Network Delay Tomography from End-to-end Unicast Measurements*

N.G. Duffield[1], J. Horowitz[2], F. Lo Presti[1,3], and D. Towsley[3]

[1] AT&T Labs–Research, 180 Park Avenue, Florham Park, NJ 07932, USA
{duffield,lopresti}@research.att.com
[2] Dept. Math. & Statistics, University of Massachusetts, Amherst, MA 01003, USA
joeh@math.umass.edu
[3] Dept. of Computer Science University of Massachusetts, Amherst, MA 01003, USA
towsley@cs.umass.edu

Abstract. In this paper, we explore the use of end-to-end unicast traffic measurements to estimate the delay characteristics of internal network links. Experiments consist of back-to-back packets sent from a sender to pairs of receivers. Building on recent work [11, 5, 4], we develop efficient techniques for estimating the link delay distribution. Moreover, we also provide a method to directly estimate the link delay variance, which can be extended to the estimation of higher order cumulants. Accuracy of the proposed techniques depends on strong correlation between the delay seen by the two packets along the shared path. We verify the degree of correlation in packet pairs through network measurements. We also use simulation to explore the performance of the estimator in practice and observe good accuracy of the inference techniques.

1 Introduction

Background and Motivation. As the Internet grows in size and complexity, it becomes increasingly important for users and providers to characterize and measure its performance and to detect and isolate problems. Yet, because of the sheer size of the network and the limit imposed by administrative diversity, it is not generally possible to directly access and measure but a small portion of the network. Consequently, there is a growing need for practical and efficient procedures that can take an internal snapshot of a significant portion of the network.

A promising approach to network measurements, the so called Network Tomography approach, addresses these problems by exploiting the end-to-end traffic behavior to reconstruct the network internal performance. The idea is that correlation in performance seen on intersecting end-to-end paths can be used to draw inferences about the performance characteristics of their common portion, without cooperation from the network. Multicast traffic is in particular well suited for this since a given packet only occurs once per link in the multicast

* This work was supported in part by DARPA and the AFL under agreement F30602-98-2-0238

distribution tree. Thus multicast traffic introduces a well structured correlation in the end-to-end behavior observed by the receiver that share the same multicast session. This correlation allows to infer the performance characteristics as packet loss rates, [1], packet delay distributions, [11], and packet delay variance, [6].

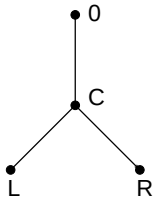

Fig. 1. 2-Leaf Tree.

To illustrate the idea behind multicast based delay inference, consider the simple tree in Figure 1 with the source (the root node) sending multicast packets to the two leaf nodes L and R and assume we collect the end-to-end measurements at the two receivers. If we consider the events where the delay seen by L is zero (assume for simplicity that the transmission and propagation delay are zero), the corresponding additional delays seen at R can be attributed to the link from C to R alone. We can thus form an estimate of the delay distribution for the link from C to R. The delay distribution of the other links can be derived by similar arguments.

Despite the encouraging results, multicast measurements suffer from two serious limitations. First, large portions of the Internet do not support network-level multicast. Second, the internal performance observed by multicast packets often differs significantly from that observed by unicast packets. This is especially serious given that unicast traffic constitutes the largest portion of the traffic on the Internet.

To overcome the limitation of multicast measurements, methods to extend the inference techniques to unicast measurements have been recently proposed in [3, 7] for the inference of loss rates and [4, 5] for delay distributions. The key idea is to design unicast measurement whose correlation properties closely resemble those of multicast traffic, so that it is possible to use the inference techniques developed for multicast inference; the closer the correlation properties are to that of multicast traffic, the more accurate the results. The basic approach, which has been further refined in [7] for the estimation of the loss rates, is to dispatch two back-to-back packets (a packet pair) from a probe source to a pair of distinct receivers. The premise is that, when the duration of network congestion events exceeds the temporal width of the packets, packets experience very similar behavior when they traverse common portions of their paths. Difference in the packets behavior occurs because congestion events may not affect packets uniformly: packet loss could not be uniform if lossy periods last less than the

time between the arrival of the two packets; delays will differ because of the interleaving of background traffic. Still, if the packets experience very similar behavior, the error in using the multicast based estimator is very small.

As an example, consider again the tree in Figure 1 with the source now sending two packets, back-to-back, the first to L and the second to R. In correspondence of the events where the delay seen by L is zero we will still attribute the additional delays seen at R to the link from C to R. But, because the two packets will possibly experience slightly different delays along the link from 0 to C, our estimate of the delay distribution for the link from C to R will contain an error roughly equal to the difference in delay seen by the two packets along common link. The smaller this difference, the more accurate the estimates.

We observe that a more accurate approach would consist in taking into account the difference experienced by the two packets along the shared link and incorporating it in our model. Unfortunately, we found out that it is not possible to estimate its value, at least not without additional assumptions. Therefore, here we rely on small deviations from the ideal behavior and proceed as the two packets experience the same delay along the shared path.

Contributions. In this paper we describe efficient techniques for the estimation of link delay characteristics, namely, the per link delay distribution and per link delay cumulants, via end-to-end packet pairs measurements.

For the distribution analysis, our starting point is the work by Lo Presti, et al. [11] and subsequent work by Coates and Novak in [5, 4]. Following [11], we model link delay by non-parametric discrete distributions. The discrete distribution can be a regarded as binned or discretized version of the (possibly continuous) true delay distribution, where we explicitly trade-off the detail of the distribution with the cost of calculation. A potential limitation of this approach lies in the accuracy/complexity trade-off itself. Since the complexity of the analysis is function of the numbers of bins, it results that under the usual discrete model, whereby delay is discretized using a fixed bin size q, a small q to ensure a desired level of accuracy in the estimates results in too many parameters (bins) and excessive computational costs.

To overcome these limitations, here, we describe a novel approach to delay modeling. The idea is to discretize delay using variable sized bins. Smaller bins are used only in correspondence of concentrations of probability mass to ensure adequate resolution while larger bins are used otherwise. Intuitively, this allows us to reduce the number parameters (bins), and hence complexity, significantly, without losing accuracy. A complication with this approach is that a discrete model with variable bin size does not lend itself to analysis. To this end, we propose an approach to variable bin size modeling which, while restricting the possible choices of bin size to a specific format, lends itself to analysis. In particular, we can formulate the estimation problem for the proposed variable bin size model, by generalizing the Maximum Likelihood formulation of [5]. Estimation is carried out by adapting the Expectation-Maximization (EM) algorithm used in [5] to compute the MLE estimates.

Then, we also describe an efficient method to directly infer the per link delay variance. By a simple argument, we show that it is possible to express the link delay variance in terms of the covariance of the end-to-end delays. Therefore, we can estimate the variance directly from the sample covariance of the end-to-end delays. The same method can be extended for the estimation of higher order cumulants. Distribution and cumulants are closely related: knowledge of (all) the cumulants of a random variable is equivalent to know its distribution.

The rest of the paper is organized as follows. In Section 2 we specify the tree and delay model. In Section 3 we describe the estimators of the delay distribution. In Section 4 we describe the link delay variance estimator (for lack of space, we omit the extension to higher order cumulants). In Section 5 we use the National Internet Measurement Infrastructure (NIMI) [13] to gather end-to-end data from a diverse set of Internet paths, and verify the conditions for the accuracy of our methods. In Section 6 we use network level simulation to evaluate the accuracy of the estimators. We conclude in Section 7.

Related Work. There exist several tools and methodologies for characterizing link-level behavior from end-to-end unicast measurements. One of the first methodologies focuses on identifying the bottleneck bandwidth on a unicast route. The key idea is that, in an uncongested network, two packets sent back-to-back will arrive at the receiver with a spacing that is inversely proportional to the lowest link bandwidth on the path. This was noted by Jacobson [9], and analyzed by Keshav [10].

Use of end-to-end measurements of packet pairs in a tree connecting a single sender to several receivers for estimation of the link delay has been first considered in [5]. The inference of the link delay distribution is formulated as a maximum likelihood estimation problem which is solved using the Expectation Maximization (EM) algorithm. In [5, 4] the authors extend this approach to the nonstationary case and in [14] investigate unicast based inference in context of passive monitoring, whereby inference is based on observation of ongoing unicast sessions. Preliminary results on these methods reported in these papers show promise.

Our approach extend the results in [5] in that we consider a more general form of discrete model which allows us to significantly improve the accuracy/complexity trade-off. We remark that the variable bin size scheme presented in this paper can be used in other setting, *e.g.*, multicast based inference techniques.

2 The Tree and Delay Models

Tree Model. We represent the underlying physical network as a graph $G_{\text{phys}} = (V_{\text{phys}}, L_{\text{phys}})$ comprising the physical nodes V_{phys} (e.g. routers and switches) and the links L_{phys} between them. We consider a single source of probes $0 \in V_{\text{phys}}$ and a set of receivers $R \subset V_{\text{phys}}$. We assume that the set of paths from 0 to each $r \in R$ is stationary and forms a tree $\mathcal{T}_{\text{phys}}$ in $(V_{\text{phys}}, L_{\text{phys}})$; thus two such paths

never intersect again once they have diverged. We form the logical source tree $\mathcal{T} = (V, L)$ whose vertices V comprise 0, R and the branch points of $\mathcal{T}_{\text{phys}}$. The link set L contains the link (j, k) if one or more of the probe paths in $\mathcal{T}_{\text{phys}}$ pass through j and then k without encountering another element of V in between. We will sometimes refer to link $(j, k) \in L$ simply as link k. For $k \neq 0$, $f(k)$ denotes the parent of k. We write $j \succ k$ if j is an ancestor of k in $\mathcal{T}$. $i \vee j$ denotes the minimal common ancestor of i and j in the $\preceq$-ordering.

Packet Pair and Delay Model. Let $\langle i, j \rangle$ denote a packet pair dispatched to destination nodes i, j in that order. The paths traverse a common set of links down to node $i \vee j$. Let $p(i, j)$ denote the set of nodes traversed by at least one member of the packet pair. For $k \in p(i, j)$ let $G(k) \subseteq \{1, 2\}$, where 1 and 2 denote the two packets sent in order to i and j, denote the set of packets that transit k. We describe the progress of the packet pair in $\mathcal{T}$ by the variable $X_k(l)$, $l \in G(k)$, which represents the accrued queueing delay of packet d along the route to k. We assume that we only observe the end-to-end delay $X_{ij} = (X_i(1), X_j(2))$ at receivers i and j.

We specify a delay model for the packet pair. We associate with each node k a pair of random variables D_k and D'_k that take values in the extended positive real line $R_+ \cup \{\infty\}$. By convention $D_0 = D'_0 = 0$. D_k (D'_k) is the delay that would be encountered by the first (second) packet attempting to traverse the link $(f(k), k) \in L$. A delay equal to ∞ indicates that the packet is lost on the link. We assume that delays are independent between different pairs, and for packets of the same pair on different links. The delay experienced by packet 1 on the path from root 0 to node k is $X_k(1) = \sum_{l \succeq k} D_l$. The delay experienced by packet 2 is $X_k(2) = \sum_{l \succeq (i \vee j) \vee k} D'_l + \sum_{(i \vee j) \vee k \succ l \succeq k} D_l$. Note that $X_k(\cdot) = \infty$ iff any delay along the path to k is infinite, i.e. if the packet is lost on some link between nodes 0 and k.

For any $k \in V$, $E_k = D'_k - D_k$ is the difference between the delays experienced by the back-to-back packets of a packet pair traversing k. Ideally, $E_k = 0$, and the packet pair behaves like a notional multicast packet sent to the two receivers. In practice, we expect the two delays to be different. This is because congestion events at intervening nodes may not affect packets uniformly if they are not back-to-back. This occurs, for example, because of the packets being spaced apart as a result of traversing a bottleneck (low available bandwidth) link, and the interleaving of background traffic in between. Observe that $E_k \neq 0$ even in the case of perfectly back-to-back packets, *e.g.*, packet 2 suffers on additional delay due to the time required to transmit packet 1.

Measurement A measurement experiment consists of sending, for each pair of distinct receivers $i,j \in R$, n packet pairs $\langle i, j \rangle$. As a result of the experiment we collect a set of measurements $X^{i,j} = (X^{i,j(m)})_{m=1,\ldots,n}$, where $X^{i,j(m)} = (X_i(1)^{(m)}, X_j(2)^{(m)})$ and $(X_i(1)^{(m)}, X_j(2)^{(m)})$ is the end-to-end delay of the m-th packet pair $\langle i, j \rangle$. Let $X = (X^{i,j})_{i \neq j \in R}$ denote the complete set of measurements.

3 Non-parametric Estimation of Delay Distribution

In this section we describe techniques for the estimation of the probability distribution of the per link variable delay D_k. We quantize the delay to a finite set of values $\mathcal{Q}$. We assume that once quantized, $D_k = D'_k$. In other words, we assume E_k small enough that it can be ignored in the discrete model. We consider two cases. First, in Section 3.1 we consider the most usual form of discretization, where we discretize delay to a set $\mathcal{Q} = \{0, q, 2q, \ldots, Bq, \infty\}$, where q is a suitable fixed bin size. Then, in Section 3.2 we consider a different approach whereby delay is discretized to a more general set $\mathcal{Q}$.

For the analysis, we thus model the link delay by a nonparametric discrete distribution that we can regard as a discretized version of the (possibly continuous) actual delay distribution. We denote the distribution of D_k by $\alpha_k = (\alpha_k(d))_{d\in\mathcal{Q}}$, where $\alpha_k(d) = \mathsf{P}[D_k = d]$, $d \in \mathcal{Q}$. We will denote by $\alpha = (\alpha_k)_{k\in V}$ the set of links distributions.

3.1 Delay Analysis with a Fixed Bin Size Discrete Model

Here, we consider the usual discrete model wherein D_k takes a value in $\mathcal{Q} = \{0, q, 2q, \ldots, Bq, \infty\}$, where q is a suitable fixed bin size. The point ∞ is interpreted as "packet lost" or "encountered delay greater than Bq". We define the bin associated to $iq \in \mathcal{Q}$ to be the interval $[iq - \frac{q}{2}, iq + \frac{q}{2})$, $i = 1, \ldots, B$, and $[Bq - \frac{q}{2}, \infty)$ the one associated to the value ∞. Because delay is non negative, we associate with 0 the bin $[0, \frac{q}{2})$. We denote this model as the (q, B) model.

Our goal is to estimate α using maximum likelihood based on the overall observed data X (also discretized to the set $\mathcal{Q}$). Denote by $\Omega = \mathcal{Q} \times \mathcal{Q}$ the set of possible outcomes for the packet pairs delays For each outcome $x_{i,j} \in \Omega$ denote $n(x_{i,j})$ the number of pairs $\langle i, j\rangle$, $m = 1, \ldots, n$, for which $X^{i,j(m)} = x_{i,j}$. Let $p_\alpha(x_{i,j}) = \mathsf{P}_\alpha[X_{i,j} = x_{i,j}]$ denote the probability of the outcome $x_{i,j}$. $p_\alpha(x^{i,j})$ can be expressed in terms of convolutions of the distribution α_k, $k \in p(i, j)$.

The log-likelihood of the measurement X is

$$\mathcal{L}(X; \alpha) = \log \mathsf{P}_\alpha[X] = \sum_{i\neq j\in R} \sum_{x_{i,j}\in\Omega} n(x_{i,j}) \log p_\alpha(x_{i,j}) \tag{1}$$

We estimate α by the maximizer of the likelihood (1), namely, $\widehat{\alpha} = \arg\max_\alpha \mathcal{L}(\alpha)$. Unfortunately, given the form of (1) we have been unable to obtain a direct expression for $\widehat{\alpha}$. Instead, we follow the approach in [4, 5], and employ the Expectation Maximization (EM) algorithm to obtain an iterative approximation $\widehat{\alpha}^{(\ell)}$, $\ell = 0, 1, \ldots$, to (local) maximizer of the likelihood (1). The basic idea behind the EM algorithm is that, rather then performing a complicated maximization, we "augment" the observed data with *unobserved* or *latent* data so that the resulting likelihood has a simpler form. Following [5], we augment the observations X with the *unobserved* actual delay experienced by the packet pairs along each link, namely, $D = (D_k^{i,j})_{k\in p(i,j), i\neq j\in R}$, where $D_k^{i,j} = (D_k^{i,j(m)})_{m=1,\ldots,n}$ are the delays experienced by the n packet pairs $\langle i, j\rangle$ along link k. The pair (X, D)

represents the *complete data* for our inference problem. The log-likelihood of the *complete data* (X, D) is

$$\mathcal{L}(X, D; \alpha) = \log \mathsf{P}_\alpha[X, D] = \log \mathsf{P}_\alpha[X|D] + \log \mathsf{P}_\alpha[D]. \tag{2}$$

The first term is 0 (since D uniquely determines X, we have that $\mathsf{P}_\alpha[X|D] = 1$). Expansion of the second term yields

$$\log \mathsf{P}_\alpha[D] = \sum_{i \neq j \in R} \sum_{k \in p(i,j)} \log P_\alpha[D_k^{i,j}] = \sum_{k \in V} \sum_{d \in \mathcal{Q}} n_k(d) \log \alpha_k(d) \tag{3}$$

where $n_k(d)$ is the total number of packets pairs that experienced a delay equal to d along link k. Should D be observable, the counts $n_k(d)$ would be known, and maximization of (3) would directly yield the MLE estimate of $\alpha_k(d)$,

$$\widehat{\alpha}_k(d) = \frac{n_k(d)}{\sum_{d \in \mathcal{Q}} n_k(d)} \tag{4}$$

Since D and $n_k(d)$ are not known, the EM algorithm uses the complete data log-likelihood $\mathcal{L}(X, D; \alpha)$ to iteratively find $\widehat{\alpha}$ as follows:

1. *Initialization.* Select the initial link delay distribution $\widehat{\alpha}^{(0)}$. As shown in Appendix, we select $\widehat{\alpha}^{(0)}$ as an estimate of α we compute by adapting the approach in [11].
2. *Expectation.* Given the current estimate $\widehat{\alpha}^{(\ell)}$, compute the conditional expectation of the log-likelihood given the observed data X under the probability law induced by $\widehat{\alpha}^{(\ell)}$, $Q(\alpha'; \widehat{\alpha}^{(\ell)}) = E_{\widehat{\alpha}^{(\ell)}}[\mathcal{L}(X, D; \alpha')|X] = \sum_{k \in V} \sum_{d \in \mathcal{Q}} \widehat{n}_k(d) \log \alpha'_k(d)$ where $\widehat{n}_k(d) = \mathsf{E}_{\widehat{\alpha}^{(\ell)}}[n_k(d)|X]$. $Q(\alpha'; \widehat{\alpha}^{(\ell)})$ has the same expression as $\mathcal{L}(X, D; \alpha')$ but with the actual *unobserved* counts $n_k(d)$ replaced by their conditional expectations $\widehat{n}_k(d)$. To compute $\widehat{n}_k(d)$, observe that we can write the counts $n_k(d)$ as $n_k(d) = \sum_{i \neq j \in R: k \in p(i,j)} \sum_{m=1}^{n} \mathbf{1}_{\{D_k^{i,j(m)} = d\}}$. Then

$$\widehat{n}_k(d) = \sum_{i \neq j \in R: k \in p(i,j)} \sum_{m=1}^{n} \mathsf{P}_{\widehat{\alpha}^{(\ell)}}[D_k^{i,j(m)} = d | X^{i,j(m)}] \tag{5}$$

$$= \sum_{i \neq j \in R: k \in p(i,j)} \sum_{x_{ij} \in \Omega} n(x_{ij}) \mathsf{P}_{\widehat{\alpha}^{(\ell)}}[D_k = d | X_{ij} = x_{ij}] \tag{6}$$

3. *Maximization.* Find the maximizer of the conditional expectation $\alpha^{(\ell+1)} = \arg\max_{\alpha'} Q(\alpha', \widehat{\alpha}^{(\ell)})$. The maximizer is given by (4) with the conditional expectation $\widehat{n}_k(d)$ in place of $n_k(d)$.
4. *Iteration.* Iterate steps 2 and 3 until some termination criterion is satisfied. Set $\widehat{\alpha} = \widehat{\alpha}^{(\ell)}$, where ℓ is the terminal number of iterations.

Convergence. Because the complete data likelihood can be shown to derive from a standard exponential family, the EM iterates $\widehat{\alpha}^{(\ell)}$ converge to a stationary point

of the likelihood α^*, *i.e.*, $\frac{\partial \mathcal{L}(X,D;\alpha)}{\partial \alpha}(\alpha^*) = 0$, (see *e.g.* [15]) . This implies that when there are multiple stationary points, *e.g.* local maxima, the EM iterates may not converge to the global maximizer. Unfortunately, we were not able to establish whether there is a unique stationary point or conditions under which unicity holds. Therefore, in general the estimates $\widehat{\alpha}^{(\ell)}$ converge to a local (but not necessary global) maximizer. Since the point of convergence depends on the initial estimate, we must carefully choose the initial estimate $\widehat{\alpha}^{(0)}$. Here we select as initial distribution the estimate of α obtained by using the approach in [11] (see the Appendix). We expect that, for large enough n, $\widehat{\alpha}^{(0)}$ (which converges to α), is close enough to the actual likelihood maximizer so to ensure, in most cases, the desired convergence.

Complexity. The complexity of the algorithm is dominated by the computation of the conditional expectation $\widehat{n}_k(d)$ which can be accomplished in time that is $O(npB^2)$, where p is the average number of links between the source and a leaf node, using the upward-downward probability propagation algorithm [8].

Choice of bin size. Since packet delay is essentially continuous in nature, the use of a discrete model introduces a quantization error, which is a function of the bin size q. The choice of q is thus primarily dictated by the trade-off between accuracy and computational complexity: a smaller q provides better accuracy but at rapidly increasing computational cost (observe that since the product qB is constant the complexity is basically $O(np/q^2)$); on the other hand, use of larger bin size, reduces the computational complexity but may not be adequate to accurately capture very small delays.

We must also consider that, in the context of unicast measurements, the delay resolution must be large enough so that, once discretized to $\mathcal{Q}$, we can use the approximation $D_k \approx D'_k$. Our network experiments in Section 5 suggest that q should not be smaller than $1msec$ to satisfy this condition.

3.2 Delay Analysis with Variable Bin Size Discrete Model

Here we consider a more general form of discrete model in which D_k takes values in a more general finite set $\mathcal{Q}$. This is motivated by the observation that the use of a fixed bin size may be too restrictive in the analysis of large networks where delay characteristics significantly vary from node to node: a value of q chosen to adequately capture the delay behavior of very fast links would result in too many parameters if slower or congested links are also present. Ideally, to overcome the limitations of the accuracy/complexity trade-off of the fixed bin size models, it is preferable to discretize delay to a suitable set $\mathcal{Q}$, which guarantees the desired resolution in the delay range of interest while keeping the overall number of bins sufficiently small. For example, smaller bins could be used only in correspondence of concentrations of probability mass to ensure adequate resolution while larger bins could be used otherwise. Intuitively, this would allow us to reduce the number parameters (bins), and hence complexity, significantly, without losing accuracy.

A complication with this approach is that a discrete model with a general set of values $\mathcal{Q}$ does not lend itself to analysis. The problem, is that a general discrete set $\mathcal{Q}$ is not closed under the sum operation. Therefore, we cannot express the observable delay (discretized to $\mathcal{Q}$) in terms of sum of link delays (also discretized to $\mathcal{Q}$).

To overcome these difficulties, we now describe a simple approach to variable bin size modeling which, while restricting the possible choices of $\mathcal{Q}$ to a specific format, lends itself to analysis. The key idea is to consider variable bin size models, the analysis of which can be reduced to that of a set of fixed bin size models. We proceed as follows: (1) we define a variable bin size model as the *composition* (in the sense described below) of fixed bin size models; and (2), we choose the constituent fixed bin size models so that the estimates of the distribution for these models can be *composed* to form the estimate of the distribution of the variable bin size model itself. By appropriate choice of the fixed bin size models, the resulting variable bin size model has a better accuracy/complexity trade-off. We detail the approach below.

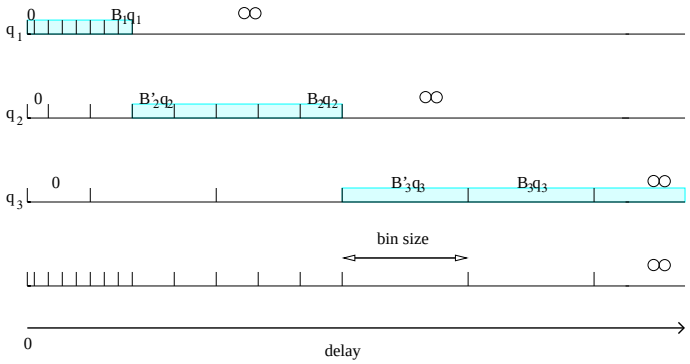

Fig. 2. Variable bin size model as composition of fixed bin size models.

We define the variable bin size discrete model as the *composition* of M uniform bin size discrete models $((q_l, B_l))_{l=1,\ldots,M}$, with increasing bin size, $0 < q_1 < \ldots < q_M$ and such that $B_1 q_1 < \ldots < B_M q_M$ (see Figure 2). We assume that for $l = 2, \ldots, M$, each bin of level l either corresponds to an integer number of level $l-1$ bins (*i.e.*, the boundaries of the bin of level l correspond to boundaries of a group of adjacent bins of level $l-1$) or is contained in the ∞ bin of level $l-1$. We let $g_l(j)$ denote the set of level $l-1$ bins which corresponds to the j-th level l bin, $j = 0, \ldots, B'_l < B_l$ where B'_l is the first level l bin contained in level $l-1$ last bin (the one corresponding to ∞).

In the variable bin size model, D_k takes values in $\mathcal{Q} = \{0, q_1, \ldots, B_1 q_1, B'_2 q_2, \ldots, B'_M q_M, \infty\}$. We define the bin associated to $iq_l \in \mathcal{Q}$ as the interval $[iq_l - \frac{q_l}{2}, iq_l + \frac{q_l}{2})$, and $[B_M q_M - \frac{q_M}{2}, \infty)$ the one associated to ∞. With this definition, we create a correspondence between the bins of the variable bin size model and bins of the fixed bin size models (the shaded bins in Figure 2). This allows us to

express the distribution α in the variable bin size model in terms of the delay distribution in the M uniform bin size models. For $k \in V$, denote $\alpha_k(d; q_l) = P[D_k = d]$, $d \in Q_l = \{0, \ldots, B_l q_l, \infty\}$ the distribution for the model with fixed bin size q_l. The distribution of D_k in the variable bin size model is then $\alpha_k = (\alpha_k(d))_{d \in \mathcal{Q}}$, where $\alpha_k(d) = \alpha_k(iq_l; q_l)$, $d = iq_l \in \mathcal{Q}$, and $\alpha_k(\infty) = \alpha_k(\infty; q_M)$, $k \in V$. We will take advantage of this correspondence for the estimation.

With the above definition, we are limited to variable bin size models where the bin size progressively increases. However, we do not believe this choice to be restrictive. Indeed, we expect that in most cases it is desirable to have smaller bins in correspondence with small delay values and larger bins otherwise; while the small bins guarantee enough resolution for very fast or uncongested links, the larger bins prevent the explosion of the number of parameters due to the large delays experienced by the slower and congested links.

Example. We consider the *ternary* variable bin size model defined, for a given base bin size q and number of levels M, as $((3^{(l-1)}q, 2]))_{l=1,\ldots,M}$. Delay is thus discretized to the set $\{0, q, 3q, 9q, \ldots, 3^{M-1}q, \infty\}$(see Figure 3). This can be considered as an extreme case where each level has only three bins, 0, $3^{(l-1)}q$ and ∞, and the bin size grows exponentially with the level. Observe that this model covers the delay range from 0 up to a maximum value d_{max} with only $O(\log_3 \frac{d_{max}}{q})$ bins.

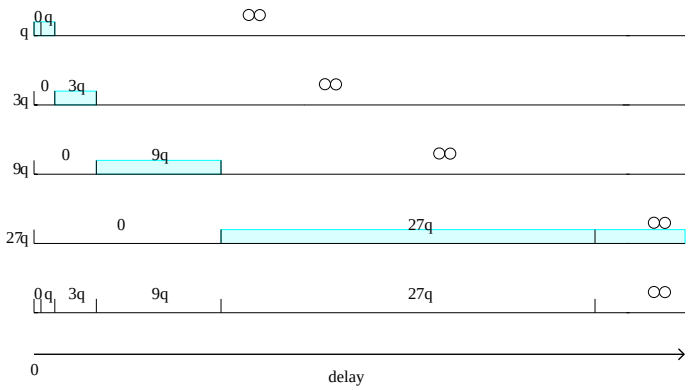

Fig. 3. Example: The Ternary Variable Bin Size Model ($M = 4$).

We estimate the distribution α of the variable bin size model indirectly by taking advantage of the relationship between the bins of the variable bin size model and those of the component fixed bin size models. Basically, we estimate the probabilities of the former by the corresponding estimates of the latter. More precisely, estimation of α proceeds by computing recursively the MLE estimate of M discrete models (q_l, B_l), starting with $l = 1$ as follows:

1. Discretize the delays to the set Q_l.

2. Estimate the probabilities $\alpha_k(d; q_l)$, $d \in Q_l$, $k \in V$. For $l = 1$, we use the EM algorithm directly. For $l > 1$, to have consistency between the estimates of the different models, we compute the estimates of the probabilities of level l bins corresponding to a group of level $l-1$ bins, directly as the sum of the probabilities of those bins. In other words, we let $\widehat{\alpha}_k(d; q_l) = \sum_{j \in g_l(d/q_{l-1})} \widehat{\alpha}_k(jq_{l-1}; q_{l-1})$ for $d \leq q_l B'_l$. We then use the EM algorithm to estimate the remaining probabilities $\alpha_k(d; q_l)$ for $d \geq q_l B'_l$ assuming the probabilities $\alpha_k(d; q_l)$, $d \leq q_l B'_l$, as known parameters (set equal to the estimates above). This is equivalent to the EM algorithm shown in Section 3.1, where we replace (4) with

$$\widehat{\alpha}_k^{(\ell)}(d; q_l) = \left(1 - \sum_{d' \leq B'_l q_l} \widehat{\alpha}_k(d'; q_l)\right) \frac{\widehat{n}_k(d)}{\sum_{d' > B'_l q_l} \widehat{n}_k(d')} \tag{7}$$

3. Iterate 1 and 2 for $l = 1, \ldots, M$.
4. Compose the estimates of the M models to estimate α, *i.e.*, set $\widehat{\alpha}_k(d) = \widehat{\alpha}_k(iq_l; q_l)$, $d = iq_l \in \mathcal{Q}$, $k \in V$.

Complexity. The computational cost equals the sum of the costs of computing the MLE estimates of each model. Assuming for simplicity that the number of iterations required by the EM algorithm does not vary, the complexity is then $O(np \sum_{l=1}^{M} B_i^2)$.

Choice of the Variable Bin Size Model. The use of the variable bin size model provides great flexibility in terms of both accuracy and computational cost. We consider two examples below. To ensure high accuracy a simple solution lies in using a variable bin size model with only two levels, *i.e.*, $M = 2$: the first level has a small bin size, chosen according the desired level of accuracy and enough bins to include most of the probability mass, *e.g.*, B_1 large enough that $P[D_k \leq B_1 q_1] > 0.999$; the second level has a larger bin size and covers the rest of the delay interval. We expect that capturing the tail of the distribution with a larger bin size can provide a significant reduction in the computational cost without accuracy degradation. At the other extreme, we might consider the solution which has the smallest complexity. Since the complexity is proportional to $\sum_{l=1}^{M} B_i^2$, we simply have to minimize the number of bins per level and use as many levels as necessary. We thus obtain the ternary variable bin size model. In between these extreme cases, it is possible to consider several models which provide the desired accuracy complexity trade-off. In general we expect the model to be determined either *a priori* or based on the measurements themselves.

3.3 Comparison of the Variable and Fixed Bin Size Model.

We illustrate the potential benefit of the variable bin size model using model-based simulations in which link delays are independent, exponentially distributed random variables. We assume no packet loss. We conducted 1000 independent

experiments over the 2-leaf tree in Figure 1. In each experiment, we sent 1000 packet-pairs down the tree. We assumed that the back-to-back packets have the same delay along the common link. Average link delays were chosen independently with a uniform distribution in the interval $[0.1, 10] msec$.

For the analysis, we consider three different discrete models: the first two models are the two bin size models $(1\text{msec}, 100)$ and $(10\text{msec}, 10)$; the third model is the ternary model $(3^{l-1}1\text{msec}, 2)_{l=1,\ldots,5}$. The number of bins in each model was chosen so that the largest finite delay in each model was about $100msec$. We used the EM algorithm for the estimation. Initialization was performed as described in the Appendix. The termination criterion for the EM algorithm was that successive iterates of any probability should have an absolute distance less of 10^{-3}.

Complexity. The computational costs differ substantially. To compare the costs, observe that each iteration has a complexity proportional to the square of the number of bins, which for the three models is 10,000, 100 and 9. The average number of iterations for the different models, was respectively, 22, 13 and 31 (the last number is the sum over the 5 fixed bin size models). Thus, the fixed bin size model with bin size $1msec$ requires about two orders of magnitude more operations than what is needed by the variable bin size model, which, despite the need of executing the EM algorithm multiple times, has the smallest computational complexity.

	fixed bin size		variable bin size
	$q = 1msec$	$q = 10msec$	
all links	2.6%	8.1%	17.6%
links with average delay $< 1msec$	9.7%	64.4%	14.6%

Table 1. Median of the Absolute Relative Error of the Average Delay Estimates.

Accuracy. We now compare the accuracy of the different approaches. In order to quantify the accuracy, in Table 1 we list the median of the absolute relative error of the average delay estimates. As expected the best performance is achieved by the fixed bin size model with $q = 1msec$; use of a larger bin, while greatly reducing the complexity, resulted in very poor accuracy for the smaller delays (if we consider only links with average delay smaller than 1msec, the typical error was 64.4%). By contrast, the variable bin size model achieves good accuracy across the entire delay range, while at the same time enjoying a low computational cost.

4 Non-parametric Estimation of Link Delay Variance

In this section we present a class of non-parametric estimators of the link delay variance. We assume initially that all delays are finite: $\mathsf{P}[D_k = \infty] = 0$. We will later relax this assumption.

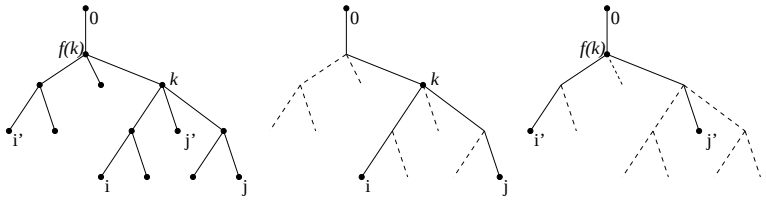

Fig. 4. Logical multicast Tree (left) and the two subtrees traversed by the pairs $\langle i, j\rangle$ (center) and $\langle i', j'\rangle$ (right).

For a node $k \in V$, consider the packet pairs $\langle i, j\rangle$ and $\langle i', j'\rangle$, dispatched to the nodes i and j and i' and j', respectively, such that $i \vee j = k$ and $i' \vee j' = f(k)$; see Figure 4. From the assumption that delays along different links are independent and the bilinearity of the covariance, for the packet pair $\langle i, j\rangle$ it follows that

$$\begin{aligned}
\mathsf{Cov}[X_i(1), X_j(2)] &= \mathsf{Cov}[X_k(1) + (X_i(1) - X_k(1)), X_k(2) + (X_j(2) - X_k(2))] && (8)\\
&= \mathsf{Cov}[X_k(1), X_k(2)] && (9)\\
&= \mathsf{Var}[X_k(1)] + \mathsf{Cov}[X_k(1), X_k(2) - X_k(1)] && (10)\\
&= \mathsf{Var}[X_k(1)] + \sum_{l \succeq k} \mathsf{Cov}[D_l, E_l]. && (11)
\end{aligned}$$

Similarly, for the the packet pair $\langle i', j'\rangle$ we have that $\mathsf{Cov}[X_{i'}(1), X_{j'}(2)] = \mathsf{Var}[X_{f(k)}(1)] + \sum_{l \succeq f(k)} \mathsf{Cov}[D_l, E_l]$. Observe that $X_k(1) = X_{f(k)}(1) + D_k$, and $X_{f(k)}(1) = \sum_{l \succeq f(k)} D_l$ and D_k are independent. Therefore, $\mathsf{Var}[D_k] = \mathsf{Var}[X_k] - \mathsf{Var}[X_{f(k)}]$ which we can rewrite

$$\begin{aligned}
\mathsf{Var}[D_k] &= \mathsf{Cov}[X_i(1), X_j(2)] - \mathsf{Cov}[X_{i'}(1), X_{j'}(2)] - \mathsf{Cov}[D_k, E_k] && (12)\\
&\approx \mathsf{Cov}[X_i(1), X_j(2)] - \mathsf{Cov}[X_{i'}(1), X_{j'}(2)] && (13)
\end{aligned}$$

under the assumption that $|\mathsf{Cov}[D_k, E_k]| \ll \mathsf{Var}[D_k]$ (Observe that $\mathsf{Cov}[D_k, E_k] = 0$, in particular, if E_k and D_k are independent, or if E_k is constant). (13) expresses the variance of the packet delay along link k in terms of the covariance of delays measured at receivers. We can form an estimator of $\mathsf{Var}[D_k]$ (which is unbiased if $\mathsf{Cov}[D_k, E_k] = 0$) from the unbiased estimators of the end-to-end covariances. More precisely, abbreviate $\mathsf{Cov}[X_i(1), X_j(2)] = s_{ij}$ and $\mathsf{Var}[D_k] = v_k$. We can then estimate v_k by the difference $\widehat{s}_{ij} - \widehat{s}_{i'j'}$ of the unbiased estimators of s_{ij} and $s_{i'j'}$, namely

$$\widehat{s}_{ij} = \frac{1}{n-1}\left(\sum_{m=1}^{n} X_i^{i,j}(1)^{(m)} X_j^{i,j}(2)^{(m)} - \frac{1}{n}\sum_{m,m'=1}^{n} X_i^{i,j}(1)^{(m)} X_j^{i,j}(2)^{(m')}\right) \tag{14}$$

and similarly for $\widehat{s}_{i'j'}$.

More generally, let $Q(k) = \{\{i,j\} \subset R \mid i \vee j = k,\}$ be the set of distinct pairs of receivers whose $\prec$-least common ancestor is $k \in V$. Measurements of the packet pairs $\langle i,j\rangle$, $\{i,j\} \in Q(k)$ and $\langle i',j'\rangle$, $\{i',j'\} \in Q(f(k))$ yields estimates of v_k, namely, $\widehat{s}_{ij} - \widehat{s}_{i'j'}$ as does any convex combination $\sum_{\{i,j\}\in Q(k),\{i',j'\}\in Q(f(k))} \eta_{iji'j'}(\widehat{s}_{ij} - \widehat{s}_{i'j'})$ (where the $\eta_{iji'j'}$ are non negative and sum to 1), which we can rewrite as

$$V_k(\mu,\widehat{s}) := \sum_{\{i,j\}\in Q(k)} \mu_{ij}(k)\widehat{s}_{ij} - \sum_{\{i',j'\}\in Q(f(k))} \mu_{i'j'}(f(k))\widehat{s}_{i'j'} \qquad (15)$$

where $\widehat{s} = \{\widehat{s}_{ij} : \{i,j\} \in Q(k), k \in V\}$, $\mu(k) = (\mu_{ij}(k))_{\{i,j\}\in Q(k)}$, $\mu_{ij}(k) = \sum_{\{i',j'\}\in Q(f(k))} \eta_{iji'j'} \geq 0$, $\sum_{\{i,j\}\in Q(k)} \mu_{ij}(k) = 1$, and similarly for $\mu_{i'j'}(f(k))$. Finally, denote $\mu = (\mu(k), \mu(f(k)))$. An example is the uniform estimator where all $\mu(k)$ and $\mu(f(k))$ are constant. The uniform estimator has the disadvantage that a high variance of any summand may result in a high variance in the overall estimate. By proper choice of the weights we can determine the estimator $V_k(\mu,\widehat{s})$ of minimum variance.

The next theorem characterizes the asymptotic behavior of $V_k(\mu,\widehat{s})$ and gives a form for the estimator of minimum variance. The proofs follow the same lines of those for the multicast case in [6] and are omitted. Define $Z_i(l) = X_i(l) - \mathsf{E}[X_i(l)]$, $i \in R$, $l = 1,2$, and let $w_{ij} = \mathsf{Var}[Z_i(1)Z_j(2)]$, $i \neq j \in R$ and $m_k = \mathsf{Cov}[D_k, E_k]$.

Theorem 1. *For each $k \in V$:*

(i) the random variables $\sqrt{n}\cdot(\widehat{s}_{ij} - v_k + m_k)$, $\{i,j\} \in Q(k)$ are independent and converge in distribution to a Gaussian random variable with mean 0 and variance w_{ij};

(ii) for any choice of μ, $\sqrt{n}(V_k(\mu,\widehat{s}) - v_k + m_k)$ converges in distribution to a Gaussian random variable of mean zero and variance $\sum_{\{i,j\}\in Q(k)} \mu_{ij}^2(k)w_{ij} + \sum_{\{i',j'\}\in Q(f(k))} \mu_{i'j'}^2(f(k))w_{i'j'}$;

(iii) the minimal asymptotic variance of the estimator $V_k(\mu,\widehat{s})$ is achieved when $\mu_{ij}(h) = \mu_{ij}^*(h) := \frac{w_{ij}^{-1}}{\sum_{\{i',j'\}\in Q(h)} w_{i'j'}^{-1}}$, $h \in \{k, f(k)\}$. The corresponding asymptotic variance of the estimator is $\frac{1}{\sum_{\{i,j\}\in Q(k)} w_{ij}^{-1}} + \frac{1}{\sum_{\{i',j'\}\in Q(f(k))} w_{i'j'}^{-1}}$.

Theorem 1 shows that $V_k(\mu,\widehat{s})$ is asymptotically normal. We define the estimator bias as $b_k = |E[V_k(\mu,\widehat{s}) - v_k]|$, $k \in V$. For large n, we can use the approximation $b_k \approx |\mathsf{Cov}[D_k, E_k]|$. Thus, under the assumption that $|\mathsf{Cov}[D_k, E_k]| \ll \mathsf{Var}[E_k]$, we have $\mathsf{E}[V_k(\mu,\widehat{s})] \approx v_k$.

Operationally, the weights μ need to be calculated from an *estimate* $\widehat{w}_{i,j}$ of the variances w_{ij}. These can be computed as shown in [6]. The resulting estimator $V_k(\widehat{\mu}^*,\widehat{s})$, where μ^* is obtained by using $\widehat{w}_{ij}$ in place of w_{ij} , has the same asymptotic behavior of $V_k(\mu^*,\widehat{s})$.

Impact of Loss on the Estimators. We now relax the assumption of finite delays. We associate infinite delays to packet losses. Although lost packets will not

provide delay samples at receivers, clearly, the foregoing still applies to cumulant estimation based on the end-to-end delays of the received packets. For any packet pair $\langle i, j \rangle$, define $I_n(i,j) \subset \{1, \ldots, n\}$ the set of pairs for which both packets reach the leaf nodes; define $N_n(i,j) = \#I_n(i,j)$ the number of such pairs. Denote by $B(i,j) = \prod_{k \succeq i,j} \mathsf{P}[D_k < \infty]$ the probability that the two packets of the packet pair reach the leaf nodes. $N_n(i,j)/n$ converges almost surely to $B(i,j)$ as $n \to \infty$. For large n we have approximatively $N_n(i,j) \approx B(i,j)n$ delay measurements from both packets of the pair $\langle i, j \rangle$.

We adapt the approach of the foregoing theory by estimating s_{ij} using only the measurements from the pairs in $I_n(i,j)$. This corresponds to replacing n with $N_n(i,j)$ and $\sum_{m=1}^{n}$ with $\sum_{m \in I_n(i,j)}$ in (14). The effect of packet loss is to reduce the number of packet pairs available for estimation, thus increasing the variability of the estimates. The asymptotic behavior is characterized by results similar to Theorem 1 where we replace w_{ij} by $\frac{w_{ij}}{B(i,j)}$.

5 Network Experiments

The accuracy of the techniques described in Sections 3 and 4 rely on the assumptions that: (1) the back-to-back packets in the packet pair experience roughly the same delay on each link along their common path, *i.e.*, $D_k \approx D'_k$; (2) the additional delay experienced by the second packet is uncorrelated to the delay experienced by the first packet (or practically so), *i.e.*, $|\mathsf{Cov}[D_k, E_k]| \ll \mathsf{Var}[D_k]$. In this section we investigate conformance of both of measurements of packet pairs transmitted across a number of end-to-end paths in the Internet to both of these assumptions. Although these experiments did not access the transmission properties of individual links (which are very difficult to measure), they are able to detect link-wise departures from the assumptions, since these would also be reflected in the properties of end-to-end paths over non-conformant links.

Measurement Infrastructure. We conducted the experiments using the National Internet Measurement Infrastructure (NIMI) [13]. NIMI consists of a number of measurement platforms deployed across the Internet (primarily in the U.S.) that can be used to perform end-to-end measurements. We made the measurements using the `zing` utility, which sends UDP packets in selectable patterns, recording the time of transmission and reception. `zing` was extended to transmit packets pairs with minimal spacing between packets. The resulting inter-packet spacings were of about 40μsec.

Measurements were performed along end-to-end paths, by sending packet pairs from a sender to a receiver host. These measurements did not allow us to directly study the delay behavior of the pair along internal links, which would have required measurement inside the network.

Here we report the results from 13 successful measurements made between 11 NIMI sites (two of which are in Europe). Each measurement recorded at both sender and receiver the transmission of 6000 back-to-back packet pairs sent at exponentially distributed intervals with a mean of $100msec$. All measurements

were made at either 2PM EDT (a busy time) or 2AM EDT (a fairly unloaded time) separated by a mean of $100msec$. Since our focus is on the variable portion of the delay, in the results reported below we normalize each delay measurement by subtracting the minimum delay seen at the receiver. A delay equal to the minimum delay is thus regarded as a variable delay equal to 0. In other words, we interpret the observed minimum delay as the constant propagation and transmission delay along the path, under the assumption that at least one packet experienced no queueing delay along the path.

Delay Characteristics. In Table 2 we display the relevant delay statistics (measured in *msec*) along each path, ordered in increasing average delay.

$\mathsf{E}[D_k]$	$\sqrt{\mathsf{Var}[D_k]}$	$\mathsf{E}[E_k]$	$\sqrt{\mathsf{Var}[E_k]}$	$\frac{\mathsf{Cov}[D_k,E_k]}{\mathsf{Var}[D_k]}$
0.58	0.40	0.06	0.14	$-2.73 \cdot 10^{-2}$
0.62	10.22	0.08	0.22	$7.81 \cdot 10^{-4}$
0.89	2.63	1.03	0.36	$-3.16 \cdot 10^{-4}$
1.27	7.50	0.18	0.88	$2.23 \cdot 10^{-3}$
1.58	8.74	0.10	0.22	$1.34 \cdot 10^{-4}$
1.95	1.62	1.01	0.18	$-2.78 \cdot 10^{-3}$
2.64	3.61	0.08	0.04	$-1.81 \cdot 10^{-3}$
4.30	23.31	0.25	0.34	$-1.10 \cdot 10^{-4}$
5.44	42.70	0.29	0.90	$2.80 \cdot 10^{-5}$
5.62	60.31	0.32	0.45	$-5.66 \cdot 10^{-5}$
37.28	47.55	0.26	0.83	$2.79 \cdot 10^{-5}$
63.72	65.67	0.61	0.82	$5.98 \cdot 10^{-5}$
65.97	62.16	0.42	1.57	$5.19 \cdot 10^{-5}$

Table 2. Summary Delay Statistics (in *msec*).

The average delay ranged from $0.6msec$ to $66msec$, a span of two orders of magnitude. The entries in Table 2, with either a large average delay, a large standard deviation, or both, correspond to six experiments involving sites in Europe (the last rows in Table 2). Despite the delay diversity in our measurements, the difference in the average delay seen by back-to-back packets was somewhat more uniform, increasing with larger delay, but with an average and standard deviation typically below $1msec$. This suggests that in practice we can use the approximation $D_k \approx D'_k$ as long as we adopt a delay resolution larger than $1msec$, *i.e.*, we discretize delay with bin size larger than $1msec$. At these resolutions, indeed, the delays seen by the two packets can be considered identical.

Finally, we turn our attention to the bias of the variance estimator. In Table 2, we list the relative bias $\frac{\mathsf{Cov}[D_k,E_k]}{\mathsf{Var}[D_k]}$. The results show that variability of E_k is much smaller than that of D_k. The bias is only 3% in the worst case.

6 Simulation Results

The experiments of Section 5 show that the delay properties of back-to-back packets make packet pairs suitable for delay inference. In this Section, we employ simulation to evaluate how accurate the estimators might be in practice.

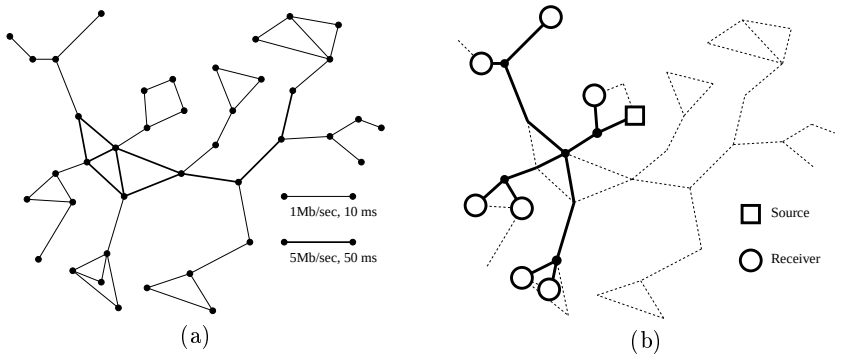

Fig. 5. Network Topology and Logical Source Tree used in the Simulations.

We used the `ns` simulation environment [12]; this enables the representation of transport-protocol details of packet transmissions. The simulations reported in this paper used the 39-node topology of Figure 5(a). The buffer on each link accommodated 20 packets. Background traffic came from 420 sessions comprised of a mixture of TCP sessions and exponential and Pareto on-off UDP sources.

We performed different sets of experiments. In each set we fixed a source and a set of receivers and conducted 100 experiments across the logical tree spanning those nodes. Measurement probes comprised packet pairs with a 1μsec interpacket time. The packet pairs were generated periodically with an interpacket time of 16 msec by cycling through pairs $\langle i, j \rangle$ sent to distinct receivers i, j. In each experiment, for each pair of distinct receivers $i, j \in R$, $n = 1000$ packets pairs $\langle i, j \rangle$ were transmitted.

In order to evaluate the inference methods, we compare inferred delay statistics, namely, mean and variance, against the actual link delay as determined by instrumentation of the simulation. Here we will report the results for the logical 7 receiver tree in Figure 5(b) which covers part of the network.

	$\mathsf{E}[D_k]$	$\sqrt{\mathsf{Var}[D_k]}$	$\mathsf{E}[E_k]$	$\sqrt{\mathsf{Var}[E_k]}$	$\left\lvert\frac{\mathsf{Cov}[D_k, E_k]}{\mathsf{Var}[D_k]}\right\rvert$
min.	0.3	1	0.04	0	$2.1 \cdot 10^{-6}$
median	17.4	17.6	0.27	1.1	$2 \cdot 10^{-3}$
max	55	38.7	0.47	2	$2.03 \cdot 10^{-2}$

Table 3. Simulations Summary Delay Statistics (in *msec*) .

Link Statistical Properties. We first examine the statistical properties of the underlying link processes. Characteristics vary considerably across the different links and in the different simulations. The average delay ranged from 0.3msec to 55msec, and the delay variance from 1msec^2 to 1,500msec^2. The link loss rates ranged from 0% to 18%. The link delay statistics are displayed in Table 3. The behavior and range is very similar to that observed in the network experiments. The important observation is that for 98% of the packet pairs the difference in

the delay seen by back-to-back packets was less than $1msec$. We can thus use the approximation $D_k \approx D'_k$ as long as the delay resolution is larger than this value. The bias due to ignoring the term $\mathsf{Cov}[D_k, E_k]$ in the estimation of the variance is negligible and only about 2% in the worst case.

Accuracy of Inference. We now compare inferred and actual link delay in the simulations. Here we focus on the estimation of the summary delay statistics, namely mean and variance. Given the large delay spread across the different links (delay was as large as a few hundreds $msec$), to infer the average delay we estimated the link delay distribution using the variable bin size model. We used the ternary variable bin size model $(3^{l-1}msec, 2)_{l=1,\ldots,5}$. For the analysis, delay was thus discretized to the set $\mathcal{Q} = \{0, 1, 3, 9, 27, 81, 243, \infty\}msec$, only eight bins. The estimate of the average delay is then $\mathsf{E}[D_k|\widehat{D_k} < \infty] = \frac{\sum_{d \in \mathcal{Q}\setminus\{\infty\}} d\widehat{\alpha_k}(d)}{\sum_{d \in \mathcal{Q}\setminus\{\infty\}} \widehat{\alpha_k}(d)}$, $k \in V$. As shown below, even if this model can be considered too coarse to adequately capture the probabilities of large delays, it allowed us to compute the estimates of the average delay efficiently and accurately. To estimate the link delay variance, we used directly the method described in Section 4.

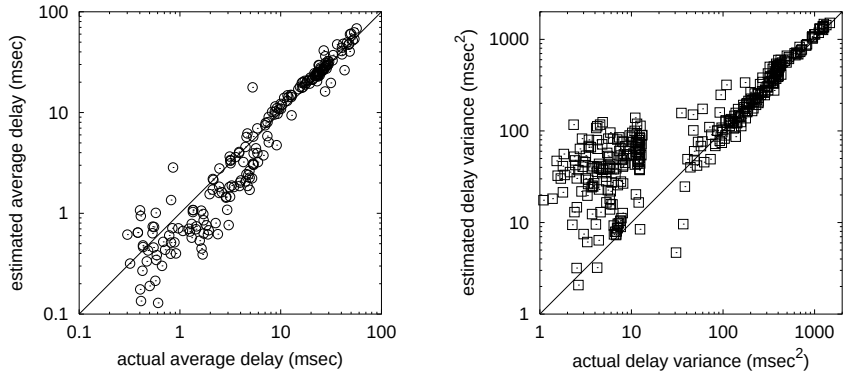

Fig. 6. INFERRED VS. ACTUAL AVERAGE AND VARIANCE OF LINK DELAY IN SIMULATIONS. Scatter plot for 100 experiments: (a) average link delay; (b) link delay variance.

In Figure 6, we display scatter plots of inferred vs. actual link delay mean and variance. Accuracy increases for higher values as exhibited by the clustering about the line $y = x$. In order to quantify the accuracy of the estimates, we computed the median of the absolute error of the estimates of the link delay mean and variance. The median was 22.05% for the mean and 40% for the delay variance. Estimates were more accurate for larger delays: if we consider delay means larger than $10msec$ or delay variances larger than $10msec^2$, the median of the relative absolute error fell to 10% and 11.75%, respectively.

We can attribute the larger inference errors for smaller delays only in part to the fact that $D_k \neq D'_k$ and $\mathsf{Cov}[D_k, E_k] \neq 0$. Observe, indeed, that especially for the variance estimates, the relative errors are quite large despite $|\mathsf{Cov}[D_k, E_k]| \ll \mathsf{Var}[D_k]$. We ascribe these larger errors to departure of the actual packets delay

from the independence assumption of the model. We calculated the coefficient of correlation of packet delays on consecutive links. The median was 0.09, the maximum value 0.57. We believe that the higher correlations are a result of the small scale of the simulated network. In general, we expect correlations to be smaller in real networks because of the wide traffic and link diversity. The large effect that correlation has on the estimates of the variance can be explained by observing that, because of the independence assumption we ignore all the cross-correlation terms when we derive the expression for the variance estimator (equations (8)-(13)). In the presence of correlation, these terms are not negligible and can be significantly larger than the smaller variances. On the other hand, we observed that estimation of the average delay is more robust and does not significantly suffer from the violation of the independence assumption. This is not unexpected since unlike the variance, the mean of a sum is always equal to the sum of the means irrespective of the underlying correlation structure.

7 Conclusions

In this paper, we explored the use of end-to-end unicast traffic measurements to estimate the delay characteristics of internal network links. Measurement experiments consist of back-to-back packets (a packet pair) sent from a sender to pairs of receivers. We develop efficient techniques to estimate the link delay characteristics, namely, delay distribution and delay variance.

For the estimation of the delay distribution, building on previous work in [11] and the recent work of the authors of [4, 5], we proposed a novel approach for the estimation of the link delay distribution. The key idea is the use a variable bin size model, wherein smaller bins are used in correspondence of concentrations of probability mass and larger bins otherwise. We consider a variable bin size model the analysis of which can be reduced to that of fixed bin size models. Through examples, we showed that, compared to previous approaches, we are able to significantly reduce the computational complexity, without losing accuracy.

We also provided methods to directly estimates the link delay variance. We express the link delay variance in terms of the covariance of the end-to-end delays. Therefore, we can estimate the variance from the sample covariance of the end-to-end delays. The method can be extended to the estimation of higher order cumulants.

Accuracy of the proposed approaches depends on strong correlation between the delay seen by the two packets along the shared path. We verified the degree of correlation in packet pairs through network measurements. We also used simulation to explore the performance of the estimator in practice and observed good accuracy of the proposed inference techniques, although violation of some of the model assumptions, *e.g.*, spatial correlation, introduces systematic errors. This will be object of further study.

References

1. R. Caceres, N.G. Duffield, J.Horowitz and D. Towsley, "Multicast-Based Inference of Network Internal Loss Characteristics", *IEEE Trans. on Information Theory*, November 1999.

2. R. Carter, M. Crovella, 'Measuring bottleneck link-speed in packet-switched networks,' *Performance Evaluation*, 27&28, 1996.
3. M. Coates, R. Nowak, "Network loss inference using unicast end-to-end measurement", *Proc. ITC Conf. IP Traffic, Modeling and Management*, Monterey, CA, September 2000.
4. M. Coates, R. Nowak, "Sequential Monte Carlo Inference of Internal Delays in Nonstationary Communication Networks," submitted for pubblication, Jan 2001.
5. M.J. Coates and R. Nowak, "Network Delay Distribution Inference from End-to-end Unicast Measurement," Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing, May 2001.
6. N.G. Duffield and F. Lo Presti, "Multicast Inference of Packet Delay Variance at Interior Network Links", Proc. IEEE Infocom 2000, Tel Aviv, March 2000.
7. N.G. Duffield, F. Lo Presti, V. Paxson and D. Towsley "Inferring Link Loss Using Striped Unicast Probes", Proc. IEEE Infocom 2001, Anchorage, AK, April 2001.
8. B. Frey. Graphical Models for Machine Learning and Digital Communication. MIT Press, Cambridge London (1998).
9. V. Jacobson, "Congestion Avoidance and Control," *Proc. SIGCOMM '88*, pp. 314-329, August. 1988.
10. S. Keshav. " A control-theoretic approach to flow control," *Proc. SIGCOMM '91*, 3–15, September 1991.
11. F. Lo Presti, N.G. Duffield, J.Horowitz and D. Towsley, "Multicast-Based Inference of Network-Internal Delay Distributions", submitted for publication, September 1999.
12. `ns` – Network Simulator. See http://www-mash.cs.berkeley.edu/ns/ns.html
13. V. Paxson, J. Mahdavi, A. Adams, M. Mathis, "An Architecture for Large-Scale Internet Measurement," IEEE Communications Magazine, Vol. 36, No. 8, pp. 48–54, August 1998.
14. Y. Tsang, M.J. Coates and R. Nowak, "Passive Network Tomography using EM Algorithms," Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing, May 2001.
15. C.F. Jeff Wu, "On the convergence properties of the EM algorithm", Annals of Statistics, vol. 11, pp. 95-103, 1982.

A Computation of $\widehat{\alpha}^{(0)}$

We illustrate the method for the computation of $\widehat{\alpha}^{(0)}$. Let $A_k(d) = \mathsf{P}_\alpha[X_k(1) = d]$, $k \in V$ the probability that the first packet of the pair reaches k in d unit of time. For each pair $\{i,j\} \in Q(k)$, we use the approach in [11] to compute an estimate $\widehat{A}_k^{i,j}(d)$ of $A_k(d)$ from the empirical distribution of $X_{i,j}$ by solving a system of polynomial equations. Since $X_k(1) = X_{f(k)}(1) + D_k$ and $X_{f(k)}$ and D_k are independent we obtain an estimate of the distribution of D_k by deconvolution of the estimates of the distributions of $X_k(1)$ and $X_{f(k)}(1)$. We use this estimate as initial distribution. More precisely, for $k \in V$, we let $\widehat{\alpha}_k^{(0)}(d) = (\widehat{A}_k(d) - \sum_{d' \in \mathcal{Q}, d' \leq d} \widehat{A}_{f(k)}(d')\widehat{\alpha}_k^{(0)}(d-d'))/\widehat{A}_{f(k)}(0)$, $d \in \mathcal{Q} \setminus \infty$, where $\widehat{A}_k(d) = \frac{1}{\#Q(k)} \sum_{\{i,j\} \in Q(k)} \widehat{A}_k^{i,j}(d)$, and let $\widehat{\alpha}_k^{(0)}(\infty) = 1 - \sum_{d \in \mathcal{Q} \setminus \infty} \widehat{\alpha}_k^{(0)}(d)$. It is possible to show that $\widehat{\alpha}^{(0)}$ is a consistent estimator of α and, as n goes to infitity, $\sqrt{n}(\widehat{\alpha}^{(0)} - \alpha)$ converges in distribution to a multivariate Gaussian random variable with mean 0 and covariance matrix σ_α.