation.

We first introduce the SVD, and then we present the method to build the sparse basis. If we assume that the traffic matrix Z is N by T, where N is the number of OD flows and T denotes the length of time series, then its SVD can be formulated into:

$$Z^T = U\Sigma V^T, \tag{5}$$

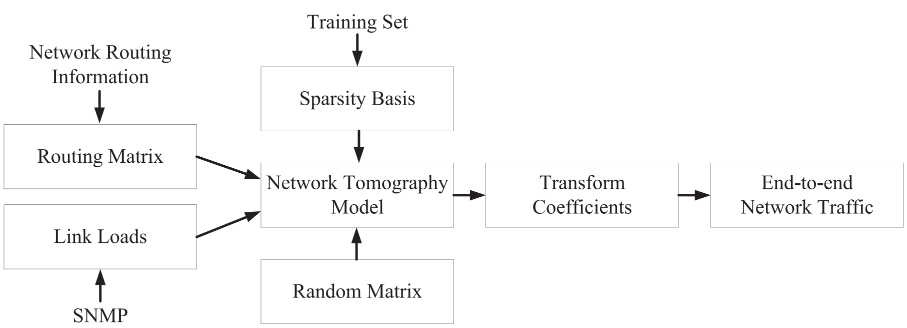

Figure 2. System model used for end-to-end network traffic reconstruction.

DOI: 10.1002/dac

where the $N \times N$ matrix V is made up of the eigenvectors of matrix ZZ^T, $(\cdot)^T$ denotes transposition, and U is a $T \times N$ matrix. Σ is an $N \times N$ diagonal matrix whose diagonal entries are singular values of Z^T. Equation (5) is also represented as:

$$Z^T = U\Sigma V^T = \sum_{r=1}^{R} \sigma_r u_r v_r^T, \tag{6}$$

where the elements of $\{\sigma_r\}$ are singular values of Z^T, u_r, and v_r are columns of matrices U and V, respectively. R is the rank of matrix Z^T. Singular values exhibit the energy of each principal component v_r, thereby we can describe the traffic matrix approximately using partial principal components corresponding to large singular values. It is denoted by mathematical formula

$$Z^T \approx \tilde{Z}^T = \tilde{U}\tilde{\Sigma}\tilde{V}^T = \sum_{r=1}^{K} \sigma_r u_r v_r^T \tag{7}$$

with $K < R$. Previous equation takes advantage of K principal components to represent traffic matrix approximately.

In our method, the prior measurements of traffic matrix will be used as a training set to construct the sparsity basis. We assume that prior traffic data is denoted by Z', then its SVD can be formulated into:

$$\begin{cases} Z'^T = U' \Sigma' V'^T \\ Z'^T \approx \sum_{r=1}^{K} \sigma'_r u'_r v_r'^T = \tilde{U}'\tilde{\Sigma}'\tilde{V}'^T \end{cases}. \tag{8}$$

In this case, let the number of principal components be K, then the rank of traffic matrix Z' is K. Equation (8) is equivalent to

$$Z' = \tilde{V}'\tilde{\Sigma}'\tilde{U}'^T. \tag{9}$$

In particular, because matrix $\tilde{V}'$is made up of the eigenvectors corresponding to different eigenvalues of matrix $Z'Z'^T$, it is an orthogonal matrix. Thereby, we have attained a sparsity basis $\tilde{V}'$ from Equation (8). If the training set Z' is an $N \times T'$ matrix with $N < T'$, then the matrices $\tilde{V}'$, $\tilde{\Sigma}'$, and $\tilde{U}'^T$ are $N \times N$, $N \times N$, and $N \times T'$, respectively. Obviously, only K diagonal entries of matrix $\tilde{\Sigma}'$are nonzero, hence each column of transform coefficient matrix $\tilde{\Sigma}'\tilde{U}'^T$ is K-sparse.

If we calculate the SVD of traffic matrix, Z^T, which needs recovering in terms of Equation (1), the following equation holds:

$$Y = AZ = A\tilde{V}\tilde{\Sigma}\tilde{U}^T. \tag{10}$$

And then we use $\tilde{V}'$*via* training set to replace $\widetilde{V}$ approximately, then Equation (10) is equivalent to

$$Y = AZ = A\tilde{V}'\tilde{\Sigma}\tilde{U}^T. \tag{11}$$

By this way, an orthogonal basis is constructed, which can make sure that each column of $\widetilde{\Sigma}\widetilde{U}^T$ is K-sparse. There is a hidden assumption in the SVD method, that is, the training set (i.e., the prior measurements of traffic matrix) is available and inerrable. In this paper, we sustain this assumption as before.

2.2.2. Random measurement matrix. Our approach starts with Equation (1), and we obtain Equation (11) by the SVD of training set. As a deterministic matrix, it is difficult to make sure that it obeys RIP. Consequently, according to routing matrix, we propose a framework of measurement matrix denoted by $\Psi = G\ C(\gamma)\ A$, where

- A is the $M \times N$ routing matrix.

DOI: 10.1002/dac

- G is a $P \times M$ Gaussian random matrix whose entries are i.i.d asymptotically normally distributed $N(0, O(M^{-1}))$, where $O(M^{-1})$ means that $a_M = O(M^{-1})$ is on the order of $O(M^{-1})$, namely, $\lim_{M\to\infty} \frac{a_M}{M^{-1}}$ is a constant.
- $C(\gamma)$ is a diagonal matrix whose diagonal entries are 1 or 0. The number of zero entries is γ. In our framework, $C(\gamma)$ is a sampling matrix in practice. It is used to delete γ rows of routing matrix. In detail, $C(\gamma)$ can delete γ rows of routing matrix with γ-largest ℓ_1-norms.

Then Equation (11) can be converted into

$$L = GC(\gamma)\ \ Y = GC(\gamma)A\tilde{V}'\tilde{\Sigma}\tilde{U}^T = \Psi\tilde{V}'S, \tag{12}$$

where $S = \tilde{\Sigma}\tilde{U}^T$ and $L = G\ C(\gamma)\ Y$. According to Equation (12), the problem is defined as that one needs to recover S when L, Ψ and $\tilde{V}'$are known. According to CS theory, the matrix S can be recovered provided that Ψ obeys the RIP.

2.2.3. Stochastic independence analysis. The main idea of CS theory consists of a set of stochastic and independent sampling strategies and the sparsity of the recovered signal. The stochastic and independent sampling is a related condition of the RIP [11–15]. In this context, the Gaussian matrix whose entries are i.i.d is popular as the measurement matrix in practice. In our framework, the deterministic routing matrix A and the entries in measurement matrix Ψ are not independent and identically distributed at all.

At first, we analyze the coherence between the entries of the measurement matrix Ψ without the matrix $C(\gamma)$. For simplicity, we assume $\Omega = GA$. To describe the coherence, we denote each entry of the matrix Ω by $\omega_{i,j}$ that obeys $N(0, O(M^{-1}))$. The correlation coefficient between $\omega_{i,j}$ and $\omega_{i,n}$ is

$$\rho\left(\omega_{i,j}\ ,\ \omega_{i,n}\right) = \frac{< A_j\ ,\ A_n >}{\|A_j\|_2 \|A_n\|_2}, \tag{13}$$

where A_j (A_n) denotes the *jth* (*nth*) column of the routing matrix A and $<\cdot>$ denotes the inner product. Equation (12) illustrates that the coherence between different entries in the same row of the matrix Ω depends on the relevant columns of the routing matrix A. Likewise, one can see that the correlation coefficients between the entries in different rows of the matrix Ω are 0. Therefore, to generate an i.i.d random matrix, we have to make each element of the correlation coefficient set $\{< A_j\ ,\ A_n >/(\|A_j\|_2\|A_n\|_2)\}$ $(j \neq n,\ j, n = 1, 2, \ldots, N)$ as small as possible. A simple approach is deleting the large elements of the correlation coefficient set. Consequently, we refer to $C(\gamma)$ in our measurement matrix. The diagonal matrix $C(\gamma)$ is employed in this paper as a sampling matrix to delete partial rows in the routing matrix A and decrease the coherence. If γ is sufficiently large, and then the coherence is really small. At that time, the entries of matrix Ψ can be viewed as i.i.d Gaussian variables. In other words, the matrix Ψ can satisfy the RIP with overwhelming probability. Hence, the matrix $\Psi\tilde{V}'$ will obey the RIP because of the unitary invariant nature of the RIP. Such an approximately i.i.d assumption can cause much more estimation error. Motivated by that, we propose an efficient and exact algorithm in order to achieve small estimation error.

2.2.4. End-to-end network traffic reconstruction. According to our modified network tomography model in Equation (12), in the CS theory, we can recover the transform coefficient vector s_t by solving convex optimization (e.g., ℓ_1-minimization) problem at time t

$$\hat{s}_t = \arg\min\ \|s_t\|_1 \quad s.t. \quad \Psi\tilde{V}'s_t = l_t, \tag{14}$$

where l_t and s_t are the column vectors of L and S, respectively. Generally, we usually make use of basis pursuit and greedy algorithm (e.g., match pursuit algorithm and orthogonal match pursuit algorithm) to solve Equation (14) [18]. By iteration computation, we can obtain the estimation results $\hat{S}$. Afterwards, we can achieve the estimates of traffic matrix $\hat{Z} = \tilde{V}'\hat{S}$. However, ℓ_1-minimization

DOI: 10.1002/dac

problem is unable to make sure that the solutions are nonnegative. Taking account of the nonnegative nature of a traffic matrix, we set the negative entries to zero, and utilize iterative proportional fitting algorithm [1] to calibrate the estimation results. In our method, the prior measurements of traffic matrix are used to build sparsity basis at first. Because of the dynamic nature of the traffic matrix, the sparsity basis can change over time and therefore the sparsity basis will be refined as well. We bring in the relative root mean squared error as a metric of estimation error, which is defined as

$$RRMSE(t) = \frac{\sqrt{\sum_{i=1}^{N} \left(y_{i,t} - \hat{y}_{i,t}\right)^2}}{\sqrt{\sum_{i=1}^{N} {y_{i,t}}^2}}, \tag{15}$$

where $y_{i,t}$ and $\hat{y}_{i,t}$ are real link loads and its estimation result from traffic estimator, respectively. Once the relative root mean squared error is greater than the trigger threshold, the sparsity basis will be recalibrated.

Up to now, we have formulated our end-to-end network traffic estimation method in detail. As is described in previous paragraph, the following will demonstrate the pseudo-codes of our method:

Algorithm 1 CS-OMP algorithm

Input: γ, K, trigger threshold and the length of prior traffic data T'.
Output: Traffic matrix $\hat{\boldsymbol{Z}}$.
Initialize: $\hat{\boldsymbol{Z}}_0 = []$ (empty matrix).

1. Generate a $P \times M$ random measurement matrix $\boldsymbol{G}$ in terms of distribution $N(0, \mathrm{O}(M^{-1}))$;
2. Build matrix $\boldsymbol{C}(\gamma)$ according to the routing matrix $\boldsymbol{A}$;
3. Calculate $\boldsymbol{L}$ according to Equation (12);
4. Obtain sparsity basis $\tilde{\boldsymbol{V}}'$ by the SVD of the prior traffic data with length T';
5. Compute the measurement matrix $\Psi = \boldsymbol{G}\,\boldsymbol{C}(\gamma)\,\boldsymbol{A}$;
6. for $t = 1$ to T do
    1) Recover the transform coefficient vector $\hat{\boldsymbol{s}}_t$ from $\boldsymbol{l}_t$ by the OMP algorithm with K iterations;
    2) Calibrate estimates $\hat{\boldsymbol{z}}_t = \tilde{\boldsymbol{V}}\hat{\boldsymbol{s}}_t$ by iterative proportional fitting algorithm;
    3) $\hat{\boldsymbol{Z}}_t = [\hat{\boldsymbol{Z}}_{t-1}, \hat{\boldsymbol{z}}_t]$;
    4) if RRMSE > trigger threshold
        Recalibrate sparsity basis according to previous T' estimation results and compute $\hat{\boldsymbol{z}}_t$ again.
       end if
   end for

return $\hat{\boldsymbol{Z}} = \hat{\boldsymbol{Z}}_t$

3. SIMULATION RESULT AND ANALYSIS

In this section, we provide numerical results to validate the performance of our algorithm by using real network traffic data from the Abilene network. The Abilene consists of 12 routers, 30 internal links, and 24 external links and is used for education and research. The data is sampled by NetFlow

DOI: 10.1002/dac

in the 5-min interval, and 2016-point traffic data from 1 week-long experiment is used to testify our algorithm. We compare our algorithm (named CS-OMP) with the PCA method. Traffic matric estimation by PCA supposes that $\tilde{V}$ and $\tilde{\Sigma}$ in the model $Y = A\,\tilde{V}\tilde{\Sigma}\tilde{U}^T$ shown by Equation (10) are known and stable. They can be achieved by a training set. Furthermore, because of the intrinsic dimensionality of the traffic matrix, only the principal components are retained, that is, the small singular values and the corresponding rows of $\tilde{U}^T$ are deleted. In this context, Equation (10) is a well-posed estimation model. Then traffic matrix can be estimated by the pseudo-inverse of $A\,\tilde{V}\tilde{\Sigma}$. Both CS-OMP and PCA use a training set to build the reconstruction model, and draw on idea from 'low rank' [8, 24]. Thus, we will compare them in our simulations. We use the first 500-point data as prior traffic data to build sparsity basis (i.e., $T' = 500$). The same training set is employed in the PCA method. As is known from the previous work in [8], the hundreds of OD flows from both networks can be accurately described by 5–10 principal components. Hence, we set $K = 10$ in the CS-OMP and PCA. Besides, we set $\gamma = 10$, and let the trigger threshold be 0.001 empirically.

We first validate the ability of our method to trace the behavior of OD flows over time. We select two OD flows arbitrarily to analyze the performance of our method. In Figure 3, it depicts OD 63 and its estimates *via* our algorithm and the PCA method. We find that both CS-OMP and PCA can track the changes of OD 63 over time. The PCA method has some fluctuations during time slot 800–1000. During time slot 1700–1900, PCA even can't estimate the trend of OD 63. Figure 4 plots OD 108 and its estimates. The real data have a good deal of fluctuations round time slot 800. Our method does not be confused by this, and it is able to estimate these fluctuations properly. Nevertheless, the PCA method yields underestimation for them.

We assess two algorithms directly and intuitively by testing OD flow in the previous text, and now we will analyze these algorithms quantitatively. Another metric is employed to evaluate the performance of our method adequately. The spatial relative errors (SREs) and temporal relative errors (TREs) [1] are defined as

$$SRE(i) = \frac{\left\|\hat{z}_{i,t} - z_{i,t}\right\|_2}{\left\|z_{i,t}\right\|_2}, \tag{16}$$

$$TRE(t) = \frac{\left\|\hat{z}_{i,t} - z_{i,t}\right\|_2}{\left\|z_{i,t}\right\|_2}, \tag{17}$$

where $i = 1, 2, \ldots, N$ and $t = 1, 2, \ldots, T$ denote the index of OD flows and time slot, respectively.

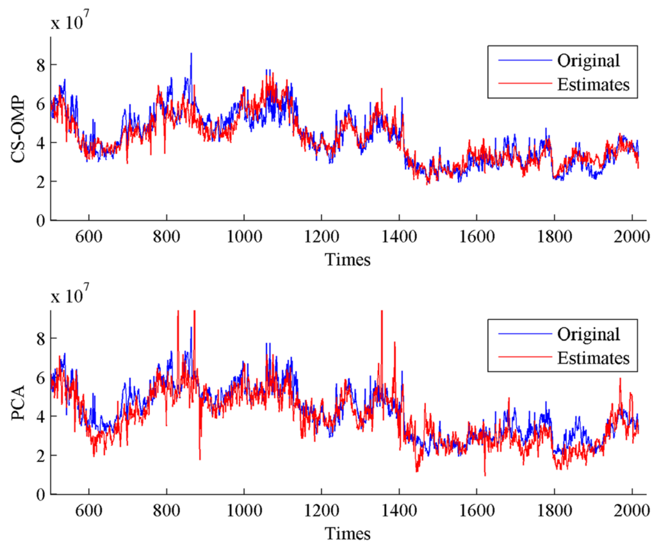

Figure 3. Origin–destination 63 and its estimation result.

Figure 5(a) plots the SREs of two algorithms, and it shows that the SREs decrease as the mean of OD flows descend. The reason is that the large traffic data is much easier to be estimated and vice versa. In detail, CS-OMP and PCA have small estimation errors intuitively. The average SREs of the CS-OMP and PCA methods are 0.39 and 0.59, respectively. Obviously, CS-OMP has smaller SREs. For TREs showing in Figure 5(b), the average TREs of the CS-OMP and PCA methods are 0.15 and 0.27. In our simulations, the cumulative distribution functions of SREs and TREs are utilized to validate our algorithm. In Figure 6(a), it states that about 97% of OD flows arrive at SREs = 0.89 for our method. Likewise, it is 90% for the PCA method. Consider the TREs, for about 90% of measurement moments, the TREs of the CS-OMP and PCA methods are 0.18 and 0.35, respectively.

We have analyzed the performance of our algorithm in detail. Figures 3 and 4 demonstrate that two algorithms yield underestimation or overestimation more or less; however, none of the previous metrics can describe it. We here will use the bias of estimation results to verify, which is denoted as

$$bias(i) = \frac{1}{T}\sum_{t=1}^{T}\left(\hat{z}_{i,t} - z_{i,t}\right). \tag{18}$$

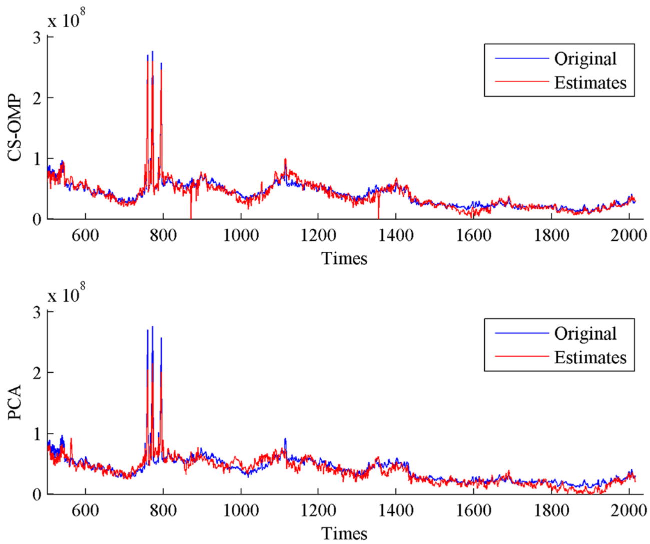

Figure 4. Origin–destination 108 and its estimation result.

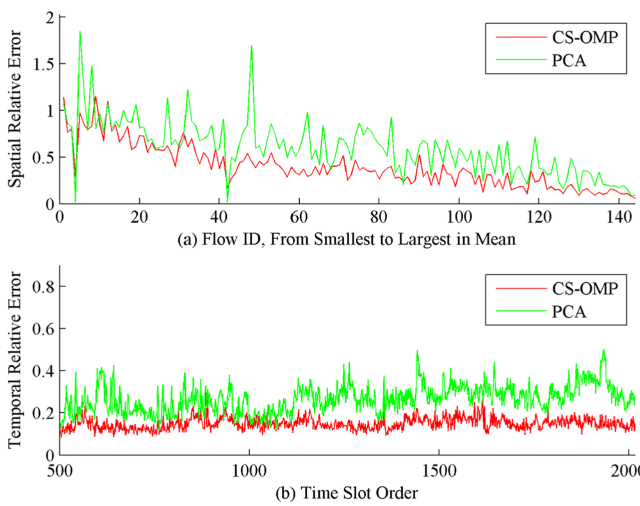

Figure 5. Spatial relative errors and temporal relative errors in Abilene.

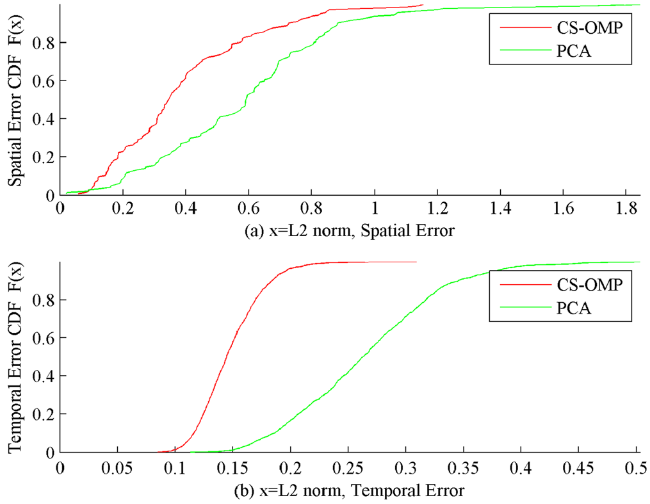

Figure 6. Cumulative distribution functions of spatial relative errors and temporal relative errors in Abilene.

In Figure 7, it shows that the estimation bias of large OD flows is much smaller. We see that the bias of CS-OMP is smaller than that of the PCA method.

As is known, for a variable, it may have large variance as usual though its bias is small. Consequently, using bias as the unique metric is not exact enough. For this reason, we combine bias and standard deviation together to assess our algorithm. The sample standard deviation is defined as

$$sd(i) = \sqrt{\frac{1}{T-1}\sum_{t=1}^{T}\left(err(i) - bias(i)\right)^2}, \tag{19}$$

where $err(i) = \hat{z}_{i,t} - z_{i,t}$.

Figure 8 plots the bias for each of OD flows against their standard deviation. From Figure 8, we see that the bias of CS-OMP is small, but partial estimation results *via* CS-OMP have larger standard deviation. Inversely, PCA exhibits underestimation or overestimation constantly, but its standard deviation is smaller.

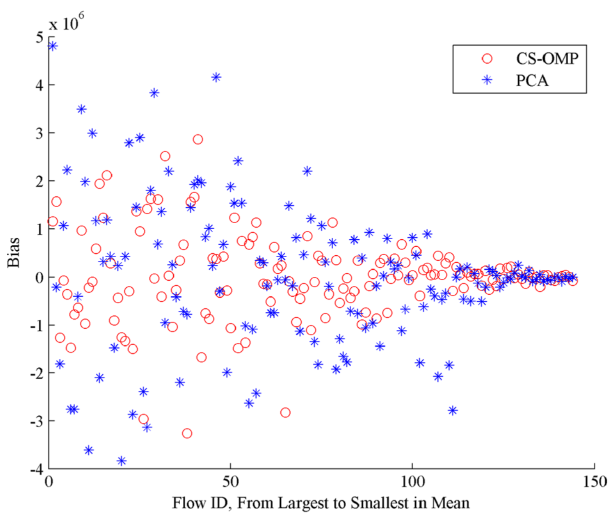

Figure 7. Bias of estimation results.

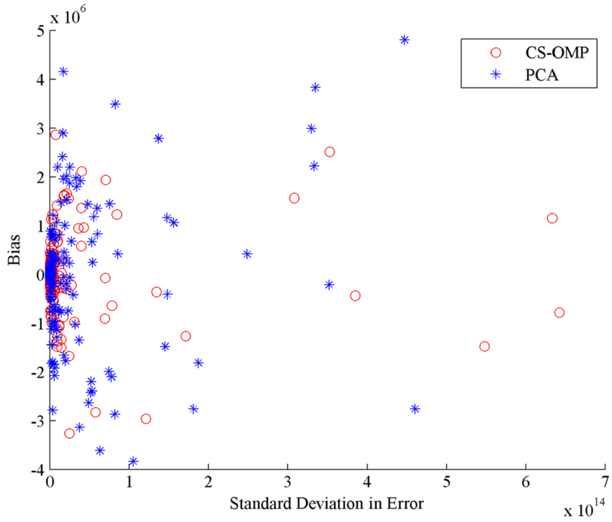

Figure 8. Bias versus standard deviation.

We will analyze the computational complexity of the PCA and CS-OMP methods. We just give the complexity of the estimation result during a time slot. According to the parameters set in the previous text, the computational complexity of SVD is $O(T'N^2)$. In the proposed algorithm, the complexity of the OMP algorithm for estimating a column of the traffic matric is $O(N[\log N]^2)$. If we infer the well-posed problem in the PCA method by the least square technology, the computational complexity will be $O(N^3)$. Obviously, our model is much more efficient in the reconstruction procedure, though we add a stochastic perturbation in our model.

4. CONCLUSIONS

In this paper, we address the problem of end-to-end network traffic reconstruction based on network tomography in the large-scale IP backbone networks. We study CS-based reconstruction algorithm to solve network tomography problem in order to attain end-to-end network traffic precisely. Limited by the technology constraints of the CS, we can't solve network traffic by CS reconstruction algorithms directly. Hence, we propose an algorithm to construct a novel model such that this model satisfies the constraints of CS. In our method, we build a sparse basis to ensure the sparsity of the traffic matrix. Afterwards, we also propose a framework of measurement matrix in terms of routing matrix. By the proposed iteration algorithm, we can obtain the estimates of end-to-end network traffic. Finally, we provide numerical results to assess the performance of our method. Simulation results show that our algorithm can improve the performance of end-to-end network traffic estimation observably.

ACKNOWLEDGEMENTS

This work was supported in part by the National Natural Science Foundation of China (Nos. 61071124, 61172051), the Specialized Research Fund for the Doctoral Program of Higher Education (No. 20100042120035), the Program for New Century Excellent Talents in University (No. NCET-11-0075), the Fundamental Research Funds for the Central Universities (Nos. N120804004, N110404001). The authors wish to thank the reviewers for their helpful comments.

REFERENCES

1. Soule A, Lakhina A, Taft N, Papagiannaki K, Salamatian K, Nucci A, Crovella M, Diot C. Traffic matrices: balancing measurements, inference and modeling, *Proc. SIGMETRICS*, Banff, Canada, June 2005.
2. Vardi Y. Network tomography: estimating source-destination traffic intensities from link data. *Journal of American Statistics Association* 1996; **91**(433):365–377.

DOI: 10.1002/dac

3. Jin X, Min G, Velentzas S, Jiang J. Quality-of-service analysis of queuing systems with long-range-dependent network traffic and variable service capacity. *IEEE Transactions on Wireless Communications* 2012; **11**(2):562–570.
4. Yu M, Daut D. A new traffic model for backbone networks and its application to performance analysis. *International Journal of Communication Systems* 2008; **21**(6):587–608.
5. Jiang D, Xu Z, Zhang P, Zhu T. A transform domain-based anomaly detection approach to network-wide traffic. *Journal of Network and Computer Applications* 2013; published online.
6. Yin H, Lin C, Sebastien B, Li B, Min G. Network traffic prediction based on a new time series model. *International Journal of Communication Systems* 2005; **18**(8):711–729.
7. Zhang Y, Roughan M, Duffield N, Greenberg A. Fast accurate computation of large-scale IP traffic matrices from link loads. *Proceedings of ACM SIGMETRICS* 2003; **31**(1):206–217.
8. Lakhina A, Papagiannaki K, Crovella M, Diot C, Kolaczyk E, Taft N. Structural analysis of network traffic flows, *Proc. SIGMETRICS*, 2004; 61–72.
9. Yen N, Huang R, Ma J, Jin Q, Shin T. Intelligent route generation: discovery and search of correlation between shared resources. *International Journal of Communication Systems* 2013; **26**(6):732–746.
10. Nie L, Jiang D, Guo L. A power laws-based reconstruction approach to end-to-end network traffic. *Journal of Network and Computer Applications* 2013; **36**(2):898–890.
11. Donoho D. Compressive sensing. *IEEE Transactions on Information Theory* 2006; **52**(4):1289–1306.
12. Engelberg S. Compressive sampling. *IEEE Instrumentation and Measurement Magazine* 2012; **15**(1):42–46.
13. Candès E, Romberg J, Tao T. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory* 2006; **54**(2):489–509.
14. Candès E. The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Mathematique* 2008; **346**(9–10):589–592.
15. Fornasier M, Rauhut H. *Compressive sensing, Handbook of Mathematical Methods in Imaging*, 2011; 187–228.
16. Atia G, Saligrama V. Boolean compressed sensing and noisy group testing. *IEEE Transactions on Information Theory* 2012; **58**(3):1880–1901.
17. Haupt J, Bajwa W, Rabbat M, Nowak R. Compressed sensing for networked data. *IEEE Signal Processing Magazine* 2008; **25**(2):92–101.
18. Tropp J, Gilbert A. Signal recovery from random measurements *via* orthogonal matching pursuit. *IEEE Transactions on Information Theory* 2007; **53**(12):4655–4666.
19. Zhang Y, Roughan M, Willinger W, Qiu L. Spatio-temporal compressive sensing and Internet traffic matrices, *Proc. SIGCOMM*, 2009; 267–278.
20. Feizi S, Medard M, Effros M. Compressive sensing over networks, *Proc. Annual Allerton Conference on Communication, Control, and Computing*, 2010; 1129–1136.
21. Xu W, Mallada E, Tang A. Compressive sensing over graph, *Proc. INFOCOM*, 2011; 2087–2095.
22. Do T, Gan L, Nguyen N, Tran T. Fast and efficient compressive sensing using structurally random matrices. *IEEE Transactions on Signal Processing* 2012; **60**(1):139–154.
23. Baraniuk R. Compressive sensing. *IEEE Signal Processing Magazine* 2007; **24**(4):118–121.
24. Xiao H, Lamare R. Reduced-rank space-time adaptive interference suppression for navigation systems based on QR decomposition and approximations of basis functions. *International Journal of Communication Systems* 2012; **25**(5):616–635.

DOI: 10.1002/dac