Rice University, ECE Department, Technical Report TR-0005

November 2, 2000

Passive Unicast Network Tomography based on TCP Monitoring

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

Network tomography is a promising new technique for studying the (internal) behavior of large-scale networks based solely on end-to-end measurements. Most network loss tomography methods utilize active probing. While such measurement schemes are efficient, the probing burden may become prohibitive for large-scale networks. As an alternative, we propose a new, completely passive measurement framework based on sampling of existing TCP traffic flows.

The new passive unicast network tomography methodology we propose shows considerable promise. We demonstrate its performance using extensive ns-2 simulations. We observe that we are able to accurately estimate the losses experienced by existing TCP flows. As these can differ substantially from losses suffered by other forms of traffic, we surmise that in some situations inference from active probing may offer a poor reflection of existing TCP loss rates. We conduct theoretical queuing analysis which corroborates the results obtained through simulation experiment.

*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.

I. Introduction

A. Background and Motivation

Accurately characterizing the performance of a large-scale network is essential if it is to be successfully managed and controlled. The characterization must extend further than path-level behavior; it is necessary to acquire information about internal (link-level) performance. One way to achieve this is to gather statistics at as many internal routers as possible, but the collection and compilation of such statistics is an onerous and expensive task. Often it proves impossible for any one organization or individual to collect relevant statistics when various parts of a network are administered by different parties.

An attractive alternative is to infer the internal performance from end-to-end measurements that are comparatively easy to make. A number of authors have proposed methods for achieving this task; the general problem has been termed "network tomography". One of the most promising proposals is MINC (Multicast Inference of Network Characteristics) [1], which proposes strategies for estimating loss, delay distributions and variances, and topology [6, 12, 5]. All of these techniques use active multicast probing, then exploit the inherent correlation between the losses/delays observed by multicast receivers. The performance of these algorithms is impressive [7], but there are two serious deficiencies in the methodology. Firstly, multicast protocols are not supported by significant portions of the Internet. Secondly, the internal performance measured by active multicast probes often differs significantly from that encountered by unicast packets, which comprise by far the most substantial component of Internet traffic [13].

Recently, strategies have been proposed to avoid these limitations [9, 10, 13]. These strategies involve the use of unicast packets to acquire the end-to-end statistics required for network tomography. The difficulty encountered by these techniques is that unicast packets do not obey the same well-behaved correlations as multicast packets. Network probing using back-to-back packets has been proposed in a number of measurement schemes [3, 4, 8, 14]. A network tomography procedure based on measurements of back-to-back (closely time-spaced) unicast packets was first proposed in [9]. To address the problem of imperfect correlations, the authors used probabilistic modeling methods and likelihood maximization techniques (Expectation-Maximization (EM) and exact inference on factor graphs). The maximum bias of estimates was characterized in terms of the observed path-level correlations. The authors in [13] proposed an alternative strategy based on sending multiple-packet probes to improve the observed correlations, and then applied the multicast-based algorithms of the MINC project for loss estimation.

B. Contribution

Although the unicast network tomography proposal [9] suggested that the collection of statistics is possible using both active probing and passive sampling of existing traffic, the experiments and performances in reported in previous work [9, 10, 13] use active probing strategies. Moreover, a number of issues encountered in performing passive sampling have not been not addressed. In this paper, we concentrate on the inference of internal link losses from passive unicast end-to-end measurement. The motivation for passive inference is two-fold. Firstly, there is a risk that for a substantial network the insertion of the large number of probes required for accurate estimates might significantly impact the performance of the network (perturbing the very quantity to be estimated). Secondly, the sampling of existing flows potentially offers an opportunity to estimate the losses experienced by the existing flows (which can be substantially different from those experienced by inserted probes). We propose a new, truly passive methodology for unicast network tomography, and we assess the feasibility and performance our approach through theoretical analyses and `ns-2` simulation experiments.

We also make an important modification to the framework proposed in [9]. The success of the framework, as discussed in the following section, hinges on a conjecture. The conjecture states that the (causal) conditional probability of the second (in temporal order) packet in a pair successfully traversing a subpath given that the first packet did so is higher than the unconditional success probability (and close to one). In this paper, we consider the anti-causal conditional probability instead (the probability that the first packet is successful given that the second is). We are able to prove a critical result. The anti-causal conditional success probability is always greater than the unconditional success probability, irrespective of the nature of the traffic arrival process and queueing service strategy (assuming a drop-tail queue). If a single link (direct connection between two neighboring routers) is considered and the packets are truly back-to-back, then the conditional probability is one. The conditional probability then decays according to the nature of the traffic, eventually reaching the unconditional probability.

We use extensive `ns-2` [2] simulations to explore various aspects of our new methodology's performance. We assess whether sufficient back-to-back packet statistics can be extracted from existing TCP flows between the source and receivers. We quantify the performance of the inference algorithm in terms of mean absolute error.

The paper is organized as follows. In Section II, we review the basic unicast network tomography problem and the technical issues involved. In Section III, we describe the new passive measurement framework, focusing on the collection of statistics from TCP flows. In Section IV, we apply the loss modeling and likelihood analysis techniques devised in [9] to our new approach. In Section V, we perform a queuing analysis of the back-to-back measurement procedure, concentrating on the anti-causal conditioning proposal. In Section VI, we describe and discuss the results of `ns-2` simulations exploring the performance of the passive measurement and inference methodology. In Section VII, we discuss some of the limitations of the passive scheme, and propose some potential remedies. Conclusions are made in Section VIII.

C. Related Work

The proposals and performance analysis presented in this paper are important extensions of the work presented in [9, 10, 13], but the basic idea of exploiting correlations of closely-spaced packets

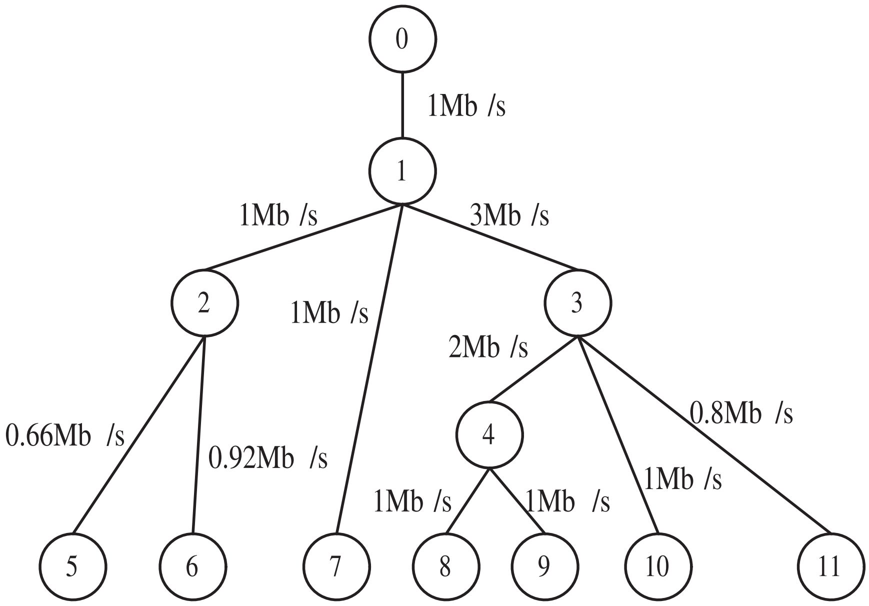


Figure 1: An example network topology with a single source (node 0), 4 internal routers, and 7 receivers.

remains the same. The authors of [7] presented extensive performance analysis of the multicast loss inference technique proposed in [6]; in this paper, we aim to perform a similar analysis for passive unicast inference. In [9, 10] the nature of the probing (active vs. passive) was not specifically addressed, and the methodology proposed in [13] is based on active probing.

II. Unicast Loss Tomography

In this section, we provide a brief overview of the unicast loss inference problem, describe the back-to-back measurement framework proposed in [9], and review the loss modeling framework. We illustrate the problem of unicast loss inference by considering the simple case in which a single source sends packets to multiple receivers. The problem and methodology are readily extended to the multiple-source case. Figure 1 depicts an example of this form of topology; the network appears to the source as a tree. The nodes of the tree correspond to the source (node 0), internal routers (nodes 1–4) and receivers (nodes 5–11). We define a 'link' as the connection between any two adjacent nodes in the tree, deem the set of links connecting a source and any receiver a 'path', and a subset of connected links in a path is referred to as a 'subpath'. The tree of Figure 1 does not necessarily depict all routers encountered by packets traveling from the source to receivers. It is possible that a number of routers are passed as a packet travels from node 1 to node 3, for example.

We consider the situation where measurements can only be made at the edge of the network and assume that the routing (and thus the topology) table is fixed for the duration of the measurement. The goal of passive unicast loss inference is to estimate the loss rates on the internal links of the network using solely passive sampling of the existing traffic. Estimating the loss rates is equivalent to estimating success rates, and henceforth we shall speak solely of success rates, since they simplify mathematical expressions in the proposed framework. (The success rate is simply one minus the loss rate).

It is straightforward to estimate *path* success rates, but, unfortunately, there is no unique mapping of the path success rates to the success rates on *individual* links in the path. To overcome this difficulty, the authors in [9] propose a methodology based on measurements of back-to-back packet pairs. These measurements provide an opportunity to collect more informative statistics that can help to resolve the links.

Back-to-back packet pairs have been utilized for inferring a number of network performance metrics [3, 4, 8, 14]. A back-to-back packet pair refers to two closely time-spaced packets, possibly destined for different receivers, but sharing a common set of links in their paths. If two back-to-back packets are sent across a link and the one of the pair was received, then it is highly likely that the other packet was also received. In other words, we expect that the conditional success probability of one packet (given that the other was received) may often be close to one. This observation has been verified experimentally in real networks [14]. Results can also be established theoretically without assumptions about traffic arrival and queuing service processes (see Section V).

A unicast network tomography method that exploits the correlation between back-to-back packet losses is developed in [9]. The basic idea is quite straightforward. Suppose two 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 and the other successfully received, then (assuming strong correlation of losses on common links) one can infer that the packet was probably dropped on one of the unshared links. This enables the resolution of losses on individual links.

III. TCP-Based Measurement Framework

Earlier unicast network tomography schemes have focused on active probing [9, 10, 13] (insertion of additional packets whose sole purpose is for measurement), although the possibility of passive measurement (collecting measurements directly from existing traffic) was mentioned, but not investigated, in [9]. Although active probing allows one to control the timing and nature of the measurements, thus assuring that sufficient measurements are made, it imposes an extra burden on the network which may make it impractical in many situations. Furthermore, active probing will disrupt the transmission of normal traffic, which may lead to biased estimates of the true losses (experienced in the absence of probe packets). Also, probe traffic may experience significantly different losses than normal traffic (e.g., TCP) since the temporal structure of the probing is specified by the experimenter, and is generally different than that of normal traffic. For example, TCP often results in clusters of packets separated by a significant (round-trip) time, contrary to a uniform or Poisson probing scheme.

We propose a passive traffic monitoring/sampling scheme in order to circumvent the problematic issues surrounding active probing. We focus on TCP-based measurement, because we are interested in estimating the link-level losses experienced by TCP connections flowing from the source to the receivers. When estimating TCP losses, the spacing of measurements should clearly not be uniform or exponentially distributed. Where the TCP traffic to a particular connection is dense, we should make many measurements; where light, only a few. The measurement process should ideally be a subsampled version of the true traffic. This is impossible to achieve exactly, because we have other constraints. For example, we need to make back-to-back measurements and we need to ensure that measurements are sufficiently spaced to provide the approximate inter-pair temporal independence assumed in our statistical model.

The situation we consider is one where the source has numerous contemporaneous TCP connections with a number of receivers. We want to extract as many informative measurements as we can from the existing TCP traffic. We concentrate first on extracting the important packet-pair measurements, which are less common than the isolated packet measurements. We first inspect the sending times of the TCP traffic at the source and decide that two packets form a packet-pair if their time-spacing is less than a threshold δ_t seconds. This threshold is dependent on the sending rate of the source. Commencing at the start of the measurement period, we sweep forward in time, seeking and identifying the first packet-pair. We then step forward by a fixed time interval $\Delta_t \gg \delta_t$ and begin to search for the next pair. In this way, we ensure that the pairs we include in our analysis are separated by a reasonable time interval, making the assumption of statistical independence between pairs more realistic.

Following the collection of these pairs, we include any isolated packets that do not violate the time separation requirement. In general, we observe the number of such packets to be significantly larger than the number of pairs. We are now in a position to define the following statistics. For the purpose of anti-causal conditioning (see Section V), let $n_{i,j}$ denote the number of pairs wherein the first packet is sent to receiver i and the second to receiver j AND the second packet is successfully received. Let $m_{i,j}$ denote the number of these pairs in which both packets are successful. Let n_i denote the number of isolated packets sent to receiver i and let m_i denote the number successfully received. Collecting all the measurements, define:

$$\mathcal{M} \equiv \{m_i\} \cup \{m_{i,j}\} \qquad \text{and} \qquad \mathcal{N} \equiv \{n_i\} \cup \{n_{i,j}\}, \tag{1}$$

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

IV. Loss Modeling and Likelihood Analysis

The modeling frameworks of [6, 9, 10, 13] assume a simple Bernoulli loss model for each link for individual packet transmissions. The *unconditional* success probability of link i (the link into node i) is defined as

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

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 $1 - \alpha_i$. Loss processes on separate links are modeled as mutually independent.

If a back-to-back packet pair is sent from node $\rho(i)$ to node i, then we define two conditional success probabilities:

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

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

where 1st and 2nd refer to the temporal order of the two packets, and $\rho(i) \to i$ is shorthand notation denoting the successful transmission of a packet from $\rho(i)$ to i. In earlier work [9, 10], only the first (causal) conditional success probability was considered. Here we draw an important distinction between causal β_i and anti-causal γ_i conditional probabilities. We demonstrate in Section V that the anti-causal conditioning offers a significant advantage, and focus on the anti-causal case in the remainder of the paper.

The sampled TCP flows and corresponding traffic statistics lead to following likelihood function. We denote the collections of the unconditional and (anti-causal) conditional link success probabilities as $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$, respectively. The *joint* likelihood of all measurements is given by

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

where $\mathcal{B}i(m|n,p) \equiv \binom{n}{m} p^m (1-p)^{n-m}$, the binomial likelihood function, $p_i(\boldsymbol{\alpha})$ is a product the unconditional success probabilities in the path from the source to receiver i, and $p_{i,j}(\boldsymbol{\alpha}, \boldsymbol{\gamma})$ is a product of conditional success probabilities (on the common links in the paths to receivers i and j) and unconditional success probabilities on the links to j not shared in the path to i.

The EM Algorithm developed in [9] can be used to compute maximum likelihood estimates of $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$. Beginning with an initial guess 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 unobserved packet losses at internal nodes given the observed data, under the probability law induced by the current estimates of $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$. The Maximization (M) Step combines the observed (path) losses and expected unobserved (internal) losses to compute new estimates of $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$. Each iteration of the EM Algorithm is close to $O(N)$ in complexity, where N is the number of nodes in the network. The exact complexity depends on the topology of the network. For example, the complexity is $O(N \log^2 N)$ for a binary tree network topology. Our `ns-2` experiments (see Section VI) have shown that the algorithm typically converges in a small number of iterations (typically 5-15). 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.

V. Queuing Analysis of Back-to-Back Losses

Suppose that a pair of closely time-spaced packets arrives a finite length queue. Let K denote the length 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. If $k_0 = K$, then the packet is dropped (*e.g.,* droptail assumption). The probability of $k_0 = j,\ j = 0, \ldots, K$ is denoted by $p(j)$, and we define the success probability is $\alpha = \sum_{j=0}^{K-1} p(j) = 1 - p(K)$. Between the arrival of the first and second packet several arrival and/or service events may occur. Let r denote the total number of these events (both arrivals and services), and let ℓ_r denote the length of the queue immediately before the second packet arrives. Note that $\ell_0 = \min(k_0 + 1, K)$. We consider two possible conditional success probabilities.

Causal conditioning:

Probability that second packet successfully enters the queue, conditioned on the event that the first packet entered the queue. $\beta_r \equiv p(\ell_r < K \,|\, k_0 < K)$

Anti-causal conditioning:

Probability that the first packet successfully enters the queue, conditioned on the event that the second packet entered the queue. $\gamma_r \equiv p(k_0 < K \,|\, \ell_r < K)$

Previous work [9] has focused on the causal conditional probability, and follow-up work [11] has shown that $\beta_r \approx 1$ for inhomogeneous traffic (inhomogeneous Poisson arrival process), which may be typical in many cases. However, if the traffic is homogeneous traffic (fixed-rate Poisson arrival process), then $\beta_r < \alpha \leq 1$, which can degrade the performance of loss estimation algorithms. Although homogeneous traffic is rarely observed in practice, this undesirable property of causal conditioning motivates our investigation of the anti-causal case.

Notice, however, that for inference purposes, there is no problem with anti-causal conditioning, since we are interested in determining the success/loss rates responsible for a set of previous observations (*i.e.,* our inferences are made *after* the actual measurements are taken). In this paper, we emphasize the anti-causal conditional probability γ_r instead, for the following reasons. It is guaranteed that for *all possible traffic models*: **(a)** $\gamma_0 = 1$, and **(b)** $\gamma_r \geq \alpha$, for all r. Neither condition is met by β_r in general. In fact, for all cases considered in [11], $\beta_r < 1$ for all r. Thus, it appears that γ_r is a much more suitable choice for network loss tomography, as it generally provides a higher level of conditional success probability and requires no assumptions on the traffic behavior. We summarize the results in the following theorem.

Theorem 1

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

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.*

Proof: First note that

$$\begin{aligned} p(k_0 < K \,|\, \ell_r < K) &= \frac{p(k_0 < K \,,\, \ell_r < K)}{p(\ell_r < K)} \\ &= \frac{p(k_0 < K \,,\, \ell_r < K)}{p(\ell_r < K \,|\, k_0 < K)p(k_0 < K) + p(\ell_r < K \,|\, k_0 = K)p(k_0 = K)} \\ &= \frac{1}{1 + c_r}. \end{aligned}$$

Now observe that if $n = 0$ (no intervening events between packets), then $p(k_0 = K \,,\, \ell_r < K) = 0$ and thus $\gamma_0 = 0$. Also note that as r tends to infinity the effect of conditioning diminishes, and we have $c_r \rightarrow \frac{p(K)}{1-p(K)}$. Thus, $\gamma_0 = 1$ and $\rho_\infty = 1 - p(K) = \alpha$, the unconditional success probability. Next, re-express c_r as

$$c_r = \frac{p(\ell_r < K \,|\, k_0 = K)\, p(k_0 = K)}{p(\ell_r < K \,|\, k_0 < K)\, p(k_0 < K)} = \left[\frac{1 - p(\ell_r = K \,|\, k_0 = K)}{1 - p(\ell_r = K \,|\, k_0 < K)}\right] \frac{p(K)}{1 - p(K)}.$$

To show that $c_r \leq \frac{p(K)}{1-p(K)}$ we simply need to verify that $\frac{1-p(\ell_r = K \,|\, k_0 = K)}{1-p(\ell_r = K \,|\, k_0 < K)} \leq 1$. To see that this is so, consider the probabilistic behavior of the queue after the event $k_0 = K$ or $k_0 < K$. There are r (random) arrival and/or service events prior to the arrival of the second packet in the pair. Consider a realization of r such events. If, under $k_0 < K$ we find that $\ell_r = K$, then we can be certain that $\ell_r = K$ under $k_0 = K$ as well. This is true since if the number of arrivals outweigh the number of services in the former case, then starting with a full queue ($k_0 = K$) will lead to the same result. On the other hand, $\ell_r = K$ given that $k_0 = K$ does not imply that $\ell_r = K$ given $k_0 < K$. Because this holds for every realization, $p(\ell_r = K \,|\, k_0 = K) \geq p(\ell_r = K \,|\, k_0 < K)$ or conversely $p(\ell_r < K \,|\, k_0 = K) \leq p(\ell_r < K \,|\, k_0 < K)$, which finishes the proof. □

The theorem establishes that $\gamma_r \geq \alpha$ for all r, regardless of the specific traffic and queueing behavior. If one makes further assumptions on the queuing behavior (*e.g.,* an M/M/1/K model), then it is possible to determine the rate at which γ_r decays from 1 to the steady-state value α. We do not pursue this investigation further here due to space limitations.

VI. `ns`-2 Simulation Experiments

We evaluated the passive loss inference framework using the `ns-2` simulation environment. In the simulations that we perform, we strive to investigate a number of issues. We gauge the performance of the combined EM loss inference algorithm and passive measurement framework under a variety of traffic conditions and queueing policies. We also explore the measurement period required to collect a sufficient number of data for accurate inference in a passive framework.

A. Simulation Framework

Network Topology: We use the same 12-node network topology in all experiments (see Figure 1). This topology is intended to reflect (to some extent) the nature of many networks — a slower entry link from the source, a fast internal backbone, and then slower exit links to the receivers. The chosen topology gives us the flexibility to explore the effects of having receivers at different distances from the source, and to examine the effect of varying fan-outs. We fix the size of all queues to be 35 packets. We consider four different traffic scenarios and perform all experiments twice, once using the droptail queuing policy throughout the network, and once using the Random-Early-Detection (RED) policy throughout.

Traffic Generation and Statistics Collection: In all experiments, we assume that there are TCP connections to the receivers that last for the extent of the measurement period. In addition, we set up a variety of short-duration TCP sessions, both from source to receiver and as cross-traffic on internal links, as well as exponential on/off traffic sources traversing various paths. In total there are approximately thirty TCP connections and thirty UDP connections operating within the network at any one time. The average utilization of the network is in all cases relatively high; otherwise, we experience very drops and loss estimation is of little interest.

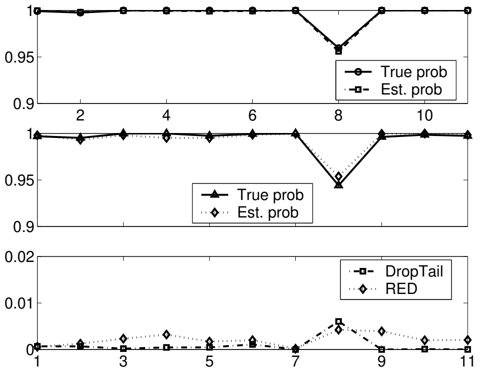
We utilize all the TCP connections flowing from the source to the receivers when collecting statistics using the procedure discussed in Section III. We set the maximum time-spacing between packets within a pair to $\delta_t = 1$ ms and the minimum spacing between pairs to $\Delta_t = 10$ ms. We collect measurements over a 300 second interval. The four traffic scenarios are described as follows.

Traffic Scenarios 1–3 (heavy losses at 1 or 2 links): In these three traffic scenarios, we strive to ascertain the capability of our passive framework to discern where significant losses are occurring within the network. We assess its ability to determine how extensive the heavy losses are and to provide accurate estimates of loss rates on the better performing links. In each case, we establish each heavy loss link by adding substantial exponential on-off and short-duration TCP session cross-traffic flows to the link traffic. In Scenario 1, link 4-8 experiences heavy losses (enabling assessment of the framework's ability to localize losses at links near the receivers). In Scenario 2, links 1-2 and 2-5 experience substantial losses (testing the framework's capacity to separate cascaded losses). In Scenario 3, links 1-2 and 4-8 experience substantial loss. This last scenario tests the ability to resolve distributed losses in different branches of the network.

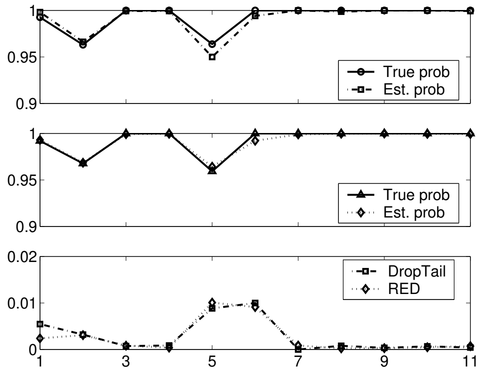
Traffic Scenario 4 (mixed traffic with medium losses): In this last scenario, we introduce many on-off UDP and on-off TCP connections throughout the topology and insert extra links (not depicted in Figure 1) connecting to the internal nodes. These links allow us to develop TCP cross-flows that have a range of different round-trip-times.

B. Simulation Results

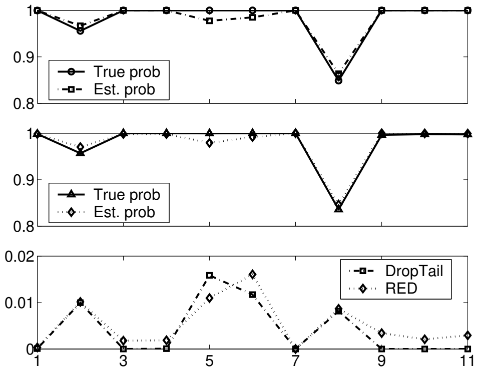
We conduct ten independent simulations of each traffic scenario and queuing policy over a measurement period of 300 seconds. Figure 2 displays the results of our simulations for each of the different traffic scenarios. We see that the estimated success rates are in good agreement with the true TCP success rates (based on direct counts of total losses on each link). In the heavy-traffic scenarios, we see that the worst-case mean absolute error is about one percent. The passive framework is thus capable of identifying where heavy losses occur. We do observe (see Figure 2 (d)) very slightly worse performance with RED in some cases, which is expected, since RED occasionally reduces the



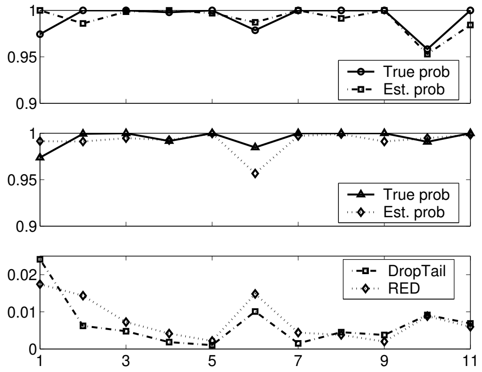
(a) Heavy losses on link 8



(b) Heavy losses on links 2 and 5



(c) Heavy losses on links 2 and 8



(d) Traffic mixture - medium losses

Figure 2: Simulation Results. True and estimated link-level success rates of TCP flows from source to receivers for all traffic scenarios. In each subfigure, the three panels display for each link 1-11 (horizontal axis): (top) true and estimated success rates using drop-tail queues, (middle) true and estimated success rates using RED queuing policy, and (bottom) mean absolute error for each link (RED and drop-tail).

degree of loss correlation.

In Figure 3, we examine the relationship between the average mean absolute error and the measurement period. As expected, the error decreases as the measurement period increases. Note that even for a 60 second measurement period, the averaged mean absolute error is less than 0.6 percent.

Finally, we draw attention to the fact that the losses we measure for the TCP flows can be very different from those of the UDP traffic. For example, in the mixed traffic scenario, we observed average TCP losses of approximately 3 percent on links 6 and 10, whereas the on-off exponential traffic experienced losses of nearly 20 percent. This is a strong indication that active probing may provide a poor indication of losses in the existing TCP traffic.

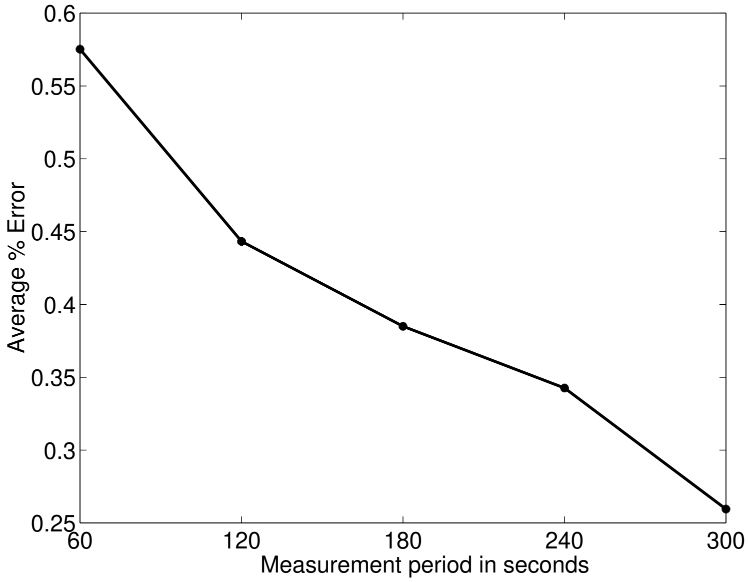


Figure 3: The performance error (mean absolute error averaged over all links) versus measurement period.

VII. Discussion

The sampling (or perhaps more aptly "mining") of the TCP traffic flows for packet-pair events is a crucial step in our methodology. As discussed in Section III, the simplest approach is simply to scan the traffic flows, locating packet-pair occurrences in a sequential fashion (beginning at the start of the measurement period). When a packet pair is located, we skip ahead Δ_t seconds and resume the scan. This results in a fast extraction algorithm, with the amount of measurements involving each receiver dependent on the throughput to that receiver. The disadvantage of this simple algorithm is that some pairs are more informative than others. Due to the nature of TCP, there are large numbers of back-to-back pairs to the same receiver (arising whenever the source receives notification that it can send a group of packets equal to the current window size). The pairs involving different receivers are rarer, occurring when the source switches from one connection to another. Because they are fewer in number, they provide more information for inference.

A potentially more effective alternative algorithm is to consider the reduced set of back-to-back "cross-pairs" that involve different receivers. We begin by locating and including all cross-pairs. The time intervals between cross-pairs are often sufficiently large so that we can include all of them without violating the requirement of a Δ_t separation. If not, we scan through the set; when we have to decide which of a set of closely-spaced cross-pairs to eliminate, we eliminate all but the pair with the least representation in the current set of included pairs. After the set of cross-pairs has been finalized, we incorporate "auto-pairs" (back-to-back packets destined for the same receiver), excepting those that violate the Δ_t time-separation criterion with the pairs already included. If we discover that this algorithm has resulted in the under-representation of some type of auto-pair,

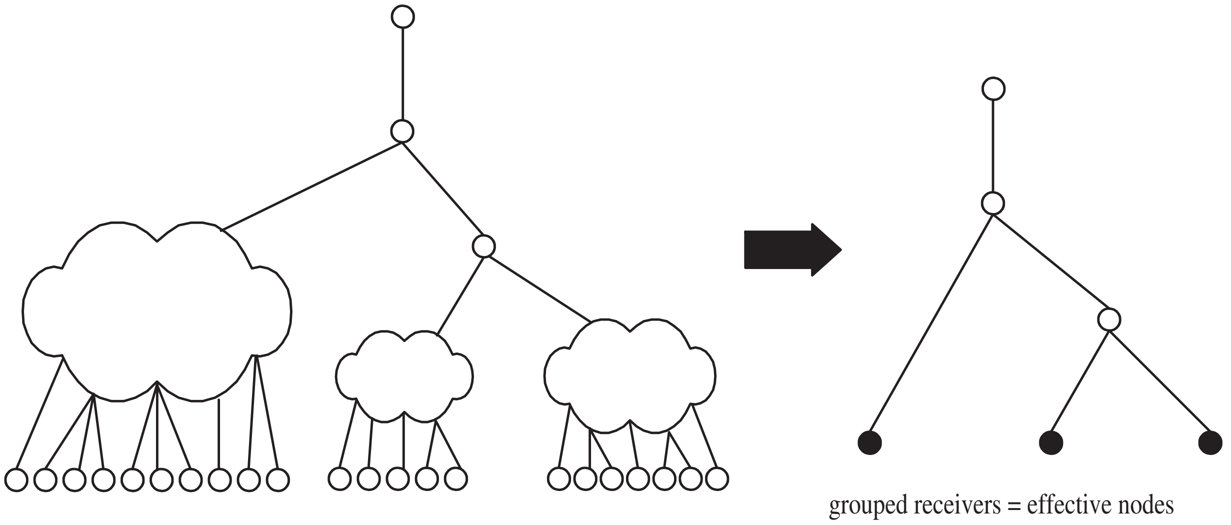


Figure 4: An example of "grouping" receivers to reduce network complexity.

then we start again, including the under-represented pair in the initial set. Finally, single (isolated) packets are included, again maintaining the Δ_t time-separation. In this way, we can extract more informative statistics from a given set of traffic data, which should lead to improved estimates. Conversely, a more effective sampling strategy of this nature should reduce the measurement time duration required for accurate loss inferences.

A more aggressive approach to obtain more informative data could involve alternative servicing strategies at the source. For example, instead of a basic round-robin service strategy, the source could employ a scheme that would enhance the chances of cross-pairs occurrences, without necessarily deviating from a TCP format. We are currently investigating this possibility as well as the more sophisticated data collection process described above.

One final point of discussion is the issue of scalability. The EM algorithm itself is reasonably scalable (approximately linear in the number of nodes), but as the number of receivers grows, the potential for a sufficient number of cross-pairs may diminish. Alternative data collection and servicing strategies, like those mentioned proposal above, could mitigate this problem. Nonetheless, the possibility of "data-starvation" may limit one's ability to passively estimate all link-level losses in very large networks. This may not be as bad as it seems. For example, rather than estimating all link-level loss rates, in many practical situations it may be sufficient to determine loss rates on a few of the first links along the paths from the source to the receivers. By grouping receivers into clusters, the actual network can be abstracted into a smaller network with "effective nodes" replacing the original clusters. This is illustrated in Figure 4, where we use a "cloud" to denote the corresponding aggregation of links (subnetwork) to the clustered receivers. Auto-pairs and cross-pairs can be shared among clustered receivers, and we should be able to reliably estimate the loss rates on the upper links (close to the source) as well as average loss rates for the subnetworks associated with the clustered receivers. We are currently investigating this approach through theoretical analysis and `ns` simulations.

VIII. Conclusions

The new passive unicast network tomography methodology we have proposed shows considerable promise. We have demonstrated 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 have observed that we are able to accurately estimate the losses experienced by existing TCP flows. As these can differ substantially from losses suffered by other forms of traffic, we surmise that in some situations inference from active probing may offer a poor reflection of existing TCP loss rates. We have conducted theoretical queuing analysis which corroborates the results obtained through simulation experiment.

Recognizing some of the potential limitations of the passive scheme (data starvation, scalability problems), we have proposed alternative data-mining and servicing strategies at the source that may provide more informative data. We also discuss a method for clustering receivers to reduce the effective complexity of the network, thus allowing us to focus on identifying losses on a subset of interesting links. Both of these are topics under current investigation.

References

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

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

[3] M. Allman and V. Paxson. On estimating end-to-end network path properties. In *Proc. Sigcomm*, 1999.

[4] J-C. Bolot. End-to-end packet delay and loss behaviour in the Internet. In *Proc. ACM Sigcomm '93*, pages 289–298, Sept. 1993.

[5] R. Cáceres, N. Duffield, J. Horowitz, F. Lo Presti, and D. Towsley. Loss-based inference of multicast network topology. In *Proc. IEEE Conf. Decision and Control*, Dec. 1999.

[6] R. Cáceres, N. Duffield, J. Horowitz, and D. Towsley. Multicast-based inference of network-internal loss characteristics. *IEEE Trans. Info. Theory*, 45(7):2462–2480, November 1999.

[7] R. Cáceres, N. Duffield, J. Horowitz, D. Towsley, and T. Bu. Multicast-based inference of network-internal characteristics: Accuracy of packet loss estimation. In *Proc. IEEE Infocom'99*, March 1999.

[8] R. Carter and M. Crovella. Measuring bottleneck link speed in packet-switched networks. Technical Report BU-CS-96-006, Computer Science Dept., Boston University, Mar. 1996.

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

[10] M. Coates and R. Nowak. Networks for networks: Internet analysis using Bayesian graphical models. to appear, *IEEE Neural Network for Signal Processing Workshop*, Dec. 2000.

[11] M. Coates and R. Nowak. Unicast network tomography using EM algorithms. Technical Report TR-0004, Rice University, Sept. 2000.

[12] N. Duffield and F. Lo Presti. Multicast inference of packet delay variance at interior network links. In *Proc. IEEE Infocom*, Mar. 2000.

[13] 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.

[14] V. Paxson. End-to-end Internet packet dynamics. *IEEE/ACM Trans. Networking*, 7(3):277–292, June 1999.