PASSIVE NETWORK TOMOGRAPHY USING EM ALGORITHMS

Yolanda Tsang, Mark Coates and Robert Nowak

Department of Electrical and Computer Engineering, Rice University
6100 South Main Street, Houston, TX 77005–1892
Email: {*ytsang, mcoates, nowak*}*@ece.rice.edu, Web: www.dsp.rice.edu*

ABSTRACT

The paper presents a new method for characterizing communication network performance based solely on passive traffic monitoring at the network edge. More specifically, we devise a novel expectation-maximization (EM) algorithm to infer internal packet loss rates (at routers inside the network) using only observed end-to-end (source to receiver) loss rates. The major contributions of this paper are three-fold: we formulate a passive monitoring procedure for network loss inference based on end-to-end packet pair observations, we develop a statistical modeling and computation framework for inferring internal network loss characteristics, and we evaluate the performance with realistic network simulations.

1. INTRODUCTION

In large-scale networks, end-systems cannot rely on the network itself to cooperate in characterizing its own behavior. This has prompted several groups to investigate methods for inferring internal network behavior based on end-to-end network measurements [1, 2]; the so-called *network tomography* problem. However, earlier methods relied on multicast protocols , in which probe packets are sent from one source to multiple receivers in a single send operation [3]. Although multicast network tomography shows promise, many networks do not support multicast, limiting the practical utility of such schemes. Moreover, routers treat the multicast packets differently from unicast packets (which account for the vast majority of network traffic), and therefore inferences drawn from multicast measurements may poorly reflect the actual network performance observed by most traffic.

In this paper, we describe a new methodology for network tomography (specifically, inferring packet loss rates at internal network routers) based on *passive* monitoring of unicast traffic (as opposed to active probing). This methodology builds on our earlier work in unicast network tomography [4], and is also related to more recent efforts to apply multicast tomography techniques to unicast measurements based on active probing [5]. In unicast protocols, each packet is sent from the source to a single receiver. Most traffic in the Internet is unicast in nature, so our approach is broadly applicable. Furthermore, in contrast to multicast techniques which rely on active probing, passive unicast monitoring avoids the problematic issues associated with active probing (*e.g.,* overburdening the network with probes). Thus, passive unicast network tomography is scalable to very large networks and it should provide a more accurate description of the network performance. Throughout the remainder of the paper we work with "success" rates (rates of non-loss) instead of loss rates. This provides a more convenient mathematical parameterization of the problem, and the rate of loss is simply one minus the rate of success.

The paper is organized as follows. In Section 2, we introduce the basic unicast network tomography problem and the technical issues involved. In Section 3, we formally define our loss modeling assumptions and passive measurement framework. In Section 4 we pose unicast network tomography as a maximum likelihood estimation problem, and we propose a novel EM algorithm for computing maximum likelihood estimates of internal success (or loss) rates. In Section 5, we examine the performance of our methods through simulation, and concluding remarks are made in Section 6.

This work was supported by the National Science Foundation, grant no. MIP–9701692, the Army Research Office, grant no. DAAD19-99-1-0349, the Office of Naval Research, grant no. N00014-00-1-0390, and Texas Instruments.

2. UNICAST TOMOGRAPHY

We consider a scenario in which a number of receivers are connected to a single source with some common links in the paths (extensions to multiple sources are possible). In this case, the network topology (from the perspective of the source) is a tree-structure. Figure 1 depicts an example topology with source (node 0) and seven receivers (nodes 5 through 11). Also shown are four internal routers (nodes 1 through 4). We assume that we are able to measure network traffic only at the edge; that is, we can determine whether or not a packet sent from the source is successfully received by one of the receivers. This type of confirmation can be obtained via Transmission Control Protocol's (TCP) acknowledgment system [3], for example. We also assume that the routing table is fixed for the duration of the measurement process, which ensures the tree-structured topology.

The goal of this work is to estimate the loss rates associated with each individual link (between two routers) in the network. Restricting ourselves to edge-based measurement, we can measure the numbers of packets sent to and received by each receiver, providing us with a simple means of estimating the rates of success along each path (from source to receiver). Unfortunately, there is no unique mapping of the path success rates to the success rates on individual links (between routers) in the path. To overcome this difficulty, we propose a methodology based on measurements made using back-to-back packet pairs. These measurements provide an opportunity to collect more informative statistics that can help to resolve the links.

The basic idea is quite straightforward. Suppose two closely time-spaced (back-to-back) packets are sent to two different receivers. The paths to these receivers share a common set of links from the source but later diverge. If one of the packets is dropped

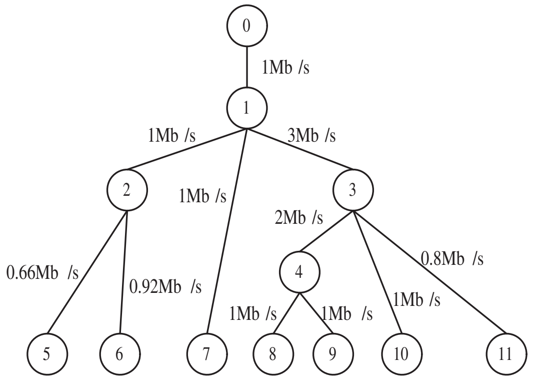

Fig. 1. Tree-structured graph representing a single-source, multiple-receiver network. Node 0 is the source, nodes 1-4 internal routers, and nodes 5-11 receivers. Beside each link we indicate the capacity in megabits per second.

and the other successfully received, then (assuming total correlation of losses on common links) one can infer that the packet must have been dropped on one of the unshared links. This enables the resolution of losses on individual links. Collecting measurements from an assortment of such back-to-back packet pairs (sent to different combinations of receivers) allows us to resolve the losses occurring on all links in the network. The key to this approach is the exploitation of the correlation between packet-pair losses on common subpaths. In practice, however, this correlation is not perfect, and therefore more sophisticated statistical modeling and inference strategies are necessary, as described next.

3. LOSS MODELING AND MEASUREMENT

Here we define the relevant parameters in tomographic loss rate estimation, describe the passive measurement scheme, and develop statistical models relating these measurements to the parameters of interest. For individual packet transmissions, we assume a simple Bernoulli loss model for each link. The *unconditional* success probability of link i (the link into node i) is defined as

$$\alpha_i \;\equiv\; \text{Pr(packet successfully transmitted from } \rho(i) \text{ to } i),$$

where $\rho(i)$ denotes the index of the parent node of node i (the node above i-th node in the tree; *e.g.,* referring to Figure 1, $\rho(1) = 0$). A packet is successfully sent from $\rho(i)$ to i with probability α_i and is dropped with probability $1 - \alpha_i$. We model the loss processes on separate links as mutually independent.

If two, back-to-back packets are sent from node $\rho(i)$ to node i, then we define the *conditional* success probability as

$$\gamma_i \;\equiv\; \text{Pr(1st packet } \rho(i) \to i \mid \text{2nd packet } \rho(i) \to i\text{)},$$

where $\rho(i) \to i$ is shorthand notation denoting the successful transmission of a packet from $\rho(i)$ to i, and "first" and "second" refer to the temporal order of the two packets. That is, given that the second packet of the pair is received, then the first packet is received with probability γ_i and dropped with probability $1 - \gamma_i$. We expect that γ_i should be very close to one (if the interarrival time between the two is small). In general, the actual value of γ_i depends on the number of events (other arrivals and services) between the arrivals of the two packets under consideration. An arrival event increments the queue length by one. A service event decrements the length by one. We also assume that the service events are independent of the length (e.g. DropTail [6]). Denote the number of intervening events by r, and let $\gamma_i^{(r)}$ denote the specific value of γ_i in this case. Then $\gamma_i^{(r)}$ satisfies the two key conditions: **(a)** $\gamma_i^{(0)} = 1$, and **(b)** $\gamma_i^{(r)} \geq \alpha_i$, for all r. These conditions hold for any finite non-adaptive queue, independent of the traffic arrival and service processes, as shown by the following theorem.

Theorem 1 *Let K denote the size of the queue and k_0 the number of packets in the queue immediately before (no intervening services or arrivals) the first packet in the pair arrives. The probability of $k_0 = j,\ j = 0, \ldots, K$ is denoted by $p(j)$. Let r denote the total number of both arrivals and services events that occur between the first and second packets, and let ℓ_r denote the length of the queue immediately before the second packet arrives.*

$$p(k_0 < K \mid \ell_r < K) \;=\; \frac{1}{1 + c_r},$$

where

$$c_r \;\equiv\; \frac{p(k_0 = K\,,\ \ell_r < K)}{p(k_0 < K\,,\ \ell_r < K)}.$$

Furthermore, $c_0 = 0$ and $c_r \leq \frac{p(K)}{1-p(K)}$ for all r, which implies conditions **(a)** *and* **(b)** *above, since $\alpha_i = 1 - p(K)$.*

The proof of the theorem is lengthy, but is not overly difficult to establish, and therefore we refer the interested reader to a technical report containing the proof [7]. Of importance here is that the conditional success rate for closely time-spaced packets is approximately one (total correlation), which then allows us to isolate losses on unshared links, as mentioned in Section 2.

Packet measurements can be collected by passively monitoring connections. For example, we have developed a method in which single packet and back-to-back packet events are selected from TCP traffic flows [7]. The two types of measurements we require are formally described below.

Single Packet Measurement: Suppose that n_i packets are sent to receiver i and that of these a number m_i are actually received ($n_i - m_i$ are dropped). The likelihood of m_i given n_i is binomial (since Bernoulli losses are assumed) and is given by

$$l(m_i \mid n_i, p_i) \;=\; \binom{n_i}{m_i} p_i^{m_i}\,(1 - p_i)^{n_i - m_i},$$

where $p_i = \prod_{j \in \mathcal{P}(0,i)} \alpha_j$ and $\mathcal{P}(0,i)$ denotes the sequence of nodes in the path from the source 0 to receiver i. For example, in Figure 1, $\mathcal{P}(0,8) = \{1,3,4,8\}$ and so $\prod_{j \in \mathcal{P}(0,8)} \alpha_j = \alpha_1\alpha_3\alpha_4\alpha_8$.

Back-to-Back Packet Pair Measurement: Suppose that the source sends a large number of back-to-back packet pairs in which the first packet is destined for receiver i and the second for receiver j. We assume that the timing between pairs of packets is considerably larger than the timing between two packets in each pair. Let $n_{i,j}$ denote the number of pairs for which the second packet is successfully received at node j, and let $m_{i,j}$ denote the number of pairs for which both the first and second packets are received at their destinations. Furthermore, let $k_{i,j}$ denote the node at which the paths $\mathcal{P}(0,i)$ and $\mathcal{P}(0,j)$ diverge, so that $\mathcal{P}(0,k_{i,j})$ is their common subpath. For illustration, refer to Figure 1 and let $i=8$ and $j=11$, then $k_{8,11}=3$. With this notation, the likelihood of $m_{i,j}$ given $n_{i,j}$ is binomial and is given by

$$l(m_{i,j}\,|\,n_{i,j},p_{i,j}) \;=\; \binom{n_{i,j}}{m_{i,j}} p_{i,j}^{m_{i,j}}\,(1-p_{i,j})^{n_{i,j}-m_{i,j}},$$

where

$$p_{i,j} = \prod_{q\in\mathcal{P}(0,k_{i,j})}\gamma_q \prod_{s\in\mathcal{P}(k_{i,j},j)}\alpha_s.$$

4. MAXIMUM LIKELIHOOD TOMOGRAPHY

Assume that we have made an assortment of single packet and back-to-back packet measurements (sent to different receivers or combinations of receivers) as described in the previous section. Collecting all the measurements, define

$$\begin{aligned}\mathcal{M} &\equiv \{m_i\}\cup\{m_{i,j}\}\\ \mathcal{N} &\equiv \{n_i\}\cup\{n_{i,j}\},\end{aligned}$$

where the index i alone runs over all receivers and the indices i,j run over all pairwise combinations of receivers in the network.

Let us also denote the collections of the unconditional and conditional link success probabilities as $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$, respectively. The *joint* likelihood of all measurements is given by

$$l(\mathcal{M}\,|\,\mathcal{N},\boldsymbol{\alpha},\boldsymbol{\gamma}) \;=\; \prod_i l(m_i\,|\,n_i,p_i)\;\times \prod_{i,j} l(m_{i,j}\,|\,n_{i,j},p_{i,j}).$$

The maximum likelihood estimates of $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$ are defined as

$$(\widehat{\boldsymbol{\alpha}},\widehat{\boldsymbol{\gamma}}) \;=\; \arg\max_{\boldsymbol{\alpha},\boldsymbol{\gamma}}\; l(\mathcal{M}\,|\,\mathcal{N},\boldsymbol{\alpha},\boldsymbol{\gamma}).$$

Computing maximum likelihood estimates or marginal likelihood functions can be a formidable task. Multidimensional maximizations or integrations are time-consuming and directly attempting any of the inference tasks outlined in the previous section leads to extremely computationally demanding algorithms that are not scalable to large networks. The basic problem is that the individual likelihood functions $l(m_i\,|\,n_i,p_i)$ or $l(m_{i,j}\,|\,n_{i,j},p_{i,j})$ for each type of measurement involve products of subsets of the $\boldsymbol{\gamma}$ and/or $\boldsymbol{\alpha}$ probabilities. Consequently, it is difficult to separate the effects of each individual success probability.

We overcome this difficulty using a common device in computational statistics known as *unobserved* data or variables. To introduce the notion of unobserved data, let us consider the likelihood $l(m_i\,|\,n_i,p_i)$ for a single packet measurement. Assuming that the path consists of more than one link, the effects of the individual link success probabilities on this measurement are combined through the product p_i over the entire path. However, suppose it were possible to measure the numbers of packets making it to each node. Let us denote these unobserved measurements by $u_{j,i}$, $j\in\mathcal{P}(0,i)$, $j\neq i$. With these measurements in hand, we can write the *complete data likelihood* function as

$$l\left(u_{j,i}\,|\,n_i,p_i\right) \;=\; \prod_{j\in\mathcal{P}(0,i)} \binom{u_{\rho(j),i}}{u_{j,i}} \alpha_j^{u_{j,i}}\,(1-\alpha_j)^{u_{\rho(j),i}-u_{j,i}},$$

where $\rho(j)$ again denotes the parent of node j. Also, since we are able to measure at the source and receiver, in the expression above we set $u_{0,i}=n_i$ and $u_{i,i}=m_i$. The example in Figure 2 illustrates the notion of unobserved data. In a similar fashion, we introduce unobserved data for all measurements (including packet pairs) and paths, and these variables allow us to factorize the joint likelihood function into a product of univariate functions. The key feature of the complete data likelihood function is that it factorizes into a product of individual binomial likelihood functions, each involving just a single success probability. Thus, the complete data likelihood function is a trivial multivariate function, and the effects of the individual link probabilities are easily separated.

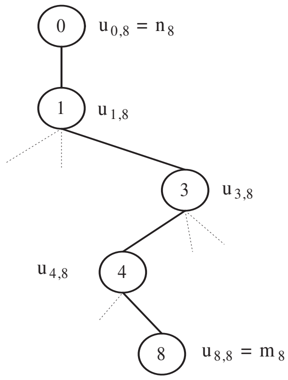

Fig. 2. Path from source to receiver $i=8$ with unobserved data at each internal router.

The Expectation-Maximization Algorithm: The EM Algorithm [8] can be used for our problem to compute maximum likelihood estimates of $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$. Beginning with an initial starting point for $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$, the algorithm is iterative and alternates between two steps until convergence. The Expectation (E) Step computes the conditional expected value of the complete data likelihood given the observed data, under the probability law induced by the current estimates of $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$. The E Step can be computed in $O(L^2N)$ operations [7], where N is the total number of receivers and L is the average number of links in each path, using an upward-downward probability propagation (or message passing) algorithm. In the M Step, the expected complete data likelihood function is maximized with respect to $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$. Since the complete data likelihood factorizes into a product of univariate functions, each involving just one success probability, the maximizers have closed-form, analytic expressions. The M Step can also be computed in $O(LN)$ operations. Each iteration of the EM Algorithm is therefore $O(L^2N)$ in complexity. Moreover, it can be shown that the original (observed data only) likelihood function is monotonically increased at each iteration of the algorithm, and the algorithm converges to a local maximum of the likelihood function [8]. In our experiments in Section 6, we declare that the algorithm has converged when the maximum difference between the vector of unconditional success rates at the k-th iteration $\boldsymbol{\alpha}^{(k)}$ is within a certain tolerance of the previous iterate. Specifically, we adopt the following stopping criterion:

$$\max_k \|\boldsymbol{\alpha}^{(k)}-\boldsymbol{\alpha}^{(k-1)}\| < 10^{-3}.$$

We have found that the algorithm typically converges in a small number of iterations (15-50).

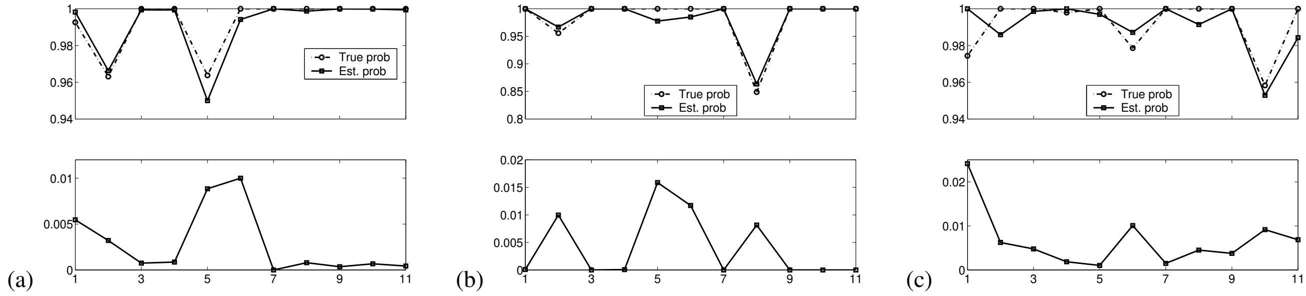

Fig. 3. Simulation Results. True and estimated link-level success rates of TCP flows from source to receivers for several traffic scenarios: (a) Heavy losses on links 2 and 5, (b) Heavy losses on links 2 and 8, and (c) Traffic mixture - medium losses. In each subfigure, the two panels display for each link 1-11 (horizontal axis): (top) an example of true and estimated success rates and (bottom) mean absolute error between estimated and true success rates over 10 trials for each link.

5. SIMULATION EXPERIMENTS

Using the 12-node network topology of Figure 1, we evaluate the performance of the combined EM loss inference algorithm and passive measurement framework in the `ns-2` simulation environment [9]. The topology is intended to reflect (to some extent) the heterogeneous nature of many networks – a slower entry link from the source, a faster internal backbone, and then slower exit links to the receivers. This chosen topology gives us the flexibility to explore the effects of having receivers at different distances from the source (number of links in path), and to examine the effect of varying fan-outs. We fix the queue size at each router to be 35 packets, and drops (losses) occur when a queue overflows.

Our experiments investigate a variety of network traffic conditions, comprised of TCP connections from the source to receivers as well as background cross-traffic flows. Single packet and packet pair statistics are collected by monitoring the TCP connections. Within these connections, we identify two packets as a "pair" if the time-spacing between them is less than 2 msec. Details of packet pair identification appear in [7].

In this paper, we report the results from measurements collected over a 300 second interval in three different traffic scenarios. The first two scenarios investigate cases in which traffic and losses are heaviest on two links. The scenarios test the ability of the algorithm to resolve cascaded losses (links 2 and 5, Scenario (a) in Figure 3) or identify isolated lossy links in the network (links 2 and 8, Scenario (b) in Figure 3). In the third scenario, more evenly distributed traffic introduces medium losses at several links, exploring performance in more benign conditions (Scenario (c) in Figure 3).

In each case, we conduct ten independent simulations. Figure 3 displays the results. The top panel illustrates an example of the estimated and true success rate for each link, chosen arbitrarily from the ten realizations. We see that the estimated success rates are in good agreement with the true TCP success rates. The bottom panel shows the mean absolute error for each link over the 10 trials. In all three scenarios, we see that the worst-case mean absolute error is roughly 2%.

6. CONCLUSIONS

In this paper, we introduce a new methodology for network loss tomography based on passive monitoring of unicast traffic. Our approach takes advantage of the correlation between the losses experienced by back-to-back packet pairs. We pose the network tomography problem as a maximum likelihood estimation, and develop an EM algorithm for computing our estimates. We demonstrate using extensive ns-2 simulations that sufficient data can be collected using passive sampling to perform accurate loss inference, even for relatively short measurement periods. Moreover, we are able to accurately estimate the losses experienced by existing TCP flows.

7. REFERENCES

[1] R. Cáceres, N. Duffield, J. Horowitz, and D. Towsley, "Multicast-based inference of network-internal loss characteristics," *IEEE Trans. Info. Theory*, vol. 45, no. 7, pp. 2462–2480, November 1999.

[2] "Multicast-based inference of network-internal characteristics (MINC)," See gaia.cs.umass.edu/minc.

[3] J. Kurose and K. Ross, *Computer Networking: A top-down approach featuring the Internet*, Addison Wesley, 2001.

[4] M. Coates and R. Nowak, "Network loss inference using unicast end-to-end measurement," in *Proc. ITC Seminar on IP Traffic, Measurement and Modelling*, Monterey, CA, Sep. 2000, pp. 28–1–28–9.

[5] N.G. Duffield, F. Lo Presti, V. Paxson, and D. Towsley, "Inferring link loss using striped unicast probes," to appear, *Proc. IEEE Infocom'01*, April 2001. Available as http://www.research.att.com/projects/minc/dlpt00.ps.

[6] S. Keshav, *An Engineering Approach to Computer Networking: ATM Networks, the Internet, and the Telephone Network*, Addison-Wesley, Massachusetts, 1998.

[7] Y. Tsang, M. Coates, and R. Nowak, "Passive unicast network tomography based on TCP monitoring," Tech. Rep. TR0005, Rice University, Nov. 2000.

[8] G. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*, Wiley, New York, 1997.

[9] "The network simulator-2," For more information, see http://www.isi.edu/nsnam/ns/.