DENOISING NETWORK TOMOGRAPHY ESTIMATIONS

Muhammad H. Raza, Bill Robertson, William J. Phillips
Department of Engineering Mathematics and Internetworking, Dalhousie University, Nova Scotia, Halifax, Canada

Jacek Ilow
Department of Electrical Engineering, Dalhousie University, Nova Scotia, Halifax, Canada

Keywords: Network tomography, Sparse code shrinkage, Error modeling, Link delays.

Abstract: In this paper, we apply the technique of sparse shrinkage coding (SCS) to denoise the network tomography model with errors. SCS is used in the field of image recognition for denoising of the image data and we are the first one to apply this technique for estimating error free link delays from erroneous link delay data. To make SCS properly adoptable in network tomography, we have made some changes in the SCS technique such as the use of Non Negative Matrix Factorization (NNMF) instead of ICA for the purpose of estimating sparsifying transformation. Our technique does not need the knowledge of the routing matrix which is assumed known in conventional tomography. The estimated error free link delays are compared with the original error free link delays based on the data obtained from a laboratory test bed. The simulation results reveal that denoising of the tomography data has been carried out successfully by applying SCS.

1 INTRODUCTION

Computer networks have emerged as the primary setup for communication in present global scenario. With a broad range of applications on the networks using diverse technologies, there is a growing need to better understand and characterize the network dynamics. High quality traffic measurements are a key to successful network management. Direct observation of the desired statistics in a network is not possible without the special cooperation of the internal network resources. For example, routers do not maintain per user or per flow information, but performance metrics such as loss or utilization statistics are available at router interfaces (Zhao et al., 2006),(Coates and Nowak, 2001).

Cooperation to obtain internal information from privately owned networks is almost impossible to get. The network communication research community has always been looking for the alternatives to get around this problem. However, some useful parameters can be obtained from passive monitoring of traffic or active probing of a network. The desired statistics (that needs internal cooperation of private networks) are then indirectly estimated from these directly measured statistics (requiring no internal cooperation). Also, for the diverse nature of network applications of today, the service providers need differential measurements such as individual link performance to avoid congestion and keep the service level agreements (Zhao et al., 2006), (Coates and Nowak, 2001). The phenomenon of estimating desired statistics indirectly from directly measured parameters is called network tomography. The simplest model of network tomography is represented by the following equation,

$$Y = AX, \tag{1}$$

linking the measured parameters matrix (Y) with the matrix of unknown parameters (X) with dependence on the routing matrix (A) of the network. If Y has I rows and X has J rows, then the size of the routing matrix (A) is I$\times$J. The rows of A (A_i) correspond to paths from the sender to the receivers and the columns (A_j) correspond to individual links in those paths.

In reality, all the practical networks have the potential of errors that should be reflected in the network tomographic model as Y = AX+ε, where ε represents the error in the model.

There are various sources that contribute towards the error term (ε) such as Simple Network Management Protocol (SNMP) operation and NetFlow measurements. The heterogeneity of the network components in terms of vendors and hardware/software plat-

H. Raza M., Robertson B., J. Phillips W. and Ilow J. (2010).
DENOISING NETWORK TOMOGRAPHY ESTIMATIONS.
In *Proceedings of the International Conference on Data Communication Networking and Optical Communication Systems*, pages 67-72
DOI: 10.5220/0002972800670072

forms that are used by various types of networking technologies, is also a contributing factor toward the error term, ε.

In this paper, we have applied SCS technique to denoise the noisy link delay data. A key idea that constitutes the rationale behind sparse code shrinkage (SCS) is to use a basis that is more suitable for data at hand. For denoising, it is required to transform data to a sparse code, apply maximum likelihood (ML) estimation procedure component-wise, and transform back to the original variables. The simulation results show that the proposed technique needs less input and assumptions to denoise and recover almost noise free (original) data.

The rest of the paper is organized as follows. Section 2 briefly describes network tomography and various factors that introduce errors in tomography data. Section 3 reviews related work. Section 4 discusses SCS and the rationale for using SCS. Section 5 explains application of NNMF in the context of network tomography and sparsity. Section 6 presents and discusses results to show that SCS successfully denoises the noisy link delay data with out *a priori* knowledge of routing matrix. Section 7 concludes the paper.

2 FACTORS INTRODUCING ERRORS IN NETWORK TOMOGRAPHY

Vardi (Vardi, 1996) was the first one to introduce the term of network tomography for an indirect inference of desired statistics. Three categories of network tomography problems (active, passive, and topology identification) have been addressed in the literature. In passive network tomography (Vardi, 1996), link level statistics such as bit rate are passively measured as matrix Y and origin destination (OD) flows are estimated as X.

In active network tomography (Castro et al., 2004), (Coates and Nowak, 2001), unicast or multicast probes are sent from a single or multiple source(s) to destination(s) and parameters such as packet loss rate (PLR), delay and bandwidth are determined from source destination measurements.

The key idea in most of the existing topology identification methods is to collect measurements at pairs of receivers (Castro et al., 2004).

Simple Network Management Protocol (SNMP) and NetFlow are the main contributors towards the error term (ε) along with the heterogeneity of the network components in terms of vendors and hardware/software platforms that are used by various types of networking technologies.

SNMP is applied for collecting data that is used for management purposes including network delay tomography. SNMP (Zhao et al., 2006) periodically polls statistics such as byte count of each link in an IP network. In SNMP, the commonly adopted sampling interval is 5 minutes. The management station cannot start the management information base (MIB) polling for hundreds of the router interfaces in a network at the same time (at the beginning of the 5-minutes sampling intervals). Therefore, the actual polling interval is shifted and could be different than 5 minutes. This polling discrepancy becomes a source of error in SNMP measurements.

The traffic flow statistics are measured at each ingress node via NetFlow (Clemm, 2006), (Systems, 2010). A flow is an unidirectional sequence of packets between a particular source and destination IP address pair. The high cost of deployment limits the NetFlow capable routers. Also, products from vendors other than Cisco have limited or no support at all for NetFlow (Clemm, 2006), (Systems, 2010). Therefore, sampling is a common technique to reduce the overhead of detailed flow level measurement. The flow statistics are computed after applying sampling at both levels; packet level and flow level. Since the sampling rates are often low, inference from the NetFlow data may be noisy.

Both SNMP and NetFlow use the user datagram protocol (UDP) as the transport protocol. The operating nature of UDP may add to the error term of the model due to hardware or software problem resulting in data loss in transit (Zhao et al., 2006), (Clemm, 2006), (Systems, 2010).

Having different vendors for network components along with hardware/software platforms that are used by various types of networking technologies and the inherited shortcomings of the distributed computing also introduce errors. The risk of errors increases if there are more components in a system. The physical and time separation and consistency is also a problem and a source of error (Zhao et al., 2006).

The next section describes related work and distinguishes our contribution from the related work.

3 REVIEW OF RELATED WORK

The authors of (Zhao et al., 2006), on their way to estimate traffic matrix with imperfect information, have mentioned the presence of errors in network measurements. But, they did not present any solution in particular to the errors in link measurements. Though they have considered these errors when they have compared the traffic matrix with and with out network measurement errors.

A traffic matrix quantifies aggregate traffic volume between any origin/destination (OD) pairs in a network, which is essential for efficient network provisioning and traffic engineering.

They have applied statistical signal processing techniques to correlate the data obtained from both (SNMP and NetFLow) measurement infrastructures. They have determined traffic under the passive tomography by considering a bi-model approach for error modeling. As they have used one model for the SNMP errors and another model for NetFlow errors. They have also categorized errors in various categories such as erroneous data and dirty data. We, on the other hand, have used a single model to represent noise irrespective of the nature of noise source as shown in Equation 2. Our model is simpler as it considers all the errors as a single collective parameter, ε, irrespective of the sources that have caused these errors. Though we have collected data for our simulations by active tomography, our method could be applied to any type of tomographic data.

As described in Section 2, various kinds of sources introduce errors in the original data and the use of this data for making further estimation can multiply the errors. There is need for a techniques that may denoise this data and SCS is one of such techniques. A brief description of SCS is given in the next section.

4 SPARSE CODE SHRINKAGE (SCS)

SCS (Hyvarinen, 1999) exploits the statistical properties of data to be denoised. To explain the SCS model, assume that we observe a noisy version $\widetilde{X}$ = x+ν of the data x, where ν is Gaussian White Noise (WGN) vector. To denoise $\widetilde{X}$,

1. we transform the data to a sparse code,
2. apply ML estimation procedure component-wise,
3. transform back to the original variables.

Following are the steps involved:

1. Using a noise-free training set of x, use a sparse coding method for determining the orthogonal matrix W so that the components s_i in s = Wx have as sparse distributions as possible. Originally, SCS uses ICA in (Hyvarinen, 1999) for the estimation of the sparsifying transformation. There are various other ways to implement BSS such as Principal Component Analysis (PCA) and Singular Value Decomposition (SVD). In this paper, we use Non Negative Matrix Factorization (NNMF) instead of ICA for this purpose. The ICA approach may result in negative values in estimated matrices whereas all the involved components in NNMF are always positive and the same is true for link delays. NNMF is briefly explained in the next section.
2. Estimate a density model $p_i(s_i)$ for each sparse component, using the following two models:
   - Model 1: the first model is suitable for supergaussian densities that are not sparser than the Laplace distribution, and is given by the family of densities:
     $$p(s) = Cexp(\frac{-as^2}{2} - b|s|) \tag{2}$$
     where a, b > 0 are parameters to be estimated, and C is an irrelevant scaling constant. A simple method for estimating a and b was given in (Hyvarinen, 1999). For this density, the nonlinearity g takes the form:
     $$g(u) = 1/(1+\sigma^2 a)sign(u)max(0,|u|-\sigma^2) \tag{3}$$
     where σ^2 is the noise variance.
   - Model 2: this model describes densities that are sparser than the Laplace density:
     $$p(s) = \frac{1}{2d}\frac{(\alpha+2)[\frac{\alpha(\alpha+1)}{2}]^{(\frac{\alpha}{2}+1)}}{[\sqrt{\frac{\alpha(\alpha+1)}{2}}+|\frac{s}{d}|]^{\alpha+3}} \tag{4}$$
     When $\alpha \rightarrow$infinity, the Laplace density is obtained as the limit. A simple consistent method for estimating the parameters d, $\alpha > 0$ can be obtained from the relations d = $\sqrt{Es^2}$ and α = $\frac{(2-k+\sqrt{K(K+4)})}{(2k-1)}$. The resulting shrinkage function can be obtained as below:
     $$U = \frac{1}{2}\sqrt{(|u|+ad)^2 - 4\sigma^2(\alpha+3)} \tag{5}$$
     $$g(u) = sign(u)max(0, \frac{|u|-ad}{2} + U) \tag{6}$$
     Where a = $\sqrt{\frac{\alpha(\alpha+1)}{2}}$ and g(u) is a set of zeros in case the square root in the above equation is imaginary. Compute for each noisy observation $\widetilde{X}$(t) of X, the corresponding sparse component. Apply the shrinkage no-linearity g_i(.) as defined in the above equations for g(u) on each component y_i(t) for every observation

index t. Denote the obtained component by $S_i(t) = g_i(y_i(t))$.

3. Invert the relationship, s=Wx, to obtain estimates of the noise free X, given by $\widetilde{x}(t) = W\widetilde{X}(t)$.

To estimate the sparsifying transform W, an access to a noise-free realization of the underlying random vector is assumed. This assumption is not unrealistic in many applications: for example, in image denoising it simply means that we can observe noise free images that are somewhat similar to the noisy image to be treated, i.e., they belong to the same environment or context. In terms of link delays in networking it means having link delay readings while a system is operating in normal condition with no abnormalities to cause errors.

5 NON NEGATIVE MATRIX FACTORIZATION (NNMF)

Non Negative Matrix Factorization (NNMF) is one of the implementations of Blind Source Separation (BSS). If a non negative matrix V is given, then the NNMF finds non-negative matrix factors W and H such that (Cichocki et al., 2009):

$$V \approx WH \tag{7}$$

To find an approximate factorization, a cost function is defined that quantifies the quality of the approximation. Such a cost function can be constructed using some measure of distance between two non negative matrices, A and B. One popular cost function is simply the square of the Euclidean distance between A and B,

$$\|A - B\|^2 = \sum (A_{ij} - B_{ij})^2 \tag{8}$$

and another is based on divergence,

$$D(A\|B) = \sum_{ij} (A_{ij} log \frac{CA_{ij}}{B_{ij}} - A_{ij} + B_{ij}) \tag{9}$$

For each cost function, there are rules for updating W and H after selecting initial values of W and H. At each iteration W and H are multiplied and $\|$V-WH $\|^2$ or D(V $\|$ WH) is calculated. The values of W and H are updated until $\|$V-WH $\|^2$ or D(V $\|$ WH) reach a minimum threshold. At this moment, the values of W and H represent the final estimate.

5.1 Sparsity with NNMF

A useful property of NNMF is the ability to produce a sparse representation of data. Such a representation encodes much of the data using a few active components, which makes the encoding easy to interpret. On theoretical grounds, sparse coding is considered useful middle ground between completely distributed representations on one hand and unary representations on the other (Cichocki et al., 2009). In terms of network terminology, a highly sparse network means using a fewer links out of the total number of links available in a network and low sparse network means closer to the original topology of a network. As the feature of sparsity plays a significant role in SCS, so NNMF has been considered for the estimation of the sparsifying transformation in the initial step of SCS.

6 SIMULATION RESULTS OF DENOISING TOMOGRAPHY DATA THROUGH SCS

For validating SCS as a technique to denoise the erroneous link delays, we designed a test bed to collect real link delays. We introduced WGN into the measured link delays to create the affect of errors in the measured link delays. We input this erroneous data to SCS and denoised this data to get an estimate of the link delays close to the measured link delays. The next subsection describes the test bed that was used for data collection to obtain end to end delays and link delays for bench marking.

6.1 Description of Networking Test Bed

We set up a test bed in the Advanced Internetworking Laboratory (AIL) at Dalhousie University that consists of six 38 series Cisco routers, Agilent Router Tester (N2X), and a Multi Router Traffic Grapher (MRTG) capable workstation. OSPF routing has been implemented on routers and N2X.

The test bed is of smaller size and has limited number of links, because we have to collect the actual values of the error free link delays for bench marking the accuracy of estimated link delays. As no related work is available to bench mark our novel contribution, the original link delays remains the only choice for bench marking. In contrast to this test bed, the practical networks are larger in scale, but scalability is not an issue as SCS (Hyvarinen, 1999) and NNMF (Cichocki et al., 2009) both can handle larger sizes of matrices.

The Echopath option of the Cisco Service Level Agreement (CSLA) was implemented. All probes were grouped together. All the probes in the group start at the same time. The group of probes was repeated 100 times with a time difference of 10 sec between two consecutive repetitions. The results of 200 runs were averaged. The MRTG enabled workstation verified the end to end RTT.

Figure 1 shows a test bed with the four probes (traveling from right to left) and two of the links (Link1 and Link6) were stressed with an extended ping of 200 Bytes. The other source of disturbance was the traffic from the Agilent router tester (N2X). The condition of the network remains unchanged during the CSLA operation.

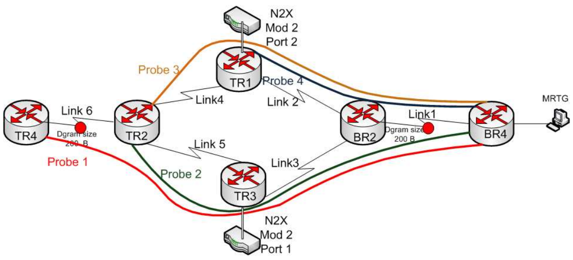

Figure 1: Testbed Setup with a mixture of extended pings and N2X traffic.

6.2 Use of Data from Test Bed

The data obtained from the CSLA is in the form of accumulative hop-wise round trip time, the following steps are followed to process the data for obtaining two matrices; a matrix of end to end delays and a matrix of link level delays.

A parsing software, written in java, extracts link delays and end to end delays in the form of two matrices. From the accumulative round trip time from source to each hop, hop to hop delays are calculated to form the delay matrix. From the accumulative round trip time (from the source to the destination), end to end delay matrix is determined. This data has been used as a baseline for judging the accuracy of the SCS.

The WGN was simulated through a Matlab based function and measured link delays were converted into the noisy link delays. This noisy data was used as an input to SCS. We expected SCS to denoise this noisy data in such a way that the denoised link delays are closer to measured link delays.

As part of the SCS, we needed to apply a BSS technique as a sparse coding method for determining the orthogonal matrix W so that the components s_i in s = Wx have as sparse distributions as possible. We applied NNMF for this purpose. The end to end link delays obtained from CSLA were input to NNMF. The Matlab tool NMFpack (Hoyer, 2004) had been used for NNMF factorization . The NMFpck Matlab package implements and tests NNMF with the feature of sparsity. Various combinations of measured link delays and the routing matrix with various sparsity levels were tried to get s_i as sparse as possible. These sparse estimation of s_i were input to step 2 of the implementation of SCS as described Section 3.

6.3 Comparison of Measured, Errored, and Denoised Link Delays

The results have been displayed in six diagrams (Figure 2 to Figure 7). Each diagram representing one link, from Link1 to Link6. In each diagram, three types of data lines are shown:

1. the actual measurement of the link delays collected from CSLA is shown as solid lines in graphs,
2. the link delays after the introduction of the error are shown as the dotted lines,
3. the denoised link delays after the application of SCS are shown as dashed lines.

The vertical axis represents the link delays and horizontal axis is the number of samples at various times.

It is clear from these six graphs that the denoised link delays are very close to the actual link delays. The errored link delays were input to SCS and the estimated (denoised) values of link delays are close to the measured values. This shows that the SCS has successfully denoised the noisy link delay data and the denoised data is in the proximity of benchmarks.

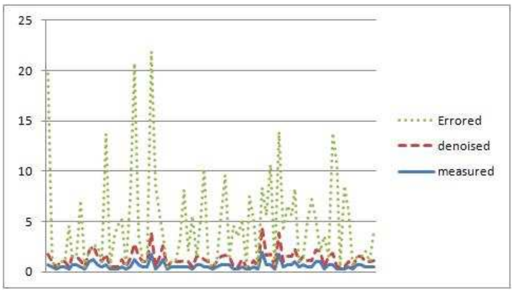

Figure 2: Comparison of measured, errored, and denoised link delays on Link1.

7 CONCLUSIONS

High quality traffic measurements are a key to successful network management. Direct observation of the desired statistics in a network is not possible without the special cooperation of the internal network resources. Network tomography facilitates indirect estimation of the desired network parameters. Various sources introduce errors in the estimated parame-

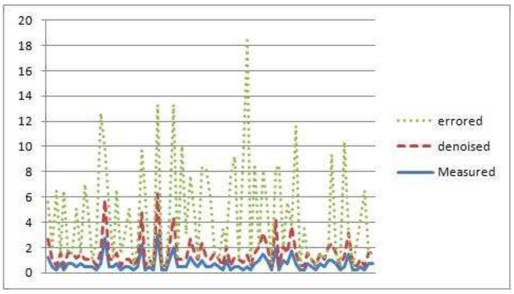

Figure 3: Comparison of measured, errored, and denoised link delays on Link2.

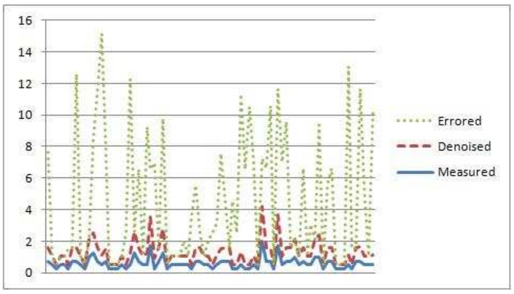

Figure 4: Comparison of measured, errored, and denoised link delays on Link3.

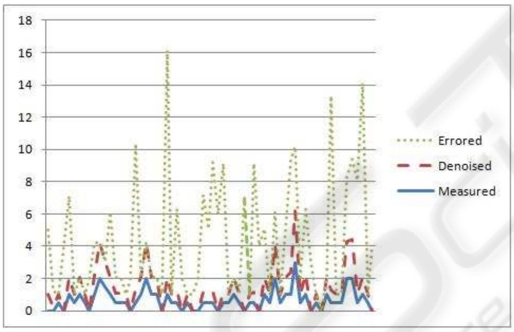

Figure 5: Comparison of measured, errored, and denoised link delays on Link4.

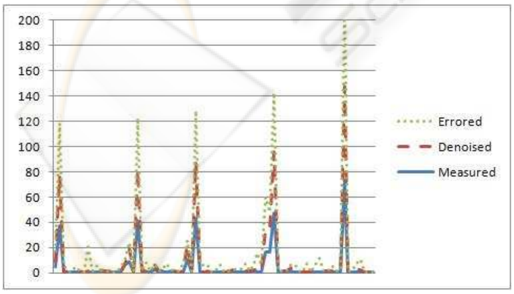

Figure 6: Comparison of measured, errored, and denoised link delays on Link5.

ters and reduce the effectiveness of the estimated parameters. We applied the technique of sparse shrinkage coding (SCS) to denoise the network tomography

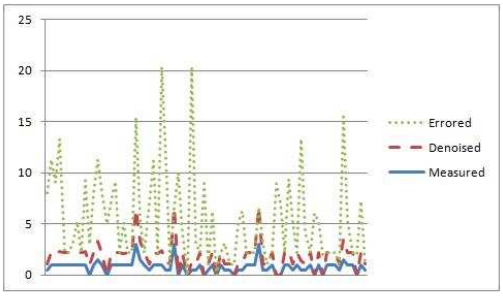

Figure 7: Comparison of measured, errored, and denoised link delays on Link6.

model with errors. To fit well to our research objectives, we modified SCS by replacing ICA with NNMF to get all the positive values in the estimated link delay matrices. The results obtained from the laboratory test bed based simulations proved that SCS successfully denoised the link delays. The comparison of denoised link delays with the error free benchmark data showed them very close to each other.

REFERENCES

Castro, R., Coates, M., Liang, G., Nowak, R., and Yu, B. (2004). Network tomography: Recent developments. In *Statistical Science*. Volume 9, 499–517.

Cichocki, A., Zdunek, R., Phan, A. H., , and ichi Amari, S. (2009). *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*. Wiley.

Clemm, A. (2006). *Network management fundamentals, ACM SIGMETRICS Performance Evaluation Review*. Cisco Press.

Coates, M. and Nowak, R. (2001). Network tomography for internal delay estimation. In *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing, Proceedings (ICASSP'01)*. IEEE.

Hoyer, P. (2004). Non-negative matrix factorization with sparseness constraints. In *The Journal of Machine Learning Research*. MIT Press Cambridge, MA, USA, Volume 5.

Hyvarinen, A. (1999). Sparse code shrinkage: Denoising of nongaussian data by maximum likelihood estimation. In *Neural Computation*. Volume 11, 1739–1768, MIT Press.

Systems, C. (2010). *NetFlow Services Solutions Guide*. available at www.cisco.com.

Vardi, Y. (1996). Network tomography: Estimating source-destination traffic intensities from link data. In *Journal of the American Statistical Association*. Volume 91, 365–377.

Zhao, Q., Ge, Z., Wang, J., and Xu, J. (2006). Robust traffic matrix estimation with imperfect information: Making use of multiple data sources. In *ACM SIGMETRICS Performance Evaluation Review*. Volume 34, 144.