Multicast-Based Inference of Network-Internal Characteristics: Accuracy of Packet Loss Estimation

R. Cáceres N.G. Duffield J. Horowitz D. Towsley T. Bu

Abstract—**We explore the use of end-to-end multicast traffic as measurement probes to infer network-internal characteristics. We have developed in an earlier paper [2] a Maximum Likelihood Estimator for packet loss rates on individual links based on losses observed by multicast receivers. This technique exploits the inherent correlation between such observations to infer the performance of paths between branch points in the multicast tree spanning the probe source and its receivers. We evaluate through analysis and simulation the accuracy of our estimator under a variety of network conditions. In particular, we report on the error between inferred loss rates and actual loss rates as we vary the network topology, propagation delay, packet drop policy, background traffic mix, and probe traffic type. In all but one case, estimated losses and probe losses agree to within 2 percent on average. We feel this accuracy is enough to reliably identify congested links in a wide-area internetwork.**

Keywords—**Internet performance, end-to-end measurements, Maximum Likelihood Estimator, tomography**

I. Introduction

A. Background and Motivation

Fundamental ingredients in the successful design, control and management of networks are mechanisms for accurately measuring their performance. Two approaches to evaluating network performance have been (i) collecting statistics at internal nodes and using network management packages to generate link-level performance reports; and (ii) characterizing network performance based on end-to-end behavior of point-to-point traffic such as that generated by TCP or UDP. A significant drawback of the first approach is that gaining access to a wide range of internal nodes in an administratively diverse network can be difficult. Introducing new measurement mechanisms into the nodes themselves is likewise difficult because it requires persuading large companies to alter their products. Also, the composition of many such small measurements to form a picture of end-to-end performance is not completely understood.

Regarding the second approach, there has been much recent experimental work to understand the phenomenology of end-to-end performance (e.g., see [1], [3], [15], [20], [22], [23]). A number of ongoing measurement infrastructure projects (Felix [6], IPMA [8], NIMI [14] and Surveyor [31]) aim to collect and analyze end-to-end measurements across a mesh of paths between a number of hosts. `pathchar` [11] is under evaluation as a tool for inferring link-level statistics from end-to-end point-to-point measurements. However, much work remains to be done in this area.

In a recent paper [2], we considered the problem of characterizing link-level loss behavior through end-to-end measurements. We presented a new approach based on the measurement and analysis of the end-to-end loss behavior of *multicast* probe traffic. The key to this approach is that multicast traffic introduces correlation in the end-to-end losses measured by receivers. This correlation can, in turn, be used to infer the loss behavior of the links within the multicast routing tree spanning the sender and receivers. Our principal analytical tool is a *Maximum Likelihood Estimator* (MLE) of the link loss rates. This estimate is derived under the assumption that link losses are described by independent Bernoulli losses. The data for this inference is a record of which of n probes were observed at each of the receivers. We have shown that these estimates are strongly consistent (converge almost surely to the true loss rates). Moreover, the asymptotic normality property of MLEs allows us to derive an expression for their rate of convergence to the true rates as n increases. The presence of spatial and temporal correlation between losses would violate the assumptions of the model. However, we showed in [2] that spatial correlations deform the Bernoulli based estimator continuously (i.e. small correlations give rise to only small inaccuracies). Moreover, the deformation is a second order effect in that it depends only on the change in loss correlations between different parts of the network. Temporal correlations do not alter the strong consistency of the estimator; they only slow the rate of convergence.

We envisage deploying inference engines as part of a measurement infrastructure comprised of hosts exchanging probes in a wide-area network (WAN). Each host will act as the source of probes down a multicast tree to the others. A strong advantage of using multicast rather than unicast traffic is efficiency. N multicast servers produce a network load that grows at worst linearly as a function of N. On the other hand, the exchange of unicast probes can lead to local loads which grow as N^2, depending on the topology.

B. Contribution

Whereas the experimental component of our previous work focused on comparing inferred and actual probe losses, the focus of this paper is on asking how close are the inferred losses to those of background traffic. We do this under a variety of network configurations. These are specified by varying the following: (i) network topology (ii) background traffic mix (iii) packet

This work was sponsored in part by the DARPA and Air Force Research Laboratory under agreement F30602-98-2-0238.

Ramon Cáceres is with AT&T Labs–Research, Rm. B125, 180 Park Avenue, Florham Park, NJ 07932, USA; E-mail: ramon@research.att.com

Nick Duffield is with AT&T Labs–Research, Rm. B139, 180 Park Avenue, Florham Park, NJ 07932, USA; E-mail: duffield@research.attcom

Joseph Horowitz is with the Dept. of Math. & Statistics, University of Massachusetts Amherst, MA 01003-4515, USA; E-mail: joeh@math.umass.edu

Don Towsley is with the Dept. of Computer Science, University of Massachusetts, Amherst, MA 01003-4610, USA; E-mail: towsley@cs.umass.edu

Tian Bu is with the Dept. of Computer Science, University of Massachusetts, Amherst, MA 01003-4610, USA; E-mail: tbu@cs.umass.edu

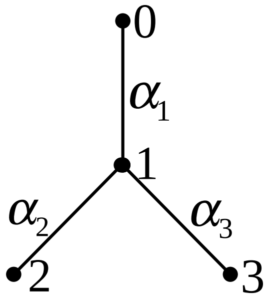

Fig. 1. A two-leaf logical multicast tree

drop policy (iv) probe traffic type, and (v) network propagation delay. In analyzing potential differences between inferred and actual losses we identify three potential causes.

The first is the statistical variability expected on the basis of the loss model. The general theory of MLE's furnished the asymptotic variance of the estimators as the number of probes grows. These tell us how many probes must be used in order to achieve measurements of a desired level of accuracy. It can be shown that the asymptotic variance of each estimated loss probability is, to first order, equal to the true loss probability and otherwise independent of the topology. The role of such theoretical values is to establish a baseline for variance of loss estimates of background traffic.

The second potential cause of differences is the non-conformance of probe losses to the Bernoulli model. In practice we find quite close agreement between inferred and actual probe losses. An examination of the underlying loss process shows that deviations from the Bernoulli model are quite small. The correlation between packet losses on different links is usually less than 0.1.

The main contribution to the difference comes from differences in the loss patterns exhibited by probe and background traffic. We have mainly used TCP background traffic in the simulations, reflecting the dominant use of TCP as a transport protocol on the Internet [32]. However, TCP flows are known to exhibit correlations. A well-known example of this is synchronization between TCP flows which can occur as a result of *slow start* after packet loss [10]. This mechanism can be expected to give rise to spatial and temporal correlations between losses. However, we believe that large and long-lasting spatial dependence is unlikely in a real network because of traffic heterogeneity. In our experiments we investigated the effects of two different discard methods: Drop from Tail and Random Early Detection (RED) [7]. One of the motivations for the introduction of RED has been to break dependence introduced through TCP.

The choice of probe process is one means by which we can aim to improve the accuracy of inference. A constraint on the interprobe time is that probe traffic should not itself contribute noticeably to congestion. Beyond the question of the mean, the choice of interarrival time distribution can affect the bias and variance of the MLE. Probes with exponentially distributed spacings will see time averages; this is the PASTA property (Poisson Arrivals See Time Averages; see e.g. [33]). This approach has been proposed for network measurements [24] and is under consideration in the IP Performance Metrics working group of the IETF [9]. We compare the effect of using constant rate probes and Poisson probes. In most cases the difference in accuracy is quite small. We find a far greater degradation in accuracy when network round trip times were reduced below the interprobe time.

The remaining sections of the paper are organized as follows. After a review of related work, in Section II we describe the loss model, in Section III the MLE and its properties. In Section IV we describe the algorithm used to compute the MLE from data. We discuss our framework for quantifying the errors in inference in Section V. The simulations themselves are reported in Section VI.

C. Related Work

In the opening paragraphs we listed a number of ongoing measurement infrastructure projects in progress ([6], [8], [14], [31]). We believe our multicast-based techniques would be a valuable addition to these measurement platforms.

Simultaneously with the present work, Ratnasamy and McCanne [26] have proposed using a multicast-based loss estimator to infer topology. The emphasis in their study is on grouping multicast receivers, rather than estimating the loss probabilities themselves. They use the same estimate as we do for loss on the shared path to two receivers, and this gives rise to an algorithm for inferring binary trees. Ad hoc extensions to trees with higher branching ratios are proposed.

There is a multicast-based measurement tool, `mtrace` [17], already in use in the Internet. `mtrace` reports the route from a multicast source to a receiver, along with other information about that path such as per-hop loss and delay statistics. Topology discovery through `mtrace` is performed as part of the `tracer` tool [13]. However, `mtrace` suffers from performance and applicability problems in the context of large-scale measurements. First, `mtrace` traces the path from the source to a single receiver by working back through the multicast tree starting at that receiver. In order to cover the complete multicast tree, `mtrace` needs to run once for each receiver, which does not scale well to large numbers of receivers. In contrast, the inference techniques described in this paper cover the complete tree in a single pass. Second, `mtrace` relies on multicast routers to respond to explicit measurement queries. Current routers support these queries. However, Internet service providers may choose to disable this feature since it gives anyone access to detailed delay and loss information about paths in their part of the network. (We have received reports that this is already occurring). In contrast, our inference techniques do not rely on cooperation from any network-internal elements.

There has been some ad hoc, statistically non-rigorous work on deriving link-level loss behavior from end-to-end multicast measurements. An estimator proposed in [34] attributes the absence of a packet at a set of receivers to loss on the common path from the source. However, this is biased, even as the number of probes n goes to infinity.

II. Description of the Loss Model

Let $\mathcal{T} = (V, L)$ denote the *logical* (as opposed to physical) multicast tree, consisting of the set of nodes V, including the source and receivers, and the set of links L, which are ordered pairs (j, k) of nodes, indicating a (directed) link from j to k. The set of *children* of node j is denoted by $d(j)$; these are the nodes with a link coming from j. For each node j, other than the root 0, there is a unique node $f(j)$, the *parent* of j, such that $j \in d(f(j))$. Each link can therefore be identified by its "child" endpoint. We define "ancestors" (grandparents and the like) in an obvious way, and likewise "descendants". The difference between a logical and a physical tree is that, whereas it is possible for a node to have only one child in the physical tree, in the logical tree each node except the root and leaves must have at least two children. A physical tree can be converted into a logical tree by deleting all nodes, other than the root, which have one child and adjusting the links accordingly.

The root $0 \in V$ represents the source of the probes and the set of *leaf* nodes $R \subset V$ (i.e., those with no children) represents the receivers.

A probe packet is sent down the tree starting at the root. If it reaches a node j a copy of the packet is produced and sent down the link toward each child of j. As a packet traverses a link k (recall that k denotes the endpoint), it is lost with probability $\overline{\alpha}_k = 1 - \alpha_k$ and arrives at k with probability α_k. We shall use the notation $\overline{\alpha} = 1 - \alpha$ for any quantity α (with or without subscripts) between 0 and 1. The losses on different links are assumed to be independent and to occur with the probabilities $\overline{\alpha}_k$ as described. In [2] we have discussed the potential limitations of this model, and how the model can be corrected if there are dependencies between the losses. The two-leaf logical multicast tree is shown in Figure 1.

We describe the passage of probes down the tree by a stochastic process $X = (X_k)_{k\in V}$ where each X_k equals 0 or 1: $X_k = 1$ signifies that a probe packet reaches node k, and 0 that it does not. The packets are generated at the source, so $X_0 = 1$. For all other $k \in V$, the value of X_k is determined as follows. If $X_k = 0$ then $X_j = 0$ for the children j of k (and hence for all descendants of k). If $X_k = 1$, then for j a child of k, $X_j = 1$ with probability α_j, and $X_j = 0$ with probability $\overline{\alpha}_j$, independently for all the children of k. We write $\alpha_0 = 1$ to simplify expressions concerning the α_k.

III. Maximum Likelihood Estimation of Loss

If a probe is sent down the tree from the source, the outcome is a record of whether or not a copy of the probe was received at each receiver. Expressed in terms of the process X, the outcome is a configuration $X_{(R)} = (X_k)_{k\in R}$ of zeroes and ones at the receivers (1 = received, 0 = lost). Notice that only the values of X at the receivers are observable; the values at the internal nodes are invisible. The state space of the observations $X_{(R)}$ is thus the set of all such configurations, $\Omega = \{0, 1\}^R$. For a given set of link probabilities $\alpha = (\alpha_k)_{k\in V}$, the distribution of $X_{(R)}$ on Ω will be denoted by P_α. The probability mass function for a single outcome $x \in \Omega$ is $p(x; \alpha) = \mathsf{P}_\alpha(X_{(R)} = x)$.

Let us dispatch n probes, and, for each $x \in \Omega$, let $n(x)$ denote the number of probes for which the outcome x is obtained. The probability of n independent observations $x^1, \ldots, x^n$ (with each $x^m = (x^m_k)_{k\in R}$) is then

$$p(x^1, \ldots, x^n; \alpha) = \prod_{m=1}^{n} p(x^m; \alpha) = \prod_{x\in\Omega} p(x; \alpha)^{n(x)} \qquad (1)$$

We estimate α using maximum likelihood, based on the data $(n(x))_{x\in\Omega}$, and we find that the usual regularity conditions that imply good large-sample behavior of the MLE are satisfied in the present situation. This is useful for the applications we have in mind because (a) we want to assess the accuracy of our estimates via confidence intervals, and (b) it is important to determine the smallest number n of probes needed to achieve the desired accuracy. We want to minimize n because, although sending out probes is inexpensive in itself, networks are subject to various fluctuations (e.g., [20]) which can perturb the model, and the measurement process itself ties up network resources.

We begin with a review of our main results on the existence and uniqueness of the MLE. Another question, not treated here, but which is important for applications, is the feasibility and organization of the computations. We work with the log-likelihood function

$$\mathcal{L}(\alpha) = \log p(x^1, \ldots, x^n; \alpha) = \sum_{x\in\Omega} n(x) \log p(x; \alpha). \qquad (2)$$

In the notation we suppress the dependence of $\mathcal{L}$ on n and $x^1, \ldots, x^n$. For each node k, let $\Omega(k)$ be the set of outcomes $x \in \Omega$ such that $x_j = 1$ for at least one receiver $j \in R$ which is a descendant of k, and let $\gamma_k = \Gamma_k(\alpha) := \mathsf{P}_\alpha[\Omega(k)]$. An estimate of γ_k is

$$\widehat{\gamma}_k = \sum_{x\in\Omega(k)} \widehat{p}(x), \qquad (3)$$

where $\widehat{p}(x) := n(x)/n$ is the observed proportion of trials with outcome x. We will show how to find α as a function of the γ. The MLE $\breve{\alpha}$ is precisely that α which maximizes $\mathcal{L}(\alpha)$:

$$\breve{\alpha} = \arg\max_{\alpha\in[0,1]^L} \mathcal{L}(\alpha) \qquad (4)$$

We shall see that, at least for large n, $\breve{\alpha} = \Gamma^{-1}(\widehat{\gamma})$, using the inverse of the function Γ that expresses the γ_k in terms of the α_k. Candidates for the MLE are solutions $\widehat{\alpha}$ of the *likelihood equation*:

$$\frac{\partial\mathcal{L}}{\partial\alpha_k}(\alpha) = 0, \quad k \in U. \qquad (5)$$

Set $\mathcal{A} = \{(\alpha_k)_{k\in U} : \alpha_k > 0\}$, and $\mathcal{G} = \{(\gamma_k)_{k\in U} : \gamma_k > 0\,\forall k;\ \gamma_k < \sum_{j\in d(k)} \gamma_j\ \forall k \in U \setminus R\}$.

Theorem 1: When $\widehat{\gamma} \in \mathcal{G}$, the likelihood equation has the unique solution $\widehat{\alpha} := \Gamma^{-1}(\widehat{\gamma})$ that can be expressed as follows. Define $(\widehat{A}_k)_{k\in V}$ for the root node by $\widehat{A}_0 = 1$, for leaf nodes $k \in R$ by $\widehat{A}_k = \widehat{\gamma}_k$, and for all other nodes $k \in U \setminus R$ as the unique solution in $(0, 1]$ of

$$1 - \widehat{\gamma}_k/\widehat{A}_k = \prod_{j\in d(k)} (1 - \widehat{\gamma}_j/\widehat{A}_k). \qquad (6)$$

Then for $k \in U$, $\widehat{\alpha}_k = \widehat{A}_k/\widehat{A}_{f(k)}$.

The form (6) follows from the corresponding relations that express γ_k in terms of $A_k := \alpha_k \alpha_{f(k)} \dots \alpha_0$.

We complete the picture by showing that the solution of the likelihood equation actually maximizes the likelihood function under some additional conditions. The set $\mathcal{A}$ contains all positive α_k, including the possibility $\alpha_k > 1$. Let us now restrict our attention to link probabilities $\alpha \in \mathcal{B} = (0,1)^{\#R} \subset \mathcal{A}$. Being a solution of the likelihood equation does not preclude $\widehat{\alpha}$ from being either a minimum or a saddlepoint for the likelihood function, with the maximum falling on the boundary of $\mathcal{B}$. For some simple topologies we are able to establish directly that $\mathcal{L}(\alpha)$ is (jointly) concave in the parameters at $\alpha = \widehat{\alpha}$, which is hence the MLE $\breve{\alpha}$. For more general topologies we use general results on maximum likelihood to show that $\widehat{\alpha} = \breve{\alpha}$ for all sufficiently large n.

Theorem 2:
(i) The model is identifiable in $\mathcal{B}$, i.e., $\alpha, \alpha' \in \mathcal{B}$ and $\mathsf{P}_\alpha = \mathsf{P}_{\alpha'}$ implies $\alpha = \alpha'$. Thus, distinct link probabilities α produce distinct statistical behavior of the $\widehat{\gamma}$ as $n \to \infty$.
(ii) As $n \to \infty$, $\breve{\alpha} \to \alpha$, with P_α- probability 1, i.e., the MLE is strongly consistent.
(iii) With probability 1, for sufficiently large n, $\breve{\alpha} = \widehat{\alpha}$, i.e., the solution of the likelihood equation maximizes the likelihood.
This is proven using large sample theory for MLE, such as in [30]. Finally we have a result on asymptotic normality of the MLE. The *Fisher Information Matrix* at α based on $X_{(R)}$ is the matrix $\mathcal{I}_{jk}(\alpha) := \mathsf{Cov}\left(\frac{\partial \mathcal{L}}{\partial \alpha_j}(\alpha), \frac{\partial \mathcal{L}}{\partial \alpha_k}(\alpha)\right)$.

Theorem 3: $\mathcal{I}(\alpha)$ is non-singular, and as $n \to \infty$, under P_α, $\sqrt{n}(\widehat{\alpha} - \alpha)$ converges in distribution to a multivariate normal random vector with mean vector 0 and covariance matrix $\mathcal{I}^{-1}(\alpha)$.

Example: MLE for the Two-Leaf Tree. Denote the 4 points of $\Omega = \{0,1\}^2$ by $\{00, 01, 10, 11\}$. Then

$$\widehat{\gamma}_1 = \widehat{p}(11) + \widehat{p}(10) + \widehat{p}(01), \tag{7}$$

$$\widehat{\gamma}_2 = \widehat{p}(11) + \widehat{p}(10), \quad \widehat{\gamma}_3 = \widehat{p}(11) + \widehat{p}(01), \tag{8}$$

and equations (6) for $\widehat{A}_k$ in terms of the $\widehat{\gamma}_k$ yield

$$\widehat{\alpha}_1 = \frac{\widehat{\gamma}_2 \widehat{\gamma}_3}{\widehat{\gamma}_2 + \widehat{\gamma}_3 - \widehat{\gamma}_1} \tag{9}$$

$$= \frac{(\widehat{p}(01) + \widehat{p}(11))(\widehat{p}(10) + \widehat{p}(11))}{\widehat{p}(11)} \tag{10}$$

$$\widehat{\alpha}_2 = \frac{\widehat{\gamma}_2 + \widehat{\gamma}_3 - \widehat{\gamma}_1}{\widehat{\gamma}_3} = \frac{\widehat{p}(11)}{\widehat{p}(01) + \widehat{p}(11)} \tag{11}$$

$$\widehat{\alpha}_3 = \frac{\widehat{\gamma}_2 + \widehat{\gamma}_3 - \widehat{\gamma}_1}{\widehat{\gamma}_2} = \frac{\widehat{p}(11)}{\widehat{p}(10) + \widehat{p}(11)} \tag{12}$$

Note that although it is possible that $\widehat{\alpha}_1 > 1$ for some finite n, this will not happen when n is sufficiently large, due to Theorem 2.

```
procedure main ($k$ ) {
    find_x ( $k$ ) ;
    infer ( $k$, $1$ ) ;
}

procedure find_x ( $k$ ) {
    foreach ( $j \in d(k)$ ) {
        $\widehat{X}_j$ = find_x ( $j$ ) ;
        foreach ( $i \in \{1, \dots, n\}$ ) {
            $\widehat{X}_k[i]$ = $\widehat{X}_k[i] \vee \widehat{X}_j[i]$ ;
        }
    }
    $\widehat{\gamma}_k$ = $n^{-1} \sum_{i=1}^n \widehat{X}_k[i]$ ;
    return $\widehat{X}_k$ ;
}

procedure infer ( $k$, $A$ ) ;
    $A_k$ = solvefor( $A_k$ , $(1 - \frac{\widehat{\gamma}_k}{A_k})$ == $\prod_{j \in d(k)} (1 - \frac{\widehat{\gamma}_j}{A_k})$ );
    $\widehat{\alpha}_k$ = $A_k / A$ ;
    foreach ( $j \in d(k)$ ) {
        infer ( $j$ , $A_k$ ) ;
    }
}
```

Fig. 2. PSEUDOCODE FOR INFERENCE OF LINK PROBABILITIES

IV. COMPUTATION OF THE MLE ON A GENERAL TREE

In this section we describe the algorithm for computing $\widehat{\alpha}$ on a general tree. An important feature of the calculation is that it can be performed recursively on trees. First we show how to calculate the $\widehat{\gamma}_k$. These can be calculated by reconstruction of a sample path of the full process $(X_k)_{k \in V}$ that is consistent with the measured data $X^1_{(R)}, \dots, X^n_{(R)}$ from n probes. We define the n-element binary vector $(\widehat{X}_k)_{k \in V}$ recursively by

$$\widehat{X}_k = X_k, \quad k \in R \tag{13}$$

$$\widehat{X}_k(i) = \bigvee_{j \in d(k)} \widehat{X}_j(i), \quad k \in V \setminus R \tag{14}$$

so that

$$\widehat{\gamma}_k = n^{-1} \sum_{i=1}^{n} \widehat{X}_k(i). \tag{15}$$

For simplicity we assume now that $\widehat{\gamma} \in \Gamma((0,1)^{\#V})$. The calculation of $\widehat{\alpha}$ can be done by another recursion. We formulate both recursions in pseudocode in Figure 2. The procedure `find_x` calculates the $\widehat{X}_k$ and $\widehat{\gamma}_k$, assuming $\widehat{X}_k$ initializes to X_k for $k \in R$ and 0 otherwise. The procedure `infer` calculates the $\widehat{\alpha}_k$. The procedures could be combined. The full set of link probabilities is estimated by executing `main(1)`; recall 1 is the single descendant of the root node 0. Here, an empty product (which occurs when the first argument of `infer` is a leaf node) is understood to be zero. Here `solvefor` is a routine that finds the unique solution $\widehat{A}_k$ in $(0, 1]$ to (6).

The recursive nature of the algorithm has important consequences for its implementation in a network setting. The calculation of $\widehat{\gamma}_k$ and A_k depends on X only through the $(\widehat{X}_j)_{j \in d(k)}$. In a networked implementation this would enable the calculation to be localized in subtrees at a representative node. The computational effort at each node would be at worst proportional to the depth of the tree (for the node which is unlucky enough to be the representative for all distinct subtrees to which it belongs). The network load induced by the communication of data could be kept local, e.g., by scoped multicast amongst sibling representatives.

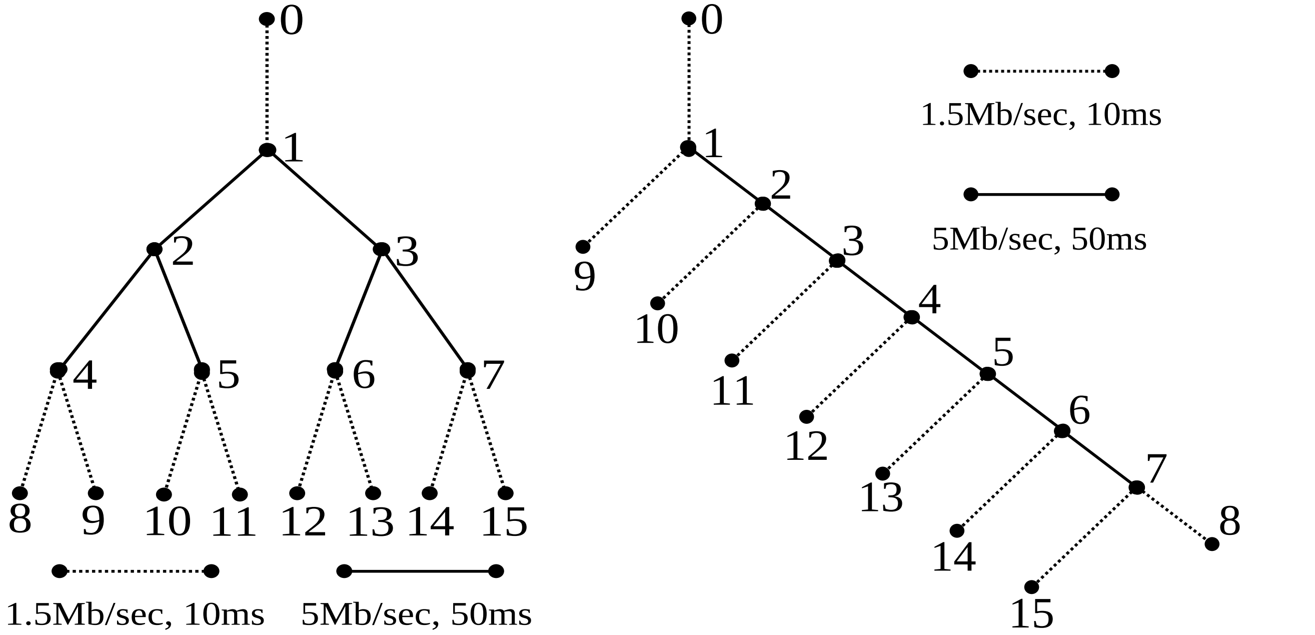

Fig. 3. SIMULATION TOPOLOGY: Links are of two types: "edge" links of 1.5Mb/s capacity and 10ms latency, and interior links of 5Mb/s capacity and 50ms latency. LEFT: "regular" topology with branching ratio 2. RIGHT: "irregular" topology.

V. FRAMEWORK FOR SIMULATION STUDY

We evaluated our loss inference algorithm using the ns simulator [19]. This enabled us to investigate the effectiveness of the estimator over a range of network topologies, link delays, packet drop policies, background traffic types, and probe traffic types. In particular we were able to determine the actual loss experienced by background traffic, and by probe traffic, and compare these values to those predicted by the inference algorithm on the basis of measurements at the leaf nodes. The experiments show that the agreement between inferred and probe loss is extremely good. This shows that the model of probe loss and the associated inference technique are quite effective in the small networks used in the simulation. This is encouraging since we expect flow synchronization effects (that would violate the model) to be more noticeable amongst a smaller numbers of flows. Agreement between inferred loss and background traffic loss is quite reasonable, although not as close as between inferred and probe loss. Some difference is expected due to the difference in temporal statistics of TCP flows and probes.

A. *Comparing Loss Probabilities*

We describe our approach to comparing two sets of loss probabilities p and q. For example p could be an inferred probability on a link, q the corresponding actual probability. For some **error margin** $\varepsilon > 0$ we define the **error factor**

$$F_\varepsilon(p,q) = \max\left\{\frac{p(\varepsilon)}{q(\varepsilon)}, \frac{q(\varepsilon)}{p(\varepsilon)}\right\} \qquad (16)$$

where $p(\varepsilon) = \max\{\varepsilon, p\}$ and $q(\varepsilon) = \max\{\varepsilon, q\}$. Thus, we treat p and q as being not less than ε, and having done this, the error factor is the maximum ratio, upwards or downwards, by which they differ. Unless otherwise stated, we used the default value $\varepsilon = 10^{-3}$ in this paper. The choice of this metric is motivated by the expectation that it is desirable to estimate the relative magnitude of loss ratios on different links in order to distinguish those which suffer higher loss. In summarizing the relative accuracy of a set of loss measurements, we will calculate statistics of the error factor, such as mean and quantiles of $F_\varepsilon(p_i, q_i)$ where $p = (p_i)$ and $q = (q_i)$ are two sets of loss probabilities (inferred and actual, say). Here the index i runs over a set of links, a set of measurements on the same link made at different times or during different simulations, or some combination of these.

B. *Summary Statistics of the Error Factor*

In describing the mean and variability of the error factors, we shall use the following summary statistics. We shall estimate the center of the distribution of a set of error factors x_i by the two-sided quartile-weighted median

$$m(\{x\}) := (Q_{.25} + 2Q_{.5} + Q_{.75})/4 \qquad (17)$$

where Q_p denotes the $p^{\rm th}$ quantile of the x_i. m is particularly suited to skewed distributions; see [29] for further detail. We characterize the high values of the error factors through the $90^{\rm th}$ percentile. Both these summary statistics are robust, being independent of any assumption on the distribution of the error factors.

C. *Experimental Variables*

We explored the performance of the inference algorithm under variation of the following quantities.

C.1 Network Topology

We investigated three topologies. We used the two-leaf binary tree of Figure 1 to explore the variables listed below within a tightly controlled environment. We also explored two larger binary topologies: the regular 8 leaf binary tree of Figure 3(left), and the irregular tree of Figure 3(right). In both of the larger trees we arranged for some heterogeneity between the edges and the center in order to mimic the difference between the core

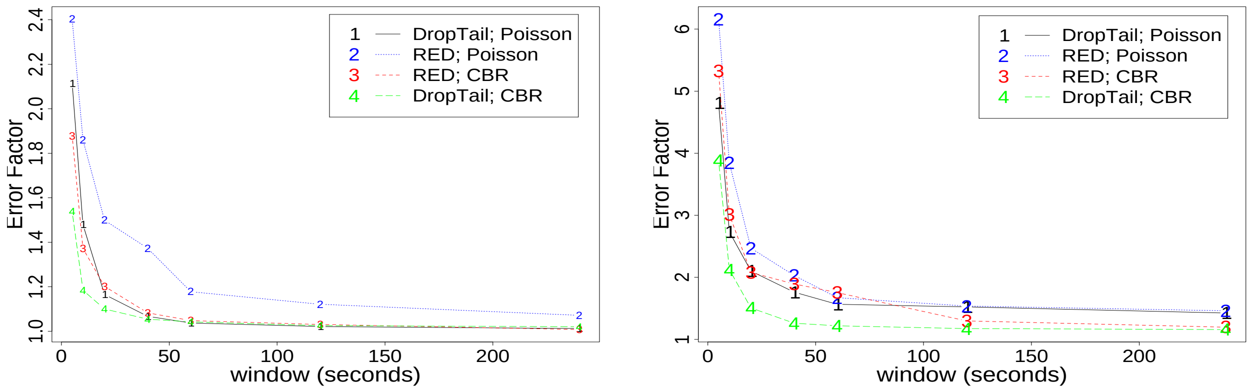

Fig. 4. ACCURACY OF INFERENCE VS. SAMPLE WINDOW: Mean error factor over all links and windows of regular topology in Figure 3(left) for RED or DropTail queueing; Poisson or CBR probes. LEFT: inferred loss vs. probe loss. RIGHT: inferred loss vs background loss. Probe bytes are 1.8% of total; average utilization is 60%.

and edges of a large WAN, with the interior of the tree having higher capacity (5Mb/sec) and latency (50ms) than at the edge (1Mb/sec and 10ms).

C.2 Packet Discard Method

Each node had a buffer capacity of 20 packets, independent of packet size. We compare the effects of two methods of packet discard: Drop from Tail (DT), and discard based on Random Early Detection (RED) [7]. One of the benefits expected from the deployment of RED is increased utilization through the breaking of synchronization that can occur due to slow start of TCP after congestion, as identified in [10]. We used the ns default parameters of RED in the simulations.

C.3 Background Traffic

Each of the trees was equipped with a variety of flows of background traffic. Flows were of two types: infinite data sources that use the Transmission Control Protocol (TCP), and on-off sources using the Unreliable Datagram Protocol (UDP), the on and off periods having either a Pareto or an exponential distribution. In most of the simulations on the larger trees we used predominantly TCP, with a mixture of UDP. We chose this mix because TCP is the dominant transport protocol on the Internet [32].

C.4 Probe Characteristics

It is desirable that probe traffic only use a small part of the available link capacity. For the experiments in the large topologies we used 40-byte probes with a mean interprobe time of 16ms, i.e. a 20 kbit/sec stream. This is just over 1% of the capacity of the smallest link used; it would be a far smaller fraction of capacities commonly used in today's Internet backbones. We used two types of probes: constant rate probes and Poisson probes. The use of the latter has been proposed [24] for end-to-end measurements on the basis that Poisson Arrivals See Time Averages; see e.g. [33].

C.5 Relative Time Scales

We investigated the effects of network roundtrip time on estimator accuracy. This is potentially important because the roundtrip time determines the time it takes TCP to respond to packet losses. Thus the relative size of this time and the interprobe time determines the number of probe packets that sample congestion due to TCP traffic. In these experiments we reverted to a uniform link latency of between 1ms and 100ms.

VI. SIMULATION RESULTS

A. *Qualitative Sample Path Behavior*

We start by illustrating some properties of sample paths of the MLE. We shall make mostly qualitative observations initially; quantitative statistical measures of the accuracy of inference will be applied later.

In the regular topology of Figure 3(left) we conducted experiments of 240 seconds duration. Background traffic was generated by 30 infinite FTP sources using TCP, and another 30 on-off UDP sources, mostly with low rates and either exponential or Pareto distributed. There was one experiment for each of the four combinations of DropTail or RED packet discard and Poisson or CBR probes. The mean time between probes was 16ms, so about 15,000 probes were used in each experiment. For each of the experiments we calculated $\widehat{\alpha}$ on a moving window of a given width, using jumps of half the width. We display the mean error factor as a function of window size in Figure 4. On the left we show the error factor between inferred and actual probe loss; on the right between inferred and actual background loss. The main points to observe are that (i) error factors decrease as window size increases; (ii) the error factor between inferred and probe losses is small when compared with that between inferred and background losses; (iii) the error factors are reasonably insensitive to choice of packet discard method and probe type. To the extent that there are differences, mean error factors between inferred and background losses for CBR probes are slightly smaller than for Poisson probes, at least for larger window sizes (about 1.2 compared with about 1.5). Error factors for RED are marginally worse than for DropTail. We shall comment upon these differences later.

B. *Dynamic Tracking of Loss*

In Figure 5 we display the time series of background, probe and inferred loss on one link over the moving windows of a sim-

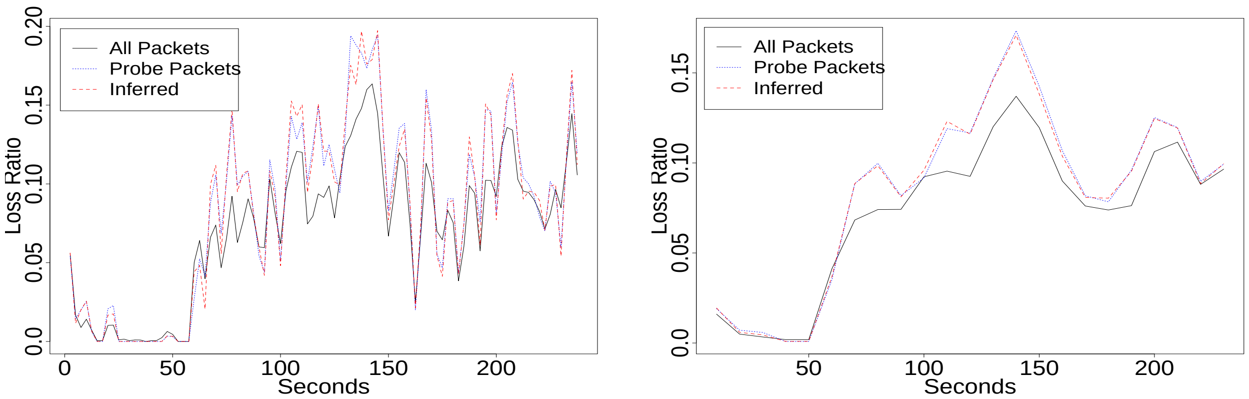

Fig. 5. Dynamic Accuracy of Inference: Loss rates of background packets, probe packets and inferred on link 8 in regular topology in Figure 3(left) for RED queueing and Poisson probes. Left: 5 second window. Right: 20 second window. Additional sources started at 60 seconds; note tracking by estimator of induced congestion. Probe bytes are 2% of total on 1.5Mb/s link with 60% utilization.

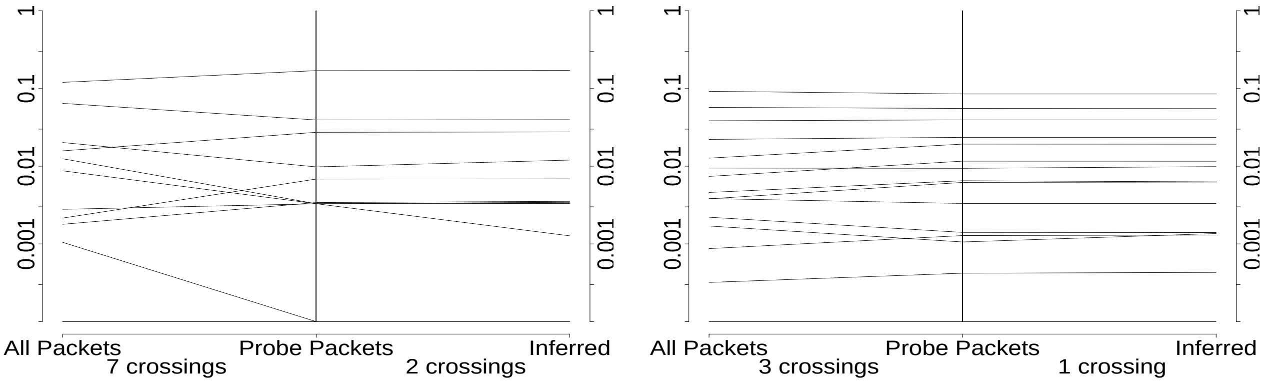

Fig. 6. Accuracy and Ordering of Inference vs. Sample Window: Loss rates in regular topology of Figure 3(left) for RED queueing and Poisson probes. Left: 5 second window. Right: 240 second window. Lines join probabilities of a given link. Fewer crossings indicate better preservation of order between actual and estimated probabilities. Flatter lines indicate better accuracy of estimates. Probe bytes are 3% of total on 1.5Mb/s link with 50% utilization.

ulation similar to that just described. However, we arrange for some additional sources to be turned on after 60 seconds have elapsed. We display how inferred losses track the real ones on a 5 second window (left) and a 20 second window (right). There is considerable variability between the inferred and actual loss at the 5 second window, not all of which is removed by increasing to a 20 second window. However, even at the 5 second window it appears that the estimator responds rapidly to the increase in actual loss that occurs after 60 seconds have elapsed.

From Figure 5 it is evident that the inferred loss tracks the probe loss more closely than the loss of background packets. Increasing the window size narrows some of the difference. We illustrate this for a single window in Figure 6. For a 5 second and a 240 second window, we display how the ordering of the links according to loss probability differs according to whether the loss used for ordering is that for background or probe or inferred loss. To do this we have placed each set of probabilities on an axis (background loss on left, probe loss in middle and inferred loss on right) and joined the values for given links. The flatter the lines, the greater the accuracy; the less they cross, the better the ordering is preserved. In this example, both accuracy and ordering are improved by using the larger window. It is clear in this example that despite error factors of about 2 between some of the inferred and background traffic losses, the inference is sufficiently accurate to distinguish the links with the highest loss for either probe or background packets.

C. *Quantitative Statistical Measures of Accuracy*

We now present some broad statistical measures of the accuracy of the inference in different network configurations in topologies with 15 links. We conducted 10 experiments of 240 seconds duration for each of the four combinations of DropTail or RED packet discard with CBR or Poisson probes. We then calculated the center m and 90^{th} percentile of the 150 error factors (10 experiments × 15 links).

The results are tabulated for the regular topology with mixed TCP and UDP sources in Table I; for the regular topology with TCP source only in Table II; and for the irregular topology with mixed sources in Table III. Taking these as a group, the accuracy of inference of probe loss is striking. Looking at the first pair of columns in each table we see that the error is no more than 2% of the true value on average (i.e. an error factor 1.02), the 90^{th} percentile of the error being 17% of the true value at worst (i.e. an error factor 1.17).

The error factors between actual probe loss and background traffic loss are somewhat larger; this difference is then the main contribution to errors in inferring the background traffic loss by the probe loss. The center m is less than 1.5, and the 90^{th} per-

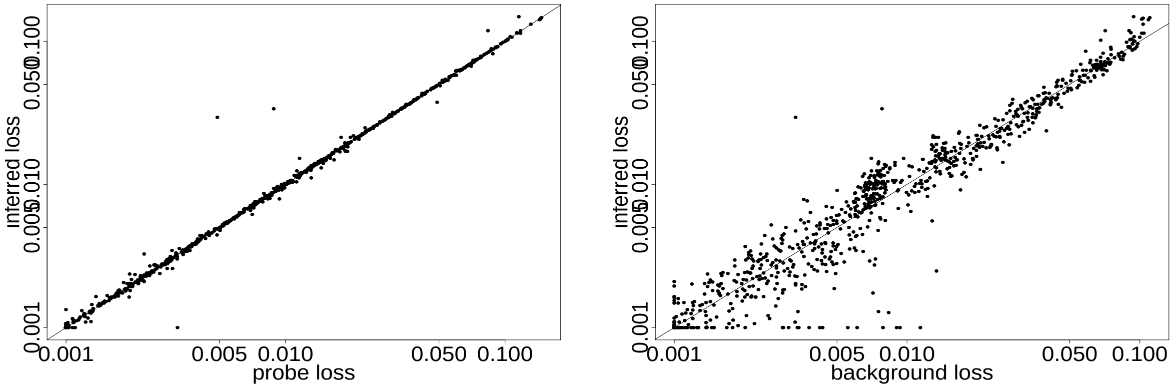

Fig. 7. ESTIMATOR ACCURACY: Scatter plots of 1110 pairs of loss probabilities gathered from all simulations: LEFT: inferred loss vs. probe loss; RIGHT: inferred loss vs. background loss. All probabilities truncated with error margin $\varepsilon = 10^{-3}$.

Discard Method	Probe Type	inf vs. probe		probe vs. b'grnd		inf vs. b'grnd	
		m	$Q_{.9}$	m	$Q_{.9}$	m	$Q_{.9}$
DT	PP	1.01	1.07	1.21	1.58	1.23	1.68
DT	CBR	1.00	1.03	1.11	1.43	1.11	1.43
RED	PP	1.01	1.04	1.14	1.54	1.15	1.56
RED	CBR	1.00	1.03	1.10	1.36	1.09	1.39

TABLE I

STATISTICS OF ERROR FACTOR VS. PACKET DISCARD AND PROBE METHOD. TCP and UDP background traffic. Regular Topology. Weighted Median and 90$^{\rm th}$ percentile of error factor over all links during 10 simulations of 240 seconds. Error margin was $\varepsilon = 10^{-3}$. In about 20% of cases, one or both probabilities compared were less than ε.

Discard Method	Probe Type	inf vs. probe		probe vs. b'grnd		inf vs. b'grnd	
		m	$Q_{.9}$	m	$Q_{.9}$	m	$Q_{.9}$
DT	PP	1.02	1.11	1.47	2.03	1.45	2.19
DT	CBR	1.01	1.06	1.31	1.82	1.33	1.81
RED	PP	1.01	1.06	1.42	1.92	1.43	1.91
RED	CBR	1.01	1.03	1.19	1.55	1.20	1.53

TABLE II

STATISTICS OF ERROR FACTOR VS. PACKET DISCARD AND PROBE METHOD. TCP background traffic only. Regular Topology. Weighted Median and 90$^{\rm th}$ percentile of error factor over all links during 10 simulations of 240 seconds. Error margin was $\varepsilon = 10^{-3}$. In about 15% of cases, one or both probabilities compared were less than ε.

Discard Method	Probe Type	inf vs. probe		probe vs. b'grnd		inf vs. b'grnd	
		m	$Q_{.9}$	m	$Q_{.9}$	m	$Q_{.9}$
DT	PP	1.02	1.17	1.34	1.83	1.39	2.24
DT	CBR	1.02	1.11	1.24	1.66	1.27	1.84
RED	PP	1.01	1.13	1.18	1.62	1.23	1.74
RED	CBR	1.01	1.08	1.13	1.54	1.17	1.61

TABLE III

STATISTICS OF ERROR FACTOR VS. PACKET DISCARD AND PROBE METHOD. TCP and UDP background traffic. Irregular Topology. Mean and 90$^{\rm th}$ percentile of error factor over all links during 10 simulations of 240 seconds. Error margin was $\varepsilon = 10^{-3}$. In no more than 8% of cases, one or both probabilities were less than ε.

Link Delay	inf vs. probe		probe vs. b'grnd		inf vs. b'grnd	
	m	$Q_{.9}$	m	$Q_{.9}$	m	$Q_{.9}$
100ms	1.00	1.04	1.07	1.45	1.07	1.44
30ms	1.00	1.02	1.17	1.54	1.17	1.53
10ms	1.09	1.71	1.28	1.88	1.19	1.49
1ms	1.49	6.83	1.30	1.61	1.71	5.07

TABLE IV

STATISTICS OF ERROR FACTOR VS. LINK DELAY. TCP and UDP background traffic. Regular Topology. DropTail with Poisson Probes. Weighted Median and 90$^{\rm th}$ percentile of error factor over all links during 10 simulations of 240s for each delay value. One or both probabilities compared were less than error margin $\varepsilon = 10^{-3}$ in up to 40% of cases.

centile is less than 2.2. Pure TCP background traffic has somewhat higher error factors than mixed TCP and UDP. The irregular topology has somewhat higher error factors than the regular topology. The average utilization in these simulations was about 60%. We also conducted simulations at up to 90% utilization on the two-leaf binary tree with approximately the same number of probes. In most cases the summary statistics were of the same order.

Comparing the different packet discard methods, we see that RED always gives somewhat lower values for m and the 90$^{\rm th}$ percentile than the corresponding DropTail. This fits with our expectation that the randomization induced by RED will break correlations induced by TCP flow control, and hence cause patterns of loss for background traffic to more closely resemble the Bernoulli loss model.

Comparing the different packet probe types, we see that CBR has m and 90$^{\rm th}$ percentile consistently slightly lower than for Poisson probes. The reason for this small difference is not clear at present. Poisson probes see time averages [33] and hence yield unbiased measurements. It is possible though that they exhibit higher variances for the reason that the potentially extreme (long or short) interarrival times lead to worse sampling of network congestion events.

We examined the influence of network propagation delay on error factors. For DropTail packet discard and Poisson queueing, we find (see Table IV) that error factors increase as propagation delay decreases. A possible explanation for this is the following. We observe an increase in utilization as the propagation delay is decreased, the utilization being close to 100% on some links when propagation delay is 1ms. Since recovery after TCP losses will be correspondingly quick, any spare capacity will be rapidly exploited, and congestion may be long lived, leading to temporal correlations between probe losses. Whereas this would not alter the asymptotic accuracy of the MLE, it would slow the rate of convergence as the number of probes is increased, leading to high estimator variance. This hypothesis is supported by

Table IV: at 1ms feedback delay, most of the error is between the inferred and probe loss. 1ms is far shorter than the minimum link propagation delays on the Internet, so we do not expect this phenomenon to occur in practice. We stress, however, that it remains to obtain a full understanding of the effect on accuracy of the interactions between interprobe time, propagation delay and variables such as packet discard method and probe type.

We summarize all our experiments hitherto in Figure 7, where we show a scatter plot of pairs of (inferred loss, probe loss) on the left, and pairs of (inferred loss, background loss) on the right. Thus each point corresponds to a single link on a single simulation run. Also included here are points for experiments conducted with the combinations of traffic types, discard method, probe distribution and topology described above, but with a more variable flow duration. The flow durations were obtained by choosing random beginning and end times for each flow in a given simulation, rather than having the flows present for the whole simulation. In these examples, inferred loss is a better predictor of background loss when the latter is at least 1%: for this subset of data points the mean error factor is 1.20 compared with 1.28 for the complete set.

VII. Conclusions

In this paper we have analyzed the efficacy of multicast-based inference in estimating loss probabilities in the interior of a network from end-to-end measurements. The principal tool was a Maximum Likelihood Estimator of the link loss probabilities. Probes are multicast from a source; the data for the MLE is a record of which probes were received at each leaf of the multicast tree. Although the method assumes that losses are independent, we have shown in some cases that it is relatively insensitive to the presence of spatial loss correlations; temporal correlations increase its variance, so that a longer measurement period is required; see [2].

We evaluated the method by conducting `ns` simulations that used topologies and traffic flows with quite a rich structure, with several hops per flow and flows per link. We compare inferred and actual loss probabilities on the links of the logical multicast tree. The experiments showed that the loss probabilities for probe packets were inferred extremely closely by the MLE.

The probe traffic was typically only 1% to 2% of the traffic on each link. We investigated how closely loss rates for background traffic were inferred. We examined the effect of changing traffic mix, topology, packet discard method and probe type. We found small differences between these, compared with the inherent variability of the estimates. Varying the network feedback delay also affected the accuracy of inference. For very short propagation delays we believe that the aggressive behavior of TCP slow start is a factor in decreasing accuracy. We intend to investigate this phenomenon more fully.

Over a range of experiments our summary statistics show that the relative error of the inferred and actual losses had a distribution whose center was no greater than about 1.5 and whose 90^{th} percentile was no worse than a factor of about 2.2. If one is limited to using inferred probe loss to estimate background traffic loss, this would mean that only in 1% of the worst cases would a *single* inference fail to distinguish between two background loss rate that separated by a factor of 5. We believe that this is sufficiently accurate to identify the most congested links.

References

[1] J-C. Bolot and A. Vega Garcia "The case for FEC-based error control for packet audio in the Internet" ACM Multimedia Systems, to appear.

[2] R. Cáceres, N.G. Duffield, J. Horowitz, D. Towsley, "Multicast-based inference of network-internal characteristics", Comp. Sci. Tech. Rep. 98-17, University of Massachusetts at Amherst, February 1998. `ftp://gaia.cs.umass.edu/pub/CDHT98:MINC.ps.Z`

[3] R. L. Carter and M. E. Crovella, "Measuring Bottleneck Link Speed in Packet-Switched Networks," *PERFORMANCE '96*, October 1996.

[4] A. Dembo and O. Zeitouni, "Large deviations techniques and applications", Jones and Bartlett, Boston, 1993.

[5] B. Effron and D.V. Hinkley, "Assessing the accuracy of the maximum likelihood estimator: Observed versus expected Fisher information", Biometrika, 65, 457–487, 1978.

[6] Felix: Independent Monitoring for Network Survivability. For more information see `ftp://ftp.bellcore.com/pub/mwg/felix/index.html`

[7] S. Floyd and V. Jacobson, "Random Early Detection Gateways for Congestion Avoidance," IEEE/ACM Transactions on Networking, 1(4), August 1993.

[8] IPMA: Internet Performance Measurement and Analysis. For more information see `http://www.merit.edu/ipma`

[9] IP Performance Metrics Working Group. For more information see `http://www.ietf.org/html.charters/ippm-charter.html`

[10] V. Jacobson, "Congestion Avoidance and Control", Proceedings of ACM SIGCOMM '88, August 1988, pp. 314–329.

[11] V. Jacobson, Pathchar - A Tool to Infer Characteristics of Internet paths. For more information see `ftp://ftp.ee.lbl.gov/pathchar`

[12] E.L. Lehmann. "Theory of point estimation". Wiley-Interscience, 1983.

[13] B.N. Levine, S. Paul, J.J. Garcia-Luna-Aceves, "Organizing multicast receivers deterministically according to packet-loss correlation", Preprint, University of California, Santa Cruz.

[14] J. Mahdavi, V. Paxson, A. Adams, M. Mathis, "Creating a Scalable Architecture for Internet Measurement," *to appear in Proc. INET '98*.

[15] M. Mathis and J. Mahdavi, "Diagnosing Internet Congestion with a Transport Layer Performance Tool," *Proc. INET '96*, Montreal, June 1996.

[16] S.P. Meyn and R.L. Tweedie, "Markov chains and stochastic stability", Springer, New York, 1993.

[17] `mtrace` – Print multicast path from a source to a receiver. For more information see `ftp://ftp.parc.xerox.com/pub/net-research/ipmulti`

[18] `nam` – Network Animator. For more information see `http://www-mash.cs.berkeley.edu/ns/nam.html`

[19] `ns` – Network Simulator. For more information see `http://www-mash.cs.berkeley.edu/ns/ns.html`

[20] V. Paxson, "End-to-End Routing Behavior in the Internet," *Proc. SIGCOMM '96*, Stanford, Aug. 1996.

[21] V. Paxson, "Towards a Framework for Defining Internet Performance Metrics," *Proc. INET '96*, Montreal, 1996.

[22] V. Paxson, "End-to-End Internet Packet Dynamics," *Proc. SIGCOMM 1997*, Cannes, France, 139–152, September 1997.

[23] V. Paxson, "Automated Packet Trace Analysis of TCP Implementations," *Proc. SIGCOMM 1997*, Cannes, France, 167–179, September 1997.

[24] V. Paxson, "Measurements and Analysis of End-to-End Internet Dynamics," Ph.D. Dissertation, University of California, Berkeley, April 1997.

[25] J. Postel, "Transmission Control Protocol," RFC 793, September 1981.

[26] S. Ratnasamy & S. McCanne, "Inference of Multicast Routing Tree Topologies and Bottleneck Bandwidths using End-to-end Measurements", Proceedings IEEE Infocom'99, New York, (1999).

[27] K. Ross & C. Wright, "Discrete Mathematics", Prentice Hall, Englewood Cliffs, NJ, 1985.

[28] W. Rudin, "Functional Analysis", McGraw-Hill, New York, 1973.

[29] L. Sachs, "Applied Statistics", Springer, New York, 1982.

[30] M.J. Schervish, "Theory of Statistics", Springer, New York, 1995.

[31] Surveyor. For more information see `http://io.advanced.org/surveyor/`

[32] K. Thompson, G.J. Miller and R. Wilder, "Wide-Area Internet Traffic Patterns and Characteristics," *IEEE Network*, 11(6), November/December 1997.

[33] R.R. Wolff "Poisson Arrivals See Time Averages", Operations Research, 30: 223–231, 1982

[34] M. Yajnik, J. Kurose, D. Towsley, "Packet Loss Correlation in the MBone Multicast Network," *Proc. IEEE Global Internet*, Nov. 1996