Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 32 (2014) 1030 – 1036
Procedia Computer Science

2nd International Workshop on Survivable and Robust Optical Networks (IWSRON)

Indirect Estimation of Link Delays by Directly Observing a Triplet of Network Metrics

Muhammad H. Raza*, Ali Nafarieh, Bill Robertson

Internetworking Program, Faculty of Engineering, Dalhousie University, Halifax, B3H 2Z9, Canada

Abstract

This paper presents an improved indirect estimation link delays from a triplet of network metrics; path delays, packet loss rate (PLR), and jitter by using indirect inverse modeling techniques. conventionally a network metric is estimated by directly observing another network parameter. Based on the evidence in the literature that path delays, PLR, and jitter are interdependent, this work exploits this mutual interdependent of this triplet of metrics based on the notion that a better observation leads to better estimation. We applied NTF1 model, a variation of non negative tensor factorization (NTF) for this purpose and estimated link delay from a triplet of metrics. Evaluation process used data from an experimental test bed that consists of standard networking devices. The estimated link delays were correlated to actual link delays to benchmark the accuracy of estimation. Results showed a better correlation between the estimated and measured link delays when a triplet of metrics is used.

Selection and Peer-review under responsibility of the Program Chairs.

Keywords: Network Tomography; Delay Estimation; Network Monitoring; Inverse Model Application; Packet Loss

1. Introduction

With the time, the data networks are now dealing with an explosion of applications and hosting a great variety of applications with diverse network needs. With the emergence of new applications and new technologies of data networks, it is important for service provides and customers to accurately predict the impact of new applications with respect to the performance of data networks. In the past, It was easy for many applications to adapt to the exploding nature of traffic flows through timeout and retransmission functions of the upper layer protocols. Now, however, new applications, such as voice and video, are more susceptible to changes in the transmission characteristics of data networks. It is imperative to understand the traffic characteristics of the network before deployment of new applications to ensure successful implementations.

Network monitoring and evaluation of the performance of a network is essential for successful network management. Some important network parameters are always required to properly monitor, predict, and diagnose the status of a network. Getting such network parameters is not an easy task, because access to measurements are restricted, and

* Muhammad H. Raza. Tel.: +1-902-494-6869.
E-mail address: hraza@dal.ca

Selection and Peer-review under responsibility of the Program Chairs.
doi:10.1016/j.procs.2014.05.529

there are many users and a high volume of traffic [1,2].

There are a number of network parameters that cannot be estimated directly. For example, many flows can pass through a single link. It is difficult to measure the link delay per flow in each link whereas a combined link delay of all the traffic passing through a link is available. Similarly, a flow passes through many links on it path. Path delay can be easily measured, but getting individual link delays on this path is difficult due to the following reasons. The goals of assessing network performance, capacity planning and efficient routing become difficult when networks are decentralized and multilayered. Quantitative network performance assessment is very difficult, and the expectation of full cooperation of the routing equipment is unrealistic in most situations [1,2,3]. Internal network information is usually considered confidential and is not shared with outsiders. The owner of a network domain has self-restricted information about a self owned network and a little or no knowledge about the properties of other domains. Service providers cannot depend on internal network elements to freely transmit vital network statistics such as traffic rates, link delays, and packet loss rates. Routers already bear the burden of managing large amounts of incoming traffic across multiple outgoing links at very high data rates and any increased burden is inadvisable. Even if the internal link-level characteristics are assumed to be collectable, collecting such statistics on various hosts may result in a considerable increase in overheads. On demand processing and communication of performance-related statistics may make network monitoring an impractical approach.

Network tomography is capable of measuring the statistics of interest that may not be measured directly [4]. Network tomography measures a parameter (that is usually not required for network management) actively or passively, and the desired parameter is indirectly measured by applying statistical techniques using an inverse modeling.

The simplest model of network tomography is shown 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. An element (A_{ij}) of the routing matrix is 1 if the link j is included in the path i and 0 otherwise [4].

In contrast to the conventional model, this paper proposes the use of direct measurements of multiple metrics to recover indirectly a single parameter with the expectation of getting a better estimate as compared to using a single directly measured metric to estimate another metric indirectly. The new model is represented by Equation 2, where Y_1, Y_2, and Y_3 are directly observed in order to estimate X indirectly by solving the following inverse equation.

$$Y_1 Y_2 Y_3 = AX \tag{2}$$

For example, instead of estimating link delays from merely end-to-end path delays, we can estimate link delays from a combination of path delays (Y_1), PLR (Y_2) and jitter (Y_3).

Many applications are susceptible to network behaviors, referred to as delay and jitter, which can degrade some application to the point of being unacceptable to the average user. Delay is the time taken from pointtopoint in a network. Delay can be measured in either oneway or roundtrip delay. Jitter is the variation in delay over time from pointtopoint. The amount of jitter tolerable on the network is affected by the depth of the jitter buffer on the network equipment in the path. The more jitter buffer available, the more the network can reduce the effects of jitter.

Packet loss is losing packets along the data path, which severely degrades the voice application. Prior to deploying applications, it is important to assess the delay, jitter, and packet loss on the data network. The delay, jitter, and packet loss measurements can then aid in the correct design and configuration of traffic prioritization, as well as buffering parameters in the data network equipment.

By having a better input in terms of three interdependent metrics, a better estimation of link delays can be obtained than using only one parameter such as path delays for the estimation link delays. The correlation of network parameters has been discussed in the literature. For example, the authors of [5] report on the correlation between delay and loss observed by a continuous-media traffic source. This study [5] determines the extent to which one performance measure could be used as a predictor of the future behavior of the other (for example, an increasing delay is a good predictor of future loss) so that an adaptive continuous media application might take anticipatory action based on the observed performance. Similarly, the authors of [6] studied the time-dynamic behavior of delay jitter as captured by the auto-correlation function; the second order statistics provide information relating to consecutive packet loss in real-time services. The interdependence of delay, PLR, and jitter is the motivation behind this work. A better observation is expected to lead to a better estimate. A variation of NTF called the NTF1 model has been applied for the purpose of estimating link delays for the knowledge of path delays, PLR, and jitter.

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 details the performance evaluation based on experimental setup. Section 4 concludes this paper.

2. Related Work

This paper proposes an improved methodology to estimate link delays from a triplet of metrics (path delay, PLR, and jitter.

To the best of the authors' knowledge, there has never been an implicit consideration of directly measured multiple metrics for an indirect estimate of a network metric except a position paper on multi-metric network tomography[7]. This position paper only introduces the concept of multi-metric tomography without presenting an analytical evaluation. The model introduced in[7] consists of only two observed interdependent network metrics; path delays and PLR. This paper introduces a new model with triplet of observed interdependent network metrics; path delays, PLR, and jitter. A complete set of analytical result based on the data obtained from a test bed backs up the claim. The following subsection explains the interdependence of path delay and PLR.

2.1. Interdependence of Path Delays, PLR, and Jitter

The authors of[5] examine the correlation between packet delay and packet loss experienced by a continuous media traffic source. The work in[5] (also referred to in[7]) studies the extent to which one performance measure can be used to predict the future behavior of the other (for example, whether an observed increasing delay is a good predictor of future loss) so that an adaptive continuous media application might take anticipatory action based on the observed performance. They provide a quantitative study of the extent to which such correlation exists. There are two examples in this regard mentioned in the following discussion.

When the buffer reaches its capacity, packet losses begin to occur. The receiver of the continuous-media application thus sees increased delay, and eventually losses.

When packets from a continuous-media application arrive at a buffer that is already full, they are dropped. As other sources (for example, TCP connections) detect congestion and decrease their transmission rate, the queue length at the buffer will decrease, and packets from the continuous-media application will start to be queued, rather than dropped. The receiver sees losses followed by high, but possibly decreasing, packet delays.

The authors of[5] introduce a lag, loss-conditioned average delay, in calculating the average delay conditioned on loss. Specifically, the average packet delay, conditioned on a loss occurring at a time lag *j* packets in the past, is the average delay of all packets in the trace that have a loss *j* packets before them in the trace. That is,

$$E[d_i \mid l_{i-j} = 1] = \sum_{k \epsilon P} d_k / \mid P \mid, \tag{3}$$

where P = $k : l_{k-j} = 1$ and $l_k = 0$.

If the loss-conditioned average delay at a positive lag of *j* is higher than the unconditional average delay (that is, the delay averaged over all received packets), the packets that arrive *j* packets after a loss have a higher average delay than the unconditional average delay. That is, a loss occurring *j* packets in the past can be taken as a precursor to a higher delay later.

The authors of[6] study the second-order statistics of the delay jitter sequence because the time dynamics play an important role in network design and operation. To meet the quality of service requirements, the jitters for real-time services must be removed prior to delivery at the destination. Intuitively, the greater the positive amplitude and duration of the jitter correlation, the more likely consecutive packet losses will occur.

In[6], the time-dynamic behavior of delay jitter is captured by the autocorrelation function. The second order statistics provide information relating to consecutive packet loss in real-time services. The jitter second-order statistics are shown to depend strongly on those of the queue, and the queue second-order statistics depend strongly on those of the input traffic. A numerical study places emphasis on a periodic traffic stream (such as CBR video) multiplexed on a major communication node.[6] investigates the impact of input traffic correlations and queueing system parameters on the queue length and delay jitter second-order statistics.

The jitter correlation provides some indication of the probability of consecutive packet loss. The jitter autocorrelation function is strongly dependent on the second-order statistics of the queueing delays, which are influenced greatly by the input traffic correlations.

Previous analyses have concentrated on the steady-state behavior of delay jitter; in contrast, the authors of[6] investigated the time dynamics as captured by the second-order statistics.

[6] demonstrates the strong dependence of the jitter second order statistics on those of the queue, showing that the shape of the jitter correlation is determined by the weighted difference in queue correlations. The jitter power (variance) tends to behave similarly to queueing power (variance, mean).

This evidence of interdependence among path delay, PLR, and jitter motivated the consideration of indirect estimation of link delays from this triplet of network metrics. NTF1 has been applied to carry out the multiple metric network tomography[8]. There are several possible approaches to find or identify an extended NTF1 model such as global strategy, or local strategy, or a combination of both. A global strategy based on alternating minimization of cost function has been applied in this paper[8].

3. Experimental Performance Evaluation

This section presents the evaluation process of the estimation of link delays from a triplet of network metrics, path delays, PLR, and jitter.

3.1. Purpose and Methodology

The data used in the evaluation process is obtained from an experimental test bed. A matrix of actual link delays is measured from this test bed. Path delays, PLR, and jitter are also measured directly from this test bed. This triplet of network metrics is input to NTF1 model to get the estimated matrix of link delays. Correlation is determined between the best matching rows of the actual link delays and the estimated link delays to establish the accuracy of the proposed technique.

For the purpose of performance comparison, the following set of two correlation types between the estimated and measured link delays are determined.

1. Link delays are estimated from only path delays. Correlation is determined between the actual link delays and the estimated link delays from only path delays. NNMF is used for this purpose as described in[4].
2. Link delays are estimated from a triplet of observed metrics (path delays, PLR, and jitter) by using NTF1 model. Correlation is determined between the actual link delays and the estimated link delays from path delays, PLR, and jitter. Multiple types of traffic and traffic disturbances are used to obtain the experimental data.

As the idea of the estimation of link delays from the observation of multiple link is novel, no previous work is available for performance comparison[7]. Due to this reason, the actual link delays are also measured directly for verifying the accuracy of estimated link delays. A correlation, between estimated and measured link delays, close to 1 means the estimated link delays are close to measured link delays.

3.2. Test Bed Setup

The experimental data was obtained from a test bed that consisted of six 3800 series Cisco routers and four Cisco IPTV nodes as shown in Figure 1. Various types of traffic was utilized in the test bed, such as a multimedia traffic through Cisco IPTV setup, Path Echo option of Cisco CSLA, and multiple extended pings. The Cisco IPTV source was connected to the router, BR4, and all the four probes were also initialized from the same router. Routers TR1, TR2, and TR4 were the recipients of the multimedia transmission from the cisco IP/TV and the four probes.

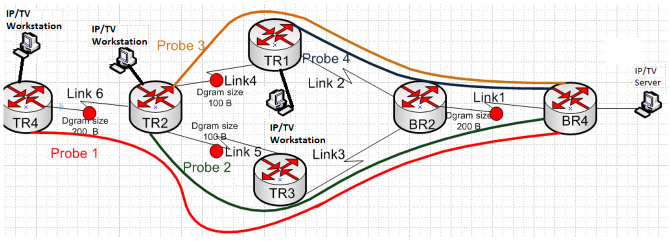

Fig. 1. Testbed setup for multiple network tomography

3.3. *Estimation of Link Delay from Path Delays*

As the first part of evaluation process, the link delays were estimated from path level delays and then the correlation between the estimated and measured link delays was determined[4]. The Echopath option of the Cisco Service Level Agreement (CSLA) was implemented to send four probes and collect the cumulative RTT from source to each hop. 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 seconds between two consecutive repetitions. The selected links were stressed with extended ping.

Figure 1 shows the test bed with the four probes and two of the links (Link 1 and Link 6) were stressed with an extended ping of 200 Bytes. The condition of the network remained unchanged during the CSLA operation.The data obtained from the CSLA is in the form of accumulative hop-wise RTT.

The data obtained from the CSLA was in the form of accumulative hop-wise RTT; the following steps were followed to process the data for obtaining two matrices; a matrix of path delays and a matrix of link level delays.

The parsing software, written in Java, extracted link delays and path delays in the form of two matrices from the accumulative round trip time at each hop on the path of a probe.

The hop-wise RTT is cumulative in nature meaning that RTT for the Hop 1 is the RTT from the source to the Hop 1 and RTT for the Hop 2 is the RTT from the source to the Hop 2. Subtracting RTT of the previous hop from the current hop gives the RTT from the previous to the current hop. Therefore the RTT for the last hop represented by the field, Hop, of a probe defines the RTT for that hop and is used to make a matrix of the path delays. The differential delays between the subsequent hops are used to get a matrix of true link delays.

Path delays were the input to NNMF to get estimate of link delays. The MatLab tool NMFpack was used for NNMF factorization as described in[4]. The sparsity of the routing matrix is kept fixed at 0.3 and the sparsity of the link delays varies from 0.1 to 0.9. The lower value of sparsity in routing matrix forces the correlation between the measured and estimated link delays to lower values. Otherwise, as mentioned in[4], network tomography with NNMF is capable of calculating the correlation between measured and estimated link delays close to 1. The purpose of forcing lower sparsity here is to produce significant space in correlation to show the impact of the estimation of link delays from a triplet of network metrics.

The coefficient of correlation between the estimated link delays (H) and actual link delays (X) was determined by using a modified component of EEGLAB[4,9]. As a result a column vector of correlation coefficients between the best-correlating rows of the measured and estimated link delays was obtained.
Figure 2 shows the correlation between the estimated and true link delays as a function of the sparsity.

3.4. *Estimation of Link Delay from a Combination of Path Delays, PLR, and Jitter*

This subsection presents the evaluation of the estimation of link delays from a combination of path delays, PLR, and jitter. The correlation between the estimated and the measured link delays was measured again and it was expected that this correlation would be better than the correlation shown in Figure 2. In the same test bed, two types of traffic were injected; CSLA and Cisco IP/TV.

In this section, in addition to finding path delays and PLR, a matrix of jitter was also obtained. For this purpose, the data was viewed using the Cisco IOS show command at the command line on the delay and jitter probes. A java

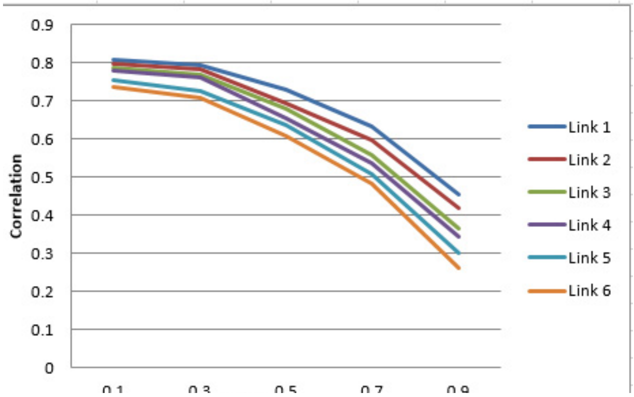

Fig. 2. Correlation between the true and the estimated link delay from path delays

program was used to gather data from the command line and export it to a text file for later analysis. Alternatively, for general statistics on delay and jitter, HP OpenView can also be polled. The data can be gathered from the rttMonJitterStatsTable from an HP OpenView Network Node Manager MIB poll.
From the collected statistic of the router, the fields of MinOfPositivesSD, MaxOfPositivesSD, MinOfNegativesSD, MaxOfNegativesSD, and the sum of these fields were the instrumental parameters in calculating the matrix of jitter values.

Quality of service (QoS) is implemented by either raising the priority of a flow or limiting the priority of another flow. The Cisco IP/TV traffic and the CSLA traffic were identified as the two traffic classes and the Cisco IP/TV traffic class was provided preferential service over the CSLA traffic.

Two types of CSLA ping traffic were used in this experiment as explained in the following text. The Path Echo operations record statistics for each hop along the path that the operation takes to reach its destination. A show command displays various parameters including *PacketLossSD* and *PacketLossDS*, that are packet loss from source to destination and destination to source packet loss, and these two parameters give values at each repetition of the show command. These two parameters (*PacketLossSD* and *PacketLossDS*) were instrumental in determining PLR. The observed PLR on multimedia traffic was in the range of 5 to 15 percent due the combination of quality of service enforcement on two types of traffic (ping and multimedia) and traffic disturbances from extended pings.

Cisco IP/TV multimedia traffic was sent from the source (BR4) to workstation connected to TR4, TR2, and TR1, Figure 1.

Three matrices, pathe delay, PLR, and jitter were input to NTF MatLab package to get the matrix of estimated link delays. Correlation between the measured and estimated link delays was calculated as a function of sparsity. Figure 3 shows a correlation between the estimated (from a triplet of metrics) link delays and true link delays as function of sparsity. By comparing Figure 2 and Figure 3, it is evident that the estimation of the link delays by using the input of two matrices, the path delays and the PLR is better than the estimation of the link delays from a single metric of the path delays.

3.5. Discussion on Performance Comparison

This discussion on performance comparison is based on the outcomes of the two subsections of the evaluation process in Section 1V. Two types of traffic and measurement mechanisms with different scopes are applied. A combination of traffic types, disturbances, data collection from test bed, and implementation of various mathematical approaches resulted in two types of link delay estimations. Link delays estimated from two different methods were correlated (one by one) to the measured version of link delays. Figure 2 and Figure 3 show the correlation of the estimated and measured link delays as function of the sparsity in the experimental network with a single observed metrics and with a triplet of metric, respectively.

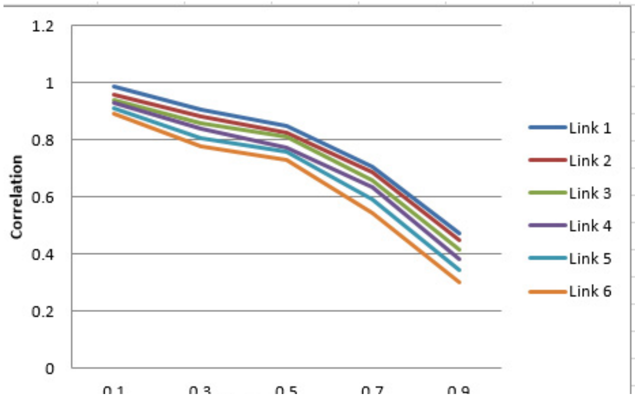

Fig. 3. Correlation between the true and the estimated link delay from a triplet of metrics, path delay, PLR, and Jitter

It is obvious that on the lower sparsity when most of the links are in use and the network behavior is close to a normal network, the correlation values improve by using a triplet instead of a single metric for estimation. This improvement in correlation values is evident that if we observe a a triplet of metrics (path delays, PLR, and jitter), the correlation values improve to a range of 1 from a range of 0.8. Thus, better observation in the form of a triplet of metrics led to better estimation of link delays.

4. Conclusions

This paper exhibited a successful implementation of the estimation of link delays from a triplet of network metrics. An evaluation process based on the data obtained from an experimental test bed proved that link estimation from the triplet of network metrics was better than that those of by a couple of metrics or a single metric. This intuition was based on the interdependent of link delay, PLR, and jitter. Improved observation in the form of interdependent parameters led the better estimation of link delays. The results of this work proved that by observing easily available metric, some out of reach metrics can be estimated with great accuracy.

References

1. R. Castro, M. Coates, G. Liang, R. Nowak and B. Yu. Network tomography: recent developments. *J. Statistical Science* 2004. p. 499-517.
2. E. Lawrence, G. Michailidis, V. Nair and B. Xi. Network tomography: a review and recent developments. *J. Ann Arbor*2006. p. 1087-1107.
3. R. Castro, M. Coates, G. Liang, R. Nowak and B. Yu. Network tomography: recent developments. *IEEE Signal Processing Magazine* 2002. p. 45-65.
4. M. Raza, B. Robertson, W. Phillips and J. Ilow. Network tomography by non negative matrix factorization (NNMF)In: Jones BS, Smith RZ, editors. *In proc. of IEEE SPECTS*, July 2010. p. 58-64.
5. S. Moon, J. Kurose, P. Skelly and D. Towsley. Correlation of packet delay and loss in the Internet. *Univ. Massachusetts, Amherst, MA, Tech. Repor* 1999.
6. Cathy Fulton and San-Qi Li. Delay jitter correlation analysis for traffic transmission on high speed networks. *In INFOCOM95. Fourteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Bringing Information to People. Proceedings.*. IEEE; 1995. p. 717-727.
7. M. Raza, B. Robertson, W. Phillips and J. Ilow. Multimetric network tomography. *In procedings of IEEE DCNET*. Greece: IEEE; 2010.
8. A. Cichocki, R. Zdunek, A. Phan, and S. Amari. *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*. Wiley, 2009.
9. A. Delorme and S. Makeig. EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *J. Neuroscience Methods*. 2004