

Adaptive Multicast Topology Inference

N.G. Duffield¹ J. Horowitz² F. Lo Presti^{1,3}

¹AT&T Labs–Research
180 Park Avenue
Florham Park, NJ 07932, USA
{duffield,lopresti}@research.att.com

²Dept. Math. & Statistics
University of Massachusetts
Amherst, MA 01003, USA
joeh@math.umass.edu

³Dept. of Computer Science
University of Massachusetts
Amherst, MA 01003, USA

Abstract— The use of end-to-end multicast traffic measurements has been recently proposed as a means to infer network internal characteristics as packet link loss rate and delay. In this paper, we propose an algorithm that infers the multicast tree topology based on these end-to-end measurements. Differently from previous approaches which make only partial use of the available information, this algorithm adaptively combines different performance measures to reconstruct the topology. We establish its consistency and evaluate its accuracy through simulation. We show that in general it requires many fewer probes to correctly identify the topology than other methods.

Keywords. End-to-end measurements, Topology Discovery, Adaptive, Estimation Theory, Multicast Tree.

I. INTRODUCTION

Background and Motivation. As communications networks grows in size and complexity, it has become increasingly important to measure their performance. To overcome the limitations imposed by administrative diversity which *de facto* prevents general direct access to large portions of the network, there has been increasing interest in approaches that aim to characterize the network internal behavior from the sole external end-to-end measurements. Currently, there are several measurements infrastructure projects (including CAIDA [2], Felix [9], IPMA [10], NIMI [15] and Surveyor [18]) that collect and analyze end-to-end measurements across a mesh of paths between hosts.

In these approaches, a fundamental design issue is the type of measurements to be performed across the network and the methodology adopted to infer the internal network behavior in terms of the performance experienced by the measurements hosts. A promising approach, MINC (*Multicast Inference of Network Characteristics*), relies on the use of multicast end-to-end measurements. In contrast to unicast traffic, multicast traffic introduces a well structured correlation in the end-to-end behavior observed by the receivers that share the same multicast session. This in turn allows to draw inferences about the performance characteristics of the internal links without the cooperation of network elements in the path such as packet loss rates, [3], packet delay distributions, [11], and packet delay variance, [6]. There is ongoing work [1] to incorporate some of these techniques into the NIMI measurements infrastructure.

All these inference methods require knowledge of the multicast tree topology. Unfortunately, this is typically unknown. This motivates the need for algorithms that can identify the topology of the tree. Another motivation is that knowledge of the multicast topology can be of use to multicast applications. There are several reliable multicast protocols (e.g., RMTP[14]) which organize receivers in logical hierarchies using the under-

lying topology, if possible. Other applications attempt to identify receivers that share the same network bottleneck [16].

Several algorithms have been proposed for identifying multicast topologies based on the sole loss observations at receivers. An algorithm for inferring the topology of a binary tree was first proposed in [16]. The main idea was the simple observation that as the number of packets grows multicast receivers sharing a longer portion of the multicast distribution tree also have higher shared loss rates; this information could in turn be used to reconstruct the topology by recursively grouping the pair of nodes with the highest shared loss. In [8] the correctness this algorithm was proven and the approach was extended to general topologies by introducing several other loss-based algorithms. More recently, algorithms have been proposed for identifying multicast topologies based on delay measurements instead. By observing that the approach in [16] and [8] can be generalized to any performance measures that (i) monotonically increases as the packet traverse the tree, and (ii) can be estimated on the sole basis of end-to-end measurements at the receivers, in [7] several algorithms are specified based on delay performance measures as link utilization, delay average and delay variance.

The accuracy of these approaches is limited by the fact that each of the above algorithm reconstructs the topology using only the information provided by one single performance measure, e.g., loss rates or delay averages, thus making only partial use of the available measurements. In addition, as shown in [7], no algorithm appears to perform better than the others in general. Our experience has shown that typically under moderate and heavy load network conditions (high link loss and utilization) the loss based algorithm is generally the most accurate while under light load condition (low link loss and utilization), the algorithm based on link utilization performs best. Therefore, it is then not clear which algorithm could be best suited to reconstruct multicast topologies across large internetworks where different portions of the network can experience quite different conditions. In the most general case, the different algorithms could yield quite different reconstructed topologies; clearly, a method which allows to choose among them or better to compose them is much desired.

Contribution. In this paper we propose a new algorithm for identifying multicast topologies based on joint loss and delay measurements at the receivers. This algorithm combines the different performance measures and reconstruct the tree by adaptively choosing step by step that which insures the best accuracy. Intuitively, by so doing we compose the topologies each performance measure would yield by choosing for each portion of the tree its more accurate reconstruction.

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

The key contribution underlying this approach is the ability to determine which performance measure minimizes the probability of making an error. We propose a technique for estimating the probability of incorrect identification of the topology. This is accomplished by a careful enumeration of all the possible erroneous decisions and by estimating the probability of each of them. We also analyze the modes of misclassification and verify that our estimate converges to the true error probability as the number of packets increases. Therefore we can use this estimate to determine the level of accuracy of a given reconstructed topology, or more importantly, the number of probe packets required to achieve a desired level of accuracy.

We establish that the joint algorithm is consistent, *i.e.* the probability of correctly identifying the topology converges to 1 as the number of probes grows to infinity. Analysis of a simple scenario shows that the joint algorithm can significantly outperform any of the algorithms previously considered. We also use simulation to evaluate its accuracy. In all the scenarios considered, we find that the joint algorithm has the best performance, requiring in general many fewer probes to correctly identify the topology than other methods.

In this paper, we will restrict our attention to topology inference based solely on loss and utilization performance measures. A first reason is simplicity; as later shown, the loss process and the utilization process are formally identical once we substitute the event of “packet not lost” with the event of “packet not delayed”; as a consequence the very same results apply in both cases. A second reason is that they also have the lowest computational complexity. Finally, they are the most accurate: as previously mentioned, in most cases, either the loss based or the utilization based algorithms has the best performance. Hence, while the joint algorithm extends to accommodate other performance measures, in practice most of the benefit is achieved by combining the loss and utilization estimators.

Implementation Requirement. In contrast to loss, delay measurements require the deployment of measurements hosts with synchronized clocks. Global Positioning System (GPS) which is used in some of the mentioned measurements infrastructures allows accuracy within tens of microseconds. This is sufficient for accurate utilization measurements which, in particular, require the accurate assessment of the minimum end-to-end delay. We believe this is not the case for the more widely deployed Network Time Protocol [12], which only provides accuracy on the order of tens of milliseconds.

Structure of the Paper. The rest of the paper is organized as follows. In Section II and III we review our model and the loss and utilization topology inference algorithms. In Section IV we introduce the joint loss/utilization algorithm; we also describe the technique for estimating the probability of topology misclassification. In Section V we analyze the performance of the different algorithms. Their accuracy is then evaluated in Section VI through simulation. We conclude in Section VII; some proofs are deferred to the Appendix.

II. MODEL & INFERENCE

Tree Model. The physical multicast tree comprises actual network elements (the nodes), and the communication links than

join them. The logical multicast tree comprises the branch points of the physical tree, and the logical links between them. The logical links comprise one or more physical links. Thus each node in the logical tree, except the leaf nodes and possibly the root, must have 2 or more children. We can construct the logical tree from the physical tree by deleting all links with one child (except for the root) and adjusting the links accordingly by directly joining its parent and child.

Let $\mathcal{T} = (V, L)$ denote a logical multicast tree with nodes V and links L . We identify the root node 0 with the source of probes, and $R \subset V$ will denote the set of leaf nodes (identified as the set of receivers). The set of children of node $k \in V$ is denoted by $d(k)$. For each node k , other than the root 0, there is a unique node $f(k)$, the *parent* of k , such that $(f(k), k) \in L$. We will refer to the link $(f(k), k)$ as simply link k . We shall define $f^n(k)$ recursively by $f^n(k) = f(f^{n-1}(k))$ with $f^1 = f$. We say that j is a descendant of k if $k = f^n(j)$ for some integer $n > 0$, and write the corresponding partial order in V as $j \prec k$. $a(i, j)$ will denote the minimal common ancestor of i and j in the \prec -ordering. For $k \in V$ we let $\mathcal{T}(k) = (V(k), L(k))$ denote the subtree of \mathcal{T} that is rooted at k , and set $R(k) = R \cap V(k)$.

Delay and Loss Model. Probe packets are dispatched down the tree from the root node 0. With multicast, each probe arriving at a node k gives rise to copy sent to each child node of k . On each link, the packet is either lost, or transmitted with some delay. We regard the delay as the sum of two components: a fixed propagation delay, and a variable queueing delay. We represent the latter by a random variable $Z_k \in [0, \infty]$ that specifies the queueing delay encountered by a packet attempting to traverse link k , with $Z_k = \infty$ signifying packet loss. By convention $Z_0 = 0$. The accrued queueing delay for the path from the root to a node k is $Y_k = \sum_{j \succ k} Z_k$. This yields the property that $Y_k = \infty$ for a packet lost on some link between node 0 and k ; likewise $Y_k = 0$ if no queueing delay is encountered on any link of the path.

Let $\alpha_l(k) = \mathbb{P}[Z_k < \infty]$ denote the probability of transmission on link k , and $\alpha_u(k) = \mathbb{P}[Z_k = 0]$ the probability of transmission with no queueing delay. A tree is said to be *canonical* if for all links k , $0 < \alpha_u(k) \leq \alpha_l(k) < 1$. A tree can be reduced to canonical form by (i) removing each link k for which with $\alpha_l(k) = 1$ or $\alpha_u(k) = 1$ and identifying its endpoints; and (ii) pruning all subtree descended from links that have $\alpha_l(k) = 0$ or $\alpha_u(k) = 0$. Henceforth we work exclusively with canonical trees; only for these are the link characteristics uniquely identifiable.

Loss and Utilization Processes. Here it suffices to analyze a projection of the delay processes Z_k . For each $k \in V$ let $X_l(k) = \mathbf{1}_{\{Y(k) < \infty\}}$. We call $X_l = (X_l(k))_{k \in V}$ the **loss process**: $X_l(k) = 1$ if the probe reaches k and 0 otherwise. For each $k \in V$ let $X_u(k) = \mathbf{1}_{\{Y(k) = 0\}}$. We call $X_u = (X_u(k))_{k \in V}$ the **utilization process**: $X_u(k) = 1$ if the probe reaches k with no queueing delay, and 0 otherwise. The name arises since link queueing delay is zero iff the link is not utilized: $1 - \alpha_u(k)$ is hence the link utilization.

We assume the Z_k are independent random variables. Then X_u and X_l are Markov processes on \mathcal{T} . Their structure is for-

mally identical. The loss process satisfies

$$\begin{aligned} X_l(0) &= 1; & X_l(f(k)) &= 0 \Rightarrow X_l(k) = 0; \\ \mathbb{P}[X_l(k) = 1 \mid X_l(f(k)) = 1] &= \alpha_l(k). \end{aligned} \quad (1)$$

The utilization process is formally identical upon replacing the event of “no loss” with that of “no delay”. Then (1) holds when X_l, α_l are replaced by X_u, α_u . In the rest of the paper we will omit the subscripts l and u when the same statement holds for both cases.

Inference of Shared Path Characteristics. When probes are sent down the tree we cannot observe the entire processes X but only the outcomes at the receivers $(X(k))_{k \in R}$. By exploiting the correlation of multicast traffic, in [3] it was shown how the link loss rates can be computed from the distribution of $(X(k))_{k \in R}$ when the topology is known. Here, to infer the topology, we will use the following generalization of the results in [3].

Let $A(k) = \prod_{j \succeq k} \alpha(j)$ denote the probability that a probe reaches node k (the $A_l(k)$ version) or reaches is without queuing delay (the $A_u(k)$ version). A short probabilistic argument shows that for any two nodes i and j , $i, j \neq a(i, j)$,

$$A(k) = A(i, j) := \frac{\mathbb{P}[\vee_{\ell \in R(i)} X(\ell) = 1] \mathbb{P}[\vee_{\ell \in R(j)} X(\ell) = 1]}{\mathbb{P}[\vee_{\ell \in R(i)} X(\ell) = \vee_{\ell \in R(j)} X(\ell) = 1]} \quad (2)$$

where $k = a(i, j)$. (2) expresses the behavior along the shared portion of the path from the source to a pair of nodes in terms of the probabilities of leaf-measurable events.

To infer the probabilities from measurements, consider an experiment in which a set of n probes is dispatched from the source. From the outcomes $(x^{(1)}, \dots, x^{(n)})$ with $x^{(m)} = (X^{(m)}(k))_{k \in R}$, we can estimate $A(k)$ by substituting the probabilities in (2) by their empirical means, obtaining

$$A^{(n)}(i, j) = \frac{\sum_{m=1}^n X^{(m)}(i) \cdot \sum_{m=1}^n X^{(m)}(j)}{n \cdot \sum_{m=1}^n X^{(m)}(i) \cdot X^{(m)}(j)} \quad (3)$$

where we define $X^{(m)}(k) := \vee_{\ell \in R(k)} X^{(m)}(\ell)$. It is possible to show that $A^{(n)} = (A^{(n)}(i, j))_{i, j \in V}$ is consistent ($A^{(n)} \xrightarrow{n \rightarrow \infty} A$ with probability 1) and, as n goes to infinity, $\sqrt{n}(A^{(n)} - A)$ converges in distribution to a multivariate Gaussian random variable with mean 0 and covariance matrix $\sigma_A = \sigma_A(A)$. Details can be found in [8].

A complication arises in case of utilization estimation as we have to account for (i) the presence of the fixed delay component in the experimental data due to propagation delays and (ii) the inherent limitation of time measurements accuracy due to clocks resolution. To this end, we (i) normalize each measurement by subtracting the minimum delay seen at the leaf and (ii) introduce a tolerance τ (typically smaller than $1ms$) in deciding whether a given delay is a “minimum” delay. In other words, operationally we define $X_u^{(m)}(k) = \mathbf{1}_{\{Y^{(m)}(k) - \min_{l=1}^n Y^{(l)}(k) \leq \tau\}}$ where $Y^{(m)}(k)$ is the delay experienced by the m^{th} probe sent to receiver k . This amounts to assign the observed minimum delay as the propagation delay, under the assumption that at least one probe has experienced no queuing delay along the path.

1. *Input:* The set of receivers $R = \{i_1, \dots, i_r\}$
2. $R' := R; V' := R'; L' = \emptyset;$
3. **while** $|R'| > 1$ **do**
4. $U :=$ **select pair** ;
5. $V' := V' \cup \{U\};$
6. $L' := L' \cup \{(U, \ell) : \ell \in U\};$
7. $\alpha(\ell) = A(\ell)/A(j, k), \ell \in U;$
8. $R' := (R' \setminus U) \cup \{U\};$
9. **enddo**
10. $V' := V' \cup \{0\}; L' = L' \cup \{(0, R')\};$
11. *Output:* tree $(V', L');$
12. **procedure select pair**
13. **return** $U = \{j, k\} \subseteq R'$ with minimal $A(j, k);$
14. **end procedure**

Fig. 1. Deterministic Binary Tree Classification Algorithm (DBT).

III. LOSS AND UTILIZATION TOPOLOGY INFERENCE

Deterministic Reconstruction of Binary Trees. Our approach to loss (or utilization) topology inference relies on being able through (2) to identify the characteristics along internal paths of the multicast tree from the probability of measurable events at receivers. The key observation is that $a(j, k) \prec a(j', k')$ implies $A(j, k) < A(j', k')$, from which it follows that the pair $\{j, k\} \subset R$ which has minimal $A(j, k)$ is a sibling pair; a short argument shows that if not, $A(j, k)$ would not be minimal. The idea is to proceed recursively, starting from the receivers, by adding the parent node as sibling are identified. This approach is formalized in the Deterministic Binary Tree Classification Algorithm (DBT); see Figure 1.

DBT operates as follows. R' denotes the current set of nodes from which a pair of siblings will be chosen, initially equal to the receiver set R . We first use the procedure *select pair* below

- procedure select pair**
return $U = \{j, k\} \subseteq R'$ with minimal $A(j, k);$
end procedure

to find the pair $U = \{j, k\}$ that minimizes $A(j, k)$ (line 4). This identifies the members of U as siblings, and the set U is used to represent their parent. Correspondingly, we add U to the list V' of nodes (line 5), $(U, j), (U, k)$ to the list L' of links (line 6), compute $\alpha(j)$ and $\alpha(k)$ by taking the appropriate quotient (line 7) and replace j and k by U in the set R' of nodes available for pairing in the next stage (line 8). This process is repeated until all sibling pairs have been identified (loop from line 3). Finally, we adjoin the root node 0 and the link joining it to its single child (line 10).

We say that DBT reconstructs the binary logical multicast tree (V, L) if given the receiver set R it produces (V, L) as its output.

Theorem 1: Let \mathcal{T} be a binary tree. Then DBT reconstructs \mathcal{T} .

We postpone the proof to the Appendix.

Reconstruction of Binary Trees from Measurements. It is straightforward to derive from DBT an algorithm that estimates the topology from the end to end measurements $(x^{(1)}, \dots, x^{(n)})$. The idea is to estimate \mathcal{T} by the topology $\mathcal{T}^{(n)}$

obtained by using the estimates $A^{(n)}(j, k)$ in place of $A(j, k)$. This amounts to modifying the procedure *select pair* as follows

```

procedure select pair
  return  $U = \{j, k\} \subseteq R'$  with minimal  $A^{(n)}(j, k)$ ;
end procedure

```

Computation of $A^{(n)}(j, k)$ is accomplished via (3); to this end, observe that $X^{(m)}(k) = \vee_{\ell \in d(k)} X^{(m)}(\ell)$, so they can be recursively computed as the tree is reconstructed. It therefore suffices to add the line

4a. **foreach** $m = 1, \dots, n$ **do** $X^{(m)}(U) = X^{(m)}(j) \vee X^{(m)}(k)$;

We call the resulting algorithm the Binary Tree Classification Algorithm (BT).

Theorem 2: With probability 1, $\mathcal{T}^{(n)} = \mathcal{T}$ for sufficiently large n . Hence $\mathcal{T}^{(n)}$ is a consistent estimator of \mathcal{T} , i.e., $\lim_{n \rightarrow \infty} \mathbb{P}[\mathcal{T}^{(n)} \neq \mathcal{T}] = 0$.

Proof of Theorem 2: Since $A^{(n)}(j, k)$ converges almost surely to $A(j, k)$, then, with probability 1, for all sufficiently large n , the relative ordering of the $A^{(n)}(j, k)$ is the same as that of the $A(j, k)$ for pairs j, k for which the $A(j, k)$ are distinct. Hence, for all n sufficiently large, BT reconstructs the tree in the same manner as DBT, except possibly varying the order in which it groups pairs $\{j, k\}$ with identical $A(j, k)$. The last two statements then directly follow by standard results. ■

Finally, observe that in line 7 BT computes an estimate $\alpha^{(n)}(\ell) = A^{(n)}(U)/\alpha^{(n)}(\ell)$ of $\alpha(\ell)$. From Theorem 2 then it immediately follows that as n goes to infinity $\alpha^{(n)}(\ell)$ converges with probability 1 to $\alpha(\ell)$.

Extension to General Trees. Inference of general trees can be accomplished by extending BT. In [8] we propose and analyze different alternatives. The simplest approach, which also turns out to be the most computationally efficient and accurate, proceeds in two steps: first it reconstructs a binary tree using BT; then it applies a threshold ε and prune all links k such that $\alpha^{(n)}(k) > 1 - \varepsilon$. The idea comes from the observation that the application of DBT to an arbitrary tree results in a binary tree in which all links k which do not exist in the original tree satisfy $\alpha(k) = 1$. In BT, the use of a threshold ε accounts for the statistical variability of the estimates.

IV. A JOINT LOSS-UTILIZATION ALGORITHM

We now extend the framework for topology inference by proposing an algorithm which combines loss and utilization measurements. We contrast this to BT which is based on a single performance measure. The idea consists in reconstructing the topology by adaptively choosing at each step the performance measures which insures the best accuracy. We describe the algorithm below. The algorithm bases its decisions on estimates of the probability of misclassification. In the remainder of the section we will present a technique for estimating this probability.

The Joint Loss-Utilization Classification Algorithm. The joint algorithm proceeds like BT by recursively grouping nodes starting from the set of receivers. Differently from BT, here we choose at each step the performance measure on which to base

the grouping decision; more precisely, at each step we determine the two pairs that minimize $A^{(n)}_l(\cdot, \cdot)$ and $A^{(n)}_u(\cdot, \cdot)$ and group that which also minimizes the probability of making an error. Specifically, we modify the procedure *select pair* as follows

```

procedure select pair
  foreach  $X \in \{l, u\}$ 
    select  $U_X = \{j_X, k_X\} \subseteq R'$  with
      minimal  $A^{(n)}_X(j_X, k_X)$ ;
  return  $U = \{j, k\} = \operatorname{argmin}_{\{j_X, k_X\}, X \in \{l, u\}} P_{X, R'}^{f, (n)}$ ;
end procedure

```

where $P_{X, R'}^{f, (n)}$ denotes the (estimated) probability of misclassification, given the current set of nodes R' , pairing nodes according to performance measure X . We will detail how to compute this estimate in Section IV-A.

We call the resulting algorithm the Joint Binary Tree Classification Algorithm (JBT). Denote $\mathcal{T}_j^{(n)}$ the topology obtained by JBT.

Theorem 3: With probability 1, $\mathcal{T}_j^{(n)} = \mathcal{T}$ for sufficiently large n . Hence $\mathcal{T}_j^{(n)}$ is a consistent estimator of \mathcal{T} , i.e., $\lim_{n \rightarrow \infty} \mathbb{P}[\mathcal{T}_j^{(n)} \neq \mathcal{T}] = 0$.

We formalize the proof in the Appendix. The intuition beyond the proof is that, for all sufficiently large n , with probability 1, the relative ordering of the $A^{(n)}(j, k)$ is the same as that of $A(j, k)$ (which observe can be different for loss and utilization) from which it follows that the two pairs of nodes which minimize $A_l(\cdot, \cdot)$ and $A_u(\cdot, \cdot)$ are both siblings pairs.

Extension to General Trees. Inference of general trees is accomplished by reconstructing a binary tree using JBT first and by then pruning all links k such that $\alpha_l^{(n)}(k) > 1 - \varepsilon_l$ and $\alpha_u^{(n)}(k) > 1 - \varepsilon_u$, where we use possibly different loss and utilization thresholds, ε_l and ε_u . The estimates $\alpha_l^{(n)}(k)$ and $\alpha_u^{(n)}(k)$ are computed in line 7 of JBT by taking the appropriate ratio.

A. Estimation of the Misclassification Probability

In this section we describe the estimate of the probability of misclassification that is used in JBT. Classification proceeds by a sequence of comparison operations; the analysis of misclassification is therefore potentially complex due to the need to analyze a large number of statistically dependent modes of failure. Our approach to this is to divide and conquer. Correct classification requires correct ordering of quantities $A(j, k)$ in a number of comparison. For each such comparison, we approximate the probability of incorrect ordering in terms of the tail probability of a Gaussian random variable whose variance we calculate. For large numbers of probes, the probability of misclassification is dominated by the largest such misordering probability.

The generic comparison involves three nodes j, k and l , where $a(j, k) \neq a(j, l)$. Since $a(j, k) \prec a(j, l)$ iff $A(j, k) < A(j, l)$, the correct dependency relation between $a(j, k)$ and $a(j, l)$ is identified if

$$D^{(n)}(j, k, l) := A^{(n)}(j, l) - A^{(n)}(j, k) \quad (4)$$

has the same sign as its deterministic counterpart $D(j, k, l) = A(j, l) - A(j, k)$. Let $Q(j, k, l)$ denote this event.

The following theorem, essentially proved for loss-based classification in [8], characterizes the asymptotic behavior of $D^{(n)}(j, k, l)$ first for large n , then for small loss and delays. Denote $\bar{\alpha} = 1 - \alpha$ and let $s(k) := \sum_{l \preceq k} \bar{\alpha}(k)$.

Theorem 4: For each triple (j, k, l) , $\sqrt{n} \cdot (D^{(n)}(j, k, l) - D(j, k, l))$, (j, k, l) , converges in distribution, as the number of probes $n \rightarrow \infty$, to a Gaussian random variable with mean 0 and variance $\sigma^2(j, k, l)$. Moreover, as $\|\bar{\alpha}\| = \max_{k \in V} \bar{\alpha}(k) \rightarrow 0$:

- (i) $D(j, k, l) = s(a(j, l)) - s(a(j, k)) + O(\|\bar{\alpha}\|^2)$;
- (ii) $\sigma^2(j, k, l) = |s(a(j, l)) - s(a(j, k))| + O(\|\bar{\alpha}\|^2)$;

Measurements yield the statistic $D^{(n)}(j, k, l)$ with which to infer the descendency relations. From this we would infer $a(j, k) \succ a(k, l)$ if and only if $D^{(n)}(j, k, l) > 0$. Misordering occurs when $D(j, k, l)$ and $D^{(n)}(j, k, l)$ have opposite signs. For large n , Theorem 4 suggests the following approximation for the probability of misordering

$$P[Q^c(j, k, l)] \approx \Psi \left(-\sqrt{n} \frac{|D(j, k, l)|}{\sigma(j, k, l)} \right) \quad (5)$$

where Ψ is the cdf of the standard normal distribution. Since $D(j, k, l)$ and $\sigma^2(j, k, l)$ are unknown, we need to estimate them first. The idea is to simply estimate $D(j, k, l)$ by $D^{(n)}(j, k, l)$. For the variance, we use the fact that $\sigma^2(j, k, l)$ is a continuous function \mathcal{D}_{jkl} of A , $\sigma^2(j, k, l) = \text{Var}[A^{(n)}(j, l)] + \text{Var}[A^{(n)}(j, k)] - 2\text{Cov}[A^{(n)}(j, l), A^{(n)}(j, k)] = (\sigma_{A(j, l)(j, l)} + \sigma_{A(j, k)(j, k)} - 2\sigma_{A(j, l)(j, k)})/n = \mathcal{D}_{jkl}(A)$, and estimate it by $\sigma^{(n)2}(j, k, l) = \mathcal{D}_{jkl}(A^{(n)})$. We thus approximate the probability of incorrect ordering $P[Q^c(j, k, l)]$ by

$$P_{jkl}^{f, (n)} := \Psi \left(-\sqrt{n} \frac{|D^{(n)}(j, k, l)|}{\sigma^{(n)}(j, k, l)} \right) \quad (6)$$

where we used in place of $D(j, k, l)$ and $\sigma^2(j, k, l)$ their estimates. The accuracy of (6) relies on the convergence of the estimates $D^{(n)}(j, k, l)$ and $\sigma^{(n)2}(j, k, l)$. We will verify this in Section VI.

Misclassification Probability Estimate. Consider now the ℓ -th step of JBT(or BT) and denote by $R_\ell^{(n)}$ the current set of nodes and $\{j_n, k_n\} \subset R_\ell^{(n)}$ the pair with minimal $A^{(n)}(j_n, k_n)$. This pair is chosen on the basis of the orderings $D^{(n)}(j, k, l) > 0$ for each triple $(j, k, l) \in \mathcal{S}(R_\ell^{(n)}) = \{(j_n, k_n), (k_n, j_n)\} \times (R_\ell^{(n)} \setminus \{j_n, k_n\})$. With each such ordering we associate a misordering probability $P_{jkl}^{f, (n)}$ as in (6). From the union bound $P[\cup_{\mathcal{S}(R_\ell^{(n)})} Q^c(j, k, l)] \leq \sum_{\mathcal{S}(R_\ell^{(n)})} P[Q^c(j, k, l)]$ we associate with the selection of (j_n, k_n) an estimated misclassification probability through the sum

$$P_{R_\ell^{(n)}}^{f, (n)} = \sum_{(j, k, l) \in \mathcal{S}(R_\ell^{(n)})} P_{jkl}^{f, (n)} \quad (7)$$

$$\approx \max_{(j, k, l) \in \mathcal{S}(R_\ell^{(n)})} P_{jkl}^{f, (n)} \quad (8)$$

$$= \Psi \left(-\sqrt{n} \min_{(j, k, l) \in \mathcal{S}(R_\ell^{(n)})} \frac{|D^{(n)}(j, k, l)|}{\sigma^{(n)}(j, k, l)} \right). \quad (9)$$

This is the misclassification estimate we use in JBT. The approximation arises because for large n , the term with the smallest argument $|D^{(n)}(j, k, l)|/\sigma^{(n)}(j, k, l)$ will dominate the rest.

Observe that $P_{R_\ell^{(n)}}^{f, (n)}$, $D^{(n)}(j, k, l)$ and $\sigma^{(n)}(j, k, l)$ can be directly computed from $\{A^{(n)}(j, k) : \{j, k\} \in R_\ell^{(n)}\}$. Furthermore, when selecting between the loss and utilization methods during step ℓ , we need only select that with the smallest composite argument $\min_{(j, k, l) \in \mathcal{S}(R_\ell^{(n)})} |D^{(n)}(j, k, l)|/\sigma^{(n)}(j, k, l)$.

Topology Misclassification Probability Estimate. (7) associates a misclassification probability estimate with a single grouping decision. Using a simple union bound argument, we can also associate a misclassification probability estimate with the entire reconstructed topology $\mathcal{T}^{(n)}$. In JBT, since we group the pair of nodes which yields the smallest $P_{R_\ell^{(n)}}^{f, (n)}$, we can estimate the topology misclassification probability by summing over the minimum between the loss and utilization misclassification estimates,

$$P_j^{f, (n)} := \sum_{\ell=1}^{|V \setminus R| - 1} \min\{P_{l, R_\ell^{(n)}}^{f, (n)}, P_{u, R_\ell^{(n)}}^{f, (n)}\}. \quad (10)$$

It is easy to realize that we can also associate a misclassification probability estimate to the topology inferred by BT. The difference is that it is simply computed by summing over (7), *i.e.*, $P^{f, (n)} := \sum_{\ell=1}^{|V \setminus R| - 1} P_{R_\ell^{(n)}}^{f, (n)}$. In Section VI we will illustrate applications of these estimates.

V. ANALYSIS OF CLASSIFIER PERFORMANCE

A. Performance of Single Classifier using BT

The analysis of the actual misclassification probabilities mirrors much of the previous analysis. Consider a node $i \in V$ which is to be identified during the step ℓ of BT. Let $h(i)$ and $h^*(i)$ denote its two children. Correct identification of i occurs if neither $h(i)$ nor $h^*(i)$ is incorrectly paired with some other element of R_ℓ , the set of nodes available for pairing at step ℓ . Thus, the event of correct classification at step ℓ is $Q_\ell = \cap_{(j, k, l) \in \mathcal{S}(R_\ell)} Q(j, k, l)$ where $\mathcal{S}(R_\ell) = \{(h(i), h^*(i)), (h^*(i), h(i))\} \times (R_\ell \setminus \{h(i), h^*(i)\})$. Correct classification of the whole tree is the event $Q = \cap_{\ell=1}^{|V \setminus R| - 1} Q_\ell$.

Now, the various $Q(j, k, l)$ are not independent events, and neither are the Q_ℓ . However, we can use union bounds to bound above the probability of misclassification:

$$P^f := P[Q^c] \leq \sum_{\ell=1}^{|V \setminus R| - 1} P_{R_\ell}^f, \quad \text{where} \quad (11)$$

$$P_{R_\ell}^f := P[Q_\ell^c] \leq \sum_{(j, k, l) \in \mathcal{S}(R_\ell)} P[Q^c(j, k, l)] \quad (12)$$

According to Theorem 4, then for large n , these sums will be dominated by the expression $\Psi(-\sqrt{n}\beta)$ where

$$\beta = \min_{\ell=1}^{|V \setminus R| - 1} \min_{(j, k, l) \in \mathcal{S}(R_\ell)} \frac{D^2(j, k, l)}{\sigma^2(j, k, l)} \quad (13)$$

For large n , the approximation for $\log P^f$ is asymptotically linear in n with negative slope $\beta/2$. A simple approximation is thus $P^f \approx e^{-n\beta/2}$.

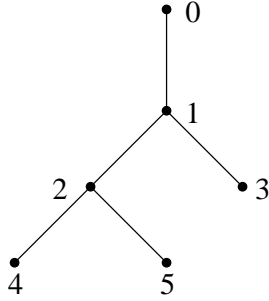


Fig. 2. THE THREE-RECEIVER BINARY TREE.

If we consider the asymptotic regime of small loss and delay, $\|\bar{\alpha}\| \rightarrow 0$, from relations (i) and (ii) in Theorem 4 it follows that

$$\min_{(j,k,l) \in \mathcal{S}(R_\ell)} \frac{D^2(j,k,l)}{\sigma^2(j,k,l)} = \bar{\alpha}(i) + O(\|\bar{\alpha}\|^2), \quad (14)$$

the minimum being attained, for small enough $\|\bar{\alpha}\|$, where $a(j,k) = i$ and $a(j,l) = f(i)$. Picking the dominant contribution to (11) then $\beta \approx \inf_{i \in V \setminus R} \bar{\alpha}(i)$ yielding $P^f \approx e^{-n\bar{\alpha}(i)/2}$. Thus, in this regime, the probability of correctly identifying the topology is controlled by the smallest loss rate or link utilization.

The above argument can be formalized using Large Deviation theory. However, calculation, of the decay rate appears computationally infeasible, although the leading exponent $\inf_{i \in V \setminus R} \bar{\alpha}(i)$ can be recovered in the small $\|\bar{\alpha}\|$ regime.

B. Comparative Performance of Loss and Utilization-Based Classifiers

As an example we consider the three receiver tree with uniform link probabilities $\alpha_u(k) = \alpha_u$ and $\alpha_l(k) = \alpha_l$; see Figure 2. The topology is correctly inferred when nodes 4 and 5 are grouped together; this requires $A^{(n)}(4,5) < A^{(n)}(4,3)$ and $A^{(n)}(5,4) < A^{(n)}(5,3)$. The argument controlling the misclassification probability is $\beta = D^2(4,5,3)/\sigma^2(4,5,3) = D(5,4,3)^2/\sigma^2(5,4,3)$. We plot this as a function of the common probability α in Figure 3. The curve is approximately linear in $\bar{\alpha}$ for small $\bar{\alpha} = 1 - \alpha$, in agreement with (14). As $\bar{\alpha}$ increases, β reaches a maximum at about $\bar{\alpha} = 0.2$ ($\alpha = 0.8$), then decreases to 0. Thus in this homogeneous tree, the misclassification probability is minimized when $\bar{\alpha} \approx 0.2$.

We compare the relative performance of the loss and utilization classifiers in Figure 4, indicating the regions where each of the relevant slopes β_u, β_l is higher. The loss classifier is best when loss rates are higher than about 0.2 (i.e., $\alpha_l \leq 0.8$) or when utilization is high (i.e., low α_u). However, it is outperformed by the utilization classifier when there is low utilization (i.e. high α_u).

C. Performance of JBT

In this case, the analysis of the misclassification probability is complicated by the fact that JBT uses the misclassification estimates to take grouping decisions. Here, to illustrate its modes of misclassification and assess its relative benefit with respect to BT we analyze the performance of JBT in the three

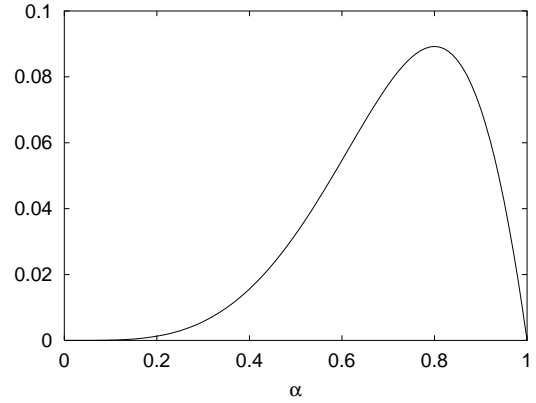


Fig. 3. THREE-RECEIVER TREE. Asymptotic slope of misclassification probability for a single classifier, as function of uniform link probability α

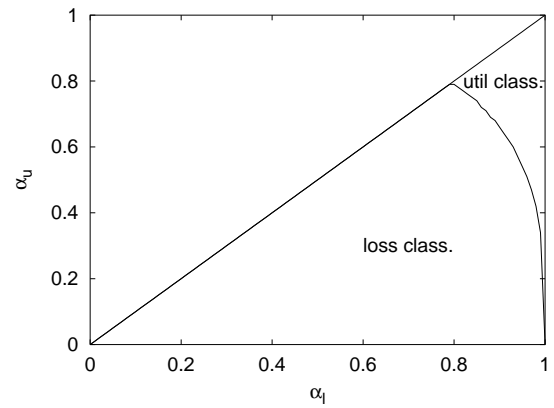


Fig. 4. THREE-RECEIVER TREE. Partition of parameter space (α_l, α_u) where loss or utilization estimator has better performance, i.e. largest asymptotic slope for misclassification probability. Note $\alpha_l < \alpha_u$.

receiver binary tree scenario in Figure 2 with uniform link probabilities. In JBT, the topology is correctly inferred when for the chosen performance measure $A^{(n)}(4,5) < A^{(n)}(4,3)$ and $A^{(n)}(4,5) < A^{(n)}(5,3)$. To keep the complexity manageable, we focus on the first event and assume misclassification occurs when $A^{(n)}(4,5) \geq A^{(n)}(4,3)$, i.e., when $D^{(n)}(4,5,3) < 0$.

The behavior of the classifier is then completely characterized by the bivariate random variable $x^{(n)} = (x_l^{(n)}, x_u^{(n)})$ where $x^{(n)} = \frac{D^{(n)}(4,5,3)}{\sigma^{(n)}(4,5,3)}$. From (6), the misclassification estimate for both performance measures is $P_{453}^{f,(n)} = \Psi(-\sqrt{n}|x^{(n)}|)$; the joint algorithm groups the nodes based on loss information when $|x_l^{(n)}| \geq |x_u^{(n)}|$ and on utilization otherwise (we assume ties are resolved in favor of loss). Misclassification occurs when the chosen performance measure results in grouping the wrong pair; this happens when $|x_l^{(n)}| \geq |x_u^{(n)}|$ and $x_l^{(n)} < 0$ or when $|x_u^{(n)}| > |x_l^{(n)}|$ and $x_u^{(n)} < 0$ which simply amounts to the condition $x_l^{(n)} + x_u^{(n)} \leq 0$. The misclassification probability is then

$$P_j^f := \mathbb{P}[x_l^{(n)} + x_u^{(n)} \leq 0] \quad (15)$$

Normal Approximation. We now consider the asymptotic behavior of P_j^f . An application of the Delta method (see Chapter 7 of

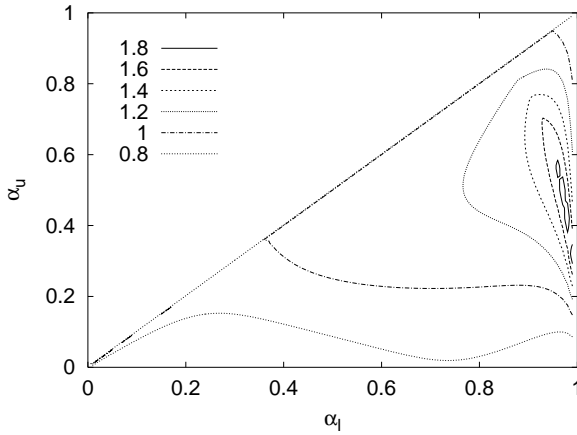


Fig. 5. JOINT CLASSIFIER. Contour plot of the ratio of the (log-scale) misclassification probability asymptotic slope between the joint and best basic classifier.

[17]) shows that as $n \rightarrow \infty$, $\sqrt{n}(\mathbf{x}^{(n)} - \mathbf{x})$, where $\mathbf{x} = (x_l, x_u)$, $\mathbf{x} = \frac{D(4,5,3)}{\sigma(4,5,3)} = \mathcal{H}(A)$ with continuous \mathcal{H} , converges in distribution to a bivariate Gaussian random variable with mean zero and covariance matrix $\sigma_{\mathbf{x}} = (\nabla \mathcal{H}(A_l), \nabla \mathcal{H}(A_u)) \cdot \sigma_{A_l, A_u} \cdot (\nabla \mathcal{H}(A_l), \nabla \mathcal{H}(A_u))^T$, where σ_{A_l, A_u} the asymptotic covariance matrix of $\sqrt{n} \cdot (A_l, A_u)$ and \cdot^T denotes the transpose. (σ_{A_l, A_u} can be computed generalizing the approach used in [8] to compute σ_A .)

Therefore, we have the following approximation

$$\begin{aligned} P_j^f &\approx \int_{x_l^{(n)} + x_u^{(n)} \leq 0} e^{-\frac{n}{2}(\mathbf{x}^{(n)} - \mathbf{x}) \cdot \sigma_{\mathbf{x}}^{-1} \cdot (\mathbf{x}^{(n)} - \mathbf{x})^T} d\mathbf{x} \\ &\approx e^{-\frac{n}{2} \inf_{x_l^{(n)} + x_u^{(n)} = 0} (\mathbf{x}^{(n)} - \mathbf{x}) \cdot \sigma_{\mathbf{x}}^{-1} \cdot (\mathbf{x}^{(n)} - \mathbf{x})^T} \end{aligned} \quad (17)$$

where for large n , we consider the leading exponential order. The infimum in (17) is $x_j^2 = (\mathbf{x}' - \mathbf{x}) \cdot \sigma_{\mathbf{x}}^{-1} \cdot (\mathbf{x}' - \mathbf{x})^T$, where $\mathbf{x}' = (x'_l, x'_u) = (x'_l, -x'_l)$ is the tangent point between the line $x_l^{(n)} + x_u^{(n)} = 0$ and the ellipse of the family $(\mathbf{x}^{(n)} - \mathbf{x}) \cdot \sigma_{\mathbf{x}}^{-1} \cdot (\mathbf{x}^{(n)} - \mathbf{x})^T = a^2$ parameterized in a . Thus, as n goes to infinity we expect the curve $\log P_j^f$ vs. n being asymptotically linear with negative slope $x_j^2/2$. A simple approximation is then $P_j^f \approx e^{-n x_j^2/2}$. Moreover, the minimizing pair $(x_l^{(n)}, x_u^{(n)}) = (x'_l, -x'_l)$ indicates that misclassification most likely occurs by having the two estimated misclassification probabilities equal, loss and utilization yielding two different pairs for grouping, and picking the wrong pair.

To illustrate the results, we study the relative performance of JBT by comparing the asymptotic slope of the logarithm of the misclassification probability x_j^2 with that of the best single classifier. This is computed by considering the leading exponential order approximation $P^f \approx \Psi\left(-\sqrt{n} \frac{D(4,5,3)}{\sigma(4,5,3)}\right) \approx e^{-n x^2/2}$ of the misclassification probability in BT. Figure 5 shows the contour plot of the ratio $\frac{x_j^2}{\max\{x_l^2, x_u^2\}}$ of the (log-scale) asymptotic slopes as function of link characteristics. (α_l, α_u) . JBT performs better than either version of BT for a significant range of values (the region within the contour line corresponding to 1). The performance improvement is more pronounced in the

region where the loss and utilization classifiers have similar performance (which corresponds to the line separating the two regions in Figure 4) and loss and utilization estimates have low correlation (which occurs when $\alpha_l \gg \alpha_u$). This is not surprising since we expect that: (i) little improvement can be achieved when one classifier significantly outperforms the other; and (ii) strong correlation offsets the benefits of using both loss and utilization estimates.

To show the effect of correlation, consider the case $x_l = x_u$, i.e., when the loss and utilization classifiers have the same performance. In this case, it is easy to verify that $x_j^2 = \frac{2}{1+\rho} x_l^2$, where ρ denotes the coefficient of correlation of $x_l^{(n)}$ and $x_u^{(n)}$. At one extreme, $\rho = 1$ and $x_j^2 = x_l^2$, i.e., $P_j^f = P^f$: we have maximal correlation between the loss and utilization classifiers and JBT cannot provide any performance improvement; at the other extreme, $\rho = 0$ and $x_j^2 = 2x_l^2$, i.e., $P_j^f = P_l^f P_u^f$: we have statistical independence and the probability of misclassification is the product of the two misclassification probabilities.

From Figure 5 we also observe that JBT does not always provide better performance. In this example, we have that under very high or very low utilization the loss and utilization classifiers, respectively, have better performance than JBT. In these cases, because of the high variance of the misclassification probabilities estimates, JBT is likely to mistakenly give preference to the worst performance measure.

VI. EXPERIMENTAL EVALUATION

In this section we evaluate the performance of JBT and compare it with that of BT through two types of simulation. In *model simulations* delay and loss are chosen to follow our statistical model, allowing us to test algorithm performance in the setting on which our analysis is based. *Network simulations*, using the ns [13] simulator, test the algorithms in a more realistic setting, where delay and loss are due to queueing delay and buffer overflows at nodes as multicast probes compete with background TCP/UDP traffic.

Model Simulation. We conducted 10000 experiments over randomly generated 15 node binary trees. In Figure 6, we plot the fraction of incorrectly classified topologies as a function of the number of probes for the different classifiers. We considered two regimes: a light load regime with low loss (randomly chosen between 1% and 5%) and utilization (randomly chosen between 10% and 40%), and a heavy load regime with higher loss (randomly chosen between 1% and 20%) and utilization (randomly chosen in between 30% and 80%).

In both cases, the joint classifier dramatically outperform the loss and utilization classifiers with a difference in accuracy already of more than one order of magnitude in accuracy for just 400 probes.

The accuracy of our approach to joint classification lies in that of the misclassification probability estimates. In Figure 6 we also superimposed the mean over the experiments of the topology misclassification probability estimates. From the Figure, we observe that the curves well track the actual slopes, bound from above the actual values and preserve their relative order.

We can use the topology misclassification probability estimate to determine the number of probes required to achieve a

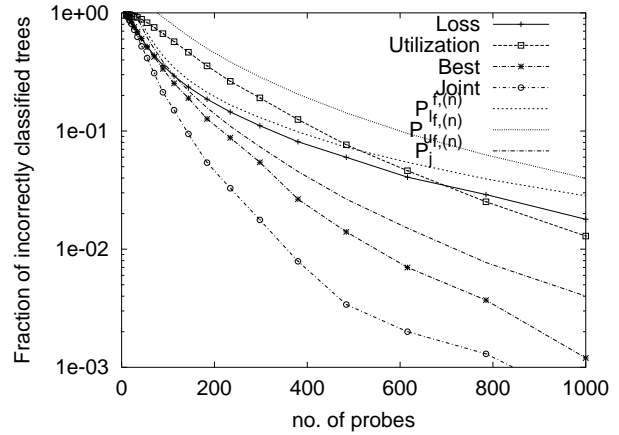
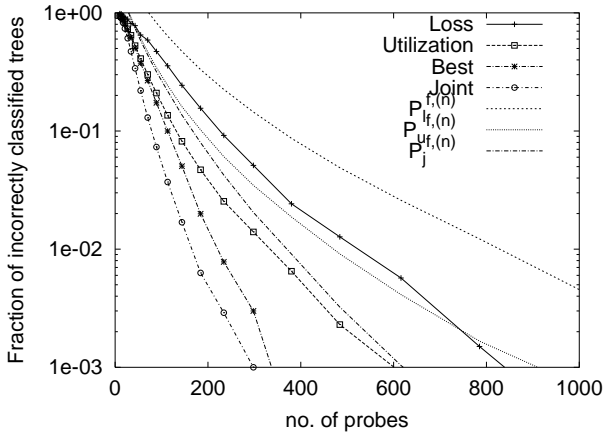


Fig. 6. MODEL SIMULATION. Fraction of incorrectly classified topologies and misclassification estimates for different classifiers as function of number of probes: (a) light load scenario; (b) heavy load scenario.

JBT			
δ	0.05	0.1	0.2
fract. of mis. topologies	0.003	0.008	0.032
average # of probes	145	117	86

BT (loss)			
δ	0.05	0.1	0.2
fract. of mis. topologies	0	0	0.011
average # of probes	415	318	240

TABLE I

ACCURACY OF THE INFERRED TOPOLOGY. FRACTION OF MISCLASSIFIED TOPOLOGIES AND AVERAGE NUMBER OF DISPATCHED PROBES FOR DIFFERENT VALUES OF δ .

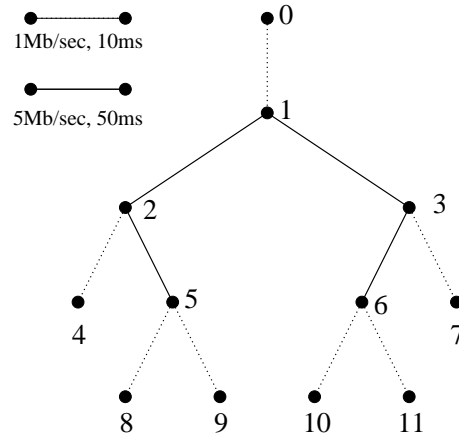


Fig. 7. ns SIMULATION TOPOLOGY.

desired level of accuracy of the inferred topology. The idea is to proceed by dispatching probes until the estimated misclassification probability is below a given threshold δ corresponding to a desired level of accuracy. Thus, for example, to insure a probability of misclassification no greater than 0.05, we send probes until $P_{f,(n)}^{l,u} \leq 0.05$.

We performed 1000 experiments over random generated 15 node binary trees. In each experiment probes were dispatched until the misclassification probability fell below a given threshold δ and we verified whether the inferred topology was correct. For JBT and BT under the light load regime, we summarise the results in Table I where, for different values of δ , we display the average number of probes that were dispatched and the fraction of topologies that were misclassified. Since the estimate bounds from above the misclassification probability, it is no surprise that the fraction of misclassified topologies is well below the chosen threshold. Observe that the number of probes required by JBT is about one third of those required by BT with loss.

Finally, to illustrate the benefit of combining loss and utilization measurements we compare JBT with a simpler approach which simply consists in choosing among the inferred topologies separately computed with the loss and utilization classifiers

that with the smallest misclassification probability estimate. Denote $\mathcal{T}_X^{(n)}$ the topology inferred by classifier X , $X \in \{l, u\}$ and $P_X^{f,(n)}$ its estimated probability of misclassification. We select $\mathcal{T}_{best}^{(n)} = \mathcal{T}_Y^{(n)}$, where $Y = \operatorname{argmin}_{X \in \{l, u\}} P_X^{f,(n)}$. In Figure 6 we also superimposed the fraction of times $\mathcal{T}_{best}^{(n)}$ was incorrect. This approach yields more accurate results than either loss and utilization classifiers, yet not as accurate as JBT: the distance from the JBT curve quantifies the significant gain achievable by the adaptive scheme which use both performance measures; the fact the two curves are parallel suggests that misclassification is ultimately dominated by the same event in both cases.

TCP/UDP Network Simulation. The ns simulations used the topology shown in Figure 7. We arranged for some heterogeneity with the interior links having higher capacity (5Mb/sec) and propagation delay (50ms) than at the edge (1Mb/sec and 10ms). Each link is modeled as a FIFO queue with a 20-packets buffer capacity.

The root node 0 generates probes as a 20Kbit/s stream comprising 40 byte UDP packets according to a Poisson process with a mean interarrival time of 16ms. The background traffic comprises a mix of infinite data source TCP connections (FTP) and

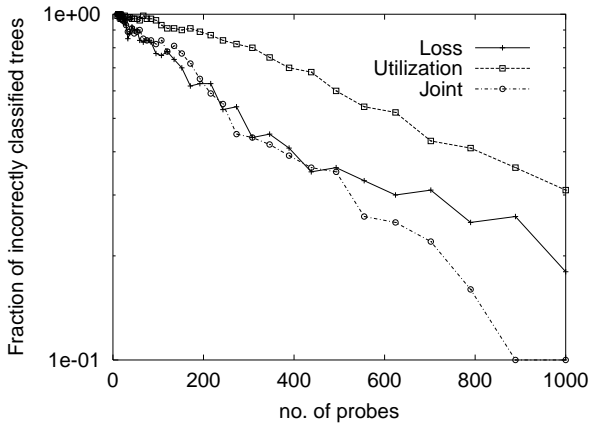


Fig. 8. ns SIMULATION. Fraction of incorrectly classified topologies for different classifiers as function of the number of probes.

exponential on-off sources using UDP. Averaged over the different simulations, the link loss ranges between 1% and 13% and link utilization ranges between 10% and 88%.

Figure 8 plots the fraction of incorrectly identified topologies over 100 simulations. The relative accuracy among the different classifiers is in good agreement with the results from the model simulations. Performance of the utilization and joint classifiers are somewhat inferior due to: (i) wide spread of link utilization values among the different links; (ii) presence of spatial correlation among probe delays. In the simulations, probes are more likely to experience similar level of congestion on consecutive or sibling links than dictated by the modes independence assumption. We calculated the off-diagonal elements of the correlation matrix of the actual link delays. The mean was 0.021 and the maximum 0.17. Despite correlation affected its accuracy, JBT shows, albeit reduced, performance gain over BT.

In the simulations we also observed the presence of short-term temporal correlation among successive probes that encountered the same congestion events. This does not affect estimator consistency, although the convergence rate may be slowed.

VII. CONCLUSIONS

In this paper we have presented an algorithm for the inference of the multicast tree topology from end-to-end measurements. The algorithm combines different performance measures and reconstruct the tree by adaptively choosing that which insures the best accuracy. This is accomplished by a careful enumeration of all the possible erroneous decisions and by estimation of their probability. These estimates in turn can be used to determine the number of probe packets to achieve a desired level of accuracy.

We investigated the statistical properties of the algorithm and showed that it is consistent. Analysis of a simple scenario showed that it can significantly outperform any of the algorithms previously considered. We also used simulation to evaluate its accuracy and found out that, in general, it required many fewer probes to correctly identify the topology than other approaches. ns experiments showed that spatial correlation negatively affects its accuracy. We believe that diversity of traffic in real networks makes large and long lasting correlation unlikely. We are

currently investigating the effect of correlation on the accuracy of topology inference algorithms; this is part of a more general effort to characterize network traffic correlation and its effects on end-to-end measurements based inference.

Acknowledgment. We thank Don Towsley for useful comments and suggestions.

APPENDIX

The proof of Theorem 1 is based on the following result. We will find it useful to identify a subset S of V as a **stratum** if $\{R(k) : k \in S\}$ is a partition of R .

Lemma 1: Let S be a stratum. Then,

(i) a pair of nodes $\{j, k\} \subseteq S$ are siblings if and only if

$$A(j, k) < \min_{\{j', k'\} \subseteq S: |\{j', k'\} \cap \{j, k\}| = 1} A(j', k'); \quad (18)$$

(ii) if $\{j, k\} \subseteq S$ are such that

$$A(j, k) = \min_{\{j', k'\} \subseteq S} A(j', k') \quad (19)$$

then $\{j, k\}$ are sibling;

(iii) if $\{j, k\} \subseteq S$ is a pair of sibling nodes, then $(S \setminus \{j, k\}) \cap \{a(j, k)\}$ is a stratum.

Proof. Observe first that by definition of stratum, if $j \in S$, then no ancestor or descendent of j can belong to S . (i) the *only* part follows from the observation that if j and k are sibling, then $a(j, k) \prec a(j, \ell), a(\ell, k)$ for any $\ell \in S \setminus \{j, k\}$ which implies $A(j, k) < A(j, \ell), A(\ell, k)$. For the *if* part assume that $\{j, k\} \subseteq S$ satisfies (18) and suppose j and k are not siblings. Let ℓ be the sibling of j . Then, $\ell \notin S$ since, if $\ell \in S$, $a(j, \ell) \prec a(j, k)$ implies $A(j, \ell) < A(j, k)$, contradicting (18). Thus, since S is a stratum, there is a set of nodes $T = \{t_1, \dots, t_{n_\ell}\} \subseteq V(\ell) \cap S$ such that $\cup_{i=1}^{n_\ell} R(t_i) = R(\ell)$ since otherwise $\cup_{i \in S} R(i)$ would not cover R . Now either $k \in T$ or $k \notin T$. But $k \in T$ implies that $a(i, k) \prec a(j, k), i \in T$ so that $A(i, k) < A(j, k)$ contradicting (18) while $k \notin T$ implies that $a(j, i) \prec a(j, k), i \in T$ so that again $A(j, i) < A(j, k)$ contradicts (18). Therefore j and k are siblings. (ii) then is an immediate consequence of (i) and (iii) follows immediately from the definition of stratum. ■

Proof of Theorem 1. It suffices to observe that in DBT, at the beginning of each iteration, R' is a stratum; therefore, the pair of nodes which minimizes $A(\cdot, \cdot)$ is always a pair of sibling nodes. This property holds before the first loop (R is a stratum), and (ii) and (iii) of Lemma 1 ensure it holds subsequently. ■

Proof of Theorem 3. Since $A^{(n)}(j, k)$ converges almost surely to $A(j, k)$, then, with probability 1, for all sufficiently large n , the relative ordering of the $A^{(n)}(j, k)$ is the same as that of $A(j, k)$ (which observe can be different for loss and utilization). Then, it suffices to observe that for all sufficiently large n , the two pairs of nodes which minimize $A_l(\cdot, \cdot)$ and $A_u(\cdot, \cdot)$ are both siblings provided R' is a stratum. This property holds before the first loop (R is a stratum), and (iii) of Lemma 1 insure it holds subsequently, irrespectively of the actual pair of nodes selected for grouping. Then the last two statements directly follows from standard results.

REFERENCES

- [1] A. Adams, T. Bu, R. Caceres, N.G. Duffield, T. Friedman, J. Horowitz, F. Lo Presti, S.B. Moon, V. Paxson, D. Towsley, "The Use of End-to-End Multicast Measurements for Characterizing Internal Network Behavior", *IEEE Communications Magazine*, May 2000.
- [2] CAIDA: Cooperative Association for Internet Data Analysis. For more information see <http://www.caida.org>
- [3] 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.
- [4] R. Caceres, N.G. Duffield, J.Horowitz F. Lo Presti and D. Towsley, "Statistical Inference of Multicast Network Topology", *Proc. IEEE Conference on Decision and Control*, Phoenix, AZ, Dec 1999.
- [5] Cooperative Association for Internet Data Analysis, "Internet Measurement Efforts," <http://www.caida.org/Tools/taxonomy.html#InternetMeasurement>
- [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, J.Horowitz, F. Lo Presti, D. Towsley, "Multicast Topology Inference from End-to-End Measurements", *to appear in Proc. of ITC Specialist Seminar on IP Traffic Measurement, Modeling and Management*, Monterey, CA, Sept 2000.
- [8] N.G. Duffield, J.Horowitz, F. Lo Presti and D. Towsley, "Multicast Topology Inference from Measured End-to-End Loss", submitted for publication.
- [9] Felix: Independent Monitoring for Network Survivability. For more information see <ftp://ftp.bellcore.com/pub/mwg/felix/index.html>
- [10] IPMA: Internet Performance Measurement and Analysis. For more information see <http://www.merit.edu/ipma>
- [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] D. Mills, "Network Time Protocol (Version 3): Specification, Implementation and Analysis", *RFC 1305*, Network Information Center, SRI International, Menlo Park, CA, Mar. 1992.
- [13] ns - Network Simulator. For more information see <http://www-mash.cs.berkeley.edu/ns/ns.html>
- [14] S. Paul at al. "Reliable Multicast Transport Protocol (RMTP)", *IEEE JSAC* Vol. 15, No. 3, pp. 407-421, April 1997.
- [15] V. Paxson, J. Mahdavi, A. Adams, M. Mathis, "An Architecture for Large-Scale Internet Measurement," *IEEE Communications*, Vol. 36, No. 8, pp. 48-54, August 1998.
- [16] S. Ratnasamy & S. McCanne, "Inference of Multicast Routing Tree Topologies and Bottleneck Bandwidths using End-to-end Measurements", *Proc. IEEE Infocom '99*, New York, NY (1999)
- [17] M. Schervish, "Theory of Statistics", Springer, New York, 1995.
- [18] Surveyor. For more information see <http://io.advanced.org/surveyor/>